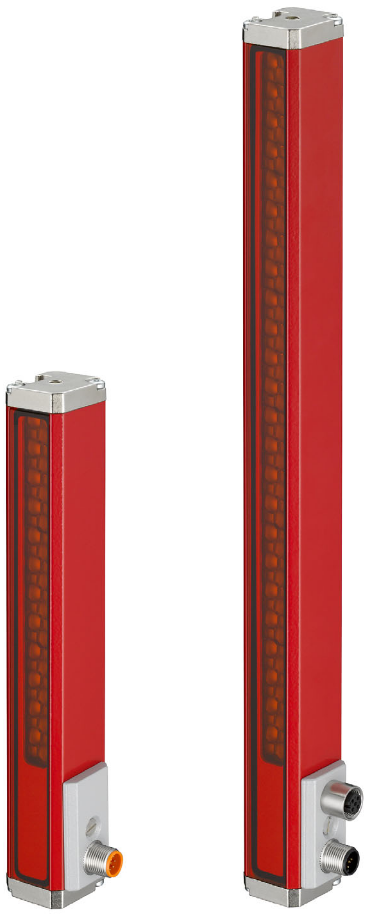
Original operating instructions

OGS 600

Optical guidance sensor



The Sensor People

We reserve the right to make technical changes
EN 2021/11/04 - 50137686

Leuze electronic GmbH + Co. KG
In der Braike 1
D-73277 Owen / Germany
Phone: +49 7021 573-0
Fax: +49 7021 573-199
http://www.leuze.com
info@leuze.de

1 About this document

This technical description contains information regarding the proper use of the OGS 600 optical guidance sensors.

1.1 Explanation of symbols

The symbols used in this technical description are explained below.

	⚠ CAUTION!
⚠	This symbol precedes text messages which must strictly be observed. Failure to observe the provided instructions could lead to personal injury or damage to equipment.

	NOTE
i	This symbol indicates text passages containing important information.

1.2 Terms and abbreviations

AGV	Automated Guided Vehicle
DTM	Device Type Manager
EMC	Electromagnetic compatibility
EN	European standard
FDT	Field Device Tool
FE	Functional earth
GUI	Graphical User Interface
IO or I/O	Input/Output
OGS	Optical Guidance Sensor
PD	Process data
RO	Read Only
RW	Read/Write
PLC	Programmable Logic Control
UART	Universal Asynchronous Receiver Transmitter, here: RS232 / RS422 / RS485
WO	Write Only

Table 1.1: Terms and abbreviations

2 Safety

This sensor was developed, manufactured and tested in line with the applicable safety standards. It corresponds to the state of the art.

2.1 Intended use

The OGS 600 optical guidance sensor measures the contrast of a guide trace that is affixed to the ground. The sensor supplies the vehicle's position data on the guide trace, which determines the course of travel.

Areas of application

The OGS 600 optical guidance sensor is designed for the following area of application:

- Intralogistics – in-house material flow with automated guided vehicles (AGV).

	⚠ CAUTION!
⚠	**Observe intended use!** The protection of personnel and the device cannot be guaranteed if the device is operated in a manner not complying with its intended use. ↳ Only operate the device in accordance with its intended use. ↳ Leuze electronic GmbH + Co. KG is not liable for damages caused by improper use. Read the supplement and these operating instructions for the device before commissioning the device. Knowledge of these documents is required in order to use the equipment for its intended purpose.

	NOTE
ℹ	The optical guidance sensors of the OGS 600 series correspond to the following classification with respect to the integrated lighting: ↳ Illumination red: risk group 0 (exempt group) in acc. with EN 62471

	NOTE
ℹ	**Comply with conditions and regulations!** ↳ Observe the locally applicable legal regulations and the rules of the employer's liability insurance association.

2.2 Foreseeable misuse

Any use other than that defined under "Intended use" or which goes beyond that use is considered improper use.

In particular, use of the device is not permitted in the following cases:

- in rooms with explosive atmospheres
- as stand-alone safety component in accordance with the machinery directive [1)]
- for medical purposes

	NOTE
ℹ	**Do not modify or otherwise interfere with the device!** ↳ Do not carry out modifications or otherwise interfere with the device. The device must not be tampered with and must not be changed in any way. ↳ The device must not be opened. There are no user-serviceable parts inside. ↳ Repairs must only be performed by Leuze electronic GmbH + Co. KG.

2.3 Competent persons

Connection, mounting, commissioning and adjustment of the device must only be carried out by competent persons.

1) Use as a safety-related component within a safety function is not permissible.

Prerequisites for competent persons:

- They have a suitable technical education.
- They are familiar with the rules and regulations for occupational safety and safety at work.
- They are familiar with the original operating instructions of the device.
- They have been instructed by the responsible person on the mounting and operation of the device.

Certified electricians

Electrical work must be carried out by a certified electrician.

Due to their technical training, knowledge and experience as well as their familiarity with relevant standards and regulations, certified electricians are able to perform work on electrical systems and independently detect possible dangers.

In Germany, certified electricians must fulfill the requirements of accident-prevention regulations BGV A3 (e.g. electrician foreman). In other countries, there are respective regulations that must be observed.

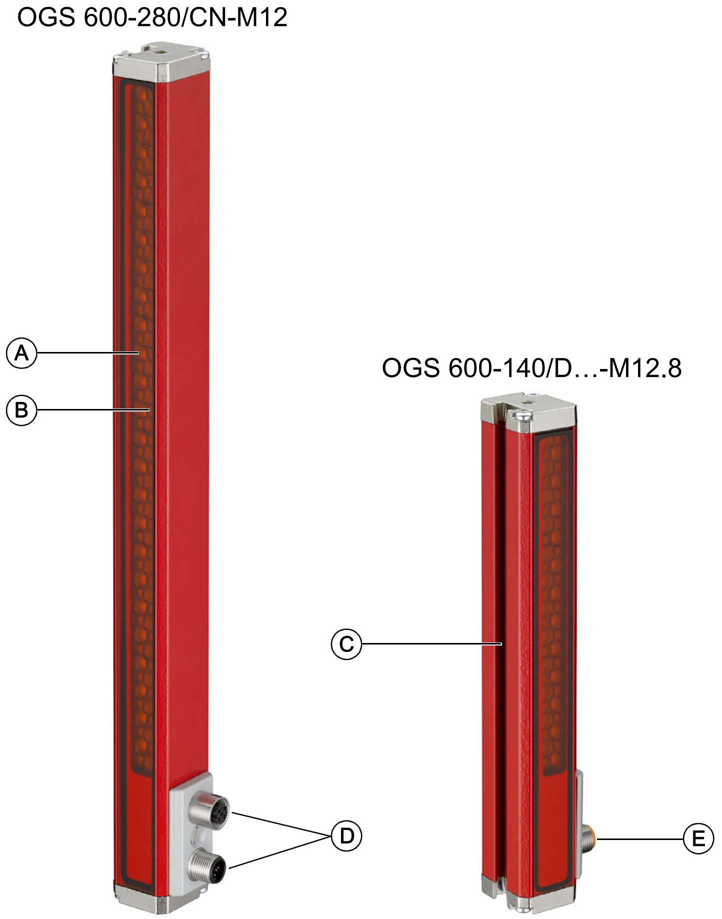
2.4 Exemption of liability

Leuze electronic GmbH + Co. KG is not liable in the following cases:

- The device is not being used properly.
- Reasonably foreseeable misuse is not taken into account.
- Mounting and electrical connection are not properly performed.
- Changes (e.g., constructional) are made to the device.

3 Device description

3.1 Device overview



A Integrated illumination – transmitter (small lenses)
B Receiver modules – receiver (large lenses)
C Groove with 2 sliding blocks for fastening the mounting bracket
D 2 x M12 connector, 5-pin
E 1 x M12 connector, 8-pin

Figure 3.1: Device overview

3.2 Performance characteristics

3.2.1 Trace detection

The sensor is designed to detect an optical guide trace on the floor and to output the position of the guide trace relative to the sensor.

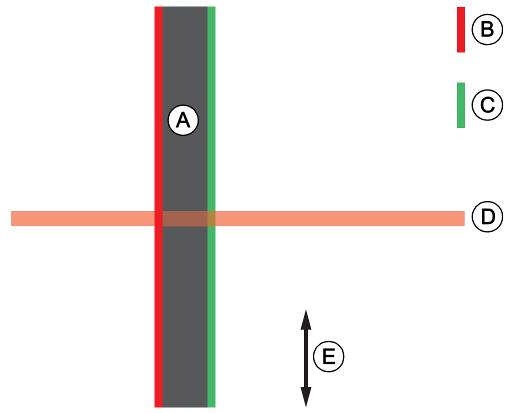
The sensor can detect a light trace on a dark background or, inversely, a dark trace on a light background.

The sensor can detect up to 6 guide traces. Each guide trace consists of a left edge (shown in red below) and a right edge (show in green below). This edge information is output for each detected guide trace.

When a guide trace is detected, the sensor therefore outputs two pieces of information in the process data for each trace:

- position of the left edge of the guide trace and
- position of the right edge of the guide trace

The difference between these two edge positions is the track width.



A Guide trace; here dark trace on light background
B Left edge of guide trace
C Right edge of guide trace
D Width of the sensor's measurement field
E AGV direction of travel

Figure 3.2: Schematic illustration of the guide trace under the optical guidance sensor

3.2.2 Measurement time

The sensor delivers an updated measurement every 10ms.

3.2.3 Filter

The detection of incorrect floor markings can be minimized considerably by means of the following filters, which can be activated separately: "Trace width", "Minimum contrast" and "Trace amplitude".

Filtered-out traces can be read out in separate parameters (see the chapter "Index overview – More data on correct and incorrect traces").

Chapter 9 "Tips for initial commissioning" contains information on using the filters.

3.2.4 Switches

At each switch, the sensor outputs two or more traces. The users themselves decide which trace they want to follow. The switch function is provided to ensure that the wide midpoint of the type 2 switch (see chapter 8.4 "Switch") is detected properly when the trace width filter is active.

Example:

When the vehicle moves over a seamlessly bonded switch (type 2) and a turn request is given, the vehicle control unit can very early on follow the edge position in the direction in which the vehicle is to turn.

If the vehicle is to turn left, the vehicle is guided by the left edge. The turn operation then begins before the sensor has passed the midpoint of the switch and outputs two traces.

3.2.5 Faults

If floor markings occur which, despite activated filters, are detected as valid, then these markings are output. The vehicle's control unit must ensure that position jumps in the output traces are detected and that they are not followed.

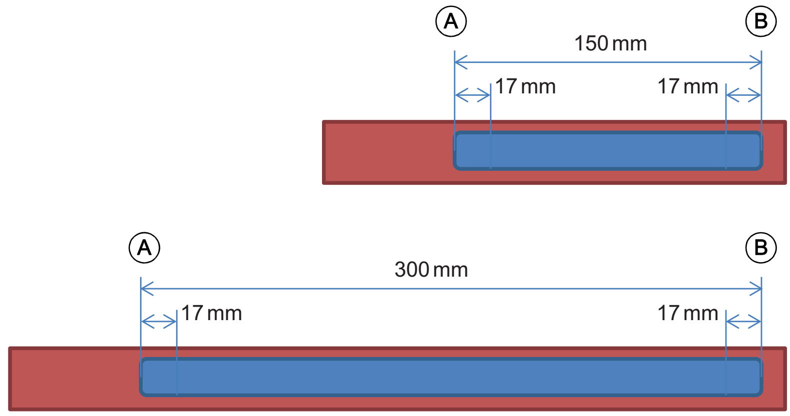
3.2.6 Output value

The sensor outputs the position of the left edge and the right edge of the optical guide trace in **mm * 10**. The output value range is therefore:

- Short version **OGS 600-140…**: 0 … 1500.
- Long version **OGS 600-280…**: 0 … 3000.

A trace is detected if it enters **the sensor's measurement field by at least 17mm at the left or right edge.** This corresponds to an output value of:

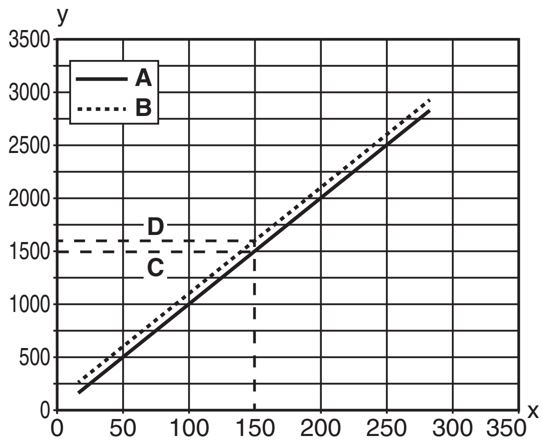
- Short version **OGS 600-140…**: 170 … 1330.
- Long version **OGS 600-280…**: 170 … 2830.



A Start of measurement field, output value **0**
B End of measurement field, output value **1500** or **3000**

Figure 3.3: Measurement field of the guidance sensor

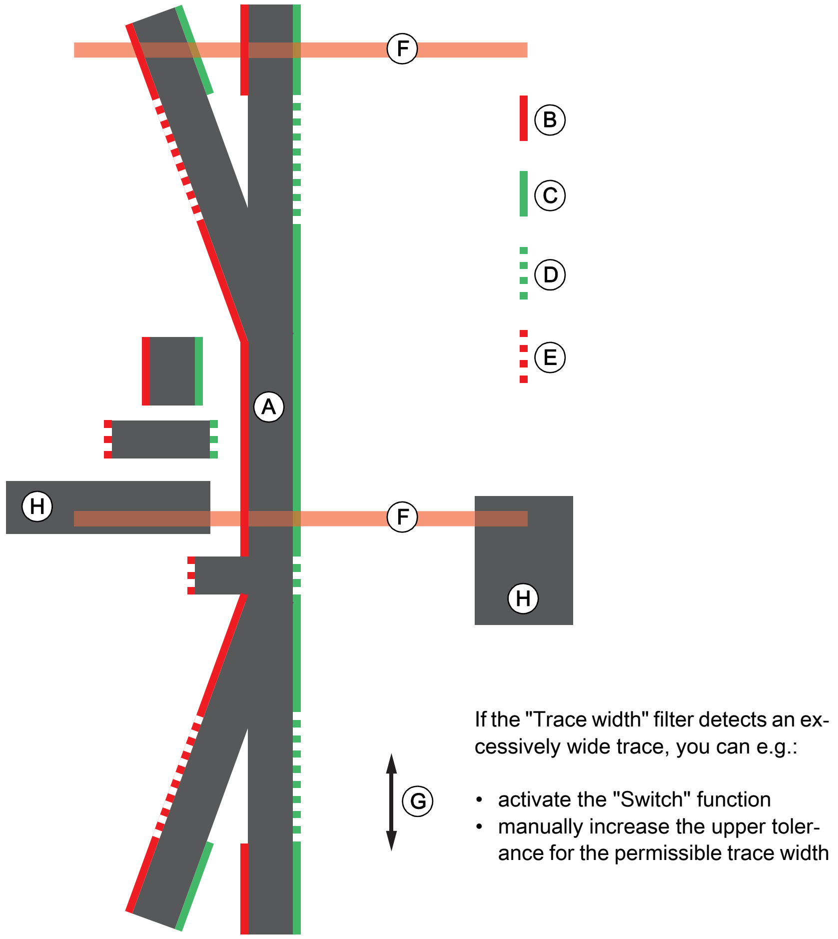
The trace width is the absolute value of the difference between the right and left edge of the trace.



A Measurement value for left edge of guide trace
B Measurement value for right edge of guide trace
C Output value: Position of left edge
D Output value: Position of right edge
x Position of left edge of guide trace under sensor
y Output value: Edge positions

Figure 3.4: Sensor characteristic curve with one trace (long version)

3.2.7 Example: Guide trace detection with active "Trace width" filter



A Guide trace; here dark trace on light background
B Left edge of guide trace
C Right edge of guide trace
D, E Trace too wide according to active "Trace width" filter
F Width of the sensor's measurement field
G AGV direction of travel
H Objects not detected as trace

Figure 3.5: Schematic illustration of the guide trace under the optical guidance sensor

3.3 Guide trace requirements

To ensure error-free detection of the optical guide trace on the floor, the guide trace must meet the requirements described in the following sections.

