TELEGRAM LISTING

User Protocol Services for Operating/ Configuring the LD-OEM/LD-LRS Laser Measurement System

LD-OEM1000
LD-LRS1000/2100/3100

SICK
Sensor Intelligence.

Software Versions

Device	Function	Version
LD-OEM1000 LD-LRS1000/2100/3100	Firmware	from V 2.1.7

Latest Manual Version

For the latest version of this manual (PDF), see **www.sick.com**.

Contents

Abbreviation

CAN	**C**ontroller **A**rea **N**etwork
CRC	**C**yclic **r**edundancy **c**heck
DID	Transmitter **ID**
DSP	**D**igital **S**ignal **P**rocessor
EDM	**E**lectronic **D**istance **M**eter
LD	**L**adar **D**igital (Ladar = Laser Radar)
LD-LRS	**L**adar **D**igital - **L**onge **R**ange **S**canner
SID	**S**ender **ID**
UPF	**U**ser **P**rotocol **F**rame

Tables

Figures

Notes:

1 Notes on this document

1.1 Purpose

This document shows you how to use and configure (parameterise) the LD-OEM/LD-LRS Laser Measurement Systems by means of a compact command language (User Protocol Services), based on telegrams.

The document contains information on:

- Data communication between the host/driver and laser measurement system
- Configuration by means of telegrams
- Commands/responses in the telegrams
- Troubleshooting

Note From now on, the LD-OEM/LD-LRS Laser Measurement Systems (LD-OEM1000, LD-LRS1000/2100/3100) will simply be referred to as the "LD-OEM/LD-LRS".

1.2 Target audience

This document is aimed at technicians and engineers.

1.3 Information content

This document contains all the information required for communicating with the LD-OEM/LD-LRS by means of telegrams.

The LD-OEM1000 is **mounted, installed electrically and for the communication parameters basic setted** in accordance with the specifications in the *LD-OEM1000 Operating Instructions* (German edition: order no. 8011503, English edition: order no. 8011504).

The LD-LRS1000/2100/3100 is **mounted, installed electrically and for the communication parameters basic setted** in accordance with the specifications in the *LD-LRS1000/2100/3100 Operating Instructions* (German edition: order no. 8011305, English edition: order no. 8011306).

For the basic description of the data communication via the three interface types of the LD-OEM/LD-LRS see the appendix in the operating instructions.

For further information on laser measurement technology, please contact the SICK AG or visit the Sick Web site at **www.sick.com**.

1.4 Symbols

Certain information in this documentation is specially highlighted to draw your attention:

Reference Italics are used to refer to more detailed information elsewhere.

Explanation Explanations provide background information on technical correlations.

Note Provide information on special features.

Default settings Lists the default factory settings for the LD-OEM/LD-LRS.

This symbol indicates that further technical documentation is available for the subject in question.

This symbol indicates important information.

This symbol warns against improper use of the LD-OEM/LD-LRS.

2 Safety information

2.1 Authorized users

To ensure that the LD-OEM/LD-LRS works properly and safely, it must be installed, parameterised, and operated by sufficiently qualified personnel.

The following qualifications are required for commissioning and operation:

- Basic, practical training in electrical engineering
- Knowledge of the relevant safety guidelines
- Knowledge of the hardware and software environment for the relevant application
- Basic data transfer knowledge
- Basic programming knowledge

2.2 Indended use

The LD-OEM/LD-LRS is a non-contact, stand-alone remote (RS 232/422) or networking (CAN, Ethernet) distance measuring system designed for use in industrial environments. The LD-OEM/LD-LRS outputs measured conture values as raw data via a data interface. These data can be queried and evaluated in real time by a host computer with fast data communication by means of application software (driver) provided by the customer.
The four switching outputs “OUT 1“ to “OUT 4“ as well as the two yellow LEDs can be freely assigned to a function by means of the application software.

Implementing the device in any other applications, modifying it in any way, whether during mounting and electrical installation, or making changes to the SICK software will result in an annulment of any warranty claims vis-à-vis SICK AG.

2.3 General safety instructions and protection measures

1. The LD-OEM/LD-LRS uses a class 1 laser (eye-safe).
   Observe the laser safety standards to EN/IEC 60825-1 (latest version).
2. When using electrical systems, observe the standard safety precautions.
   (The LD-OEM/LD-LRS requires 24 V DC).

The LD-OEM/LD-LRS Laser Measurement Systems are not devices for personnel protection in the sense of valid safety standards for machines.

3 Introduction

3.1 General

This document desribes the services, which control the action of the LD-OEM/LD-LRS.

3.1.1 Purpose of the LD-OEM/LD-LRS

The LD-OEM/LD-LRS scans the surroundings by a laser beam and supplies the measured distances. The resulting datas are polar coordinates. The LD-OEM/LD-LRS head rotates clockwise (mathematical negative). The origin of the coordinate system is shown in *Fig. 3-1*.

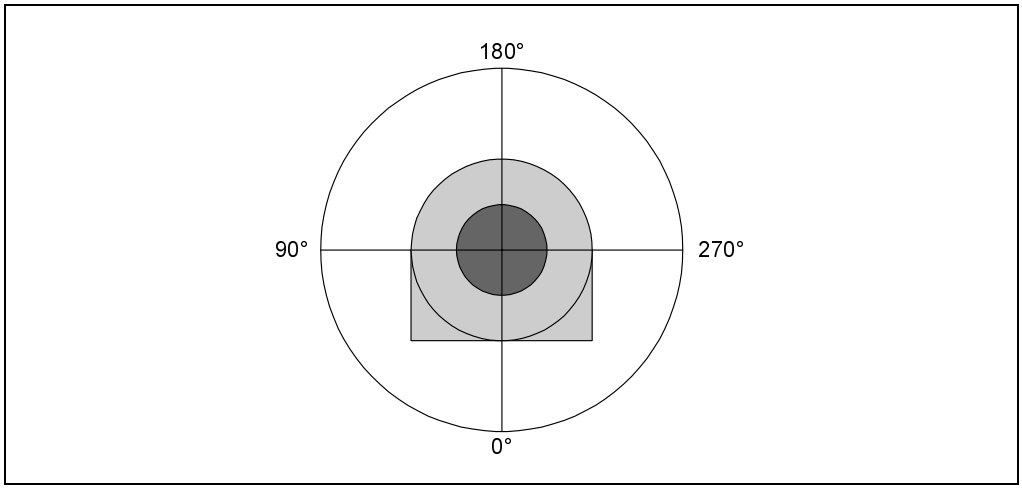

Fig. 3-1: Position of the LD-OEM/LD-LRS in the polar coordinates (top view)

Communication with the sensor is possible via several interfaces:

- RS 232/RS 422
- CAN
- Ethernet
- Internal with application processor/DSP

The LD-OEM1000/LD-LRS1000 owns 4 LEDs:

- The red LED indicates a fatal error.
- The green LED blinks permanently, indicating, that the device "lives".
  When the device emitts laser pulses, the green LED blinks with a higher frequency.
- The programmable function of the two yellow LEDs are accessible by the user.

3.2 Host protocol

The host protocol provides a set of commands to control the LD-OEM/LD-LRS. This set of commands is divided into different service groups.

When the LD-OEM/LD-LRS has received a host protocol command, it answers with a certain response. The LD-OEM/LD-LRS responds when the service has been processed. Generally, this takes less than one second. An exception is the service TRANS_ROTATE, which can take several seconds.

Host protocol commands can be received via all interfaces. The interface dependent description of the protocol can be found in the *LD-OEM1000 Operating Instructions* (German edition: order no. 8011503, English edition: order no. 8011504) respectively in the *LD-LRS1000/2100/3100 Operating Instructions* (German edition: order no. 8011305, English edition: order no. 8011306).

3.2.1 Data type definition

Type	Format	Valid values
signed		
INT16	2 Byte	–32,768 to +32,767
unsigned		
BYTE	1 Byte	0 to 255
WORD	2 Byte	0 to 65535
DWORD	4 Byte	0 to 4,294,967,295

Table 3-1: Data type definition

3.2.2 Service code

The service code is used to determine the kind of service of a host request. It can be seen as the command. Its data format is WORD.

The most significant bit defines whether the code is a request or a response. For a request the most significant bit is set to zero and for a response it is set to one. Usually the LD-OEM/LD-LRS does not request any data. Consequently, requests are sent by the host whereas responses are sent by the LD-OEM/LD-LRS. The remaining of the high byte determines the service group and the low byte defines the service number (*Table 3-2*).

Bit 15	Bit 14 to 8	Bit 7 to 0
Response bit	Service group 0 to 127	Service number 0 to 255

Table 3-2: Service code

4 Available services

The following list gives an overview of the provided services.

Service group 01h: Status Services		
Service No.	Service	See
01h	GET_IDENTIFICATION	*Section 5.1.1, Page 16*
02h	GET_STATUS	*Section 5.1.2, Page 16*
03h	Reserved	-
04h	GET_SIGNAL	*Section 5.1.3, Page 17*
05h	SET_SIGNAL	*Section 5.1.4, Page 17*
06h	REGISTER_APPLICATION	*Section 5.1.5, Page 18*

Table 4-1: Overview: Status Services

Service group 02h: Configuration Services		
Service No.	Service	See
01h	SET_CONFIG	*Section 5.2.1, Page 19*
02h	GET_CONFIG	*Section 5.2.2, Page 21*
03h	SET_SYNC_ABS	*Section 5.2.3, Page 22*
04h	SET_SYNC_REL	*Section 5.2.4, Page 22*
05h	GET_SYNC_CLOCK	*Section 5.2.5, Page 22*
06h	Reserved	-
07h	Reserved	-
08h	Reserved	-
09h	SET_FILTER	*Section 5.2.6, Page 23*
0Ah	SET_FUNCTION	*Section 5.2.7, Page 23*
0Bh	GET_FUNCTION	*Section 5.2.8, Page 28*

Table 4-2: Overview: Configuration Services

Service group 03h: Measurement Services		
Service No.	Service	See
01h	GET_PROFILE	*Section 5.3.2, Page 31*
02h	CANCEL_PROFILE	*Section 5.3.3, Page 32*

Table 4-3: Overview: Measurement Services

Service group 04h: Working Services		
Service No.	Service	See
01h	DO_RESET	*Section 5.4.1, Page 33*
02h	TRANS_IDLE	*Section 5.4.2, Page 33*
03h	TRANS_ROTATE	*Section 5.4.3, Page 34*
04h	TRANS_MEASURE	*Section 5.4.3, Page 34*

Table 4-4: Overview: Working Services

Service group 06h: Interface Routing Services		
Service No.	Service	See
01h	COM_ATTACH	*Section 5.5.1, Page 35*
02h	COM_DETACH	*Section 5.5.2, Page 35*
03h	COM_INIT	*Section 5.5.3, Page 36*
04h	COM_OUTPUT	*Section 5.5.5, Page 38*
05h	COM_DATA	*Section 5.5.7, Page 40*

Table 4-5: Overview: Interface Routing Services

Service group 07h: File Services		
Service No.	Service	See
01h	DIR	*Section 5.6.1, Page 41*
02h	SAVE	*Section 5.6.2, Page 41*
03h	LOAD	*Section 5.6.3, Page 42*
04h	DELETE	*Section 5.6.4, Page 42*

Table 4-6: Overview: File Services

Service group 08h: Monitor Services		
Service No.	Service	See
01h	MONITOR_ENABLE_LOG	*Section 5.7.1, Page 43*
02h	MONITOR_PROFILE_LOG	*Section 5.7.2, Page 43*

Table 4-7: Overview: Monitor Services

Service group 10 to 3Fh: Application Services	
Service No.	Service
Services are depending on application software in application DSP	

Table 4-8: Overview: Application Services

Service group 7Fh: Special Services		
Service No.	Service	See
00h	SERVICE_FAILURE	*Section 5.8.1, Page 44*

Table 4-9: Overview: Special Services

5 Service definitions

5.1 Status Services

5.1.1 GET_IDENTIFICATION

Request command **0101h**:

Description	Information about the LD-OEM/LD-LRS type, firmware and application version	
Parameter	*Type*	*Meaning*
IDENTITEM	WORD	0000h Part number of the sensor (LD-OEM/LD-LRS) 0001h Name of sensor (LD-OEM/LD-LRS) 0002h Version of the sensor (LD-OEM/LD-LRS) 0003h Serial number of the LD-OEM/LD-LRS unit 0004h Serial number of EDM unit 0010h Part number of the firmware 0011h Name of the firmware 0012h Version of the firmware 0020h Part number of the application software 0021h Name of the application software 0022h Version of the application software

Table 5-1: Status Services: GET_IDENTIFICATION (request command)

LD-OEM/LD-LRS response **8101h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
IDENTITEM	WORD[6]	*See request command*
SENSSTAT	DWORD	Status of LD-OEM/LD-LRS

Table 5-2: Response to GET_IDENTIFICATION

The return value of GET_IDENTIFICATION is a string of ASCII characters. Two characters are transmitted within a single WORD.

