OPERATING INSTRUCTIONS

multiScan165

3D LiDAR sensor

SICK
Sensor Intelligence.

Described product

multiScan165

NOTE

The functional scope of the multiScan depends on the selected configuration. Certain functions are supported or not supported, depending on the configured variant. The operating instructions describe the full functional scope of the multiScan.

Manufacturer

SICK AG
Erwin-Sick-Str. 1
79183 Waldkirch
Germany

Legal information

Original document

This document is an original document of SICK AG.

Contents

1 About this document

1.1 Information on the operating instructions

These operating instructions provide important information on how to use devices from SICK AG.

Prerequisites for safe work are:

- Compliance with all safety notes and handling instructions supplied.
- Compliance with local work safety regulations and general safety regulations for device applications

The operating instructions are intended to be used by qualified personnel and electrical specialists.

NOTE
Read these operating instructions carefully to familiarize yourself with the device and its functions before commencing any work.

The operating instructions are an integral part of the product. Store the instructions in the immediate vicinity of the device so they remain accessible to staff at all times. Should the device be passed on to a third party, these operating instructions should be handed over with it.

These operating instructions do not provide information on the handling and safe operation of the machine or system in which the device is integrated. Information on this can be found in the operating instructions for the machine or system.

1.2 Explanation of symbols

Warnings and important information in this document are labeled with symbols. Signal words introduce the instructions and indicate the extent of the hazard. To avoid accidents, damage, and personal injury, always comply with the instructions and act carefully.

DANGER
... indicates a situation of imminent danger, which will lead to a fatality or serious injuries if not prevented.

WARNING
... indicates a potentially dangerous situation, which may lead to a fatality or serious injuries if not prevented.

CAUTION
... indicates a potentially dangerous situation, which may lead to minor/slight injuries if not prevented.

NOTICE
... indicates a potentially harmful situation, which may lead to material damage if not prevented.

NOTE
... highlights useful tips and recommendations as well as information for efficient and trouble-free operation.

1.3 Further information

More information can be found on the product page. The product page can be accessed via the **SICK Product ID: pid.sick.com/{P/N}/{S/N}**

{P/N} corresponds to the part number of the product (see type label).

{S/N} corresponds to the serial number of the product (see type label).

The following information is available depending on the product:

- Data sheets
- This document in all available language versions
- CAD files and dimensional drawings
- Certificates (e.g., declaration of conformity)
- Other publications
- Software
- Accessories

2 Safety information

2.1 Intended use

The multiScan 3D LiDAR sensor is an intelligent sensor for invisibly detecting objects in areas to be monitored. It has been designed for indoor or outdoor and mobile or stationary use in stand-alone operation.

Typical application areas are, for example, anti-collision monitoring and rear area monitoring in industrial (autonomous) vehicles, person counts at access gates, monitoring of land and buildings, volume monitoring, automated guided vehicle systems for outdoors, robot area, traffic and park management systems.

NOTE
The functional scope of the multiScan depends on the selected configuration. Certain functions are supported or not supported, depending on the configured variant. The operating instructions describe the full functional scope of the multiScan.

SICK AG assumes no liability for losses or damage arising from the use of the product, either directly or indirectly. This applies in particular to use of the product that does not conform to its intended purpose and is not described in this documentation.

2.2 Improper use

Any use outside of the stated areas, in particular use outside of the technical specifications and the requirements for intended use, will be deemed to be incorrect use.

- The device does not constitute a safety component in accordance with the respective applicable safety standards for machines.
- The device must not be used in explosion-hazardous areas, in corrosive environments or under extreme environmental conditions.
- Any use of accessories not specifically approved by SICK AG is at your own risk.

WARNING
Danger due to improper use!

Any improper use can result in dangerous situations.

Therefore, observe the following information:

- Product should be used only in accordance with its intended use.
- All information in the documentation must be strictly observed.
- Shut down the product immediately in case of damage.

2.3 Cybersecurity

Overview

To protect against cybersecurity threats, it is necessary to continuously monitor and maintain a comprehensive cybersecurity concept. A suitable concept consists of organizational, technical, procedural, electronic, and physical levels of defense and considers suitable measures for different types of risks. The measures implemented in this product can only support protection against cybersecurity threats if the product is used as part of such a concept.

You will find further information at www.sick.com/psirt, e.g.:

- General information on cybersecurity
- Contact option for reporting vulnerabilities
- Information on known vulnerabilities (security advisories)

2.4 Limitation of liability

Relevant standards and regulations, the latest technological developments, and our many years of knowledge and experience have all been taken into account when compiling the data and information contained in these operating instructions. The manufacturer accepts no liability for damage caused by:

- Non-adherence to the product documentation (e.g., operating instructions)
- Incorrect use
- Use of untrained staff
- Unauthorized conversions or repair
- Technical modifications
- Use of unauthorized spare parts, consumables, and accessories

2.5 Modifications and conversions

NOTICE

Modifications and conversions to the device may result in unforeseeable dangers.

Interrupting or modifying the device or SICK software will invalidate any warranty claims against SICK AG. This applies in particular to opening the housing, even as part of mounting and electrical installation.

2.6 Requirements for skilled persons and operating personnel

WARNING

Risk of injury due to insufficient training.

Improper handling of the device may result in considerable personal injury and material damage.

- All work must only ever be carried out by the stipulated persons.

The following qualifications are required for various activities:

Table 1: Activities and technical requirements

Activities	Qualification
Mounting, maintenance	▪ Basic practical technical training ▪ Knowledge of the current safety regulations in the workplace
Electrical installation, device replacement	▪ Practical electrical training ▪ Knowledge of current electrical safety regulations ▪ Knowledge of the operation and control of the devices in their particular application
Commissioning, configuration	▪ Basic knowledge of the computer operating system used ▪ Basic knowledge of the design and setup of the described connections and interfaces ▪ Basic knowledge of data transmission
Operation of the device for the particular application	▪ Knowledge of the operation and control of the devices in their particular application ▪ Knowledge of the software and hardware environment for the particular application

2.7 Operational safety and specific hazards

Please observe the safety notes and the warnings listed here and in other sections of this product documentation to reduce the possibility of risks to health and avoid dangerous situations.

CAUTION

Optical radiation: Class 1 Laser Product

The accessible radiation does not pose a danger when viewed directly for up to 100 seconds. It may pose a danger to the eyes and skin in the event of incorrect use.

- Do not open the housing. Opening the housing may increase the level of risk.
- Current national regulations regarding laser protection must be observed.

WARNING

Electrical voltage!

Electrical voltage can cause severe injury or death.

- Work on electrical systems must only be performed by qualified electricians.
- The power supply must be disconnected when attaching and detaching electrical connections.
- The product must only be connected to a voltage supply as set out in the requirements in the operating instructions.
- National and regional regulations must be complied with.
- Safety requirements relating to work on electrical systems must be complied with.

WARNING

Risk of injury and damage caused by potential equalization currents!

Improper grounding can lead to dangerous equipotential bonding currents, which may in turn lead to dangerous voltages on metallic surfaces, such as the housing. Electrical voltage can cause severe injury or death.

- Work on electrical systems must only be performed by qualified electricians.
- Follow the notes in the operating instructions.
- Install the grounding for the product and the system in accordance with national and regional regulations.

3 Product description

3.1 Scope of delivery

Depending on the chosen device version, the scope of delivery of a device will include the following components:

Table 2: Scope of delivery

No. of units	Component	Note
1	Device in the ordered version (complete device or basic device). The functional scope of the device depends on the ordered configuration.	**Complete device:** • Components are mounted at the factory (housing and system plug). **Basic device:** • To mount the housing and system plug yourself, see "Mounting the system plug on the device", page 38. **All devices:** • Without holders and connecting cables
1	Printed safety notes, multilingual	Brief information and general safety notes

The actual scope of delivery may differ for special designs, additional orders or due to the latest technical changes.

3.2 Status indicators

Figure 1: Position of the four status LEDs, front and top view

① LED1
② LED2
③ LED3
④ LED4

LED1/LED3 (color)	LED2/LED4 (color)	Description
● (Red)	● (Red)	Start-up, parameterization, firmware update, correctable error
-	-	Off
✳ (Red)	✳ (Red)	Fatal error
-	● (Green)	On / Ready for operation
-	● (Yellow)	Standby / energy saving
-	✳ (Yellow)	Warning
● (Green)	✳ (Yellow)	Restart after time; input
● (Green)	✳ (Yellow)	Contamination warning
✳ (Red)	✳ (Yellow)	Contamination error
✳ (Green)	✳ (Red)	Alignment mode
● (Green) ● (Yellow) ● (Red)	● (Green) ● (Yellow) ● (Red)	Identifying the device
● (Yellow)	● (Green)	Object detected

● = illuminated; ✳ = flashing

3.3 Type label

Device

Information for identifying the sensor can be found on the bottom of the device.

Figure 2: multiScan type label (example)

① Type code
② Part number
③ Serial number
④ Production date
⑤ Conformity mark/certification mark, protection class, symbol: Observe the operating instructions!
⑥ Production site
⑦ Manufacturer
⑧ Typical power, max. power
⑨ Voltage supply
⑩ MAC address
⑪ Data Matrix code with product data and link to product page
⑫ Web address of product page

Male connector

Information for identifying the male connector is located on the connector.

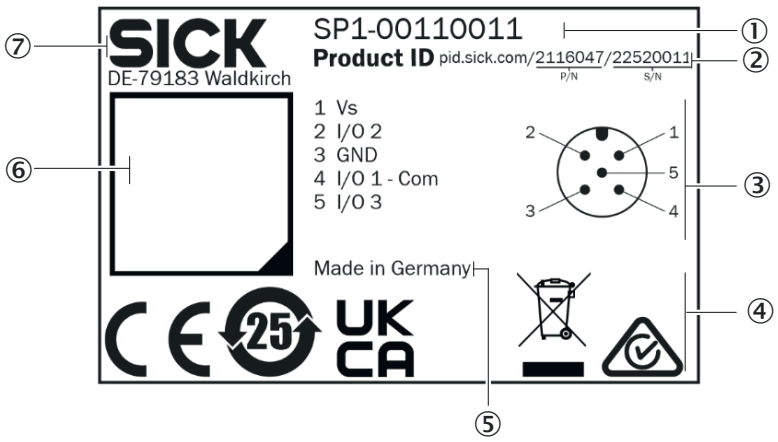


Figure 3: Type label for system plug (example)

① Type code
② Product ID with part number (P/N) and serial number (S/N)
③ Pin assignment or wire colors
④ Conformity mark/certification mark
⑤ Production site
⑥ Data Matrix code with product data and link to product page
⑦ Manufacturer

3.4 Principle of operation

3.4.1 Measurement principle

The device is an opto-electronic LiDAR sensor that scans the outline of its surroundings with the help of laser beams without making contact. The device measures its surroundings in spherical coordinates relative to its measurement origin. This is marked by a circular indentation in the center of the optics cover. If a laser beam strikes an object, the position of that object is determined in terms of distance and angle.

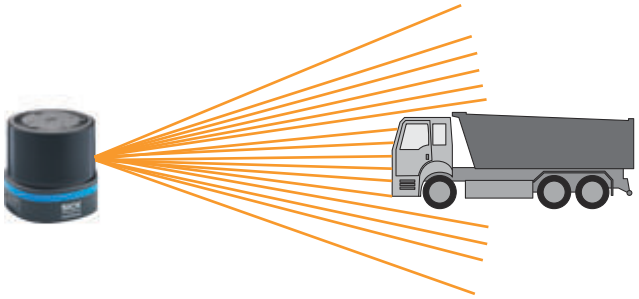


Figure 4: Device with 16 scan layers, side view

Scan layers

The device has 16 scan layers:

- A planar scan layer at an elevation angle of 0°, which was developed for accurate navigation.
- 15 conical scan layers (2 oriented downwards and 13 upwards).

The scan layers are numbered in ascending order - starting with 1 - with descending elevation angle.

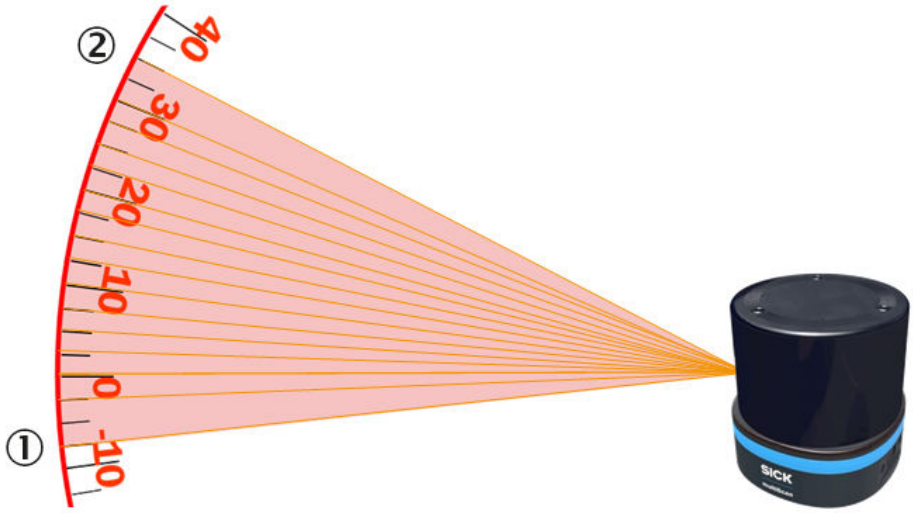


Figure 5: Positions of the 16 scan layers, side view

① Scan layer 1
② Scan layer 16

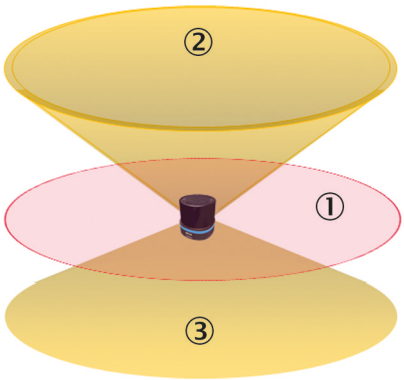


Figure 6: Possible shapes of a scan layer, 3D view

① Scan layer with elevation angle 0°
② Scan layers with an elevation angle < 0°
③ Scan layers with an elevation angle > 0°

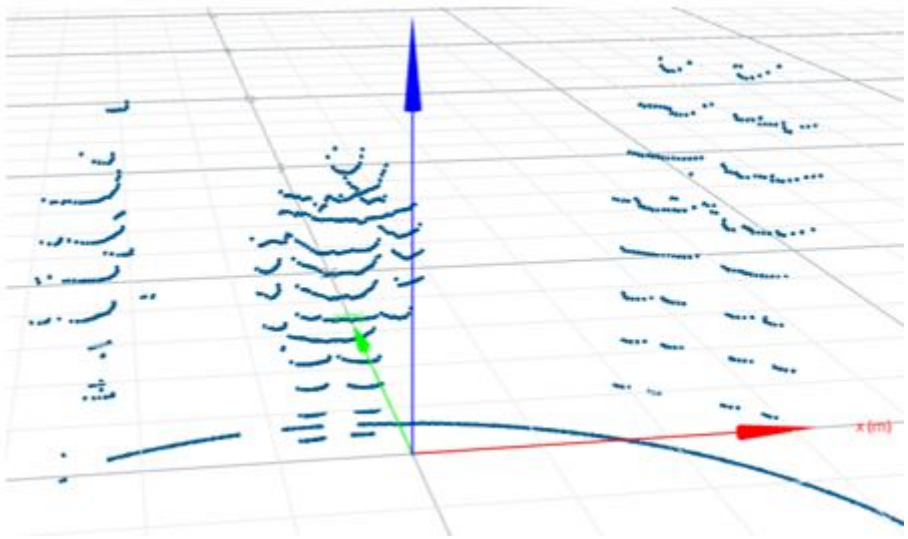


Figure 7: Each of the 16 horizontal point lines is created by one scan layer of the device. In the example, four people have been detected by the device.

Due to their conical shape, all scan layers except for the one with an elevation angle of 0° are bent slightly upwards or downwards when they hit a flat object depending on where the cone opens out. The larger the elevation angle value, the stronger the curvature.

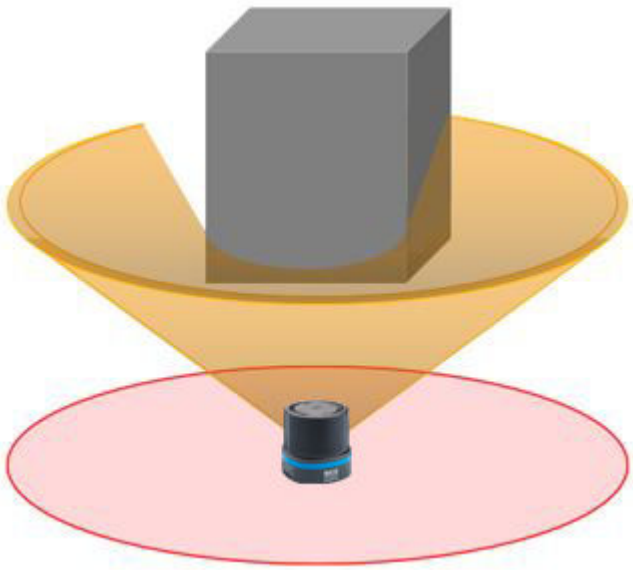


Figure 8: Cone-shaped scan layer as the cause for a bent scan line

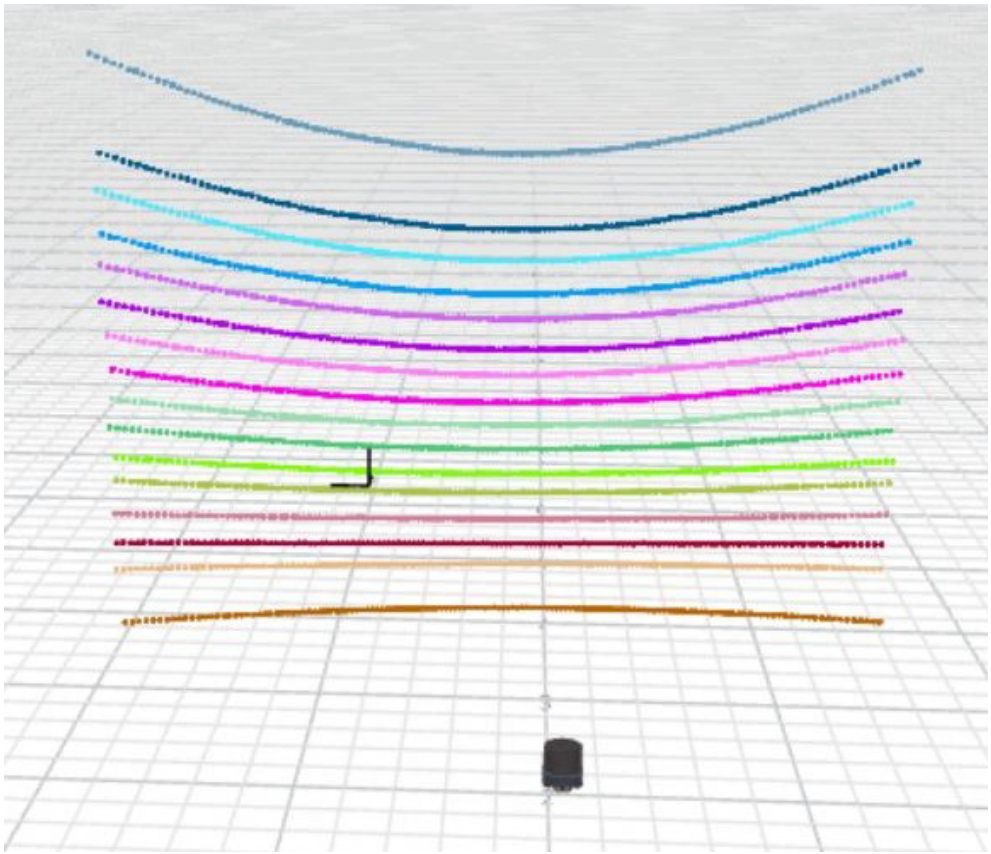


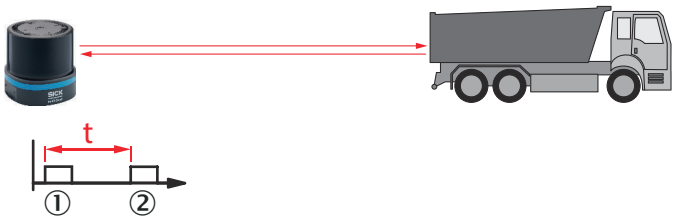
Figure 9: Visualization of a scanned wall as an example of the effect of the cone-shaped scan layer

3.4.2 Distance measurement

The device emits beams pulsed by a laser diode. If the laser beam is reflected by an object, the reflected beam is received by the sensor.

The distance to the object is calculated on the basis of the time that the pulsed light beam requires to be reflected and received by the sensor.

The device uses an in-house technology from SICK. With this measurement process, a measured value is formed statistical evaluation of multiple single pulses. The multi-echo concept evaluates up to 691200 measured values per second. The measured value consists not only of a single time-of-flight measurement, but includes evaluated information from numerous pulses. This ensures a significantly more stable time and distance measurement.



① Emitted pulse

② Receive pulse

3.4.3 Multi-echo analysis

The distance between the device and an object is calculated via the time-of-flight of the emitted pulse. The device can evaluate up to three echo signals for each measuring beam to deliver reliable measurement results, even under adverse ambient conditions.

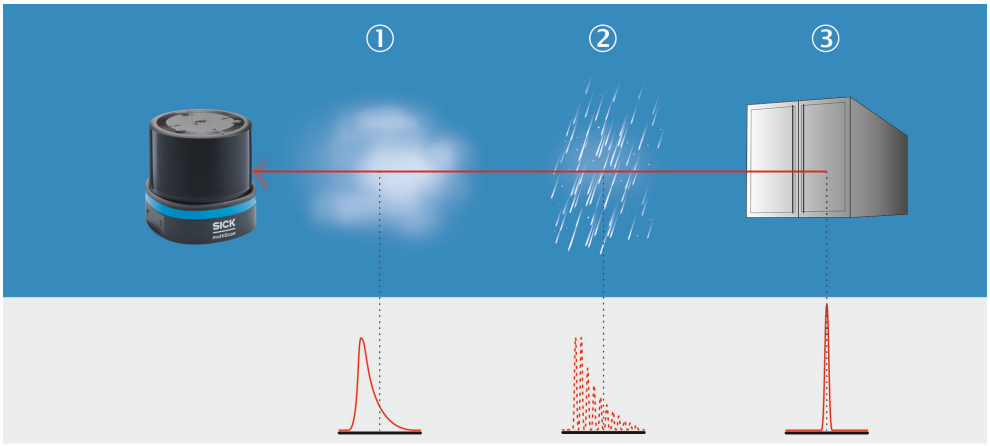


Figure 10: Multi-echo analysis: example industrial application for building management.

① Fog
② Rain
③ Measuring object

3.4.4 Direction measurement

The laser beams are emitted using internally rotating sender-receiver units (SRUs) and scan the surroundings orbitally. The received measured values are assigned to the associated angular cut and thus to the direction.

The scan layers send a set of 24 pulses over an angular range of 0.125° every 0.5°. A measured value is then derived from the received signals for these pulses. This results in an angular resolution of 0.5°.

3.4.5 Multi-layer technology

The multi-layer technology of the device uses 16 scan layers at different vertical angles to compensate for pitch angle, for example when the device is attached to a vehicle. This enables the device to reliably detect an object even, for example, when the vehicle accelerates or brakes.

Figure 11: Multi-plane technology

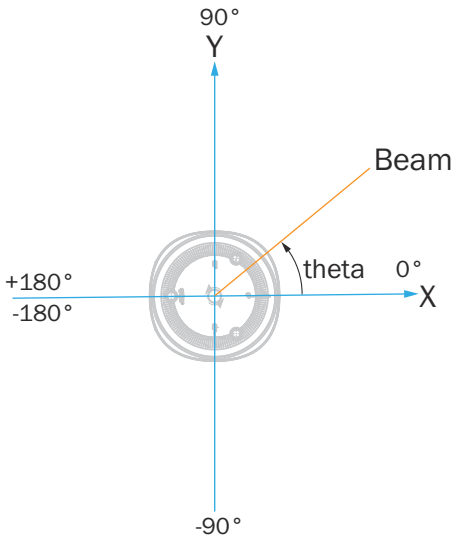
① LiDAR sensor
② Scan layers
③ Object

3.4.6 Coordinate system

Device coordinate system

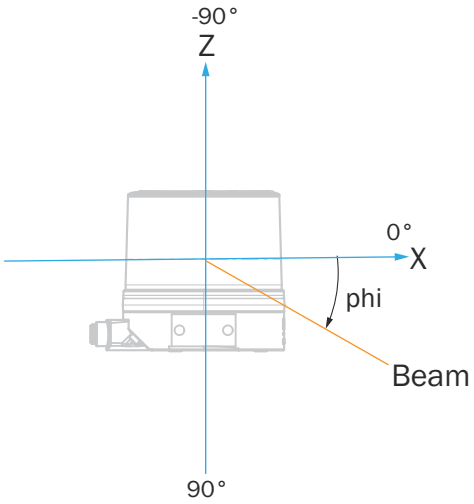
The origin of the device coordinate system (X=0, Y=0, Z=0) is a single point that serves as the origin and reference for all laser beams and the distance measurement of the device. When no translation is applied to the device in the world coordinate system, this point coincides with the origin of the world coordinate system.

The azimuthal (horizontal) angle of a beam is called theta. The beam at zero azimuth angle lies in the middle of the main viewing direction of the device so the scan is symmetrical.



The elevation angle (vertical angle) of a beam is designated phi and is measured relative to the x-y plane:

- Elevation angle < 0 is above the x-y plane, i.e., for positive z-values
- Elevation angle > 0 is below the x-y plane, i.e., for negative z-values



The data is always output in the device coordinate system, not in the world coordinate system.

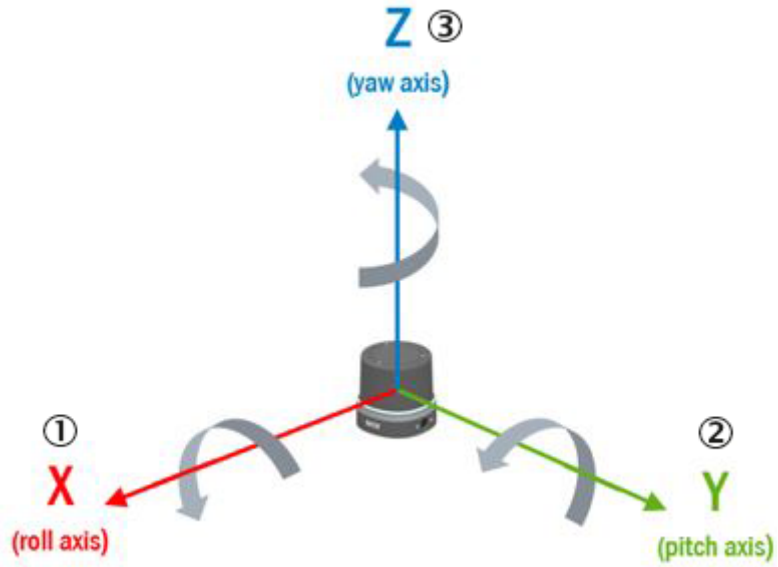
World coordinate system

The world coordinate system is based on the DIN ISO 8855 standard. The directions of rotation are based on a clockwise coordinate system. The orientation of the device in the world coordinate system is specified by means of a yaw angle, pitch angle and roll angle and the position using the Cartesian coordinates x, y and z. If no difference between the world and device coordinate systems is defined, for example to define a specific mounting position, then the origin of both coordinate systems is identical and corresponds to the origin of the device.

The alignment of the device is defined as follows:

- The beam with the azimuthal angle (horizontal) of 0° and the elevation angle (vertical) of 0° points along the x-axis
- The scan layer with an elevation angle of 0° lies on the x-y plane of the coordinate system

- The origin of the device coincides with the world coordinates origin
- The top of the device is oriented towards the increasing z-values of the coordinate system
- The direction of rotation is determined by the orientation of the device, where theta corresponds to the yaw angle and phi to the pitch angle.



① Roll angle
② pitch angle
③ Yaw angle

3.4.7 Filter

By using digital filters to pre-process and optimize the measured distance values, the device can be tailored to the specific requirements of the respective application. This makes it possible to prevent virtually all faults.

The active filter functions affect the outputted measured values. It is not possible to recalculate the original measured values from the filtered output values.

3.4.7.1 Fog filter

The fog filter enables the device to eliminate unwanted echoes at close range. This considerably lowers the probability of false activations at close range in fog.

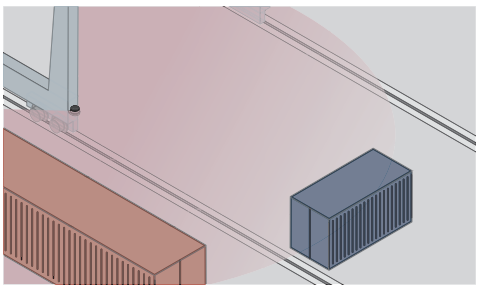


Figure 12: Without the fog filter: objects are difficult to detect through the fog due to reflections.

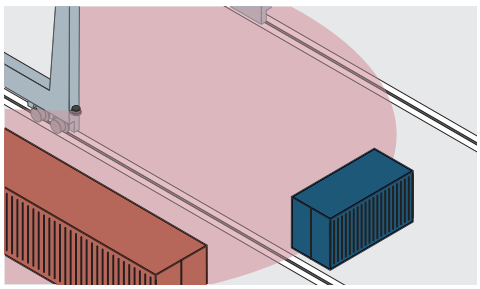


Figure 13: Using the fog filter: objects can be detected reliably because unwanted echoes are screened out.

3.4.7.2 Echo filter

The echo filter screens out unwanted measurement data and signals caused by edge hits, rain, dust, snow and other ambient conditions.

You can set whether the first, the last, or all echoes are output. With the **All echoes** setting, the first, the second and the last echo are output.

The other pulses triggered by undesirable ambient conditions are not taken into account.

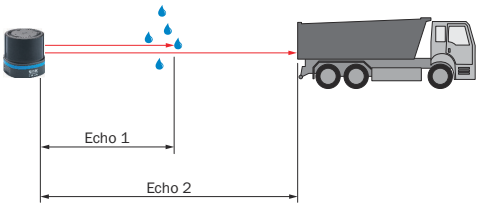


Figure 14: Without the echo filter: The device receives unwanted echoes from ambient conditions such as rain.

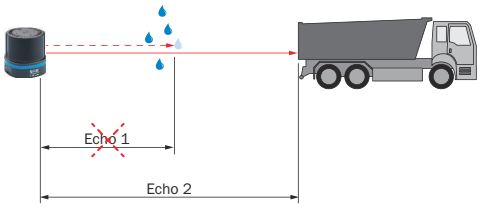


Figure 15: Using the echo filter (setting: last echo): the device screens out unwanted echoes from ambient conditions as per the settings chosen. Measured on white objects, two echoes can be separated at a distance of 1.5 m or greater.

3.4.7.3 Particle filter

The particle filter blanks small, irrelevant reflection pulses in dusty environments and in rain or snow which are caused by dust particles, raindrops, snowflakes or the like.

In doing so, successive scans are continuously evaluated in order to detect static objects.

If the distance between a measured value and its temporal spatial neighbors is greater than a defined threshold value, this measured value is discarded as faulty.

NOTE

If the particle filter is activated, measurement data output or reaction of the field evaluation is delayed by one scan.

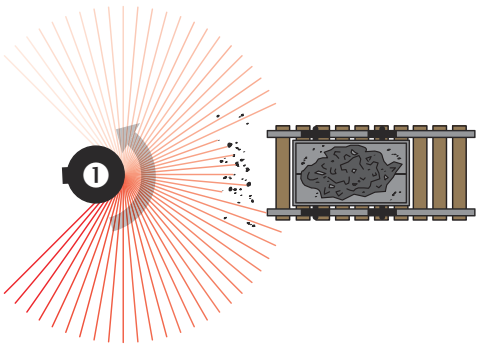


Figure 16: Without the particle filter: Violation of the contour due to dust particles in the vicinity of the object.

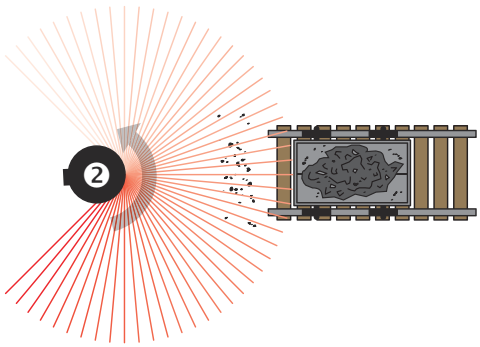


Figure 17: Using the particle filter: The response to dust particles in the detection field is delayed by one scan. Particles can thereby be blanked.

3.4.7.4 Moving average filter

The sliding average filter smooths the distance value. It does this by calculating the arithmetic mean from several scans of the same point. The number of scans can be configured. Each scan layer is filtered separately.

Table 3: Example: Moving average filter over 4 scans

	Angle (distance values in mm)									
Scan	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**...**
1	0	0	1100	1100	1150	1150	1380	1380	0	...
2	0	0	1200	1200	1190	**950**	1500	1500	0	...
3	0	0	1150	**1450**	1200	1200	1450	1450	0	...
4	0	0	1170	1170	1220	1220	1470	**1150**	0	...
1. Output value (scan 1-4)	0	0	1155	1230	1190	1130	1450	1370	0	...
5	0	0	**0**	1110	1150	1150	1380	1380	0	...
2. Output value (scan 2-5)	0	0	1173	1233	1190	1130	1450	1370	0	
6	0	0	1200	1210	1190	**0**	1500	1500	0	...
3. Output value (scan 3-6)	0	0	1173	1235	1190	1190	1450	1370	0	
7	0	**730**	1150	**0**	1200	1200	1450	1450	0	...
4. Output value (4-7)	0	730	1173	1163	1190	1190	1450	1370	0	
...	...	...	...	...	...	...	...	...	...	...

Individual outliers (shown in **bold** in the table) influence the average value.

Once the measured value telegram has been confirmed, the first measured value is not output until after the configured number of scans. Therefore, there is always a time delay equivalent to the number of scans configured for averaging. The scan counter is taken from the latest scan included in the averaging process. Invalid distance values (= 0) are not included in the averaging calculation, so that in these places a smaller number of scans is used in the division calculation.

Based on the scanning frequency of 20 Hz, a measured value is generated every 50 ms. The time delay for data output results from this base value multiplied by the number of averaging operations (e.g., 2 averaging operations = 100 ms, 10 averaging operations = 500 ms).

3.4.7.5 Data reduction filter

A data reduction filter is an algorithm that selects, based on various criteria, the relevant measurement data that should be excluded from the further processing.

3.4.7.5.1 Scan layer filter

The scan layer filter can be used to hide the measurement data of individual scan layers.

3.4.7.5.2 Scan range filter

The scan range filter is used to restrict the horizontal angular range output per scan.

The scan range filter is switched off by default. When it is activated, it can be set to a value between - 180° and + 180°.

The range can be restricted by increasing the start angle or reducing the stop angle. Please note that the angle beam orthogonal to the front screen is defined as 0°, and the direction of rotation of the device is set to counterclockwise.

If a complete range of the output data is outside the angular range, it is not output. If a range is partially within the specified angular range, it is filled with 0 values.

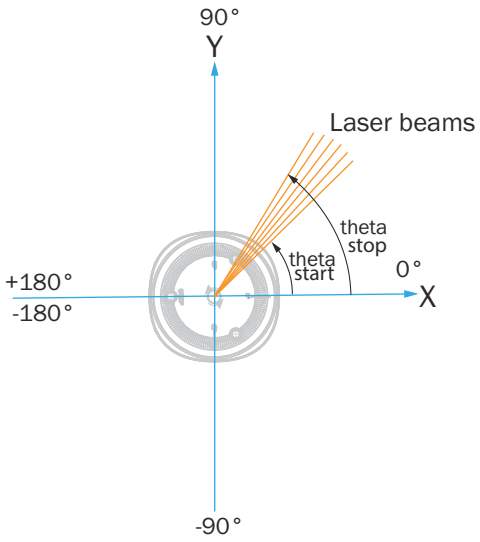


Figure 18: Definition of the thetaStart and thetaStop angle (top view)

3.4.7.5.3 Interval filter

The interval filter reduces the scan output rate by a configurable factor (reduction factor). When the reduction factor is set to three, for example, the output rate is reduced to one third. In this case only every third scan is output.

By combining the moving average filter with an averaging depth d and the interval filter with a reduction factor of d, a “block average filter” can be implemented that outputs the average over the previous scans while at the same time reducing the output rate by a factor d.

3.4.7.6 Region of interest

The region of interest is an algorithm that uses various criteria to select relevant measurement data that are set to a distance value and RSSI value of zero for further processing. Angle values are retained.

3.4.7.6.1 Cubic area filter

When the cubic area filter is activated, it cuts out everything except for the parts of the scan within an axis parallel cuboid. Note that this filter does not reduce the data, it sets the data points outside the cuboid to zero.

The cuboid can be adjusted by setting the minimum and maximum values [mm] for the X, Y and Z axis.

3.4.7.6.2 Distance filter

The distance filter affects the display of a spherical area around the device by limiting the minimum and maximum radial distance that is measured.

The distance filter does not reduce the data, but sets the data points outside the radial distance to zero.

To cut out a spherical area, set the min range to a specific radius [mm] for a very large selected max range.

To keep a spherical area, set the max area to a specific radius [mm] and the min area to 0 [mm].

To keep a hollow sphere, set the min area and max area to a specific radius [mm].

3.4.8 Measurement data output

3.4.8.1 Data formats

The device offers two data output formats: MSGPACK and Compact. Both data formats allow the data to be output segment by segment via UDP.

Both data formats contain information such as serial number and time stamp. While MSGPACK can be integrated easily using existing libraries and is easy to parse, it requires more computing power and bandwidth than the compact data format due to the descriptive names. Compact is more efficient and requires a lower bandwidth. Compared to MSGPACK, however, the compact data format is not descriptive and may require more integration effort.

NOTE
The contents and structure of the two data formats is explained in the Data format description technical information (www.sick.com/8028132).

3.4.8.2 Scan layer address

The actual position of a scan layer can vary. The exact actual vertical angle value of each scan layer can be read from the device.

NOTE
For more information on this, see the Data format description technical information at www.sick.com/8028132.

Table 4: Address assignment method

Scan layer	Measuring module allocation to scan layers	DIN ISO 8855	Physical
1	Measuring module 0	7.2°	-7.2°
2	Measuring module 0	2.5°	-2.5°
3	Measuring module 0	0.0°	-0.0°
4	Measuring module 0	-2.4°	2.4°
5	Measuring module 1	-5.1°	5.1°
6	Measuring module 0	-7.3°	7.3°
7	Measuring module 1	-9.6°	9.6°
8	Measuring module 0	-12.1°	12.1°
9	Measuring module 1	-14.3°	14.3°
10	Measuring module 0	-17.0°	17.0°
11	Measuring module 1	-19.1°	19.1°
12	Measuring module 0	-22.0°	22.0°
13	Measuring module 1	-24.0°	24.0°
14	Measuring module 1	-26.6°	26.6°
15	Measuring module 1	-29.1°	29.1°
16	Measuring module 1	-34.3°	34.3°

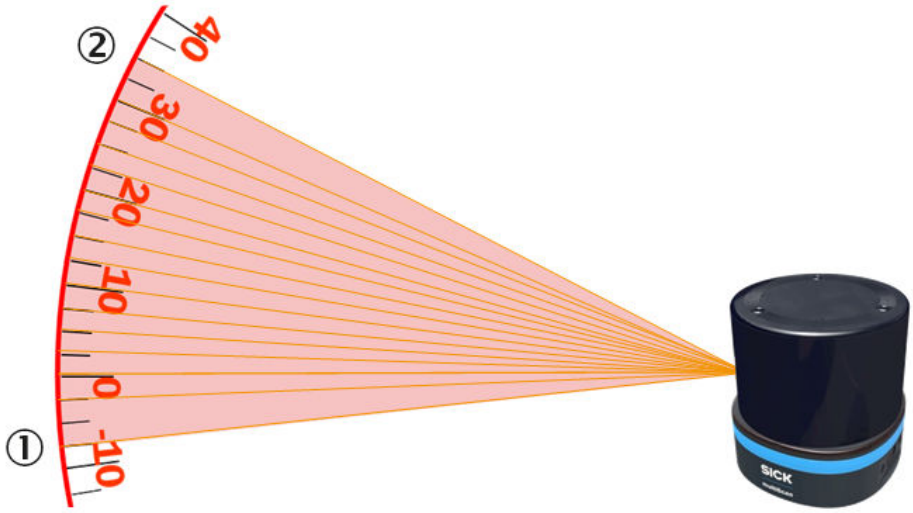


Figure 19: Positions of the 16 scan layers, side view

① Scan layer 1
② Scan layer 16

3.4.8.3 Segmented data output

The device records data over an azimuth range of 360°. The data acquired within a 360° rotation for all scan layers are referred to below as a frame. For data output, a frame is divided into segments each of which contain the data of all scan layers for a smaller azimuth interval.

In this case it is important to note that a segment defines a temporally related portion of the data which does not (necessarily) have to be a spatially related portion of the data, i.e., the azimuth range of each scan layer in a segment may be different.

The following figure shows 12 segments each covering a range of 30°. The segment i is recorded in the time interval [t_(i-1),t_i], which means the motor rotates by 30° during that time interval.

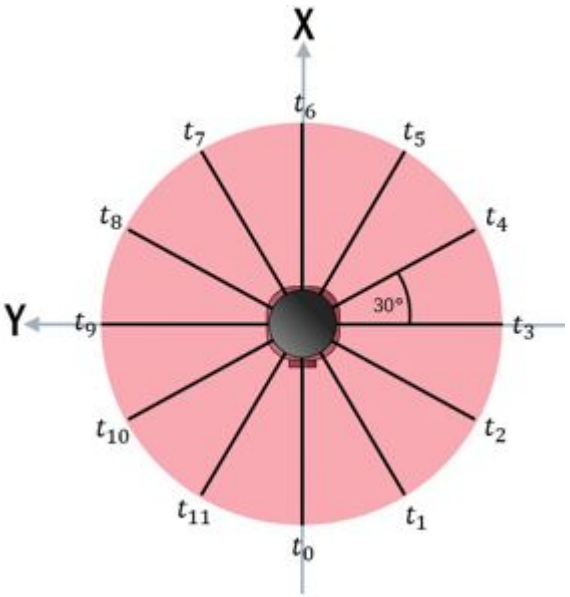


Figure 20: Sequence of segment output. The segment i is from the time t_{i-1} to t_i

3.4.8.4 Data preparation

The yellow bars in the following figures indicate the number of measuring points depending on the resolution over a range of 30°.

Each measuring module generates 8 scan layers. The measuring modules are offset from each other spatially by 180°. In accordance with DIN ISO 8855, the individual layers below the zero line are output as positive and above the zero line as negative.

Raw data (RAW)

Because the device has two measuring modules that point in opposite directions and the sender/receiver units on each measuring module have an azimuth offset, the azimuth range that is recorded for each scan layer in a segment is different.

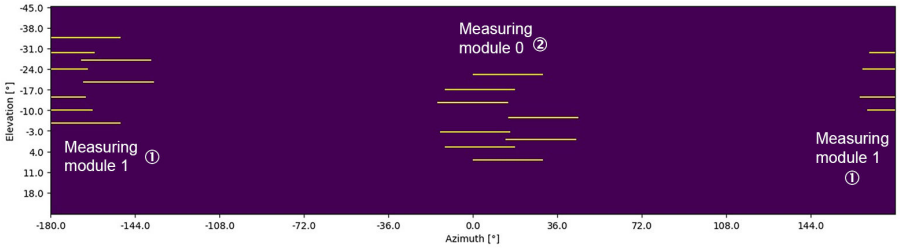


Figure 21: Example azimuth and elevation angles for the data recorded within a segment (RAW)

① Measuring module 1

② Measuring module 0

Rectified data (RECTIFIED)

The data for each scan layer recorded by a measuring module are rearranged so their start angles match (apart from a small deviation that is attributable to the different sending times of the transmitter elements of a measuring module). This rearrangement affects the latency, however, and leads to a delay of at least one time interval.