3.3.1 Trace color

The illumination of the sensors emits red light. As a result, the contrast that the sensor sees is different to that perceived by the human eye.
The following overview shows how the sensor sees different colors.

Color perceived by human eye	Floor/background			Guide trace
	RAL color	RAL no.	Measurement value of the sensor: Amplitude [LSB]	Suitable trace color
White	Traffic white	9016	21200	Leuze black [1)]
Black	Jet black	9005	400	Leuze white [1)]
Red	Tomato red	3013	11800	Black
Orange	Deep orange	2011	17400	Black
Yellow	Melon yellow	1028	19800	Black
Green	Emerald green	6001	1200	White
Blue	Ultramarine blue	5002	700	White

1) Leuze trace tapes available as accessories:
OTB 40-BK250, black, 40mm wide, self-adhesive, 25m roll (part no. 50137874)
OTB 40-WH250, white, 40mm wide, self-adhesive, 25m roll (part no. 50137875)

Table 3.1: Color comparison between the sensor and the human eye.

NOTE

A detailed table with sensor measurement values can be found in the appendix (see chapter 14 "Appendix – Sensor measurement values for RAL colors").

3.3.2 Trace width

The maximum width of the trace is limited only by the sensor's measurement field (see figure 3.3). The trace must have a minimum width so that a sufficiently good contrast can be achieved. The "Trace width" filter can be adjusted to the trace by means of a trace width teach.
The recommended trace width is approx. 30 … 40mm.

Trace width	OGS 600-280…	OGS 600-140…
Maximum	266mm	106mm
Minimum	20mm	20mm

Table 3.2: Maximum/minimum trace widths

3.3.3 Free space next to the trace

To ensure error-free detection, there should not be any other marking within a distance of at least 30mm from the actual trace.
At distances greater than 30mm from the trace, the floor can have any color.

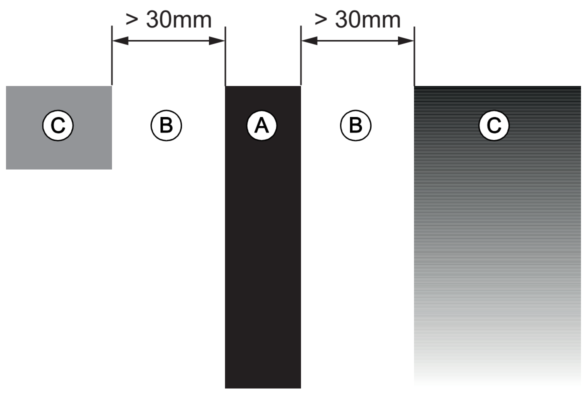


Figure 3.6: Minimum distance between the guide trace and other objects on the floor

The same applies to an inverted layout with a light guide trace on a dark background.

3.4 Connection technology

All device connections are based on M12 connection technology, see chapter 5 "Electrical connection".

	NOTE
ℹ	**Shielding!** The shielding is connected via the M12 connector housing. ↳ Use only shielded connection cables!

3.5 Controls and indicators

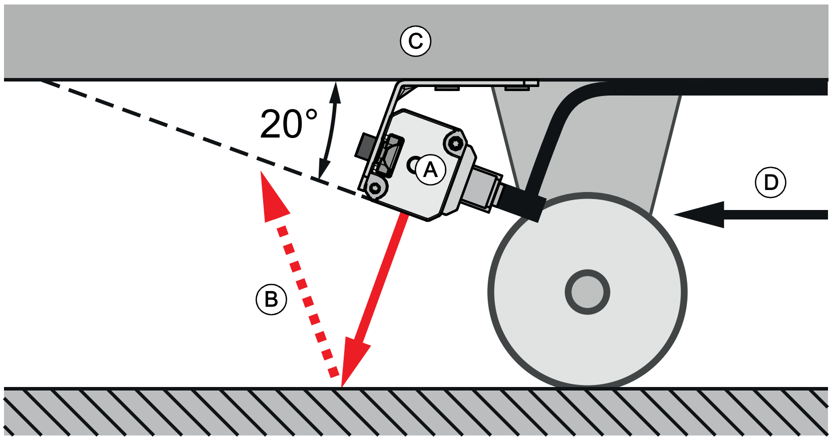
The optical guidance sensor does not have any operating elements or indicators.
The sensor is operated and its configuration checked only via the serial interface or via the CAN bus.

4 Mounting

4.1 General mounting instructions

The device is mounted using the groove integrated in the profile. Two sliding blocks with M6 thread are included in the delivery and already inserted into the groove.

NOTE
Mount the sensor at an angle! The sensor must be mounted at an angle of 20° to ensure that reflective surfaces have no effect on the evaluation.



A Optical guidance sensor
B Reflection of reflecting surfaces
C Automated guided vehicle (AGV)
D Direction of travel

Figure 4.1: Angled sensor mounting to prevent unwanted reflection

The sensor can be mounted using the mounting brackets included in the delivery (see chapter 4.3 "Mounting accessories"). They ensure that the sensor is pointing at the floor at the correct angle.

4.2 Selecting a mounting location

Reliable detection of the guide trace primarily depends on how good the contrast is between the trace and the background.
In order to select the right mounting location, several factors must be considered:

- The distance of the sensor to the trace to be detected should be 10 … 70mm.
- The guide trace must have a minimum width of 20mm.
- The linearity error of the output value depends on the distance to the floor.
- Diffuse reflection of the trace. Ideally, a jet black guide trace on a pure white background should be used.

4.3 Mounting accessories

The following items are included with the sensor:

- 2x M6 sliding blocks (inserted in the groove)
- 2x mounting brackets for mounting the sensor at an angle of 20°.

NOTE
Dimensioned drawings! The dimensioned drawings with the installation dimensions of the sensor can be found in Chapter 11.2.

5 Electrical connection

NOTE

The corresponding mating connectors and ready-made cables are available as accessories for all M12 connections. For further information, see chapter 12 "Order guide and accessories".

5.1 Safety notices for the electrical connection

⚠ CAUTION!

- ↳ Before connecting the device, be sure that the supply voltage agrees with the value printed on the name plate.
- ↳ The device may only be connected by a qualified electrician.
- ↳ Ensure that the functional earth (FE) is connected correctly. Unimpaired operation is only guaranteed when the functional earth is connected properly.
- ↳ If faults cannot be cleared, the device should be switched off and protected against accidental use.

NOTE

Protective Extra Low Voltage (PELV)!

The OGS 600 optical guidance sensors are designed in accordance with protection class III for supply by PELV (protective extra-low voltage with reliable disconnection).

NOTE

Shielding connection!

The shielding is connected via the M12 connector housing.

- ↳ Use only shielded connection cables!

NOTE

Degree of protection IP65 is achieved only if the connectors and caps are screwed into place.

5.2 Voltage supply

The OGS 600 guidance sensors are designed for a voltage supply of 18 … 30VDC (PELV – protective extra low voltage with reliable disconnection). The current consumption with 24 V DC is approx. 180 mA.

5.2.1 Shielding

NOTE

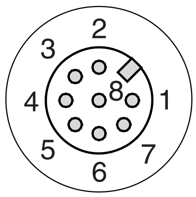
Shielded connection cables!

Only shielded connection cables should be used; they ensure that the housing of the OGS 600 is connected to functional earth.

- ↳ Use only shielded connection cables!
- ↳ The shielding must be connected to earth potential on the connection side.
- ↳ If unshielded connection cables are used, a separate cable must be routed from the housing to the earth potential (additional earthing screw on the housing cover and in the fastening groove).

5.3 Pin assignment

5.3.1 OGS 600-…/D3-M12.8 with RS485 interface

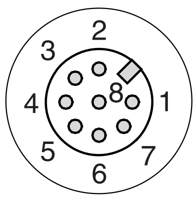


M 12 connector (A-coded)

PWR/RS485, 8-pin M12 connector, A-coded			
Pin	Name	Comment	IN / OUT
1	VIN	Operating voltage +18 … +30VDC	IN
2	IO	Switching input or switching output	IN / OUT
3	GND	Operating voltage 0VDC / reference ground	IN
4	SW_IO	Switching output	OUT
5	RX / TX +	Signal line of RS485 interface	IN / OUT
6	RX / TX -	Signal line of RS485 interface	IN / OUT
7	n. c.	Not connected	
8	n. c.	Not connected	
Thread	FE	Functional earth (housing)	

Table 5.1: PWR/RS485 – Pin assignment for OGS 600 with RS485 interface

5.3.2 OGS 600-…/D2-M12.8 with RS422 interface

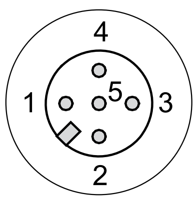


M 12 connector (A-coded)

PWR/RS422, 8-pin M12 connector, A-coded			
Pin	Name	Comment	IN / OUT
1	VIN	Operating voltage +18 … +30VDC	IN
2	IO	Switching input or switching output	IN / OUT
3	GND	Operating voltage 0VDC / reference ground	IN
4	SW_IO	Switching output	OUT
5	TX +	Signal line of RS422 interface	OUT
6	TX-	Signal line of RS422 interface	OUT
7	RX +	Signal line of RS422 interface	IN
8	RX-	Signal line of RS422 interface	IN
Thread	FE	Functional earth (housing)	

Table 5.2: PWR/RS422 – Pin assignment for OGS 600 with RS422 interface

5.3.3 OGS 600-…/CN-M12 with CANopen and RS232 interface



M 12 connector (A-coded)

PWR/RS232, 5-pin M12 connector, A-coded			
Pin	Name	Comment	IN / OUT
1	VIN	Operating voltage +18 … +30VDC	IN
2	RxD	Signal line of RS232 interface	IN
3	GND	Operating voltage 0VDC / reference ground	IN
4	SW_IO	Switching output	OUT
5	TxD	Signal line of RS232 interface	OUT
Thread	FE	Functional earth (housing)	

Table 5.3: PWR/RS232 – Pin assignment for OGS 600 with CANopen/RS232 interface

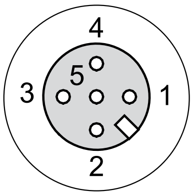
CAN, 5-pin M12 socket, A-coded				
	Pin	Name	Comment	IN / OUT
 M 12 socket (A-coded)	1	SHIELD	CAN functional earth	
	2	n. c.	Not connected	
	3	CAN_GND	Reference level for CAN signal lines	
	4	CAN_High	CAN bus A signal line	IN / OUT
	5	CAN_Low	CAN bus B signal line	IN / OUT
	Thread	FE	Functional earth (housing)	

Table 5.4: CAN – Pin assignment for OGS 600 with CANopen/RS232 interface

5.4 Switching inputs/outputs

NOTE

The devices with RS485 and RS422 interface have two IO pins:
- SW_IO (pin 4) switching output (configurable)
- IO (pin 2) switching input or switching output (configurable)

The devices with CANopen and RS232 interface have only one IO pin:
- SW_IO (pin 4) switching output (configurable)

5.4.1 Function of the SW_IO and IO switching outputs

The switching outputs are configured via index accesses only. The possible function range is the same for both switching outputs. The switching outputs can be configured independently of each other.

Two functions are available which can be signaled via the switching output.

Trace monitoring

An upper and lower position value can be defined using two parameters. The limit values are compared with the values of the detected trace.

If the left or the right edge of the detected trace is greater than the limit value, the switching output is activated.

If more than one trace is detected, then the outermost edges are always used for monitoring.

The function has a hysteresis.

Contrast monitoring

An upper and a lower value can be defined for contrast using two parameters. The limit values are compared internally with the values of the contrast measured for the current trace.

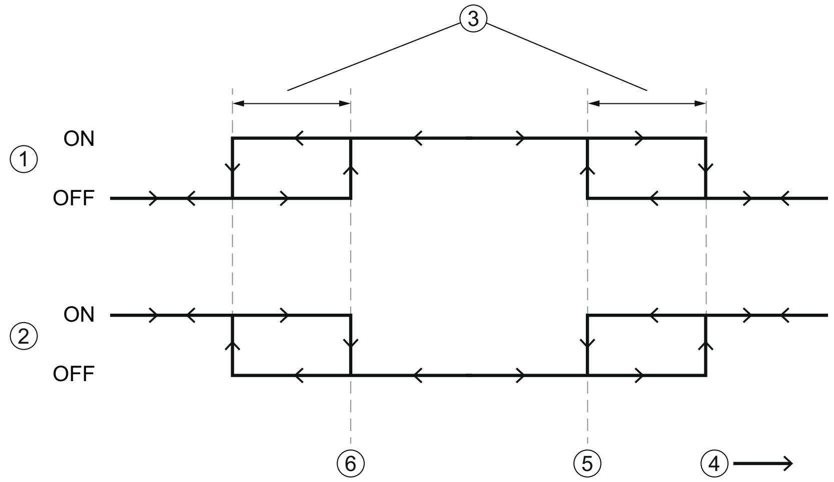
If the contrast is greater or less than the limit value, the switching output is activated.

NOTE

Deactivation of a switching output

Both switching outputs SW_IO and IO can also be deactivated independently of each other.

Switching behavior



1 High signal within switching points
2 High signal outside switching points
3 Hysteresis
4 Measurement value: object position or contrast
5 Upper switching point
6 Lower switching point

Figure 5.1: Switching behavior of the switching outputs

	NOTE
ℹ	The switching outputs can be configured independently of each other as: • Push-pull switching output • PNP switching output • NPN switching output

5.4.2 Switching output SW_IO (pin 4)

The functions of the switching output are described in Chapter 5.4.1.

The switching output SW_IO is connected to pin 4 on all device models (see chapter 5.3 "Pin assignment"). The function of the switching output can be configured via indices.

Name	Index UART	Index [sub-index] CANopen	Index length [byte]	Access	Default data	Function / value [dec.]
Q1UserConfig	87_d	2003_h [6_h]	2	RW	0_d	0_d: not active 1_d: Out_PP (push-pull) 2_d: Out_NPN 3_d: Out_PNP
Q1SwitchPtMode	80_d	2003_h [4_h]	2	RW	0_d	0_d: deactivated 1_d: trace monitoring 2_d: contrast monitoring
Q1UpperSwitchingPoint	77_d	2003_h [1_h]	2	RW	0_d	Upper limit. Trace position in mm * 10 Contrast value in LSB
Q1LowerSwitchingPoint	78_d	2003_h [2_h]	2	RW	0_d	Lower limit. Trace position in mm * 10 Contrast value in LSB
Q1Hysteresis	81_d	2003_h [5_h]	2	RW	20_d	Hysteresis in absolute values. Applies to both limits. Unit: mm * 10 or LSB
Q1LightDark	79_d	2003_h [3_h]	2	RW	0_d	0_d: output has high signal outside switching points 1_d: output has high signal within switching points
Qproperty	76_d	2005_h [0_h]	2	RW	0_d	0_d: switching output switches OFF 1_d: switching output switches ON 2_d: switching output remains unchanged Takes effect in the case of • Activation/deactivation • Global error (UART index 200_d, and CAN index 2020_h [1_h], value 0001_h) with detailed info in UART index 201_d and CAN index 2020_h [2_h]

Table 5.5: Configuration options for switching output SW_IO (pin 4)

5.4.3 IO switching output/switching input (pin 2)

The functions of the switching output are described in Chapter 5.4.1.

The IO switching output is connected to pin 2 on the device models with RS485 and RS422 interface (see chapter 5.3 "Pin assignment"). The function of the switching output can be configured via indices.