If the requested IDENTITEM is invalid, the LD-OEM/LD-LRS responses to the IDENTITEM 0000h (part number).

5.1.2 GET_STATUS

Request command **0102h**:

Description	Status query	
Parameter	*Type*	*Meaning*
-	-	-

Table 5-3: Status Services: GET_STATUS (request command)

LD-OEM/LD-LRS response **8102h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
SENSSTAT	DWORD	Status of the sensor (LD-OEM/LD-LRS)

Table 5-4: Response to GET_STATUS

The definition of the SENSTAT value is shown in *Chapter 6.3 Sensor mode values, Page 47*.

5.1.3 GET_SIGNAL

Request command **0104h**:

Description	Reads the value of the switch and LED port	
Parameter	*Type*	*Meaning*
-	-	-

Table 5-5: Status Services: GET_SIGNAL (request command)

LD-OEM/LD-LRS response **8104h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
PORTVAL	WORD	Lower Byte: Port bits Upper Byte: 00h

Table 5-6: Response to GET_SIGNAL

The definition of the 8 port bits of PORTVAL:

Bit	Meaning
0	LED 0 (yellow)
1	LED 1 (yellow)
2	LED 2 (green)
3	LED 3 (red)
4	Switch 0
5	Switch 1
6	Switch 2
7	Switch 3

Table 5-7: Definition of the 8 port bits of PORTVAL

Note A LED or switch is on when the corresponding bit is set to 1.

5.1.4 SET_SIGNAL

Request command **0105h**:

Description	Sets the switches and LEDs	
Parameter	*Type*	*Meaning*
PORTVAL	WORD	Lower byte: Port bits Upper byte: Do not care

Table 5-8: Status Services: SET_SIGNAL (request command)

LD-OEM/LD-LRS response **8105h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
PORTVAL	WORD	Request successful: Lower byte: Port bits Upper byte : 00h Request failed: Lower byte: FFh Upper byte: FFh

Table 5-9: Response to SET_SIGNAL

Note The definition of PORTVAL is given in GET_SIGNAL.

Bit 2 (LED 2, green) and Bit 3 (LED 3, red) are not accessible via the service SET_SIGNAL.

5.1.5 REGISTER_APPLICATION

Request command **0106h**:

Description	Registers the identification datas of the application firmware	
Parameter	*Type*	*Meaning*
APPL_ARTICLE	WORD[6]	Part number of the application software
APPL_NAME	WORD[6]	Name of the application software
APPL_VERSION	WORD[6]	Version of the application software

Table 5-10: Status Services: REGISTER_APPLICATION (request command)

LD-OEM/LD-LRS response **8106h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
RESULT	WORD	0, if successful

Table 5-11: Response to REGISTER_APPLICATION

Note Each parameter in the request command is a byte sequence of 12 ASCII characters.
This command is only accepted, if it was sent by the application board.

5.2 Configuration Services

The configuration services handle the parameter setting and the internal clock of the LD-OEM/LD-LRS. The internal clock is a 16-bit counter that supplies the internal time in milliseconds.

5.2.1 SET_CONFIG

Request command **0201h**:

One time set-up for the application (all parameters are stored permanently in a non-volatible memory).

Description	Sets the configuration	
Parameter	*Type*	*Meaning*
CONFIGITEM	WORD	0000h Reserved 0001h RS 232/RS 422 configuration key 0002h CAN configuration key 0003h Reserved 0005h Ethernet configuration key 0010h Global configuration
CONFIGDATA	WORD[k]	Configuration parameters; depend on configuration key

Table 5-12: Configuration Services: SET_CONFIG (request command)

Configuration parameters for RS 232/422 configuration, CONFIGITEM = 0001h

Parameter	*Meaning*
CONFIGDATA[0]	Baud rate: 0001h 4,800 bd 0002h 9,600 bd 0003h 19,200 bd 0004h 38,400 bd 0005h 57,600 bd 0006h 115,200 bd
CONFIGDATA[1]	**Not supported, always written as 0** 0: No parity 1: Even parity 2: Odd parity
CONFIGDATA[2]	**Not supported, always written as 1** 1: One stop bit 2: Two stop bits
CONFIGDATA[3]	**Not supported, always written as 8** Number of bits per character (1 to 8 bit)

Table 5-13: Configuration Services: Configuration parameters for RS 232/422

Configuration parameters for CAN configuration, CONFIGITEM = 0002h

Parameter	*Meaning*
CONFIGDATA[0]	CAN data transmission rate: 0000h 10 kbit/s 0001h 20 kbit/s 0002h 50 kbit/s 0003h 125 kbit/s 0004h 250 kbit/s 0005h 500 kbit/s 0006h 1 Mbit/s
CONFIGDATA[1]	Basic value of the host CAN identifier
CONFIGDATA[2]	Mask value of the host CAN identifier
CONFIGDATA[3]	Basic value of the LD-OEM/LD-LRS CAN identifier
CONFIGDATA[4]	Broadcast-ID (0000 to 07FFh) FFFFh: Broadcast ID disabled

Table 5-14: Configuration Services: Configuration parameters for CAN

Note The basic value and the mask value of the host CAN identifier determines, which CAN messages the LD-OEM/LD-LRS accepts.

Example:

HostBase = 0x180, HostMask = 0x007, the LD-OEM/LD-LRS accepts all CAN identifiers in the range of 0x180 to 0x187.

In addition, the LD-OEM/LD-LRS accepts the CAN-identifier specified by the parameter Broadcast ID. The CAN identifier of the host sensor results from a logical OR operation between the LD-OEM/LD-LRS sensor ID and the base value of the LD CAN identifier.

Configuration parameters for Ethernet configuration, CONFIGITEM = 0005h

Parameter	*Meaning*	*Example*
CONFIGDATA[0]	IP address: leftmost part	192 = 00C0h
CONFIGDATA[1]	IP address	168 = 00A8h
CONFIGDATA[2]	IP address	1 = 0001h
CONFIGDATA[3]	IP address: rightmost part	11 = 000Bh
CONFIGDATA[4]	Subnet mask: leftmost part	255 = 00FFh
CONFIGDATA[5]	Subnet mask	255 = 00FFh
CONFIGDATA[6]	Subnet mask	255 = 00FFh
CONFIGDATA[7]	Subnet mask: rightmost part	0 = 0000h
CONFIGDATA[8]	Standard gateway: leftmost part	192 = 00C0h
CONFIGDATA[9]	Standard gateway	168 = 00A8h
CONFIGDATA[A]	Standard gateway	1 = 0001h
CONFIGDATA[B]	Standard gateway: rightmost part	1 = 0001h
CONFIGDATA[C]	Node ID	16 = 10h = Sensor ID
CONFIGDATA[D]	TCP/IP Transparent port	

Table 5-15: Configuration Services: Configuration parameters for Ethernet

Configuration parameters for global configuration, CONFIGITEM = 0x10h

Parameter	*Meaning*
CONFIGDATA[0]	LD-OEM/LD-LRS sensor ID: 0000h to 00FEh (1 to 254)
CONFIGDATA[1]	Nominal value motor speed 0005h to 0014h (5 to 20)
CONFIGDATA[2]	Angle step; difference between two laser pulses in 1/16 degrees. The value must be a divisor of 5,760 and greater than 1.

Table 5-16: Configuration Services: Configuration parameters for global configuration

Note All three values must be set at the same time.

LD-OEM/LD-LRS response **8201h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
CONFIGRESULT	WORD	0000h Configuration setting was successful FFFFh Configuration setting was not successful

Table 5-17: Response to SET_CONFIG

5.2.2 GET_CONFIG

Request command **0202h**:

Description	Reads the configuration settings	
Parameter	*Type*	*Meaning*
CONFIGITEM	WORD	Configuration key

Table 5-18: Configuration Services: GET_CONFIG (request command)

LD-OEM/LD-LRS response **8202h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
CONFIGRESULT	WORD	Configuration key. If this value is FFFFh, the requested configuration key was invalid and the CONFIGDATA field is empty
CONFIGDATA	WORD [k]	Configuration parameters; depend on configuration key

Table 5-19: Response to GET_CONFIG

See *Chapter 5.2.1 SET_CONFIG, Page 19* (SET_CONFIG service) for details of configuration key values and parameters.

5.2.3 SET_TIME_ABS

Request command **0203h**:

Description	Sets the internal clock to the time stamp value	
Parameter	*Type*	*Meaning*
SYNCABS	WORD	New value of the internal clock (ms)

Table 5-20: Configuration Services: SET_TIME_ABS (request command)

LD-OEM/LD-LRS response **8203h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
SYNCTIME	WORD	New value of the internal clock (ms)

Table 5-21: Response to SET_TIME_ABS

5.2.4 SET_TIME_REL

Request command **0204h**:

Description	Corrects the internal clock by a defined value	
Parameter	*Type*	*Meaning*
SYNCREL	INT16	Offset value (ms), which corrects the internal clock

Table 5-22: Configuration Services: SET_TIME_REL (request command)

LD-OEM/LD-LRS response **8204h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
SYNCTIME	WORD	New value of the internal clock (ms)

Table 5-23: Response to SET_TIME_REL

5.2.5 GET_SYNC_CLOCK

Request command **0205h**:

Description	Reads the internal time of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
-	-	-

Table 5-24: Configuration Services: GET_SYNC_CLOCK (request command)

LD-OEM/LD-LRS response **8205h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
SYNCTIME	WORD	Actual time of the internal clock (ms)

Table 5-25: Response to GET_SYNC_CLOCK

5.2.6 SET_FILTER

Request command **0209h**:

Description	Sets the filter configuration	
Parameter	*Type*	*Meaning*
FILTERITEM	WORD	0001h Nearfield suppression
FILTERDATA	WORD[k]	Filter parameters, depend on Filteritem

Table 5-26: Configuration Services: SET_FILTER (request command)

Configurations parameters for nearfield suppression, FILTERITEM = 0001h:

Parameter	*Meaning*
FILTERDATA[0]	0000h Nearfield suppression OFF 0001h Nearfield suppression ON

Table 5-27: Configuration Services: configuration parameters for nearfield suppression

LD-OEM/LD-LRS response **8209h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
FILTERITEM	WORD	Filter item: If this values FFFFh, the requested item was invalid and the FILTERDATA field is empty
FILTERDATA	WORD[k]	Filter parameters, depend on Filteritem

Table 5-28: Response to SET_FILTER

5.2.7 SET_FUNCTION

Request command **020Ah**:

Description	Assigns a measurement function to an angle range	
Parameter	*Type*	*Meaning*
SECTORNUM	WORD	Number of the measurement sector. Valid values: 0 to 7 (first sector is always 0)
SECTORFUNC	WORD	Measurement function for the sector. 0: Not initialised (always needed for last sector) 1: No measurement 2: Reserved 3: Normal measurement 4: Reference measurement
SECTORSTOP	WORD	Last angle of the current sector. NOTE: This Angle is given in 1/16 degrees. It must be an integer multiple of the angle step, i.e. of the angle between two laser pulses otherwise this sector is unreachable for the measurement kernel. Example: SECTORSTOP for sector 0 to 180°: (180° minus 1 angle step) x 16. Next sector starts at 180° (SECTORSTOP + 1 angle step)
FLASHFLAG	WORD	1: The sector configuration is written to flash memory Else: The sector configuration stays temporary and is lost after a reset.