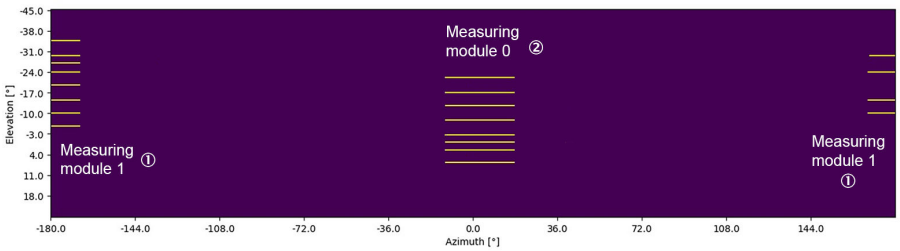


Figure 22: Example azimuth and elevation angles for the data recorded within a segment (RECTIFIED)

① Measuring module 1

② Measuring module 0

3.4.8.5 ROS driver

Suitable drivers for integrating the product into the ROS (Robot Operating System) are available for download on the product page.

The product page can be accessed via the **SICK Product ID**: **pid.sick.com/{P/N}/{S/N}**

{P/N} corresponds to the part number of the product, see type label.

{S/N} corresponds to the serial number of the product, see type label (if indicated).

3.4.9 Object sizes

Due to the angle between two scan layers, the detection of an object depends on its size and distance from the device. The device may not be able to detect the object if the beams go past it rather than hit it. This applies both to the width and also the height of the relevant object.

The transmit spot is never smaller than 13 mm (diameter of the transmitter opening/ lens)

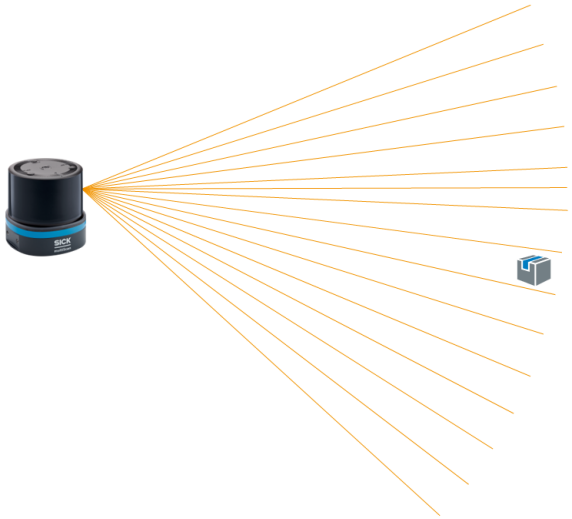


Figure 23: Laser beams pass the object without hitting it (side view)

The smallest object at a desired distance that can still be reliably detected is described by the size obj_sz_{min} [mm] and is defined by the following formula.

The formula is only valid for objects larger than 13 mm.

$$obj\_sz_{min} = Distance \times \tan(obj_{min})$$

$$obj_{min} = \alpha_{spot} + \alpha_{meas} + \alpha_{res}$$

obj_{min} Minimum object angle [°]

α_{spot} Beam spot size [°], here 0.3°

α_{meas} Measuring interval [°]

α_{res} Angular resolution [°], horizontal 0.5°, vertical 0°, see table 5, page 28

Table 5: Typical minimum object size [mm] as a function of distance and angular resolution

Angular resolution [α_{res}]	Horizontal 0.5°	Vertical 2.5°	Vertical 5,0° [1]
Distance [mm]			
100	13	13	13
200	13	13	19
500	13	26	47
1000	16	51	95
2000	32	102	190

Angular resolution [α_{res}]	Horizontal 0.5°	Vertical 2.5°	Vertical 5,0° [1]
Distance [mm]			
3000	48	153	285
5000	81	255	475
10000	161	511	950
15000	242	766	1425
20000	323	1022	1899
25000	404	1277	2374
30000	484	1533	2849

1) between scan layer 1 and 2 / scan layer 15 and 16

NOTE

For reliable measurement, in particular when using the device to output measured values, the laser needs to hit the object with multiple beams. An object should therefore be larger than the minimum object size.

3.4.10 Impact of object surfaces on the measurement

Reflection

Most surfaces produce a diffuse reflection of the laser beam in all directions. The structure (smooth or rough), shape (flat or curved), and color (light or dark) of the surface determine how well the laser beam is reflected.

On very rough surfaces, a large proportion of the energy is lost due to absorption. Curved surfaces produce a higher diffusion. Dark surfaces reflect the laser beam worse than light ones (brilliant white plaster reflects approx. 100% of the light, while black foam rubber reflects approx. 2.4%). The aforementioned surface characteristics can reduce the scanning range of the device, in particular for surfaces with low remission values.

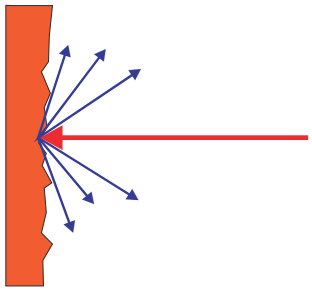


Figure 24: Reflection of light on the surface of the object

Angle of reflection

The angle of reflection corresponds to the angle of incidence. If the laser beam hits a surface at right angles, the energy is optimally reflected. If the laser beam hits a surface at an oblique angle, energy and range are lost accordingly.

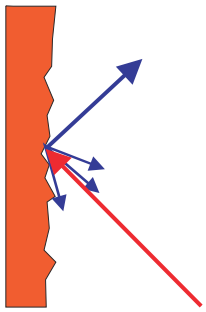


Figure 25: Angle of reflection

Retroreflection

If the reflective energy is greater than 100%, the beam is not reflected diffusely in all directions; instead it is reflected in a targeted way (retroreflection). Thus a large part of the emitted energy can be received by the laser distance measurer. Plastic reflectors (cat's eyes), reflective tape, and triple prisms have these properties.

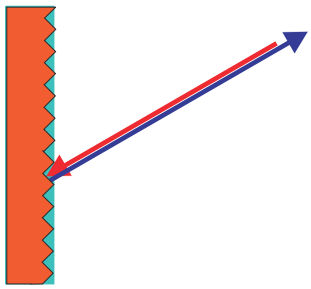


Figure 26: Retroreflection

Reflective surfaces

The laser beam is almost completely deflected on reflective surfaces. This means that an object hit by the deflected beam may be detected instead of the reflective surface.

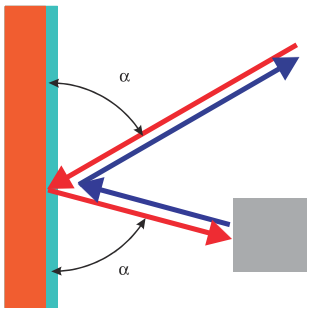


Figure 27: Specular surfaces

Small objects

Objects that are smaller than the diameter of the laser beam cannot reflect the laser light's full energy. The portion of the light beam that does not reach the object is lost. If all of the light reflected to the sensor is insufficient, the object may not be detected.

The portion of the light that does not reach the front object can be reflected by a larger object in the background. If all of the light reflected to the sensor is sufficient, this object is detected. This can lead to a corruption of the measured value.

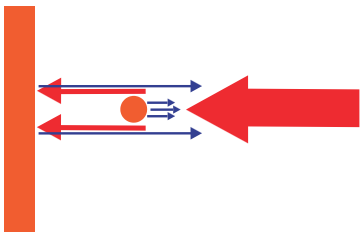


Figure 28: Object smaller than the laser beam diameter

3.4.11 Scanning range

The scanning range of the device depends on the remission of the object to be detected. The better a surface reflects the incident beam back to the device, the greater the scanning range of the device.

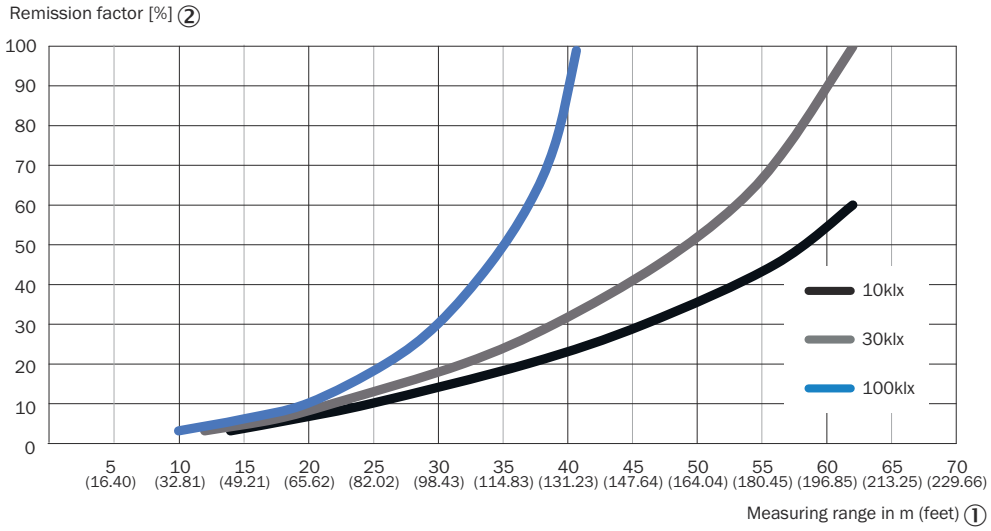


Figure 29: Scanning range as a function of the remission factor for various ambient light influences (no filter activated; no influence from fog, rain or dust)

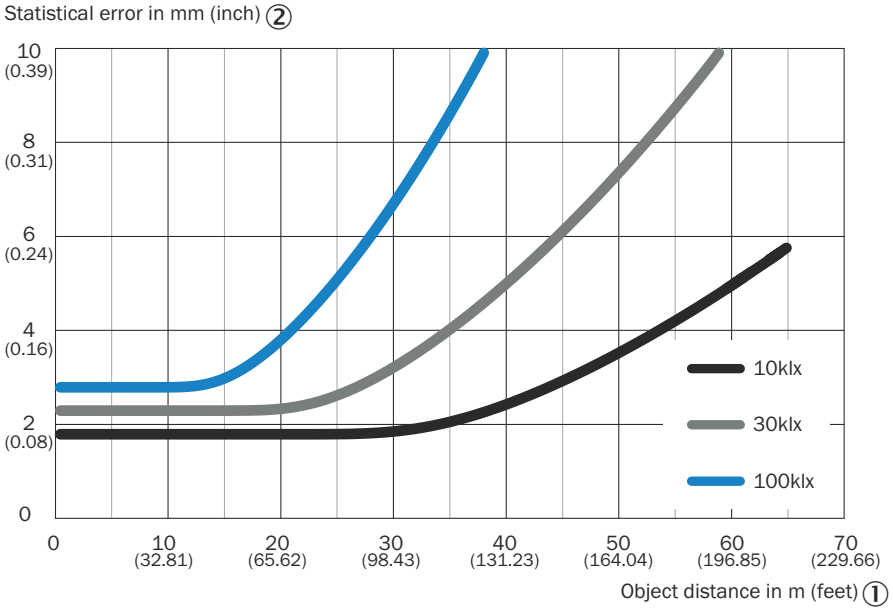


*Figure 30: Statistical error for **white objects** as a function of object distance for various ambient light influences*

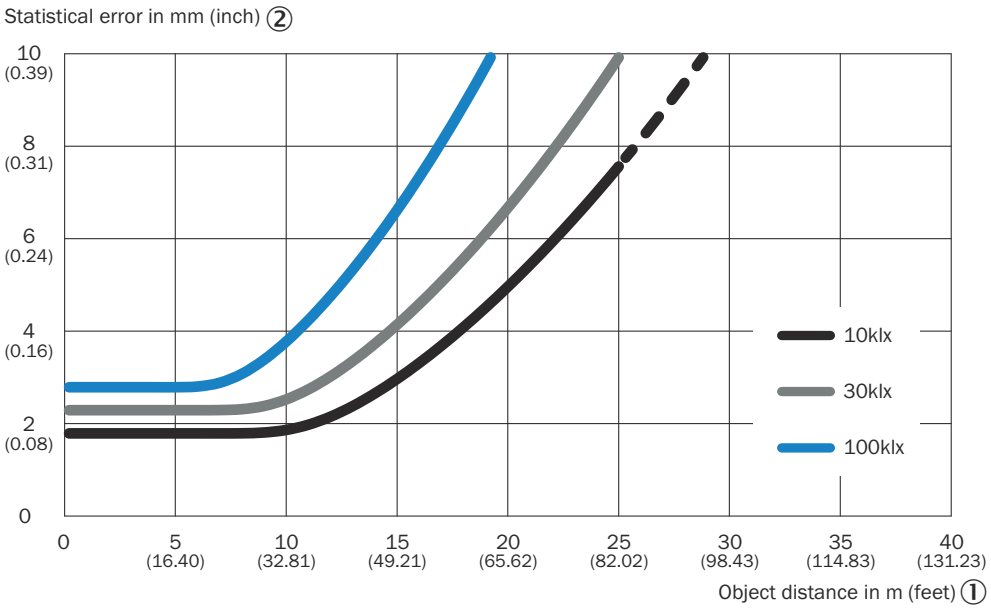


*Figure 31: Statistical error for **black objects** as a function of object distance for various ambient light influences*

3.4.12 RSSI values

RSSI (Received Signal Strength Indicator) is the measure of the signal strength that the device receives. This value is calculated for every measurement. The device therefore provides, for every echo signal, an associated RSSI value for the signal strength.

The value 0 (zero) means that the received energy was too low to produce a valid measured value and also represents the lowest possible RSSI value. An RSSI value of 1 represents the highest possible measured value. A linear scaling is applied between the values 0 and 1 using a resolution specific to the data format (see "Data formats", page 25).

If the RSSI value is 0, then no distance measurement is possible. There can be two reasons for this:

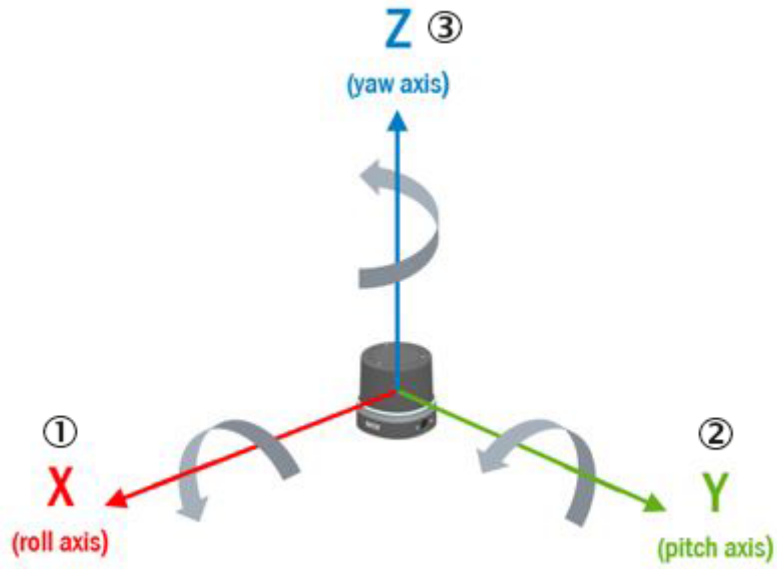
- The target object lies outside the sensing range.
- The target object has an extremely low remission.

Please note that white paper can have very similar values as a reflector at a short distance.

The RSSI values are sensor-specific, relative values that can vary slightly between different devices and during the service life of the device.

3.4.13 Inertial measuring unit (IMU)

The device is equipped with an inertial measuring unit (IMU). This can be used to identify vibrations and movements of the device. The IMU can output accelerations in X, Y and Z as well as the position angle in yaw, pitch and roll. The orientation of the IMU data is based on the coordinate system of the device.



① Roll angle
② pitch angle
③ Yaw angle

3.4.14 Contamination indication

The device has an optics cover to protect it. This optics cover can get dirty. Contamination reduces the energy emitted and received by the laser beam. As a result, scanned objects appear to have a lower remission factor than they actually have and, from a certain degree of contamination, it will no longer be possible to perform measurements.

The contamination is constantly measured by a separate system during operation. A contamination warning is output first for the different degrees of contamination. If the optics cover is not cleaned and contamination increases, then a contamination error is output. These thresholds can be individually adjusted.

The device supports a contamination indication over the full 360°. 12 sectors are monitored independently. When a scan configuration of, for example, -144° to +144° is selected, sectors without measurement outside the configured viewing range are set to deactivated status. Each sector can be individually enabled for the contamination indication.

You can select different settings, depending on the application in which the device is used.

Warning/error output

- **All sectors contaminated**: If all sectors have the same or a higher value, the device status “Contamination warning” or “Contamination error” is displayed.
- **One sector contaminated**: At least one sector must have a level of “Warning” or “Error” for a contamination warning or contamination error to be displayed.
- **No output**: Warning and error device status display is deactivated. The contamination measurement for the individual sectors continues to be performed but has no effect on the device status.

Sensitivity: Low, medium, high: Threshold for triggering contamination warnings and errors. The parameter makes it possible to tailor the display to the specific requirements of the application.

Response time: This can be used to define how quickly the contamination should result in an error or warning.

Default setting: In the default setting, the contamination display is deactivated. The monitoring of individual sectors is active and can be seen for each sector in the user interface and the command interface to assist with integrating the device into the application.

Strategy and evaluation are deactivated so no device warnings or errors are triggered. All associated parameters can be permanently changed.

NOTE

- The cleaner the application environment is, the lower you can set the contamination indication sensitivity. If a high precision of the measured values is required, the contamination indication must be set to the most sensitive level.
- Sectors that are not relevant should be deactivated to ensure a higher availability.

Contamination warnings and contamination errors are indicated on the display elements of the device see "Status indicators", page 11.

3.4.15 Field evaluation

The device uses the integrated field evaluation to evaluate the fields within its scan area. You can use the field evaluation, for example, to implement systems for collision protection, object protection or access monitoring.

Up to 48 fields can be defined. The number of allowed simultaneously active fields can change depending on the configuration that has been set. For a typical configuration (echo filter: last echo, no data preparation, IMU or particle filter active, web interface closed), 20 fields can be active simultaneously. Under real-life ambient conditions, it may also be possible to have more simultaneously active fields.

Limitations of simultaneously active fields:

- More active fields results in a greater computing effort and therefore a higher system load.
- Only the measuring points located within an active field result in an additional system load.
- A too high system load can lead to performance problems or system errors.
- The system load can be reduced by deactivating other functions, for example GUI, filter, and measurement data output, which may allow more fields to be active at the same time.
- We recommend testing the exact application requirements.

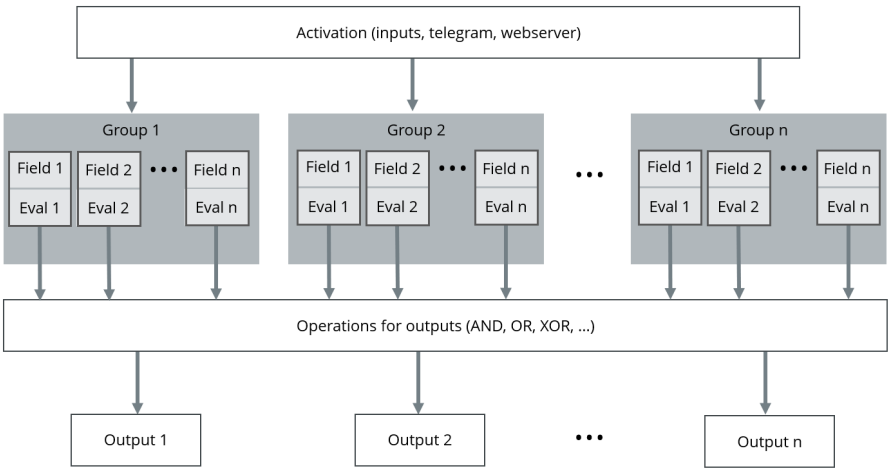


Figure 32: Principle of field evaluation

Multiple groups can be created, depending on application requirements. A group can be switched to active either by switching inputs of the device, telegrams, or permanently. A group can comprise several fields, each of which can be configured for the particular situation via its own evaluation.

The results of the evaluation can be linked to a switching output. It is also possible to link the respective evaluation to other signals using logical operands (AND, OR, XOR, ...).

For more information, see the online help in SOPASair see "Overview", page 48.

4 Transport and storage

4.1 Transport

NOTICE
Damage due to improper transport!

- The product must be packaged with protection against shock and damp.
- Recommendation: Use the original packaging.
- Note the symbols on the packaging.
- Do not remove packaging until immediately before you start mounting.

4.2 Unpacking

- To protect the device against condensation, allow it to equilibrate with the ambient temperature before unpacking if necessary.
- Handle the device with care and protect it from mechanical damage.

4.3 Transport inspection

Immediately upon receipt in Goods-in, check the delivery for completeness and for any damage that may have occurred in transit. In the case of transit damage that is visible externally, proceed as follows:

- Do not accept the delivery or only do so conditionally.
- Note the scope of damage on the transport documents or on the transport company's delivery note.
- File a complaint.

NOTE
Complaints regarding defects should be filed as soon as these are detected. Damage claims are only valid before the applicable complaint deadlines.

4.4 Storage

- Do not store outdoors.
- Store in a place protected from moisture and dust.
- Recommendation: Use the original packaging.
- Do not expose to any aggressive substances.
- Protect from sunlight.
- Avoid mechanical shocks.
- Storage temperature: see "Technical data", page 56.
- Relative humidity: see "Technical data", page 56.
- For storage periods of longer than 3 months, check the general condition of all components and packaging on a regular basis.

5 Mounting

5.1 Mounting instructions

- Observe the technical data.
- Protect the sensor from direct sunlight.
- To prevent condensation, avoid exposing the device to rapid changes in temperature.
- The mounting site has to be designed for the weight of the device.
- The device can be mounted in any position.
- It should be mounted so that it is exposed to as little shock and vibration as possible. Optional mounting accessories are available, see "Accessories", page 63.
- Regularly check the tightness of the fixing screws.
- Do not mount the device on or directly in front of a bright metallic surface or other reflective surface, since reflections can falsify the measurements.
- Avoid having shiny or reflective surfaces in the scanning range, e.g., stainless steel, aluminum, glass, reflectors, or surfaces with these types of coatings.
- Protect the device from moisture, contamination, and damage.
- Make sure that the status indicator is clearly visible.
- Do not affix any labels or stickers to the optics cover.
- Do not subject the device to excessive shock or vibrations. In systems subjected to heavy vibrations, secure the fixing screws with screw-locking devices.
- The ventilation element must not be sealed off during installation.
- The device must be mounted in such a way that no water can pool on the ventilation element. When using a mounting bracket, we recommend providing a drill hole in the area of the ventilation element.
- Ensure suitable ESD protective measures during mounting.

5.1.1 Ventilation element

The ventilation element ensures an improved pressure equalization and allows the exchange of air and heat between the housing and surroundings.

The breathable membrane allows ambient air to either penetrate into the device or escape again depending on the prevailing ambient conditions see "Dimensional drawing", page 60.

In particular for applications with frequently changing environmental influences (e.g., large temperature fluctuations or rapid temperature changes) or with standing water, the ventilation element ensures a reliable pressure equalization and thereby relieves the seals and adhesive joints of the housing. This can improve the expected service life of the device in the application.

Note the following information:

- Do not affix any labels or stickers to the ventilation element.
- Do not paint over the ventilation element.
- Devices that have been subjected to a long period of moisture or very rapid temperature changes need to first equilibrate after being switched on. In some circumstances, therefore, a period of time should be allowed before measurement readiness of the device because any moisture in the housing must first be taken up by the air in the housing, which is heated up through the operation of the device, so that it can then escape via the ventilation element. Depending on the nature of the precipitated moisture, this time period might be several minutes or even up to hours.

5.2 Mounting the system plug on the device

Notes

NOTICE

Risk of damage due to electrostatic discharge

Electrostatic discharge from the human body may damage the device or the system plug.

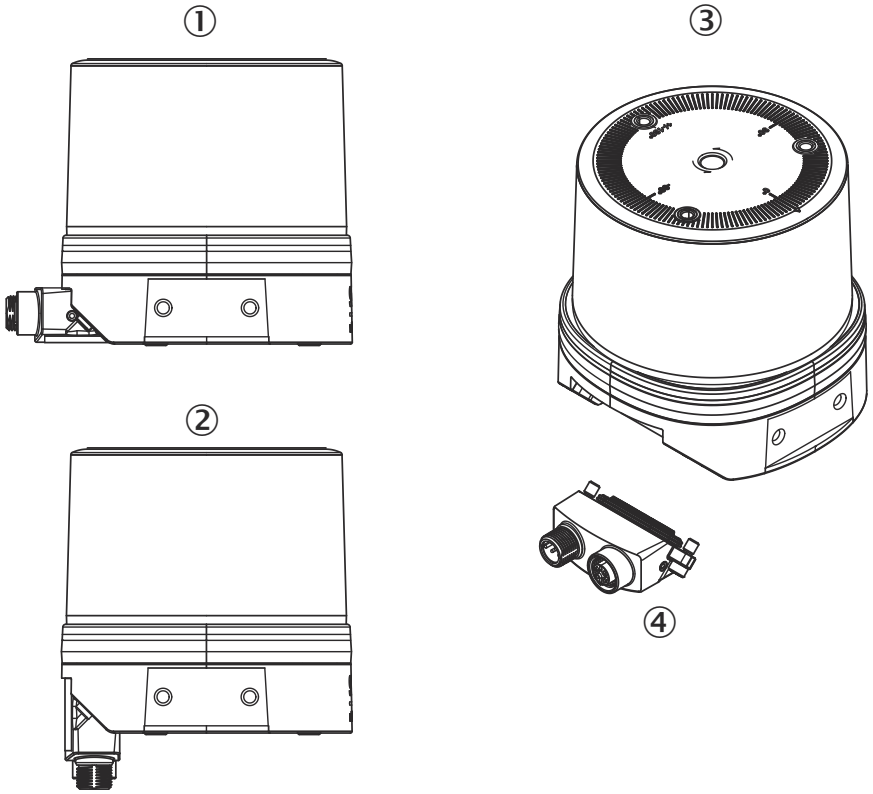
- Take the necessary ESD precautions when mounting the system plug.
- Do not touch the contact surfaces with your fingers.

- Depending on the device variant ordered, the system plug is either already mounted on the device or is supplied separately.
- Compliance with the technical specifications of the device is possible only when the system plug is mounted.

Prerequisites

- The system plug, the seal on the device, and the entire connection area are free of contamination and moisture and show no signs of damage.

Fitting the system plug



① System plug, installation at top
② System plug, installation at bottom
③ Device - system plug not mounted
④ System plug

1. Attach the system plug at the desired installation position on the device.
2. Tighten the screws (tightening torque: max. 2 Nm).

5.3 Mounting the device

Prerequisites

- The connector is mounted on the device see "Mounting the system plug on the device", page 38.

1. Mount the device in a suitably prepared bracket using the fixing holes provided (see "Dimensional drawing", page 60). Mounting brackets are available as accessories, see "Accessories", page 63.
2. Make the electrical connection. Attach and tighten the tension-free cable, see "Connecting the device electrically", page 47.
3. Align the vertical center line of the field of view of the device with the center of the area to be monitored. The marking on the upper side of the optics cover serves as a bearing alignment aid.
4. Switch on the supply voltage.

✓ After successful initialization, the two status LEDs light up green. The device is ready for use.

5. Perform a fine adjustment using a test target and, if necessary, use the alignment aid.

5.4 Mounting multiple devices

NOTICE RISK OF INTERFERENCE FROM OTHER DEVICES!
Radiation sources with a wavelength of 905 nm can cause interference if they affect the device directly.

The device has been designed to minimize the probability of mutual interference, including between different LiDAR sensors. To rule out even the slightest effects on the measurement accuracy, the devices should be arranged in such a way that as few laser beams as possible are received from other devices.

6 Electrical installation

6.1 Wiring instructions

NOTE

Pre-assembled cables can be found on the product page.

The product page can be accessed via the **SICK Product ID**: **pid.sick.com/{P/N}/{S/N}**

{P/N} corresponds to the part number of the product, see type label.

{S/N} corresponds to the serial number of the product, see type label (if indicated).

NOTICE

Faults during operation and defects in the device or the system

Incorrect wiring may result in operational faults and defects.

- Follow the wiring notes precisely.

The electrical connection of the device is configured as an M12 round connector.

The enclosure rating stated in the technical data is achieved only with screwed plug connectors or protective caps.

All circuits connected to the device must be configured as SELV or PELV circuits. SELV = safety extra-low voltage, PELV = protective extra-low voltage.

Protect the device with an external fuse of 3 A slow-blow at the beginning of the supply cable.

Connect the connecting cables in a de-energized state. Do not switch on the supply voltage until installation is complete and all connecting cables are connected to the device and control.

Wire cross-sections in the supply cable from the customer's power system must be implemented in accordance with the applicable standards.

Prior to connecting the I/O line, check the device configuration for the inputs/outputs.

Avoid tensile loads to the connecting cables.

Maximum cable lengths for the voltage supply, depending on the available power supply voltage. The maximum cable length and the permissible minimum voltage at the power supply unit can be calculated with the help of the calculation rule in the following sections.

6.2 Prerequisites for safe operation of the device

WARNING
Risk of injury and damage caused by electrical current!

As a result of equipotential bonding currents between the device and other grounded devices in the system, faulty grounding of the device can give rise to the following dangers and faults:

- Dangerous voltages are applied to the metal housings.
- Devices will behave incorrectly or be destroyed.
- Cable shielding will be damaged by overheating and cause cable fires.

Remedial measures

- Only skilled electricians should be permitted to carry out work on the electrical system.
- If the cable insulation is damaged, disconnect the voltage supply immediately and have the damage repaired.
- Ensure that the ground potential is the same at all grounding points.
- Where local conditions do not meet the requirements for a safe earthing method, take appropriate measures. For example, ensure low-impedance and current-carrying equipotential bonding.

The device is connected to the peripheral devices (any local trigger sensor(s), system controller) via shielded cables. The cable shield – for the data cable, for example – rests against the metal housing of the device.

The device can be grounded through the cable shield or through a blind tapped hole in the housing, for example.

If the peripheral devices have metal housings and the cable shields are also in contact with their housings, it is assumed that all devices involved in the installation have the **same ground potential**.

This is achieved by complying with the following conditions:

- Mounting the devices on conductive metal surfaces
- Correctly grounding the devices and metal surfaces in the system
- If necessary: low-impedance and current-carrying equipotential bonding between areas with different ground potentials

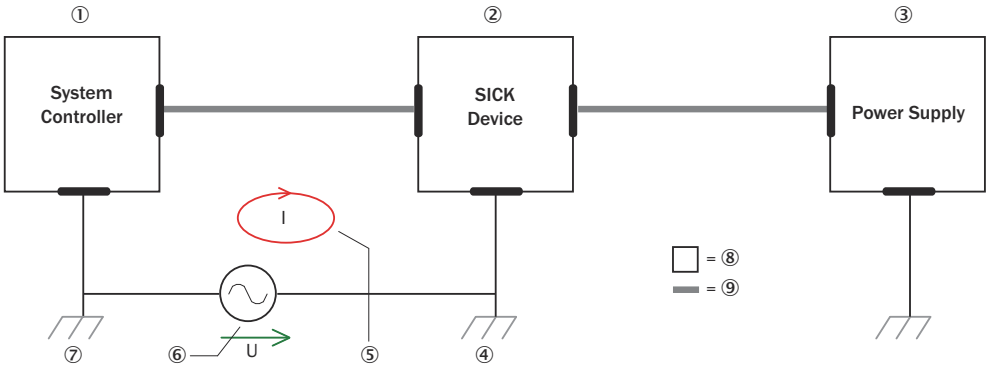


Figure 33: Example: Occurrence of equipotential bonding currents in the system configuration

① System controller
② Device
③ Voltage supply
④ Grounding point 2
⑤ Closed current loop with equalizing currents via cable shield
⑥ Ground potential difference
⑦ Grounding point 1
⑧ Metal housing
⑨ Shielded electrical cable

If these conditions are not fulfilled, equipotential bonding currents can flow along the cable shielding between the devices due to differing ground potentials and cause the hazards specified. This is, for example, possible in cases where there are devices within a widely distributed system covering several buildings.

Remedial measures

The most common solution to prevent equipotential bonding currents on cable shields is to ensure low-impedance and current-carrying equipotential bonding. If this equipotential bonding is not possible, the following solution approaches serve as a suggestion.

NOTICE

We expressly advise against opening up the cable shields. This would mean that the EMC limit values can no longer be complied with and that the safe operation of the device data interfaces can no longer be guaranteed.

Measures for widely distributed system installations

On widely distributed system installations with correspondingly large potential differences, the setting up of local islands and connecting them using commercially available **electro-optical signal isolators** is recommended. This measure achieves a high degree of resistance to electromagnetic interference.

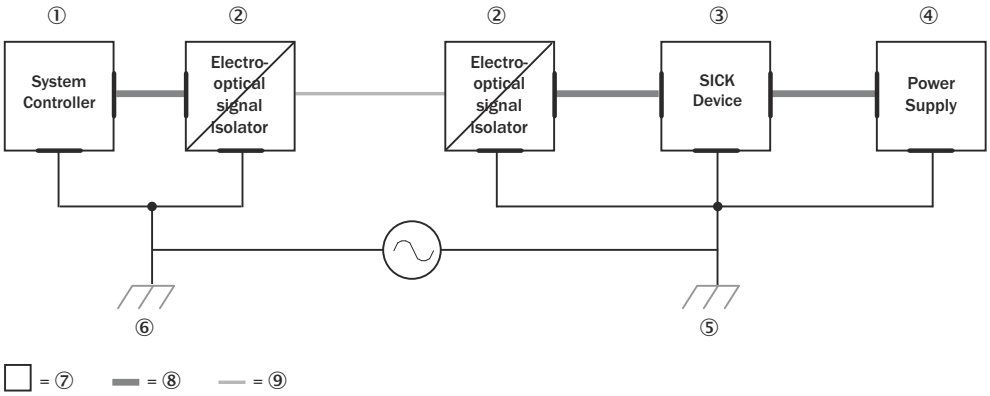


Figure 34: Example: Prevention of equipotential bonding currents in the system configuration by the use of electro-optical signal isolators

① System controller
② Electro-optical signal isolator
③ Device
④ Voltage supply
⑤ Grounding point 2
⑥ Grounding point 1
⑦ Metal housing
⑧ Shielded electrical cable
⑨ Optical fiber

The use of electro-optical signal isolators between the islands isolates the ground loop. Within the islands, a stable equipotential bonding prevents equalizing currents on the cable shields.

Measures for small system installations

For smaller installations with only slight potential differences, insulated mounting of the device and peripheral devices may be an adequate solution.

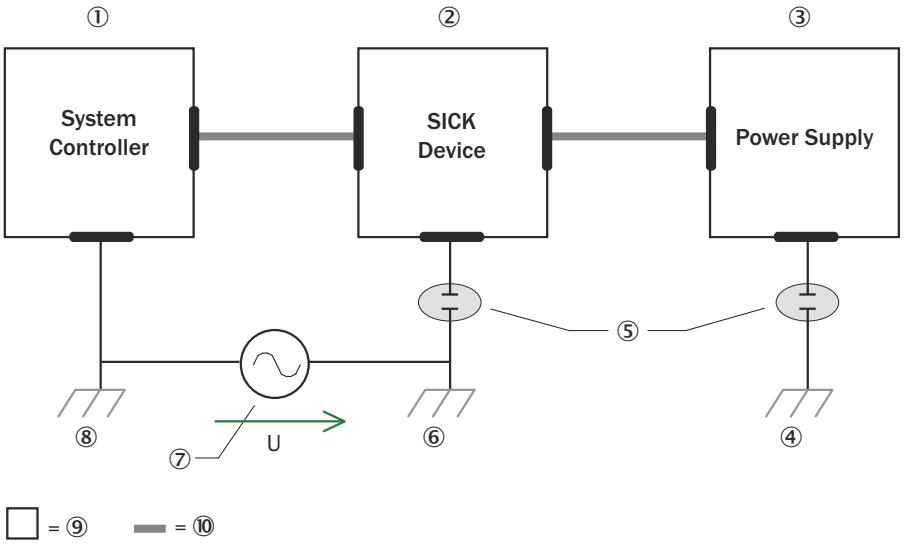


Figure 35: Example: Prevention of equipotential bonding currents in the system configuration by the insulated mounting of the device

① System controller
② Device
③ Voltage supply
④ Grounding point 3
⑤ Insulated mounting
⑥ Grounding point 2
⑦ Ground potential difference
⑧ Grounding point 1
⑨ Metal housing
⑩ Shielded electrical cable

Even in the event of large differences in the ground potential, ground loops are effectively prevented. As a result, equalizing currents can no longer flow via the cable shields and metal housing.

NOTICE

The voltage supply for the device and the connected peripheral devices must also guarantee the required level of insulation.

Under certain circumstances, a tangible potential can develop between the insulated metal housings and the local ground potential.

6.3 Calculation rule

The device can be connected via optional accessories, see "Accessories", page 63.

The following formulas can be used to estimate the required cable lengths or supply voltages. Other conditions of the system must be considered in detail.

Formula for the voltage drop to be considered

$$\Delta V = \frac{I \bullet 2 \bullet L}{A} \bullet \rho \bullet (1 + \alpha \bullet (T - T_0))$$

Formula for permissible length of cable

$$L = \frac{\Delta V \cdot A}{2 \cdot I \cdot \rho \cdot (1 + \alpha \cdot (T - T_0))}$$

Sample calculations

Prerequisites:

- Steady state of the voltage supply
- Only applies for copper cable material

Table 6: Values used in both example calculations

Cable properties	
$A = 0.34 \cdot 10^{-6}\ m^2$	Cross-section of the cable surface [m^2]
$\rho = 1.72 \cdot 10^{-8}\ \Omega m$	Specific resistance of copper [Ωm]
$\alpha = 3.9 \cdot 10^{-3}\ K^{-1}$	Temperature coefficient of copper [1/K]
Ambient conditions	
$T_0 = 20\ °C$	Reference temperature [°C]
$T = 80\ °C$	Cable temperature [°C]
Cable load	
I = P/U = 1.46 A P = 35 W U= 24 V	Load current I [A] Maximum expected power consumption P [W] Supply voltage U [V]

Table 7: Example: voltage drop to be considered for cable part no. 2096241

L = 10 m	Cable length [m]
$\Delta V = \frac{I \cdot 2 \cdot L}{A} \cdot \rho \cdot (1 + \alpha \cdot (T - T_0)) = 1.82\ V$	Voltage drop ΔV [V]

Table 8: Calculation of the cable length for allowed voltage drop of 1.82 V

ΔV = 1.82 V	Voltage drop on the cable [V]
$L = \frac{\Delta V \cdot A}{2 \cdot I \cdot \rho \cdot (1 + \alpha \cdot (T - T_0))} = 10\ m$	Permissible length of cable [m]

6.4 Cable reserve on system plug

Allow for sufficient cable reserve of the supplied cables at the system plug. You can easily exchange the device with the cable reserve if needed.

Keep the cable reserve only long enough that the system plug cannot be accidentally plugged into an adjacent device when replacing the device! This prevents a device with an incorrect configuration being put into operation. Experience has shown that 200 to 300 mm of cable reserve on the device is ideal.

The reserve cable should be laid as a drip loop so no moisture (e.g., condensation) is directed towards the device but instead drips off the cable beforehand.

6.5 Connection diagram

NOTE
The recommended connecting cables and their associated technical data can be found on the online product page.
The product page can be accessed via the **SICK Product ID**: **pid.sick.com/{P/N}/{S/N}**
{P/N} corresponds to the part number of the product, see type label.
{S/N} corresponds to the serial number of the product, see type label (if indicated).

NOTE
The connections depend on the mounted system plug.

PWR/IOs connection

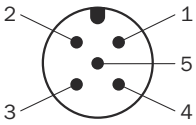


Figure 36: Male connector, M12, 5-pin, A-coded

Table 9: Pin assignment for PWR & 3 I/Os connection (part no. 2116047)

Contact	Signs	Description	Wire color, part number 2095733 [1]
1	Vs	Supply voltage: +9 ... +30 V DC	Brown
2	IN2 / OUT2	Digital input 2 / digital output 2	White
3	GND	Supply voltage: 0 V	Blue
4	IN1 / OUT1	Digital input 1 / digital output 1	Black
5	IN3 / OUT3	Digital input 3 / digital output 3	Gray

1) Data only valid when using the specified connecting cable with flying leads, which is available as an accessory

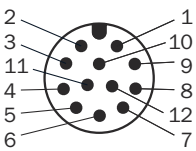


Figure 37: M12 male connector, 12-pin, A-coded

Table 10: Pin assignment for PWR & 8 I/Os connection (part no. 2130754)

Contact	Labels	Description
1	IN1/OUT1	Digital input 1 / digital output 1
2	GND	Supply voltage: 0 V
3	IN2/OUT2	Digital input 2 / digital output 2
4	IN7/OUT7	Digital input 7 / digital output 7
5	IN8/OUT8	Digital input 8 / digital output 8
6	IN3/OUT3	Digital input 3 / digital output 3
7	IN4/OUT4	Digital input 4 / digital output 4
8	IN6/OUT6	Digital input 6 / digital output 6
9	Vs	Supply voltage: +9 ... +30 V DC
10	IN5/OUT5	Digital input 5 / digital output 5

Contact	Labels	Description
11	–	–
12	–	–

Ethernet connection

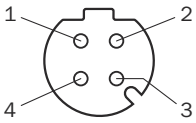


Figure 38: M12 female connector, 4-pin, D-coded

Table 11: Pin assignment for Ethernet connection (part no. 2116047 & 2130754)

Contact	Signs	Description
1	TX+	Sender+
2	RX+	Receiver+
3	TX-	Sender-
4	RX-	Receiver-

6.6 Connecting the device electrically

NOTICE
Observe the wiring instructions, see "Wiring instructions", page 40.

1. Ensure that the power supply unit can provide the necessary voltage and current for operating the device. Particular attention must be given to the voltage drop across the supply line (see "Calculation rule", page 44),and for digital outputs the additional voltage drop in the opposite direction and the required start-up power (see "Mechanics/Electronics", page 58), without which the device cannot start reliably.
2. Ensure the voltage supply is not connected.
3. Connect the device according to the connection diagram, see "Connection diagram", page 46.

7 Commissioning

7.1 Operation using SOPASair

The browser-based SOPASair software can be used to parameterize the device and for service and diagnostic purposes.

To parameterize the device, you will require a computer with a web browser installed and a free Ethernet connection. Alternatively, the connection can be established via a USB connection using an Ethernet USB adapter.

7.1.1 Opening the web server user interface (SOPASair)

Before opening the user interface, perform the following work steps:

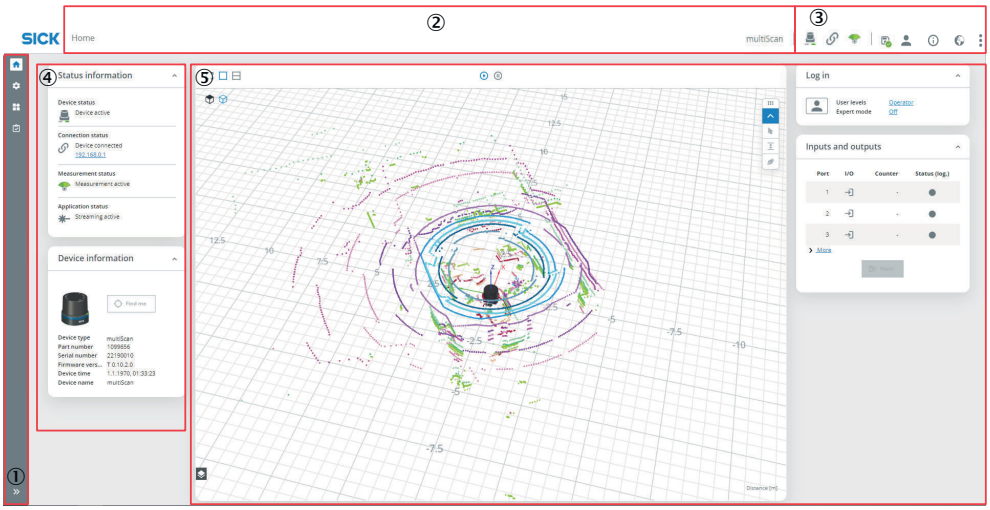
- Connect the device to the computer via Ethernet.
- Set up the voltage supply for the device.
- Ensure that the computer and device are located in the same network.
- Ensure that the computer uses a different IP address than the device, but is in the same IP address range (e.g. 192.168.0.xxx)

Opening user interface:

1. Open web browser (recommendation: Google Chrome).
2. Enter the device IP address into the address line. The standard IP address is: 192.168.0.1

✓ The SOPASair user interface is displayed.