Name	Index UART	Index [sub-index] CANopen	Index length [byte]	Access	Default data	Function / value [dec.]
Q2UserConfig	88_d	2004_h [6_h]	2	RW	0_d	0_h: inactive 1_h: Out_PP (push-pull) 2_h: Out_NPN 3_h: Out_PNP 104_h: In_NPN deactivation input 105_h: In_PNP deactivation input 304_h: In_NPN activation input 305_h: In_PNP activation input
Q2SwitchPtMode	85_d	2004_h [4_h]	2	RW	0_d	0_d: deactivated 1_d: trace monitoring 2_d: contrast monitoring
Q2UpperSwitchingPoint	82_d	2004_h [1_h]	2	RW	0_d	Upper limit. Trace position in mm * 10 Contrast value in LSB
Q2LowerSwitchingPoint	83_d	2004_h [2_h]	2	RW	0_d	Lower limit. Trace position in mm * 10 Contrast value in LSB
Q2Hysteresis	86_d	2004_h [5_h]	2	RW	20_d	Hysteresis in absolute values. Applies to both limits. Unit: mm * 10 or LSB
Q2LightDark	84_d	2004_h [3_h]	2	RW	0_d	0_d: output has high signal outside switching points 1_d: output has high signal within switching points
Qproperty	76_d	2005_h [0_h]	2	RW	0_d	0_d: switching output switches OFF 1_d: switching output switches ON 2_d: switching output remains unchanged Takes effect in the case of • Activation/deactivation • Global error (UART index 200_d, and CAN index 2020_h [1_h], value 0001_h) with detailed info in UART index 201_d and CAN index 2020_h [2_h]

Table 5.6: Configuration options IO switching output/switching input (pin 2)

5.4.4 Function of the IO switching input (pin 2)

The switching input is configured via index accesses only (see Table 5.6).

Two functions are available which can be activated via the switching input.

Activation

A high signal at the switching input activates the sensor illumination; a low signal deactivates the sensor illumination.

Deactivation

A high signal at the switching input deactivates the sensor illumination; a low signal activates the sensor illumination.

> **NOTE**
>
> **Output behavior with deactivated sensor illumination**
>
> With the sensor illumination deactivated, the sensor does not deliver any measurement values. In this case, the output behavior of the switching output (pin 2, pin 4) with the trace monitoring or contrast monitoring function can be controlled via UART index 76_d (CANopen index 2005_h) *Qproperty*.
>
> This setting has no effect on the output of process data.

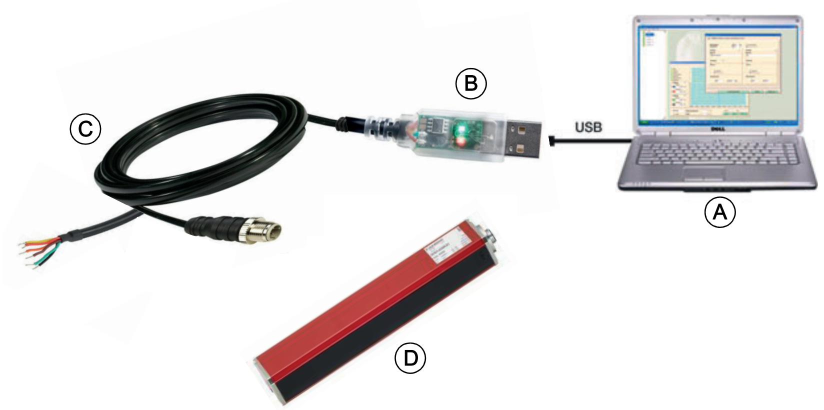
> **NOTE**
>
> **Deactivation of the switching input**
>
> The switching input IO can also be deactivated.

5.5 Connection to the PC via RS232/RS422/RS485

The devices can be configured via the RS232/RS422/RS485 interface using the Windows software OGS600.exe or Sensor Studio.

All connections via the serial interfaces require a USB adapter which provides a virtual COM port on the PC.

For the RS422/RS485 interface, a USB adapter and a Y-cable are available as accessories for setting up the connection between the sensor, voltage supply and USB adapter.



A PC with OGS 600 software
B RS485-USB adapter (on request)
C Y-connection cable (on request)
D Optical guidance sensor

Figure 5.2: Connection of the OGS 600 to the PC using the RS485 interface

The adapter set and the Y-connection cable are available as accessories on request.

Information about installation and use of the software can be found in Chapter 6 "OGS 600 GUI of configuration/diagnostic software" on page 23.

6 OGS 600 GUI of configuration/diagnostic software

6.1 Installing the required software

6.1.1 System requirements

Operating system:	Windows 7, Windows 8, Windows 10	
Computer:	PC with USB interface version 1.1 or higher	
	Processor type:	1 GHz or higher
	Main memory:	1 GB RAM (32-bit operating system)
		2 GB RAM (64-bit operating system)
		Required hard drive capacity: Approx. 10 MB
Graphics card:	Min. resolution 1280 x 1024	

> **NOTE**
>
> Administrator privileges on the PC are necessary for installing the OGS 600 GUI.

6.1.2 Installation instructions

↳ Download the configuration software from the Internet: www.leuze.com > Products > Measuring sensors > Sensors for positioning > Optical guidance > OGS 600 > (Device model) > Downloads > Software/drivers

↳ Copy the file into a suitable folder on your hard drive and unpack the zip file.

↳ Start the Setup_OGS600.exe file and follow the instructions on the screen.

↳ The installation wizard installs the software and places a shortcut on the desktop and in the start menu.

6.2 Starting the configuration/diagnostic software

↳ Start the configuration software using the shortcut *OGS 600* on the desktop or from the Start menu.

6.3 Short description of the configuration/diagnostic software

The control software is intended to provide an overview of the sensor's functions. For this purpose, measurement data and detected traces are visualized.

There is a function for recording raw values and data of the guide trace.

This can be used for configuring CANopen devices via the RS232 interface.

The control software offers the following functions

- Firmware update via UART boot loader
- Visualization of the measurement values
- Saving of measurement values
- Visualizing the detected guide traces
- Visualizing the filter settings
- Changing filter settings manually
- Performing the various teach modes for the filters
- Querying of the process data
- Reading out of valid and invalid traces
- Reading and writing indices
- Configuring the CANopen properties

7 Commissioning

7.1 Communication protocol for serial interfaces (UART)

The following default settings apply for the RS232, RS485 and RS422 serial interfaces.

Baud rate [bit/s]	115200
Parity	Odd
Data bits	8
Stop bits	1
Node number	1
Minimum response time	Can be set for RS485; see the parameter *RS485Delay* (index 149).

Table 7.1: Factory settings for the communication protocol of serial interfaces

7.1.1 RS485/RS422 node address

The node address is set via index 70 *UART Node No* (see chapter 7.2 "Object directory for serial interfaces (UART)"). It is recommended to change the default address if, in the case of RS485/RS422, multiple devices are connected to the bus.

If a device is reset to the factory settings, the default address (1) is restored. This avoids an address collision.

7.1.2 Error handling

The following communication errors are intercepted and/or returned:

- Too few characters:
  after timeout (1.6ms), the receive buffer is cleared -> no error telegram.
- Too many characters: cannot be detected. The valid characters are processed (CRC check) and the remaining characters are discarded.
- Incorrect CRC: error telegram 8112_h
- Error during reception (parity error, ...): error telegram 8113_h
- Incorrect identifier: error telegram 8111_h
- Maximum time taken for the sensor to respond to a query: 1.2 ms

7.1.3 Index access

Basic structure of the protocol:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9
Node no./ identifier	Length	Index lowbyte	Index highbyte	Sub-index	Data 0	Data 1	Data 2	Data 3	CRC

Byte 0: Always contains the device address (node number). This can be changed. The identifier indicates what is to be done: read, write, request PD.
Bit 3…0: Identifier
Bit 7…4: Node no. *n*

Byte 1: Contains the number of data bytes.
The length is counted from byte 5 to byte n-1.

Byte 2: Contains the lowbyte of the index to be read or written.

Byte 3: Contains the highbyte of the index to be read or written.

Byte 4: Contains the sub-index of the index to be read or written.

Byte 5…n: Data which is written or read.

Byte n+1: CRC is calculated from byte 0 to byte n. Process: XOR with start value 0.

In the case of access to an index, the identifier indicates what is to be done. There are three different query identifiers. The sensor returns a corresponding identifier in response to the query.

If an error was detected in the data transfer, the identifier nF_h and an error code (see chapter 7.1.5 "Error codes") are returned as the response.

Identifiers

Type	Identifier for query	Identifier Sensor response	Function of byte 1 "Length"
Read	$n1_h$	$n4_h$	Response from sensor: Length indicates how much data is sent by the sensor: from byte 5 without CRC byte
Write	$n2_h$	$n8_h$	Write to sensor: Length indicates how much data is sent to the sensor; if the data length exceeds the object length, an error is returned.
Process data	$n3_h$	nC_h	
Error		nF_h	
n = node number			

Table 7.2: Identifiers for working with indices

Example:

	Type	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte n + 1
		Node no./ identifier	Length	Index lowbyte	Index highbyte	Sub-index	Data *n*	CRC
Query	Read	11_h	0	$C8_h$	00_h	0	CRC	
Response	Read	14_h	Quantity Data bytes	$C8_h$	00_h	0	Data *n*	CRC

Table 7.3: Example of a query to read byte

Node no.: 1

Index: 200 (LowByte: $C8_h$, HighByte: 00_h)

7.1.4 Process data

There are different types of process data. They are used for calling up various information.

In addition, a setting for the "Switch" function can be changed in the sensor by sending the process data byte query.

Process data query

The structure of the query from byte 0 to byte 4 is the same for all process data types.

	Type	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4
Query	PD	$n3_h$	PD type	PD-In1	PD-In2	CRC

Byte 0: Always contains the device address (node number). This can be changed. The identifier indicates what is to be done: request PD.

Bit 3…0: Identifier

Bit 7…4: Node no. *n*

Byte 1: Contains the process data types: 1, 2, 4, 5, 6 or 7.

Byte 2: PD-In1:
Data can be written to the sensor in order to change settings (example: "Switch" function). The changed setting is applied for the first time with the next PD query.

Byte 3: PD-In2:
Reserve.

Byte 4: CRC is calculated from byte 0 to byte 3. Process: XOR with start value 0.

Process data response

	Type	Byte 0	Byte 1	Byte 2	Byte 3	Byte n	Byte n + 1
Response	PD	nC_h	Length User data	PD status	Contrast	Data	CRC

Table 7.4: Request for process data:

Byte 0: Device number as well as identifier (in this case nC_h)

Byte 1: Number of sent user data bytes, varies depending on the PD type.

Byte 2: Status PD contains the eight most important pieces of information about the detected traces.

Byte 3: Contrast is a value reduced to 8 bits and indicates the contrast of the trace.

Byte 4: Data of the detected traces

Byte n+1: The last byte is the CRC byte.

7.1.4.1 Status byte in the process data

The process data status byte is composed of 8 bits which convey a state for each filter and when the warning threshold for the filter is reached. It is output if a global error is present or when no trace at all has been detected.

The state is always active if the corresponding bit is set.

Bit 0:	General error	–> Reading out of index 201 *Error*
Bit 1:	Minimum contrast warning	see chapter 8.6 ""Minimum contrast" filter"
Bit 2:	Trace amplitude warning	see chapter 8.7 ""Trace amplitude" filter"
Bit 3:	Trace width error	see chapter 8.5 ""Trace width" filter"
Bit 4:	Minimum contrast error	see chapter 8.6 ""Minimum contrast" filter"
Bit 5:	Trace amplitude error	see chapter 8.7 ""Trace amplitude" filter"
Bit 6:	Switch active	see chapter 8.4 "Switch"
Bit 7:	No trace detected	–> Check guide trace/background

NOTE

Data for traces (edge positions, contrast) which are declared as invalid by a filter is **never** output via the process data.

7.1.4.2 Contrast byte in the process data

The difference between the amount of light reflected by the ground next to the guide trace and the amount of light reflected by the guide trace itself is an important measure for assessment of the optical state of the trace.

This value is defined by the following calculation (see also Figure 8.6):

Contrast = Amplitude_of_environment - Amplitude_of_trace

This value is known during commissioning of the system. By moving over the guide trace when it is in its new condition, it is possible to determine the poorest contrast of the system.

The state of the guide trace can be checked continuously during operation.

	NOTE
ℹ	**Contrast output values in the process data** **If one** valid trace is detected, the contrast of this trace is output. **If more than one** valid trace is detected, the contrast of the trace with the poorest contrast, e.g. at a switch, is output.

Conversion

To be able to compare the contrast value in the process data with the values in the indices of the "Minimum contrast" or "Trace amplitude" filters, the value from the process data must be multiplied by **100**.

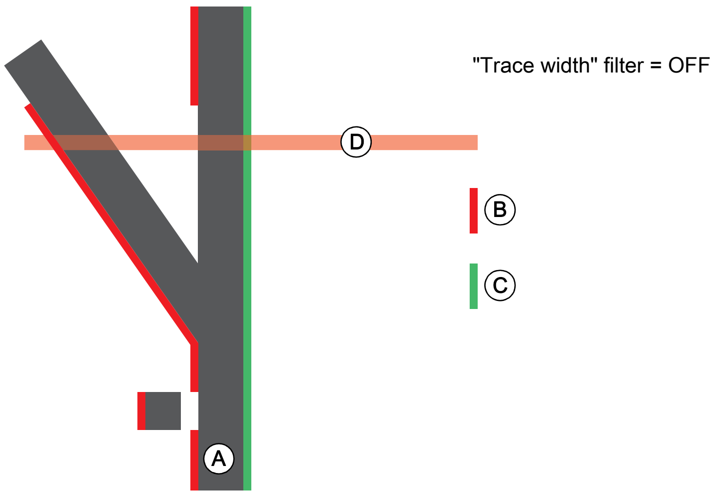
Contrast = Hex2Dec(byte 3) * 100

7.1.4.3 Type 1 process data

Type 1 process data outputs the position of a left and a right edge.

If the sensor finds one trace, the left and the right edge of this trace is output. If the sensor finds two traces, then the outermost left edge and the outermost right edge of the detected traces are output.

If filters such as "trace width", "minimum contrast" or "trace amplitude" are active, they are effective for PD type 1.



A Guide trace; here dark trace on light background
B Left edge of guide trace
C Right edge of guide trace
D Width of the sensor's measurement field

Figure 7.1: Output of the left edge and right edge with PD type 1.

The "Trace width" filter is OFF in Figure 7.1 because the wide trace would otherwise not be detected at the midpoint of the switch.

With active "Trace width" filter, the "Switch" function can be used as an alternative (see chapter 8.4 "Switch"). In this case, additional information is sent in byte 2 (data) when the process data is requested.

Type 1 process data query

	Type	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4
		Node no./ identifier	PD type	PD-In1	PD-In2	CRC
Query	PD	13_h	1_h	0_h	0_h	CRC

Table 7.5: Type 1 process data query

Type 1 process data response

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
	Node no./ identifier	User data length	PD status	Contrast	Left edge lowbyte	Left edge highbyte	Right edge lowbyte	Right edge highbyte	CRC
Response	$1C_h$	04_h	0_h	78_h	$B0_h$	04_h	14_h	05_h	

Table 7.6: Example response for type 1 process data

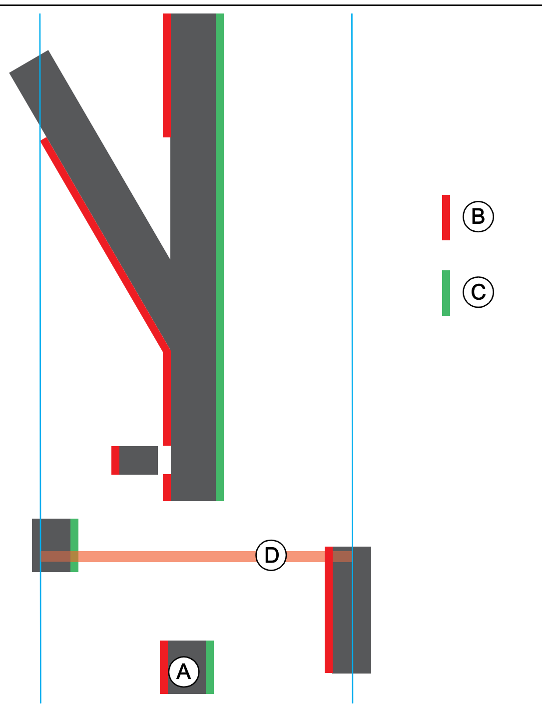
Contrast = 120 * 100 = 12000 LSB
Left edge of trace = 1200 / 10 = 120.0 mm
Right edge of trace = 1300 / 10 = 130,0 mm

7.1.4.4 Type 2 process data

Type 2 process data outputs the position of the left edge found first and the position of the right edge found first. No correlation is determined between the found edges. No search for traces takes place.
If only one edge exists, this is also output. Either as the left or right edge.