Table 5-29: Configuration Services: SET_FUNCTION (request command)

Note

- If SECTORNUM > 7, SECTORNUM will be set to 7.
- If SECTORFUNC has an invalid number, SECTORFUNC will be set to 0 (Not Initialized).
- If SECTORSTOP is greater than a full circle (≥ 5,760 steps), SECTORSTOP will be reduced to an angle less than 5,760 steps.
  The value SECTORSTOP is not checked relating to the fact, if it is an integer multiple of the angle step. The service TRANS_MEASURE will do this.

A Sector (SECTORNUM) is defined by its function (SECTORFUNC) and its end angle (SECTORSTOP). The chosen function is performed until the end angle has been reached. Once the end angle has been reached, the defined function is performed for the last time. After this the measurement kernel switches to the next defined function. Consequently the start angle of a sector is defined by the end angle of the previous sector. The start angle of a sector equals the end angle of the previous sector plus the angle step width.

Sectors must be defined in increasing order starting at zero. If N sectors are configured, the function of sector N+1 must be 0 (*Not initialized*). This is not neccessary, if all 8 sectors are configured. The following pages show 3 different examples.

Example 1:
Supposed the LD-OEM1000/LD-LRS1000 is to be set up for 360° measurement (*Fig. 5-1*). (Not suitable for the LD-LRS2100/3100 due to the restricted scan angle of max. 300°).

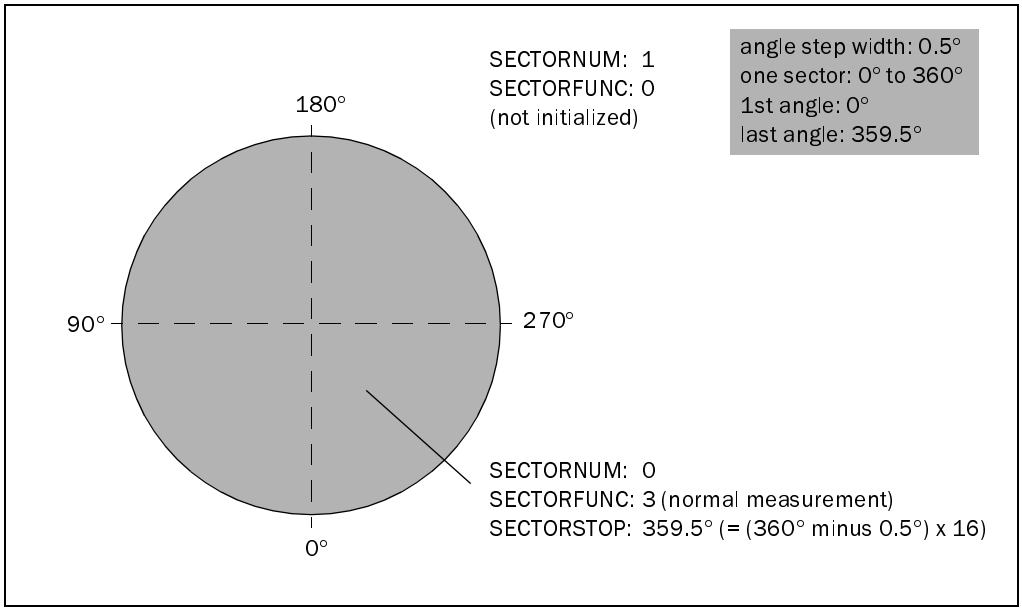

Fig. 5-1: Example 1: Using the whole range as one sector for measurement (top view)

In this case two sectors need to be defined: The first sector is sector 0, which is defined as *Normal measurement*. Assumed the defined stop angle is 0° and the angle step width is 0.5°, the first measurement of the sector is taken at 0.5° and the last measurement is taken at 0°. To get scans starting at 0°, the end angle must be defined as 359.5°(= 5,752/16)°.
The second sector is sector 1 which must be defined as *Not initialized*.

Example 2:
Scans starting at 90° and ending at 180° shall be taken (*Fig. 5-2, Page 26*). The angle step width is 0.5°.
Three sectors need to be defined: one for the measurement area, one for the none measurement area and one as *Not initialized*. To set this configuration, sector 0 is defined as *No measurement* with end angle 89.5° and sector 1 is defined as *Normal measurement* with end angle 179.5°. Sector 2 must be defined as *Not initialized*.

Explanantion The first sector should be defined as *No measurement*, because the transmission of the profile datas starts with sector 0. The transmission of profile data is much more efficient, when the sensor is not scanning.

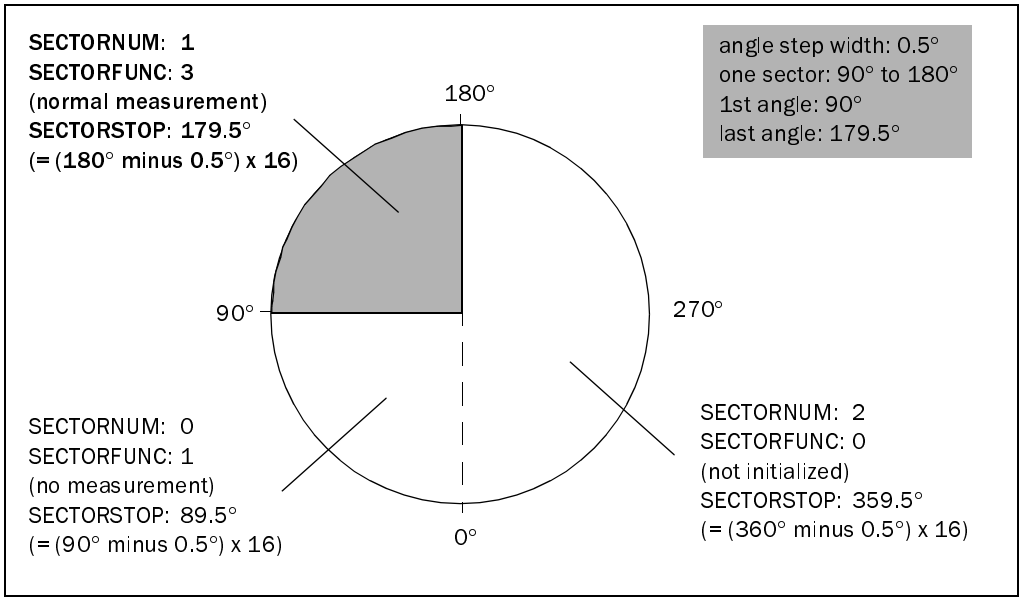

Fig. 5-2: Example 2: Using the sector 90° to 180° for measurement (top view)

***Example 3*:**

The scan should be done in 2 sectors. Scans starting from 125° to 147° and 270° to 360° shall be taken (*Fig. 5-3*). The angle step width is 0.5°. The area from 125° to 147° shall be used for a reference measurement. The nearfield suppression is on.

Five sectors need to be defined, two for the measurement areas, two for the none measurement areas and one as *Not initialized*. To set this configuration, sector 0 is defined as *No measurement* with end angle 124.5° and sector 1 is defined as *Normal measurement* with end angle 146.5°. Sector 2 must be defined as *No measurement* with end angle 269.5°. Sector 3 must be defined as *Normal measurement* with end angle 359.5°. Sector 4 must be defined as *Not initialized*.

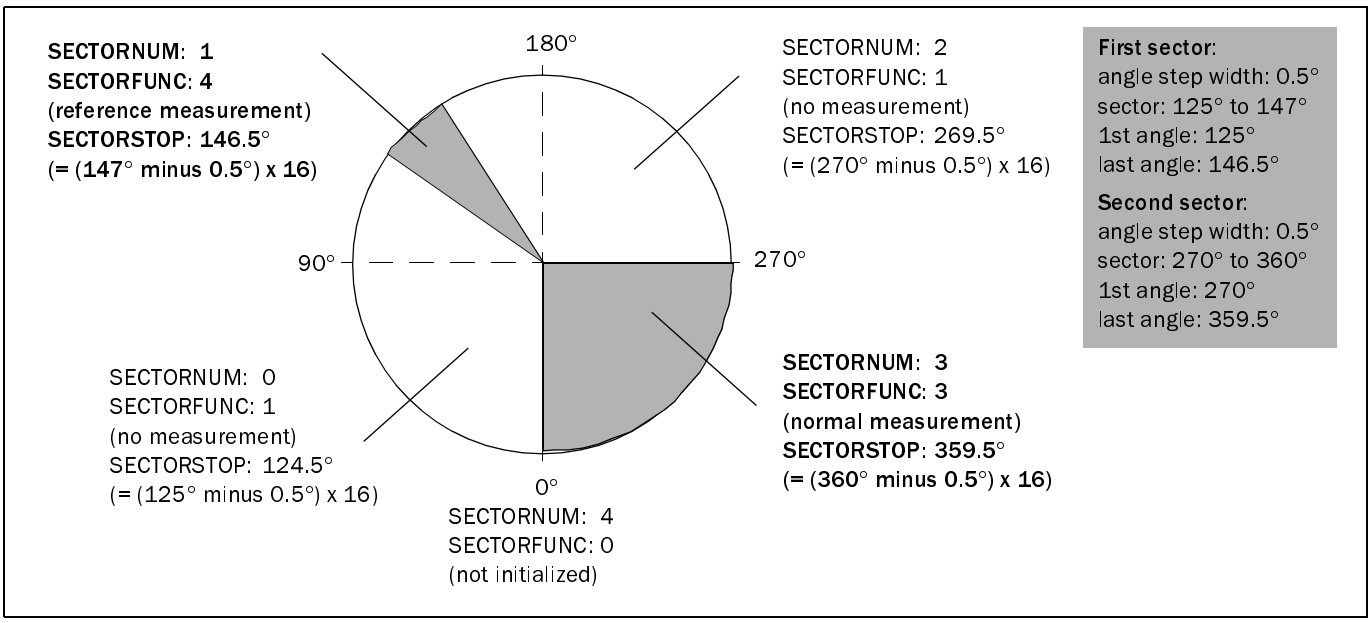

Fig. 5-3: Example 3: Using two sectors (125° to 147°) and (270° to 360°) for measurement (top view)

Explanantion The first sector should be defined as *No measurement*, because the transmission of the profile datas starts with sector 0. The transmission of profile data is much more efficient, when the sensor is not scanning.