7.1.2 Overview



① Show and hide menu bar
② Menu path for opened menu
③ Status indicators | toolbar
④ Menu panel **Status** and **Device**
⑤ Workspace with live image and menu panels

Status indicators

LED display

Device connection status

Measurement status

Toolbar

Save permanently

Open login window

Change user interface language

Open the global online help. The same icon is also displayed for help contents right next to parameters.

Device menu

Navigation

1. Click on the desired menu.

✓ The workspace changes depending on the selected menu.

7.1.3 Navigating in the live image

Overview

The view of the scan data In SOPASair can be adjusted using the computer mouse.

Using the computer mouse

Function	Computer mouse
Zoom in and zoom out	Using the scroll wheel
Rotate and tilt the view	Left mouse button
Move the view	Right mouse button

7.1.4 User levels

The device has different user levels.

The user levels have different authorizations for configuring the device.

The current user level is displayed in the **Log in** panel.

Activate the user level during initial commissioning of the device

1. Click on the button.

✓ The **Log in to device** input screen opens. The **Service** user level is selected.

2. Enter the password servicelevel and click on **Log in**.
3. Activate the desired user level.

Logging in to the device

1. User levels have been activated.
2. Click on the button.

✓ The **Log in to device** input screen opens.

3. Select user level(**User level**), enter a password(**Password**) then click **Login** in.

User levels	Password	User and authorizations
Maintenance	main	Customers: Display only, no configuration
Authorized customer	client	Technical staff: Install and configure device
Service	servicelevel	Service staff: Make advanced configuration settings

NOTE
Change the passwords during initial commissioning to protect your device.
A higher user level can change the password of a lower user level.

7.1.5 Changing the password

1. Establish a connection to the product in the web browser.
2. Select:
3. Log in with the last assigned password.
4. Select **Change password**.
5. Assign a new password.

✓ The new password is valid immediately.

7.1.6 Resetting the password

NOTE
The responsible SICK sales company or the responsible SICK service partner carefully checks each code request to reset the password. A risk of deception by third parties nevertheless exists. The operating entity should therefore take suitable security measures.
The operating entity should also take suitable measures to limit, as best as possible, access to the product. This includes, in particular, physical access as well as access to the software interfaces of the product.

Resetting the password for the Service user

1. Click on the button.

✓ The **Log in to device** input screen opens.

2. Click on **Password forgotten?**.

Resetting the password for the Authorized client/Maintenance personnel user

1. The password for the **Service** user has been reset.
2. Save the device parameterization using the parameter export in SOPAS ET.
3. Click on the button.
4. Select **Reset to factory settings**.

7.1.7 Displaying live data

1. In SOPASair: Open the user interface see "Opening the web server user interface (SOPASair)", page 48.

If no scan data are shown:

1. Click **Configuration** > **Defaults**.
2. Select **Measuring**.

7.1.7.1 Activating/deactivating filters

Prerequisites

- You must be logged into the device see "User levels", page 49.

Selecting filters

1. Menu: Select **Configuration** > **Default**.
2. Activate or deactivate one or more filters.

Selecting the data reduction filter/region of interest filter

1. Menu: Select **Application** > **Data output** .
2. Activate or deactivate one or more filters.

7.1.8 Configuring interfaces

Settings for using the multifunctional I/Os can be configured in the **Application** > **Inputs and outputs** menu. In addition, the current status and saved function of the input/output are displayed.

The multifunctional I/Os are switchable and can therefore each be used as either a digital input or output.

The inputs may switch on, switch off, and switch over analysis cases, for example. The inputs can also be used to activate other functions, such as measurement data output triggering.

The outputs can be used as digital switching outputs, for example to ground (PNP) (depending on the device type). For each output, it is necessary to define whether it is to be switched by means of SOPAS ET telegrams, or whether it is being used to signal device readiness.

7.2 Operation in SOPAS ET

Execute the functions listed below via the SOPAS ET configuration software.

Functions

- Install firmware updates
- Import and export data

NOTE

To use the sensor with SOPAS ET, we recommend using port 2122 or port 80 in SOPAS ET. When using the aforementioned ports, no limitation with regard to SOPASair can be expected.

If a legacy protocol (CoLa A/B) is used on the ports 2111 and 2112, however, functional limitations in the SOPAS ET interface may arise.

The most up-to-date version of the SOPAS ET software can be downloaded from www.sick.com/software, category: **Configuration software**, software type: **SOPAS ET**.

7.2.1 Operation with SOPAS ET

Version 3.3.3 and higher of the SOPAS Engineering Tool (SOPAS ET) software can be used to parameterization of the device and for service and diagnostic purposes.

To configure the device, you will require a computer with SOPAS ET installed and a free Ethernet connection. Alternatively, the connection can be established via a USB connection using an Ethernet USB adapter.

1. Connect the communication interface (Ethernet, 4-pin M12 female connector) of the device to the computer.
2. Switch on and start the computer.
3. Supply the device with voltage (5-pin M12 male connector, supply voltage 9 ... 30 V DC).

✓ After successful initialization, the two status LEDs light up green. The device is ready for use.

NOTE

To use SOPAS ET with the device, you need a device description file (SDD, SOPAS Device Description) for this device. You can install this within SOPAS ET using the device catalog. The device description file is saved on the device and can be installed there. Alternatively, installation is possible from the SICK website (Internet connection required).

Following installation of the device description file, the device can be selected from the device catalog and added to a project.

A connection to the device is established via the communication interface. The connection must be activated for data transmission (**online**).

Certain functions (e.g., Edit parameters) require you to be logged in to the device (**Device** > **Log In** menu, **User Level**: Authorized customer, **Password** (factory default): client).

Table 12: Keywords for factory setting

User levels	Keyword according to factory setting
Maintenance personnel	main
Authorized client	client
Service	servicelevel

NOTE

Change the passwords during initial commissioning to protect your device.

A higher user level can change the password of a lower user level.

Information about the device is displayed in the device window and the device can also be configured here (**Device**> **Open** menu).

Resetting the password

NOTE

The responsible SICK sales company or the responsible SICK service partner carefully checks each code request to reset the password. A risk of deception by third parties nevertheless exists. The operating entity should therefore take suitable security measures.

The operating entity should also take suitable measures to limit, as best as possible, access to the product. This includes, in particular, physical access as well as access to the software interfaces of the product.

Resetting the password for the Service user

1. Click on the button.

✓ The **Log in to device** input screen opens.

2. Click on **Password forgotten?**.

Resetting the password for the Authorized client/Maintenance personnel user

1. The password for the **Service** user has been reset.
2. Save the device parameterization using the parameter export (**Device** menu > **Export** > **To a file**).
3. In the device window, click on the button.
4. Select **Reset to factory settings.**

8 Maintenance

8.1 Maintenance plan

During operation, the device works maintenance-free.

NOTE
No maintenance is required to ensure compliance with the laser class.

Depending on the assignment location, the following preventive maintenance tasks may be required for the device at regular intervals:

Table 13: Maintenance plan

Maintenance work	Interval	To be carried out by
Check device and connecting cables for damage at regular intervals.	Depends on ambient conditions and climate.	Specialist
Clean housing.	Depends on ambient conditions and climate.	Specialist
Clean housing and optics cover.	Depends on ambient conditions and climate.	Specialist
Check the screw connections and plug connectors.	Depends on the place of use, ambient conditions or operating requirements. Recommended: At least every 6 months.	Specialist
Check the mounting accessories and vibration dampers used.	Depends on the place of use, ambient conditions or operating requirements. Recommended: At least every 6 months.	Specialist

8.2 Cleaning

NOTICE
Equipment damage due to improper cleaning.
Improper cleaning may result in equipment damage.

- Only use recommended cleaning agents and tools.
- Never use sharp objects for cleaning.

► Clean the optics cover at regular intervals and in the event of contamination with a lint-free lens cloth and plastic cleaning agent. Rinse off coarse dirt first with water. The cleaning interval essentially depends on the ambient conditions.

NOTICE
If the optics cover is scratched or damaged (cracked, broken), it must be replaced. Contact SICK Support to arrange this.

- If the optics cover is cracked or broken, take the device out of operation immediately for safety reasons and have it repaired by SICK.

9 Troubleshooting

9.1 General faults, warnings, and errors

Possible faults and corrective actions are described in the table below for troubleshooting. For faults that cannot be rectified using the information below, please contact SICK Service. To find your agency, see the final page of this document.

NOTE
Before calling, make a note of all type label data such as type designation, serial number, etc., to ensure faster assistance.

Table 14: Troubleshooting questions and replies

Question / status	Response / remedial actions
Both LEDs flash red.	Device error: Read the error code via the SOPAS ET PC software and remedy the cause of the error.
LEDs indicate an undefined status.	Check the device status, if necessary contact the SICK Service department.
All LEDs are off	Check the voltage supply to the device. In SOPASair, check whether the LEDs were switched off.
All LEDs of the device light up red at startup and do not change to green.	Check the voltage supply to the device. The power supply unit may not be supplying the required current or the required voltage to start the device.
All LEDs of the device flash red.	The device may not be able to recognize the system plug. Check that the system plug is mounted correctly and that both contact sides are clean and dry.
Measurement data show anomalies.	Optics cover contaminated: Clean the optics cover.
When accessing the device via a web browser, the SOPASair user interface is not loaded, the SOPASair loading screen is permanently displayed.	Try connecting again. If this does not work: Restart the device.
SOPASair is not started in the browser.	Check the IP address of device and network adapter (e.g., using device search in SOPAS ET) and adjust if necessary. Then try to establish the connection again.
No connection to the device possible.	Check whether the system plug is securely mounted see "Mounting the system plug on the device", page 38.

9.2 Repairs

Repair work on the device may only be performed by qualified and authorized personnel from SICK AG. Interruptions or modifications to the device by the customer will invalidate any warranty claims against SICK AG.

9.3 Returns

- ► Only send in devices after consulting with SICK Service.
- ► The device must be sent in the original packaging or an equivalent padded packaging.

NOTE

To enable efficient processing and allow us to determine the cause quickly, please include the following when making a return:

- Details of the contact person
- Description of the application
- Description of the fault that occurred

9.4 Disposal

If a device can no longer be used, dispose of it in an environmentally friendly manner in accordance with the applicable country-specific waste disposal regulations. Do not dispose of the product along with household waste.

NOTICE

Danger to the environment due to improper disposal of the device.

Disposing of devices improperly may cause damage to the environment.

Therefore, observe the following information:

- Always observe the national regulations on environmental protection.
- Separate the recyclable materials by type and place them in recycling containers.

10 Technical data

NOTE

The relevant online product page for your product, including technical data, dimensional drawing, and connection diagrams, can be downloaded, saved, and printed from the Internet.

The product page can be accessed via the **SICK Product ID: pid.sick.com/{P/N}/{S/N}**

{P/N} corresponds to the part number of the product, see type label.

{S/N} corresponds to the serial number of the product, see type label (if indicated).

Please note: This documentation may contain further technical data.

10.1 Features

Measurement principle	Statistical measurement procedure
Application	Indoor and outdoor
Light source	Infrared (wavelength 905 nm)
Laser class	Laser class 1 (EN 60825-1:2014+A11:2021, IEC 60825-1:2014, EN/IEC 60825-1:2007) Complies with 21 CFR 1040.10 and 1040.11 except for conformance with IEC 60825-1 Ed. 3 as described in Laser Notice No. 56, dated May 8, 2019.
Horizontal aperture angle	360°
Vertical aperture angle	42° (approx. +34.5° to -7.5°, DIN ISO 8855)
Scan field flatness	± 0.65°
Scan rate	20 Hz
Angular resolution	Horizontal: 0.5° Vertical: approx. 2.5° and 5° see "Scan layer address", page 25. Note: The actual position of a scan layer may differ. The actual vertical angle value is included in the output of measured values.
Working range	0.05 m ≤ x ≤ 62 m [1]
Scanning range at 10% remission and > 99% detection probability	20 m @ 100 klx 22 m @ 30 klx 25 m @ 10 klx
Scanning range at 60% remission and > 99% detection probability	62 m @ 10 klx
Scanning range at 90% remission and > 99% detection probability	40 m @ 100 klx 60 m @ 30 klx 62 m @ 10 klx
Spot divergence	Vertical: 5.3 mrad / 0.3° Horizontal [2]: 5.3 mrad + 2.2 mrad / 0.3° + 0.125°
Light spot size at front screen	13 mm
Maximum number of echoes that are output	3

1) Specified measurement accuracy for ≥ 0.1 m
2) In the scan direction

Working range diagram

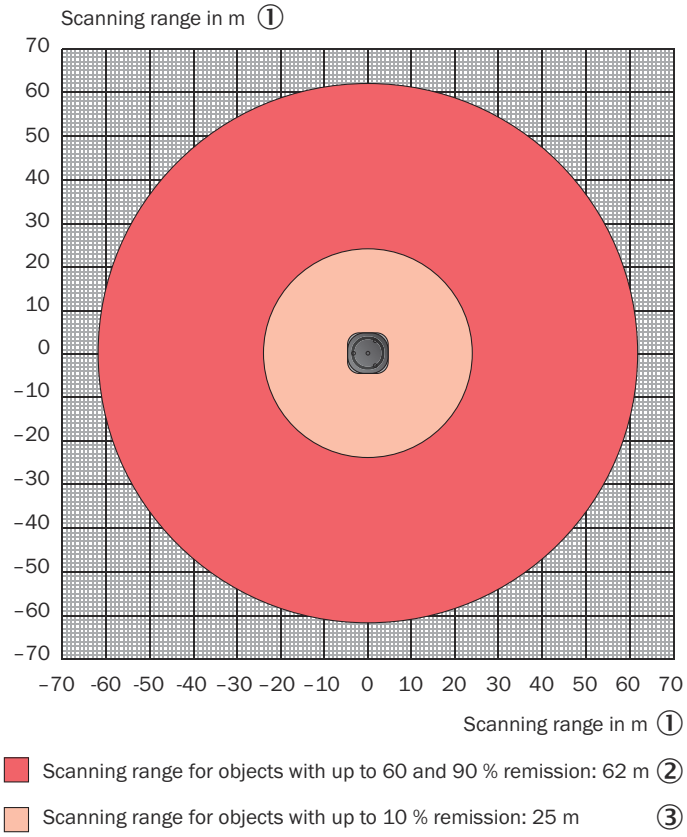


Figure 39: Diagram of the working range (10 klx), topview

① Scanning range in m
② Scanning range for object with up to 60% and 90% remission: 62 m
③ Scanning range for object with up to 10% remission: 25 m

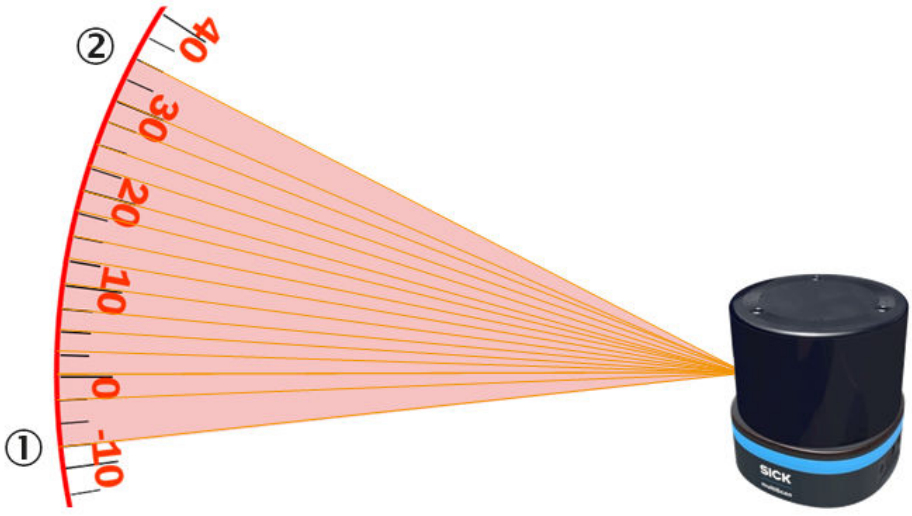


Figure 40: Positions of the 16 scan layers, side view

① Scan layer 1
② Scan layer 16

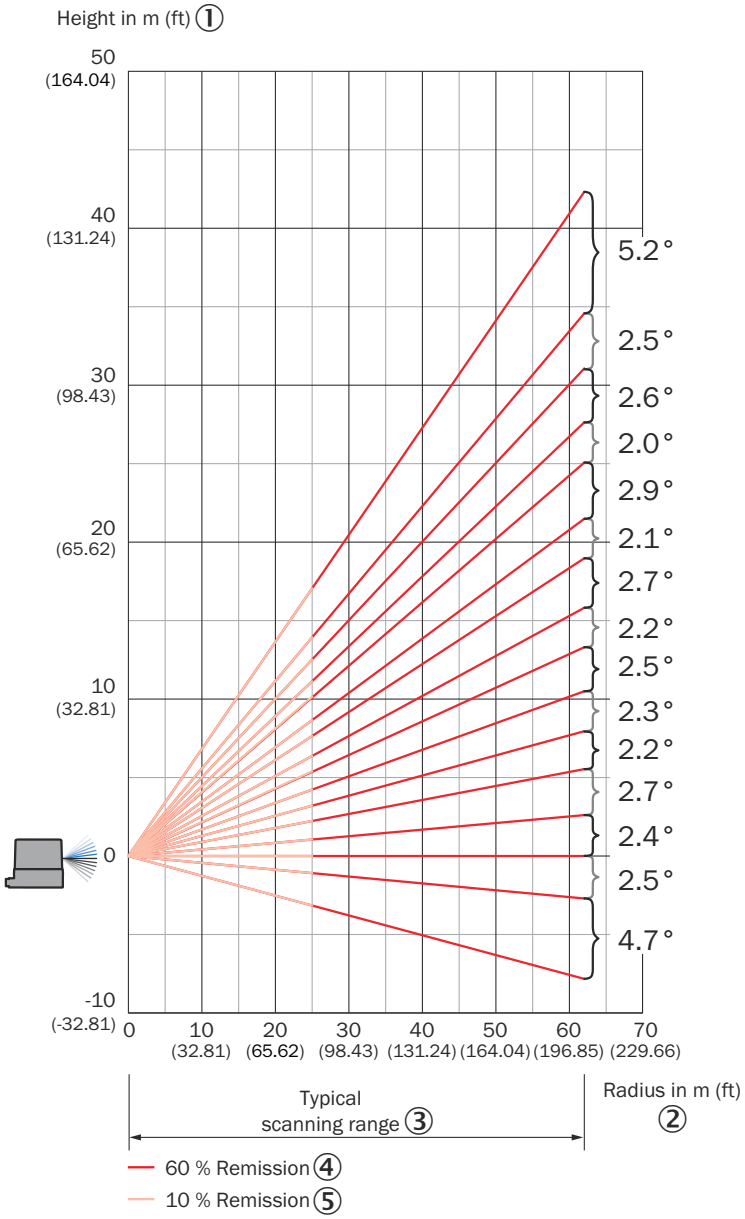


Figure 41: Working range diagram, side view. The actual position of a scan layer can vary. The exact actual vertical angle value of each scan layer can be read from the device.

① Height in m (ft)
② Radius in m (ft)
③ Typical scanning range
④ 60% and 90% remission
⑤ 10% remission

10.2 Mechanics/Electronics

Connection type	Depending on the mounted system plug, 2 x M12 round connectors
Supply voltage	9 V DC ... 30 V DC
Permissible residual ripple	± 5%

Power consumption	P_{typ} = 10 W P_{start} = 35 W for 5 s (motor start-up) P_{max} = 22 W (with full specified current at all outputs)
Housing	ALSi12 Optics cover: polycarbonate
Housing color	Grey (RAL 7016)
Enclosure rating [1]	IP65 / IP67 / IP69 (IEC 60529:1989+AMD1:1999+AMD2:2013) IPX9K (ISO 20653:2013) **Test conditions:** • Water spray volume: 14 l/min ... 16 l/min • Water pressure/temperature: 10000 KPa (100 bar) / 80 °C • Flat jet nozzle distance: 100 mm ... 150 mm • Spray angle: 0°, 30°, 60°, 90° • Cycle: 30 seconds per position • Rotational speed of test specimen: 5 rpm
Protection class	III (IEC 61140:2016-11)
Electrical safety	IEC 61010-1:2010-06
Weight	0.7 kg
Dimensions (L x W x H)	100.3 mm x 100.3 mm x 98.5 mm
Maximum output current	Depends on the number of outputs used/parameterized. 100 mA per channel when using 4 outputs, 200 mA per channel for 2 outputs, 50 mA per channel for 8 outputs.
MTBF	≥ 50 years
$MTTF_D$	≥ 100 years, at 25 °C ambient temperature (EN/ISO 13849-1:2015)

1) **Prerequisites:**
- The system plug is mounted.
- The cables plugged into the electrical connections must be screwed tight.
- Unused electrical connections are sealed off with a protective cap.

10.3 Dimensional drawing

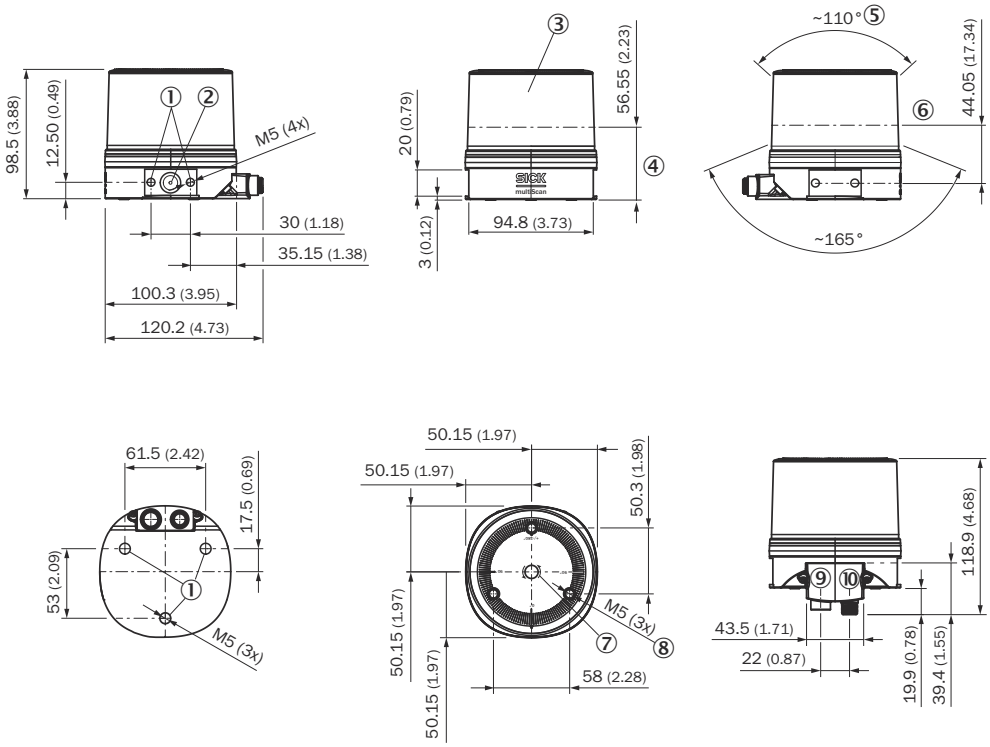


Figure 42: Device structure and dimensions, dimensions in mm (inch)

① M5x7.5 fixing holes for mounting the device
② Ventilation element (membrane)
③ Optics cover
④ Dimension relates to the flat floor, not to the support surfaces on the attachment points. These set the device 0.5 mm higher.
⑤ Aperture angle (vertical field of vision)
⑥ Visual zero position with maximum viewing range
⑦ Direction of rotation
⑧ M5x7.5 fixing holes for accessories only
⑨ Supply voltage connection
⑩ Ethernet connection

10.4 Performance

Measurement data rate	230,400 ... 691,200 measuring points/second
Scanning frequency	20 Hz
Frame rate per frame	≤ 50 ms
Frame rate per segment	< 5 ms
Power-up time	typ. 15 s
Systematic error	± 35 mm Temperature drift: Typically ± 0.5 mm/K
Statistical error	≤ 10 mm
Integrated application	Measurement data output (based on ordered configuration), integrated field evaluation with flexible fields (based on ordered configuration)

Number of configurable fields	Up to 48 fields
Simultaneous detection field	Up to 20 fields see "Field evaluation", page 34.
Filters	Fog filter, echo filter, particle filter, moving average filter, interval filter, scan range filter, scan layer filter, cuboid area filter, distance filter

10.5 Interfaces

Ethernet	✓, UDP/IP (Compact, MSGPACK), TCP/IP (configuration and secondary data via COLA telegrams) Function: HOST, NTP, PTP Measured data output (distance, RSSI) Data transmission rate: up to 1 Gbit/s, depending on the mounted system plug
Digital inputs/outputs	I/O (8 multiports), depending on the mounted system plug
Measurement data output format	MSGPACK, Compact
Output data	Contamination indication, IMU (secondary sensor data)
IMU (inertial measuring unit)	Sampling rate: 100 Hz **Relative position of the IMU to the optical origin:** • X: -17 mm • Y: +23.5 mm • Z: -28.1 mm
Optical displays	4 LEDs
Configuration software	SOPASair (web server), SOPAS ET (software), REST API
Driver	ROS1, ROS2, C++

10.6 Ambient data

Remission factor	2 % ... > 1,000% (reflector)
Electromagnetic compatibility (EMC)	EN 61000-6-2:2005 EN 61000-6-3:2007+A1:2011 ISO 14982-1, ISO 14982-2 [1)] ISO 13766-1 [1)] UN ECE R10 ready [1)]
Vibration resistance	Sine resonance scan: 10 Hz ... 1,000 Hz (IEC 60068-2-6:2007) Sine test: 10 Hz ... 500 Hz; 0.35 mm/5 g; 10 cycles (IEC 60068-2-6:2007) Noise test: 10 Hz ... 250 Hz; 4.24 g RMS, 5 h (IEC 60068-2-64:2008)
Shock resistance	IEC 60068-2-27:2008 50 g, 11 ms, ± 3 single shocks/axis 25 g, 6 ms, ± 1,000 continuous shocks/axis 50 g, 3 ms, ± 5,000 continuous shocks / axis
Shock resistance	IEC 60068-2-75: Hammer impact test: 0.5 joule & 2 joule
Ambient operating temperature	-30 °C ... +50 °C (IEC 60068-2-1, IEC 60068-2-2, IEC 60068-2-14 (Nb))
Storage temperature	-40 °C ... +70 °C (IEC 60068-2-14 (Nb))
Operating and storage air humidity	Max. 90 % RH (non-condensing) (IEC 60068-2-30 (Db1))

Temperature change	–33 °C ... +50 °C, 10 cycles (EN 60068-2-14:2009)
Chemical resistance	Salt mist test (IEC 60068-2-52, Procedure 4)
Altitude	< 5,000 m above sea level
Ambient light immunity	100 klx (indirect)

1) Load dump: from ISO 16750-2 Test B Severity Level 4 passed for 12 V systems. Required in case of transient disturbances on the input filtering signal lines (debounce > 10 ms)

11 Accessories

NOTE

On the product page you will find accessories and, if applicable, related installation information for your product.

The product page can be accessed via the **SICK Product ID**: **pid.sick.com/{P/N}/{S/N}**

{P/N} corresponds to the part number of the product, see type label.

{S/N} corresponds to the serial number of the product, see type label (if indicated).

Support Portal

NOTE

In the SICK Support Portal (**supportportal.sick.com**, registration required) you will find not only useful service and support information for your product as well as commands for test purposes via the REST communication interface, but also the Cybersecurity Hardening Guide as well as further detailed information on the available accessories and their use.

12 Annex

12.1 Declarations of conformity and certificates

You can download declarations of conformity and certificates via the product page.

The product page can be accessed via the **SICK Product ID: pid.sick.com/{P/N}/{S/N}**

{P/N} corresponds to the part number of the product, see type label.

{S/N} corresponds to the serial number of the product, see type label (if indicated).

12.2 Licenses

SICK uses open source software which is published by the rights holders under a free license. Among others, the following license types are used: GNU General Public License (GPL version 2, GPL version 3), GNU Lesser General Public License (LGPL), MIT license, zlib license and licenses derived from the BSD license.

This program is provided for general use without warranty of any kind. This warranty disclaimer also extends to the implicit assurance of marketability or suitability of the program for a particular purpose.

More details can be found in the GNU General Public License.

For license texts see www.sick.com/licensetexts.

Printed copies of the license texts are also available on request.

12.3 Communication interfaces

Communication with the device is possible via CoLa A/B and REST. For more information, see the following English telegram listing.

Under Downloads on the product page, you can download, for example, the Open API file.

The product page can be accessed via the **SICK Product ID: pid.sick.com/{P/N}/{S/N}**

{P/N} corresponds to the part number of the product, see type label.

{S/N} corresponds to the serial number of the product, see type label (if indicated).

12.4 Telegram listing (EN)

Contents

12.4.1.1 About this document

12.4.1.1.1 Information on the telegram listing

NOTE
In case you prefer to use complete drivers instead of single telegrams, the following options are availabe:

C++ drivers: https://github.com/SICKAG/sick_scan_xd

ROS drivers: https://github.com/SICKAG/sick_scan_xd

ROS2 drivers: https://github.com/SICKAG/sick_scan_xd

NOTE
Telegrams that are not described in this document for the device should not be implemented as they may either be incompetible or cause undesired effects.

NOTE
CoLa 2 is a SICK specific communication protocol which is used for communication between SICK devices and SICK specific tools and services only.

Please read this chapter carefully before beginning to use the telegram listing.

The document shows how to send telegrams via a terminal program using the SICK protocol CoLa A (ASCII and hexadecimal values, with TCP port 2111) or CoLa B (binary/ hexadecimal values, with TCP port 2112 only) to the device . This comprises the query of the current device state or certain parameter values, how to modify parameter values and the way in which the device confirms or responds to commands/telegrams.

The devices generally support automatic IP address discovery.

Default IP address is:

- 192.168.0.1

Subnet mask is 255.255.255.0.

IP ports:

- 2111: CoLa A
- 2112: CoLa B

Most parameter changes also require certain user levels. Additionally, commands may change during the product lifecycle and development process with a new firmware.

This document is based on the following firmware version (or newer):

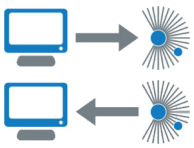
- V2.0.0

If commands do not seem to work, please verify that your device version supports this functionality, that the minimum required user level has been selected and check on updates of this documentation.

12.4.1.1.2 Explanation of symbols

NOTE

... highlights useful tips and recommendations as well as information for efficient and trouble-free operation.



Telegram to device

Telegram from device

Unformatted example to copy and paste

12.4.1.2 Communication format

12.4.1.2.1 Binary telegram (CoLa B)

The binary telegram is a basic protocol of the scanner (CoLa B). All values are in hexadecimal code and grouped into pairs of two digits (= 1 byte). The string consists of four parts: header, data length, data and checksum (CS). It is highly recommended to use this protocol as the transmitted data amount is only about half as much as with CoLa A).

The header indicates with 4 × STX (02 02 02 02) the start of the telegram.

The data length defines the size of the data part (command part) by indicating the number of digit pairs in the third part. The size of the data length itself is 4 bytes, which means that the data part might have a maximum of 16^8 = 4,294,967,295 digit pairs (bytes).

The data part comprises the actual command with letters and characters converted to Hex (according to the ASCII chart) and the parameters of either decimal numbers converted to Hex or fixed Hex values with a specific, intrinsic meaning (no conversion). There is always a space (20) between the command and the parameters, but not between the different parameter values.

The checksum finally serves to verify that the telegram has been transferred correctly. The length of the checksum is 1 byte, CRC8. It is calculated with XOR.

Table 15: Example: Binary telegram

02 02 02 02	00 00 00 17	73 4D 4E 20 53 65 74 41 63 63 65 73 73 4D 6F 64 65 20 03 F4 72 47 44	B3

Header	Length	Data	CS

This is an example telegram for setting the user level “Authorized Client”:

- Header = 02 02 02 02
- Length = 23 bytes (17h)
- Data:
  - 73 4D 4E 20 = sMN = start of Sopas command (and space)
  - 53 65 74 41 63 63 65 73 73 4D 6F 64 65 20 = Set Access Mode = the actual command for setting the user level (and space)
  - 03 = fixed Hex value meaning user level “Authorized Client”
  - F4 72 47 44 = fixed Hex value, serving as password for the selected user level “Authorized Client”
- Checksum = B3 from XOR calculation

12.4.1.2.2 ASCII telegram (CoLa A)

The ASCII telegram is an alternative to the binary telegram, suitable especially to parametrize the sensor. However, due to the variable string length of ASCII telegrams, the Binary telegram is still recommended when using scanners with a PLC.

The ASCII telegram has the advantage that commands can be written in plaintext. The string consists only of two parts: the framing and the data part.

The framing indicates with <STX> and <ETX> the start and stop of each telegram.

The data part comprises the actual command with letters and characters (plaintext), parameter values either in decimal (special indicator required) or in hexadecimal (example: a frequency of 25 Hz = +2500 (decimal) = 09C4 (Hex)) and fixed hexadecimal values with a specific, intrinsic meaning.

NOTE
Leading zeros are deleted in ASCII. Therefore a space is always required between all command parts and parameter parts.

As further alternative within CoLa A, depending on the preferences of the user, all values can be written directly in Hex. This means however a 1:1 conversion of all letters and characters including numbers and fixed hexadecimal values via the ASCII chart.

NOTE
The device will confirm parameter values always in hexadecimal code, regardless of the code sent.

Table 16: Example: ASCII telegram

ASCII	<STX>	sMN{SPC}SetAccessMode{SPC}03{SPC}F4724744	<ETX>
Hex	02	73 4D 4E 20 53 65 74 41 63 63 65 73 73 4D 6F 64 65 20 30 33 20 46 34 37 32 34 37 34 34	03
	Start	Data	Stop

This is again an example telegram for setting the user level “Authorized Client”. As only fixed hexadecimal parameter values are needed, the option to use parameter values in decimal code with special indicator cannot be applied here:

- Framing = <STX> = telegram start = 02 (Hex)
- Data:
  - sMN = start of Sopas command (and blank) = 73 4D 4E 20 (Hex)
  - SetAccessMode = the actual command for setting the user level (and blank) = 53 65 74 41 63 63 65 73 73 4D 6F 64 65 20 (Hex)

  - 03 = fixed Hex value meaning user level “Authorized Client” (and blank) = 30 33 20 (Hex)
  - F4 72 47 44 = fixed Hex value, serving as password for the selected user level “Authorized Client” = 46 34 37 32 34 37 34 34 (Hex)
- Framing = <ETX> = telegram stop = 03 (Hex)

12.4.1.2.3 Variable types

Variable type	Length (byte)	Value range	Sign
Bool_1	1	0 or 1	No
Uint_8	1	0 ... 255	No
Int_8	1	-128 ... +127	Yes
Uint_16	2	0 ... 65,535	No
Int_16	2	-32,768 ... +32,767	Yes
Uint_32	4	0 ... 4,294,967,295	No
Int_32	4	-2,147,483,648 ... +2,147,483,647	Yes
Dint_32	4	-9,223,372,036,854,775,808 ... 9,223,372,036,854,775,807	Yes
Enum_8	1	Certain values defined in a list of Choices (0 ... 255)	No
Enum_16	2	Certain values defined in a list of Choices (0 ... 65535)	No
String	Context-depend-ent	Strings are not terminated in zeroes	
FlexString	array of visible characters with preceeding cur-rent length (UInt lenght) (array of 8 bit)	See description of String and FlexArray	
Real	4	Float according to IEEE754 (see www.h-schmidt.net/FloatConverter/IEEE754de.html)	Yes

Data length is always given in Bytes!

Struct	A structure is a sequence of further types. These types can be of a Basic-Type, Structs again or an Array.
Array	An Array is a repetition of a type. The length of the array is defined with each Array. The types can be of a BasicType, a Struct or an Array again (n-dimensional).
Flex Array	A FlexArray is a repetition of a type with a variable length. The maximum length of the array is defined with each FlexArray. The current length of the FlexArray is transferred as a UInt preceeding the Array itself. The types can be of a BasicType, a Struct or an Array again (n- dimensional).

12.4.1.2.4 Command basics

Description	Value ASCII	Value Hex	Value Binary
Start of text	<STX>	02	02 02 02 02 + given length
End of text	<ETX>	03	Calculated checksum
Read	sRN	73 52 4E	
Write	sWN	73 57 4E	
Method	sMN	73 4D 4E	

Description	Value ASCII	Value Hex	Value Binary
Event	sEN	73 45 4E	
Answer	sRA sWA sAN sEA sSN	73 52 41 73 57 41 73 41 4E 73 45 41 73 53 4E	
Space	{SPC}	20	20

If values are divided into two parts (e.g. measurement data), they are documented according to LSB 0 (e.g. 00 07), output however is according to MSB (e.g. 07 00).

12.4.1.2.5 Log in: Required user level

Task	Required user level
Change sensor parameters	Authorized Client
Requests or queries (e.g. for measurement data or device state)	None
Manage passwords	Service

In general, every sWN command for changing paramters requires to log in to the device first see "Log in [sMN SetAccessMode] ", page 73. When being logged in, any desired parameter valid for this user level can be changed. All changes become active only after having logged off again from the device via the sMN Run command see "Set to run [sMN Run]", page 91.

In this document, a required, specific user level is indicated in the telegram structure head line.

12.4.1.3 Workflows

12.4.1.3.1 Parameterize the scan

Log in: `sMN SetAccessMode` see "Log in [sMN SetAccessMode] ", page 73

Configure scandata output: `sWN LMPoutputRange` see "Configure aperture angle of the scandata for output [sWN LMPoutputRange]", page 92

Store parameters: `sMN mEEwriteall` see "Save parameters permanently [sMN mEEwriteall]", page 90

Log out: `sMN Run` see "Set to run [sMN Run]", page 91

12.4.1.3.2 Set timestamp/data angle

Log in: `sMN SetAccessMode` see "Log in [sMN SetAccessMode] ", page 73

SOPAS command: `sMN LSPsetdatetime` see "Set time stamp [sMN LSPsetdatetime]", page 100

Log out: `sMN Run` see "Set to run [sMN Run]", page 91

12.4.1.3.3 Common telegrams

The following telegrams are valid for a wide range of non-safe LiDAR sensors from SICK. Please refer to the telegram listing of the respective device for a detailed description of all valid telegrams.

"Log in [sMN SetAccessMode] ", page 73
"Start measurement [sMN LMCstartmeas]", page 77

12.4.1.4 Telegrams

Telegrams listed in this document are described in the following basic structure:

Table 17: Telegram structure: "Command type" "Command"

Telegram structure: "Command type" "Command" (Minimum required user level. If nothing is stated, no user level required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Lists the different parts of the telegram.	Describes the corresponding telegram parts.	Defines the type of the variable.	Defines the length in byte.	Gives further information regarding the values in CoLa A/ CoLa B if neccessary.	Defines the value of the telegram part in CoLa A (ASCII).	Defines the value of the telegram part in CoLa B (Binary).

NOTE

Commands are colored blue, parameters orange for further differentiation.

Table 18: Example: "Command type" "Command"

CoLa A	<"Start of text">"Command type value (ASCII)""space""Command value (ASCII)""space""Parameter value (ASCII)""space""Parameter value (ASCII)"<"End of text">	
	Copy example with framing (ASCII)	
	Copy example without framing (ASCII)	
	Copy example with framing (Hex)	
CoLa B	"Start of text and given length" "Command type value (Binary)""space""Command value (Binary)""space""Parameter value (Binary)""Parameter value (Binary)""Calculated checksum"	
	Copy example without framing (Binary)	

Table 19: Telegram structure: "Command type" "Command" (Answer)

Telegram structure: "Command type" "Command"						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Lists the different parts of the telegram.	Describes the corresponding telegram parts.	Defines the type of the variable.	Defines the length in byte.	Gives further information regarding the values in CoLa A/ CoLa B if neccessary.	Defines the value of the telegram part in CoLa A (ASCII).	Defines the value of the telegram part in CoLa B (Binary).

Table 20: Example: "Command type" "Command" (Answer)

CoLa A	<"Start of text">"Command type value (ASCII)""space""Command value (ASCII)""space""Parameter value (ASCII)""space""Parameter value (ASCII)"<"End of text">
	<"Start of text">"Command type value (Hex)""space""Command value (Hex)""space""Parameter value (Hex)""space""Parameter value (Hex)"<"End of text">
CoLa B	"Start of text and given length""Command type value (Binary)""space""Command value (Binary)""space""Parameter value (Binary)""Parameter value (Binary)""Calculated checksum"

12.4.1.4.1 Log in [sMN SetAccessMode]

NOTE

- An automated hash-value calculator can be found in SOPAS ET under menu “password”. Required userlevel 'Service' (see "Change password [sMN SetPassword]", page 81).

A log in to the device is necessary to change parameters. In most cases, the user level 'Authorized client' is needed. Changed paramters will be reset to the previous state via a reboot unless the are saved. To save paramter changes the command "sMN mEEwriteall" (see "Save parameters permanently [sMN mEEwriteall]", page 90) must be send before log out.

Table 21: Telegram structure: sMN SetAccessMode

Telegram structure: sMN SetAccessMode						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E
Command	User level	String	13		SetAccessMode	53 65 74 41 63 63 65 73 73 4D 6F 64 65

Telegram structure: sMN SetAccessMode						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
User level	Select user level	Int_8	1	Maintenance: Authorized client: Service:	2 3 4	02 03 04
Password	Hash value for the selected user level	Uint_32	4	Maintenance: Authorized client: Service:	B21ACE26 F4724744 81BE23AA	B2 1A CE 26 F4 72 47 44 81 BE 23 AA

Table 22: Example: sMN SetAccessMode - Log in as "Authorized client" with password "F4724744"

CoLa A	<STX>sMN{SPC}SetAccessMode{SPC}3{SPC}F4724744<ETX>
	<STX>sMN SetAccessMode 3 F4724744<ETX>
	sMN SetAccessMode 3 F4724744
	02 73 4D 4E 20 53 65 74 41 63 63 65 73 73 4D 6F 64 65 20 30 33 20 46 34 37 32 34 37 34 34 03
CoLa B	02 02 02 02 00 00 00 17 73 4D 4E 20 53 65 74 41 63 63 65 73 73 4D 6F 64 65 20 03 F4 72 47 44 B3
	73 4D 4E 20 53 65 74 41 63 63 65 73 73 4D 6F 64 65 20 03 F4 72 47 44

Table 23: Telegram structure: sAN SetAccessMode

Telegram structure: sAN SetAccessMode						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 41 4E
Command	User level	String	13		SetAccessMode	53 65 74 41 63 63 65 73 73 4D 6F 64 65
Change user level	Changed level	Bool_1	1	Error: Success:	0 1	00 01

Table 24: Example: sAN SetAccessMode

CoLa A	<STX>sAN{SPC}SetAccessMode{SPC}1<ETX>
	02 73 41 4E 20 53 65 74 41 63 63 65 73 73 4D 6F 64 65 20 31 03
CoLa B	02 02 02 02 00 00 00 13 73 41 4E 20 53 65 74 41 63 63 65 73 73 4D 6F 64 65 20 01 38

12.4.1.4.2 Basic Settings

12.4.1.4.2.1 Read for frequency and angular resolution [sRN LMPscancfg]

Read the scanning frequency, angular resolution and aperture angle of the device. Values show the information of the values before any filters are applied. Therefore aperature angle is always showing the maximum, scanning frequency and angular resolution may also be static depending on device family.