Definition of left and right

Left on the device is where the connector plug is located.



A Guide trace; here dark trace on light background
B Left edge of guide trace
C Right edge of guide trace
D Width of the sensor's measurement field

Figure 7.2: Output of the left edge and right edge with PD type 2.

Type 2 process data query

	Type	Byte 0	Byte 1	Byte 2	Byte 3
		Node no./ identifier	PD type	Data In	CRC
Query	PD	13_h	02_h	00_h	CRC

Table 7.7: Type 2 process data query

Type 2 process data response

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
	Node no./ identifier	User data length	PD sta-tus	Contrast	Left edge lowbyte	Left edge highbyte	Right edge lowbyte	Right edge highbyte	CRC
Response	$1C_h$	04_h	00_h	78_h	$B0_h$	04_h	14_h	05_h	BD_h

Table 7.8: Example response for type 2 process data

Contrast = 120 * 100 = 12000 LSB
Left edge of trace = 1200 / 10 = 120.0 mm
Right edge of trace = 1300 / 10 = 130,0 mm

Maximum output value

If only one edge is detected, the value for the undetected edge changes to 380.0.

Filters for type 2 process data

The following filter types function with type 2 process data:

1. "Minimum contrast" filterChapter 8.6
   This acts on all edges which produce a continuous trace where the left and right edge directly follow on from each other.
2. "Trace amplitude" filter Chapter 8.7
   This acts on all edges which produce a continuous trace where the left and right edge directly follow on from each other.

New:

3. "Minimum contrast" filter for outer edges UART Index 113
4. Hysteresis for the position of the outer edges UART Index 114

Outer edges are edges that do not define a trace and occur mainly when the trace leaves the detection range of the sensor.

NOTE

The "Minimum contrast" and "Hysteresis" filters for outer edges always interact. They are always active and cannot be disabled.

Example:

The trace runs to the left out of the sensor's field of vision. With position 15.0 mm, the contrast drops below the minimum contrast set in index 113. The sensor remembers this position as the last valid value.

If the trace moves back from the left into the sensor's field of vision, then the minimum contrast must be fulfilled first. If it is the same trace, this takes place again at position 15.0 mm. Additionally, before the edge is output, the position of the edge must be larger by the value in index 114.

Index 114 = 100 -> 10 mm
15,0 mm + 10 mm = 25 mm
The edge is output as of position 25 mm.

Name	Index	Index length [byte]	Access	Data *Default values	Info
Minimum contrast for outer edges	113	2	RW	5500	Unit: [LSB]
Hysteresis for position of outer edges	114	2	RW	50	100 corresponds to a hysteresis of 10 mm Unit [mm * 10]

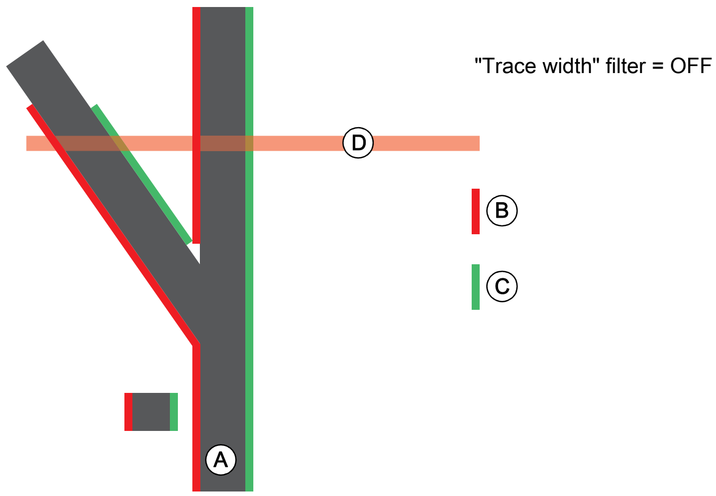
Table 7.9: Index overview for "Minimum contrast" filter

7.1.4.5 Type 4 process data

Type 4 process data outputs the positions of up to six detected traces.

If the sensor finds one trace, the left and the right edge of this trace is output. If the sensors finds two or more traces, the left and right edges of all valid traces are output. The traces in the process data are arranged in ascending order according to the position at which they were found. The trace with the smallest position is always output first and is therefore always trace no. 1. All subsequent traces are arranged in ascending order according to their position and the trace number is incremented.

If filters such as "trace width", "minimum contrast" or "trace amplitude" are active, they are effective for PD type 4.



A Guide trace; here dark trace on light background
B Left edge of guide trace
C Right edge of guide trace
D Width of the sensor's measurement field

Figure 7.3: Output of the left edge and right edge with PD type 1.

The "Trace width" filter is OFF in Figure 7.3, recognizable by the fact that the wide trace is detected at the midpoint of the switch. With active "Trace width" filter, the "Switch" function can be used as an alternative (see chapter 8.4 "Switch"). In this case, additional information is sent in byte 2 (data) when the process data is requested.

Type 4 process data query

	Type	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4
		Node no./ identifier	PD type	PD-In1	PD-In2	CRC
Query	PD	13_h	04_h	0_h	0_h	CRC

Table 7.10: Type 4 process data query

Type 4 process data response

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
					Trace 1			
	Node no./ identifier	User data length	PD status	Contrast	Left edge lowbyte	Left edge highbyte	Right edge lowbyte	Right edge highbyte
Response	$1C_h$	08_h	0_h	78_h	$B0_h$	4_h	14_h	05_h

Byte 8	Byte 9	Byte 10	Byte 11	Byte 12
Trace 2				
Left edge lowbyte	Left edge highbyte	Right edge lowbyte	Right edge highbyte	CRC
DC_h	05_h	40_h	6_h	

Table 7.11: Example response for type 4 process data with 2 traces

Contrast = 120 * 100 = 12000 LSB
Left edge of trace 1 = 1200 / 10 = 120.0 mm
Right edge of trace 1 = 1300 / 10 = 130,0 mm
Left edge of trace 2 = 1500 / 10 = 150.0 mm
Right edge of trace 2 = 1600 / 10 = 160,0 mm

From the number of user data bytes, it is possible to calculate how many traces have been found:

- per edge: 2 bytes
- per trace: 2 edges

=> This results in 4 bytes of user data per trace.

7.1.4.6 Type 5-7 process data

The function and content are the same as for type 2 process data. With type 5 to 7 process data, however, the data quantity may be reduced.

It is used to call up the following information:

- Type 5 process data: left edge
- Type 6 process data: center of trace
- Type 7 process data: right edge

Type 5-7 process data query

	Type	Byte 0	Byte 1	Byte 2	Byte 3
		Node no./ identifier	PD type	Data In	CRC
Query	PD	13_h	5_h	00_h	CRC

Table 7.12: Type 5-7 process data query

Type 5-7 process data response

	Byte 0	Byte 1	Byte 2	Byte 3
	Node no./ identifier	Edge lowbyte	Edge highbyte	CRC
Response	$1C_h$	04_h	00_h	CRC

Table 7.13: Example response for type 5-7 process data

7.1.4.7 Process data type 8 (firmware v1.9 or higher)

Process data type 8 functions in exactly the same way as process data type 4. The only difference is that the number of data items coming from the sensor is always the same. In a query, 6 edges are always output irrespective of how many the sensor has actually detected.

The advantage of this is that the data length is always 17 bytes long and processing in the control is simplified.

If fewer than 6 edges (3 traces) are detected by the sensor, the value 3800 is written to the empty edge position.

Type 8 process data query

	Type	Byte 0	Byte 1	Byte 2	Byte 3
		Node no./ identifier	PD type	Data In	CRC
Query	PD	13_h	8_h	00_h	CRC

Table 7.14: Type 8 process data query

Type 8 process data response

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
					Trace 1			
	Node no./ identifier	User data length	PD status	Contrast	Left edge lowbyte	Left edge highbyte	Right edge lowbyte	Right edge highbyte
Response	$1C_h$	08_h	0_h	78_h	$B0_h$	4_h	14_h	05_h

Byte 8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15	Byte 16
Trace 2				Trace 3				
Left edge lowbyte	Left edge highbyte	Right edge lowbyte	Right edge highbyte	Left edge lowbyte	Left edge highbyte	Right edge lowbyte	Right edge highbyte	CRC
DC_h	05_h	40_h	6_h	$D8_h$	$0E_h$	$D8_h$	$0E_h$	

Table 7.15: Example response for type 8 process data with 2 detected edges with output of 3 traces.

Contrast	= 0x78	= 120 * 100 = 12000 LSB
Left edge of trace 1	= 0x04B0	= 1200 / 10 = 120.0 mm
Right edge of trace 1	= 0x0514	= 1300 / 10 = 130,0 mm
Left edge of trace 2	= 0x05DC	= 1500 / 10 = 150.0 mm
Right edge of trace 2	= 0x0640	= 1600 / 10 = 160,0 mm
Left edge of trace 3	= 0x0ED8	= 3800 / 10 = 380.0 mm
Right edge of trace 3	= 0x0ED8	= 3800 / 10 = 380,0 mm

7.1.5 Error codes

Error code	Error description	Reaction
8011_h	Index is not available / released	Check index
8012_h	Sub-index is not available / released	Sub-index must always be 0
8020_h	Service temporarily unavailable (memory function for flash is still assigned)	Repeat several times, otherwise sensor defective
8023_h	Access denied (index write only)	Check index (see Table 7.18)
8030_h	Value is outside permissible value range	Check value which is to be written to index (see Table 7.18)
8031_h	Maximum is above permissible value range	Check value which is to be written to index (see Table 7.18)
8032_h	Minimum is below permissible value range	Check value which is to be written to index (see Table 7.18)
8033_h	Length of object is above maximum	Check data length (see Table 7.18)
8034_h	Length of object is below minimum	Check data length (see Table 7.18)
8035_h	Unknown command in index 2	Check value. Command not present (see Table 7.19)
8082_h	Internal error -> abort	Repeat several times, otherwise sensor defective
8111_h	UART: incorrect identifier	Check identifier (for valid identifiers, see Table 7.2)
8112_h	UART: incorrect CRC	Check CRC calculation
8113_h	Receive error (parity, …)	Repeat several times, otherwise sensor defective

Table 7.16: Error codes for data transfer

7.2 Object directory for serial interfaces (UART)

Data types		Access:	
string	Convert bytes into ASCII characters in incoming sequence	RW	Read Write
uint16	Arrangement: [LowByte, HighByte]	RO	Read Only
uint32	Arrangement: [LowByte, LowerByte, HigherByte, HighByte]	WO	Write Only
array_uint16	Arrangement: [LowByte1, HighByte1, LowByte2, HighByte2, …]		
int16	Arrangement: [LowByte, HighByte]		

Table 7.17: Object directory – data types and access

UART Index	UART Sub Index	Name	Description	Access	Comment	Default	Value range	Length [byte]	Data type
2_d	0_d	*System Command*	System command	WO	See Table 7.19			2	
16_d	0_d	*Vendor Name*	Device manufacturer	RO	Leuze electronic GmbH + Co. KG			32	string
17_d	0_d	*Vendor Text*	Manufacturer text	RO	Leuze electronic - the sensor people			38	string
18_d	0_d	*Product Name*	Product designation	RO	<Product Name>			32	string
19_d	0_d	*Product ID*	Part no. of the device	RO	<Part number>			16	string
20_d	0_d	*Product Text*	Product text	RO	<Product text>			32	string
21_d	0_d	*Serial Number*	Device serial number	RO	<Serial number>			16	string
22_d	0_d	*Hardware Revision*	Version of the device hardware	RO	<Hardware revision>, e.g. 000B			8	string
23_d	0_d	*Firmware Revision*	Version of the device firmware	RO	<Firmware revision>, e.g. 1.1			8	string
70_d	0_d	*UART Node No*	UART node address	RW	RS485/RS422 device address	1	0…15	2	uint16
71_d	0_d	*UART Baud rate*	UART baud rate	RW	For future use			2	uint16
72_d	0_d	*Can Node No*	CAN node address	RW	CANopen device address	10	0…127	2	uint16
73_d	0_d	*Can Baud rate*	CAN baud rate	RW	0 = 1 Mbit/s 1 = not used 2 = 500 kbit/s 3 = 250 kbit/s 4 = 125 kbit/s 5 = 100 kbit/s 6 = 50 kbit/s 7 = 20 kbit/s 8 = 10 kbit/s	0	0…8	2	uint16
75_d	0_d	*UserMode*	UserMode	RW	Bit 0: 1 = dark trace; 0 = light trace Bit 1: angle compensation active Bit 2: filter: trace width Bit 3: filter: contrast Bit 4: filter: amplitude Bit 5: teach trace width Bit 6: teach contrast Bit 7: teach amplitude Bit 8: retro-reflective trace	Bit 0= 1	0… 65535	2	uint16
76_d	0_d	*Qproperty*	Output behavior with no measurement value	RW	0_d: off, 1_d: on, 2_d: unchanged, applies to both outputs	0	0…2	2	uint16
77_d	0_d	*Q1UpperSwitching Point*	Upper switching point for switching output SW_IO (pin 4)	RW	With trace monitoring: range: 0…3000 (long device version), range: 0…1500 (short device version), unit: 0.1 mm With contrast monitoring: range: 0…21200, unit: [LSB]	0	0… 65535	2	uint16
78_d	0_d	*Q1LowerSwitching Point*	Lower switching point for switching output SW_IO (pin 4)	RW	With trace monitoring: range: 0…3000 (long device version), range: 0…1500 (short device version), unit: 0.1 mm With contrast monitoring: range: 0…21200, unit: [LSB]	0	0… 65535	2	uint16

UART Index	UART Sub Index	Name	Description	Access	Comment	Default	Value range	Length [byte]	Data type
79_d	0_d	*Q1LightDark*	Switching behavior Light/dark switching for switching output SW_IO (pin 4)	RW	0_d: Q = high outside switching points, 1_d: Q = high inside switching points, see Table 5.1	0	0…1	2	uint16
80_d	0_d	*Q1SwitchPtMode*	Switching point mode for switching output SW_IO (pin 4)	RW	0_d: switching output deactivated 1_d: trace monitoring 2_d: contrast monitoring	0	0…2	2	uint16
81_d	0_d	*Q1Hysteresis*	Switching hysteresis for switching output SW_IO (pin 4)	RW	With trace monitoring: range: 0…3000 (long device version), range: 0…1500 (short device version), unit: 0.1 mm With contrast monitoring: range: 0…21200, unit: [LSB]	20	0… 65535	2	uint16
82_d	0_d	*Q2UpperSwitchingPoint*	Upper switching point for switching output IO (pin 2)	RW	With trace monitoring: range: 0…3000 (long device version), range: 0…1500 (short device version), unit: 0.1 mm With contrast monitoring: range: 0…21200, unit: [LSB]	0	0… 65535	2	uint16
83_d	0_d	*Q2LowerSwitchingPoint*	Lower switching point for IO switching output (pin 2)	RW	With trace monitoring: range: 0…3000 (long device version), range: 0…1500 (short device version), unit: 0.1 mm With contrast monitoring: range: 0…21200, unit: [LSB]	0	0… 65535	2	uint16
84_d	0_d	*Q2LightDark*	Switching behavior Light/dark switching for switching output IO (pin 2)	RW	0_d: Q = high outside switching points, 1_d: Q = high inside switching points, see Table 5.1	0	0…1	2	uint16
85_d	0_d	*Q2SwitchPtMode*	Switching point mode for switching output IO (pin 2)	RW	0_d: switching output deactivated 1_d: trace monitoring 2_d: contrast monitoring	0	0…2	2	uint16
86_d	0_d	*Q2Hysteresis*	Switching hysteresis for IO switching output (pin 2)	RW	With trace monitoring: range: 0…3000 (long device version), range: 0…1500 (short device version), unit: 0.1 mm With contrast monitoring: range: 0…21200, unit: [LSB]	20	0… 65535	2	uint16
87_d	0_d	*Q1UserConfig*	Configuration of switching output SW_IO (pin 4)	RW	0_d: not active 1_d: Out_PP (push-pull) 2_d: Out_NPN 3_d: Out_PNP	0	0…3	2	uint16
88_d	0_d	*Q2UserConfig*	Configuration of IO switching output/switching input (pin 2)	RW	0_h: not active 1_h: Out_PP (push-pull) 2_h: Out_NPN 3_h: Out_PNP 104_h: In_NPN deactivation input 105_h: In_PNP deactivation input 304_h: In_NPN activation input 305_h: In_PNP activation input	0	0… 65535	2	uint16