LD-OEM/LD-LRS response **820Ah**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
SECTORNUM	WORD	Number of the measurement sector. Valid values: 0 to 7
SECTORFUNC	WORD	Measurement function for the sector
SECTORSTOP	WORD	End angle of the sector

Table 5-30: Response to SET_FUNCTION

The values can differ from the ones which should be set.

The response parameters SECTORNUM, SECTORFUNC and SECTORSTOP are set to 0xFFFFh if SECTORNUM or the request ifself were invalid.

5.2.8 GET_FUNCTION

Request command **020Bh**:

Description	Returns the configuration of the declared setcor	
Parameter	*Type*	*Meaning*
SECTORNUM	WORD	Number of the measurement sector. Valid values: 0 to 7

Table 5-31: Configuration Services: GET_FUNCTION (request command)

LD-OEM/LD-LRS response **820Bh**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
SECTORNUM	WORD	Number of the measurement sector.
SECTORFUNC	WORD	Measurement function for the sector 0: Not initialised 1: No measurement 2: Reserved 3: Normal measurement 4: Reference measurement If activated, the nearfield suppression (see also *Chapter 5.2.6 SET_FILTER, Page 23*) will be not active in this sector. May be used for a reference measurement or other measurements in the close area.
SECTORSTOP	WORD	End angle of the sector

Table 5-32: Response to GET_FUNCTION

The response parameters SECTORNUM, SECTORFUNC and SECTORSTOP are set to 0xFFFFh if SECTORNUM or the request ifself were invalid.

5.3 Measurement Services

5.3.1 Data format

The distance value is represented by a 16-bit binary value with a resolution (step width) of 3.9 mm (1/256 m). The angle is also represented by a 16-bit binary value with a resolution of 1/16°.

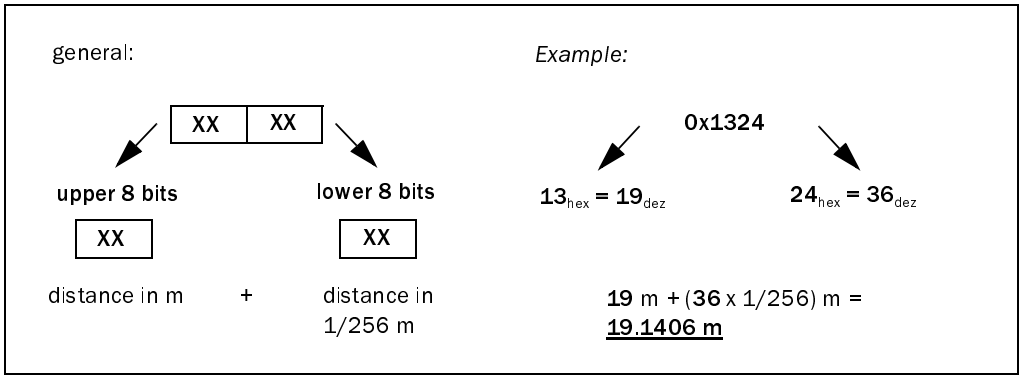

Fig. 5-4: Structure of the 16-bit binary distance values, represented in a WORD

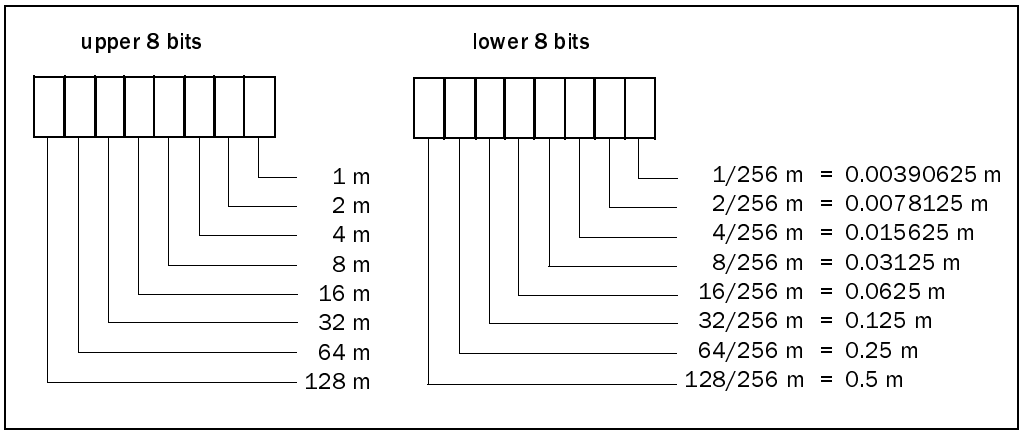

Fig. 5-5: Distance resolution of the upper and lower bit blocks

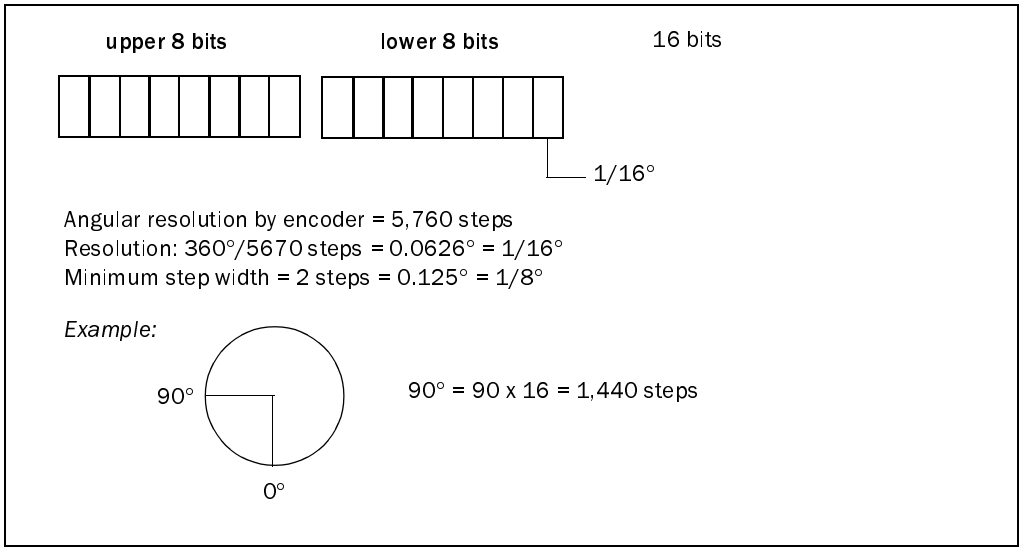

Fig. 5-6: Angular resolution by the encoder, represented in a WORD

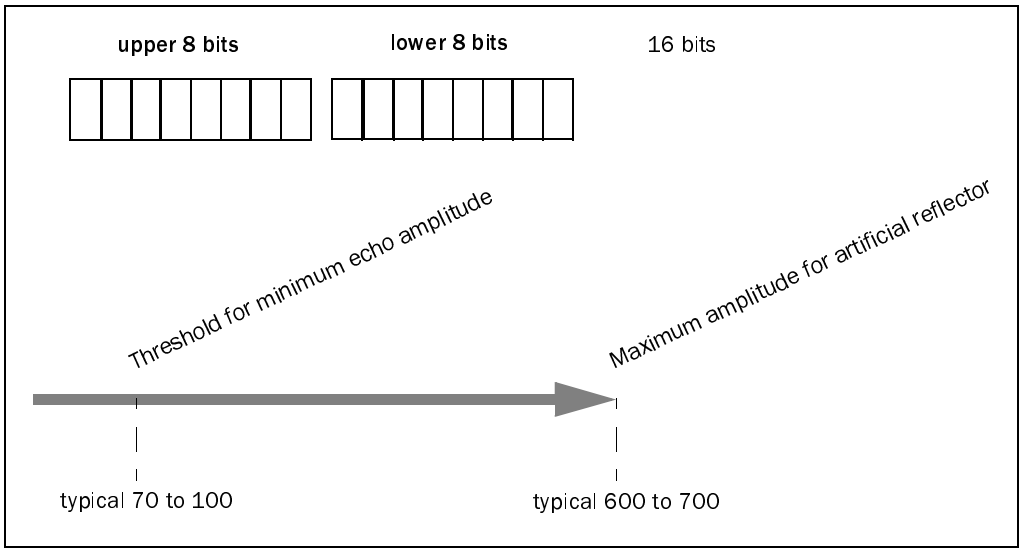

Fig. 5-7: Echo amplitude, represented in a WORD

5.3.2 GET_PROFILE

Request command **0301h**:

Description	Requests n profiles of a defined format	
Parameter	*Type*	*Meaning*
PROFILENUM	WORD	Number of profiles, if it is equals 0 the LD-OEM/LD-LRS sends profiles continuously, until the user sends the CANCEL_PROFILE command
PROFILEFORMAT	WORD	16-bit array

Table 5-33: Measurement Services: GET_PROFILE (request command)

The definition of the 16-bit array PROFILEFORMAT:

Bit	**Meaning**
0	Number of the transmitted profile
1	Profile counter
2	Number of Layer
3	Number of Sector
4	Angle step
5	Number of points of the sector
6	Time stamp when the sector starts
7	Start direction of the sector
8	Measured distances
9	Direction of measured distances
10	Echo amplitudes
11	Time stamp when the sector ends
12	End direction of the sector
13	LD-OEM/LD-LRS mode
14	reserved (always 0)
15	reserved (always 0)

Table 5-34: Definition of the 16-bit array of PROFILEFORMAT

LD-OEM/LD-LRS response **8301h**:

Description		Response of LD-OEM/LD-LRS				
CB*)	*Parameter*	*Type*	*Meaning*			
**)	PROFILEFORMAT	WORD	Format of the following profile			This data is sent with each Scan
**)	PROFILEINFO	WORD	Most significant byte: number of layers (always 1) Least significant byte: number of sectors			
0	PROFILESENT	WORD	Number of the profiles sent to the host. Counts from 0 to PROFILENUM. If PROFILENUM = 0, a 16-bit counter counts continuously.			
1	PROFILECOUNT	WORD	Number of the profiles gathered by the LD-OEM/LD-LRS. 16-bit counter that counts continuously.			
2	LAYERNUM	WORD	Number of the layer (always 0)		This data is sent with each layer	
3	SECTORNUM	WORD	Number of the sector	This data is sent with each sector		
4	DIRSTEP	WORD	Angle step in (degree x 16)			
5	POINTNUM	WORD	Number of points of the sector			
8	TSTART	WORD	Time stamp when the sector starts at the first point in (ms)			
7	STARTDIR	WORD	Start direction of the sector in (degree x 16)			
8	DISTANCE-n	WORD	Measured distance (m x 256)	This data is sent with each distance point		
9	DIRECTION-n	WORD	Direction in (degree x 16)			
10	ECHO-n	WORD	Echo amplitude			
11	TEND	WORD	Time stamp of the last point in (ms)			
12	ENDDIR	WORD	End direction in (degree x 16)			
13	SENSTAT	DWORD	Status of LD-OEM/LD-LRS			

*) CB = Corresponding Bit in PROFILEFORMAT

**) always sent

Table 5-35: Response to GET_PROFILE

The response contains – beside PROFILEFORMAT and PROFILEINFO, which are always sent – only the parameters requested in PROFILEFORMAT.