Table 25: Telegram structure: sRN LMPscancfg

Telegram structure: sRN LMPscancfg						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Info of scan frequency and angular resolution	String	10		LMPscancfg	4C 4D 50 73 63 61 6E 63 66 67

Table 26: Example: sRN LMPscancfg

CoLa A	<STX>sRN{SPC}LMPscancfg<ETX>	
	<STX>sRN LMPscancfg<ETX>	
	sRN LMPscancfg	
	02 73 52 4E 20 4C 4D 50 73 63 61 6E 63 66 67 03	
CoLa B	02 02 02 02 00 00 00 0E 73 52 4E 20 4C 4D 50 73 63 61 6E 63 66 67 63	
	73 52 4E 20 4C 4D 50 73 63 61 6E 63 66 67	

Table 27: Telegram structure: sRA LMPscancfg

Telegram structure: sRA LMPscancfg						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Info of scan frequency and angular resolution	String	10		LMPscancfg	4C 4D 50 73 63 61 6E 63 66 67
Scan frequency	[1/100 Hz]	Uint_32	4	20 Hz:	7D0h	00 00 07 D0
Reserved	-	Int_16	2	Always:	1	00 01
Angular resolution	[1/10000°]	Uint_32	4	0.125°:	4E2h	00 00 04 E2
Start angle	[1/10000°]	Int_32	4	-180° ... +180°	FFE488C0h ... 1B7740h	FF E4 88 C0 ... 00 1B 77 40
Stop angle	[1/10000°]	Int_32	4	-180° ... +180°	FFE488C0h ... 1B7740h	FF E4 88 C0 ... 00 1B 77 40

Table 28: Example: sRA LMPscancfg

CoLa A	<STX>sRA{SPC}LMPscancfg{SPC}7D0{SPC}1{SPC}4E2{SPC}FFF24460{SPC}DBBA0<ETX>
	02 73 52 41 20 4C 4D 50 73 63 61 6E 63 66 67 20 37 44 30 20 31 20 34 45 32 20 46 46 46 32 34 34 36 30 20 44 42 42 41 30 03
CoLa B	02 02 02 02 00 00 00 21 73 52 41 20 4C 4D 50 73 63 61 6E 63 66 67 20 00 00 07 D0 00 01 00 00 04 E2 FF F2 44 60 00 0D BB A0 43

12.4.1.4.2.2 Set scan configuration [sMN mCLsetscancfglist]

Sets the device to a defined scan configuration, consisting of scan frequency, angular resolution, sector definition and interlace mode.

Table 29: Telegram structure: sMN mCLsetscancfglist

Telegram structure: sMN mCLsetscancfglist						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E
Command	Set scan configuration	String	17		mCLsetscancfg-list	6D 43 4C 73 65 74 73 63 61 6E 63 66 67 6C 69 73 74
Mode	Interlace mode (see table below)	Enum_8	1	Off: On:	0 1	00 01

Interlace mode

The interlace mode allows to achieve a higher angular resolution by combining scans with lower resolution. The individual scans are shifted to each other.

Switching table:

The device has 16 layers, two layers (layer 6,14) are high resolution layers with 0.125°, the other layers do have a resolution of 1°. If the iterlaced mode is active the 1° layers will be shifted by 0.125° for each scan. After 8 scans you will have a theoretical resolution of 0.125° on all layers.

Mode	Mode Name
0	Interlaced off
1	Interlaced on

Table 30: Example: Set scan configuration 1: sMN mCLsetscancfglist 1

CoLa A	<STX>sMN{SPC}mCLsetscancfglist{SPC}1<ETX>	
	<STX>sMN mCLsetscancfglist 1<ETX>	
	sMN mCLsetscancfglist 1	
	02 73 4D 4E 20 6D 43 4C 73 65 74 73 63 61 6E 63 66 67 6C 69 73 74 20 31 03	
CoLa B	02 02 02 02 00 00 00 17 20 73 4D 4E 20 6D 43 4C 73 65 74 73 63 61 6E 63 66 67 6C 69 73 74 20 01 0E	
	73 4D 4E 20 6D 43 4C 73 65 74 73 63 61 6E 63 66 67 6C 69 73 74 20 01	

Table 31: Telegram structure: sAN mCLsetscancfglist

Telegram structure: sAN mCLsetscancfglist						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 41 4E
Command	Set scan configuration	String	17		mCLsetscancfg-list	6D 43 4C 73 65 74 73 63 61 6E 63 66 67 6C 69 73 74

Telegram structure: sAN mCLsetscancfglist						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Status code	Wrong setting	Enum_8	1	Ok: Error frequency: Error resolution: Error resolution and frequency: Error scan field: Error:	0 1 2 3 4 5	00 01 02 03 04 05

Table 32: Example: sAN mCLsetscancfglist Ok

CoLa A	<STX>sAN{SPC}mCLsetscancfglist{SPC}0<ETX>
	02 73 41 4E 20 6D 43 4C 73 65 74 73 63 61 6E 63 66 67 6C 69 73 74 20 30 03
CoLa B	02 02 02 02 00 00 00 17 73 41 4E 20 6D 43 4C 73 65 74 73 63 61 6E 63 66 67 6C 69 73 74 20 00 03

12.4.1.4.2.3 Start measurement [sMN LMCstartmeas]

Start the laser and (unless in Standby mode) the motor of the device

Table 33: Telegram structure: sMN LMCstartmeas

Telegram structure: sMN LMCstartmeas (User level 'Authorized Client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E
Command	Start measurement	String	12		LMCstartmeas	4C 4D 43 73 74 61 72 74 6D 65 61 73

Table 34: Example: sMN LMCstartmeas

CoLa A	<STX>sMN{SPC}LMCstartmeas<ETX>	
	<STX>sMN LMCstartmeas<ETX>	
	sMN LMCstartmeas	
	02 73 4D 4E 20 4C 4D 43 73 74 61 72 74 6D 65 61 73 03	
CoLa B	02 02 02 02 00 00 00 10 73 4D 4E 20 4C 4D 43 73 74 61 72 74 6D 65 61 73 68	
	73 4D 4E 20 4C 4D 43 73 74 61 72 74 6D 65 61 73	

Table 35: Telegram structure: sAN LMCstartmeas

Telegram structure: sAN LMCstartmeas						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 41 4E
Command	Start measurement	String	12		LMCstartmeas	4C 4D 43 73 74 61 72 74 6D 65 61 73
Status code	Accepted when value is 0	Enum_8	1	Success: Not allowed:	0 1	00 01

Table 36: Example: sAN LMCstartmeas

CoLa A	<STX>sAN{SPC}LMCstartmeas{SPC}0<ETX>
	02 73 41 4E 20 4C 4D 43 73 74 61 72 74 6D 65 61 73 20 30 03
CoLa B	02 02 02 02 00 00 00 12 73 41 4E 20 4C 4D 43 73 74 61 72 74 6D 65 61 73 20 00 44

12.4.1.4.2.4 Stop measurement [sMN LMCstopmeas]

Shut off the laser and the motor is running at the set up frequency.

Table 37: Telegram structure: sMN LMCstopmeas

Telegram structure: sMN LMCstopmeas (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E
Command	Stop measurement	String	11		LMCstopmeas	4C 4D 43 73 74 6F 70 6D 65 61 73

Table 38: Example: sMN LMCstopmeas

CoLa A	<STX>sMN{SPC}LMCstopmeas<ETX>	
	<STX>sMN LMCstopmeas<ETX>	
	sMN LMCstopmeas	
	02 73 4D 4E 20 4C 4D 43 73 74 6F 70 6D 65 61 73 03	
CoLa B	02 02 02 02 00 00 00 0F 73 4D 4E 20 4C 4D 43 73 74 6F 70 6D 65 61 73 10	
	73 4D 4E 20 4C 4D 43 73 74 6F 70 6D 65 61 73	

Table 39: Telegram structure: sAN LMCstopmeas

Telegram structure: sAN LMCstopmeas						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 41 4E
Command	Stop measurement	String	11		LMCstopmeas	4C 4D 43 73 74 6F 70 6D 65 61 73
Status code	Accepted when value is 0	Enum_8	1	No error: Not allowed:	0 1	00 01

Table 40: Example: sAN LMCstopmeas

CoLa A	<STX>sAN{SPC}LMCstopmeas{SPC}0<ETX>
	02 73 41 4E 20 4C 4D 43 73 74 6F 70 6D 65 61 73 20 30 03
CoLa B	02 02 02 02 00 00 00 11 73 41 4E 20 4C 4D 43 73 74 6F 70 6D 65 61 73 20 00 3C

12.4.1.4.2.5 Autostart measurement [sWN LMPautostartmeas]

Table 41: Telegram structure: sWN LMPautostartmeas

Telegram structure: sWN LMPautostartmeas (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Autostart measurement	String	16		LMPautostart-meas	4C 4D 50 61 75 74 6F 73 74 61 72 74 6D 65 61 73
Status code	Activate / Deactivate Autostart	Bool_1	1	Autostart off: Autostart on:	0 1	00 01

Table 42: Example: sWN LMPautostartmeas 1

CoLa A	<STX>sWN{SPC}LMPautostartmeas{SPC}1<ETX>	
	<STX>sWN LMPautostartmeas 1<ETX>	
	sWN LMPautostartmeas 1	
	02 73 57 4E 20 4C 4D 50 61 75 74 6F 73 74 61 72 74 6D 65 61 73 20 31 03	
CoLa B	02 02 02 02 00 00 00 16 73 57 4E 20 4C 4D 50 61 75 74 6F 73 74 61 72 74 6D 65 61 73 20 01 4F	
	73 57 4E 20 4C 4D 50 61 75 74 6F 73 74 61 72 74 6D 65 61 73 20 01	

This parameter defines whether the scanner will start directly rotate and measure when powering up or remain in idle mode. The changed setting (saved with the command `sMN mEEWriteall`, see "Save parameters permanently [sMN mEEwriteall]", page 90) will be then be active with the next power-up cycle.

Table 43: Telegram structure: sWA LMDautostartmeas

Telegram structure: sWA LMPautostartmeas						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Autostart measurement	String	16		LMPautostart-meas	4C 4D 50 61 75 74 6F 73 74 61 72 74 6D 65 61 73

Table 44: Example: sWA LMPautostartmeas

CoLa A	<STX>sWA{SPC}LMPautostartmeas<ETX>
	02 73 57 41 20 4C 4D 43 73 74 61 72 74 6D 65 61 73 03
CoLa B	02 02 02 02 00 00 00 15 73 57 41 20 4C 4D 50 61 75 74 6F 73 74 61 72 74 6D 65 61 73 20 41

12.4.1.4.2.6 Load factory defaults [sMN mSCloadfacdef]

NOTE

The Factory-Reset (Load factory defaults) deletes the entire parametrization of the device. All parameters, settings and system applications will be set to default.

Table 45: Telegram structure: sMN mSCloadfacdef

Telegram structure: sMN mSCloadfacdef (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E
Command	Load factory defaults	String	13		mSCloadfacdef	6D 53 43 6C 6F 61 64 66 61 63 64 65 66

Table 46: Example: sMN mSCloadfacdef

CoLa A	<STX>sMN{SPC}mSCloadfacdef<ETX>	
	<STX>sMN mSCloadfacdef<ETX>	
	sMN mSCloadfacdef	
	02 73 4D 4E 20 6D 53 43 6C 6F 61 64 66 61 63 64 65 66 03	
CoLa B	02 02 02 02 00 00 00 11 73 4D 4E 20 6D 53 43 6C 6F 61 64 66 61 63 64 65 66 28	
	73 4D 4E 20 6D 53 43 6C 6F 61 64 66 61 63 64 65 66	

Table 47: Telegram structure: sAN mSCloadfacdef

Telegram structure: sAN mSCloadfacdef						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 41 4E
Command	Load factory defaults	String	13		mSCloadfacdef	6D 53 43 6C 6F 61 64 66 61 63 64 65 66

Table 48: Example: sAN mSCloadfacdef

CoLa A	<STX>sAN{SPC}mSCloadfacdef<ETX>
	02 73 4D 4E 20 6D 53 43 6C 6F 61 64 66 61 63 64 65 66 03
CoLa B	02 02 02 02 00 00 00 12 73 41 4E 20 6D 53 43 6C 6F 61 64 66 61 63 64 65 66 20 04

12.4.1.4.2.7 Load application defaults [sMN mSCloadappdef]

NOTE

The Application-Reset (Load application defaults) deletes only the user parametrization of the Fields, Evaluation cases (EVC) and parameters under the header “Application”. Other parameters like Interface settings, Echo Filter, etc. remain unaffected.

Table 49: Telegram structure: sMN mSCloadappdef

Telegram structure: sMN mSCloadappdef (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E

Telegram structure: sMN mSCloadappdef (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command	Load application defaults	String	13		mSCloadappdef	6D 53 43 6C 6F 61 64 61 70 70 64 65 66

Table 50: Example: sMN mSCloadappdef

CoLa A	<STX>sMN{SPC}mSCloadappdef<ETX>	
	<STX>sMN mSCloadappdef<ETX>	
	sMN mSCloadappdef	
	02 73 4D 4E 20 6D 53 43 6C 6F 61 64 61 70 70 64 65 66 03	
CoLa B	02 02 02 02 00 00 00 11 73 4D 4E 20 6D 53 43 6C 6F 61 64 61 70 70 64 65 66 2D	
	73 4D 4E 20 6D 53 43 6C 6F 61 64 61 70 70 64 65 66	

Table 51: Telegram structure: sAN mSCloadappdef

Telegram structure: sAN mSCloadappdef						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 41 4E
Command	Load application defaults	String	13		mSCloadappdef	6D 53 43 6C 6F 61 64 61 70 70 64 65 66

Table 52: Example: sAN mSCloadappdef

CoLa A	<STX>sAN{SPC}mSCloadappdef<ETX>
	02 73 41 4E 20 6D 53 43 6C 6F 61 64 61 70 70 64 65 66 03
CoLa B	02 02 02 02 00 00 00 12 73 41 4E 20 6D 53 43 6C 6F 61 64 61 70 70 64 65 66 20 01

12.4.1.4.2.8 Change password [sMN SetPassword]

NOTE
If logged in with a higher user level you may set the password for lower user levels as well.

Changing the log in password for a specific user level.

Table 53: Telegram structure: sMN SetPassword

Telegram structure: sMN SetPassword (Same user level or higher required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E
Command	Set password request	String	13		SetPassword	53 65 74 50 61 73 73 77 6F 72 64

Telegram structure: sMN SetPassword (Same user level or higher required)						
Telegram part	Description	Variable	Length	Additional details	Values CoLa A (ASCII)	Values CoLa B (Binary)
User level	User level that the password will be applied to	Int_8	1	Maintenance: Authorized client: Service:	2 3 4	02 03 04
Password	Hash value of the new password	Uint_32	4		<Hash value>	

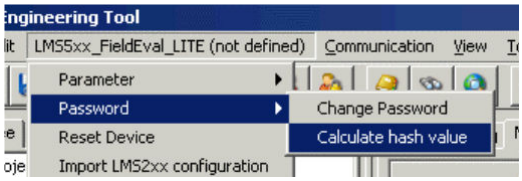
Set password for 'Authorized client' to “testtest” (hash value = 1920E4C9).

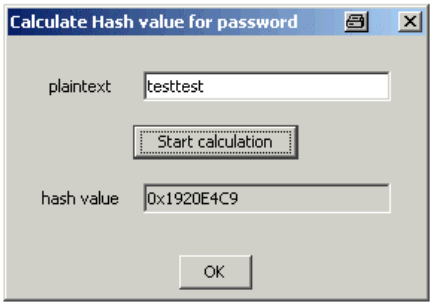
Table 54: Example: sMN SetPassword

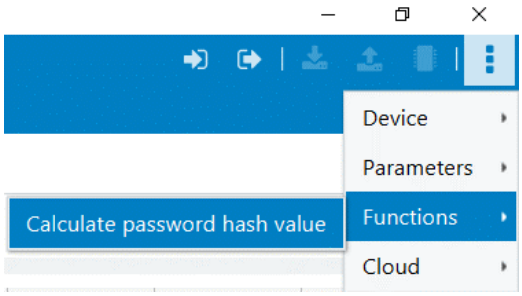
CoLa A	<STX>sMN{SPC}SetPassword{SPC}3{SPC}1920E4C9<ETX>	
	<STX>sMN SetPassword 3 1920E4C9<ETX>	
	sMN SetPassword 3 1920E4C9	
	02 73 4D 4E 20 53 65 74 50 61 73 73 77 6F 72 64 20 33 20 31 39 32 30 45 34 43 39 03	
CoLa B	02 02 02 02 00 00 00 15 73 4D 4E 20 53 65 74 50 61 73 73 77 6F 72 64 20 03 19 20 E4 C9 1A	
	73 4D 4E 20 53 65 74 50 61 73 73 77 6F 72 64 20 03 19 20 E4 C9	

Calculating the hash value of the password

- ► Login in SOPAS ET with user level “Service” to the device.
- ► Select **[Device]** > **Password** > **Calculate Hash value.**
- ► Alternatively select > **Functions** > **Calculate password hash value**







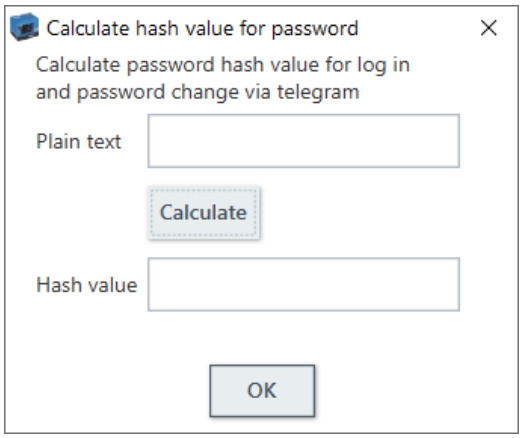


Table 55: Telegram structure: sAN SetPassword

Telegram structure: sAN SetPassword						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 41 4E
Command	Set password requested	String	13		SetPassword	53 65 74 50 61 73 73 77 6F 72 64
Success	Confirmation	Int_8	1	Failed: Success:	0 1	00 01

Table 56: Example: sAN SetPassword

CoLa A	<STX>sAN{SPC}SetPassword{SPC}1<ETX>
	02 73 4D 4E 20 53 65 74 50 61 73 73 77 6F 72 64 20 31 03
CoLa B	02 02 02 02 00 00 00 11 73 41 4E 20 53 65 74 50 61 73 73 77 6F 72 64 20 01 00

12.4.1.4.2.9 Check password [sMN CheckPassword]

Table 57: Telegram structure: sMN CheckPassword

Telegram structure: sMN CheckPassword (Same User level or higher required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E
Command	Check password request	String	13		CheckPassword	43 68 65 63 6B 50 61 73 73 77 6F 72 64
User level	User level to check the password for	Int_8	1	Maintenance: Authorized client: Service:	2 3 4	02 03 04
Password	Hash value of the password to be checked	Uint_32	4		<Hash value>	

Check password “testtest” for 'Authorized client'.

Table 58: Example: sMN CheckPassword

CoLa A	<STX>sMN{SPC}CheckPassword{SPC}3{SPC}1920E4C9<ETX>	
	<STX>sMN CheckPassword 3 1920E4C9<ETX>	
	sMN CheckPassword 3 1920E4C9	
	02 73 4D 4E 20 43 68 65 63 6B 50 61 73 73 77 6F 72 64 20 33 20 31 39 32 30 45 34 43 39 03	
CoLa B	02 02 02 02 00 00 00 17 73 4D 4E 20 43 68 65 63 6B 50 61 73 73 77 6F 72 64 20 03 19 20 E4 C9 1E	
	73 4D 4E 20 43 68 65 63 6B 50 61 73 73 77 6F 72 64 20 03 19 20 E4 C9	

Table 59: Telegram structure: sAN CheckPassword

Telegram structure: sAN CheckPassword						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 41 4E
Command	Check password request	String	13		CheckPassword	43 68 65 63 6B 50 61 73 73 77 6F 72 64
Success	Confirmation	Int_8	1	Failed: Success:	0 1	00 01

Table 60: Example: sAN CheckPassword

CoLa A	<STX>sAN{SPC}CheckPassword{SPC}1<ETX>
	02 73 41 4E 20 43 68 65 63 6B 50 61 73 73 77 6F 72 64 20 31 03
CoLa B	02 02 02 02 00 00 00 13 73 41 4E 20 43 68 65 63 6B 50 61 73 73 77 6F 72 64 20 01 04

12.4.1.4.2.10 Set contamination indication settings [sWN ContaminationConfig]

Table 61: Telegram structure: sWN ContaminationConfig

Telegram structure: sWN ContaminationConfig (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command		String	19		Contamination-Config	43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 43 6F 6E 66 69 67
Strategy	Strategy code	Enum_8	1	Inactive: High available: Sensitive:	0 1 2	00 01 02
Response time	Error / warning reaction time in sec	Uint_16	2	Value Range 3 ... 60 Initialisation 3	+3d ... +60d (03h ... 3Ch)	00 03 ... 00 3C
Threshold warning	Threshold value	Enum_8	1	LOW: MID: HIGH:	0 1 2	00 01 02
Cover	Selection of either custom sectors or used wheather protection hood	Enum_8	1	No weather protection hood: Weather protection hood custom sectors:	0 255d (FFh)	00 FF

Telegram structure: sWN ContaminationConfig (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Custom sectors	Array length is 12	Array	2	Default value for all sectors = true	1 1 1 1 1 1 1 1 1 1 1 1	01 01 01 01 01 01 01 01 01 01 01 01
Enable Warning	Warning monitoring	Bool_1		Off (False): On (True):	0 1	00 01
Enable Error	Error monitoring	Bool_1		Off (False): On (True):	0 1	00 01

Table 62: Example: sWN ContaminationConfig strategy inactive, response time 3 sec, threshold low, Weather protection hood custom sectors, sector 1-12, enable warning, enable error

CoLa A	<STX>sWN{SPC}ContaminationConfig{SPC}0{SPC}3{SPC}0{SPC}FF{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1<ETX>
	02 73 57 4E 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 43 6F 6E 66 69 67 20 30 20 33 20 30 20 46 46 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 03
CoLa B	02 02 02 02 00 00 00 2B 73 57 4E 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 43 6F 6E 66 69 67 20 00 00 03 00 FF 01 01 01 01 01 01 01 01 01 01 01 01 01 FC

Table 63: Telegram structure: sWA ContaminationConfig

Telegram structure: sWA ContaminationConfig						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command		String	19		Contamination-Config	43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 43 6F 6E 66 69 67

Table 64: Example: sWA ContaminationConfig

CoLa A	<STX>sWA{SPC}ContaminationConfig<ETX>
	02 73 57 41 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 43 6F 6E 66 69 67 03
CoLa B	02 02 02 02 00 00 00 18 73 57 41 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 43 6F 6E 66 69 67 20

12.4.1.4.2.11 Read contamination indication settings [sRN ContaminationConfig]

Table 65: Telegram structure: sRN ContaminationConfig

Telegram structure: sRN ContaminationConfig						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command		String	19		Contamination-Config	43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 43 6F 6E 66 69 67

Table 66: Example: sRN ContaminationConfig

CoLa A	<STX>sRN{SPC}ContaminationConfig<ETX>	
	<STX>sRN ContaminationConfig<ETX>	
	sRN ContaminationConfig	
	02 73 52 4E 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 43 6F 6E 66 69 67 03	
CoLa B	02 02 02 02 00 00 00 17 73 52 4E 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 43 6F 6E 66 69 67 25	
	73 52 4E 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 43 6F 6E 66 69 67 25	

Table 67: Telegram structure: sRA ContaminationConfig

Telegram structure: sRA ContaminationConfig						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command		String	19		Contamination-Config	43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 43 6F 6E 66 69 67
Strategy	Strategy code	Enum_8	1	Inactive: High available: Sensitive:	0 1 2	00 01 02
Response time	Error / warning reaction time in sec	Uint_16	2	Value Range 3 ... 60 Initialisation 3	+3d ... +60d (03h ... 3Ch)	00 03 ... 00 3C
Threshold warning	Threshold value	Enum_8	1	LOW: MID: HIGH:	0 1 2	00 01 02
Cover	Selection of either custom sectors or used wheather protection hood	Enum_8	1	No weather protection hood: Weather protection hood custom sectors:	0 FF	00 FF

Telegram structure: sRA ContaminationConfig						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Custom sectors	Array length is 12	Array	2	Default value for all sectors = true	1 1 1 1 1 1 1 1 1 1	01 01 01 01 01 01 01 01 01 01
Enable Warning	Warning monitoring	Bool_1		Off (False): On (True):	0 1	00 01
Enable Error	Error monitoring	Bool_1		Off (False): On (True):	0 1	00 01

Table 68: Example: sRA ContaminationConfig strategy inactive, response time 3 sec, threshold low, Weather protection hood custom sectors, sector 1-12, enable warning, enable error

CoLa A	<STX>sRA{SPC}ContaminationConfig{SPC}0{SPC}3{SPC}0{SPC}FF{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1<ETX>
	02 73 52 41 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 43 6F 6E 66 69 67 20 30 20 33 20 30 20 46 46 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 03
CoLa B	02 02 02 02 00 00 00 2B 73 52 41 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 43 6F 6E 66 69 67 20 00 00 03 00 FF 01 01 01 01 01 01 01 01 01 01 01 01 01 01 F6

12.4.1.4.2.12 Send contamination indication data permanently [sEN ContaminationData]

Table 69: Telegram structure: sEN ContaminationData

Telegram structure: sEN ContaminationData						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sEN	73 45 4E
Command		String	17		ContaminationData	43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 44 61 74 61
Data	Start/ Stop	Bool_1	1	Stop: Start:	0 1	00 01

Table 70: Example: sEN ContaminationData

CoLa A	<STX>sEN{SPC}ContaminationData{SPC}1<ETX>	
	<STX>sRN ContaminationData 1<ETX>	
	sRN ContaminationData 1	
	02 73 45 4E 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 44 61 74 61 20 31 03	
CoLa B	02 02 02 02 00 00 00 17 73 52 4E 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 44 61 74 61 20 01 09	
	73 52 4E 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 44 61 74 61 20 01	

Table 71: Telegram structure: sSN ContaminationData

Telegram structure: sSN ContaminationData						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sSN	73 53 4E
Command		String	17		Contamination-Data	43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 44 61 74 61
Contamination data for different sectors	Status of contamination in order of the different sectors	Array of Enum_8	12	Deactivated: Clean: Warning: Error:	0 1 2 3	00 01 02 03

Table 72: Example: sSN ContaminationData

CoLa A	<STX>sSN{SPC}ContaminationData{SPC}0{SPC}3{SPC}2{SPC}2{SPC}2{SPC}2{SPC}2{SPC}2{SPC}2{SPC}2{SPC}2<ETX>
	02 73 53 4E 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 44 61 74 61 20 30 20 33 20 32 20 32 20 32 20 32 20 3220 32 20 3220 3220 3220 32 03
CoLa B	02 02 02 02 00 00 00 22 73 53 4E 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 44 61 74 61 20 00 03 02 02 02 02 02 02 02 02 02 02 02 1D

12.4.1.4.2.13 Read contamination indication result [sRN ContaminationResult]

Table 73: Telegram structure: sRN ContaminationResult

Telegram structure: sRN ContaminationResult						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command		String	19		Contamination-Result	43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 52 65 73 75 6C 74

Table 74: Example: sRN ContaminationResult

CoLa A	<STX>sRN{SPC}ContaminationResult<ETX>	
	<STX>sRN ContaminationResult<ETX>	
	sRN ContaminationResult	
	02 73 52 4E 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 52 65 73 75 6C 74 03	

CoLa B	02 02 02 02 00 00 00 17 73 52 4E 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 52 65 73 75 6C 74 26
	73 52 4E 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 52 65 73 75 6C 74

Table 75: Telegram structure: sRA ContaminationResult

Telegram structure: sRA ContaminationResult						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command		String	19		Contamination-Result	43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 52 65 73 75 6C 74
Contamination Result	Result of contamination indication (Order of results: Warning / Error)	Enum_8	1	Warning Inactive: Active:	 0 1	 00 01
				Error Inactive: Active:	 0 1	 00 01

Table 76: Example: sRA ContaminationResult warning active error inactive

CoLa A	<STX>sRA{SPC}ContaminationResult{SPC}1{SPC}0<ETX>
	02 73 52 41 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 52 65 73 75 6C 74 20 31 20 30 03
CoLa B	02 02 02 02 00 00 00 1A 73 52 41 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 52 65 73 75 6C 74 20 01 00 08

12.4.1.4.2.14 Send contamination indication result permanently [sEN ContaminationResult]

This telegram is intended to activate the event for read the contamination result permanently. The telegram returns two Boolean. One for warning and one for error. Depending on the strategy of the contamination indication the state changes if one or all sectors are in the specific status (warning or error).

Table 77: Telegram structure: sEN ContaminationResult

Telegram structure: sEN ContaminationResult						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sEN	73 52 4E
Command		String	19		Contamination-Result	43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 52 65 73 75 6C 74
Data	Start/ Stop	Bool_1	1	Stop: Start:	0 1	00 01

Table 78: Example: sEN ContaminationResult

CoLa A	<STX>sEN{SPC}ContaminationResult{SPC}1<ETX>
	<STX>sEN ContaminationResult 1<ETX>
	sEN ContaminationResult 1
	02 73 45 4E 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 52 65 73 75 6C 74 20 31 03
CoLa B	02 02 02 02 00 00 00 19 73 45 4E 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 52 65 73 75 6C 74 20 01 10
	73 45 4E 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 52 65 73 75 6C 74 20 01

Table 79: Telegram structure: sSN ContaminationResult

Telegram structure: sSN ContaminationResult						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sSN	73 53 4E
Contamination Result	Result of contamination indication (Order of results: Warning / Error)	Enum_8	1	Warning Inactive: Active:	 0 1	 00 01
				Error Inactive: Active:	 0 1	 00 01

Table 80: Example: sSN ContaminationResult warning active error inactive

CoLa A	<STX>sRA{SPC}ContaminationResult{SPC}1{SPC}0<ETX>
	02 73 52 41 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 52 65 73 75 6C 74 20 31 20 30 03
CoLa B	02 02 02 02 00 00 00 1A 73 52 41 20 43 6F 6E 74 61 6D 69 6E 61 74 69 6F 6E 52 65 73 75 6C 74 20 01 00 08

12.4.1.4.2.15 Save parameters permanently [sMN mEEwriteall]

Table 81: Telegram structure: sMN mEEwriteall

Telegram structure: sMN mEEwriteall (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E
Command	Store parameters permanently	String	11		mEEwriteall	6D 45 45 77 72 69 74 65 61 6C 6C

Table 82: Example: sMN mEEwriteall

CoLa A	<STX>sMN{SPC}mEEwriteall<ETX>
	<STX>sMN mEEwriteall<ETX>
	sMN mEEwriteall
	02 73 4D 4E 20 6D 45 45 77 72 69 74 65 61 6C 6C 03
CoLa B	02 02 02 02 00 00 00 0F 73 4D 4E 20 6D 45 45 77 72 69 74 65 61 6C 6C 21
	73 4D 4E 20 6D 45 45 77 72 69 74 65 61 6C 6C

Table 83: Telegram structure: sAN mEEwriteall

Telegram structure: sAN mEEwriteall						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 41 4E
Command	Store parameters permanently	String	11		mEEwriteall	6D 45 45 77 72 69 74 65 61 6C 6C
Status code	Accepted when value is 1	Bool_1	1	Error: Success:	0 1	00 01

Table 84: Example: sAN mEEwriteall

CoLa A	<STX>sAN{SPC}mEEwriteall{SPC}1<ETX>
	02 73 41 4E 20 6D 45 45 77 72 69 74 65 61 6C 6C 20 31 03
CoLa B	02 02 02 02 00 00 00 11 73 41 4E 20 6D 45 45 77 72 69 74 65 61 6C 6C 20 01 0C

12.4.1.4.2.16 Set to run [sMN Run]

Log out from device and activate all parameter changes.

Table 85: Telegram structure: sMN Run

Telegram structure: sMN Run						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E
Command	Start the device	String	3		Run	52 75 6E

Table 86: Example: sMN Run

CoLa A	<STX>sMN{SPC}Run<ETX>	
	<STX>sMN Run<ETX>	
	sMN Run	
	02 73 4D 4E 20 52 75 6E 03	
CoLa B	02 02 02 02 00 00 00 07 73 4D 4E 20 52 75 6E 19	
	73 4D 4E 20 52 75 6E	

Table 87: Telegram structure: sAN Run

Telegram structure: sAN Run						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 41 4E
Status code	Accepted when value is 1	Bool_1	1	Error: Success:	0 1	00 01

Table 88: Example: sAN Run

CoLa A	<STX>sAN{SPC}Run{SPC}1<ETX>
	02 73 41 4E 20 52 75 6E 20 31 03
CoLa B	02 02 02 02 00 00 00 09 73 41 4E 20 52 75 6E 20 01 34

12.4.1.4.2.17 Reboot device [sMN mSCreboot]

This command includes saving all parameters.

Table 89: Telegram structure: sMN mSCreboot

Telegram structure: sMN mSCreboot (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E
Command	Reboot device	String	9		mSCreboot	6D 53 43 72 65 62 6F 6F 74

Table 90: Example: sMN mSCreboot

CoLa A	<STX>sMN{SPC}mSCreboot<ETX>	
	<STX>sMN mSCreboot<ETX>	
	sMN mSCreboot	
	02 73 4D 4E 20 6D 53 43 72 65 62 6F 6F 74 03	
CoLa B	02 02 02 02 00 00 00 0D 73 4D 4E 20 6D 53 43 72 65 62 6F 6F 74 2C	
	73 4D 4E 20 6D 53 43 72 65 62 6F 6F 74	

Table 91: Telegram structure: sAN mSCreboot

Telegram structure: sAN mSCreboot						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 41 4E
Command	Reboot device	String	9		mSCreboot	6D 53 43 72 65 62 6F 6F 74

Table 92: Example: sAN mSCreboot

CoLa A	<STX>sAN{SPC}mSCreboot<ETX>
	02 73 41 4E 20 6D 53 43 72 65 62 6F 6F 74 03
CoLa B	02 02 02 02 00 00 00 0E 73 41 4E 20 6D 53 43 72 65 62 6F 6F 74 20 00

12.4.1.4.3 Measurement output telegram

12.4.1.4.3.1 Configure aperture angle of the scandata for output [sWN LMPoutputRange]

NOTE
Verify the definition of the angle positions for your product.

Select start and stop angle of the measurement data output. In general only one output range can be be configured.

Table 93: Telegram structure: sWN LMPoutputRange

Telegram structure: sWN LMPoutputRange (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional Details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Change output angle range	String	14		LMPoutpu-tRange	4C 4D 50 6F 75 74 70 75 74 52 61 6E 67 65
Reserved	-	Int_16	2	Always:	1	00 01
Angular resolu-tion [1)]	[1/10000°] Fixed value, not changeable	Uint_32	4	0.125°:	+1250d (4E2h)	00 00 04 E2
Start angle	[1/10000°]	Int_32	4	-180° ... +180°	-1800000d (FFE488C0h) ... +1800000d (1B7740h)	FF E4 88 C0 ... 00 1B 77 40
Stop angle	[1/10000°]	Int_32	4	-180° ... +180°	-1800000d (FFE488C0h) ... +1800000d (1B7740h)	FF E4 88 C0 ... 00 1B 77 40

1) Angular resolution can not be changed here, it is taken automatically from the basic scan settings!

Table 94: Example: sWN LMPoutputRange - set output data for angular resolution at 0.125° and range from -90° to +90°

CoLa A	<STX>sWN{SPC}LMPoutputRange{SPC}1{SPC}+1250{SPC}-900000{SPC}+900000<ETX>	
	<STX>sWN LMPoutputRange 1 +1250 -900000 +900000<ETX>	
	sWN LMPoutputRange 1 +1250 -900000 +900000	
	02 73 57 4E 20 4C 4D 50 6F 75 74 70 75 74 52 61 6E 67 65 20 31 20 2B 31 32 35 30 20 2D 39 30 30 30 30 30 20 2B 39 30 30 30 30 30 03	
CoLa B	02 02 02 02 00 00 00 21 73 57 4E 20 4C 4D 50 6F 75 74 70 75 74 52 61 6E 67 65 20 00 01 00 00 04 E2 FF F2 44 60 00 0D BB A0 A3	
	73 57 4E 20 4C 4D 50 6F 75 74 70 75 74 52 61 6E 67 65 20 00 01 00 00 04 E2 FF F2 44 60 00 0D BB A0	

Table 95: Telegram structure: sWA LMPoutputRange

Telegram structure: sWA LMPoutputRange						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Change output angle range	String	14		LMPoutpu-tRange	4C 4D 50 6F 75 74 70 75 74 52 61 6E 67 65

Table 96: Example: sWA LMPoutputRange

CoLa A	<STX>sWA{SPC}LMPoutputRange<ETX>
	02 73 57 41 20 4C 4D 50 6F 75 74 70 75 74 52 61 6E 67 65 03
CoLa B	02 02 02 02 00 00 00 13 73 57 41 20 4C 4D 50 6F 75 74 70 75 74 52 61 6E 67 65 20 74

12.4.1.4.3.2 Read for actual output range [sRN LMPoutputRange]

Table 97: Telegram structure: sRN LMPoutputRange

Telegram structure: sRN LMPoutputRange						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Output range	String	14		LMPoutpu-tRange	4C 4D 50 6F 75 74 70 75 74 52 61 6E 67 65

Table 98: Example: sRN LMPoutputRange

CoLa A	<STX>sRN{SPC}LMPoutputRange<ETX>	
	<STX>sRN LMPoutputRange<ETX>	
	sRN LMPoutputRange	
	02 73 52 4E 20 4C 4D 50 6F 75 74 70 75 74 52 61 6E 67 65 03	
CoLa B	02 02 02 02 00 00 00 12 73 52 4E 20 4C 4D 50 6F 75 74 70 75 74 52 61 6E 67 65 5E	
	73 52 4E 20 4C 4D 50 6F 75 74 70 75 74 52 61 6E 67 65	

Table 99: Telegram structure: sRA LMPoutputRange

Telegram structure: sRA LMPoutputRange						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Output range	String	14		LMPoutpu-tRange	4C 4D 50 6F 75 74 70 75 74 52 61 6E 67 65
Reserved		Int_16	2	Always:	1h	00 01
Angular resolution	[1/10000°]	Uint_32	4	0.125°:	4E2h	00 00 04 E2
Start angle	[1/10000°]	Int_32	4	-180° ... +180°	FFE488C0h ... 1B7740h	FF E4 88 C0 ... 00 1B 77 40
Stop angle	[1/10000°]	Int_32	4	-180° ... +180°	FFE488C0h ... 1B7740h	FF E4 88 C0 ... 00 1B 77 40

Table 100: Example: sRA LMPoutputRange – device output set at 0.125° angular resolution and output range from -90° to +90°

CoLa A	<STX>sRA{SPC}LMPoutputRange{SPC}1{SPC}4E2{SPC}FFF24460{SPC}DBBA0<ETX>
	02 73 52 41 20 4C 4D 50 6F 75 74 70 75 74 52 61 6E 67 65 20 31 20 34 45 32 20 46 46 46 32 34 34 36 30 20 44 42 42 41 30 03
CoLa B	02 02 02 02 00 00 00 21 73 52 41 20 4C 4D 50 6F 75 74 70 75 74 52 61 6E 67 65 20 00 01 00 00 04 E2 FF F2 44 60 00 0D BB A0 A9

12.4.1.4.3.3 Set scan data enable [sWN ScanDataEnable]

Enables/ Disables streaming data output

Table 101: Telegram structure: sWN ScanDataEnable

Telegram structure: sWN ScanDataEnable (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 E4
Command	Enables/ Disables streaming data output.	String	14		ScanDataEna-ble	53 63 61 6E 44 61 74 61 45 6E 61 62 6C 65
Data		Bool	1	Off: On:	0d (00h) +1d (01h)	00 ... 01

Table 102: Example: sWN ScanDataEnable 0 - Disable the streaming data output

CoLa A	<STX>sWN{SPC}ScanDataEnable{SPC}0<ETX>	
	<STX>sWN ScanDataEnable 0<ETX>	
	sWN ScanDataEnable 0	
	02 73 57 E4 20 53 63 61 6E 44 61 74 61 45 6E 61 62 6C 65 20 30 03	
CoLa B	02 02 02 02 00 00 00 14 73 57 E4 20 53 63 61 6E 44 61 74 61 45 6E 61 62 6C 65 20 00 44	
	73 57 E4 20 53 63 61 6E 44 61 74 61 45 6E 61 62 6C 65 20 00	

Table 103: Telegram structure: sWA ScanDataEnable

Telegram structure: sWA ScanDataEnable (Required User Level authorized clinet)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Enables/ Disables streaming data output.	String	14		ScanDataEna-ble	53 63 61 6E 44 61 74 61 45 6E 61 62 6C 65

Table 104: Example: sWA ScanDataEnable

CoLa A	<STX>sWA{SPC}ScanDataEnable<ETX>
	02 73 57 41 20 53 63 61 6E 44 61 74 61 45 6E 61 62 6C 65 03
CoLa B	02 02 02 02 00 00 00 13 73 57 41 20 53 63 61 6E 44 61 74 61 45 6E 61 62 6C 65 20 4B

12.4.1.4.3.4 Set streaming ethernet settings [sWN ScanDataEthSettings]

Ethernet settings for the scan data streaming functionality of the device

Table 105: Telegram structure: sWN ScanDataEthSettings

Telegram structure: sWN ScanDataEthSettings (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E

Telegram structure: sWN ScanDataEthSettings (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command	String	String	19		ScanDataEth-Settings	53 63 61 6E 44 61 74 61 45 74 68 53 65 74 74 69 6E 67 73
Protocol	Transport protocol for streaming data	Enum_8	1	UDP: TCP:	+1d (01h) +2d (02h)	01 02
IPAddress	IP address of the destination for data receiver	Array	4		0 ...+255d (00...FF) 0 ...+255d (00..FF) 0 ...+255d (00...FF) 0 ...+255d (00...FF)	00 ... FF 00 ... FF 00 ... FF 00 ... FF
Port	Port destination of the data reseiver	Uint_16	2		0 .. +65535d(00 00...FF FF)	00 00 ... FF FF

Example: sWN ScanDataEthSettings +1 +192 +168 +0 +100 +2115

Protocol is set to UPD (1), IPAddress (192.168.0.100), Port (2115)

Table 106: Example: sWN ScanDataEthSettings

CoLa A	<STX>sWN{SPC}ScanDataEthSettings{SPC}+1{SPC}+192{SPC}+168{SPC}+0{SPC}+100{SPC}+2115<ETX>	
	<STX>sWN ScanDataEthSettings +1 +192 +168 +0 +100 +2115<ETX>	
	sWN ScanDataEthSettings +1 +192 +168 +0 +100 +2115	
	02 73 57 4E 20 53 63 61 6E 44 61 74 61 45 74 68 53 65 74 74 69 6E 67 73 20 31 20 43 30 20 41 38 20 30 20 36 34 20 38 34 33 03	
CoLa B	02 02 02 02 00 00 00 1F 73 57 4E 20 53 63 61 6E 44 61 74 61 45 74 68 53 65 74 74 69 6E 67 73 20 01 C0 A8 00 64 08 43 5F	
	73 57 4E 20 53 63 61 6E 44 61 74 61 45 74 68 53 65 74 74 69 6E 67 73 20 01 C0 A8 00 64 08 43	

Table 107: Telegram structure: sWA ScanDataEthSettings

Telegram structure: sWA ScanDataEthSettings						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	String	String	19		ScanDataEth-Settings	53 63 61 6E 44 61 74 61 45 74 68 53 65 74 74 69 6E 67 73

Table 108: Example: sWA ScanDataEthSettings

CoLa A	<STX>sWA{SPC}ScanDataEthSettings<ETX>
	02 73 57 41 20 53 63 61 6E 44 61 74 61 45 74 68 53 65 74 74 69 6E 67 73 03
CoLa B	02 02 02 02 00 00 00 18 73 57 41 20 53 63 61 6E 44 61 74 61 45 74 68 53 65 74 74 69 6E 67 73 20 16

12.4.1.4.3.5 Set IMU data enable [sWN ImuDataEnable]

Enables/ Disables streaming IMU data output.