UART Index	UART Sub Index	Name	Description	Access	Comment	Default	Value range	Length [byte]	Data type
100_d	0_d	*TraceWidthMax*	Maximum trace width	RW	For manual configuration (changed by means of a trace width teach!), unit: 0.1 mm	490	0… 65535	2	uint16
101_d	0_d	*TraceWidthMin*	Minimum trace width	RW	For manual configuration (changed by means of a trace width teach!), unit: 0.1 mm	290	0… 65535	2	uint16
102_d	0_d	*TraceWidthTol*	Trace width tolerance	RW	Only required for teach, unit: 0.1 mm.	100	0… 65535	2	uint16
103_d	0_d	*TraceContrastMin*	Minimum contrast	RW	Unit: [LSB]	5500	0… 65535	2	uint16
104_d	0_d	*TraceContrast-Warning*	Contrast warning threshold in %	RW	Unit: %	20	1…100	2	uint16
105_d	0_d	*TraceContrastTol*	Contrast tolerance	RW	Only required for teach, unit: [LSB]	30	0… 65535	2	uint16
106_d	0_d	*TraceAmplitude-Min*	Minimum amplitude	RW	Unit: [LSB]	2500	0… 65535	2	uint16
107_d	0_d	*TraceAmplitude-Warning*	Amplitude warning threshold in %	RW	Unit: %	20	1…100	2	uint16
108_d	0_d	*TraceAmplitudeTol*	Amplitude tolerance for teach	RW	Only required for teach, unit: [LSB]	1000	0… 65535	2	uint16
109_d	0_d	*UserOffset*	Offset for process data output	RW	PD output value = edge position + offset	0	-32768 … 32767	2	int16
110_d	0_d	*SwitchTraceWidth-Factor*	Trace width factor for switch function	RW	Factor for widening trace with active switch function (see index 170_d), unit: %	150	0… 65535	2	uint16
111_d	0_d	*SwitchDeviation-Thr*	Lower limit value for deviation with switch	RW	Used with active switch function, unit: [LSB]	250	0… 65535	2	uint16
112_d	0_d	*TraceTeachThr*	Threshold which is taught	RW	Unit: [LSB]	7000	0… 65535	2	uint16
149_d	0_d	*RS485Delay*	Delay before sending on RS485	RW	Delay after reception of a telegram until response is sent, unit: ms	1	0… 65535	2	uint16
151_d	0_d	*UserState*	Status	RO	Bit 0 = 1: angle compensation OK Bit 1 = 1: trace teach OK	0	0… 65535	2	uint16
170_d	0_d	*SwitchNumber*	Switch function	RW	Activation of switch function for guide trace: 0_d: Switch function not active 1_d: Switch function active for guide trace 1 2_d: Switch function active for guide trace 2 3_d: Switch function active for guide trace 3 4_d: Switch function active for guide trace 4 5_d: Switch function active for guide trace 5 6_d: Switch function active for guide trace 6	0	0…6	2	uint16

UART Index	UART Sub Index	Name	Description	Access	Comment	Default	Value range	Length [byte]	Data type
200_d	0_d	*Status*	Sensor state	RO	Bit 0: Global error Bit 1: Compensation factors valid Bit 2: Teach, compensation measurement running Bit 3: Trace contrast warning Bit 4: Trace amplitude warning Bit 5: Trace width error Bit 6: Contrast error Bit 7: Amplitude error Bit 8: Supply voltage warning Bit 9: Supply voltage error Bit 10: Teaching error Bit 11:Compensation error Bit 12:Switch function active Bit 13:Switch error: unknown trace Bit 14:No trace detected (number of edges < 2) Bit 15:LED illumination active if bit = 1	0	0… 65535	2	uint16
201_d	0_d	*Error*	Error description	RO	Bit 0: Teach: compensation values missing Bit 1: Teach: valid traces > 1; invalid traces; switch active Bit 2: Angle compensation: compensation values missing Bit 3: Angle compensation: trace or edge detected Bit 4: Hardware error: Measurement interrupt error Bit 5: Supply voltage warning Bit 6: Supply voltage error Bit 7: Switch: unknown trace	0	0… 2^{32}-1	4	uint32
202_d	0_d	*Pixel*	Single pixel measurement values	RO	Amplitude of 94 receiver signals, unit: [LSB]		0… 65535	188	array _uint16
205_d	0_d	*TraceValidNum*	Valid traces: number	RO	Value: 0 … 6	0	0…6	2	uint16
206_d	0_d	*TraceValidPixel*	Valid traces: pixels	RO	Contains raw edge data of valid traces	0	0… 65535	24	array _uint16
207_d	0_d	*TraceValidSubPixel*	Valid traces: subpixels in mm	RO	Contains edge positions of valid traces, unit: [mm] (see Chapter 8.8)	0	0… 65535	24	array _uint16
208_d	0_d	*TraceValidAmp*	Valid traces: amplitude	RO	Contains amplitude of environment and of valid trace, unit: [LSB] (see Chapter 8.8)	0	0… 65535	24	array _uint16
209_d	0_d	*TraceValidThreshold*	Valid traces: threshold	RO	Contains threshold for edge position of every detected trace, unit: [LSB]	0	0… 65535	24	array _uint16
210_d	0_d	*TraceValidStatus*	Valid traces: status	RO	Status is signaled for every valid trace: Bit 0: Contrast warning Bit 1: Trace amplitude warning (see Chapter 8.8)	0	0… 65535	12	array _uint16
211_d	0_d	*TraceInvalidNum*	Invalid traces: number	RO	Value: 0 … 6	0	0…6	2	uint16
212_d	0_d	*TraceInvalidPixel*	Invalid traces: pixels	RO	Contains raw edge data of invalid traces	0	0… 65535	24	array _uint16
213_d	0_d	*TraceInvalidSubPixel*	Invalid traces: subpixels in mm	RO	Contains edge positions of invalid traces, unit: [mm] (see Chapter 8.8)	0	0… 65535	24	array _uint16
214_d	0_d	*TraceInvalidAmp*	Invalid traces: amplitude	RO	Contains amplitude of environment and of invalid trace, unit: [LSB] (see Chapter 8.8)	0	0… 65535	24	array _uint16

UART Index	UART Sub Index	Name	Description	Access	Comment	Default	Value range	Length [byte]	Data type
215_d	0_d	*TraceInvalidStatus*	Invalid traces: status	RO	Status is signaled for every invalid trace: Bit 0: Contrast error Bit 1: Trace amplitude error Bit 1: Trace width error (see Chapter 8.8)	0	0… 65535	12	array _uint16
216_d	0_d	*Contrast*	Minimum contrast of all traces	RO	Unit: [LSB]	0	0… 65535	2	uint16
220_d	0_d	*SupplyVoltage*	Supply voltage	RO	Unit: [mV]	0	0… 65535	2	uint16
221_d	0_d	*TempController*	Temperature controller	RO	Unit: [°C]	0	0… 65535	2	uint16
836_d	0_d	TraceSensitivity	Sensitivity of trace detection	RW	50 = high sensitivity	100	50… 1000	2	uint16

Table 7.18: Object directory for serial interfaces (UART)

7.2.1 System commands for serial interfaces

Commands can be sent to the sensor via UART index 2 *System Command*.

Command	Value		Function / description
	Dec.	Hex	
Device Reset	128_d	80_h	Software reset
Factory reset	130_d	82_h	Reset to factory settings
Activation	176_d	$B0_h$	Sensor illumination ON See Index 200, Table 7.17
Deactivation	177_d	$B1_h$	Sensor illumination OFF
UART Boot	180_d	$B4_h$	Start UART boot loader
Teach: on trace mode 4	192_d	$C0_h$	Trace width, trace amplitude, minimum contrast
Teach: angle compensation measurement	193_d	$C1_h$	Angle compensation teach
Teach: on trace mode 1	194_d	$C2_h$	Trace width only
Teach: on trace mode 2	195_d	$C3_h$	Minimum contrast only
Teach: on trace mode 3	196_d	$C4_h$	Trace amplitude only
Dark trace, light background	212_d	$D4_h$	
Light trace, dark background	213_d	$D5_h$	
Retro-reflective trace	214_d	$D6_h$	
Mode: "Trace width" filter ON	229_d	$E5_h$	
Mode: "Trace width" filter OFF	230_d	$E6_h$	
Mode: "Minimum contrast" filter ON	231_d	$E7_h$	
Mode: "Minimum contrast" filter OFF	232_d	$E8_h$	
Mode: "Trace amplitude" filter ON	233_d	$E9_h$	
Mode: "Trace amplitude" filter OFF	234_d	EA_h	
Delete angle compensation factors	240_d	$F0_h$	
Delete error	242_d	$F2_h$	Delete error bits / error status

Table 7.19: System commands

7.3 CANopen communication protocol

7.3.1 General information on CANopen

7.3.1.1 Topology

The CAN bus is a serial 2-wire bus system to which all participants are connected in parallel (i.e., using short stub cables). To avoid reflections, the bus must be terminated with a terminating resistor of 120 ohm at each end of the trunk line. Terminating resistors are also required for very short trunk line cable lengths.

7.3.1.2 Bus line (trunk line)

For CAN, the maximum cable length of the trunk line is predominantly limited by the signal propagation time. The multi-master bus-access process (arbitration) requires that the signals are present virtually simultaneously at all nodes/participants. Therefore, the cable length of the trunk cable must be adapted to the baud rate.

Baud rate	Bus length
1Mbit/s	< 20 m
500kbit/s	< 100 m
250kbit/s	< 250 m
125kbit/s	< 500 m
50kbit/s	< 1000 m
20kbit/s	< 2500 m

Table 7.20: CANopen bus length in relation to baud rate

7.3.1.3 Address assignment

NOTE

The participant-specific address for CANopen is also called the **Node ID**. Throughout this handbook, the term "address" is used, which is identical to Node ID.

Each participant connected to CANopen is assigned its own address (Node ID). Up to 127 participants can be connected to one network. The addresses range from 1 to 127. The address 0 is usually reserved for the CANopen master.

The node ID can be set in two ways:

- Via the object directory:

Index	Sub-index	Name	Description	Length [byte]	Data type
2001_h	[1_h]	*Can Node No*	CAN node address	2	uint16

- Via the **Layer Setting Services** function (LSS, see DS305 of CiA).

7.3.1.4 Baud rate setting

The OGS 600 supports the following baud rates:

- 1 Mbit/s
- 500 kBit/s
- 250 kBit/s
- 125 kBit/s
- 100 kBit/s
- 50 kBit/s
- 20 kBit/s
- 10 kBit/s

The OGS 600 is set to 1 Mbit/s by default.
The baud rate can be set in two ways:

- Via the object directory:

Index	Sub-index	Name	Description	Length [byte]	Data type
2001_h	[2_h]	*Can Baud rate*	CAN baud rate: 0 = 1 Mbit/s 1 = not used 2 = 500 kBit/s 3 = 250 kBit/s 4 = 125 kBit/s 5 = 100 kBit/s 6 = 50 kBit/s 7 = 20 kBit/s 8 = 10 kBit/s 9 = automatically by LSS	2	uint16

- Via the **Layer Setting Services** function (LSS, see DS305 of CiA).

7.3.1.5 Communication mechanisms of the OGS 600 in the CANopen network

In a CANopen network, all participants have in principle equal privileges. Each participant can initiate its data transmission independently. Here, the arbitration specified by the CIA controls the access of the individual participants to the network.

Generally, each CAN participant listens in on the bus. The transmission process is started only if the bus is not occupied by another CAN participant. When transmitting, the current bus status is always compared to the own transmitted frame.

Arbitration process

If several participants start a transmission simultaneously, the arbitration process decides which participant gains access to the network next. The individual participants are integrated into a prioritization scheme via their bus address and the type of data to be transmitted (index address of the data). Process data (PDOs) of a device are transmitted with a higher priority than, for example, variable objects (SDOs) of a device.

The node address of the participant is another criterion for prioritizing a participant in the network. The smaller the node address, the higher the priority of the participant in the network.

Since every participant compares its own priority with that of the other participants at the time of bus access, the participants with low priority discontinue their transmission activities immediately. The participant with the highest priority obtains temporary access to the bus. The arbitration process controls the access of all participants so that even participants with a low priority have access to the bus.

7.3.1.6 Objects

All process data and parameters are stored as objects in the OGS 600. The object directory (see Chapter 7.4) is the compilation of all process data and parameters of the OGS 600.

The object directory is structured in such a way that all objects are stored in the manufacturer-specific object area.

The objects are uniquely identified using an index addressing scheme. The structure of the object directory, the assignment of the index numbers, as well as some mandatory entries are specified in the CIA standard DS301 for CANopen.

7.3.1.7 EDS file

For the user, the object directory of the OGS 600 is available as an EDS file (Electronic Data Sheet).

↳ Download the EDS file for the device at **www.leuze.com**.

	NOTE
ℹ	**Download EDS file from the Internet!**
	↳ Call up the Leuze home page: **www.leuze.com**.
	↳ Enter the type designation or part number of the device as the search term.
	↳ The **EDS file** can be found on the product page for the device under the **Downloads** tab.

The EDS file contains all objects with index, sub-index, name, data type, default value, minimum and maximum, and access privileges.

The EDS file describes the complete functionality of the OGS 600.

7.3.1.8 SDOs and PDOs

The data exchange in CANopen distinguishes between **service data objects (SDOs)**, which are used for transmitting the service data (parameters) from and to the object directory, and **process data objects (PDOs)**, which are used to exchange the current process states.

SDOs

By using SDOs, all entries of the object directory can be accessed. Within one SDO call, only one object can be accessed at any one time. For this reason, a service data telegram must have a protocol structure which describes the exact target address by means of index and sub-index addressing. SDO telegrams place a part of the SDO addressing into the user data area. Eventually, a user data area with a width of 4 bytes out of the possible 8 bytes of user data remains for each SDO telegram.

The target address always responds to SDO transfers. In the following, the index and sub-index address of the OGS 600 parameters and variables can be found in the individual object descriptions.

PDOs

PDOs are objects (data, variables and parameters) from the object directory compiled (mapped) by the device manufacturer. A maximum of 8 bytes of user data from various objects can be mapped into one PDO.

A PDO can be received and evaluated by each participant (node). The model is referred to as the producer-consumer procedure.

Since there is no protocol structure in the telegram of a PDO, the participants in the network for whom these data are intended must know how the user data in the data area of the PDO are structured (which data are stored where in the user data area).

The exchange of process data is supported by the OGS 600 via the following accesses:

- **Event-controlled data transfer**
  Here, the data of a node are transmitted as a message whenever a change to the present state occurs.
- **Polling with remote frames**
  The CAN node which has been defined as master in the network requests the desired information via query (via remote frame). The participant which has this information (or the required data) responds by sending the requested data.
- **Synchronized mode**
  CANopen permits simultaneous querying of inputs and states of different participants and the simultaneous change of outputs or states. For this purpose, one uses the synchronization telegram (SYNC) transmitted by a master.
  The SYNC telegram is a broadcast to all network devices with high priority and without data content. Generally, the master sends the SYNC telegram cyclically. Participants working in synchronized mode read their data when receiving the SYNC message and then transmits them immediately after-

wards as soon as the bus permits this (see "Arbitration process" on page 41).
As the SYNC process can very quickly lead to high bus loads, another distinction is made between "event-controlled synchronization" and a "timer synchronization".