If the request command is invalid (no request parameters or PROFILEFORMAT = 0x0000h), the command is ignored and exactly one response, containing no parameters, is sent. No profiles are sent. If a measured point is invalid, DISTANCE is set to 0000h.

5.3.3 CANCEL_PROFILE

Request command **0302h**:

Description	Stops the profile output	
Parameter	*Type*	*Meaning*
-	-	-

Table 5-36: Measurement Services: CANCEL_PROFILE (request command)

LD-OEM/LD-LRS response **8302h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*

Table 5-37: Response to CANCEL_PROFILE

Description	Response of LD-OEM/LD-LRS	
SENSTAT	DWORD	Status of LD

Table 5-37: Response to CANCEL_PROFILE

The CANCEL_PROFILE command does not abort the transmission of the currently transmitted profile. Thus, the transmission of the current profile is completed before the CANCEL_PROFILE command terminates transmission of profiles.

5.4 Working Services

5.4.1 DO_RESET

Request command **0401h**:

Description	The LD-OEM/LD-LRS enters a reset sequence	
Parameter	*Type*	*Meaning*
RESETLEVEL	WORD	0000h Reset (CPU reinitialized) 0001h Restart (CPU not reinitialized) 0002h Halt application and enter IDLE state others reserved

Table 5-38: Working Services: DO_RESET (request command)

LD-OEM/LD-LRS response **8401h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
RESETLEVEL	WORD	The same value as in the Request command

Table 5-39: Response to DO_RESET

If RESETLEVEL 0000h or 0001h is requested, the LD-OEM/LD-LRS transmits the response to this command, before it enters the reset sequence.
If RESETLEVEL 0002h or 0010h is requested, the LD-OEM/LD-LRS transmits the response after executing the command. The LD-OEM/LD-LRS does not respond to any request during reset and system initialisation.

5.4.2 TRANS_IDLE

Request command **0402h**:

Description	Sets the LD-OEM/LD-LRS into the IDLE mode: the motor of the rotating prism stops and the laser is switched off	
Parameter	*Type*	*Meaning*
-	-	-

Table 5-40: Working Services: TRANS_IDLE (request command)

LD-OEM/LD-LRS response **8402h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
SENSTAT	DWORD	Status of LD-OEM/LD-LRS

Table 5-41: Response to TRANS_IDLE

5.4.3 TRANS_ROTATE

Request command **0403h**:

Description	Sets the LD-OEM/LD-LRS into the ROTATE mode: the motor starts, when it is off, and rotates with a speed, defined by REV. The laser is switched off.	
Parameter	*Type*	*Meaning*
REV	WORD	0 Scanning frequency corresponds to the configuration parameter 1 to 4 Reserved 5 to 20 Scanning frequency in Hz

Table 5-42: Working Services: TRANS_ROTATE (request command)

LD-OEM/LD-LRS response **8403h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
SENSTAT	DWORD	Status of LD-OEM/LD-LRS

Table 5-43: Response to TRANS_ROTATE

If the request parameter REV is invalid, the LD-OEM/LD-LRS is set to the IDLE mode.

The response to the TRANS_ROTATE command is sent when the rotation frequency is stable or after a constant time of several seconds.

5.4.4 TRANS_MEASURE

Request command **0404h**:

Description	Sets the LD-OEM/LD-LRS into the MEASURE mode: the laser starts with the next revolution; a request for a profile can be started	
Parameter	*Type*	*Meaning*
-	-	-

Table 5-44: Working Services: TRANS_MEASURE (request command)

LD-OEM/LD-LRS response **8404h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
SENSTAT	DWORD	Status of LD-OEM/LD-LRS
ERRORCODE	WORD	0: OK, LD-OEM/LD-LRS measures 1: Maximum laser pulse frequency too high 2: Mean laser pulse frequency too high 3: The sector borders are not configured correctly 4: A sector border is not a whole multiple of the angle step

Table 5-45: Response to TRANS_MEASURE

5.5 Interface routing Services

Interface routing means: a host or application program may attach to one of the available interfaces in order to have total control over communication via the attached interface. Any data received by the attached interface is forwarded to the interface that issued the attach command (the master interface). Data will not be interpreted by the measurement CPU. The master interface may output data via the interfaces it is attached to.

The interface routing comprises the services COM_ATTACH, COM_DETACH, COM_INIT, COM_OUTPUT and COM_DATA. Depending on the attached interface, the services COM_INIT, COM_OUTPUT and COM_DATA differ. So, these three services are defined for each interface.

Interface routing is only possible for RS 232/RS 422 and CAN.

Normally, the sensor transmits only command responses (after receiving the corresponding requests). The COM_DATA service is transmitted by the LD-OEM/LD-LRS without request.

5.5.1 COM_ATTACH

Request command **0601h**:

Description	Attach an interface; any data which will be received by the attached interface is forwarded to the interface that issued this command. The attached interface will be deinitialized.	
Parameter	*Type*	*Meaning*
INTERFACE_ID	WORD	1: RS 232/RS 422 2: CAN

Table 5-46: Interface routing Services: COM_ATTACH (request command)

LD-OEM/LD-LRS response **8601h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
INTERFACE_ID	WORD	*See command request*
RESULT	WORD	0x0000h: COM_ATTACH successful 0xFFFFh: COM_ATTACH not successful

Table 5-47: Response to COM_ATTACH

5.5.2 COM_DETACH

Request command **0602h**:

Description	Detaches an attached interface; the measurement module becomes again the source and destination for data transferred via the detached interface	
Parameter	*Type*	*Meaning*
INTERFACE_ID	WORD	as defined in COM_ATTACH service

Table 5-48: Interface routing Services: COM_DETACH (request command)

LD-OEM/LD-LRS response **8602h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
INTERFACE_ID	WORD	as defined in COM_ATTACH service
RESULT	WORD	0x0000h: COM_DETACH successful 0xFFFFh: COM_DETACH not successful

Table 5-49: Response to COM_DETACH

5.5.3 COM_INIT (RS 232/RS 422)

Request command **0603h**:

Description	Initializes the RS 232/RS 422 interface	
Parameter	*Type*	*Meaning*
INTERFACE_ID	WORD	1 (RS 232/RS 422)
BAUDRATE	WORD	1: 4,800 bd 2: 9,600 bd 3: 19,200 bd 4: 38,400 bd 5: 57,600 bd 6: 115,200 bd
PARITY	WORD	0: No parity 1: Even parity 2: Odd parity
STOPBIT	WORD	1: One stop bit 2: Two stop bits
CHARLENGTH	WORD	Number of bits per character (1 to 8 bit)

Table 5-50: Interface routing Services: COM_INIT (RS 232/RS 422) (request command)

LD-OEM/LD-LRS response **8603h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
INTERFACE_ID	WORD	1 (as defined in COM_ATTACH service)
RESULT	WORD	0x0000h: COM_INIT successful 0xFFFFh: COM_INIT not successful

Table 5-51: Response to COM_INIT (RS 232/RS 422)

5.5.4 COM_INIT (CAN)

Request command **0603h**:

Description	Initializes the attached interface with the baud rate and the filter for the messages to receive	
Parameter	*Type*	*Meaning*
INTERFACE_ID	WORD	2 (CAN)
BAUDRATE	WORD	1: 10 kbit/s 2: 20 kbit/s 3: 50 kbit/s 4: 125 kbit/s 5: 250 kbit/s 6: 500 kbit/s 7: 1 Mbit/s
IDRX_HIGH	WORD	Receive message identifier, upper part. Bit 15: 0: standard identifier 1: extended identifier Bit 14: Reserved Bit 13: Reserved In case of extended identifier, Bit 12 to 0 represent Bit 28 to 16 of the extended identifier
IDRX_LOW	WORD	Receive message identifier, lower part. In case of standard identifier, Bit 10 to 0 represent the standard identifier. In case of extended identifier, Bit 15 to 0 represent Bit 15 to 0 of the extended identifier.
LAM_HIGH	WORD	Local Acceptance Mask, upper part. In case of extended identifier, Bit 12 to 0 represent the mask of Bit 28 to 16 of the extended identifier.
LAM_LOW	WORD	Local Acceptance Mask, lower part. In case of standard identifier, Bit 10 to 0 represent the mask of the standard identifier. In case of extended identifier, Bit 15 to 0 represent the mask of Bit 15 to 0 of the extended identifier.

Table 5-52: Interface routing Services: COM_INIT (CAN) (request command)

Incoming messages are compared with the identifier specified in MSGID_HIGH and MSGID_LOW, omitting the bits specified in LAM_HIGH and LAM_LOW. If the identifier matches, the message is sent to the master interface by the COM_DATA service.

If the acceptance mask is set to 0x00000000h, only messages with exactly the same identifier like IDRX_HIGH/LOW are accepted. An acceptance mask of 0x1FFFFFFFh respective 0x000007FFh allows all messages to be received.

LD-OEM/LD-LRS response **8603h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
INTERFACE_ID	WORD	2 (CAN)
RESULT	WORD	0x0000h: COM_INIT successful 0xFFFFh: COM_INIT not successful

Table 5-53: Response to COM_INIT (CAN)

5.5.5 COM_OUTPUT (RS 232/RS 422)

Request command **0604h**:

Description	Output data to the RS 232/RS 422 interface	
Parameter	*Type*	*Meaning*
INTERFACE_ID	WORD	1 (RS 232/RS 422)
CHARS	WORD[0]	Bits 7 to 0 contain the char. If the defined CHARLENGTH is less than 8 bit, the leading bits will be ignored.
CHARS	...	...

Table 5-54: Interface routing Services: COM_OUTPUT (RS 232/RS 422) (request command)

LD-OEM/LD-LRS response **8604h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
INTERFACE_ID	WORD	1 (as defined in COM_ATTACH service)
RESULT	WORD	0x0000h: COM_OUTPUT successful 0xFFFFh: COM_OUTPUT not successful

Table 5-55: Response to COM_OUTPUT (RS 232/RS 422)

The command response is send immediatly and indicates, that the first character has been written to the output buffer.

5.5.6 COM_OUTPUT (CAN)

Request command **0604h**:

Description	Output data to an attached interface		
Parameter	*Type*	*Meaning*	
INTERFACE_ID	WORD	2 (CAN)	
NO_OF_MSGS	WORD	Number of the following CAN messages (1 to 128). A CAN message is composed of the datas MSGID_HIGH, MSGID_LOW, MSG_LENGTH and DATA[i]. The Number of DATA[ ] words may be less than 4, depending on MSG_LENGTH.	
MSGID_HIGH	WORD	Bit 15: 0: standard identifier 1: extended identifier Bit 14: Reserved Bit 13: Reserved In case of extended identifier, Bit 12 to 0 represent Bit 28 to 16 of the extended identifier	One CAN message
MSGID_LOW	WORD	In case of standard identifier, Bit 10 to 0 represent the standard identifier. In case of extended identifier, Bit 15 to 0 represent Bit 15 to 0 of the extended identifier.	
MSG_LENGTH	WORD	Number of bytes to transmit (0 to 8)	
DATA[0]	WORD	Bit 15 to 8: data byte #1 Bit 7 to 0: data byte #2	
DATA[1]	WORD	Bit 15 to 8: data byte #3 Bit 7 to 0: data byte #4	
DATA[2]	WORD	Bit 15 to 8: data byte #5 Bit 7 to 0: data byte #6	
DATA[3]	WORD	Bit 15 to 8: data byte #7 Bit 7 to 0: data byte #8	
The following messages			

Table 5-56: Interface routing Services: COM_OUTPUT (CAN) (request command)

LD-OEM/LD-LRS response **8604h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
INTERFACE_ID	WORD	2 (CAN)
RESULT	WORD	0x0000h: COM_OUTPUT successful 0xFFFFh: COM_OUTPUT not successful

Table 5-57: Response to COM_OUTPUT (CAN)

The command response is send immediatly and indicates, that the transmission is initiated successfully.