Table 109: Telegram structure: sWN ImuDataEnable

Telegram structure: sWN ImuDataEnable (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 E4
Command	Set streaming IMU data output.	String	13		ImuDataEnable	49 6D 75 44 61 74 61 45 6E 61 62 6C 65
IMU data stream	Enable/ disable	Bool	1	Disbale: Enable:	0d (00h) +1d (01h)	00 ... 01

Table 110: Example: sWN ImuDataEnable 0 - Disable the streaming IMU data output

CoLa A	<STX>sWN{SPC}ImuDataEnable{SPC}0<ETX>	
	<STX>sWN ImuDataEnable 0<ETX>	
	sWN ImuDataEnable 0	
	02 73 57 E4 20 49 6D 75 44 61 74 61 45 6E 61 62 6C 65 20 30 03	
CoLa B	02 02 02 02 00 00 00 13 73 57 E4 20 49 6D 75 44 61 74 61 45 6E 61 62 6C 65 20 00 2A	
	73 57 E4 20 49 6D 75 44 61 74 61 45 6E 61 62 6C 65 20 00	

Table 111: Telegram structure: sWA ImuDataEnable

Telegram structure: sWA ImuDataEnable						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Set streaming IMU data output.	String	13		ImuDataEnable	49 6D 75 44 61 74 61 45 6E 61 62 6C 65

Table 112: Example: sWA ImuDataEnable

CoLa A	<STX>sWA{SPC}ImuDataEnable<ETX>
	02 73 57 41 20 49 6D 75 44 61 74 61 45 6E 61 62 6C 65 03
CoLa B	02 02 02 02 00 00 00 12 73 57 41 20 49 6D 75 44 61 74 61 45 6E 61 62 6C 65 20 25

12.4.1.4.3.6 Set IMU data streaming ethernet settings [sWN ImuDataEthSettings]

Ethernet settings for the IMU data streaming functionality of the device

Table 113: Telegram structure: sWN ImuDataEthSettings

Telegram structure: sWN ImuDataEthSettings (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E

Telegram structure: sWN ImuDataEthSettings (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command	Set IMU data streaming ethernet settings	String	18		ImuDataEthSet-tings	49 6D 75 44 61 74 61 45 74 68 53 65 74 74 69 6E 67 73
Protocol	Transport protocol for streaming IMU data	Enum_8	1	UDP:	+1d (01h)	01
IP address	IP address of the destination for IMU data receiver	Array	4		0 ...+255d (00...FF) 0 ...+255d (00..FF) 0 ...+255d (00...FF) 0 ...+255d (00...FF)	00 ... FF 00 ... FF 00 ... FF 00 ... FF
Port	Port destination of the IMU data reseiver	Uint_16	2		0 .. +65535d(00 00...FF FF)	00 00 ... FF FF

Example: sWN ImuDataEthSettings +1 +192 +168 +0 +100 +7503

Protocol is set to UPD (1), IPAddress (192.168.0.100), Port (7503)

Table 114: Example: sWN ImuDataEthSettings

CoLa A	<STX>sWN{SPC}ImuDataEthSettings{SPC}+1{SPC}+192{SPC}+168{SPC}+0{SPC}+100{SPC}+7503<ETX>	
	<STX>sWN ImuDataEthSettings +1 +192 +168 +0 +100 +7503<ETX>	
	sWN ImuDataEthSettings +1 +192 +168 +0 +100 +7503	
	02 73 57 4E 20 49 6D 75 44 61 74 61 45 74 68 53 65 74 74 69 6E 67 73 20 31 20 43 30 20 41 38 20 30 20 36 34 20 31 44 34 46 03	
CoLa B	02 02 02 02 00 00 00 1E 73 57 4E 20 49 6D 75 44 61 74 61 45 74 68 53 65 74 74 69 6E 67 73 20 01 C0 A8 00 64 1D 4F 28	
	73 57 4E 20 49 6D 75 44 61 74 61 45 74 68 53 65 74 74 69 6E 67 73 20 01 C0 A8 00 64 1D 4F	

Table 115: Telegram structure: sWA ImuDataEthSettings

Telegram structure: sWA ImuDataEthSettings						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Set IMU data streaming ethernet settings	String	18		ImuDataEthSet-tings	49 6D 75 44 61 74 61 45 74 68 53 65 74 74 69 6E 67 73

Table 116: Example: sWA ImuDataEthSettings

CoLa A	<STX>sWA{SPC}ImuDataEthSettings<ETX>
	02 73 57 41 20 49 6D 75 44 61 74 61 45 74 68 53 65 74 74 69 6E 67 73 03
CoLa B	02 02 02 02 00 00 00 17 73 57 41 20 49 6D 75 44 61 74 61 45 74 68 53 65 74 74 69 6E 67 73 20 78

12.4.1.4.3.7 Read scan data format [sRN ScanDataFormat]

Return of the scan data format

Table 117: Telegram structure: sRN ScanDataFormat

Telegram structure: sRN ScanDataFormat (User Level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Data serialization format	String	14		ScanDataFormat	53 63 61 6E 44 61 74 61 46 6F 72 6D 61 74

Example: sRN ScanDataFormat

Read of the data serialization format

Table 118: Example: sRN ScanDataFormat

CoLa A	<STX>sRN{SPC}ScanDataFormat<ETX>	
	<STX>sRN ScanDataFormat<ETX>	
	sRN ScanDataFormat	
	02 73 52 4E 20 53 63 61 6E 44 61 74 61 46 6F 72 6D 61 74 03	
CoLa B	02 02 02 02 00 00 00 12 73 52 4E 20 53 63 61 6E 44 61 74 61 46 6F 72 6D 61 74 63	
	73 52 4E 20 53 63 61 6E 44 61 74 61 46 6F 72 6D 61 74	

Table 119: Telegram structure: sRA ScanDataFormat

Telegram structure: sRA ScanDataFormat						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Data serialization format	String	14		ScanDataFormat	53 63 61 6E 44 61 74 61 46 6F 72 6D 61 74
Data		Enum_8	1	MSGPACK Compact	1h 2h	01 02

Example: sRA ScanDataFormat

Scan data format is set to Compact = 2

Table 120: Example: sRA ScanDataFormat

CoLa A	<STX>sRA{SPC}ScanDataFormat{SPC}2<ETX>
	02 73 57 41 20 53 63 61 6E 44 61 74 61 46 6F 72 6D 61 74 20 32 03
CoLa B	02 02 02 02 00 00 00 14 73 57 41 20 53 63 61 6E 44 61 74 61 46 6F 72 6D 61 74 20 02 4E

12.4.1.4.3.8 Set Scan data format [sWN ScanDataFormat]

Set the data serialization format

Table 121: Telegram structure: sWN ScanDataFormat

Telegram structure: sWN ScanDataFormat (User Level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Data serialization format	String	14		ScanDataFormat	53 63 61 6E 44 61 74 61 46 6F 72 6D 61 74
Variable Data	Data	Enum_8	1	MSGPACK: Compact:	+1d (1h) +2d (2h)	01 02

Example: sWN ScanDataFormat

Scan data format set to **Compact** format

Table 122: Example: sWN ScanDataFormat

CoLa A	<STX>sWN{SPC}ScanDataFormat{SPC}2<ETX>
	<STX>sWN ScanDataFormat 2<ETX>
	sWN ScanDataFormat 2
	02 73 57 4E 20 53 63 61 6E 44 61 74 61 46 6F 72 6D 61 74 20 32 03
CoLa B	02 02 02 02 00 00 14 73 57 4E 20 53 63 61 6E 44 61 74 61 46 6F 72 6D 61 74 20 02 44
	73 57 4E 20 53 63 61 6E 44 61 74 61 46 6F 72 6D 61 74 20 02

Table 123: Telegram structure: sWA ScanDataFormat

Telegram structure: sWA ScanDataFormat						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Data serialization format	String	14		ScanDataFormat	53 63 61 6E 44 61 74 61 46 6F 72 6D 61 74

Table 124: Example: sWA ScanDataFormat

CoLa A	<STX>sWA{SPC}ScanDataFormat<ETX>
	02 73 57 41 20 53 63 61 6E 44 61 74 61 46 6F 72 6D 61 74 03
CoLa B	02 02 02 02 00 00 00 13 73 57 41 20 53 63 61 6E 44 61 74 61 46 6F 72 6D 61 74 20 49

12.4.1.4.4 Time stamp

12.4.1.4.4.1 Set time stamp [sMN LSPsetdatetime]

NOTE

There is no real time clock inside the device. When the scanner is switched off and after a reboot, the time has to be set again.

- However, it is possible to analyze the Off-time in order to evade this issue.

The data format in the telegram is: +2009{SPC}+7{SPC}+22{SPC}+12{SPC}+0{SPC}+0{SPC}+0.
The numbers represent year, month, day, hour, minute, second, microsecond.

If plus is used up-front the data it is interpreted as an integer decimal number, without the plus it's the scanner reads the data as hex format.

The answer is always in ASCII format.

Table 125: Telegram structure: sMN LSPsetdatetime

Telegram structure: sMN LSPsetdatetime (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E
Command	Set time stamp	String	14		LSPsetdatetime	4C 53 50 73 65 74 64 61 74 65 74 69 6D 65
Year		Uint_16	2		+1970d ... +2099d (07B2h ... 0833h)	07 B2 ... 08 33
Month		Uint_8	1		+1d ... +12d (01h ... 0Ch)	01 ... 0C
Day		Uint_8	1		+1d ... +31d (01h ... 1Fh)	00 ... 1F
Hour		Uint_8	1		+0d ... +23d (00h ... 17h)	00 ... 17
Minute		Uint_8	1		+0d ... +59d (00h ... 3Bh)	00 ... 3B
Second		Uint_8	1		+0d ... +59d (00h ... 3Bh)	00 ... 3B
Microsec-ond		Uint_32	4		+0d ... +999999d (00000000h ... 000F423Fh)	00 00 00 00 ... 00 0F 42 3F

Table 126: Example 1: sMN LSPsetdatetime

CoLa A	<STX>sMN{SPC}LSPsetdatetime{SPC}7D9{SPC}2{SPC}11{SPC}10{SPC}22{SPC}0{SPC}0<ETX>	
	<STX>sMN LSPsetdatetime 7D9 2 11 10 22 0 0<ETX>	
	sMN LSPsetdatetime 7D9 2 11 10 22 0 0	
	02 73 4D 4E 20 4C 53 50 73 65 74 64 61 74 65 74 69 6D 65 20 37 44 39 20 32 20 31 31 20 31 30 20 32 32 20 30 20 30 03	
CoLa B	02 02 02 02 00 00 00 1E 73 4D 4E 20 4C 53 50 73 65 74 64 61 74 65 74 69 6D 65 20 07 D9 02 11 10 22 00 00 00 00 00 A3	
	73 4D 4E 20 4C 53 50 73 65 74 64 61 74 65 74 69 6D 65 20 07 D9 02 11 10 22 00 00 00 00 00	

Table 127: Example 2: sMN LSPsetdatetime

CoLa A	<STX>sMN{SPC}LSPsetdatetime{SPC}+2010{SPC}+01{SPC}+26{SPC}+10{SPC}+35{SPC}0{SPC}0<ETX>	
	<STX>sMN LSPsetdatetime +2010 +01 +26 +10 +35 0 0<ETX>	
	sMN LSPsetdatetime +2010 +01 +26 +10 +35 0 0	
	02 73 4D 4E 20 4C 53 50 73 65 74 64 61 74 65 74 69 6D 65 20 2B 32 30 31 30 20 2B 30 31 20 2B 32 36 20 2B 31 30 20 2B 33 35 20 2B 30 30 20 2B 30 30 30 30 03	

CoLa B	02 02 02 02 00 00 00 1E 73 4D 4E 20 4C 53 50 73 65 74 64 61 74 65 74 69 6D 65 20 07 DA 01 1A 0A 23 00 00 00 00 00 A3
	73 4D 4E 20 4C 53 50 73 65 74 64 61 74 65 74 69 6D 65 20 07 DA 01 1A 0A 23 00 00 00 00 00

Table 128: Telegram structure: sAN LSPsetdatetime

Telegram structure: sAN LSPsetdatetime						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 41 4E
Command	Set time stamp	String	14		LSPsetdatetime	4C 53 50 73 65 74 64 61 74 65 74 69 6D 65
Status code	Code number	Enum_8	1	Success:	0	00

Table 129: Example 1, 2: sAN LSPsetdatetime

CoLa A	<STX>sAN{SPC}LSPsetdatetime{SPC}0<ETX>
	02 73 41 4E 20 4C 53 50 73 65 74 64 61 74 65 74 69 6D 65 20 30 03
CoLa B	02 02 02 02 00 00 00 14 73 41 4E 20 4C 53 50 73 65 74 64 61 74 65 74 69 6D 65 20 00 50

Activate time stamp in the output string format or on SOPAS page “data processing”.

12.4.1.4.4.2 Read device time [sRN DeviceTime]

Command to read the actual time of the internal clock (ms).

The timer is 32 counter with a resolution of 1 ms.

Table 130: Telegram structure: sRN DeviceTime

Telegram structure: sRN DeviceTime						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command		String	10		DeviceTime	44 65 76 69 63 65 54 69 6D 65

Table 131: Example: sRN DeviceTime

CoLa A	<STX>sRN{SPC}DeviceTime<ETX>
	<STX>sRN DeviceTime<ETX>
	sRN DeviceTime
	02 73 52 4E 20 44 65 76 69 63 65 54 69 6D 65 03
CoLa B	02 02 02 02 00 00 00 0E 73 52 4E 20 44 65 76 69 63 65 54 69 6D 65 42
	73 52 4E 20 44 65 76 69 63 65 54 69 6D 65

Table 132: Telegram structure: sRA DeviceTime

Telegram structure: sRA DeviceTime						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command		String	10		DeviceTime	44 65 76 69 63 65 54 69 6D 65
Device time	Time	Uint_32	4		0d ... +9999d (0h ... 270Fh)	00 00 00 00 ... 00 00 27 0F

Table 133: Example: sRA DeviceTime 0

CoLa A	<STX>sRA{SPC}DeviceTime{SPC}0<ETX>
	02 73 52 41 20 44 65 76 69 63 65 54 69 6D 65 20 00 03
CoLa B	02 02 02 02 00 00 00 13 73 52 41 20 44 65 76 69 63 65 54 69 6D 65 20 00 00 00 00 6D

12.4.1.4.4.3 Set NTP (Network Time Protocol) parameters

12.4.1.4.4.3.1 Set time synchronization [sWN TSCRole]

Table 134: Telegram structure: sWN TSCRole

Telegram structure: sWN TSCRole (User Level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Set NTP role	String	7		TSCRole	54 53 43 52 6F 6C 65
Variable data	NTP role	Uint_8	1	None: Client: Server:	0 1 2	00 01 02

Table 135: Example: sWN TSCRole

CoLa A	<STX>sWN{SPC}TSCRole{SPC}1<ETX>	
	<STX>sWN TSCRole 1<ETX>	
	sWN TSCRole 1	
	02 73 57 4E 20 54 53 43 52 6F 6C 65 20 31 03	
CoLa B	02 02 02 02 00 00 00 0D 73 57 4E 20 54 53 43 52 6F 6C 65 20 01 1B	
	73 57 4E 20 54 53 43 52 6F 6C 65 20 01	

Table 136: Telegram structure: sWA TSCRole

Telegram structure: sWA TSCRole						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41

Telegram structure: sWA TSCRole						
Telegram part	Description	Variable	Length	Additional details	Values CoLa A (ASCII)	Values CoLa B (Binary)
Command	Set NTP role	String	7		TSCRole	54 53 43 52 6F 6C 65

Table 137: Example: sWA TSCRole

CoLa A	<STX>sWA{SPC}TSCRole<ETX>
	02 73 57 41 20 54 53 43 52 6F 6C 65 03
CoLa B	02 02 02 02 00 00 00 0C 73 57 41 20 54 53 43 52 6F 6C 65 20 15

12.4.1.4.4.3.2 Set time server IP address [sWN TSCTCSrvAddr]

Table 138: Telegram structure: sWN TSCTCSrvAddr

Telegram structure: sWN TSCTCSrvAddr (User level 'Authorized client' required)						
Telegram part	Description	Variable	Length	Additional details	Values CoLa A (ASCII)	Values CoLa B (Binary)
Command type	Write	String	3		sWN	73 57 4E
Command	Set time server IP address	String	12		TSCTCSrvAddr	54 53 43 54 43 53 72 76 41 64 64 72
IP address data	Set values	Uint_8	1	First part of IP address	0 ...+255d (00...FF)	00 ... FF
				Second part of IP address	0 ...+255d (00...FF)	00 ... FF
				Third part of IP address	0 ...+255d (00...FF)	00 ... FF
				Fourth part of IP address	0 ...+255d (00...FF)	00 ... FF

Table 139: Example: sWN TSCTCSrvAddr 192.168.0.11

CoLa A	<STX>sWN{SPC}TSCTCSrvAddr{SPC}C0{SPC}A8{SPC}00{SPC}0B<ETX>
	<STX>sWN TSCTCSrvAddr C0 A8 00 0B<ETX>
	sWN TSCTCSrvAddr C0 A8 00 0B
	02 73 57 4E 20 54 53 43 54 43 53 72 76 41 64 64 72 20 43 30 20 41 38 20 30 30 20 30 42 03
CoLa B	02 02 02 02 00 00 00 15 73 57 4E 20 54 53 43 54 43 53 72 76 41 64 64 72 20 C0 A8 00 0B 3E
	73 57 4E 20 54 53 43 54 43 53 72 76 41 64 64 72 20 C0 A8 00 0B

Table 140: Telegram structure: sWA TSCTCSrvAddr

Telegram structure: sWA TSCTCSrvAddr						
Telegram part	Description	Variable	Length	Additional details	Values CoLa A (ASCII)	Values CoLa B (Binary)
Command type	Answer	String	3		sWA	73 57 41
Command	Set time server IP address	String	12		TSCTCSrvAddr	54 53 43 54 43 53 72 76 41 64 64 72

Table 141: Example: sWA TSCTCSrvAddr

CoLa A	<STX>sWA{SPC}TSCTCSrvAddr<ETX>
	02 73 57 41 20 54 53 43 54 43 53 72 76 41 64 64 72 03
CoLa B	02 02 02 02 00 00 00 11 73 57 41 20 54 53 43 54 43 53 72 76 41 64 64 72 20 52

12.4.1.4.4.3.3 Set time zone [sWN TSCTCtimezone]

Table 142: Telegram structure: sWN TSCTCtimezone

Telegram structure: sWN TSCTCtimezone (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Set time zone	String	13		TSCTCtimezone	54 53 43 54 43 74 69 6D 65 7A 6F 6E 65
Time zone data	Select the time zone of the client	Enum_8	1	List of time zones see table 143, page 105	+0d ... +104d (00h ... 68h)	00 ... 68

Table 143: Time zone data Values CoLa (ASCII)

0	DATE_LINE_STANDARD	35	MONROVIA_REYKJAVIK	70	MUMBAI_NEUDELHI
1	COORD_WORLD_TIME_11	36	AMSTERDAM_BERLIN_ROM	71	SRI_JAYAWARDENEPURA
2	HAWAII	37	BELGRAD_BUDAPEST_PRAG	72	KATMANDU
3	ALASKA	38	BRUESSEL_MADRID_PARIS	73	ASTANA
4	CALIFORNIA	39	SARAJEVO_WARSCHAU	74	DAKKA
5	USA_CANADA	40	WEST_CENTRAL_AFRICA	75	NOWOSIBIRSK
6	ARIZONA	41	WINDHUK	76	YANGON
7	LA_PAZ	42	AMMAN	77	BANGKOK_HANOI_JAKARTA
8	MOUNTAIN_TIME_USA	43	ATHEN_BUKAREST	78	KRASNOJARSK
9	CENTRAL_TIME_USA	44	BEIRUT	79	IRKUTSK
10	MEXICO_CITY	45	DAMASCUS	80	KUALA_LUMPUR_SINGAPUR
11	MIDDLE_AMERICA	46	HARARE_PRETORIA	81	PEKING_HONGKONG
12	SASKATCHEWAN	47	HELSINKI_KIEW_RIGA	82	PERTH
13	BOGOTA_LIMA	48	ISTANBUL	83	TAIPEH
14	EASTERN_TIME_USA	49	JERUSALEM	84	ULAN_BATOR
15	INDIANA	50	KAIRO	85	JAKUTSK
16	CARACAS	51	KALININGRAD	86	OSAKA_TOKIO
17	ASUNCION	52	EASTERN_EUROPE	87	SEOUL
18	ATLANTIC_KANADA	53	TRIPOLIS	88	ADELAIDE
19	CUIABA	54	BAGDAD	89	DARWIN
20	LAPAZ_SANJUAN	55	KUWAIT_RIAD	90	BRISBANE
21	SANTIAGO	56	MINSK	91	CANBERRA_SYDNEY
22	NEUFUNDLAND	57	MOSKAU_PETERSBURG	92	GUAM_PORT_MORESBY
23	BRASILIA	58	NAIROBI	93	HOBART
24	BUENOS_AIRES	59	TEHERAN	94	MAGADAN
25	CAYENNE_FORTALEZA	60	ABU_DHABI	95	WLADIWOSTOK
26	GROENLAND	61	BAKU	96	SALOMONEN_KALEDONIEN

27	MONTEVIDEO	62	ERIWAN	97	TSCHOKURDACH
28	SALVADOR	63	ISCHEWSK_SAMARA	98	ANADYR
29	COORD_WORLD_TIME_02	64	PORT_LOUIS	99	AUCKLAND_WELLINGTON
30	AZOREN	65	TIFLIS	100	FIDSCHI
31	KAP_VERDE	66	KABUL	101	COORD_WORLD_TIME_12
32	CASABLANCA	67	ASCHGABET_TASCHKENT	102	NAKUALOFA
33	DUBLIN_LISSABON_LONDON	68	ISLAMABAD_KARATSCHI	103	SAMOA
34	COORD_WORLD_TIME	69	JEKATERINBURG	104	KIRITIMATI

Table 144: Example: sWN TSCTCtimezone Amsterdam, Berlin, Rom

CoLa A	<STX>sWN{SPC}TSCTCtimezone{SPC}+36<ETX>
	<STX>sWN TSCTCtimezone +36<ETX>
	sWN TSCTCtimezone +36
	02 73 57 4E 20 54 53 43 54 43 74 69 6D 65 7A 6F 6E 65 20 24 03
CoLa B	02 02 02 02 00 00 00 13 73 57 4E 20 54 53 43 54 43 74 69 6D 65 7A 6F 6E 65 20 24 16
	73 57 4E 20 54 53 43 54 43 74 69 6D 65 7A 6F 6E 65 20 24

Table 145: Telegram structure: sWA TSCTCtimezone

Telegram structure: sWA TSCTCtimezone						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Set time zone	String	13		TSCTCtimezone	54 53 43 54 43 74 69 6D 65 7A 6F 6E 65

Table 146: Example: sWA TSCTCtimezone

CoLa A	<STX>sWA{SPC}TSCTCtimezone<ETX>
	02 73 57 41 20 54 53 43 54 43 74 69 6D 65 7A 6F 6E 65 03
CoLa B	02 02 02 02 00 00 00 12 73 57 41 20 54 53 43 54 43 74 69 6D 65 7A 6F 6E 65 20 3D

12.4.1.4.4.3.4 Set update time [sWN TSCTCupdatetime]

Table 147: Telegram structure: sWN TSCTCupdatetime

Telegram structure: sWN TSCTCupdatetime (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Set update time of synchronization	String	15		TSCTCupdate-time	54 53 43 54 43 75 70 64 61 74 65 74 69 6D 65
Update time of synchronization	Set values in seconds	Uint_32	4		+1d ... +3600d (01h ... 0E10h)	00 00 00 00 ... 00 00 0E 10

Table 148: Example: sWN TSCTCupdatetime 600 s

CoLa A	<STX>sWN{SPC}TSCTCupdatetime{SPC}+600<ETX>	
	<STX>sWN TSCTCupdatetime +600<ETX>	
	sWN TSCTCupdatetime +600	
	02 73 57 4E 20 54 53 43 54 43 75 70 64 61 74 65 74 69 6D 65 20 2B 36 30 30 03	
CoLa B	02 02 02 02 00 00 00 18 73 57 4E 20 54 53 43 54 43 75 70 64 61 74 65 74 69 6D 65 20 00 00 02 58 67	
	73 57 4E 20 54 53 43 54 43 75 70 64 61 74 65 74 69 6D 65 20 00 00 02 58	

Table 149: Telegram structure: sWA TSCTCupdatetime

Telegram structure: sWA TSCTCupdatetime						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Set update time of synchronization	String	15		TSCTCupdate-time	54 53 43 54 43 75 70 64 61 74 65 74 69 6D 65

Table 150: Example: sWA TSCTCupdatetime

CoLa A	<STX>sWA{SPC}TSCTCupdatetime<ETX>
	02 73 57 41 20 54 53 43 54 43 75 70 64 61 74 65 74 69 6D 65 03
CoLa B	02 02 02 02 00 00 00 14 73 57 41 20 54 53 43 54 43 75 70 64 61 74 65 74 69 6D 65 20 32

12.4.1.4.5 Filter

12.4.1.4.5.1 Set particle filter [sWN LFPparticle]

Table 151: Telegram structure: sWN LFPparticle

Telegram structure: sWN LFPparticle (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Set particle filter	String	11		LFPparticle	4C 46 50 70 61 72 74 69 63 6C 65
Status code	Code number	Bool_1	1	Inactive: Active:	0 1	00 01
Thresh-old[1)]	Particle threshold in mm	Uint_16	2	(must be taken)	+500d (1F4h)	01 F4

1) Never change the threshold here, it is taken by the dvice to handle the particles.

1) 1)

Table 152: Example: sWN LFPparticle

CoLa A	<STX>sWN{SPC}LFPparticle{SPC}1{SPC}+500<ETX>
	<STX>sWN LFPparticle 1 +500<ETX>
	sWN LFPparticle 1 +500
	02 73 57 4E 20 4C 46 50 70 61 72 74 69 63 6C 65 20 31 20 2B 35 30 30 03
CoLa B	02 02 02 02 00 00 00 13 73 57 4E 20 4C 46 50 70 61 72 74 69 63 6C 65 20 01 01 F4 D0
	73 57 4E 20 4C 46 50 70 61 72 74 69 63 6C 65 20 01 01 F4

Table 153: Telegram structure: sWA LFPparticle

Telegram structure: sWA LFPparticle						
Telegram part	**Description**	**Variable**	**Length**	**Sensor**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Set particle filter	String	11		LFPparticle	4C 46 50 70 61 72 74 69 63 6C 65

Table 154: Example: sWA LFPparticle

CoLa A	<STX>sWA{SPC}LFPparticle<ETX>
	02 73 57 41 20 4C 46 50 70 61 72 74 69 63 6C 65 03
CoLa B	02 02 02 02 00 00 00 10 73 57 41 20 4C 46 50 70 61 72 74 69 63 6C 65 20 2B

12.4.1.4.5.2 Set echo filter [sWN FREchoFilter]

NOTE
Only available with firmware versions > V1.10.

Table 155: Telegram structure: sWN FREchoFilter

Telegram structure: sWN FREchoFilter (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Set echo filter	String	12		FREchoFilter	46 52 45 63 68 6F 46 69 6C 74 65 72
Status code	Code number	Enum_8	1	First echo: All echos: Last echo:	0 1 2	00 01 02

Table 156: Example: sWN FREchoFilter

CoLa A	<STX>sWN{SPC}FREchoFilter{SPC}1<ETX>
	<STX>sWN FREchoFilter 1<ETX>
	sWN FREchoFilter 1
	02 73 57 4E 20 46 52 45 63 68 6F 46 69 6C 74 65 72 20 31 03

CoLa B	02 02 02 02 00 00 00 12 73 57 4E 20 46 52 45 63 68 6F 46 69 6C 74 65 72 20 01 7E
	73 57 4E 20 46 52 45 63 68 6F 46 69 6C 74 65 72 20 01

Table 157: Telegram structure: sWA FREchoFilter

Telegram structure: sWA FREchoFilter						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Set echo filter	String	12		FREchoFilter	46 52 45 63 68 6F 46 69 6C 74 65 72

Table 158: Example: sWa FREchoFilter

CoLa A	<STX>sWA{SPC}FREchoFilter<ETX>
	02 73 57 41 20 46 52 45 63 68 6F 46 69 6C 74 65 72 03
CoLa B	02 02 02 02 00 00 00 11 73 57 41 20 46 52 45 63 68 6F 46 69 6C 74 65 72 20 70

12.4.1.4.5.3 Set sensitivity fog filter [sWN MCSenseLevel]

Table 159: Telegram structure: sWN MCSenseLevel

Telegram structure: sWN MCSenseLevel (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Sense level	String	12		MCSenseLevel	4D 43 53 65 6E 73 65 4C 65 76 65 6C
Sensitivity level	Enable or disable fog filter and Sense Level	Uint_8	1	Fog Filter off Fog Filter on	0 1	00 01

Table 160: Example: sWN MCSenseLevel

CoLa A	<STX>sWN{SPC}MCSenseLevel{SPC}1<ETX>
	<STX>sWN MCSenseLevel 1<ETX>
	sWN MCSenseLevel 1
	02 73 57 4E 20 4D 43 53 65 6E 73 65 4C 65 76 65 6C 20 31 03
CoLa B	02 02 02 02 00 00 00 10 73 57 4E 20 4D 43 53 65 6E 73 65 4C 65 76 65 6C 20 01 70
	73 57 4E 20 4D 43 53 65 6E 73 65 4C 65 76 65 6C 20 01

Table 161: Telegram structure: sWA MCSenseLevel

Telegram structure: sWA MCSenseLevel						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41

Telegram structure: sWA MCSenseLevel						
Telegram part	Description	Variable	Length	Additional details	Values CoLa A (ASCII)	Values CoLa B (Binary)
Command	Sense level	String	12		MCSenseLevel	4D 43 53 65 6E 73 65 4C 65 76 65 6C

Table 162: Example: sWA MCSenseLevel

CoLa A	<STX>sWA{SPC}MCSenseLevel<ETX>
	02 73 57 41 20 4D 43 53 65 6E 73 65 4C 65 76 65 6C 20 03
CoLa B	02 02 02 02 00 00 00 0F 73 57 41 20 4D 43 53 65 6E 73 65 4C 65 76 65 6C 20 73

12.4.1.4.5.4 Set cubic area filter [sWN LFPcubicareafilter]

The cubic area filter limits a polar scan to a axisparallel cube defined by its extension in x-, y- and z-range.

Table 163: Telegram structure: sWN LFPcubicareafilter

Telegram structure: sWN LFPcubicareafilter (User level 'Authorized client' required)						
Telegram part	Description	Variable	Length	Additional details	Values CoLa A (ASCII)	Values CoLa B (Binary)
Command type	Write	String	3		sWN	73 57 4E
Command	CubicAreaFilter limits a polar scan to a axisparallel cube	String	18		LFPcubicareafilter	4C 46 50 63 75 62 69 63 61 72 65 61 66 69 6C 74 65 72
Variable data 1	Enables/Disables the filter.	Bool	1	Off: On:	0d 1d	00 01
Variable Data 2	X min 1/10 mm	Int_32	4		-200000d... +200000d	FF FF B1 E0 ... 00 00 4E 20
Variable Data 3	X max 1/10 mm	Int_32	4		-200000d... +200000d	FF FF B1 E0 ... 00 00 4E 20
Variable Data 4	Y min 1/10 mm	Int_32	4		-200000d... +200000d	FF FF B1 E0 ... 00 00 4E 20
Variable Data 5	Y max 1/10 mm	Int_32	4		-200000d... +200000d	FF FF B1 E0 ... 00 00 4E 20
Variable Data 6	Z min 1/10 mm	Int_32	4		-200000d... +200000d	FF FF B1 E0 ... 00 00 4E 20
Variable Data 7	Z max 1/10 mm	Int_32	4		-200000d... +200000d	FF FF B1 E0 ... 00 00 4E 20

Disables the cubic area filter and set up to the -20000mm...+20000mm in x,y,z direction.

Table 164: Example: sWN LFPcubicAreafilter

CoLa A	<STX>sWN{SPC}LFPcubicfreafilter{SPC}0{SPC}FFFFB1E0{SPC}4E20{SPC}FFFFB1E0{SPC}4E20{SPC}FFFFB1E0{SPC}4E20<ETX>
	<STX>sWN LFPcubicareafilter 0 FFFFB1E0 4E20 FFFFB1E0 4E20 FFFFB1E0 4E20<ETX>
	sWN LFPcubicareafilter 0 FFFFB1E0 4E20 FFFFB1E0 4E20 FFFFB1E0 4E20
	02 73 57 4E 20 4C 46 50 63 75 62 69 63 61 72 65 61 66 69 6C 74 65 72 20 30 20 46 46 46 46 42 31 45 30 20 34 45 32 30 20 46 46 46 46 42 31 45 30 20 34 45 32 30 20 46 46 46 46 42 31 45 30 20 34 45 32 30 03

CoLa B	02 02 02 02 00 00 00 30 73 57 4E 20 4C 46 50 63 75 62 69 63 61 72 65 61 66 69 6C 74 65 72 20 00 FF FF B1 E0 00 00 4E 20 FF FF B1 E0 00 00 4E 20 FF FF B1 E0 00 00 4E 20 66
	73 57 4E 20 4C 46 50 63 75 62 69 63 61 72 65 61 66 69 6C 74 65 72 20 00 FF FF B1 E0 00 00 4E 20 FF FF B1 E0 00 00 4E 20 FF FF B1 E0 00 00 4E 20

Table 165: Telegram structure: sWA LFPcubicareafilter

Telegram structure: sWALFPcubicareafilter						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	CubicAreaFilter limits a polar scan to a axisparallel cube	String	18		LFPcubicareafilter	4C 46 50 63 75 62 69 63 61 72 65 61 66 69 6C 74 65 72

Table 166: Example: sWA LFPcubicareafilter

CoLa A	<STX>sWA{SPC}LFPcubicareafilter<ETX>
	02 73 57 41 20 4C 46 50 63 75 62 69 63 61 72 65 61 66 69 6C 74 65 72 03
CoLa B	02 02 02 02 00 00 00 17 73 57 41 20 4C 46 50 63 75 62 69 63 61 72 65 61 66 69 6C 74 65 72 20 56

12.4.1.4.5.5 Set angle range filter [sWN LFPangleRangeFilter]

The angle range filter set up the horizontal (theta) and vertical (phi) start- and stop angle in rad.

With multiScan only the horizontal (theta) angle is adjustable. To adjust the vertical limits use the layer filter (LFPlayerFilter)

BeamIncrement = the 'beamIncrement' which is used to subsample the beams within the selected angle range. With a 'beamIncrement' of n only every nth beam from the selected angle range is copied to the output scan, i.e. the angle resolution is reduced by factor n. If the beamIncrement is zero it is set to one.

Table 167: Telegram structure: sWN LFPangleRangeFilter

Telegram structure: sWN LFPangleRangeFilter (User Level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	filter set up the horizontal (theta) and vertical (phi) start- and stop angle in rad	String	19		LFPangleRangeFilter	4C 46 50 61 6E 67 6C 65 52 61 6E 67 65 46 69 6C 74 65 72
Variable Data 1	Enables/Disables the filter	Bool_1	1	Off: On:	+0d +1d	00 01
Variable Data 2	ThetaStart	Real	4	No impact on sensor setting but needs to be filled (see example)	-1,800,000d ... +1,800,000d (FFE488C0 ... 001B7740h)	FF E4 88 C0 ... 00 1B 77 40
Variable Data 3	ThetaStop	Real	4	No impact on sensor setting but needs to be filled (see example)	-1,800,000d ... +1,800,000d (FFE488C0 ... 001B7740h)	FF E4 88 C0 ... 00 1B 77 40

Telegram structure: sWN LFPangleRangeFilter (User Level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Variable Data 4	PhiStart	Real	4	No impact on sensor setting but needs to be filled (see example)	-900,000d ... +900,000d (FFF24460 ... 000DBBA0h)	FF F2 44 60 ... 00 0D BB A0
Variable Data 5	PhiStop	Real	4	No impact on sensor setting but needs to be filled (see example)	-900,000d ... +900,000d (FFF24460 ... 000DBBA0h)	FF F2 44 60 ... 00 0D BB A0
Variable Data 6	BeamIncrement	UInt_16	2	No impact on sensor setting but needs to be filled (see example)	1d...+20d	00 01 ... 00 14

Explanation: Enable the angle range filter and set up theta (horizontal) start -90°, theta stop +90°, phi (vertical) start -90°, phi stop +90°, beam increment 1

Table 168: Example: sWN LFPanlgeRangeFilter

CoLa A	<STX>sWN{SPC}LFPangleRangeFilter{SPC}1{SPC}-900000{SPC}+900000{SPC}-900000{SPC}+900000{SPC}1<ETX>	
	<STX>sWN LFPangleRangeFilter 1 -900000 +900000 -900000 +900000 1<ETX>	
	sWN LFPangleRangeFilter 1 -900000 +900000 -900000 +900000 1	
	02 73 57 4E 4C 46 50 61 6E 67 6C 65 52 61 6E 67 65 46 69 6C 74 65 72 20 31 20 2D 39 30 30 30 30 30 20 2B 39 30 30 30 30 30 20 2D 39 30 30 30 30 30 20 2B 39 30 30 30 30 30 20 31 03	
CoLa B	02 02 02 02 00 00 00 2B 73 57 4E 20 4C 46 50 61 6E 67 6C 65 52 61 6E 67 65 46 69 6C 74 65 72 20 01 FF F2 44 60 00 0D BB A0 FF F2 44 60 00 0D BB A0 00 01 2E	
	73 57 4E 20 4C 46 50 61 6E 67 6C 65 52 61 6E 67 65 46 69 6C 74 65 72 20 01 FF F2 44 60 00 0D BB A0 FF F2 44 60 00 0D BB A0 00 01	

Table 169: Telegram structure: sWA LFP AngleRangeFilter

Telegram structure: sWA LFPangleRangeFilter						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	filter set up the horizontal (theta) and vertical (phi) start- and stop angle in rad	String	19		LFPangleRange-Filter	4C 46 50 61 6E 67 6C 65 52 61 6E 67 65 46 69 6C 74 65 72

Table 170: Example: sWA LFPangleRangeFilter

CoLa A	<STX>sWA{SPC}LFPangleRangeFilter<ETX>
	02 73 57 41 20 4C 46 50 61 6E 67 6C 65 52 61 6E 67 65 46 69 6C 74 65 72 03
CoLa B	02 02 02 02 00 00 00 18 73 57 41 20 4C 46 50 61 6E 67 6C 65 52 61 6E 67 65 46 69 6C 74 65 72 20 21

12.4.1.4.5.6 Set interval filter [sWN LFPintervalFilter]

Enables and set up the interval filter. The interval filter reduce the scan output rate by a given factor.

Table 171: Telegram structure: sWN LFPintervalFilter

Telegram structure: sWN LFPintervalFilter (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Reduce the scan output rate by a given factor	String	17		LFPintervalFilter	4C 46 50 69 6E 74 65 72 76 61 6C 46 69 6C 74 65 72
Variable data 1	Enables/Disables the filter.	Bool	1	Off: On:	0d (00h) +1d (01h)	00 01
Variable Data 2	Only every nth scan is output where n is given by the value of uiReductionFactor.	Uint_32	4		1d...+50d (00 00 00 01h ... 00 00 00 32h)	00 00 00 01 ... 00 00 00 32

Enables the interval filter an set up to the 3rd scan

Table 172: Example: sWN LFPintervalFilter

CoLa A	<STX>sWN{SPC}LFPintervalFilter{SPC}1{SPC}3<ETX>
	<STX>sWN LFPintervalFilter 1 3<ETX>
	sWN LFPintervalFilter 1 3
	02 73 57 4E 20 4C 46 50 69 6E 74 65 72 76 61 6C 46 69 6C 74 65 72 20 31 20 33 03
CoLa B	02 02 02 02 00 00 00 1B 73 57 4E 20 4C 46 50 69 6E 74 65 72 76 61 6C 46 69 6C 74 65 72 20 01 00 00 00 03 0E
	73 57 4E 20 4C 46 50 69 6E 74 65 72 76 61 6C 46 69 6C 74 65 72 20 01 00 00 00 03

Table 173: Telegram structure: sWA LFPintervalFilter

Telegram structure: sWA LFPintervalFilter						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Reduce the scan output rate by a given factor	String	17		LFPintervalFilter	4C 46 50 69 6E 74 65 72 76 61 6C 46 69 6C 74 65 72

Table 174: Example: sWA LFPintervalFilter

CoLa A	<STX>sWA{SPC}LFPintervalFilter<ETX>
	02 73 57 41 20 46 50 69 6E 74 65 72 76 61 6C 46 69 6C 74 65 72 03
CoLa B	02 02 02 02 00 00 00 16 73 57 41 20 4C 46 50 69 6E 74 65 72 76 61 6C 46 69 6C 74 65 72 20 00

12.4.1.4.5.7 Set layer filter [sWN LFPlayerFilter]

Filter complete layers in the output data

Table 175: Telegram structure: sWN LFPlayerFilter

Telegram structure: sWN LFPlayerFilter (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Filter complete layers in the output data	String	14		LFPlayerFilter	4C 46 50 6C 61 79 65 72 46 69 6C 74 65 72
Variable Data 1	Enables/Disables the filter.	Bool_1	1	Off: On:	+0d (0h) +1d (1h)	00 01
Variable Data 2	Selection of the layers for data output	Array of Bool_1	16	Layer 1 off: Layer 1 on: ... Layer 16 off: Layer 16 on:	+0d (0h) +1d (1h) ... +0d (0h) +1d (1h)	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ... 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01

Disable the layer filter and enables each layers

Table 176: Example: sWN LFPlayerFilter

CoLa A	<STX>sWN{SPC}LFPlayerFilter{SPC}0{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1{SPC}1<ETX>	
	<STX>sWN LFPlayerFilter 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1<ETX>	
	sWN LFPlayerFilter 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	
	02 73 57 4E 20 4C 46 50 6C 61 79 65 72 46 69 6C 74 65 72 20 30 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 20 31 03	
CoLa B	02 02 02 02 00 00 00 24 73 57 4E 20 4C 46 50 6C 61 79 65 72 46 69 6C 74 65 72 20 00 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 73	
	73 57 4E 20 4C 46 50 6C 61 79 65 72 46 69 6C 74 65 72 20 00 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01	

Table 177: Telegram structure: sWA LFPlayerFilter

Telegram structure: sWA LFPlayerFilter						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Filter complete layers in the output data	String	14		LFPlayerFilter	4C 46 50 6C 61 79 65 72 46 69 6C 74 65 72

Table 178: Example: sWA LFPlayerFilter

CoLa A	<STX>sWA{SPC}LFPlayerFilter<ETX>
	02 73 57 41 20 4C 46 50 6C 61 79 65 72 46 69 6C 74 65 72 03
CoLa B	02 02 02 02 00 00 00 13 73 57 41 20 4C 46 50 6C 61 79 65 72 46 69 6C 74 65 72 20 7C

12.4.1.4.5.8 Set moving averaging filter [sWN LFPmovingAveragingFilter]

Enables the moving average filter

Table 179: Telegram structure: sWN LFPmovingAveragingFilter

Telegram structure: sWN LFPmovingAveragingFilter (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	String	String	24		LFPmovingAver-agingFilter	4C 46 50 6D 6F 76 69 6E 67 41 76 65 72 61 67 69 6E 67 46 69 6C 74 65 72
Variable Data 1	Moving averaging is enabled	Bool	1	Off: On:	0d (00h) +1d (01h)	00 01
Variable Data 2	averaging depth	UInt	2	Minimum: Maximum:	+2d (02h) +4d (0Ah)	00 02 00 04

Disable the moving average filter and set averaging depth to 3

Table 180: Example: sWN LFPmovingAveragingFilter +0 +3

CoLa A	<STX>sWN{SPC}LFPmovingAveragingFilter{SPC}+0{SPC}+3<ETX>	
	<STX>sWN LFPmovingAveragingFilter +0 +3<ETX>	
	sWN LFPmovingAveragingFilter +0 +3	
	02 73 57 4E 20 4C 46 50 6D 6F 76 69 6E 67 41 76 65 72 61 67 69 6E 67 46 69 6C 74 65 72 20 2B 30 20 2B 33 03	
CoLa B	02 02 02 02 00 00 00 20 73 57 4E 20 4C 46 50 6D 6F 76 69 6E 67 41 76 65 72 61 67 69 6E 67 46 69 6C 74 65 72 20 01 00 03 41	
	73 57 4E 20 4C 46 50 6D 6F 76 69 6E 67 41 76 65 72 61 67 69 6E 67 46 69 6C 74 65 72 20 01 00 03	

Table 181: Telegram structure: sWA LFPmovingAveragingFilter

Telegram structure: sWN LFPmovingAveragingFilter						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	String	String	24		LFPmovingAver-agingFilter	4C 46 50 6D 6F 76 69 6E 67 41 76 65 72 61 67 69 6E 67 46 69 6C 74 65 72

Table 182: Example: sWA LFPmovingAveragingFilter

CoLa A	<STX>sWA{SPC}LFPmovingAveragingFilter<ETX>
	02 73 57 41 20 4C 46 50 6D 6F 76 69 6E 67 41 76 65 72 61 67 69 6E 67 46 69 6C 74 65 72 03
CoLa B	02 02 02 02 00 00 00 1D 73 57 41 20 4C 46 50 6D 6F 76 69 6E 67 41 76 65 72 61 67 69 6E 67 46 69 6C 74 65 72 20 4D

12.4.1.4.5.9 Set radial distance range filter [sWN LFPradialDistanceRangeFilter]

Restriction of the scan(s) to a specified distance range.