- Time-controlled transmission
  In this case, the transmission of a PDO is triggered when an adjustable time period has elapsed. The time-controlled transmissions are set individually for each PDO via the so-called "inhibit time" or an "event timer". The respective parameters can be found in the objects 1800_h to 1803_h for the corresponding PDOs.
- Node monitoring
  Heartbeat and guarding mechanisms are available for failure monitoring of the OGS 600. This is particularly important for CANopen, as the OGS 600 may not respond regularly in the event-controlled operating mode. In the case of guarding, the participants are cyclically queried for their state via data request telegrams (remote frame). In case of heartbeat, the nodes transmit their state themselves.
  Heartbeat and guarding/life time are standard communication objects from the DS301 CANopen specification. The corresponding objects here are:
  - Heartbeat 1017_h
  - Guarding/life time factor $100C_h$ and $100D_h$

7.3.1.9 Default 11 bit identifier

The OGS 600 sends an 11 bit identifier. 29 bit identifiers can be neither received nor sent by the OGS 600.

The node address (address of the OGS 600) is part of the 11 bit identifier. The default identifier and the node address give the COB ID, the value of which defines the prioritization in the arbitration.

	NOTE
i	Low-value identifiers have a higher priority in the arbitration.

Example:

If the same objects are queried in a CANopen network consisting of multiple OGS 600s, e.g. PDO1 (rx), then the sensor with the smallest node address has the highest priority in the arbitration.

The table below shows the value of the individual functions in the arbitration process of the CANopen.

According to the table, synchronization and emergency objects have the highest priority. This is followed by the PDOs; at the end of the prioritization are the SDOs.

11 bit identifier (binary)	Identifier Dec.	Identifier Hex.	Function
00000000000	0_d	0_h	Network management
00010000000	128_d	80_h	Synchronization
0001xxxxxxx	129_d … 255_d	81_h … FF_h	Emergency
0011xxxxxxx	385_d … 511_d	181_h … $1FF_h$	PDO1 (tx)
0100xxxxxxx	513_d … 639_d	201_h … $27F_h$	PDO1 (rx)
0101xxxxxxx	641_d … 767_d	281_h … $2FF_h$	PDO2 (tx)
0110xxxxxxx	769_d… 895_d	301_h … $37F_h$	PDO2 (rx)
0111xxxxxxx	897_d … 1023_d	381_h … $3FF_h$	PDO3 (tx)
1000xxxxxxx	1025_d … 1151_d	401_h … $47F_h$	PDO3 (rx)
1001xxxxxxx	1153_d … 1279_d	181_h … $4FF_h$	PDO4 (tx)
1010xxxxxxx	1281_d … 1407_d	501_h … $57F_h$	PDO4 (rx)
1011xxxxxxx	1409_d … 1535_d	581_h … $5FF_h$	Send SDO
1100xxxxxxx	1537_d … 1663_d	601_h … $67F_h$	Receive SDO
1110xxxxxxx	1793_d … 1919_d	701_h … $77F_h$	NMT Error Control
xxxxxxx = node address 1 - 127			

Table 7.21: 11 bit identifier

7.3.1.10 Object structure of the OGS 600

Overview of the CANopen-specific object area of the OGS 600

The following overview table shows the CANopen-specific communication objects from DS301 which are supported by the OGS 600. These operating instructions describe only the objects for which device-specific configurations can be performed. All other objects are standard objects of the CANopen specification. A description of these objects can be found in the DS301.

Object address in hex	CANopen-specific object area
1000_h	Device type
1001_h	Error register
1002_h	Manufacturer status
1003_h	Pre-defined error field
1005_h	COB ID SYNC
1006_h	SYNC cycle time
1008_h	Manufacturer device name
1009_h	Manufacturer hardware version
$100A_h$	Manufacturer software version
$100C_h$	Guard time
$100D_h$	Life-time factor
1010_h	Store parameter field
1011_h	Restore default parameters
1014_h	COB-ID emergency message
1016_h	Consumer heartbeat time
1017_h	Producer heartbeat time (necessary for heartbeat mechanism)
1018_h	Identity object (contains general information regarding the device)
1019_h	Synchronous counter overflow value
1029_h	Error behaviour

Table 7.22: Standard objects of CANopen specification CIA DS301

7.3.1.11 Process data objects

The OGS 600 provides 4 transmit process data objects (TPDOs) and 1 receive process data object (RPDO).

The TPDOs describe which objects are mapped to (integrated in) the TxPDO, and define the access (synchronous/asynchronous) to these objects.

- TPDO1: Status, contrast, number of detected traces, 1st edge and 2nd edge
- TPDO2: 3rd to 6th edge
- TPDO3: 7th to 10th edge
- TPDO4: 11th and 12th edge

The communication parameters of the PDOs are determined via defined objects. Synchronous or asynchronous access, a possible inhibit time for the PDO object in the CAN network as well as an event timer are defined in these objects.

- TPDO1: object address 1800_h
- TPDO2: object address 1801_h
- TPDO3: object address 1802_h
- TPDO4: object address 1803_h

Asynchronous transfer is controlled by the event timer in the PDOx property objects 1800_h to 1803_h.
Synchronous transfer is initiated by a SYNC telegram (80_h) sent from the master, as well as by the PDOx property objects 1800_h to 1803_h.

7.3.1.12 Overview of the mapped data in the TxPDOs

Different information can be mapped to the process data.
The default setting in Table 7.24 can be switched over to information in Table 7.25.
The TxPDO1 mapping is switched over using the System Command parameter (CAN Index 2000h).

Command	Value	Hex	Function / description
Type 2 process data	243_d	$F3_h$	Switchover from TxPDO1 to type 2
Type 4 process data	244_d	$f4_h$	Switchover from TxPDO1 to type 4 (default)

Table 7.23: System Command (CAN index 2000h)

Type 4 process data, output of all traces

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
					Trace 1			
	Low byte status	High byte status	Contrast	Number of traces	Left edge low byte	Left edge high byte	Right edge low byte	Right edge high byte
Mapped object	2020_h $[01_h]$		2030_h $[02_h]$	2021_h	2022_h $[01_h]$		2022_h $[02_h]$	

Table 7.24: TxPDO1, output of all traces

TxPDO1 contains general information about the sensor:

- Status: see object 2020_h $[1_h]$
- Contrast: see object 2030_h $[2_h]$
- Number of detected traces: see object 2021_h $[0_h]$

The first trace (1st left edge, 1st right edge) is also transmitted there: see object 2022_h $[1_h/2_h]$.
All subsequent traces (1 trace = 2 edges) are transmitted in TxPDO2 to TxPDO4. If not all traces are present, 0 is transmitted there.

TxPDO1

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
					Trace 1			
	Low byte status	High byte status	Contrast	Number of traces	Left edge low byte	Left edge high byte	Right edge low byte	Right edge high byte
Mapped object	2020_h $[1_h]$		2030_h $[2_h]$	2021_h	2022_h $[1_h]$		2022_h $[2_h]$	

TxPDO2

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
	Trace 2				Trace 3			
	Left edge low byte	Left edge high byte	Right edge low byte	Right edge high byte	Left edge low byte	Left edge high byte	Right edge low byte	Right edge high byte
Mapped object	2022_h [3_h]		2022_h [4_h]		2022_h [5_h]		2022_h [6_h]	

TxPDO3

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
	Trace 4				Trace 5			
	Left edge low byte	Left edge high byte	Right edge low byte	Right edge high byte	Left edge low byte	Left edge high byte	Right edge low byte	Right edge high byte
Mapped object	2022_h [7_h]		2022_h [8_h]		2022_h [9_h]		2022_h [A_h]	

TxPDO4

	Byte 0	Byte 1	Byte 2	Byte 3
	Trace 6			
	Left edge low byte	Left edge high byte	Right edge low byte	Right edge high byte
Mapped object	2022_h [B_h]		2022_h [C_h]	

Type 2 process data – output of edge located furthest left and furthest right (outer edges)

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
					Outer edges			
	Low byte status	High byte status	Contrast	Number of traces	Left edge low byte	Left edge high byte	Right edge low byte	Right edge high byte
Mapped object	2020_h [01_h]		2030_h [02_h]	2021_h	2033_h		2034_h	

Table 7.25: Output of edge located furthest left and furthest right (outer edges)

7.3.1.13 Overview of the mapped data in the RxPDO

The PD command is transmitted in the RxPDO.

PD-In1:

- 0: Switch not active
- 1: Trace 1 switch
- 2: Trace 2 switch
- 3: Trace 3 switch
- 4: Trace 4 switch
- 5: Trace 5 switch
- 6: Trace 6 switch

PD-In2: Reserve

RxPDO

	Byte 0	Byte 1
	PD-In1	PD-In2
Mapped object	2051_h [0_h]	

7.3.1.14 Overview TPDOs

The TPDOs describe which objects are mapped to (integrated in) the TxPDO, and define the access (synchronous/asynchronous) to these objects.

Object 1800_h TPDO1

Index (hex)	Sub-index (hex)	Name	Data type	Access	Default	Comment
1800_h	[1_h]	COB-ID for TPDO1	uint32	ro		180_h+ node ID
	[2_h]	Transmission type	uin8	rw	1	1 = synchronous
	[3_h]	Inhibit time	uint16	rw	0	Inhibit time
	[4_h]	Reserve				
	[5_h]	Event timer	uint16	rw		Event timer
	[6_h]	Sync start value	uint8	rw	0	Sync start value

Object 1A00$_h$ TPDO1

Index (hex)	Sub-index (hex)	Name	Data type	Access	Default	Comment
1A00$_h$	[1$_h$]	Status	uint32	ro	20200110$_h$	Content from object 2020$_h$ [1$_h$]
	[2$_h$]	Contrast PD	uint32	ro	20300208$_h$	Content from object 2030$_h$ [2$_h$]
	[3$_h$]	Number of traces	uint32	ro	20210008$_h$	Content from object 2021$_h$ [0$_h$]
	[4$_h$]	1st edge	uint32	ro	20220110$_h$	Content from object 2022$_h$ [1$_h$]
	[5$_h$]	2nd edge	uint32	ro	20220210$_h$	Content from object 2022$_h$ [2$_h$]

Object 1801$_h$ TPDO2

Index (hex)	Sub-index (hex)	Name	Data type	Access	Default	Comment
1801$_h$	[1$_h$]	COB-ID for TPDO2	uint32	ro		280$_h$+ node ID
	[2$_h$]	Transmission type	uin8	rw	254	254 = asynchronous
	[3$_h$]	Inhibit time	uint16	rw	0	Inhibit time
	[4$_h$]	Reserve				
	[5$_h$]	Event timer	uint16	rw		Event timer
	[6$_h$]	Sync start value	uint8	rw	0	Sync start value

Object 1A01$_h$ TPDO2

Index (hex)	Sub-index (hex)	Name	Data type	Access	Default	Comment
1A01$_h$	[1$_h$]	3rd edge	uint32	ro	20220310$_h$	Content from object 2022$_h$ [3$_h$]
	[2$_h$]	4th edge	uint32	ro	20220410$_h$	Content from object 2022$_h$ [4$_h$]
	[3$_h$]	5th edge	uint32	ro	20220510$_h$	Content from object 2022$_h$ [5$_h$]
	[4$_h$]	6th edge	uint32	ro	20220610$_h$	Content from object 2022$_h$ [6$_h$]

Object 1802_h TPDO3

Index (hex)	Sub-index (hex)	Name	Data type	Access	Default	Comment
1802_h	[1_h]	COB-ID for TPDO3	uint32	ro		380_h+ node ID
	[2_h]	Transmission type	uin8	rw	254	254 = asynchronous
	[3_h]	Inhibit time	uint16	rw	0	Inhibit time
	[4_h]	Reserve				
	[5_h]	Event timer	uint16	rw		Event timer
	[6_h]	Sync start value	uint8	rw	0	Sync start value

Object $1A02_h$ TPDO3

Index (hex)	Sub-index (hex)	Name	Data type	Access	Default	Comment
$1A02_h$	[1_h]	7th edge	uint32	ro	20220710_h	Content from object 2022_h [7_h]
	[2_h]	8th edge	uint32	ro	20220810_h	Content from object 2022_h [8_h]
	[3_h]	9th edge	uint32	ro	20220910_h	Content from object 2022_h [9_h]
	[4_h]	10th edge	uint32	ro	$20220A10_h$	Content from object 2022_h [A_h]

Object 1803_h TPDO4

Index (hex)	Sub-index (hex)	Name	Data type	Access	Default	Comment
1803_h	[1_h]	COB-ID for TPDO4	uint32	ro		480_h+ node ID
	[2_h]	Transmission type	uin8	rw	254	254 = asynchronous
	[3_h]	Inhibit time	uint16	rw	0	Inhibit time
	[4_h]	Reserve				
	[5_h]	Event timer	uint16	rw		Event timer
	[6_h]	Sync start value	uint8	rw	0	Sync start value

Object $1A03_h$ TPDO4

Index (hex)	Sub-index (hex)	Name	Data type	Access	Default	Comment
$1A03_h$	[1_h]	11th edge	uint32	ro	$20220B10_h$	Content from object 2022_h [B_h]
	[2_h]	12th edge	uint32	ro	$20220C10_h$	Content from object 2022_h [C_h]

7.3.1.15 Overview RPDOs

The RPDOs describe which objects are mapped to (integrated in) the RxPDO, and define the access (synchronous/asynchronous) to these objects.

Object 1400_h RPDO1

Index (hex)	Sub-index (hex)	Name	Data type	Access	Default	Comment
1400_h	[1_h]	COB-ID for RPDO1	uint32	ro		200_h+ node ID
	[2_h]	Transmission type	uin8	rw	255	255 = asynchronous
	[3_h]	Inhibit time	uint16	rw	0	Inhibit time
	[4_h]	Reserve				
	[5_h]	Event timer	uint16	rw		Event timer
	[6_h]	Sync start value	uint8	rw	0	Sync start value

Object 1600_h RPDO1

Index (hex)	Sub-index (hex)	Name	Data type	Access	Default	Comment
1600_h	[1_h]	PDO-CMD	uint32	ro	20510008_h	Content from object 2051_h [0_h]

7.4 CANopen object directory

Data types:		Access:	
string	Convert bytes into ASCII characters in incoming sequence	RW	Read Write
uint16	Arrangement: [LowByte, HighByte]	RO	Read Only
uint32	Arrangement: [LowByte, LowerByte, HigherByte, HighByte]	WO	Write Only
array_uint16	Arrangement: [LowByte1, HighByte1, LowByte2, HighByte2, …]		
int16	Arrangement: [LowByte, HighByte]		

Table 7.26: Object directory – data types and access

CAN index	CAN sub-index	Name	Description	Access	Comment	Default	Value range	Length [byte]	Data type
1000_h … 1029_h	See Table 7.22 "Standard objects of CANopen specification CIA DS301" on page 45								
2000_h	[0_h]	*System Command*	System command	WO	See Table 7.28			2	
2001_h	[1_h]	*CAN Node No*	CAN node address	RW	Address range: 0 … 127	10	0…127	2	uint16
2001_h	[2_h]	*Can Baud rate*	CAN baud rate	RW	0_d: 1 Mbit/s 1_d: not used 2_d: 500 kBit/s 3_d: 250 kBit/s 4_d: 125 kBit/s 5_d: 100 kBit/s 6_d: 50 kBit/s 7_d: 20 kBit/s 8_d: 10 kBit/s	0	0…8	2	uint16
2002_h	[0_h]	*UserMode*	UserMode	RW	Bit 0: 1 = dark trace; 0 = light trace Bit 1: angle compensation active Bit 2: filter: trace width Bit 3: filter: contrast Bit 4: filter: amplitude Bit 5: teach trace width Bit 6: teach contrast Bit 7: teach amplitude Bit 8: retro-reflective trace	Bit 0= 1	0… 65535	2	uint16
2003_h	[1_h]	*Q1UpperSwitching Point*	Upper switching point for switching output SW_IO (pin 4)	RW	With trace monitoring: range: 0…3000 (long device version), range: 0…1500 (short device version), unit: 0.1 mm With contrast monitoring: range: 0…21200, unit: [LSB]	0	0… 65535	2	uint16
2003_h	[2_h]	*Q1LowerSwitching Point*	Lower switching point for switching output SW_IO (pin 4)	RW	With trace monitoring: range: 0…3000 (long device version), range: 0…1500 (short device version), unit: 0.1 mm With contrast monitoring: range: 0…21200, unit: [LSB]	0	0… 65535	2	uint16
2003_h	[3_h]	*Q1LightDark*	Switching behavior Light/dark switching for switching output SW_IO (pin 4)	RW	0_d: Q = high outside switching points, 1_d: Q = high inside switching points, see Table 5.1	0	0…1	2	uint16
2003_h	[4_h]	*Q1SwitchPtMode*	Switching point mode for switching output SW_IO (pin 4)	RW	0_d: switching output deactivated 1_d: trace monitoring 2_d: contrast monitoring	0	0…2	2	uint16