5.5.7 COM_DATA (RS 232/RS 422)

Request command **0605h**:

Description	Forward data that was received on the RS 232/RS 422 interface to the master interface	
Parameter	*Type*	*Meaning*
INTERFACE_ID	WORD	1 (RS 232/RS 422)
TIMESTAMP	WORD	Timestamp of the last character of the following data
CHAR	WORD[0]	One received character, right justified
CHAR	...	...

Table 5-58: Interface routing Services: COM_DATA (RS 232/RS 422) (request command)

Note A buffer (size: 64 characters) is filled with the incoming datas. If the buffer is full or if a baud rate dependent timeout is expired, the COM_DATA service is send.

A BREAK signal forces the LD-OEM/LD-LRS to the IDLE mode. (A DO_RESET with RESETLEVEL = 2 is executed.)

This service is sent by the LD-OEM/LD-LRS. No response.

5.5.8 COM_DATA (CAN)

Request command **0605h**:

Description	Forward data that was received on an attached interface to the master interface	
Parameter	*Type*	*Meaning*
INTERFACE_ID	WORD	2 (CAN)
TIMESTAMP	WORD	Internal time of the LD-OEM/LD-LRS at reception of the CAN message
MSGID_H	WORD	Bit 15: 0: standard identifier 1: extended identifier Bit 14: Reserved Bit 13: Reserved In case of extended identifier, Bit 12 to 0 represent Bit 28 to 16 of the extended identifier.
MSGID_L	WORD	In case of standard identifier, Bit 10 to 0 represent the standard identifier. In case of extended identifier, Bit 15 to 0 represent Bit 15 to 0 of the extended identifier.
MSG_LENGTH	WORD	Number of bytes to transmit (0 to 8)
DATA[0]	WORD	Bit 15 to 8: data byte #1 Bit 7 to 0: data byte #2
DATA[1]	WORD	Bit 15 to 8: data byte #3 Bit 7 to 0: data byte #4
DATA[2]	WORD	Bit 15 to 8: data byte #5 Bit 7 to 0: data byte #6
DATA[3]	WORD	Bit 15 to 8: data byte #7 Bit 7 to 0: data byte #8

Table 5-59: Interface routing Services: COM_OUTPUT (CAN) (request command)

Depending on the length of the received message (MSG_LENGTH), the number of DATA[i] words may be less than 4.

This service is sent by the LD-OEM/LD-LRS. No response.

5.6 File Services

There is a total amount of 32k x 16-bit flash memory available in the LD-OEM/LD-LRS to store application specific configuration data. Data can be stored in a file, which is identificated by a unique ID. Valid File IDs are 0100h to FFFFh.

The following services provide access to this memory resource.

5.6.1 DIR

Request command **0701h**:

Description	Lists the stored files in the flash memory	
Parameter	*Type*	*Meaning*
-	-	-

Table 5-60: File Services: DIR (request command)

LD-OEM/LD-LRS response **8701h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
FREE_MEM	WORD	Available memory space in terms of WORDs
DIR	WORD[n]	File ID

Table 5-61: Response to DIR

The Length n of the DIR parameter depends on the number of stored files, i.e. the number of returned DIR words equals the number of stored files.

5.6.2 SAVE

Request command **0702h**:

Description	Stores data into the flash memory	
Parameter	*Type*	*Meaning*
FILE_ID	WORD	File ID. Valid values: 0100 to FFFFh
DATA_FILE	WORD[n]	Data to be stored
n equals the number of words to be stored		

Table 5-62: File Services: SAVE (request command)

Note To replace an already existent file, this file must be removed by a DELETE command first.

LD-OEM/LD-LRS response **8702h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
FILE_ID	WORD	File ID
SAVE_STAT	WORD	0: File saved successfully 1: Not enough memory 2: File already exists 3: Flash failed 4: Invalid file ID or request incorrect

Table 5-63: Response to SAVE

If a storage failed (DEL_STAT = 3), the file system may be corrupt or the flash memory may be defect.

5.6.3 LOAD

Request command **0703h**:

Description	Recalls a file	
Parameter	*Type*	*Meaning*
FILE_ID	WORD	File ID. Valid values: 0100 to FFFFh

Table 5-64: File Services: LOAD (request command)

LD-OEM/LD-LRS response **8703h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
FILE_ID	WORD	File ID
LOAD_STAT	WORD	0: File Ok 1: File does not exist 2: Request incorrect (In this case, the response parameter File_ID has no meaning)
DATA_FILE	WORD[n]	Data stored
n equals the number of words stored		

Table 5-65: Response to LOAD

5.6.4 DELETE

Request command **0704h**:

Description	Deletes a file	
Parameter	*Type*	*Meaning*
FILE_ID	WORD	File ID. Valid values: 0100 to FFFFh. It is the users responsibility to make sure that there are no duplicate IDs in the system.

Table 5-66: File Services: DELETE (request command)

LD-OEM/LD-LRS response **8704h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
DEL_STAT	WORD	0: File deleted successfully 1: File does not exist 2: Delete failed 3: Invalid File_ID or request incorrect

Table 5-67: Response to DELETE

If a deletion failed (DEL_STAT = 2), the file system may be corrupt or the flash memory is defect.

5.7 Monitor Services

Monitor services are used to get LD-OEM/LD-LRS data (scan profiles) separately from the running application.

Depending on the monitor service; the data to be monitored may be sent without prior request.

5.7.1 MONITOR_RUN

Request command **0801h**:

Description	Enables/disables monitor services. Monitor services are assigned to the interface that issued the enable command and can't be requested via other interfaces until the monitor services are disabled again	
Parameter	*Type*	*Meaning*
MON_RUN	WORD	0: Monitor functions disabled else: Monitor functions enabled

Table 5-68: Monitor Services: MONITOR_RUN (request command)

LD-OEM/LD-LRS response **8801h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
RESULT	WORD	0x0000h: Request successful 0xFFFFh: Request not successful

Table 5-69: Response to MONITOR_RUN

5.7.2 MONITOR_PROFILE_LOG

Request command **0802h**:

Description	Enables/disables profile logging. If profile logging is enabled, profiles are transferred to the monitor application whenever they are sent to the application module respective host application.	
Parameter	*Type*	*Meaning*
MON_PROFILE_LOG	WORD	0: disable profile logging else: enable profile logging

Table 5-70: Monitor Services: MONITOR_PROFILE_LOG (request command)

Profiles are sent using the response service code of the GET_PROFILE service; the profile format is the same as commanded in the GET_PROFILE service request that was issued by the application.

LD-OEM/LD-LRS response **8802h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
RESULT	WORD	0x0000h: Request successful 0xFFFFh: Request not successful

Table 5-71: Response to MONITOR_PROFILE_LOG

5.8 Special Services

5.8.1 SERVICE_FAILURE

If an illegal service request is sent to the LD-OEM/LD-LRS, the device responses with a SERVICE_FAILURE instead of the usual response.

LD-OEM/LD-LRS response **FF00h**:

Description	Response of LD-OEM/LD-LRS	
Parameter	*Type*	*Meaning*
Reserved	DWORD	Reserved (always 0)
SENSTAT	DWORD	LD-OEM/LD-LRS mode value. See *Section 6.3, Page 47*

Table 5-72: Response to an illegal service request

6 Sensor modes

6.1 Description of the sensor modes

6.1.1 IDLE mode

In the IDLE mode the motor and the laser module are off. A TRANS_ROTATE command sets the sensor into ROTATE mode.

The IDLE MODE is also entered when a TRANS_IDLE command has been received. If the LD-OEM/LD-LRS is in the ROTATE mode when the TRANS_IDLE command is received, the device is forced into IDLE mode.

6.1.2 ROTATE mode

The motor of the rotating prism rotates. The rotating frequency is monitored by the LD-OEM/LD-LRS. The laser is off.

A TRANS_IDLE command sets the LD-OEM/LD-LRS into the IDLE mode, a TRANS_MEAS command into the MEASURE mode.

6.1.3 MEASURE mode

The motor of the rotating prism rotates and is monitored by the LD-OEM/LD-LRS. The laser pulses in the defined zones. The GET_PROFILE command is available.

A TRANS_ROTATE command sets the LD-OEM/LD-LRS into the ROTATE mode.

6.2 Availability of the service commands

The following tables show the availability of the service commands. If a service is requested, which is not available in this mode, the LD-OEM/LD-LRS sends the SERVICE_FAILURE response.

Service Group	Service	IDLE mode	ROTATE mode	MEASURE mode
Status Services	GET_IDENTIFICATION	X	X	X
	GET_STATUS	X	X	X
	GET_SIGNAL	X	X	X
	SET_SIGNAL	X	X	X
	REGISTER_APPLICATION	X		

Table 6-1: Availability of the service commands: Status Services

Service Group	Service	IDLE mode	ROTATE mode	MEASURE mode
Configuration Services	SET_CONFIG	X		
	GET_CONFIG	X		
	SET_SYNC_ABS	X	X	
	SET_SYNC_REL	X	X	
	GET_SYNC_CLOCK	X	X	X
	SET_FILTER	X	X	X
	SET_FUNCTION	X	X	
	GET_FUNCTION	X	X	

Table 6-2: Availability of the service commands: Configuration Services

Service Group	Service	IDLE mode	ROTATE mode	MEASURE mode
Measurement Services	GET_PROFILE			X
	CANCEL_PROFILE			X

Table 6-3: Availability of the service commands: Measurement Services

Service Group	Service	IDLE mode	ROTATE mode	MEASURE mode
Working Services	DO_RESET	X	X	X
	TRANS_IDLE	X	X	
	TRANS_ROTATE	X	X	X
	TRANS_MEASURE		X	X

Table 6-4: Availability of the service commands: Working Services

Service Group	Service	IDLE mode	ROTATE mode	MEASURE mode
Interface Routing Services	COM_ATTACH	X		
	COM_DETACH	X		
	COM_INIT	X		
	COM_OUTPUT	X	X	X
	COM_DATA	X	X	X

Table 6-5: Availability of the service commands: Interface Routing Services

Service Group	Service	IDLE mode	ROTATE mode	MEASURE mode
File Services	DIR	X		
	SAVE	X		
	LOAD	X		
	DELETE	X		

Table 6-6: Availability of the service commands: File Services

Service Group	Service	IDLE mode	ROTATE mode	MEASURE mode
Monitor Services	MONITOR_RUN	X	X	X
	MONITOR_PROFLIE_LOG	X	X	X

Table 6-7: Availability of the service commands: Monitor Services

6.3 Sensor mode values

The sensor mode value SENSTAT is a DWORD type. The following table defines the coding of SENSTAT.

Only bits 0 to 7 are valid. Bits 8 to 32 are for future use.

Bit	Number of bits	Function
0 to 3	4	Working mode 1h: IDLE Mode 2h: ROTATE mode 3h: MEASURE mode 4h: ERROR mode 5 to Fh: Reserved
4 to 7	4	Motor mode 0h: Motor ok 1 to 8h: Reserved 9h: Motor spin to high 4h: Motor spin to low Bh: Motor stops or coder error C to Fh: Reserved
8 to 31	24	Reserved

Table 6-8: Sensor mode values

7 Examples for user protocol services

This chapter shows some simple examples how service commands are build on the RS 232/422, CAN and Ethernet interfaces.