Table 183: Telegram structure: sWN LFPradialDistanceRangeFilter

Telegram structure: sWN LFPradialDistanceRangeFilter (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Restriction of the scan(s) to a specified distance range.	String	28		LFPradialDis-tanceRangeFil-ter	4C 46 50 72 61 64 69 61 6C 44 69 73 74 61 6E 63 65 52 61 6E 67 65 46 69 6C 74 65 72
Variable Data 1	Enables/Disables the filter.	Bool_1	1	Off: On:	0d (0h) 1d (1h)	00 01
Variable Data 2	DistMin: Lower boundary of the distance range.	Int_32	4	Minimum: Maximum:	+0d (0h) +200000d (30D40h)	00 00 00 00 00 03 0D 40
Variable Data 3	DistMax: Upper boundary of the distance range.	Int_32	4	Minimum: Maximum:	+0d (0h) +200000d (30D40h)	00 00 00 00 00 03 0D 40

Disable the radial distance range filter and set up the bounderies to min 0mm and max 200000mm.

Table 184: Example: sWN LFPradialDistanceRangeFilter

CoLa A	<STX>sWN{SPC}LFPradialDistanceRangeFilter{SPC}0{SPC}0{SPC}30D40<ETX>	
	<STX>sWN LFPradialDistanceRangeFilter 0 0 30D40<ETX>	
	sWN LFPradialDistanceRangeFilter 0 0 30D40	
	02 73 57 4E 20 4C 46 50 72 61 64 69 61 6C 44 69 73 74 61 6E 63 65 52 61 6E 67 65 46 69 6C 74 65 72 20 00 20 00 20 30D40 03	
CoLa B	02 02 02 02 00 00 00 2A 73 57 4E 20 4C 46 50 72 61 64 69 61 6C 44 69 73 74 61 6E 63 65 52 61 6E 6765 46 69 6C 74 65 72 20 00 00 00 00 00 00 03 0D 40 31	
	73 57 4E 20 4C 46 50 72 61 64 69 61 6C 44 69 73 74 61 6E 63 65 52 61 6E 67 65 46 69 6C 74 65 72 20 00 00 00 00 00 00 03 0D 40	

Table 185: Telegram structure: sWA LFPradialDistanceRangeFilter

Telegram structure: sWA LFPradialDistanceRangeFilter						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Restriction of the scan(s) to a specified distance range.	String	28		LFPradialDis-tanceRangeFil-ter	4C 46 50 72 61 64 69 61 6C 44 69 73 74 61 6E 63 65 52 61 6E 67 65 46 69 6C 74 65 72

Table 186: Example: sWA LFPradialDistanceRangeFilter

CoLa A	<STX>sWA{SPC}LFPradialDistanceRangeFilter<ETX>
	02 73 57 41 20 4C 46 50 72 61 64 69 61 6C 44 69 73 74 61 6E 63 65 52 61 6E 67 65 46 69 6C 74 65 72 03
CoLa B	02 02 02 02 00 00 00 21 73 57 41 20 4C 46 50 72 61 64 69 61 6C 44 69 73 74 61 6E 63 65 52 61 6E 67 65 46 69 6C 74 65 72 20 70

12.4.1.4.6 Inputs and Outputs

12.4.1.4.6.1 Read state of the ports [sRN LIDportstate]

LIDportstate has to be available additionally or as successor of the LIDoutputstate telegram.

Valid for all sensors with Ethernet and ports (inputs / outputs).

Table 187: Telegram structure: sRN LIDportstate

Telegram structure: sRN LIDportstate						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Ask for port configuration	String	12		LIDportstate	4C 49 44 70 6F 72 74 73 74 61 74 65

Table 188: Example: sRN LIDportstate

CoLa A	<STX>sRN{SPC}LIDportstate<ETX>	
	<STX>sRN LIDportstate<ETX>	
	sRN LIDportstate	
	02 73 52 4E 20 4C 49 44 70 6F 72 74 73 74 61 74 65 03	
CoLa B	02 02 02 02 00 00 00 10 73 52 4E 20 4C 49 44 70 6F 72 74 73 74 61 74 65 60	
	73 52 4E 20 4C 49 44 70 6F 72 74 73 74 61 74 65	

Table 189: Telegram structure: sRA LIDportstate

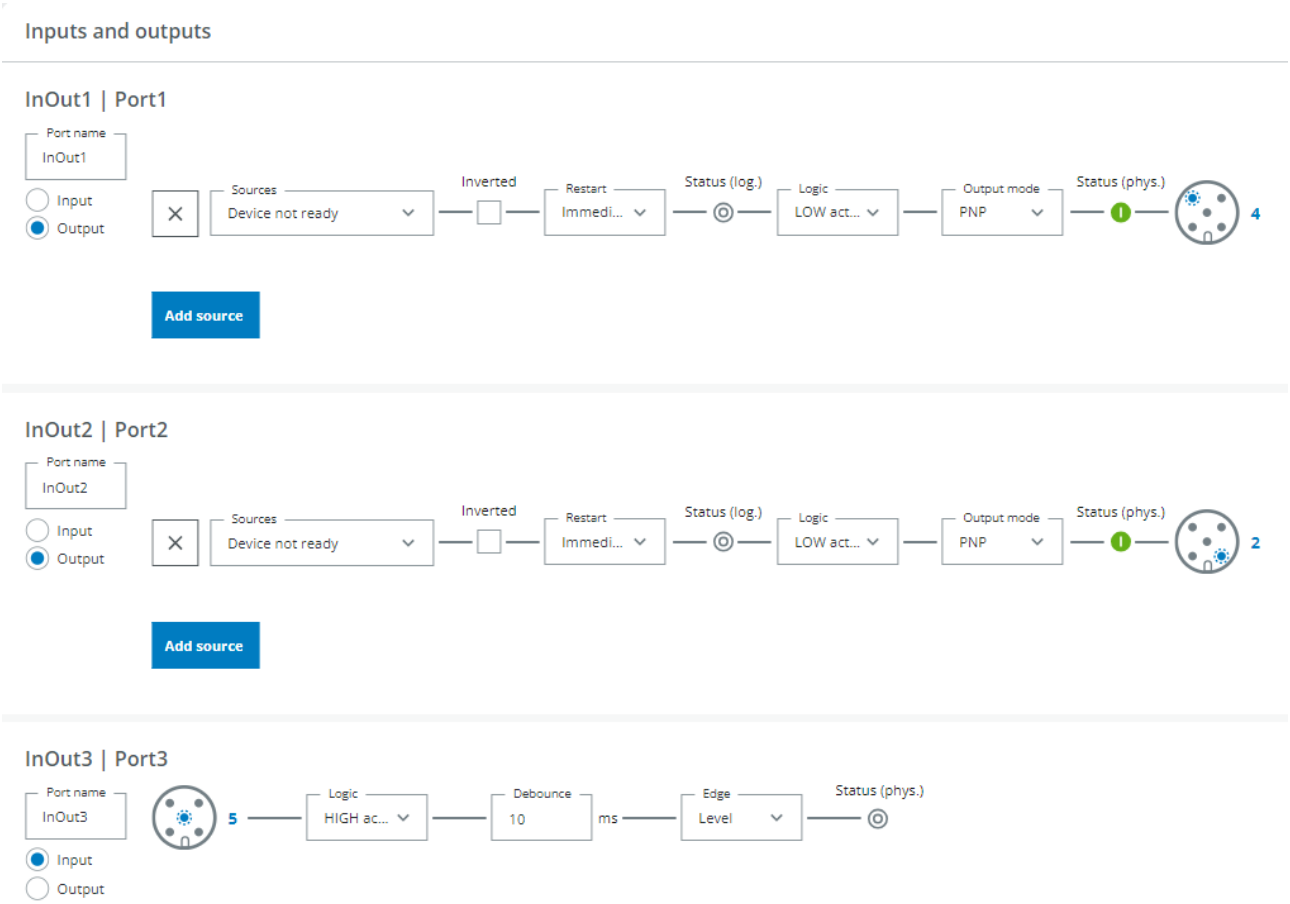
Telegram structure: sRA LIDportstate						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Port state	String	12		LIDportstate	4C 49 44 70 6F 72 74 73 74 61 74 65
Status code	Version number	Uint_16	2	Current version:	0 ... FFFFh 0	00 01 ... FF FF
	System counter (time in µs since power up max. 71min then starting from 0 again)	Uint_32	4		0 ... FFFFFFFFh	00 00 00 00 ... FF FF FF FF

Telegram structure: sRA LIDportstate						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
ARRAY which defines the number of internal ports*	Array includes 8x **internal port states** and 8x **external port states.** Structure of ports see following rows	Uint_16	2	Hex: Not available: Number of ports:	0000 - FFFF 00 01 ... n	00 00 - FF FF
State of the ports and count value in hex	Internal port state	Enum_8	1	Output voltage low: (Relays open) Output voltage high: (Relays closed) Tri-state: Input voltage high (level): Input voltage from low to high (edge) Input voltage low (level): Input voltage high to low (edge)	00 01 02 03 04	00 01 02 03 04
	Internal port counter	Uint_32	4		0 ... FFFFFFFFh	00 00 00 00 ... FF FF FF FF
	.....					
ARRAY which defines the number of external or **virtual** ports*	0...n	Uint_16	1	Hex: Not available: Numer of ports:	00 00 - FF FF 00 01 ... n	00 00 - FF FF
State of the ports and count value in hex	External port state	Enum_8	1	Output voltage low: (Relays open) Output voltage high: (Relays closed) Tri-state: Input voltage high (level): Input voltage from low to high (edge) Input voltage low (level): Input voltage high to low (edge)	00 01 02 03 04	00 01 02 03 04
	External port counter	Uint_32	4		0 ... FFFFFFFFh	00 00 00 00 ... FF FF FF FF
Time	States code	Enum_16	1	No time data: Time data:	00 00 00 01	00 00 00 01
Time Block (sensor time from the last change of min. one of the outputs)	Year	Array	2	E.g.	1970	07 B2
	Month		1		1 ... 12	01 ... 0C
	Day		1		1 ... 31	01 ... 1F
	Hour		1		0 ... 23	00 ... 17
	Minute		1		0 ... 59	00 ... 3B
	Second		1		0 ... 59	00 ... 3B
	Microsecond		4		0 ... 999999	00 00 00 00 ... 00 0F 42 3F

Inputs/outputs: If the device has separate inputs and outputs (instead of general purpose ports) the ARRAY shall start with inputs followed by the outputs.

Virtual ports are ports that can be used to expand the number of ports but are not physically available. They just show up in the corresponding ethernet telegrams (like LIDportstate).

Tri-State: Port is neither input nor output; the port is set inactive in SOPAS



Example default parameter of mulitScan136 with 3 ports. Ports configuration: Internal port counter of port 1,2 is 2

Table 190: Example: sRA LIDportstate

CoLa A	<STX>sRA{SPC}LIDport-state{SPC}0{SPC}5F0FF0{SPC}1{SPC}2{SPC}1{SPC}2{SPC}4{SPC}0{SPC}2{SPC}0{SPC}2{SPC}0{SPC}2{SPC}0{SPC}2{SPC}0{SPC}2{SPC}0{SPC}2{SPC}0{SPC}2{SPC}0{SPC}2{SPC}0{SPC}2{SPC}0{SPC}2{SPC}0{SPC}2{SPC}0{SPC}2{SPC}0{SPC}2{SPC}0{SPC}2{SPC}0{SPC}1{SPC}7B2{SPC}1{SPC}1{SPC}0{SPC}0{SPC}6{SPC}38270<ETX>
	02 73 52 41 20 4C 49 44 70 6F 72 74 73 74 61 74 65 20 30 20 35 46 30 46 46 30 20 31 20 32 20 31 20 32 20 34 20 30 20 32 20 30 20 32 20 30 20 32 20 30 20 32 20 30 20 32 20 30 20 32 20 30 20 32 20 30 20 32 20 30 20 32 20 30 20 32 20 30 20 32 20 30 20 32 20 30 20 32 20 30 20 32 20 30 20 32 20 30 20 31 20 37 42 32 20 31 20 31 20 30 20 30 20 36 20 33 38 32 37 30 03
CoLa B	02 02 02 02 00 00 00 74 73 52 41 20 4C 49 44 70 6F 72 74 73 74 61 74 65 20 00 00 00 5F 0F F0 01 00 00 00 02 01 00 00 00 02 04 00 00 00 00 02 00 00 00 00 02 00 00 00 00 02 00 00 00 00 02 00 00 00 00 02 00 00 00 00 02 00 00 00 00 02 00 00 00 00 02 00 00 00 00 02 00 00 00 00 02 00 00 00 00 02 00 00 00 00 02 00 00 00 00 02 00 00 00 00 02 00 00 00 00 02 00 00 00 00 02 00 00 00 00 02 00 00 00 00 02 00 00 00 00 02 00 00 00 00 01 07 B2 01 01 00 00 06 00 03 82 70 AA

12.4.1.4.6.2 Read Port Configration of all I/Os [sRN PortConfiguration]

Table 191: Telegram structure: sRN PortConfiguration

Telegram structure: sRN PortConfiguration						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Ask for port configuration	String	12		PortConfiguration	50 6F 72 74 43 6F 6E 66 69 67 75 72 61 74 69 6F 6E

Table 192: Example: sRN PortConfiguration

CoLa A	<STX>sRN{SPC}PortConfiguration<ETX>	
	<STX>sRN PortConfiguration<ETX>	
	sRN PortConfiguration	
	02 73 52 4E 20 50 6F 72 74 43 6F 6E 66 69 67 75 72 61 74 69 6F 6E 03	
CoLa B	02 02 02 02 00 00 00 15 73 52 4E 20 50 6F 72 74 43 6F 6E 66 69 67 75 72 61 74 69 6F 6E 26	
	73 52 4E 20 50 6F 72 74 43 6F 6E 66 69 67 75 72 61 74 69 6F 6E	

Table 193: Telegram structure: sRA PortConfiguration

Telegram structure: sRA PortConfiguration						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Configuration of all I/Os	String	12		PortConfiguration	50 6F 72 74 43 6F 6E 66 69 67 75 72 61 74 69 6F 6E
Start of loop, number of loops = amount of all current and future Inputs and Outputs of device family						
Port Type	Input or Output	Enum_8	1	Input: Output:	0 1	00 01
Port Name	Amount of characters of the following port name	Uint_16	2		0h ... 20h	00 00 ... 00 20
	Port name	String	16 (depending on string length)		[Port name]	[Port name]
Input Settings						
Logic	Logic of the input	Bool_1	1	Active high: Active low:	0 1	00 01
Debouncing	Select debouncing time in ms	Uint_16	2	(max. 10,000ms)	0h ... 2710h	00 00 ... 27 10
Sensitivity	Status change at Edge or Level	Enum_8	1	Edge: Level:	0 1	00 01

Telegram structure: sRA PortConfiguration						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Reserved	Reserved value 1	Uint_16	2		0	00 00
Reserved	Reserved value 2	Uint_16	2		0	00 00
Output Settings						
Logic	Logic of the input	Bool_1	1	Active high: Active low:	0 1	00 01
Output Mode	PNP, NPN or Push-Pull	Enum_8	1	PNP: NPN: Push-Pull:	0 1 2	00 01 02
Restart type	Restart behavior of output after event: immediatly or after specific time	Enum_8	1	Immediately: Time: Input:	0 1 2	00 01 02
Restart time	[Only with restart type = Time], time in ms	Uint_32	4	(20 ms ... 600,000 ms)	14h ... 927C0h	00 00 00 14 ... 00 09 27 C0
Restart input	[Only with restart type = Input], input for restart	Uint_16	2		1 ... 8	00 01 ... 00 08
Combination	Combining multiple Events and/or Inputs	Enum_8	1	AND: OR: XOR:	0 1 2	00 01 02
Reserved	Reserved value 3	Uint_16	2		0	00 00
Reserved	Reserved value 4	Uint_16	2		0	00 00
Sources	Amount (n) of combined sources	Uint_16	2		0h ... FFFFh	00 00 ... FF FF
	Start of source loop, number of loops = amount of combined sources					
	Source name	String	4	Device Ready: Sopas Command: (xx = Number of output port) Indexsignal: SyncOutByClock: Input = Port Name: (INxx = Number of input port)	[Source] DRDY SCxx SROT SCLK INxx	[Source]
	Source Inverted or not	Bool_1	1	Not inverted: Inverted:	0 1	00 01
	Reserved value 5	Uint_8	1		0	00
	Reserved value 6	Uint_8	1		0	00
	Stop of source loop					
Reserved	Reserved value 7	Uint_16	2		0	00 00
Reserved	Reserved value 8	Uint_16	2		0	00 00
Reserved	Reserved value 9	Uint_16	2		0	00 00
Reserved	Reserved value 10	Uint_16	2		0	00 00
Stop of loop						

Table 194: Example: sRA ProtConfiguration

CoLa A	<STX>sRA{SPC}PortConfigura- tion{SPC}1{SPC}6{SPC}InOut1{SPC}0{SPC}A{SPC}1{SPC}0{SPC}0{SPC}1{SPC}0{SPC}0{SPC}C8{SPC}1{SPC}1{SPC}0{SPC}0{SPC}1{SPC}DRDY{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}1{SPC}6{SPC}InOut2{SPC}0{SPC}A{SPC}1{SPC}0{SPC}0{SPC}1{SPC}0{SPC}0{SPC}C8{SPC}1{SPC}1{SPC}0{SPC}0{SPC}1{SPC}DRDY{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}6{SPC}InOut3{SPC}0{SPC}A{SPC}1{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}C8{SPC}1{SPC}1{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}6{SPC}InOut4{SPC}0{SPC}A{SPC}1{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}6{SPC}InOut5{SPC}0{SPC}A{SPC}1{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}C8{SPC}1{SPC}1{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}6{SPC}InOut6{SPC}0{SPC}A{SPC}1{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}C8{SPC}1{SPC}1{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}1{SPC}6{SPC}InOut7{SPC}0{SPC}A{SPC}1{SPC}0{SPC}0{SPC}1{SPC}0{SPC}C8{SPC}1{SPC}1{SPC}0{SPC}0{SPC}1{SPC}DRDY{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}1{SPC}6{SPC}InOut8{SPC}0{SPC}A{SPC}1{SPC}0{SPC}0{SPC}1{SPC}0{SPC}0{SPC}0{SPC}C8{SPC}1{SPC}1{SPC}0{SPC}0{SPC}1{SPC}DRDY{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0<ETX>	
	<STX>sRA PortConfiguration 1 6 InOut1 0 A 1 0 0 1 0 0 C8 1 1 0 0 1 DRDY 0 0 0 0 0 0 0 1 6 InOut2 0 A 1 0 0 1 0 0 C8 1 1 0 0 1 DRDY 0 0 0 0 0 0 0 0 6 InOut3 0 A 1 0 0 0 0 0 C8 1 1 0 0 0 0 0 0 0 0 6 InOut4 0 A 1 0 0 0 0 0 C8 1 1 0 0 0 0 0 0 0 0 6 InOut5 0 A 1 0 0 0 0 0 C8 1 1 0 0 0 0 0 0 0 0 6 InOut6 0 A 1 0 0 0 0 0 C8 1 1 0 0 0 0 0 0 0 1 6 InOut7 0 A 1 0 0 1 0 0 C8 1 1 0 0 1 DRDY 0 0 0 0 0 0 0 1 6 InOut8 0 A 1 0 0 1 0 0 C8 1 1 0 0 1 DRDY 0 0 0 0 0 0 0<ETX>	
	sRA PortConfiguration 1 6 InOut1 0 A 1 0 0 1 0 0 C8 1 1 0 0 1 DRDY 0 0 0 0 0 0 0 1 6 InOut2 0 A 1 0 0 1 0 0 C8 1 1 0 0 1 DRDY 0 0 0 0 0 0 0 0 6 InOut3 0 A 1 0 0 0 0 0 C8 1 1 0 0 0 0 0 0 0 0 6 InOut4 0 A 1 0 0 0 0 0 C8 1 1 0 0 0 0 0 0 0 0 6 InOut5 0 A 1 0 0 0 0 0 C8 1 1 0 0 0 0 0 0 0 0 6 InOut6 0 A 1 0 0 0 0 0 C8 1 1 0 0 0 0 0 0 0 1 6 InOut7 0 A 1 0 0 1 0 0 C8 1 1 0 0 1 DRDY 0 0 0 0 0 0 0 1 6 InOut8 0 A 1 0 0 1 0 0 C8 1 1 0 0 1 DRDY 0 0 0 0 0 0 0	
CoLa B	02 02 02 02 00 00 01 7A 73 52 41 20 50 6F 72 74 43 6F 6E 66 69 67 75 72 61 74 69 6F 6E 20 01 00 06 49 6E 4F 75 74 31 00 00 0A 01 00 00 00 00 01 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 01 44 52 44 59 00 00 00 00 00 00 00 00 00 00 00 01 00 06 49 6E 4F 75 74 32 00 00 0A 01 00 00 00 00 01 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 01 44 52 44 59 00 00 00 00 00 00 00 00 00 00 00 00 06 49 6E 4F 75 74 33 00 00 0A 01 00 00 00 00 00 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 06 49 6E 4F 75 74 34 00 00 0A 01 00 00 00 00 00 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 06 49 6E 4F 75 74 35 00 00 0A 01 00 00 00 00 00 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 06 49 6E 4F 75 74 36 00 00 0A 01 00 00 00 00 00 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 00 00 00 00 00 00 00 01 00 06 49 6E 4F 75 74 37 00 00 0A 01 00 00 00 00 01 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 01 44 52 44 59 00 00 00 00 00 00 00 00 00 00 01 00 06 49 6E 4F 75 74 38 00 00 0A 01 00 00 00 00 01 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 01 44 52 44 59 00 00 00 00 00 00 00 00 00 00 00 01	
	73 52 41 20 50 6F 72 74 43 6F 6E 66 69 67 75 72 61 74 69 6F 6E 20 01 00 06 49 6E 4F 75 74 31 00 00 0A 01 00 00 00 00 01 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 01 44 52 44 59 00 00 00 00 00 00 00 00 00 00 00 01 00 06 49 6E 4F 75 74 32 00 00 0A 01 00 00 00 00 01 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 01 44 52 44 59 00 00 00 00 00 00 00 00 00 00 00 00 06 49 6E 4F 75 74 33 00 00 0A 01 00 00 00 00 00 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 06 49 6E 4F 75 74 34 00 00 0A 01 00 00 00 00 00 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 06 49 6E 4F 75 74 35 00 00 0A 01 00 00 00 00 00 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 06 49 6E 4F 75 74 36 00 00 0A 01 00 00 00 00 00 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 00 00 00 00 00 00 00 01 00 06 49 6E 4F 75 74 37 00 00 0A 01 00 00 00 00 01 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 01 44 52 44 59 00 00 00 00 00 00 00 00 00 00 01 00 06 49 6E 4F 75 74 38 00 00 0A 01 00 00 00 00 01 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 01 44 52 44 59 00 00 00 00 00 00 00 00 00 00 00	

12.4.1.4.6.3 Set port configuration [sWN PortConfiguration]

Configuration of the given ports. Telegram structure represents the configuration of 1 port. If the device has multiple ports, use the same structure.

Table 195: Telegram structure: sWN PortConfiguration

Telegram structure: sWN PortConfiguration (Required User Level: authorized client)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Configuration of the given ports	String	17		PortConfigura-tion	50 6F 72 74 43 6F 6E 66 69 67 75 72 61 74 69 6F 6E
Port Type	Input or Output	Enum_8	1	Input: Output:	+0d (0h) +1d (1h)	00 01
Name	Name of the port	FlexString	9 (0..32)	Default:	InOut1	00 06 49 6E 4F 75 74 31 00
Input Settings						
Logic	Logic of the input	Enum_8	1	Active high: Active low:	+0d (0h) +1d (1h)	00 01
Debounce	Select debouncing time in ms	Uint_8	1		+0d ... +255d (0h ... FFh)	00 ... FF
Sensitivity	Status change at Edge or Level	Enum_8	1	Edge: Level:	+0d (0h) +1d (1h)	00 01
Reserved1		Uint_16	2		+0d (0h)	00 00
Reserved2		Uint_16	2		+0d (0h)	00 00
Output Settings						
Logic	Definition of the output logic	Enum_8	1	Active high: Active low:	+0d (0h) +1d (1h)	00 01
Output Mode	Set kind of mode for output pin	Enum_8	1	PNP: NPN: Push-Pull:	0 1 2	00 01 02
Restart Type	Defines type of restart to be used	Enum_8	1	Immediately: Time: :	+0d (0h) +1d (1h) +2d (2h)	00 01 02
Restart Time	[Only with restart type = Time], time in ms	Uint_32	4	(20 ms ... 600,000 ms)	+20d ... +600000d (C8h ... 927C0h)	00 00 00 00 ... 00 09 27 C0
Restart Input	[Only with restart type = Input], input for restart	Uint_16	2		+1d ... +8d (1h ... 8h)	00 00 ... 00 08
Combina-tion	Combining multiple Events and/or Inputs	Enum_8	1	AND: OR: XOR:	+0d (0h) +1d (1h) +2d (2h)	
Reserved	Reserved value 3	Uint_16	2		+0d (0h)	00 00
Reserved	Reserved value 4	Uint_16	2		+0d (0h)	00 00
	Source					
Source	The source parameter are only existing if the port is set to OUTPUT!		2		+1d (1h)	00 01

Telegram structure: sWN PortConfiguration (Required User Level: authorized client)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Source Name	Name of the source option	String	4	DeviceNotReady: Input1: Input2: SopasCommand:	DRDY IN01 IN02 SC01	44 52 44 59 49 4E 30 31 49 4E 30 32 53 43 30 31
Invert	Invert the source signal	Bool_1	1	Not inverted: Inverted:	+0d (0h) +1d (1h)	00 01
Reserved	Reserved value 5	Uint_8	1		+0d (0h)	00
Reserved	Reserved value 6	Uint_8	1		+0d (0h)	00
Reserved						
Reserved	Reserved value 7	Uint_16	2		+0d (0h)	00 00
Reserved	Reserved value 8	Uint_16	2		+0d (0h)	00 00
Reserved	Reserved value 9	Uint_16	2		+0d (0h)	00 00
Reserved	Reserved value 10	Uint_16	2		+0d (0h)	00 00

Example multiScan136 with 3 ports: sWN PortConfiguration

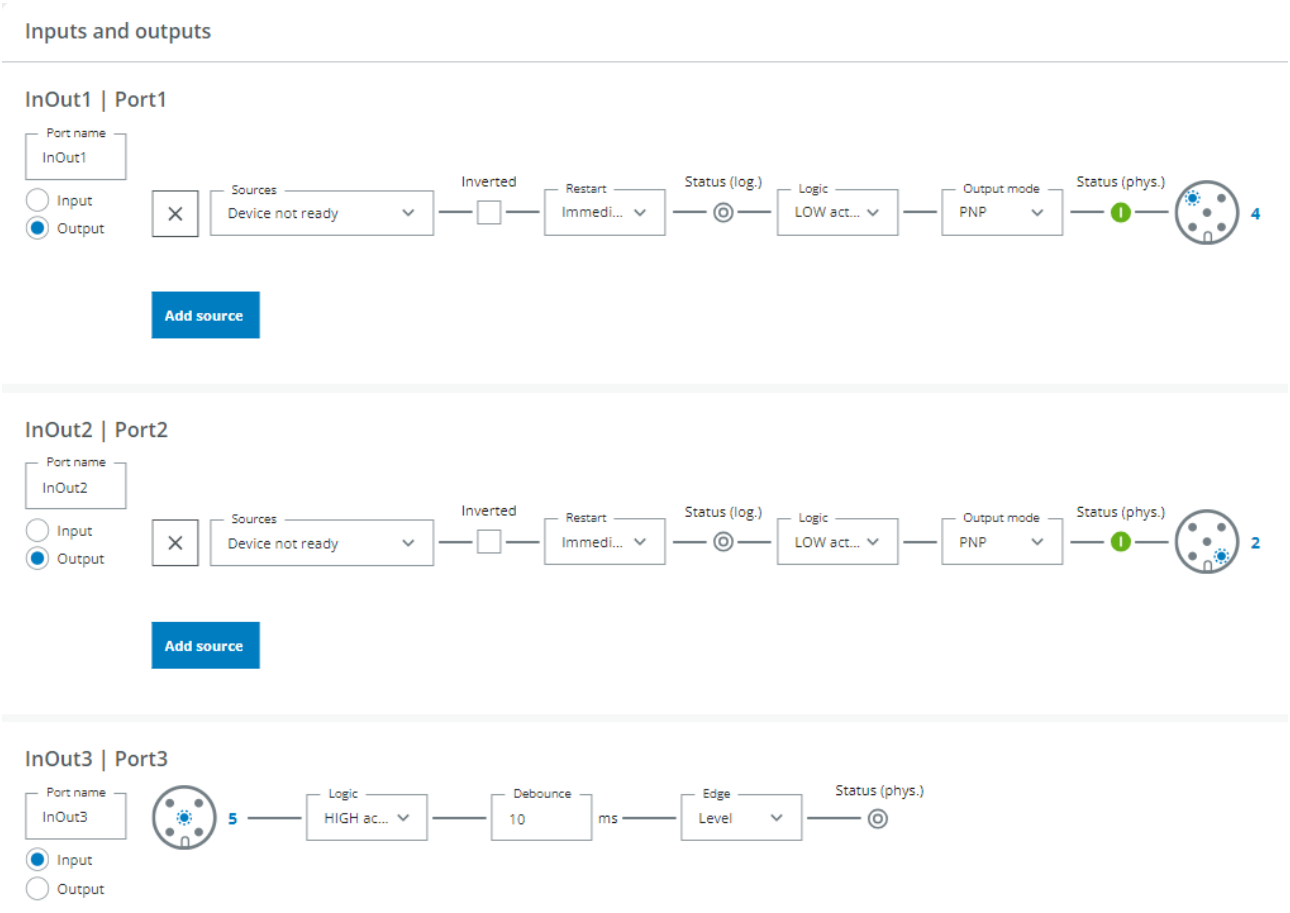


Figure 43: Example multiScan136 with 3 ports

Table 196: Example: sWN ProtConfiguration

CoLa A	<STX>sWN{SPC}PortConfiguration{SPC}1{SPC}6{SPC}InOut1{SPC}0{SPC}A{SPC}1{SPC}0{SPC}0{SPC}1{SPC}0{SPC}0{SPC}C8{SPC}1{SPC}1{SPC}0{SPC}0{SPC}1{SPC}DRDY{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}1{SPC}6{SPC}InOut2{SPC}0{SPC}A{SPC}1{SPC}0{SPC}0{SPC}1{SPC}0{SPC}0{SPC}C8{SPC}1{SPC}1{SPC}0{SPC}0{SPC}1{SPC}DRDY{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}6{SPC}InOut3v0{SPC}A{SPC}1{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}C8{SPC}1{SPC}1{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0{SPC}0<ETX>
	<STX>sWN PortConfiguration 1 6 InOut1 0 A 1 0 0 1 0 0 C8 1 1 0 0 1 DRDY 0 0 0 0 0 0 0 1 6 InOut2 0 A 1 0 0 1 0 0 C8 1 1 0 0 1 DRDY 0 0 0 0 0 0 0 0 6 InOut3 0 A 1 0 0 0 0 0 C8 1 1 0 0 0 0 0 0 0<ETX>
	sWN PortConfiguration 1 6 InOut1 0 A 1 0 0 1 0 0 C8 1 1 0 0 1 DRDY 0 0 0 0 0 0 0 1 6 InOut2 0 A 1 0 0 1 0 0 C8 1 1 0 0 1 DRDY 0 0 0 0 0 0 0 0 6 InOut3 0 A 1 0 0 0 0 0 C8 1 1 0 0 0 0 0 0 0
	02 73 57 4E 20 50 6F 72 74 43 6F 6E 66 69 67 75 72 61 74 69 6F 6E 20 31 20 36 20 49 6E 4F 75 74 31 20 30 20 41 20 31 20 30 20 30 20 31 20 30 20 30 20 43 38 20 31 20 31 20 30 20 30 20 31 20 44 52 44 59 20 30 20 30 20 30 20 30 20 30 20 30 20 30 20 30 20 31 20 36 20 49 6E 4F 75 74 32 20 30 20 41 20 31 20 30 20 30 20 31 20 30 20 30 20 43 38 20 31 20 31 20 30 20 30 20 31 20 44 52 44 59 20 30 20 30 20 30 20 30 20 30 20 30 20 30 20 30 20 30 20 36 20 49 6E 4F 75 74 33 20 30 20 41 20 31 20 30 20 30 20 30 20 30 20 30 20 30 20 43 38 20 31 20 31 20 30 20 30 20 30 20 30 20 30 20 30 20 30 20 30 03
CoLa B	02 02 02 02 00 00 00 9F 73 57 4E 20 50 6F 72 74 43 6F 6E 66 69 67 75 72 61 74 69 6F 6E 20 01 00 06 49 6E 4F 75 74 31 00 00 0A 01 00 00 00 00 01 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 01 44 52 44 59 00 00 00 00 00 00 00 00 00 00 01 00 06 49 6E 4F 75 74 32 00 00 0A 01 00 00 00 00 01 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 01 44 52 44 59 00 00 00 00 00 00 00 00 00 00 00 00 06 49 6E 4F 75 74 33 00 00 0A 01 00 00 00 00 00 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 9F
	73 57 4E 20 50 6F 72 74 43 6F 6E 66 69 67 75 72 61 74 69 6F 6E 20 01 00 06 49 6E 4F 75 74 31 00 00 0A 01 00 00 00 00 01 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 01 44 52 44 59 00 00 00 00 00 00 00 00 00 00 01 00 06 49 6E 4F 75 74 32 00 00 0A 01 00 00 00 00 01 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 01 44 52 44 59 00 00 00 00 00 00 00 00 00 00 00 00 06 49 6E 4F 75 74 33 00 00 0A 01 00 00 00 00 00 00 00 00 00 00 C8 00 01 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Table 197: Telegram structure: sWA PortConfiguration

Telegram structure: sWA PortConfiguration						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Configuration of the given ports	String	17		PortConfigura-tion	50 6F 72 74 43 6F 6E 66 69 67 75 72 61 74 69 6F 6E

Table 198: Example: sWA PortConfiguration

CoLa A	<STX>sWA{SPC}PortConfiguration<ETX>
	02 73 57 41 20 50 6F 72 74 43 6F 6E 66 69 67 75 72 61 74 69 6F 6E 03
CoLa B	02 02 02 02 00 00 00 16 73 57 41 20 50 6F 72 74 43 6F 6E 66 69 67 75 72 61 74 69 6F 6E 20 0C

12.4.1.4.6.4 Read state of the inputs [sRN LIDinputstate]

Use `sEN LIDinputstate 1` to receive a telegram each time an input signal (e.g. by trigger) changes. Compare with chapter "Receive outputstate by event [sEN LIDoutputstate]", page 128.

Table 199: Telegram structure: sRN LIDinputstate

Telegram structure: sRN LIDinputstate						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Input state	String	14		LIDinputstate	4C 49 44 69 6E 70 75 74 73 74 61 74 65

Table 200: Example: sRN LIDinputstate

CoLa A	<STX>sRN{SPC}LIDinputstate<ETX>	
	<STX>sRN LIDinputstate<ETX>	
	sRN LIDinputstate	
	02 73 52 4E 20 4C 49 44 69 6E 70 75 74 73 74 61 74 65 03	
CoLa B	02 02 02 02 00 00 00 11 73 52 4E 20 4C 49 44 69 6E 70 75 74 73 74 61 74 65 0F	
	73 52 4E 20 4C 49 44 69 6E 70 75 74 73 74 61 74 65	

Table 201: Telegram structure: sRA LIDinputstate

Telegram structure: sRA LIDinputstate						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Output state	String	14		LIDinputstate	4C 49 44 69 6E 70 75 74 73 74 61 74 65
Status code	Version number	Uint_16	2		0h ... FFFFh	00 00 ... FF FF
	System counter (time in µs since power up max. 71min then starting from 0 again)	Uint_32	4		0 h ... FFFFFFFFh	00 00 00 00 ... FF FF FF FF
State of the inputs 1 ... n	Amount of inputs (n) depending of device family	Enum_8	1	Not active: Active: Input not used:	0 1 2	00 01 02
Time	States code	Uint_16	2	No time data: Time data:	0 1	00 00 00 01
Time Block (sensor-time from the last change of min. one of the out-puts)	Year	Array	2	E. g.	1970	07 B2
	Month		1		1 ... 12	01 ... 0C
	Day		1		1 ... 31	01 ... 1F
	Hour		1		0 ... 23	00 ... 17
	Minute		1		0 ... 59	00 ... 3B
	Second		1		0 ... 59	00 ... 3B
	Microsecond		4		0 ... 999999	00 00 00 00 ... 00 0F 42 3F

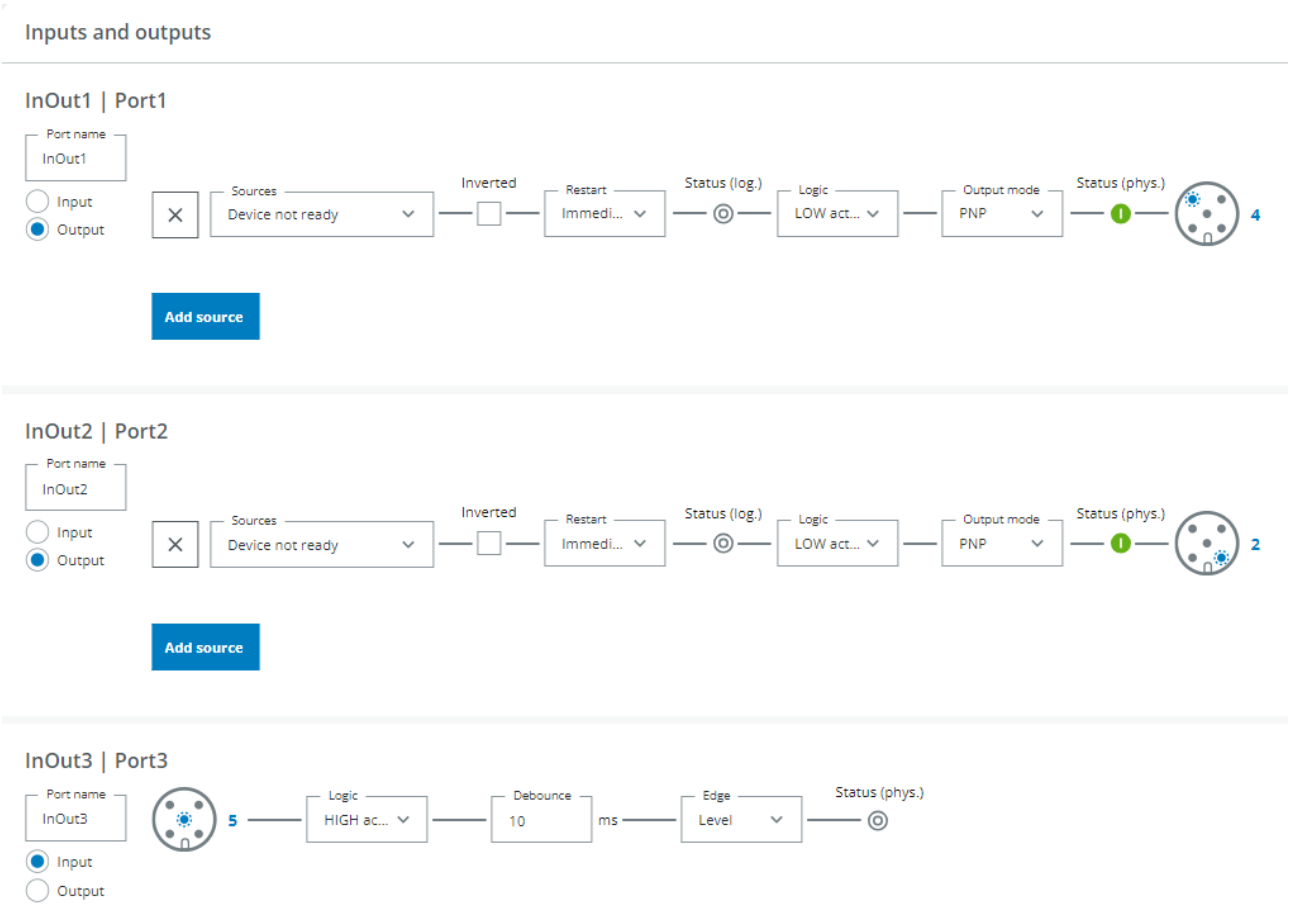


Table 202: Example: sRA LIDinputstate default settings with port 3 set as input: In1 not used, In2 not used, In3 inactive, In4 not used, In5 not used, In6 not used, In7 not used, In8 not used, time: 1970-01-01 00:00 5 sec 665,000 microseconds

CoLa A	<STX>sRA{SPC}LIDinput-state{SPC}1{SPC}566D00{SPC}2{SPC}2{SPC}0{SPC}2{SPC}2{SPC}2{SPC}2{SPC}2{SPC}1{SPC}7B2{SPC}1{SPC}1{SPC}0{SPC}0{SPC}5{SPC}A25A8<ETX>
	02 73 52 41 20 4C 49 44 69 6E 70 75 74 73 74 61 74 65 20 31 20 35 36 36 44 30 30 20 32 20 32 20 30 20 32 20 32 20 32 20 32 20 32 20 31 20 37 42 32 20 31 20 31 20 30 20 30 20 35 20 41 32 35 41 38 03
CoLa B	02 02 02 02 00 00 00 2D 73 52 41 20 4C 49 44 69 6E 70 75 74 73 74 61 74 65 20 00 01 00 56 6D 00 02 02 00 02 02 02 02 02 00 01 07 B2 01 01 00 00 05 00 0A 25 A8 2E

12.4.1.4.6.5 Read state of the outputs [sRN LIDoutputstate]

Status of all outputs

Table 203: Telegram structure: sRN LIDoutputstate

Telegram structure: sRN LIDoutputstate						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Output state	String	14		LIDoutputstate	4C 49 44 6F 75 74 70 75 74 73 74 61 74 65

Table 204: Example: sRN LIDoutputstate

CoLa A	<STX>sRN{SPC}LIDoutputstate<ETX>
	<STX>sRN LIDoutputstate<ETX>
	sRN LIDoutputstate
	02 73 52 4E 20 4C 49 44 6F 75 74 70 75 74 73 74 61 74 65 03
CoLa B	02 02 02 02 00 00 00 12 73 52 4E 20 4C 49 44 6F 75 74 70 75 74 73 74 61 74 65 66
	73 52 4E 20 4C 49 44 6F 75 74 70 75 74 73 74 61 74 65

Table 205: Telegram structure: sRA LIDoutputstate

Telegram structure: sRA LIDoutputstate						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Complete telegram structure of the answer see "Receive outputstate by event [sEN LIDoutputstate]", page 128.						

12.4.1.4.6.6 Receive outputstate by event [sEN LIDoutputstate]

Output telegram is sent every time an output state changes.