CAN index	CAN sub-index	Name	Description	Access	Comment	Default	Value range	Length [byte]	Data type
2003_h	[5_h]	*Q1Hysteresis*	Switching hysteresis for switching output SW_IO (pin 4)	RW	With trace monitoring: range: 0…3000 (long device version), range: 0…1500 (short device version), unit: 0.1 mm With contrast monitoring: range: 0…21200, unit: [LSB]	20	0… 65535	2	uint16
2003_h	[6_h]	*Q1UserConfig*	Configuration of switching output SW_IO (pin 4)	RW	0_d: not active 1_d: Out_PP (push-pull) 2_d: Out_NPN 3_d: Out_PNP	0	0…3	2	uint16
2004_h	[1_h]	*Q2UpperSwitching Point*	Upper switching point for switching output IO (pin 2)	RW	With trace monitoring: range: 0…3000 (long device version), range: 0…1500 (short device version), unit: 0.1 mm With contrast monitoring: range: 0…21200, unit: [LSB]	0	0… 65535	2	uint16
2004_h	[2_h]	*Q2LowerSwitching Point*	Lower switching point for IO switching output (pin 2)	RW	With trace monitoring: range: 0…3000 (long device version), range: 0…1500 (short device version), unit: 0.1 mm With contrast monitoring: range: 0…21200, unit: [LSB]	0	0… 65535	2	uint16
2004_h	[3_h]	*Q2LightDark*	Switching behavior Light/dark switching for switching output IO (pin 2)	RW	0_d: Q = high outside switching points, 1_d: Q = high inside switching points, see Table 5.1	0	0…1	2	uint16
2004_h	[4_h]	*Q2SwitchPtMode*	Switching point mode for switching output IO (pin 2)	RW	0_d: switching output deactivated 1_d: trace monitoring 2_d: contrast monitoring	0	0…2	2	uint16
2004_h	[5_h]	*Q2Hysteresis*	Switching hysteresis for IO switching output (pin 2)	RW	With trace monitoring: range: 0…3000 (long device version), range: 0…1500 (short device version), unit: 0.1 mm With contrast monitoring: range: 0…21200, unit: [LSB]	20	0… 65535	2	uint16
2004_h	[6_h]	*Q2UserConfig*	Configuration of IO switching output/switching input (pin 2)	RW	0_h: not active 1_h: Out_PP (push-pull) 2_h: Out_NPN 3_h: Out_PNP 104_h: In_NPN deactivation input 105_h: In_PNP deactivation input 304_h: In_NPN activation input 305_h: In_PNP activation input	0	0… 65535	2	uint16
2005_h	[0_h]	*Qproperty*	Output behavior with no measurement value	RW	0_d: off, 1_d: on, 2_d: unchanged, applies to both outputs	0	0…2	2	uint16

CAN index	CAN sub-index	Name	Description	Access	Comment	Default	Value range	Length [byte]	Data type
2006_h	$[0_h]$	*Serial Number*	Device serial number	RO	<Serial number>			16	string
2007_h	$[0_h]$	*Product ID*	Part no. of the device	RO	<Part number>			16	string
2010_h	$[1_h]$	*TraceWidthMax*	Maximum trace width	RW	For manual configuration (changed by means of a trace width teach!), unit: 0.1 mm	490	0… 65535	2	uint16
2010_h	$[2_h]$	*TraceWidthMin*	Minimum trace width	RW	For manual configuration (changed by means of a trace width teach!), unit: 0.1 mm	290	0… 65535	2	uint16
2010_h	$[3_h]$	*TraceWidthTol*	Trace width tolerance	RW	Only required for teach, unit: 0.1 mm.	100	0… 65535	2	uint16
2010_h	$[4_h]$	*TraceContrastMin*	Minimum contrast	RW	Unit: [LSB]	5500	0… 65535	2	uint16
2010_h	$[5_h]$	*TraceContrastWarning*	Contrast warning threshold in %	RW	Unit: %	20	1…100	2	uint16
2010_h	$[6_h]$	*TraceContrastTol*	Contrast tolerance	RW	Only required for teach, unit: [LSB]	30	0… 65535	2	uint16
2010_h	$[7_h]$	*TraceAmplitudeMin*	Minimum amplitude	RW	Unit: [LSB]	2500	0… 65535	2	uint16
2010_h	$[8_h]$	*TraceAmplitudeWarning*	Amplitude warning threshold in %	RW	Unit: %	20	1…100	2	uint16
2010_h	$[9_h]$	*TraceAmplitudeTol*	Amplitude tolerance for teach	RW	Only required for teach, unit: [LSB]	1000	0… 65535	2	uint16
2010_h	$[A_h]$	*UserOffset*	Offset for process data output	RW	PD output value = edge position + offset	0	-32768 … 32767	2	int16
2010_h	$[B_h]$	*SwitchTraceWidthFactor*	Trace width factor for switch function	RW	Factor for widening trace with active switch function (see index 170_d), unit: %	150	0… 65535	2	uint16
2010_h	$[C_h]$	*SwitchDeviationThr*	Lower limit value for deviation with switch	RW	Used with active switch function, unit: [LSB]	250	0… 65535	2	uint16
2010_h	$[D_h]$	*TraceTeachThr*	Threshold which is taught	RW	Unit: [LSB]	7000	0… 65535	2	uint16
2011_h	$[2_h]$	*UserState*	Status	RO	Bit 0 = 1: angle compensation OK Bit 1 = 1: trace teach OK	0	0… 65535	2	uint16

CAN index	CAN sub-index	Name	Description	Access	Comment	Default	Value range	Length [byte]	Data type
2012_h	[0_h]	*SwitchNumber*	Switch function	RW	Activation of switch function for guide trace: 0_d: Switch function not active 1_d: Switch function active for guide trace 1 2_d: Switch function active for guide trace 2 3_d: Switch function active for guide trace 3 4_d: Switch function active for guide trace 4 5_d: Switch function active for guide trace 5 6_d: Switch function active for guide trace 6	0	0…6	2	uint16
2020_h	[1_h]	*Status*	Sensor state	RO	Bit 0: Global error Bit 1: Compensation factors valid Bit 2: Teach, compensation measurement running Bit 3: Trace contrast warning Bit 4: Trace amplitude warning Bit 5: Trace width error Bit 6: Contrast error Bit 7: Amplitude error Bit 8: Supply voltage warning Bit 9: Supply voltage error Bit 10: Teaching error Bit 11: Compensation error Bit 12: Switch function active Bit 13: Switch error: unknown trace Bit 14: No trace detected (number of edges < 2)	0	0… 65535	2	uint16
2020_h	[2_h]	*Error*	Error description	RO	Bit 0: Teach: compensation values missing Bit 1: Teach: valid traces > 1; invalid traces; switch active Bit 2: Angle compensation: compensation values missing Bit 3: Angle compensation: trace or edge detected Bit 4: Hardware error: Measurement interrupt error Bit 5: Supply voltage warning Bit 6: Supply voltage error Bit 7: Switch: unknown trace	0	0… 2^{32}-1	4	uint32
2021_h	[0_h]	*TraceValidNum*	Valid traces: number	RO	Value: 0 … 6	0	0…6	2	uint16
2022_h	[1_h]…[C_h]	*TraceValidSubPixel*	Valid traces: sub-pixels in mm	RO	Contains edge positions of valid traces, unit: [mm] (see Chapter 8.8)	0	0… 65535	24	array _uint16
2023_h	[1_h]…[C_h]	*TraceValidAmp*	Valid traces: amplitude	RO	Contains amplitude of environment and of valid trace, unit: [LSB] (see Chapter 8.8)	0	0… 65535	24	array _uint16
2024_h	[1_h]…[C_h]	*TraceValidThreshold*	Valid traces: threshold	RO	Contains threshold for edge position of every detected trace, unit: [LSB]	0	0… 65535	24	array _uint16

CAN index	CAN sub-index	Name	Description	Access	Comment	Default	Value range	Length [byte]	Data type
2025_h	$[1_h]…[6_h]$	*TraceValidStatus*	Valid traces: status	RO	Status is signaled for every valid trace: Bit 0: Contrast warning Bit 1: Trace amplitude warning (see Chapter 8.8)	0	0… 65535	12	array _uint16
2026_h	$[0_h]$	*TraceInvalidNum*	Invalid traces: number	RO	Value: 0 … 6	0	0…6	2	uint16
2027_h	$[1_h]…[C_h]$	*TraceInvalidSubPixel*	Invalid traces: subpixels in mm	RO	Contains edge positions of invalid traces, unit: [mm] (see Chapter 8.8)	0	0… 65535	24	array _uint16
2028_h	$[1_h]…[C_h]$	*TraceInvalidAmp*	Invalid traces: amplitude	RO	Contains amplitude of environment and of invalid trace, unit: [LSB] (see Chapter 8.8)	0	0… 65535	24	array _uint16
2029_h	$[1_h]…[6_h]$	*TraceInvalidStatus*	Invalid traces: status	RO	Status is signaled for every invalid trace: Bit 0: Contrast error Bit 1: Trace amplitude error Bit 1: Trace width error (see Chapter 8.8)	0	0… 65535	12	array _uint16
2030_h	[01]	*Contrast*	Minimum contrast of all traces	RO	Unit: [LSB]	0	0… 65535	2	uint16
2031_h	[01]	*SupplyVoltage*	Supply voltage	RO	Unit: [mV]	0	0… 65535	2	uint16
2031_h	[02]	*TempController*	Temperature controller	RO	Unit: [°C]	0	0… 65535	2	uint16
2032_h	$[0_h]$	TraceSensitivity	Sensitivity of trace detection	RW	50 = high sensitivity	100	50… 1000	2	uint16

Table 7.27: CANopen object directory

7.4.1 CANopen system commands

Commands can be sent to the sensor via CAN index 2000_h *System Command*.

Command	Value		Function / description
	Dec.	Hex	
Device Reset	128_d	80_h	Software reset
Factory reset	130_d	82_h	Reset to factory settings
Activation	176_d	$B0_h$	Sensor illumination ON
Deactivation	177_d	$B1_h$	Sensor illumination OFF
UART Boot	180_d	$B4_h$	Start UART boot loader
Teach: on trace mode 4	192_d	$C0_h$	Trace width, trace amplitude, minimum contrast
Teach: angle compensation measurement	193_d	$C1_h$	Angle compensation teach
Teach: on trace mode 1	194_d	$C2_h$	Trace width only
Teach: on trace mode 2	195_d	$C3_h$	Minimum contrast only
Teach: on trace mode 3	196_d	$C4_h$	Trace amplitude only
Dark trace, light background	212_d	$D4_h$	
Light trace, dark background	213_d	$D5_h$	
Retro-reflective trace	214_d	$D6_h$	
Mode: "Trace width" filter ON	229_d	$E5_h$	
Mode: "Trace width" filter OFF	230_d	$E6_h$	
Mode: "Minimum contrast" filter ON	231_d	$E7_h$	
Mode: "Minimum contrast" filter OFF	232_d	$E8_h$	
Mode: "Trace amplitude" filter ON	233_d	$E9_h$	
Mode: "Trace amplitude" filter OFF	234_d	EA_h	
Delete angle compensation factors	240_d	$F0_h$	
Delete error	242_d	$F2_h$	Delete error bits / error status

Table 7.28: System commands

7.5 Performing a reset on the OGS 600

Two different resets can be performed by means of system commands:

- The **device reset** restarts the software of the OGS 600.
  All settings are retained.
- The **factory reset** resets all internal settings of the device to the factory settings. This includes all indices as well as settings determining which trace type is active and which filters are active.

8 Configuring the sensor – Overview of functions

8.1 Compensating the installation position of the sensor – Angle compensation teach

After installation of the device, it is recommended to perform a one-off compensation teach in order to compensate the installation position. The compensation teach is especially recommended if the contrast between the trace and environment to be evaluated is extremely low. A low contrast is a value less than 5000 LSB.

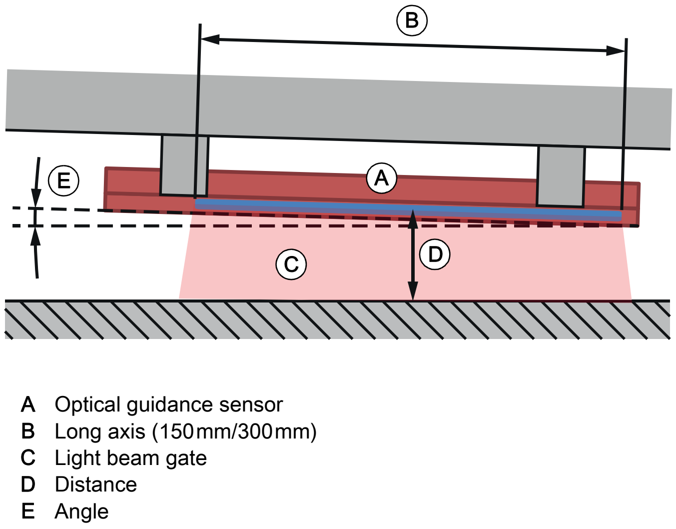


Figure 8.1: Angle compensation teach for compensation of the installation position

Procedure

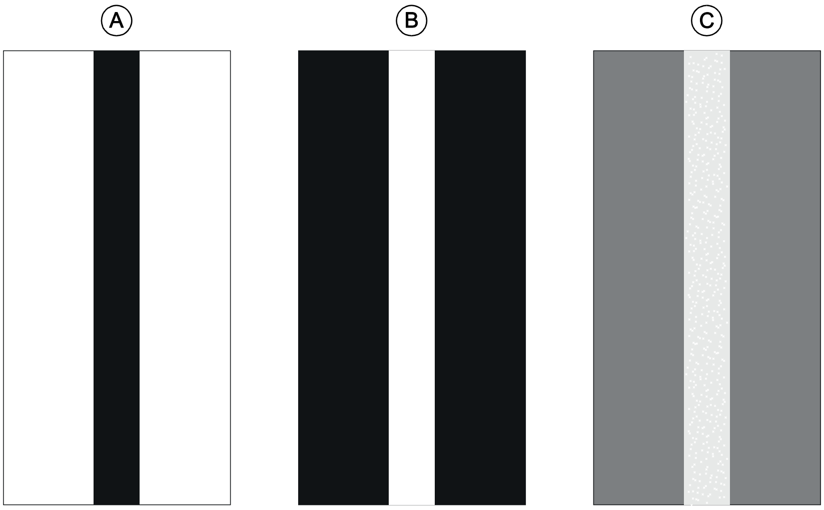
1. The sensor must be pointing at a light, homogeneous object.
   A sheet of white paper is ideal for this purpose.
2. Perform the teach –> *System Command* (UART index 2_d and CAN index 2000_h [0_h], value: 193_d).
3. Read *UserState* (UART index 151_d and CAN index 2011_h [2_h]),
   evaluation of the data –> wait until bit 1 (angle compensation OK) is set.

8.2 Configuring the guide trace – light, dark, retro-reflective

The sensor must be preconfigured for the type of trace to be detected.

The following models exist:

- Dark guide trace on light background
- Light guide trace on dark background
- Retro-reflective guide trace



A Dark guide trace on light background
B Light guide trace on dark background
C Retro-reflective guide trace

Figure 8.2: Guide trace types

Retro-reflective guide trace

The retro-reflective guide trace is a special variant of the light guide trace on a dark background. The amount of light reflected by the retro-reflective medium is greater than the amount of light reflected by the background. For the sensor, this signal looks like a light trace.

With this setting, the transmitting current of the LEDs for the sensor illumination is reduced in order to fully utilize the dynamics of the electronics.