7.1 RS 232/422 interface

7.1.1 Short user protocol frame

General settings

Sensor ID: 16 (= 10h), HOST ID: 20 (= 14h)

The intention is to read the status of the LD-OEM/LD-LRS. The corresponding request comand (= service code) is 0102h. The user protocol frame of the service is:

Service code	Data
0102h	(no data)

For serial communication, the frame must be extented to a single interface packet:

SID/DID	LEN	User protocol frame packet		CRC
1410h	0003h	0000h	0102h	CRC 16

The hexadecimal numbers are coded as ASCII caracters, one WORD consists of 4 characters. To transmit the service via the interface, it has to be packed into the following packet:

Byte no.	Meaning	Value			
0	STX	02h			interface packet
1	SID	31h (“1“)			
2	SID	34h (“4“)			
3	DID	31h (“1“)			
4	DID	30h (“0“)			
5	LEN	30h (“0“)			
6	LEN	30h (“0“)			
7	LEN	30h (“0“)			
8	LEN	33h (“3“)			
9	HEADER	30h (“0“)		frame packet	
10	HEADER	30h (“0“)			
11	HEADER	30h (“0“)			
12	HEADER	30h (“0“)			
13	COMMAND	30h (“0“)	user protocol frame		
14	COMMAND	31h (“1“)			
15	COMMAND	30h (“0“)			
16	COMMAND	32h (“2“)			
17	CRC	CRC 16			
18	CRC	CRC 16			
19	CRC	CRC 16			
20	CRC	CRC 16			
21	ETX	03h			

The LD-OEM/LD-LRS answers as followed:

Byte no.	Meaning	Value
0	STX	02h
1	SID	31h (“1“)
2	SID	30h (“0“)
3	DID	31h (“1“)
4	DID	34h (“4“)
5	LEN	30h (“0“)
6	LEN	30h (“0“)
7	LEN	30h (“0“)
8	LEN	35h (“5“)
9	HEADER	30h (“0“)
10	HEADER	30h (“0“)
11	HEADER	30h (“0“)
12	HEADER	30h (“0“)
13	USERSERVICE	38h (“8“)
14	USERSERVICE	31h (“1“)
15	USERSERVICE	30h (“0“)
16	USERSERVICE	32h (“2“)
17	SENSTAT	30h (“0“)
18	SENSTAT	30h (“0“)
19	SENSTAT	30h (“0“)
20	SENSTAT	30h (“0“)
21	SENSTAT	30h (“0“)
22	SENSTAT	30h (“0“)
23	SENSTAT	30h (“0“)
24	SENSTAT	31h (“1“)
25	CRC	CRC 16
26	CRC	CRC 16
27	CRC	CRC 16
28	CRC	CRC 16
29	ETX	03h

The hexadecimal numbers are coded as ASCII caracters, one WORD consists of 4 characters. The decoded service after the transmission via the serial interface is:

SID/DID	LEN	User protocol frame packet				CRC
1014h	0005h	0000h	8102h	0000h	0001h	CRC 16

The answer is SENSTAT = 1 (idle mode).

7.1.2 Long user protocol frame

A long user protocol frame must be divided into several packets. The length of the packets depends on the used interface.

Example: Sensor response to a GET_PROFILE service

Response command	Profile data (500 WORDS)
8301h	Data (0 to 499)

The SID is 10h, because the LD-OEM/LD-LRS sends the response. Thus, the DID is 14h.

The response must be divided into 5 interface packets for the interface. The first packet holds the sequence flag 0xFFFF and the packet ID 5. The following interface packets counts the packet ID down. The CRC value must be calculated for each interface packet.

1st interface packet

SID/DID	LEN	User Protocol Frame Packet			CRC
		Seq. flag	Packet ID	Data	
1014h	007Eh	FFFFh	0005h	Profile Data (0 to 122)	xxxxh
				User protocol frame part 1	

2nd interface packet

SID/DID	LEN	Packet ID	Data	CRC
1014h	007Eh	0004h	Profile Data (123 to 246)	xxxxh
			<- User protocol frame part 2 ->	

3rd interface packet

SID/DID	LEN	Packet ID	Data	CRC
1014h	007Eh	0003h	Profile Data (247 to 370)	xxxxh
			<- User protocol frame part 3 ->	

4th interface packet

SID/DID	LEN	Packet ID	Data	CRC
1014h	007Eh	0002h	Profile Data (371 to 494)	xxxxh
			<- User protocol frame part 4 ->	

5th interface packet

SID/DID	LEN	Packet ID	Data	CRC
1014h	007Eh	0001h	Profile Data (495 to 499)	xxxxh
			<- User protocol frame part 5 ->	

Note All interface packets must be ASCII-coded enclosed by STX and ETX.

7.1.3 Test sequence to check an RS 232/422 connection

General settings

Sensor ID: 16 (= 10h), HOST ID: 20 (= 14h)

The intention is to read the configuration of the RS 232/422 data interface. The user protocol frame of the service GET_CONFIG is:

Meaning	Request command	CONFIGITEM
User protocol frame	0202h	0001h

For serial communication, the frame must be extented to a single interface packet:

SID/DID	LEN	User protocol frame packet			CRC
1410h	0004h	0000h	0202h	0001h	2BBCh

The hexadecimal numbers are coded as ASCII caracters, one WORD consists of 4 characters. To transmit the service via the interface, it has to be packed into the following packet:

Byte no.	Meaning	Value
0	STX	02h
1	SID	31h (“1“)
2	SID	34h (“4“)
3	DID	31h (“1“)
4	DID	30h (“0“)
5	LEN	30h (“0“)
6	LEN	30h (“0“)
7	LEN	30h (“0“)
8	LEN	34h (“4“)
9	HEADER	30h (“0“)
10	HEADER	30h (“0“)
11	HEADER	30h (“0“)
12	HEADER	30h (“0“)
13	COMMAND	30h (“0“)
14	COMMAND	32h (“2“)
15	COMMAND	30h (“0“)
16	COMMAND	32h (“2“)
17	CONFIGITEM	30h (“0“)
18	CONFIGITEM	30h (“0“)
19	CONFIGITEM	30h (“0“)
20	CONFIGITEM	31h (“1“)
21	CRC	32h (“2“)
22	CRC	42h (“B“)
23	CRC	42h (“B“)
24	CRC	43h (“C“)
25	ETX	03h

If the RS 232/422 interface is set on the default settings, the LD-OEM/LD-LRS answers as followed:

Byte no.	Meaning	Value
0	STX	02h
1	SID	31h (“1“)
2	SID	30h (“0“)
3	DID	31h (“1“)
4	DID	34h (“4“)
5	LEN	30h (“0“)
6	LEN	30h (“0“)
7	LEN	30h (“0“)
8	LEN	38h (“8“)
9	HEADER	30h (“0“)
10	HEADER	30h (“0“)
11	HEADER	30h (“0“)
12	HEADER	30h (“0“)
13	USERSERV	38h (“8“)
14	USERSERV	32h (“2“)
15	USERSERV	30h (“0“)
16	USERSERV	32h (“2“)
17	CONFIGITEM	30h (“0“)
18	CONFIGITEM	30h (“0“)
19	CONFIGITEM	30h (“0“)
20	CONFIGITEM	31h (“1“)
21	CONFIGDATA 0	30h (“0“)
22	CONFIGDATA 0	30h (“0“)
23	CONFIGDATA 0	30h (“0“)
24	CONFIGDATA 0	36h (“6“)
25	CONFIGDATA 1	30h (“0“)
26	CONFIGDATA 1	30h (“0“)
27	CONFIGDATA 1	30h (“0“)
28	CONFIGDATA 1	30h (“0“)
29	CONFIGDATA 2	30h (“0“)
30	CONFIGDATA 2	30h (“0“)
31	CONFIGDATA 2	30h (“0“)
32	CONFIGDATA 2	31h (“1“)
33	CONFIGDATA 3	30h (“0“)
34	CONFIGDATA 3	30h (“0“)
35	CONFIGDATA 3	30h (“0“)
36	CONFIGDATA 3	38h (“8“)
37	CRC	32h (“2“)
38	CRC	31h (“1“)
39	CRC	64h (“d“)
40	CRC	37h (“7“)
41	ETX	03h

After decoding, the CONFIGDATA indicates:

CONFIGDATA 0 = 115,200 bd

CONFIGDATA 1 = 0 (no parity)

CONFIGDATA 2 = 1 (1 stop bit)

CONFIGDATA 3 = 8 (8 data bits, fixed)

7.2 CAN interface

7.2.1 Short data structure of CAN interface

The user protocol frame contains only two WORDS, so the command fits into one CAN message.

PACKET ID	DID	User protocol frame	
0000h	0010h	0202h	0001h

The length of the CAN message is found in the DLC field of the CAN message. The SID is coded in the CAN identifier of the host. The CAN identifiers, which will be accepted by the LD-OEM/LD-LRS, are set with the user service SET_CONFIG.

Format of the 1st UPF packet of a packet sequence

PACKET Header				PACKET Data
WORD	WORD	WORD		WORD
Sequence flag	Packet ID	Reserved	DID	
FFFFh	0002h to FFFFh	0	1 to FFh	

Format of the following UPF packets of a packet sequence

PACKET Header	PACKET Data
WORD	max. 3 WORDS
Packet ID	
0001h to FFFEh	

Sequence Flag: Indication of the begin of a set of data blocks.

Packet Header: Identification of a packet. Starts with a number that represents the total amout of packets. The next packet will be packet number minus 1

Exceptional case:

If an user protocol frame entirely fits into a single packet (UPF packet contains max. 2 WORDS data); the sequence flag is omitted and the packet ID has the value 0.

Format of a single packet sequence:

PACKET Header			PACKET Data
WORD	WORD		max. 2 WORDS
Packet ID	Reserved	DID	
0000h	0	1 to FFh	

7.2.2 Long user protocol frame

The example response must be divided into 167 interface packets for the CAN interface. The first packet holds the sequence flag 0xFFFF and the packet ID 167.

The following interface packets counts the packet ID down.

1st UPF packet

Sequence flag	Packet ID	DID	Profile data
FFFFh	00A7h	0014h	Data (0)

2nd UPF packet

Packet ID	Profile data
00A6h	Data (1 to 3)

3rd UPF packet

to

165th UPF packet

166th UPF packet

Packet ID	Profile data
0002h	Data (496 to 498)

167th UPF packet

Packet ID	Profile data
0001h	Data (499)

7.2.3 Test sequence to check a CAN connection

Setup with SOPAS configuration software:

- Sensor ID: 0x10
- Host ID Base: 0x2A8
- Host ID Mask: 0x003
- Sensor ID Base: 0x700
- Broadcast ID: 0x300

Typical stucture of a message from the Host to the LD-OEM/LD-LRS:

CAN Identifier	Data Length Code	User Data = UPF Packet (up to 8 bytes = 4 WORDS)

The identifier of the Host is coded as:

TxHostCanId = TxHostCanBaseId BITOR SID

The Host ID Base is logically linked to the address of the LD-OEM/LD-LRS.

Example:

Host ID Base: 0x2A8
Host ID: 0x00

A Logical OR results in the identifier 0x2A8.