Table 206: Telegram structure: sEN LIDoutputstate

Telegram structure: sEN LIDoutputstate						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Event	String	3		sEN	73 45 4E
Command	Output state	String	14		LIDoutputstate	4C 49 44 6F 75 74 70 75 74 73 74 61 74 65
	Start/stop	Enum_8	1	Start: Stop:	1 0	01 00

Table 207: Example: sEN LIDoutputstate

CoLa A	<STX>sEN{SPC}LIDoutputstate{SPC}1<ETX>
	<STX>sEN LIDoutputstate 1<ETX>
	sEN LIDoutputstate 1
	02 73 45 4E 20 4C 49 44 6F 75 74 70 75 74 73 74 61 74 65 20 31 03
CoLa B	02 02 02 02 00 00 00 14 73 45 4E 20 4C 49 44 6F 75 74 70 75 74 73 74 61 74 65 20 01 50
	73 45 4E 20 4C 49 44 6F 75 74 70 75 74 73 74 61 74 65 20 01

Table 208: Telegram structure: sRA/sSN LIDoutputstate

Telegram structure: sRA/sSN LIDoutputstate						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA / sSN	73 52 41 / 73 53 4E

Telegram structure: sRA/sSN LIDoutputstate						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command	Output state	String	14		LIDoutputstate	4C 49 44 6F 75 74 70 75 74 73 74 61 74 65
Status code	Version number	Uint_16	2		0h ... FFFFh	00 00 ... FF FF
	System counter (time in µs since power up max. 71min then starting from 0 again)	Uint_32	4		0h ... FFFFFFFFh	00 00 00 00 ... FF FF FF FF
State of the outputs 1 ... n and count value in hex. (values of an example) Amount of outputs (n) depending of device family	Output 1 ... n state	Enum_8	1	Not active: Active: Output not used:	0 1 2	00 01 02
	Output 1 ... n count	Uint_32	4		0h ... FFFFFFFFh	00 00 00 00 ... FF FF FF FF
Time	States code	Uint_16	2	No time data: Time data:	0 1	00 00 00 01
Time Block (sensor-time from the last change of min. one of the outputs)	Year	Array	2	E. g.	1970	07 B2
	Month		1		1 ... 12	01 ... 0C
	Day		1		1 ... 31	01 ... 1F
	Hour		1		0 ... 23	00 ... 17
	Minute		1		0 ... 59	00 ... 3B
	Second		1		0 ... 59	00 ... 3B
	Microsecond		4		0 ... 999999	00 00 00 00 ... 00 0F 42 3F

12.4.1.4.6.7 Set output state [sMN mDOSetOutput]

NOTE
Output source needs to be set to "SOPAS command" and the port configured as Output (in case of I/O).

Table 209: Telegram structure: sMN mDOSetOutput

Telegram structure: sMN mDOSetOutput						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E
Command	Set output state	String	12		mDOSetOutput	6D 44 4F 53 65 74 4F 75 74 70 75 74

Telegram structure: sMN mDOSetOutput						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Output number		Uint_8	1		1 ... 3	01 ... 03
Output state		Enum_8	1	Not active: Active:	0 1	00 01

Table 210: Example: sMN mDOSetOutput

CoLa A	<STX>sMN{SPC}mDOSetOutput{SPC}1{SPC}1<ETX>
	<STX>sMN mDOSetOutput 1 1<ETX>
	sMN mDOSetOutput 1 1
	02 73 4D 4E 20 6D 44 4F 53 65 74 4F 75 74 70 75 74 20 31 20 31 03
CoLa B	02 02 02 02 00 00 00 13 73 4D 4E 20 6D 44 4F 53 65 74 4F 75 74 70 75 74 20 01 01 6B
	73 4D 4E 20 6D 44 4F 53 65 74 4F 75 74 70 75 74 20 01 01

Table 211: Telegram structure: sAN mDOSetOutput

Telegram structure: sAN mDOSetOutput						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 41 4E
Command	Set output state	String	12		mDOSetOutput	6D 44 4F 53 65 74 4F 75 74 70 75 74
Status Code	Status code	Bool_1	1	Error: Success:	0 1	00 01

Table 212: Example: sAN mDOSetOutput

CoLa A	<STX>sAN{SPC}mDOSetOutput{SPC}1<ETX>
	02 73 41 4E 20 6D 44 4F 53 65 74 4F 75 74 70 75 74 20 31 03
CoLa B	02 02 02 02 00 00 00 12 73 41 4E 20 6D 44 4F 53 65 74 4F 75 74 70 75 74 20 01 66

12.4.1.4.7 Status

12.4.1.4.7.1 Read firmware version [sRN DeviceIdent]

Table 213: Telegram structure: sRN DeviceIdent

Telegram structure: sRN DeviceIdent						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Read firmware version	String	11		DeviceIdent	44 65 76 69 63 65 49 64 65 6E 74

Table 214: Example: sRN DeviceIdent

CoLa A	<STX>sRN{SPC}DeviceIdent<ETX>
	<STX>sRN DeviceIdent<ETX>
	sRN DeviceIdent
	02 73 52 4E 20 44 65 76 69 63 65 49 64 65 6E 74 03
CoLa B	02 02 02 02 00 00 00 0F 73 52 4E 20 44 65 76 69 63 65 49 64 65 6E 74 25
	73 52 4E 20 44 65 76 69 63 65 49 64 65 6E 74

Table 215: Telegram structure: sRA DeviceIdent

Telegram structure: sRA DeviceIdent						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command		String	11		DeviceIdent	44 65 76 69 63 65 49 64 65 6E 74
Value	Length of firmware designation	Enum_16	2		0 ... 22h	0 ... 22h
Value	Firmware designation for device family	String			(See example)	(See example)
Value	Length of firmware version	Enum_16	2		0 ... 22h	0 ... 22h
Value	Firmware version	String			(See example)	(See example)

Table 216: Example: sRA DeviceIdent

CoLa A	<STX>sRA{SPC}DeviceIdent{SPC}9{SPC}multiScan{SPC}8{SPC}2.1.0.2B<ETX>
CoLa B	02 02 02 02 00 00 00 25 73 52 41 20 44 65 76 69 63 65 49 64 65 6E 74 20 00 09 6D 75 6C 74 69 53 63 61 6E 00 08 32 2E 31 2E 30 2E 32 42 30

12.4.1.4.7.2 Read version of the application software [sRN FirmwareVersion]

Table 217: Telegram structure: sRN FirmwareVersion

Telegram structure: sRN FirmwareVersion						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Read version of the application software	String	15		FirmwareVersion	46 69 72 6D 77 61 72 65 56 65 72 73 69 6F 6E

Table 218: Example: sRN FirmwareVersion

CoLa A	<STX>sRN{SPC}FirmwareVersion<ETX>
	<STX>sRN FirmwareVersion<ETX>
	sRN FirmwareVersion
	02 73 52 4E 20 46 69 72 6D 77 61 72 65 56 65 72 73 69 6F 6E 03

CoLa B	02 02 02 02 00 00 00 13 73 52 4E 20 46 69 72 6D 77 61 72 65 56 65 72 73 69 6F 6E 24
	73 52 4E 20 46 69 72 6D 77 61 72 65 56 65 72 73 69 6F 6E

Table 219: Telegram structure: sRA FirmwareVersion

Telegram structure: sRA FirmwareVersion						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Read version of the application software	String	15		FirmwareVersion	46 69 72 6D 77 61 72 65 56 65 72 73 69 6F 6E
Value	Length of version	Uint_16	2		0 ... 28h	0 ... 28h
Value	Version	String	16		(See example)	(See example)

Table 220: Example: sRA FirmwareVersion

CoLa A	<STX>sRA{SPC}FirmwareVersion{SPC}14{SPC}1.2.0-b.0+1225.523ef<ETX>
	02 73 52 41 20 46 69 72 6D 77 61 72 65 56 65 72 73 69 6F 6E 20 31 34 20 31 2E 32 2E 30 2D 62 2E 30 2B 31 32 32 35 2E 35 32 33 65 66 03
CoLa B	02 02 02 02 00 00 00 2A 73 52 41 20 46 69 72 6D 77 61 72 65 56 65 72 73 69 6F 6E 20 00 14 31 2E 32 2E 30 2D 62 2E 30 2B 31 32 32 35 2E 35 32 33 65 66 4B

12.4.1.4.7.3 Read the device state [sRN SCdevicestate]

This telegram reads the general device state.

Table 221: Telegram structure: sRN SCdevicestate

Telegram structure: sRN SCdevicestate						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Read the device state	String	13		SCdevicestate	53 43 64 65 76 69 63 65 73 74 61 74 65

Table 222: Example: sRN SCdevicestate

CoLa A	<STX>sRN{SPC}SCdevicestate<ETX>
	<STX>sRN SCdevicestate<ETX>
	sRN SCdevicestate
	02 73 52 4E 20 53 43 64 65 76 69 63 65 73 74 61 74 65 03
CoLa B	02 02 02 02 00 00 00 11 73 52 4E 20 53 43 64 65 76 69 63 65 73 74 61 74 65 30
	73 52 4E 20 53 43 64 65 76 69 63 65 73 74 61 74 65

Table 223: Telegram structure: sRA SCdevicestate

Telegram structure: sRA SCdevicestate						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Read the device state	String	13		SCdevicestate	53 43 64 65 76 69 63 65 73 74 61 74 65
Status code	Code number	Enum_8	1	Busy / logged-in: Ready: Error:	0 1 2	00 01 02

Table 224: Example: sRA SCdevicestate

CoLa A	<STX>sRA{SPC}SCdevicestate{SPC}1<ETX>
	02 73 52 41 20 53 43 64 65 76 69 63 65 73 74 61 74 65 20 31 03
CoLa B	02 02 02 02 00 00 00 13 73 52 41 20 53 43 64 65 76 69 63 65 73 74 61 74 65 20 01 1E

12.4.1.4.7.4 Read device order number [sRN OrdNum]

This telegram reads the device order number which corresponds to the SICK part number of the device.

Table 225: Telegram structure: sRN OrdNum

Telegram structure: sRN OrdNum						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Read device order number	String	6		OrdNum	4F 72 64 4E 75 6D

Table 226: Example: sRN OrdNum

CoLa A	<STX>sRN{SPC}OrdNum<ETX>	
	<STX>sRN OrdNum<ETX>	
	sRN OrdNum	
	02 73 52 4E 20 4F 72 64 4E 75 6D 03	
CoLa B	02 02 02 02 00 00 00 0A 73 52 4E 20 4F 72 64 4E 75 6D 40	
	73 52 4E 20 4F 72 64 4E 75 6D	

Table 227: Telegram structure: sRA OrdNum

Telegram structure: sRA OrdNum						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Read device order number	String	6		OrdNum	4F 72 64 4E 75 6D

Telegram structure: sRA OrdNum						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Length	Number of characters of the following order number	Uint_16	2		0h ... 20h	00 00 ... 00 20
Order number	Order number in 7 digits	String	7		0000000 ... 9999999	00 00 00 00 00 00 00 ... FF FF FF FF FF FF FF

Example: sRA OrdNum 1134610 (Order Number for picoScan150 Pro-1)

Table 228: Example for picoScan150 Pro-1: sRA OrdNum

CoLa A	<STX>sRA{SPC}OrdNum{SPC}7{SPC}1134610<ETX>
	02 73 52 41 20 4F 72 64 4E 75 6D 20 37 20 31 31 33 34 36 31 30 03
CoLa B	02 02 02 02 00 00 00 14 73 52 41 20 4F 72 64 4E 75 6D 20 00 07 31 31 33 34 36 31 30 58

12.4.1.4.7.5 Read serial number [sRN SerialNumber]

Table 229: Telegram structure: sRN SerialNumber

Telegram structure: sRN SerialNumber						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Read serial number of the device	String	12		SerialNumber	53 65 72 69 61 6C 4E 75 6D 62 65 72

Table 230: Example: sRN SerialNumber

CoLa A	<STX>sRN{SPC}SerialNumber<ETX>
	<STX>sRN SerialNumber<ETX>
	sRN SerialNumber
	02 73 52 4E 20 53 65 72 69 61 6C 4E 75 6D 62 65 72 03
CoLa B	02 02 02 02 00 00 00 10 73 52 4E 20 53 65 72 69 61 6C 4E 75 6D 62 65 72 4C
	73 52 4E 20 53 65 72 69 61 6C 4E 75 6D 62 65 72

Table 231: Telegram structure: sRA SerialNumber

Telegram structure: sRA SerialNumber						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Read serial number of the device	String	12		SerialNumber	53 65 72 69 61 6C 4E 75 6D 62 65 72

Telegram structure: sRA SerialNumber						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Length of serial number	Number of characters of the serial number	Uint_16	2		0 ... 8h	00 00 ... 00 08h
Serial number	Production period (year, calendar week, number): YYWWxxxx	String	8		(See example)	(See example)

Table 232: Example: sRA SerialNumber

CoLa A	<STX>sRA{SPC}SerialNumber{SPC}8{SPC}23360024<ETX>
	02 73 52 41 20 53 65 72 69 61 6C 4E 75 6D 62 65 72 20 38 20 32 33 33 36 30 30 32 34 03
CoLa B	02 02 02 02 00 00 00 1B 73 52 41 20 53 65 72 69 61 6C 4E 75 6D 62 65 72 20 00 08 32 33 33 36 30 30 32 34 69

12.4.1.4.7.6 Read device type [sRN DItype]

This telegram asks for the device type.

Table 233: Telegram structure: sRN DItype

Telegram structure: sRN DItype						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Ask state	String	6		DItype	44 49 74 79 70 65

Table 234: Example: sRN DItype

CoLa A	<STX>sRN{SPC}DItype<ETX>	
	<STX>sRN DItype<ETX>	
	sRN DItype	
	02 73 52 4E 20 44 49 74 79 70 65 03	
CoLa B	02 02 02 02 00 00 00 0A 73 52 4E 20 44 49 74 79 70 65 5A	
	73 52 4E 20 44 49 74 79 70 65	

Table 235: Telegram structure: sRA DItype

Telegram structure: sRA DItype						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Ask state	String	6		DItype	44 49 74 79 70 65
Length of type key	Number of digits of the following type code length	Uint_8	1		0d ... 255d (0h ... FF)	00 ... FF

Telegram structure: sRA DItype						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Device type	Type code of the device	String	(var.)		(Device type)	(Device type)

Table 236: sRA DItype Example for multiScan136:

CoLa A	<STX>sRA{SPC}DItype{SPC}C{SPC}multiScan136<ETX>
	02 73 52 41 20 44 49 74 79 70 65 20 43 20 6D 75 6C 74 69 53 63 61 6E 31 33 36 03
CoLa B	02 02 02 02 00 00 00 19 73 52 41 20 44 49 74 79 70 65 20 00 0C 6D 75 6C 74 69 53 63 61 6E 31 33 36 1B

12.4.1.4.7.7 Read operating hours [sRN ODoprh]

Views the total number of operating hours during the lifetime of the device.

Table 237: Telegram structure: sRN ODoprh

Telegram structure: sRN ODoprh						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Read operating hours	String	6		ODoprh	4F 44 6F 70 72 68

Table 238: Example: sRN ODoprh

CoLa A	<STX>sRN{SPC}ODoprh<ETX>	
	<STX>sRN ODoprh<ETX>	
	sRN ODoprh	
	02 73 52 4E 20 4F 44 6F 70 72 68 03	
CoLa B	02 02 02 02 00 00 00 0A 73 52 4E 20 4F 44 6F 70 72 68 41	
	73 52 4E 20 4F 44 6F 70 72 68	

Table 239: Telegram structure: sRA ODoprh

Telegram structure: sRA ODoprh						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Read operating hours	String	6		ODoprh	4F 44 6F 70 72 68
Value	Operating hours in 1/10 h	Uint_32	4		0h … FFFFFFFFh	00 00 00 00 … FF FF FF FF

Table 240: Example: sRA ODoprh

CoLa A	<STX>sRA{SPC}ODoprh{SPC}1B50B<ETX>
	02 73 52 41 20 4F 44 6F 70 72 68 20 31 42 35 30 42 03
CoLa B	02 02 02 02 00 00 00 0F 73 52 41 20 4F 44 6F 70 72 68 20 00 01 B5 0B D1

Calculation of the value: 1B50B (hex) → 111883 (dez) × 1/10 h = 11188.3 h

12.4.1.4.7.8 Read operating hours since last power on [sRN ODopdaily]

Views the runtime duration since the last power on of the device.

Table 241: Telegram structure: sRN ODopdaily

Telegram structure: sRN ODopdaily						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Read operating hours	String	9		ODopdaily	4F 44 6F 70 64 61 69 6C 79

Table 242: Example: sRN ODopdaily

CoLa A	<STX>sRN{SPC}ODopdaily<ETX>	
	<STX>sRN ODopdaily<ETX>	
	sRN ODopdaily	
	02 73 52 4E 20 4F 44 6F 70 64 61 69 6C 79 03	
CoLa B	02 02 02 02 00 00 00 0D 73 52 4E 20 4F 44 6F 70 64 61 69 6C 79 22	
	73 52 4E 20 4F 44 6F 70 64 61 69 6C 79	

Table 243: Telegram structure: sRA ODopdaily

Telegram structure: sRA ODopdaily						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Read operating hours since last power on	String	9		ODopdaily	4F 44 6F 70 64 61 69 6C 79
Value	Operating hours in 1/10 h	Uint_32	4		0h … FFFFFFFFh	00 00 00 00 … FF FF FF FF

Table 244: Example: sRA ODopdaily

CoLa A	<STX>sRA{SPC}ODopdaily{SPC}424772B8<ETX>
	02 73 52 41 20 4F 44 6F 70 72 68 20 34 32 34 37 37 32 42 38 03
CoLa B	02 02 02 02 00 00 00 12 73 52 41 20 4F 44 6F 70 72 68 20 42 47 72 B8 D7

12.4.1.4.7.9 Read power on counter [sRN ODpwrc]

Table 245: Telegram structure: sRN ODpwrc

Telegram structure: sRN ODpwrc						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Read power on counter	String	6		ODpwrc	4F 44 70 77 72 63

Table 246: Example: sRN ODpwrc

CoLa A	<STX>sRN{SPC}ODpwrc<ETX>
	<STX>sRN ODpwrc<ETX>
	sRN ODpwrc
	02 73 52 4E 20 4F 44 70 77 72 63 03
CoLa B	02 02 02 02 00 00 00 0A 73 52 4E 20 4F 44 70 77 72 63 52
	73 52 4E 20 4F 44 70 77 72 63

Table 247: Telegram structure: sRA ODpwrc

Telegram structure: sRA ODpwrc						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Read power on counter	String	6		ODpwrc	4F 44 70 77 72 63
Value	Power on counter	Uint_32	4		0h ... FFFFFFFFh	00 00 00 00 ... FF FF FF FF

Table 248: Example: sRA ODpwrc

CoLa A	<STX>sRA{SPC}ODpwrc{SPC}9A<ETX>
	02 73 52 41 20 4F 44 70 77 72 63 20 39 41 03
CoLa B	02 02 02 02 00 00 00 0F 73 52 41 20 4F 44 70 77 72 63 20 00 00 00 9A E7

12.4.1.4.7.10 Read temperature [sRN OPcurtmpdev]

With this command the internal temperature of the device can be identified. Please note that it does not give an indication of the current ambient temperature.

Table 249: Telegram structure: sRN OPcurtmpdev

Telegram structure: sRN OPcurtmpdev						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Read temperature of the device	String	11		OPcurtmpdev	4F 50 63 75 72 74 6D 70 64 65 76

Table 250: Example: sRN OPcurtmpdev

CoLa A	<STX>sRN{SPC}OPcurtmpdev<ETX>
	<STX>sRN OPcurtmpdev<ETX>
	sRN OPcurtmpdev
	02 73 52 4E 20 4F 50 63 75 72 74 6D 70 64 65 76 03
CoLa B	02 02 02 02 00 00 00 0F 73 52 4E 20 4F 50 63 75 72 74 6D 70 64 65 76 2A
	73 52 4E 20 4F 50 63 75 72 74 6D 70 64 65 76

Table 251: Telegram structure: sRA OPcurtmpdev

Telegram structure: sRA OPcurtmpdev						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Read temperature of the device	String	11		OPcurtmpdev	4F 50 63 75 72 74 6D 70 64 65 76
Tempera-ture data	[°C]	Real as float according to IEEE754	4	(-50 °C ... +100 °C)	C2480000h ... 42C80000h	C2 48 00 00 ... 42 C8 00 00

Example: sRA OPcurtmpdev (35 °C)

The result is float and IEEE-754 coded

Table 252: Example: sRA OPcurtmpdev

CoLa A	<STX>sRA{SPC}OPcurtmpdev{SPC}420C0000<ETX>
	02 73 52 41 20 4F 50 63 75 72 74 6D 70 64 65 76 20 34 32 30 43 30 30 30 30 03
CoLa B	02 02 02 02 00 00 00 14 73 52 41 20 4F 50 63 75 72 74 6D 70 64 65 76 20 42 0C 00 00 4B

12.4.1.4.7.11 Set device name [sWN LocationName]

Table 253: Telegram structure: sWN LocationName

Telegram structure: sWN LocationName (User level 'Maintenance' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Set device name	String	12		LocationName	4C 6F 63 61 74 69 6F 6E 4E 61 6D 65
Value	Number of characters of the following device name	Uint_16	2		0d ... +16d (0h ... 10h)	00 00 ... 00 10
Value	Device name	String	16		[Device name]	[Device name]

Table 254: Example: sWN LocationName +9 LongRange

CoLa A	<STX>sWN{SPC}LocationName{SPC}+9{SPC}LongRange<ETX>	
	<STX>sWN LocationName +9 LongRange<ETX>	
	sWN LocationName +9 LongRange	
	02 73 57 4E 20 4C 6F 63 61 74 69 6F 6E 4E 61 6D 65 20 39 20 4C 6F 6E 67 52 61 6E 67 65 03	
CoLa B	02 02 02 02 00 00 00 1D 73 57 4E 20 4C 6F 63 61 74 69 6F 6E 4E 61 6D 65 20 00 09 4C 6F 6E 67 52 61 6E 67 65 2C	
	73 57 4E 20 4C 6F 63 61 74 69 6F 6E 4E 61 6D 65 20 00 09 4C 6F 6E 67 52 61 6E 67 65	

Table 255: Telegram structure: sWA LocationName

Telegram structure: sWA LocationName						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Set device name	String	12		LocationName	4C 6F 63 61 74 69 6F 6E 4E 61 6D 65

Table 256: Example: sWA LocationName

CoLa A	<STX>sWA{SPC}LocationName<ETX>
	02 73 57 41 20 4C 6F 63 61 74 69 6F 6E 4E 61 6D 65 03
CoLa B	02 02 02 02 00 00 00 11 73 57 41 20 4C 6F 63 61 74 69 6F 6E 4E 61 6D 65 20 7F

12.4.1.4.7.12 Read device name [sRN LocationName]

Table 257: Telegram structure: sRN LocationName

Telegram structure: sRN LocationName						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Read device name	String	12		LocationName	4C 6F 63 61 74 69 6F 6E 4E 61 6D 65

Table 258: Example: sRN LocationName

CoLa A	<STX>sRN{SPC}LocationName<ETX>	
	<STX>sRN LocationName<ETX>	
	sRN LocationName	
	02 73 52 4E 20 4C 6F 63 61 74 69 6F 6E 4E 61 6D 65 03	
CoLa B	02 02 02 02 00 00 00 10 73 52 4E 20 4C 6F 63 61 74 69 6F 6E 4E 61 6D 65 55	
	73 52 4E 20 4C 6F 63 61 74 69 6F 6E 4E 61 6D 65	

Table 259: Telegram structure: sRA LocationName

Telegram structure: sRA LocationName						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Find complete telegram structure of the answer in see table 253, page 139						

12.4.1.4.7.13 Reset output counter [sMN LIDrstoutpcnt]

Table 260: Telegram structure: sMN LIDrstoutpcnt

Telegram structure: sMN LIDrstoutpcnt (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E
Command	Reset output counter	String	13		LIDrstoutpcnt	4C 49 44 72 73 74 6F 75 74 70 63 6E 74

Table 261: Example: sMN LIDrstoutpcnt

CoLa A	<STX>sMN{SPC}LIDrstoutpcnt<ETX>	
	<STX>sMN LIDrstoutpcnt<ETX>	
	sMN LIDrstoutpcnt	
	02 73 4D 4E 20 4C 49 44 72 73 74 6F 75 74 70 63 6E 74 03	
CoLa B	02 02 02 02 00 00 00 11 73 4D 4E 20 4C 49 44 72 73 74 6F 75 74 70 63 6E 74 03	
	73 4D 4E 20 4C 49 44 72 73 74 6F 75 74 70 63 6E 74	

Table 262: Telegram structure: sAN LIDrstoutpcnt

Telegram structure: sAN LIDrstoutpcnt						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 41 4E
Command	Reset output counter	String	13		LIDrstoutpcnt	4C 49 44 72 73 74 6F 75 74 70 63 6E 74
Status code	Code number	Bool_1	1	Success: Error:	0 1	00 01

Table 263: Example: sAN LIDrstoutpcnt

CoLa A	<STX>sAN{SPC}LIDrstoutpcnt{SPC}0<ETX>
	02 73 41 4E 20 4C 49 44 72 73 74 6F 75 74 70 63 6E 74 20 30 03
CoLa B	02 02 02 02 00 00 00 13 73 41 4E 20 4C 49 44 72 73 74 6F 75 74 70 63 6E 74 20 00 2F

12.4.1.4.7.14 Initiate an acoustic or visual signal for a defined period of time [sMN FindMe]

This command can be used to make the device easier to find.

Table 264: Telegram structure: sMN FindMe

Telegram structure: sMN FindMe (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E
Command	Initiate an acoustic or visual signal	String	11		FindMe	46 69 6E 64 4D 65

Telegram structure: sMN FindMe (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Period of time	Duration in seconds	Uint_16	2		0d ... 65535d (0h .. FF FF)	00 ... FF FF

Table 265: Example: sMN FindMe

CoLa A	<STX>sMN{SPC}FindMe{SPC}1<ETX>	
	<STX>sMN FindMe 1<ETX>	
	sMN FindMe 1	
	02 73 4D 4E 20 46 69 6E 64 4D 65 20 31 03	
CoLa B	02 02 02 02 00 00 00 0D 73 4D 4E 20 46 69 6E 64 4D 65 20 01 7C	
	73 4D 4E 20 46 69 6E 64 4D 65 20 01	

Table 266: Telegram structure: sAN FindMe

Telegram structure: sAN FindMe						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 41 4E
Command	Initiate an acoustic or visual signal	String	11		FindMe	46 69 6E 64 4D 65

Table 267: Example: sAN FindMe

CoLa A	<STX>sAN{SPC}Findme<ETX>
	02 73 41 4E 20 46 69 6E 64 4D 65 03
CoLa B	02 02 02 02 00 00 00 0B 73 41 4E 20 46 69 6E 64 4D 65 20 71

12.4.1.4.7.15 Read date of last permanent save [sRN DIpara]

This command reads the date at which the last permanent save (see "Save parameters permanently [sMN mEEwriteall]", page 90) was executed.

Table 268: Telegram structure: sRN DIpara

Telegram structure: sRN DIpara						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Read date of last permanent save	String	6		DIpara	44 49 70 61 72 61

Table 269: Example: sRN DIpara

CoLa A	<STX>sRN{SPC}DIpara<ETX>	
	<STX>sRN DIpara<ETX>	
	sRN DIpara	
	02 73 52 4E 20 44 49 70 61 72 61 03	

CoLa B	02 02 02 02 00 00 00 4E 73 52 4E 20 44 49 70 61 72 61 40
	73 52 4E 20 44 49 70 61 72 61

Table 270: Telegram structure: sRA DIpara

Telegram structure: sRA DIpara						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Read date of last permanent save	String	6		DIpara	44 49 70 61 72 61
Value	Number of characters of the following date	Uint_16	2		0d ... 10d (0h ... 0Ah)	0 ... 0A
Date of last permanent save	DD.MM.YYYY	FlexString	10		(see example)	(see example)

Table 271: Example: sRA DIpara

CoLa A	<STX>sRA{SPC}DIpara{SPC}A{SPC}09.01.2024<ETX>
	02 73 52 41 20 44 49 70 61 72 61 20 41 20 30 39 2E 30 31 2E 32 30 32 34 03
CoLa B	02 02 02 02 00 00 00 17 73 52 41 20 44 49 70 61 72 61 20 00 0A 30 39 2E 30 31 2E 32 30 32 34 69

12.4.1.4.7.16 Read time of last permanent save [sRN DIparatm]

This command reads the time at which the last permanent save was executed.

Table 272: Telegram structure: sRN DIparatm

Telegram structure: sRN DIparatm						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Read time of last permanent save	String	8		DIparatm	44 49 70 61 72 61 74 6D

Table 273: Example: sRN DIparatm

CoLa A	<STX>sRN{SPC}DIparatm<ETX>
	<STX>sRN DIparatm<ETX>
	sRN DIparatm
	02 73 52 4E 20 44 49 70 61 72 61 74 6D 03
CoLa B	02 02 02 02 00 00 00 4E 73 52 4E 20 44 49 70 61 72 61 74 6D 59
	73 52 4E 20 44 49 70 61 72 61 74 6D

Table 274: Telegram structure: sRA DIparatm

Telegram structure: sRA DIparatm						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Read time of last permanent save	String	8		DIparatm	44 49 70 61 72 61 74 6D
Value	Number of characters of the following time	Uint_16	2		0d ... 5d (0h ... 5h)	00 00 ... 00 05
Time of last permanent save	-	FlexString	5		(see example)	(see example)

Table 275: Example: sRA DIparatm

CoLa A	<STX>sRA{SPC}DIparatm{SPC}5{SPC}12:28<ETX>
	02 73 52 41 20 44 49 70 61 72 61 74 6D 20 35 20 31 32 3A 32 38 03
CoLa B	02 02 02 02 00 00 00 14 73 52 41 20 44 49 70 61 72 61 74 6D 20 00 05 31 32 3A 32 38 40

12.4.1.4.8 Interfaces

12.4.1.4.8.1 Set IP address [sWN EIIpAddr]

NOTE

- Save permanently to set values. Changes will be active after rebooting the device.
- Settings must correspond with network in which scanner is used. Else device cannot be found any more.

Table 276: Telegram structure: sWN EIIpAddr

Telegram structure: sWN EIIpAddr (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Set IP address	String	8		EIIpAddr	45 49 49 50 41 64 64 72
IP address	Set values	Uint_8	1	First part of IP adress	0 ...+255d (00 ... FF)	00 ... FF
				Second part of IP adress	0 ...+255d (00 ... FF)	00 ... FF
				Third part of IP adress	0 ...+255d (00 ... FF)	00 ... FF
				Fourth part of IP adress	0 ...+255d (00 ... FF)	00 ... FF

Table 277: Example: sWN EIIpAddr 192.168.0.2

CoLa A	<STX>sWN{SPC}EIIpAddr{SPC}C0{SPC}A8{SPC}0{SPC}2<ETX>
	<STX>sWN EIIpAddr C0 A8 0 2<ETX>
	sWN EIIpAddr C0 A8 0 2
	02 73 57 4E 20 45 49 49 70 41 64 64 72 20 43 30 20 41 38 20 30 20 32 03
CoLa B	02 02 02 02 00 00 00 11 73 57 4E 20 45 49 49 70 41 64 64 72 20 C0 A8 00 02 06
	73 57 4E 20 45 49 49 70 41 64 64 72 20 C0 A8 00 02

Table 278: Telegram structure: sWA EIIpAddr

Telegram structure: sWA EIIpAddr						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Set IP address	String	8		EIIpAddr	45 49 49 50 41 64 64 72

Table 279: Example: sWA EIIpAddr

CoLa A	<STX>sWA{SPC}EIIpAddr<ETX>
	02 73 57 41 20 45 49 49 70 41 64 64 72 03
CoLa B	02 02 02 02 00 00 00 0D 73 57 41 20 45 49 49 70 41 64 64 72 20 63

12.4.1.4.8.2 Read IP address [sRN EIIpAddr]

Table 280: Telegram structure: sRN EIIpAddr

Telegram structure: sRN EIIpAddr						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Read IP address	String	8		EIIpAddr	45 49 49 50 41 64 64 72

Table 281: Example: sRN EIIpAddr

CoLa A	<STX>sRN{SPC}EIIpAddr<ETX>
	<STX>sRN EIIpAddr<ETX>
	sRN EIIpAddr
	02 73 52 4E 20 45 49 49 70 41 64 64 72 03
CoLa B	02 02 02 02 00 00 00 0C 73 52 4E 20 45 49 49 70 41 64 64 72 49
	73 52 4E 20 45 49 49 70 41 64 64 72

Table 282: Telegram structure: sRA EIIpAddr

Telegram structure: sRA EIIpAddr						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Read IP address	String	8		EIIpAddr	45 49 49 50 41 64 64 72
IP address	Default: 192.168.0.1	Uint_8	1	First part of IP address	0 ...+255d (00 ... FF)	00 ... FF
				Second part of IP address	0 ...+255d (00 ... FF)	00 ... FF
				Third part of IP address	0 ...+255d (00 ... FF)	00 ... FF
				Fourth part of IP address	0 ...+255d (00 ... FF)	00 ... FF

Table 283: Example: sRA EIIpAddr 192.168.0.2

CoLa A	<STX>sRA{SPC}EIIpAddr{SPC}C0{SPC}A8{SPC}00{SPC}02<ETX>
	02 73 57 41 20 45 49 49 70 41 64 64 72 20 C0 20 A8 20 00 20 02 03
CoLa B	02 02 02 02 00 00 00 11 73 52 41 20 45 49 49 70 41 64 64 72 20 C0 A8 00 02 0C

12.4.1.4.8.3 Read IP address assigned by DHCP [sRN EIIpAddrDHCP]

NOTE
DHCP needs to be set as mode for ethernet assignment.

Table 284: Telegram structure: sRN EIIpAddrDHCP

Telegram structure: sRN EIIpAddrDHCP						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Read IP address assigned by DHCP	String	12		EIIpAddrDHCP	45 49 49 70 41 64 64 72 44 48 43 50

Table 285: Example: srN EIIpAddrDHCP

CoLa A	<STX>sRN{SPC}EIIpAddrDHCP<ETX>	
	<STX>sRN EIIpAddrDHCP<ETX>	
	sRN EIIpAddrDHCP	
	02 73 57 4E 20 45 49 49 70 41 64 64 72 44 48 43 50 03	
CoLa B	02 02 02 02 00 00 00 10 73 52 4E 20 45 49 49 70 41 64 64 72 44 48 43 50 56	
	73 52 4E 20 45 49 49 70 41 64 64 72 44 48 43 50	

Table 286: Telegram structure: sRA EIIpAddrDHCP

Telegram structure: sRA EIIpAddrDHCP						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Read IP address assigned by DHCP	String	12		EIIpAddrDHCP	45 49 49 70 41 64 64 72 44 48 43 50
IP address	Default: 192.168.0.1	Uint_8	1	First part of IP address	0 ...+255d (00 ... FF)	00 ... FF
				Second part of IP address	0 ...+255d (00 ... FF)	00 ... FF
				Third part of IP address	0 ...+255d (00 ... FF)	00 ... FF
				Fourth part of IP address	0 ...+255d (00 ... FF)	00 ... FF

Table 287: Example: sRA EIIpAddrDHCP 192.168.0.1

CoLa A	<STX>sRA{SPC}EIIpAddrDHCP{SPC}C0{SPC}A8{SPC}0{SPC}1<ETX>
	02 73 52 41 20 45 49 49 70 41 64 64 72 44 48 43 50 20 43 30 20 41 38 20 30 20 31 03
CoLa B	02 02 02 02 00 00 00 15 73 52 41 20 45 49 49 70 41 64 64 72 44 48 43 50 20 C0 A8 00 01 10

12.4.1.4.8.4 Set mode for ethernet adress assignment [sWN EIAddrMode]

This Command determins the mode for the ethernet adress assignment.

Table 288: Telegram structure: sWN EIAddrMode

Telegram structure: sWN EIAddrMode (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Set mode for ethernet adress assignment	String	10		EIAddrMode	45 49 41 64 64 72 4D 6F 64 65
Ethernet adress assign-ment	Static IP adress / DHCP	Enum_8	1	Static: DHCP:	0 1	00 01

Table 289: Example: sWN EIAddrMode

CoLa A	<STX>sWN{SPC}EIAddrMode{SPC}1<ETX>	
	<STX>sWN EIAddrMode 1<ETX>	
	sWN EIAddrMode 1	
	02 73 57 4E 20 45 49 41 64 64 72 4D 6F 64 65 20 31 03	
CoLa B	02 02 02 02 00 00 00 4E 73 57 4E 20 45 49 41 64 64 72 4D 6F 64 65 20 01 76	
	73 57 4E 20 45 49 41 64 64 72 4D 6F 64 65 20 01	

Table 290: Telegram structure: sWA EIAddrMode

Telegram structure: sWA EIAddrMode						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Set mode for ethernet adress assignment	String	10		EIAddrMode	45 49 41 64 64 72 4D 6F 64 65

Table 291: Example: sWA EIAddrMode

CoLa A	<STX>sWA{SPC}EIAddrMode<ETX>
	02 73 57 41 20 45 49 41 64 64 72 4D 6F 64 65 03
CoLa B	02 02 02 02 00 00 00 0F 73 57 41 20 45 49 41 64 64 72 4D 6F 64 65 20 79

12.4.1.4.8.5 Set fallback for DHCP [sWN EIDHCPFallback]

This Command determins the fallback when DHCP is not successful.

Table 292: Telegram structure: sWN EIDHCPFallback

Telegram structure: sWN EIDHCPFallback (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Set fallback for DHCP	String	14		EIDHCPFallback	45 49 44 48 43 50 46 61 6C 6C 62 61 63 6B
Fallback ethernet adress assign-ment	Use Static IP adress / Retry DHCP	Enum_8	1	Static IP adress: DHCP retry:	0 1	00 01

Table 293: Example: sWN EIDHCPFallback

CoLa A	<STX>sWN{SPC}EIDHCPFallback{SPC}1<ETX>	
	<STX>sWN EIDHCPFallback 1<ETX>	
	sWN EIDHCPFallback 1	
	02 73 57 4E 20 45 49 44 48 43 50 46 61 6C 6C 62 61 63 6B 20 31 03	
CoLa B	02 02 02 02 00 00 00 14 73 57 4E 20 45 49 44 48 43 50 46 61 6C 6C 62 61 63 6B 20 01 54	
	73 57 4E 20 45 49 44 48 43 50 46 61 6C 6C 62 61 63 6B 20 01	

Table 294: Telegram structure: sWA EIDHCPFallback

Telegram structure: sWA EIDHCPFallback						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41

Telegram structure: sWA EIDHCPFallback						
Telegram part	Description	Variable	Length	Additional details	Values CoLa A (ASCII)	Values CoLa B (Binary)
Command	Set fallback for DHCP	String	14		EIDHCPFallback	45 49 44 48 43 50 46 61 6C 6C 62 61 63 6B

Table 295: Example: sWA EIDHCPFallback

CoLa A	<STX>sWA{SPC}EIDHCPFallback<ETX>
	02 73 57 41 20 45 49 44 48 43 50 46 61 6C 6C 62 61 63 6B 03
CoLa B	02 02 02 02 00 00 00 13 73 57 41 20 45 49 44 48 43 50 46 61 6C 6C 62 61 63 6B 20 5A

12.4.1.4.8.6 Set Ethernet gateway [sWN EIgate]

Change Ethernet gateway IP address (TCP/IP)

NOTE

- Save permanently to set values. Changes will be active after rebooting the device.
- Settings must correspond with network in which scanner is used. Else device cannot be found any more.

Table 296: Telegram structure: sWN EIgate

Telegram structure: sWN EIgate (User level 'Authorized client' required)						
Telegram part	Description	Variable	Length	Additional details	Values CoLa A (ASCII)	Values CoLa B (Binary)
Command type	Write	String	3		sWN	73 57 4E
Command	Set gateway adress	String	6		EIgate	45 49 67 61 74 65
Gateway address	Set values	Uint_8	1	First part of gateway address	0 ...+255d (00...FF)	00 ...FF
				Second part of gateway address	0 ...+255d (00...FF)	00 ...FF
				Third part of gateway address	0 ...+255d (00...FF)	00 ...FF
				Fourth part of gateway address	0 ...+255d (00...FF)	00 ...FF

Table 297: Example: sWN EIgate 192.168.0.1

CoLa A	<STX>sWN{SPC}EIgate{SPC}C0{SPC}A8{SPC}00{SPC}01<ETX>
	<STX>sWN EIgate C0 A8 00 01<ETX>
	sWN EIgate C0 A8 00 01
	02 73 57 4E 20 45 49 67 61 74 65 20 43 30 20 41 38 20 30 30 20 30 31 03
CoLa B	02 02 02 02 00 00 00 0F 73 57 4E 20 45 49 67 61 74 65 20 C0 A8 00 01 18
	73 57 4E 20 45 49 67 61 74 65 20 C0 A8 00 01

Table 298: Telegram structure: sWA Elgate

Telegram structure: sWA Elgate						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Set gateway adress	String	6		Elgate	45 49 67 61 74 65

Table 299: Example: sWA Elgate

CoLa A	<STX>sWA{SPC}Elgate<ETX>
	02 73 57 41 20 45 49 67 61 74 65 03
CoLa B	02 02 02 02 00 00 00 0B 73 57 41 20 45 49 67 61 74 65 20 7E

12.4.1.4.8.7 Read Ethernet gateway [sRN Elgate]

Read for the Ethernet gateway (TCP/IP)

Table 300: Telegram structure: sRN Elgate

Telegram structure: sRN Elgate						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Read gateway address	String	6		Elgate	45 49 67 61 74 65

Table 301: Example: sRN Elgate

CoLa A	<STX>sRN{SPC}Elgate<ETX>	
	<STX>sRN Elgate<ETX>	
	sRN Elgate	
	02 73 52 4E 20 45 49 67 61 74 65 03	
CoLa B	02 02 02 02 00 00 00 0A 73 52 4E 20 45 49 67 61 74 65 54	
	73 52 4E 20 45 49 67 61 74 65	

Table 302: Telegram structure: sRA Elgate

Telegram structure: sRA Elgate						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Read gateway address	String	6		Elgate	45 49 67 61 74 65

Telegram structure: sRA EIgate						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Gateway address	Default: 0.0.0.0	Uint_8	1	First part of gateway address	0 ...+255d (00...FF)	00 ... FF
				Second part of gateway address	0 ...+255d (00...FF)	00 ... FF
				Third part of gateway address	0 ...+255d (00...FF)	00 ... FF
				Fourth part of gateway address	0 ...+255d (00...FF)	00 ... FF

Table 303: Example: sRA EIgate 192.168.0.1

CoLa A	<STX>sRA{SPC}EIgate{SPC}C0{SPC}A8{SPC}00{SPC}01<ETX>
	02 73 52 41 20 45 49 67 61 74 65 20 C0 A8 00 01 03
CoLa B	02 02 02 02 00 00 00 0F 73 52 41 20 45 49 67 61 74 65 20 C0 A8 00 01 12

12.4.1.4.8.8 Read ethernet gateway IP adress assigned by DHCP [sRN EIgateDHCP]

NOTE
DHCP needs to be set as mode for ethernet assignment.

Read for the ethernet gateway IP adress which was assigned by DHCP.

Table 304: Telegram structure: sRN EIgateDHCP

Telegram structure: sRN EIgateDHCP						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Read ethernet gateway IP address assigned by DHCP	String	6		EIgateDHCP	45 49 67 61 74 65 44 48 43 50

Table 305: Example: sRN EIgate

CoLa A	<STX>sRN{SPC}EIgateDHCP<ETX>
	<STX>sRN EIgateDHCP<ETX>
	sRN EIgateDHCP
	02 73 52 4E 20 45 49 67 61 74 65 44 48 43 50 03
CoLa B	02 02 02 02 00 00 00 0E 73 52 4E 20 45 49 67 61 74 65 44 48 43 50 4B
	73 52 4E 20 45 49 67 61 74 65 44 48 43 50

Table 306: Telegram structure: sRA EIgateDHCP

Telegram structure: sRA EIgateDHCP						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41

Telegram structure: sRA EIgateDHCP						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command	Read ethernet gateway IP address assigned by DHCP	String	6		EIgateDHCP	45 49 67 61 74 65 44 48 43 50
Gateway IP address	Default: 0.0.0.0	Uint_8	1	First part of gateway IP address	0 ...+255d (00...FF)	00 ... FF
				Second part of gateway IP address	0 ...+255d (00...FF)	00 ... FF
				Third part of gateway IP address	0 ...+255d (00...FF)	00 ... FF
				Fourth part of gateway IP address	0 ...+255d (00...FF)	00 ... FF

Table 307: Example: sRA EIgateDHCP 0.0.0.0

CoLa A	<STX>sRA{SPC}EIgateDHCP{SPC}0{SPC}0{SPC}0{SPC}0<ETX>
	02 73 52 41 20 45 49 67 61 74 65 44 48 43 50 20 30 20 30 20 30 20 30 03
CoLa B	02 02 02 02 00 00 00 13 73 52 41 20 45 49 67 61 74 65 44 48 43 50 20 00 00 00 00 64

12.4.1.4.8.9 Set IP mask [sWN EImask]

NOTE

- Save permanently to set values. Changes will be active after rebooting the device.
- Settings must correspond with network in which scanner is used. Else device cannot be found any more.

Table 308: Telegram structure: sWN EImask

Telegram structure: sWN EImask (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Set IP mask	String	6		EImask	45 49 6D 61 73 6B
IP mask	Set values	Uint_8	1	First part of IP mask	0 ...+255d (00...FF)	00 ...FF
				Second part of IP mask	0 ...+255d (00...FF)	00 ...FF
				Third part of IP mask	0 ...+255d (00...FF)	00 ...FF
				Fourth part of IP mask	0 ...+255d (00...FF)	00 ...FF

Table 309: Example: sWN EImask 255.255.254.0

CoLa A	<STX>sWN{SPC}EImask{SPC}FF{SPC}FF{SPC}FE{SPC}00<ETX>	
	<STX>sWN EImask FF FF FE 00<ETX>	
	sWN EImask FF FF FE 00	
	02 73 57 4E 20 45 49 6D 61 73 6B 20 46 46 20 46 46 20 46 45 20 30 30 03	

CoLa B	02 02 02 02 00 00 00 0F 73 57 4E 20 45 49 6D 61 73 6B 20 FF FF FE 00 8C
	73 57 4E 20 45 49 6D 61 73 6B 20 FF FF FE 00

Table 310: Telegram structure: sWA EImask

Telegram structure: sWA EImask						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Set IP mask	String	6		EImask	45 49 6D 61 73 6B

Table 311: Example: sWA EImask

CoLa A	<STX>sWA{SPC}EImask<ETX>
	02 73 57 41 20 45 49 6D 61 73 6B 03
CoLa B	02 02 02 02 00 00 00 0B 73 57 41 20 45 49 6D 61 73 6B 20 7D

12.4.1.4.8.10 Read IP mask [sRN EImask]

Table 312: Telegram structure: sRN EImask

Telegram structure: sRN EImask						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Read IP mask	String	6		EImask	45 49 6D 61 73 6B

Table 313: Example: sRN EImask

CoLa A	<STX>sRN{SPC}EImask<ETX>
	<STX>sRN EImask<ETX>
	sRN EImask
	02 73 52 4E 20 45 49 6D 61 73 6B 03
CoLa B	02 02 02 02 00 00 00 0A 73 52 4E 20 45 49 6D 61 73 6B 57
	73 52 4E 20 45 49 6D 61 73 6B

Table 314: Telegram structure: sRA EImask

Telegram structure: sRA EImask						
Telegram part	**Description**	**Variable**	**Length**	**Sensor**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Read IP mask	String	6		EImask	45 49 6D 61 73 6B

Telegram structure: sRA EImask						
Telegram part	Description	Variable	Length	Sensor	Values CoLa A (ASCII)	Values CoLa B (Binary)
IP mask	Default: 255.255.255.0	Uint_8	1	First part of IP mask	0 ...+255d (00...FF)	00 ... FF
				Second part of IP mask	0 ...+255d (00...FF)	00 ... FF
				Third part of IP mask	0 ...+255d (00...FF)	00 ... FF
				Fourth part of IP mask	0 ...+255d (00...FF)	00 ... FF

Table 315: Example: sRA EImask 255.255.254.0

CoLa A	<STX>sRA{SPC}EImask{SPC}FF{SPC}FF{SPC}FE{SPC}00<ETX>
	02 73 52 41 20 45 49 6D 61 73 6B 20 45 49 6D 61 73 6B 20 45 49 6D 61 73 6B 03
CoLa B	02 02 02 02 00 00 00 0F 73 52 41 20 45 49 6D 61 73 6B 20 FF FF FE 00 86

12.4.1.4.8.11 Read IP mask assigned by DHCP [sRN EImaskDHCP]

NOTE
DHCP needs to be set as mode for ethernet assignment.

Read for the IP mask which was assigned by DHCP.

Table 316: Telegram structure: sRN EImaskDHCP

Telegram structure: sRN EImaskDHCP						
Telegram part	Description	Variable	Length	Additional details	Values CoLa A (ASCII)	Values CoLa B (Binary)
Command type	Read	String	3		sRN	73 52 4E
Command	Read IP mask assigned by DHCP	String	10		EImaskDHCP	45 49 6D 61 73 6B 44 48 43 50

Table 317: Example: sRN EImaskDHCP

CoLa A	<STX>sRN{SPC}EImaskDHCP<ETX>
	<STX>sRN EImaskDHCP<ETX>
	sRN EImaskDHCP
	02 73 52 4E 20 45 49 6D 61 73 6B 44 48 43 50 03
CoLa B	02 02 02 02 00 00 00 0E 73 52 4E 20 45 49 6D 61 73 6B 44 48 43 50 4B
	73 52 4E 20 45 49 6D 61 73 6B 44 48 43 50

Table 318: Telegram structure: sRA EImaskDHCP

Telegram structure: sRA EImaskDHCP						
Telegram part	Description	Variable	Length	Additional details	Values CoLa A (ASCII)	Values CoLa B (Binary)
Command type	Answer	String	3		sRA	73 52 41

Telegram structure: sRA EImaskDHCP						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command	Read IP mask assigned by DHCP	String	10		EImaskDHCP	45 49 6D 61 73 6B 44 48 43 50
IP mask	Default: 255.255.255.0	Uint_8	1	First part of IP mask	0 ...+255d (00...FF)	00 ... FF
				Second part of IP mask	0 ...+255d (00...FF)	00 ... FF
				Third part of IP mask	0 ...+255d (00...FF)	00 ... FF
				Fourth part of IP mask	0 ...+255d (00...FF)	00 ... FF

Table 319: Example: sRA EImaskDHCP 255.255.255.0

CoLa A	<STX>sRA{SPC}EIgateDHCP{SPC}FF{SPC}FF{SPC}FF{SPC}0<ETX>
	02 73 52 41 20 45 49 6D 61 73 6B 44 48 43 50 20 46 46 20 46 46 20 46 46 20 30 03
CoLa B	02 02 02 02 00 00 00 13 73 52 41 20 45 49 6D 61 73 6B 44 48 43 50 20 FF FF FF 00 98

12.4.1.4.8.12 Read MAC address [sRN EIMacAdr]

Table 320: Telegram structure: sRN EIMacAdr

Telegram structure: sRN EIMacAdr						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Read MAC address of the device	String	8		EIMacAdr	45 49 4D 61 63 41 64 72

Table 321: Example: sRN EIMacAdr

CoLa A	<STX>sRN{SPC}EIMacAdr<ETX>	
	<STX>sRN EIMacAdr<ETX>	
	sRN EIMacAdr	
	02 73 57 4E 20 45 49 4D 61 63 41 64 72 03	
CoLa B	02 02 02 02 00 00 00 0C 73 52 4E 20 45 49 4D 61 63 41 64 72 5B	
	73 52 4E 20 45 49 4D 61 63 41 64 72	

Table 322: Telegram structure: sRA EIMacAdr

Telegram structure: sRA EIMacAdr						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Read MAC address of the device	String	8		EIMacAdr	45 49 4D 61 63 41 64 72

Telegram structure: sRA ElMacAdr						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
MAC address	Values	Uint_8	1	First part of MAC address	0 ...+255d (00...FF)	00 ... FF
		Uint_8	1	Second part of MAC address	0 ...+255d (00...FF)	00 ... FF
		Uint_8	1	Third part of MAC address	0 ...+255d (00...FF)	00 ... FF
		Uint_8	1	Fourth part of MAC address	0 ...+255d (00...FF)	00 ... FF
		Uint_8	1	Fifth part of MAC address	0 ...+255d (00...FF)	00 ... FF
		Uint_8	1	Sixth part of MAC address	0 ...+255d (00...FF)	00 ... FF

Table 323: Example: sRA ElMacAdr 00:06:77:22:40:EA

CoLa A	<STX>sRA{SPC}ElMacAdr{SPC}0{SPC}6{SPC}77{SPC}22{SPC}40{SPC}EA<ETX>
	02 73 52 41 20 45 49 4D 61 63 41 64 72 20 30 20 36 20 37 37 20 32 32 20 34 30 20 45 41 03
CoLa B	02 02 02 02 00 00 00 13 73 52 41 20 45 49 4D 61 63 41 64 72 20 00 06 77 22 40 EA 8D

12.4.1.4.8.13 Enable/ disable CoLa Scan [sWN EnableColaScan]

NOTE

Port 30178 will not be opened.

Finding the device via device search and changing the IP adress via SICK Engineering Tools is not possible anymore.

This command enables/ disables the device search by SICK Enginnering Tools.

Table 324: Telegram structure: sWN EnableColaScan

Telegram structure: sWN EnableColaScan (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Set CoLa Scan / AutoIP	String	14		EnableColaS-can	45 6E 61 62 6C 65 43 6F 6C 61 53 63 61 6E
CoLa Scan / AutoIP	Enable/ disable	Bool_1	1	Disable: Enable:	0 1	00 01

Table 325: Example: sWN EnableColaScan

CoLa A	<STX>sWN{SPC}EnableColaScan{SPC}1<ETX>	
	<STX>sWN EnableColaScan 1<ETX>	
	sWN EnableColaScan 1	
	02 73 57 4E 20 45 6E 61 62 6C 65 43 6F 6C 61 53 63 61 6E 20 31 03	

CoLa B	02 02 02 02 00 00 00 14 73 57 4E 20 45 6E 61 62 6C 65 43 6F 6C 61 53 63 61 6E 20 01 54
	73 57 4E 20 45 6E 61 62 6C 65 43 6F 6C 61 53 63 61 6E 20 01

Table 326: Telegram structure: sWA EnableColaScan

Telegram structure: sWA EnableColaScan						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Set CoLa Scan / AutoIP	String	14		EnableColaS-can	45 6E 61 62 6C 65 43 6F 6C 61 53 63 61 6E

Table 327: Example: sWA EnableColaScan

CoLa A	<STX>sWA{SPC}EnableColaScan<ETX>
	02 73 57 41 20 45 6E 61 62 6C 65 43 6F 6C 61 53 63 61 6E 03
CoLa B	02 02 02 02 00 00 00 13 73 57 41 20 45 6E 61 62 6C 65 43 6F 6C 61 53 63 61 6E 20 5A

12.4.1.4.8.14 Enable/ disable CoLa1 interface [sWN ElAuxEnable]

After enabling the CoLa1 interface, use port 2111 for CoLa A and port 2112 for CoLa B.

Table 328: Telegram structure: sWN ElAuxEnable

Telegram structure: sWN ElAuxEnable (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Write	String	3		sWN	73 57 4E
Command	Set CoLa1 interface	String	11		ElAuxEnable	45 49 41 75 78 45 6E 61 62 6C 65
CoLa1 interface	Enable/ disable	Bool_1	1	Disable: Enable:	0 1	00 01

Table 329: Example: sWN ElAuxEnable

CoLa A	<STX>sWN{SPC}ElAuxEnable{SPC}1<ETX>
	<STX>sWN ElAuxEnable 1<ETX>
	sWN ElAuxEnable 1
	02 73 57 4E 20 45 49 41 75 78 45 6E 61 62 6C 65 20 31 03
CoLa B	02 02 02 02 00 00 00 11 73 57 4E 20 45 49 41 75 78 45 6E 61 62 6C 65 20 01 0A
	73 57 4E 20 45 49 41 75 78 45 6E 61 62 6C 65 20 01

Table 330: Telegram structure: sWA EIAuxEnable

Telegram structure: sWA EIAuxEnable						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Set CoLa1 interface	String	11		EIAuxEnable	45 49 41 75 78 45 6E 61 62 6C 65

Table 331: Example: sWA EIAuxEnable

CoLa A	<STX>sWA{SPC}EIAuxEnable<ETX>
	02 73 57 41 20 45 49 41 75 78 45 6E 61 62 6C 65 03
CoLa B	02 02 02 02 00 00 00 10 73 57 41 20 45 49 41 75 78 45 6E 61 62 6C 65 20 04

12.4.1.4.8.15 Set Webserver state [sMN SetWebserverEnabled]

This command enables/ disables the Webserver. Port 80 will not be opened after a reboot.

Table 332: Telegram structure: sMN SetWebserverEnabled

Telegram structure: sMN SetWebserverEnabled (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E
Command	Set Webserver state	String	19		SetWebserver-Enabled	53 65 74 57 65 62 73 65 72 76 65 72 45 6E 61 62 6C 65 64
State	Enable/ disable	Bool_1	1	Disable: Enable:	0 1	00 01

Table 333: Example: sMN SetWebserverEnabled

CoLa A	<STX>sMN{SPC}SetWebserverEnabled{SPC}1<ETX>	
	<STX>sMN SetWebserverEnabled 1<ETX>	
	sMN SetWebserverEnabled 1	
	02 73 4D 4E 20 47 65 74 57 65 62 73 65 72 76 65 72 45 6E 61 62 6C 65 64 20 31 03	
CoLa B	02 02 02 02 00 00 00 19 73 4D 4E 20 53 65 74 57 65 62 73 65 72 76 65 72 45 6E 61 62 6C 65 64 20 01 23	
	73 4D 4E 20 53 65 74 57 65 62 73 65 72 76 65 72 45 6E 61 62 6C 65 64 20 01	

Table 334: Telegram structure: sAN SetWebserverEnabled

Telegram structure: sAN SetWebserverEnabled						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 41 4E

Telegram structure: sAN SetWebserverEnabled						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command	Set Webserver state	String	19		SetWebserver-Enabled	53 65 74 57 65 62 73 65 72 76 65 72 45 6E 61 62 6C 65 64

Table 335: Example: sAN SetWebserverEnabled

CoLa A	<STX>sAN{SPC}SetWebserverEnabled<ETX>
	02 73 41 4E 20 47 65 74 57 65 62 73 65 72 76 65 72 45 6E 61 62 6C 65 64 03
CoLa B	02 02 02 02 00 00 00 18 73 41 4E 20 47 65 74 57 65 62 73 65 72 76 65 72 45 6E 61 62 6C 65 64 2E

12.4.1.4.8.16 Read Webserver state [sMN GetWebserverEnabled]

Returns state if Webserver is enabled.

Table 336: Telegram structure: sMN GetWebserverEnabled

Telegram structure: sMN GetWebserverEnabled (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E
Command	Webserver state	String	19		GetWebserver-Enabled	47 65 74 57 65 62 73 65 72 76 65 72 45 6E 61 62 6C 65 64

Table 337: Example: sMN GetWebserverEnabled

CoLa A	<STX>sMN{SPC}GetWebserverEnabled<ETX>	
	<STX>sMN GetWebserverEnabled<ETX>	
	sMN GetWebserverEnabled	
	02 73 4D 4E 20 47 65 74 57 65 62 73 65 72 76 65 72 45 6E 61 62 6C 65 64 03	
CoLa B	02 02 02 02 00 00 00 17 73 4D 4E 20 47 65 74 57 65 62 73 65 72 76 65 72 45 6E 61 62 6C 65 64 16	
	73 4D 4E 20 47 65 74 57 65 62 73 65 72 76 65 72 45 6E 61 62 6C 65 64	

Table 338: Telegram structure: sAN GetWebserverEnabled

Telegram structure: sAN GetWebserverEnabled						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 41 4E
Command	Webserver state	String	19		GetWebserver-Enabled	47 65 74 57 65 62 73 65 72 76 65 72 45 6E 61 62 6C 65 64

Telegram structure: sAN GetWebserverEnabled						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
State	Enabled/ disabled	Bool_1	1	Disabled: Enabled:	0 1	00 01

Table 339: Example: sAN GetWebserverEnabled

CoLa A	<STX>sAN{SPC}GetWebserverEnabled{SPC}1<ETX>
	02 73 41 4E 20 47 65 74 57 65 62 73 65 72 76 65 72 45 6E 61 62 6C 65 64 20 31 03
CoLa B	02 02 02 02 00 00 00 19 73 41 4E 20 47 65 74 57 65 62 73 65 72 76 65 72 45 6E 61 62 6C 65 64 20 01 3B

12.4.1.4.8.17 Enable/ disable LEDs [sWN LEDEnable]

This command enables/ disables the LEDs of the device.

Table 340: Telegram structure: sWN LEDEnable

Telegram structure: sWN LEDEnable (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sWN	73 57 4E
Command	Set LEDs	String	9		LEDEnable	4C 45 44 45 6E 61 62 6C 65
Status	Enable/ disable	Bool_1	1	Off: On:	0 1	00 01

Table 341: Example: sWN LEDEnable

CoLa A	<STX>sWN{SPC}LEDEnable{SPC}1<ETX>	
	<STX>sWN LEDEnable 1<ETX>	
	sWN LEDEnable 1	
	02 73 57 4E 20 4C 45 44 45 6E 61 62 6C 65 20 31 03	
CoLa B	02 02 02 02 00 00 00 0F 73 57 4E 20 4C 45 44 45 6E 61 62 6C 65 20 01 07	
	73 57 4E 20 4C 45 44 45 6E 61 62 6C 65 20 01	

Table 342: Telegram structure: sWA LEDEnable

Telegram structure: sWA LEDEnable						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sWA	73 57 41
Command	Set LEDs	String	9		LEDEnable	4C 45 44 45 6E 61 62 6C 65

Table 343: Example: sWA LEDEnable

CoLa A	<STX>sWA{SPC}LEDEnable<ETX>
	02 73 57 41 20 4C 45 44 45 6E 61 62 6C 65 03
CoLa B	02 02 02 02 00 00 00 0E 73 57 41 20 73 57 41 20 4C 45 44 45 6E 61 62 6C 65 20 09

12.4.1.4.8.18 Read state of LEDs [sRN LEDState]

Read the current state of the LEDs.

Table 344: Telegram structure: sRN LEDState

Telegram structure: sRN LEDState						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sRN	73 52 4E
Command	Read state of LEDs	String	8		LEDState	4C 45 44 53 74 61 74 65

Table 345: Example: sRN LEDState

CoLa A	<STX>sRN{SPC}LEDState<ETX>	
	<STX>sRN LEDState<ETX>	
	sRN LEDState	
	02 73 52 4E 20 4C 45 44 53 74 61 74 65 03	
CoLa B	02 02 02 02 00 00 00 0C 73 52 4E 20 4C 45 44 53 74 61 74 65 55	
	73 52 4E 20 4C 45 44 53 74 61 74 65	

Table 346: Telegram structure: sRA LEDState

Telegram structure: sRA LEDState						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	State of LEDs	String	8		LEDState	4C 45 44 53 74 61 74 65
LED Color	-	Enum_8	1	Green: Yellow: Red:	0 1 2	00 01 02
LED behavior	-	Enum_8	1	On: Off: Blinking: Blinking fast: Blinking delayed: Find me active (see "Initiate an acoustic or visual signal for a defined period of time [sMN FindMe]", page 141):	0 1 2 3 4 5	00 01 02 03 04 05
LED ID	Name of the LED	String	8		LED1 LED2	4C 45 44 31 4C 45 44 32

Table 347: Example: sRA LEDState

CoLa A	<STX>sRA{SPC}LED-State{SPC}2{SPC}0{SPC}0{SPC}LED2{SPC}0{SPC}0{SPC}0{SPC}0{SPC}1{SPC}0{SPC}LED1{SPC}0{SPC}0{SPC}0{SPC}0<ETX>
	02 73 52 41 20 4C 45 44 53 74 61 74 65 20 32 20 30 20 30 20 4C 45 44 32 00 00 00 00 20 31 20 30 20 4C 45 44 31 00 00 00 00 03

CoLa B	02 02 02 02 00 00 00 23 73 52 41 20 4C 45 44 53 74 61 74 65 20 00 02 00 00 4C 45 44 32 00 00 00 00 01 00 4C 45 44 31 00 00 00 00 7A

12.4.1.4.9 Application

12.4.1.4.9.1 Set activation of evaluation group [sMN ActivateEvaluationGroup]

The telegram is intended to activate or deactivate groups via telegram. The group activation needs to be changed from always to telegram in advance.

Table 348: Telegram structure: sMN ActivateEvaluationGroup

Telegram structure: sMN ActivateEvaluationGroup (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E
Command	Activate / deactivate evaluation group	String	23		ActivateEvaluationGroup	41 63 74 69 76 61 74 65 45 76 61 6C 75 61 74 69 6F 6E 47 72 6F 75 70
Amount of evaluation groups to activate/ deactivate	Array	Uint_16	2	**Each array element contains two parameters:** • Evalution group number • Deactivation / activation of evaluation group	1d ... 48d (1 ... 30h)	00 01 ... 00 30
Evalution group number	Evaluation group 1	UInt_16	2		1	00 01
Deactivation / activation of evaluation group	Activate / deactivate evaluation group 1	Bool_1	1	Deactivate: Activate:	0 1	00 01
Evalution group number	Evaluation group 2	UInt_16	2		2	00 02
Activate/ deactivate evaluation group	Activate / deactivate evaluation group 2	Bool_1	1	Deactivate: Activate:	0 1	00 01
...						
Evalution group number	Evaluation group 48	UInt_16	2		48d (30h)	00 30
Activate/ deactivate evaluation group	Activate / deactivate evaluation group 48	Bool_1	1	Deactivate: Activate:	0 1	00 01

Table 349: Example1: sMN ActivateEvaluationGroup – Deactivate evaluation group 1

CoLa A	<STX>sMN{SPC}ActivateEvaluationGroup{SPC}1{SPC}1{SPC}0<ETX>
	<STX>sMN ActivateEvaluationGroup 1 1 0<ETX>
	sMN ActivateEvaluationGroup 1 1 0
	02 73 4D 4E 20 41 63 74 69 76 61 74 65 45 76 61 6C 75 61 74 69 6F 6E 47 72 6F 75 70 20 31 20 31 20 30 03
CoLa B	02 02 02 02 00 00 00 21 73 4D 4E 20 41 63 74 69 76 61 74 65 45 76 61 6C 75 61 74 69 6F 6E 47 72 6F 75 70 20 00 01 00 01 00 20
	73 4D 4E 20 41 63 74 69 76 61 74 65 45 76 61 6C 75 61 74 69 6F 6E 47 72 6F 75 70 20 00 01 00 01 00

Table 350: Example2: sMN ActivateEvaluationGroup – Deactivate evaluation group 1, activate evaluation group 2 and 3

CoLa A	<STX>sMN{SPC}ActivateEvaluationGroup{SPC}3{SPC}1{SPC}0{SPC}2{SPC}1{SPC}3{SPC}1<ETX>
	<STX>sMN ActivateEvaluationGroup 3 1 0 2 1 3 1<ETX>
	sMN ActivateEvaluationGroup 3 1 0 2 1 3 1
	02 73 4D 4E 20 41 63 74 69 76 61 74 65 45 76 61 6C 75 61 74 69 6F 6E 47 72 6F 75 70 20 33 20 31 20 30 20 32 20 31 20 33 20 31 03
CoLa B	02 02 02 02 00 00 00 21 73 4D 4E 20 41 63 74 69 76 61 74 65 45 76 61 6C 75 61 74 69 6F 6E 47 72 6F 75 70 20 00 03 00 01 00 00 02 01 00 03 01 23
	73 4D 4E 20 41 63 74 69 76 61 74 65 45 76 61 6C 75 61 74 69 6F 6E 47 72 6F 75 70 20 00 03 00 01 00 00 02 01 00 03 01

Table 351: Telegram structure: sAN ActivateEvaluation

Telegram structure: sAN ActivateEvaluation						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 52 41
Command	Activate / deactivate evaluation group	String	23		ActivateEvaluationGroup	41 63 74 69 76 61 74 65 45 76 61 6C 75 61 74 69 6F 6E 47 72 6F 75 70
Amount of activated/ deactivated evaluation groups	-	UInt_16	2		1h... 30h	00 01 ... 00 30
State of activation / deactivation	-	Bool_1	1	Activation/Deactivation failed: Successfully activated/deactivated:	0 1	00 01

Table 352: Example 1: sAN ActivateEvaluation – Successfully deactivated evaluation group 1

CoLa A	<STX>sAN{SPC}ActivateEvaluation{SPC}1{SPC}1<ETX>
	02 73 41 4E 20 41 63 74 69 76 61 74 65 45 76 61 6C 75 61 74 69 6F 6E 47 72 6F 75 70 20 31 20 31 03
CoLa B	02 02 02 02 00 00 00 1F 73 41 4E 20 41 63 74 69 76 61 74 65 45 76 61 6C 75 61 74 69 6F 6E 47 72 6F 75 70 20 00 01 01 2C

Table 353: Example 2: sAN ActivateEvaluation – Successfully deactivated evaluation group 1, successfully activated evaluation group 2, failed activation of evaluation group 3

CoLa A	<STX>sAN{SPC}ActivateEvaluation{SPC}3{SPC}1{SPC}1{SPC}0<ETX>
	02 73 41 4E 20 41 63 74 69 76 61 74 65 45 76 61 6C 75 61 74 69 6F 6E 47 72 6F 75 70 20 31 20 31 20 31 20 30 03
CoLa B	02 02 02 02 00 00 00 1A 73 41 4E 20 41 63 74 69 76 61 74 65 45 76 61 6C 75 61 74 69 6F 6E 47 72 6F 75 70 20 00 03 01 01 00 73

12.4.1.4.9.2 Set field evaluation contour [sMN SetFieldEvaluationContour]

This telegram describes how to change the polygon coordinates. A predefined polygon is required. No setup of a polygon from scratch possible.

Table 354: Telegram structure: sMN SetFieldEvaluationContour

Telegram structure: sMN SetFieldEvaluationContour (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Method	String	3		sMN	73 4D 4E
Command	Set the region of interest of an object detection evaluation	String	25		SetFieldEvaluationContour	53 65 74 46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 43 6F 6E 74 6F 75 72
Evalution ID	-	UInt_16	2		+1d ... +48d (1h ... 30h)	00 01 ... 00 30
Amount of polygon verticies	E. g. triangle = 3, square = 4	Array	... 800		+3d ... +800d (3h ... 320h)	00 03... 03 20
Polygon vertex	Coordinates of first polygon vertex in mm	Dint_32	4	X coordinate:	-60,000d ... +60,000d (FFFF15A0h ... EA60h)	FF FF 15 A0 ... 00 00 EA 60
				Y coordinate:	-60,000d ... +60,000d (FFFF15A0h ... EA60h)	FF FF 15 A0 ... 00 00 EA 60
Polygon vertex	Coordinates of second polygon vertex in mm	Dint_32	4	X coordinate:	-60,000d ... +60,000d (FFFF15A0h ... EA60h)	FF FF 15 A0 ... 00 00 EA 60
				Y coordinate:	-60,000d ... +60,000d (FFFF15A0h ... EA60h)	FF FF 15 A0 ... 00 00 EA 60
...						
Polygon vertex	Coordinates of last polygon vertex in mm	Dint_32	4	X coordinate:	-60,000d ... +60,000d (FFFF15A0h ... EA60h)	FF FF 15 A0 ... 00 00 EA 60
				Y coordinate:	-60,000d ... +60,000d (FFFF15A0h ... EA60h)	FF FF 15 A0 ... 00 00 EA 60

Telegram structure: sMN SetFieldEvaluationContour (User level 'Authorized client' required)						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Lower Z limit	Lower bound of extrusion in Z direction	Dint_32	4		-60,000d ... +60,000d (FFFF15A0h ... EA60h)	FF FF 15 A0 ... 00 00 EA 60
Upper Z limit	Upper bound of extrusion in Z direction	Dint_32	4		-60,000d ... +60,000d (FFFF15A0h ... EA60h)	FF FF 15 A0 ... 00 00 EA 60

Table 355: Example: sMN SetFieldEvaluationContour – Conversion of an evaluation into a 1 m^3 cube

CoLa A	<STX>sMN{SPC}SetFieldEvaluationContour{SPC}1{SPC}4{SPC}+1000{SPC}+1000{SPC}-1000{SPC} +1000{SPC}-1000{SPC}-1000{SPC}+1000{SPC}-1000{SPC}-500{SPC}+500<ETX>
	<STX>sMN SetFieldEvaluationContour 1 4 +1000 +1000 -1000 +1000 -1000 -1000 +1000 -1000 -500 +500<ETX>
	sMN SetFieldEvaluationContour 1 4 +1000 +1000 -1000 +1000 -1000 -1000 +1000 -1000 -500 +500
	02 73 4D 4E 20 53 65 74 46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 43 6F 6E 74 6F 75 72 20 31 20 34 20 2B 31 30 30 30 20 2B 31 30 30 30 20 2D 31 30 30 30 20 2B 31 30 30 30 20 2D 31 30 30 30 20 2D 31 30 30 30 20 2B 31 30 30 30 20 2D 31 30 30 30 20 2D 35 30 30 20 2B 35 30 30 03
CoLa B	02 02 02 02 00 00 00 4A 73 4D 4E 20 53 65 74 46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 43 6F 6E 74 6F 75 72 20 00 01 00 04 00 00 03 E8 00 00 03 E8 FF FF FC 18 00 00 03 E8 FF FF FC 18 FF FF FC 18 00 00 03 E8 FF FF FC 18 FF FF FE 0C 00 00 01 F4 1A
	73 4D 4E 20 53 65 74 46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 43 6F 6E 74 6F 75 72 20 00 01 00 04 00 00 03 E8 00 00 03 E8 FF FF FC 18 00 00 03 E8 FF FF FC 18 FF FF FC 18 00 00 03 E8 FF FF FC 18 FF FF FE 0C 00 00 01 F4

Table 356: Telegram structure: sAN SetFieldEvaluationContour

Telegram structure: sAN SetFieldEvaluationContour						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sAN	73 52 41
Command	Set the region of interest of an object detection evaluation	String	25		SetFieldEvaluationContour	53 65 74 46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 43 6F 6E 74 6F 75 72
State of evaluation conversion	-	Enum_8	1	Conversion successful: Invalid Evaluation: Invalid Polygon: Invalid Z limit:	0 1 2 3	00 01 02 03

Table 357: Example: sAN SetFieldEvaluationContour – Evaluation conversion successful

CoLa A	<STX>sAN{SPC}SetFieldEvaluationContour{SPC}0<ETX>
	02 73 41 4E 20 53 65 74 46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 43 6F 6E 74 6F 75 72 20 30 03
CoLa B	02 02 02 02 00 00 00 1F 73 41 4E 20 53 65 74 46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 43 6F 6E 74 6F 75 72 20 00 14

12.4.1.4.9.3 Read the current evaluation configuration state [sRN EvaluationConfigState]

The telegram is intended to read the current configuration state of the field evaluation. Depending on the device computational limits can be reached. If so the telegram returns an error or warning if the number of beams or evaluations reaches the maximum defined threshold.

Table 358: Telegram structure: sRN EvaluationConfigState

Telegram structure: sRN EvaluationConfigState						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Info on current configuration state	String	21		EvaluationConfigState	45 76 61 6C 75 61 74 69 6F 6E 43 6F 6E 66 69 67 53 74 61 74 65

Table 359: Example: sRN EvaluationConfigState

CoLa A	<STX>sRN{SPC}EvaluationConfigState<ETX>	
	<STX>sRN EvaluationConfigState<ETX>	
	sRN EvaluationConfigState	
	02 73 52 4E 20 45 76 61 6C 75 61 74 69 6F 6E 43 6F 6E 66 69 67 53 74 61 74 65 03	
CoLa B	02 02 02 02 00 00 00 19 73 52 4E 20 45 76 61 6C 75 61 74 69 6F 6E 43 6F 6E 66 69 67 53 74 61 74 65 12	
	73 52 4E 20 45 76 61 6C 75 61 74 69 6F 6E 43 6F 6E 66 69 67 53 74 61 74 65	

Table 360: Telegram structure: sRA EvaluationConfigState

Telegram structure: sRA EvaluationConfigState						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Info on current configuration state	String	21		EvaluationConfigState	45 76 61 6C 75 61 74 69 6F 6E 43 6F 6E 66 69 67 53 74 61 74 65
Active beams limit		Enum_8	1	OK: Warning: Error:	0 1 2	00 01 02
Active evaluations limit		Enum_8	1	OK: Warning: Error:	0 1 2	00 01 02
Number of intersecting beams	Array	Uint_32	48		0 ... 65535 (0 ... FFFF)	00 00 ... FF FF

Table 361: Example: sRA EvaluationConfigState - one evaluation active, intersected by 172 beams, status of active beam limit and active evaluation limit is OK

CoLa A	<STX>sRA{SPC}EvaluationConfig-State{SPC}0{SPC}0{SPC}AC{SPC}0{SPC} ETX>
	02 73 52 41 20 45 76 61 6C 75 61 74 69 6F 6E 43 6F 6E 66 69 67 53 74 61 74 65 20 30 20 30 20 41 43 20 30 03
CoLa B	02 02 02 02 00 00 00 7C 73 52 41 20 45 76 61 6C 75 61 74 69 6F 6E 43 6F 6E 66 69 67 53 74 61 74 65 20 00 00 00 AC 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 87

12.4.1.4.9.4 Read the current field evaluation application state [sRN FieldEvaluationApplication-State]

This telegram is intended to read the field evaluation application state. It returns the information whether the field evaluation application is active, configuring or deactivated.

Further information on the different states:

- Deactivated - no evaluation is set up
- Active - at least one active evaluation (group activation has no influence on that)
- Configuring - new calculation of the e.g. field geometries

Table 362: Telegram structure: sRN FieldEvaluationApplicationState

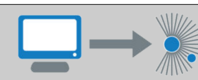
Telegram structure: sRN FieldEvaluationApplicationState						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Info on current field evaluation application state	String	31		FieldEvaluatio-nApplication-State	46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 41 70 70 6C 69 63 61 74 69 6F 6E 53 74 61 74 65

Table 363: Example: sRN FieldEvaluationApplicationState

CoLa A	<STX>sRN{SPC}FieldEvaluationApplicationState<ETX>
	<STX>sRN FieldEvaluationApplicationState<ETX>
	sRN FieldEvaluationApplicationState
	02 73 52 4E 20 46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 41 70 70 6C 69 63 61 74 69 6F 6E 53 74 61 74 65 03
CoLa B	02 02 02 02 00 00 00 23 73 52 4E 20 46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 41 70 70 6C 69 63 61 74 69 6F 6E 53 74 61 74 65 36
	73 52 4E 20 46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 41 70 70 6C 69 63 61 74 69 6F 6E 53 74 61 74 65

Table 364: Telegram structure: sRA FieldEvaluationApplicationState

Telegram structure: sRA FieldEvaluationApplicationState						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Info on current field evaluation application state	String	31		FieldEvaluatio-nApplication-State	46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 41 70 70 6C 69 63 61 74 69 6F 6E 53 74 61 74 65
Field evaluation application state	-	Enum_8	1	Deactivated: Active: Configuring: Error:	0 1 2 3	00 01 02 03

Table 365: Example: sRA FieldEvaluationApplicationState

CoLa A	<STX>sRA{SPC}FieldEvaluationApplicationState{SPC}1<ETX>
	02 73 52 41 20 46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 41 70 70 6C 69 63 61 74 69 6F 6E 53 74 61 74 65 20 31 03
CoLa B	02 02 02 02 00 00 00 25 73 52 41 20 46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 41 70 70 6C 69 63 61 74 69 6F 6E 53 74 61 74 65 20 01 18

12.4.1.4.9.5 Read field evaluation result [sRN FieldEvaluationResult]

This telegram returns the status information of all evaluations.

Table 366: Telegram structure: sRN FieldEvaluationResult

Telegram structure: sRN FieldEvaluationResult						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Read	String	3		sRN	73 52 4E
Command	Info on field evaluation result	String	21		FieldEvaluation-Result	46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 52 65 73 75 6C 74

Table 367: Example: sRN FieldEvaluationResult

CoLa A	<STX>sRN{SPC}FieldEvaluationResult<ETX>	
	<STX>sRN FieldEvaluationResult<ETX>	
	sRN FieldEvaluationResult	
	02 73 52 4E 20 46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 52 65 73 75 6C 74 03	
CoLa B	02 02 02 02 00 00 00 19 73 52 4E 20 46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 52 65 73 75 6C 74 12	
	73 52 4E 20 46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 52 65 73 75 6C 74	

Table 368: Telegram structure: sRA FieldEvaluationResult

Telegram structure: sRA FieldEvaluationResult						
Telegram part	**Description**	**Variable**	**Length**	**Additional details**	**Values CoLa A (ASCII)**	**Values CoLa B (Binary)**
Command type	Answer	String	3		sRA	73 52 41
Command	Info on field evaluation result	String	21		FieldEvaluation-Result	46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 52 65 73 75 6C 74
Version	Version number	Uint_16	2		0 ... FFFFh	00 ... FF FF
Time-stamp	Current system time since power on in ms	Uint_32	4		0 ... FFFFFFFFh	00 00 00 00 ... FF FF FF FF
Evaluation Result List	Maximum possible evaluations = 48	Array	48	Not configured: Inactive: Free: Detecting infringed: Infringed:	0 1 2 3 4	00 01 02 03 04

Table 369: Example: sRA FieldEvaluationResult

CoLa A	<STX>sRA{SPC}FieldEvaluationRe-sult{SPC}1{SPC}0{SPC}ETX>
	02 73 52 41 20 46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 52 65 73 75 6C 74 20 31 20 30 03
CoLa B	02 02 02 02 00 00 00 7C 73 52 41 20 46 69 65 6C 64 45 76 61 6C 75 61 74 69 6F 6E 52 65 73 75 6C 74 20 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 87

12.4.1.5 Diagnostics

12.4.1.5.1 SOPAS error codes

sFA ErrorCode

Table 370: SOPAS error codes

Telegram structure: sFA ErrorCode			
Error code	**Description**	**Dec.**	**Hex.**
Sopas_Ok	No error	0	0
Sopas_Error_METHODIN_ACCESSDENIED	Wrong userlevel, access to method not allowed	1	1
Sopas_Error_METHODIN_UNKNOWNINDEX	Trying to access a method with an unknown Sopas index	2	2
Sopas_Error_VARIABLE_UNKNOWNINDEX	Trying to access a variable with an unknown Sopas index	3	3
Sopas_Error_LOCALCONDITIONFAILED	Local condition violated, e.g. giving a value that exceeds the minimum or maximum allowed value for this variable	4	4

Telegram structure: sFA ErrorCode			
Error code	**Description**	**Dec.**	**Hex.**
Sopas_Error_INVALID_DATA	Invalid data given for variable, this errorcode is deprecated (is not used anymore).	5	5
Sopas_Error_UNKNOWN_ERROR	An error with unknown reason occurred, this errorcode is deprecated.	6	6
Sopas_Error_BUFFER_OVERFLOW	The communication buffer was too small for the amount of data that should be serialised.	7	7
Sopas_Error_BUFFER_UNDERFLOW	More data was expected, the allocated buffer could not be filled.	8	8
Sopas_Error_ERROR_UNKNOWN_TYPE	The variable that shall be serialised has an unknown type. This can only happen when there are variables in the firmware of the device that do not exist in the released description of the device. This should never happen.	9	9
Sopas_Error_VARIABLE_WRITE_ACCESSDENIED	It is not allowed to write values to this variable. Probably the variable is defined as read-only.	10	A
Sopas_Error_UNKNOWN_CMD_FOR_NAMESERVER	When using names instead of indices, a command was issued that the nameserver does not understand.	11	B
Sopas_Error_UNKNOWN_COLA_COMMAND	The CoLa protocol specification does not define the given command, command is unknown.	12	C
Sopas_Error_METHODIN_SERVER_BUSY	It is not possible to issue more than one command at a time to an SRT device.	13	D
Sopas_Error_FLEX_OUT_OF_BOUNDS	An array was accessed over its maximum length.	14	E
Sopas_Error_EVENTREG_UNKNOWNINDEX	The event you wanted to register for does not exist, the index is unknown.	15	F
Sopas_Error_COLA_A_VALUE_OVERFLOW	The value does not fit into the value field, it is too large.	16	10
Sopas_Error_COLA_A_INVALID_CHARACTER	Character is unknown, probably not alphanumeric.	17	11
Sopas_Error_OSAI_NO_MESSAGE	Only when using SRTOS in the firmware and distributed variables this error can occur. It is an indication that no operating system message could be created. This happens when trying to GET a variable.	18	12
Sopas_Error_OSAI_NO_ANSWER_MESSAGE	This is the same as `Sopas_Error_OSAI_NO_MESSAGE` with the difference that it is thrown when trying to PUT a variable.	19	13
Sopas_Error_INTERNAL	Internal error in the firmware, problably a pointer to a parameter was null.	20	14
Sopas_Error_HubAddressCorrupted	The Sopas Hubaddress is either too short or too long.	21	15
Sopas_Error_HubAddressDecoding	The Sopas Hubaddress is invalid, it can not be decoded (Syntax).	22	16
Sopas_Error_HubAddressAddressExceeded	Too many hubs in the address	23	17
Sopas_Error_HubAddressBlankExpected	When parsing a HubAddress an expected blank was not found. The HubAddress is not valid.	24	18
Sopas_Error_AsyncMethodsAreSuppressed	An asynchronous method call was made although the device was built with “AsyncMethodsSuppressed”. This is an internal error that should never happen in a released device.	25	19

Telegram structure: sFA ErrorCode			
Error code	**Description**	**Dec.**	**Hex.**
Sopas_Error_ComplexArraysNotSupported	Device was built with „ComplexArraysSuppressed“ because the compiler does not allow recursions. But now a complex array was found. This is an internal error that should never happen in a released device.	26	20

Table 371: Example: sFA ErrorCode Wrong userlevel

CoLa A	<STX>sFA{SPC}1<ETX>
	02 73 46 41 20 31 03
CoLa B	02 02 02 02 00 00 00 05 73 46 41 20 00 01 75

12.4.1.5.2 Additional information

SOPAS communication is a index based communication and can be identified with telegram beginning of: sRI, sWI, sMI, sAI, sEI, sSI. Since the parallel usage of one port might be confusing, the usage of separate ports is adviced.

Every response telegram starts with a separat framed string:

<STX>sSI 2 1<ETX><STX>“Answer”<ETX>

8028981/102C/2024-06-19/en

Australia
Phone +61 (3) 9457 0600
1800 33 48 02 – tollfree
E-Mail sales@sick.com.au

Austria
Phone +43 (0) 2236 62288-0
E-Mail office@sick.at

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

Brazil
Phone +55 11 3215-4900
E-Mail comercial@sick.com.br

Canada
Phone +1 905.771.1444
E-Mail cs.canada@sick.com

Czech Republic
Phone +420 234 719 500
E-Mail sick@sick.cz

Chile
Phone +56 (2) 2274 7430
E-Mail chile@sick.com

China
Phone +86 20 2882 3600
E-Mail info.china@sick.net.cn

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

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

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

Germany
Phone +49 (0) 2 11 53 010
E-Mail info@sick.de

Greece
Phone +30 210 6825100
E-Mail office@sick.com.gr

Hong Kong
Phone +852 2153 6300
E-Mail ghk@sick.com.hk

Hungary
Phone +36 1 371 2680
E-Mail ertekesites@sick.hu

India
Phone +91-22-6119 8900
E-Mail info@sick-india.com

Israel
Phone +972 97110 11
E-Mail info@sick-sensors.com

Italy
Phone +39 02 27 43 41
E-Mail info@sick.it

Japan
Phone +81 3 5309 2112
E-Mail support@sick.jp

Malaysia
Phone +603-8080 7425
E-Mail enquiry.my@sick.com

Mexico
Phone +52 (472) 748 9451
E-Mail mexico@sick.com

Netherlands
Phone +31 (0) 30 204 40 00
E-Mail info@sick.nl

New Zealand
Phone +64 9 415 0459
0800 222 278 – tollfree
E-Mail sales@sick.co.nz

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

Poland
Phone +48 22 539 41 00
E-Mail info@sick.pl

Romania
Phone +40 356-17 11 20
E-Mail office@sick.ro

Singapore
Phone +65 6744 3732
E-Mail sales.gsg@sick.com

Slovakia
Phone +421 482 901 201
E-Mail mail@sick-sk.sk

Slovenia
Phone +386 591 78849
E-Mail office@sick.si

South Africa
Phone +27 10 060 0550
E-Mail info@sickautomation.co.za

South Korea
Phone +82 2 786 6321/4
E-Mail infokorea@sick.com

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

Sweden
Phone +46 10 110 10 00
E-Mail info@sick.se

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

Taiwan
Phone +886-2-2375-6288
E-Mail sales@sick.com.tw

Thailand
Phone +66 2 645 0009
E-Mail marcom.th@sick.com

Turkey
Phone +90 (216) 528 50 00
E-Mail info@sick.com.tr

United Arab Emirates
Phone +971 (0) 4 88 65 878
E-Mail contact@sick.ae

United Kingdom
Phone +44 (0)17278 31121
E-Mail info@sick.co.uk

USA
Phone +1 800.325.7425
E-Mail info@sick.com

Vietnam
Phone +65 6744 3732
E-Mail sales.gsg@sick.com

Detailed addresses and further locations at **www.sick.com**

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

SICK
Sensor Intelligence.