Message Host to LD-OEM/LD-LRS:

Identifier		Length of data block	User data
02h	A8h	06h	

The user data for the request of the status is 01 02h

Packet ID		Reserved	DID from LD-OEM/LD-LRS	Get_Status	
00h	00h	00h	10h	01h	02h

Note No sequence flag, due to exception: only 2 data words

Answer of the LD-OEM/LD-LRS:

Identifier	Length of data block	User data

The identifier of the LD-OEM/LD-LRS is coded as:

TxSensorCanId = TxSensorCanBaseId BITOR SID

That means the sensor ID Base is logically linked to the address of the LD-OEM/LD-LRS.

Example:

Sensor ID Base: 0x700
Sensor ID: 0x10

A Logical OR results in the identifier 0x710

The LD-OEM/LD-LRS answers by 2 packets, because the answer exceeds the maximum of 8 bytes (restricted by the CAN interface):

LD-OEM/LD-LRS to Host: 1st interface packet

Identifier		Length of data block	User data
07h	10h	08h	

Information (USERDATA) of the LD-OEM/LD-LRS according to the telegram listing

Sequence flag		Packet ID		Reserved	DID from Host	Response to Get_Status	
FFh	FFh	00h	02h	00h	00h	81h	02h

LD-OEM/LD-LRS to Host: 2nd interface packet

Identifier		Length of data block	User data
07h	10h	06h	

Information (USERDATA) of the LD-OEM/LD-LRS:

Packet ID		DWORD for status e.g. Idle			
00h	01h	00h	00h	00h	01h

7.3 Ethernet interface

7.3.1 Data structure via Ethernet Interface for realtime measurement

The communication protocol follows the TCP/IP standard. The transferred data are automatically splitted up into multiple packets by the ethernet controller if necessary. For the programmer this is not relevant. On the receiving end the individual packets are automatically collected and put into the correct sequential order.

The connection will be established via the Port 49152.

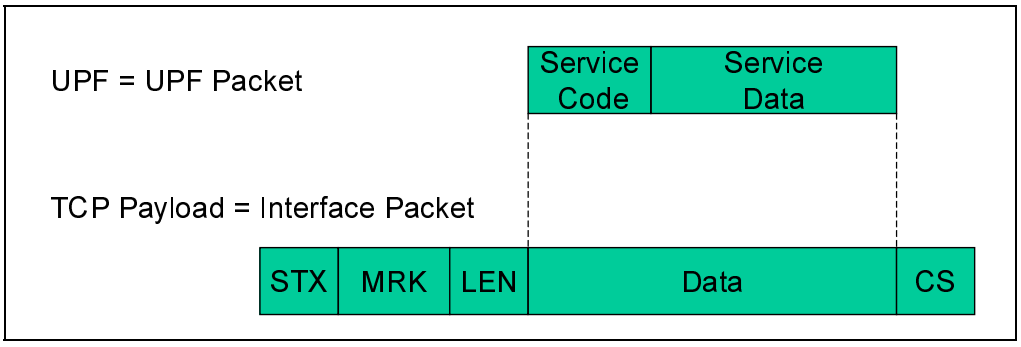

Fig. 7-1: Ethernet: Structure of the UPF packet in the User Service Protocol

STX "Start of Text", will be transferred as a single byte 0x02.

MRK Definition of the transmission format "USP"= 0x55, 0x53,0x50 (3 Bytes).

LEN UPF-Length = the number of the following bytes in <data>, coded as 32 Bit integer (four bytes) without leading sign, the MSB (most significant byte) must be transmitted first of all.

CS Checksum (single byte), calculated as exclusive-or-relation of all bytes contained in "Data".

7.3.2 Test sequence to check a Ethernet connection

Example: Request for status

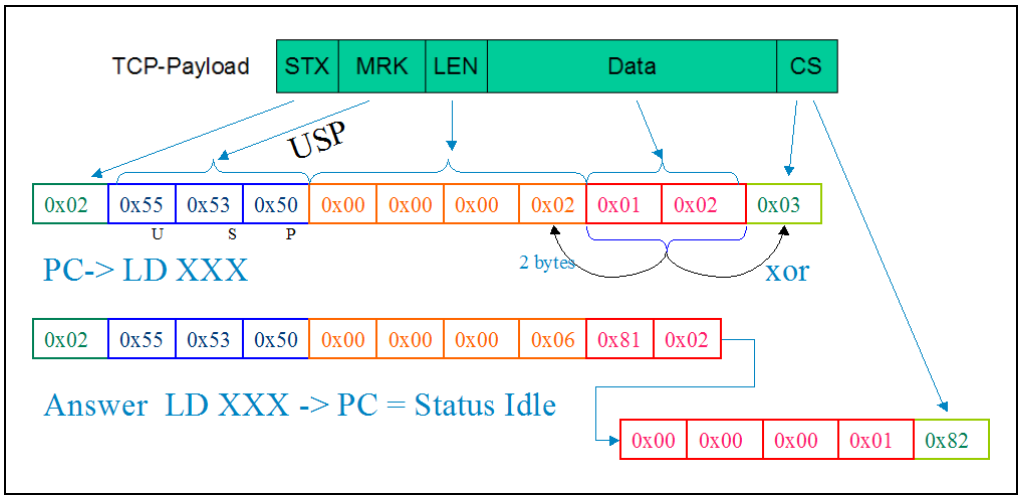

Fig. 7-2: Ethernet: Request by the PC/answer of the LD-OEM/LD-LRS to status request

8 Maximum data throughput

The following tables show the limits and maximum possible data througputs of the three on-board interfaces of the LD-OEM/LD-LRS.

The used unit is 1 Word = 2 bytes = 16 bits.

Assumed is a host computer which is directly connected to the LD-OEM/LD-LRS (peer-to-peer). The host computer must be fast enough to fetch all data from the receiving interface.

8.1 Data throughput at usage of the User Service Protocols

Interface	Format	Net rate of Payload	Overhead Framing USP	Net rate USP Data	Net rate USP Data (Words per Scan rate)
RS 232/422 (115,200 bd)	ASCII	92 kbit/s approx. 11.5 kbyte/s	5 words per 125 data words	approx. 5.5 kbyte/s	550 words at 5 Hz 275 words at 10 Hz
CAN (1Mbit/s)	Binary	577 kbit/s approx. 72 kbyte/s	1 words per 3 data words	approx. 8 kbyte/s	3,800 words at 5 Hz 1,900 words at 10 Hz 950 words at 20 Hz
Ethernet (10 Mbit/s)	Binary	approx. 200 kbyte/s	9 Byte per packet	> 110 Kbyte/s	20,000 words at 5 Hz 10,000 words at 10 Hz 5,000 words at 20 Hz
SPI (internal)	Binary	approx. 1 Mbit/s	–	58 to 110 kbyte/s depending on sector size	> 5,800 words at 5 Hz > 2,900 words at 10 Hz

Table 8-1: Data throughput of LD-OEM/LD-LRS data interfaces if User Service Protocols are used

8.2 Examples of LD-OEM/LD-LRS configurations for real time transmission

The scan area is the sum of all active sectors.

General limits:

- Maximum average pulse rate: 10,800 Hz
- Pulse-to-pulse: 14,400 Hz
- Max. data words per scan: 2,880 (plus header)

8.2.1 RS 232/422 interface

Scanning area	Angular resolution	Scanning frequency	PROFILEFORMAT (example)	Words per scan	Pulses per second (average)	Pulses per second (pulse to pulse)
60°	0.125°	5 Hz	0x39FF = Distance	480	2,400	14,400
60°	0.25°	5 Hz	0x3DFF = Dist + Echo	480	1,200	7,200
120°	0.25°	5 Hz	0x39FF = Distance	480	2,400	7,200
180°	0.5°	5 Hz	0x39FF = Distance	360	1,800	3,600
360°	1°	5 Hz	0x39FF = Distance	360	1,800	1,800
60°	0.25°	10 Hz	0x39FF = Distance	240	2,400	14,400

Table 8-2: Data throughput of the RS 232/422 interface depending of the LD-OEM/LD-LRS configuration

8.2.2 CAN interface

Scanning area	Angular resolution	Scanning frequency	PROFILEFORMAT (example)	Words per scan	Pulses per second (average)	Pulses per second (pulse to pulse)
180°	0.125°	5 Hz	0x3DFF = Dist + Echo	2,880	7,200	14,400
270°	0.125°	5 Hz	0x39FF = Distance	2,160	10,800	14,400
360°	0.1875°	5 Hz	0x39FF = Distance	1,920	9,600	9,600
360°	0.25°	7 Hz	0x39FF = Distance	1,440	10,080	10,080
360°	0.375°	9 Hz	0x3DFF = Dist + Echo	1,920	9,600	9,600
180°	0.25°	10 Hz	0x3DFF = Dist + Echo	1,440	7,200	14,400
270°	0.25°	10 Hz	0x39FF = Distance	1,080	10,800	14,400
360°	0.375°	11 Hz	0x39FF = Distance	960	10,560	10,560
180°	0.5°	20 Hz	0x3DFF = Dist + Echo	720	7,200	14,400
270°	0.5°	20 Hz	0x39FF = Distance	540	10,800	14,400
360°	0.75°	20 Hz	0x3DFF = Dist + Echo	960	9,600	9,600

Table 8-3: Data throughput of the CAN interface depending of the LD-OEM/LD-LRS configuration

8.2.3 Ethernet interface/SPI

Scanning area	Angular resolution	Scanning frequency	PROFILEFORMAT (example)	Words per scan	Pulses per second (average)	Pulses per second (pulse to pulse)
180°	0.125°	5 Hz	0x3DFF = Dist + Echo	2,880	7,200	14,400
270°	0.125°	5 Hz	0x39FF = Distance	2,160	10,800	14,400
360°	0.1875°	5 Hz	0x39FF = Distance	1,920	9,600	9,600
360°	0.25°	7 Hz	0x3DFF = Dist **+ Echo**	2,880	10,080	10,080
360°	0.375°	**10 Hz**	0x3DFF = Dist + Echo	1,920	9,600	9,600
180°	0.25°	10 Hz	0x3DFF = Dist + Echo	1,440	7,200	14,400
270°	0.25°	10 Hz	0x3DFF = Dist **+ Echo**	**2,160**	10,800	14,400
360°	0.375°	11 Hz	0x3DFF = Dist **+ Echo**	**1,920**	10,560	10,560
180°	0.5°	20 Hz	0x3DFF = Dist + Echo	720	7,200	14,400
270°	0.5°	20 Hz	0x3DFF = Dist **+ Echo**	**1,080**	10,800	14,400
360°	0.75°	20 Hz	0x3DFF = Dist + Echo	960	9,600	9,600

Table 8-4: Data throughput of the Ethernet interface/SPI depending of the LD-OEM/LD-LRS configuration

Additional possibilities compared to the CAN interface are marked in **bold**.

9 Error handling

9.1 Fatal errors

Red LED lights, communication possible:

The scan head doesn't rotate, although it should. The LD-OEM/LD-LRS switches the laser automatically off.

Red LED lights, communication not possible:

The internal service management of the LD-OEM/LD-LRS is out of order.

9.2 Service errors

A service request sent with missing or invalid parameters leads to a response, that indicates in the return value, that the request was invalid. For details see the descriptions of the services.

An invalid command request is answered by a SERVICE_FAILURE (FF00h).

Notes:

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 01-270
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 Slowenija
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

SICK AG | Waldkirch/Reute | Germany | www.sick.com

SICK
Sensor Intelligence.