Configuration of the guide trace type

Name	Index UART	Index [sub-index] CANopen	Index length [byte]	Access	Data [Dec.]	Function / value
Dark trace type	2_d	2000_h [0_h]	2	W	212_d	Dark trace, light background
Light trace type	2_d	2000_h [0_h]	2	W	213_d	Light trace, dark background
Retro-reflective trace	2_d	2000_h [0_h]	2	W	214_d	Retro-reflective trace

Table 8.1: Configuration of the guide trace type

Querying of the guide trace type

Name	Index UART	Index [sub-index] CANopen	Index length [byte]	Access	Data	Function / value
UserMode	75_d	2002_h [0_h]	2	R		Bit 0: 0 = light trace; 1 = dark trace Bit 8: 0 = inactive, 1 = retro-reflective trace

Table 8.2: Querying of the guide trace type

8.3 Offset to the edge positions

An offset can be added to the edge output values. This offset only has an effect on the output of process data.

	NOTE
ℹ	If indices with edge positions are read out, they do **not** contain the offset.

The offset can be used to compensate off-center installation of the sensor.

Name	Index UART	Index [sub-index] CANopen	Index length [byte]	Access	Data [Dec.]	Function / value
UserOffset	109_d	2010_h [A_h]	2	RW	212_d	Offset for edge position Unit: [mm * 10]

Example:

Offset the process data output values from 0 … 3000 to -1500 … 1500.

-150 mm offset: -150 mm * 10 = -1500.
–> Write the value '-1500' in *UserOffset* (UART index 109_d and CAN index 2010_h [A_h]).

	NOTE
ℹ	If the offset is to be deactivated, the value 0 must be written.

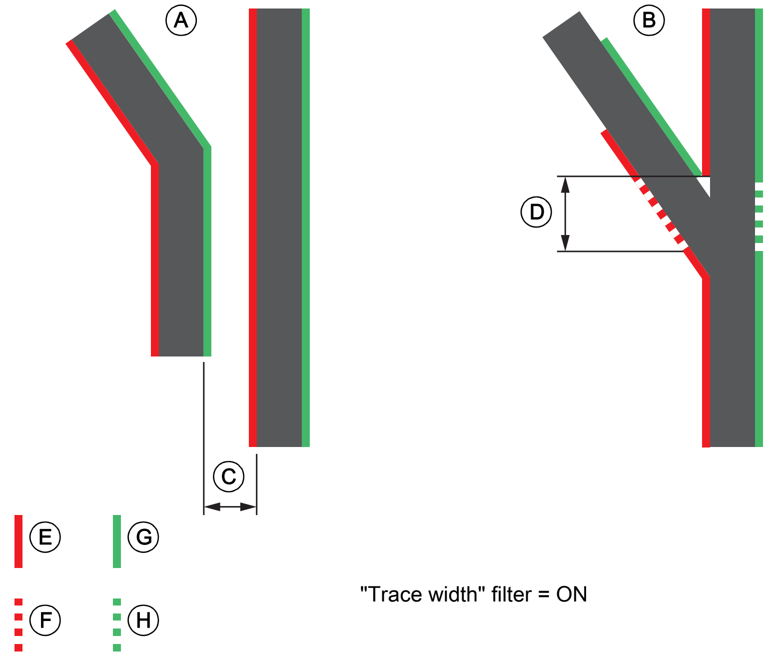
8.4 Switch

At a switch, the sensor always outputs all detected traces.

	NOTE
ℹ	The users themselves must decide in which direction they want to turn.

There are two different switch types (see figure 8.3):

- Switch **type 1** with parallel guide trace
- Switch **type 2** with branching guide trace



A Switch **type 1**
B Switch **type 2**
C Minimum distance 30 mm
D Midpoint of switch
E Left edge of guide trace
F Left edge of guide trace too wide
G Right edge of guide trace
H Right edge of guide trace too wide

Figure 8.3: Type 1 and type 2 trace switches

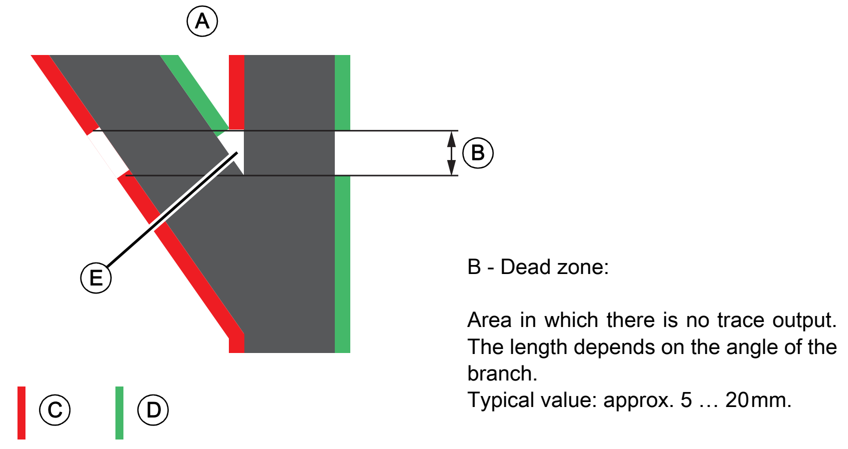
The sensor supports both switch types.

	NOTE
ℹ	One switch can have three branches.

Switch type 2

The behavior at the midpoint of a type 2 switch depends on the trace width filter and the angle of the branch.

For type 2 switches, it is recommended to use the "Switch" function (see chapter 8.4.1 ""Switch" function – Settings for type 2 switches") in order to improve detection of the wide trace at the midpoint of the switch, as well as to receive two trace values in the output as soon as possible in the triangular, extremely low-contrast area after the switch.



A Midpoint of a **type 2** switch
B Dead zone
C Left edge of guide trace
D Right edge of guide trace
E Area with extremely low contrast

Figure 8.4: Midpoint of type 2 switch

8.4.1 "Switch" function – Settings for type 2 switches

NOTE
The "Switch" function changes several settings in the sensor. **These changes are only required for type 2 switches.**

Activation of the *SwitchNumber* function (UART index 170_d and CAN index 2012_h) affects the filters as follows:

- "Minimum contrast" filter is deactivated.
- "Trace width" filter remains active/inactive -> adjustment of *TraceWidthMax*
- "Minimum contrast" filter remains active/inactive

"Trace width" filter

If the "Trace width" filter is used, the maximum trace width of the filter is increased. The minimum trace width remains unchanged.

The *SwitchTraceWidthFactor* factor (UART index 110_d and CAN index 2010_h [B_h]) is used for calculating the new maximum trace width.

The calculation temporarily changes the *TraceWidthMax* parameter (UART index 100_d and CAN index 2010_h [1_h]) until the *SwitchNumber* switch function is deactivated.

The *SwitchTraceWidthFactor* factor is preset by default for a type 2 switch with one branch. For switches with 2 branches (3 traces), the preset factor may be too small and may need to be increased.

Calculation of the maximum trace width when the switch function is activated:

TraceWidthMax_Switch = *TraceWidthMax* + (*TraceWidthMax* * *SwitchTraceWidthFactor* / 100)

The result of the calculation can be checked in *TraceWidthMax*. After deactivation of the *SwitchNumber* function, the original value is entered in *TraceWidthMax*.

NOTE
If problems with the "Trace width" filter occur at a switch, the *SwitchTraceWidthFactor* factor can be increased or reduced. The change is retained after a voltage reset of the device. Resetting to the factory settings (system command *Factory Reset*) restores the original value.

Why write trace number?

To ensure that the dead zone (see Figure 8.4) is as small as possible, internal parameters for the guide trace are set on activation of the switch function when the first measurement is performed after the query.

If the sensor detects more than one correct guide trace in the measurement cycle performed during activation, then these traces are output via the process data.

The vehicle decides which guide trace is used. The sensor does not know the decision made by the vehicle.

To allow optimum configuration to be performed, the sensor must be informed of the number of the guide trace which is followed by the vehicle.

The trace number is derived from the sequence in which the trace is output in the process data (see Table 7.11).

If the trace number used by the vehicle control unit changes during an active switch function or the second trace disappears, then the currently used trace number is transferred to the sensor. This does not cause the internal settings to change. These settings are only changed again when the switch function is deactivated by writing a '0' and then reactivated.

	NOTE
ℹ	If a trace number which is not present is written, an error occurs. In this case, bit 13 is set in the *Status* index (UART index 200_d and CAN index 2020_h). The switch function is not activated. The switch function is active if bit 12 is set in the *Status* index (UART index 200_d and CAN index 2020_h). Solution: Write the correct trace number.

Operational sequence of the "Switch" function

When should the *SwitchNumber* function be activated?

1. The system informs the vehicle that there is a switch ahead.
   Ideally, this happens 10 … 200 mm before the sensor reaches the midpoint of the switch and the trace becomes wider.
2. The vehicle notes which guide trace it is currently following. The guide traces are numbered in ascending order from 1 to 6.
   The sequence is derived from the sequence in which the edges are output in the process data (see Table 7.11).
3. This trace number must be written to the *SwitchNumber* index (UART index 170_d and CAN index 2012_h [0_h]) or sent via the query with the process data with byte 2 PD-In1.
4. Internal adaptation of the values in the sensor to the trace which the vehicle is currently following takes place once only.
   The effect on the output traces becomes apparent within the maximum time of one measurement cycle (10 ms) after the query is sent for the first time.

Deactivation of the "Switch" function

1. Writing a ’0’ in *SwitchNumber* (UART index 170_d and CAN index 2012_h [0_h])
   or
2. Writing of ’0’ in byte 2 PD-In1 during the process data query.

8.4.2 Index accesses for activation of the "Switch" function

Name	Index UART	Index [sub-index] CANopen	Index length [Byte]	Access	Default value data [Dec.]	Info
SwitchNumber	170_d	2012_h [0_h]	2	W	0	0_d = inactive 1_d = trace no. 1 2_d = trace no. 2 3_d = trace no. 3 4_d = trace no. 4 5_d = trace no. 5 6_d = trace no. 6
SwitchTrace-WidthFactor	110_d	2010_h [B_h]	2	RW	150	Factor in % for increase in *Trace-WidthMax* parameter when switch function is activated

Table 8.3: Index accesses for activation of the "Switch" function

	Type	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4
		Node no./ identifier	PD type	PD-In1 (data in)	PD-In2 (reserve)	CRC
Query	PD	13_h	04_h	0_d = off 1_d = trace no. 1 2_d = trace no. 2 3_d = trace no. 3 4_d = trace no. 4 5_d = trace no. 5 6_d = trace no. 6	0_h	CRC

Table 8.4: Settings for "Switch" function with process data query in byte 2

8.5 "Trace width" filter

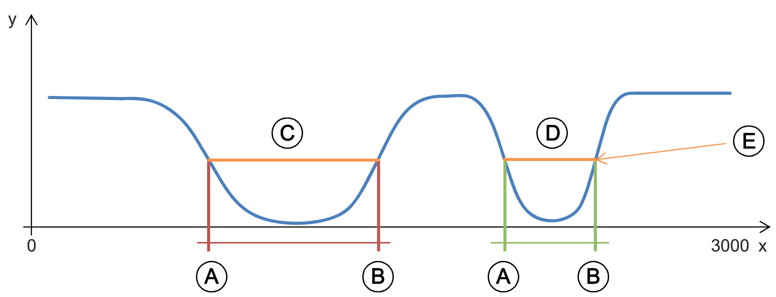
If the sensor is to output only traces which correspond to a certain trace width, the "Trace width" filter can be activated.

The filter value can be set to the trace by means of a teach or manually by entering the values in the corresponding indices.

Traces filtered out by the filter can be read out via the *TraceInvalidSubPixel* index (UART index 213_d and CAN index 2027_h $[1_h]...[C_h]$).

During the trace width teach, the *TraceTeachThr* parameter is calculated. The position of the left and right edge is determined for the amplitude of this threshold. If a trace is detected which does not allow the trace width to be calculated with the threshold determined during the teach, then the threshold is adapted for this particular trace. The taught threshold is used as soon as the amplitude of the found background-trace combination permits this.

⚠ CAUTION!
The trace width depends on the value of this threshold.



x Edge position
y Reception signal amplitude
A Left edge of guide trace
B Right edge of guide trace
C Trace too wide according to active "Trace width" filter
D Trace OK
E Teach threshold

Figure 8.5: Application of the "Trace width" filter using a dark trace as an example

8.5.1 Teaching the trace width

The *TraceWidthTol* parameter (UART Index 102_d and CAN Index 2010_h $[3_h]$) is used during the teach in order to define the upper and lower limit for the trace width on the basis of the currently measured trace width.

Calculation of the values in the sensor:

Trace width = | Position of left edge - Position of right edge |
TraceWidthMax = Trace width + *TraceWidthTol*
TraceWidthMin = Trace width - *TraceWidthTol*

8.5.2 Manual configuration of the trace width

If the trace width is to be configured manually, the values can be written directly to the parameters *TraceWidthMax* (UART index 100_d and CAN index 2010_h [1_h]) and *TraceWidthMin* (UART index 101_d and CAN index 2010_h [2_h]).
Remember to apply the factor 10 during conversion: 10_d corresponds to 1 mm.

	⚠ CAUTION!
⚠	If a trace width teach is performed, manually configured trace width values are overwritten.

8.5.3 Process data information for the "Trace width" filter

If the number of detected traces which, owing to the "Trace width" filter, are not output in the process data is greater or equal to one, then bit 3 is set in process data byte 2 *Status PD*.

8.5.4 Index overview for the "Trace width" filter

Bit counting method: bit0 … bit15

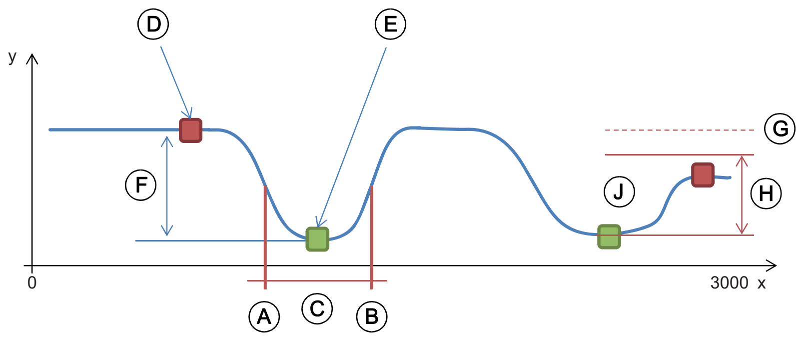
Name	Index UART	Index [sub-index] CANopen	Index length [byte]	Access	Data / (default value)	Info
Activation of "Trace width" filter	2_d	2000_h [0_h]	2	W	229_d	System command
Deactivation of "Trace width" filter	2_d	2000_h [0_h]	2	W	230_d	System command
Trace width teach	2_d	2000_h [0_h]	2	W	194_d	System command
TraceWidthMax	100_d	2010_h [1_h]	2	RW	(490_d)	Maximum trace width For manual configuration or result from teach Value: [mm * 10]
TraceWidthMin	101_d	2010_h [2_h]	2	RW	(290_d)	Minimum trace width For manual configuration or result from teach Value: [mm * 10]
TraceWidthTol	102_d	2010_h [3_h]	2	RW	(100_d)	Trace width tolerance Only required for teach Value: [mm * 10]
TraceTeachThr	112_d	2010_h [D_h]	2	R		Determined during teach
Status	200_d	2020_h [1_h]	2	R	Bit no. 5	If the number of filtered-out traces is ≥ 1, the bit is set. See also process data status byte, bit no. 3 (Chapter 7.1.4.1)
UserMode	75_d	2002_h [0_h]	2	R	Bit no. 2	If the bit is set, then the "Trace width" filter is active.

Table 8.5: Index accesses for the "Trace width" filter

8.6 "Minimum contrast" filter

The filter for the minimum contrast queries whether the brightness of the background and the brightness of the trace have a minimum difference.

This minimum difference can be taught based on a reference trace, or the value can be set manually.



x Edge position
y Reception signal amplitude
A Left edge of guide trace
B Right edge of guide trace
C Trace width
D Amplitude of background
E Amplitude of guide trace
F Contrast
G Warning threshold for minimum contrast
H Minimum contrast (*TraceContrastMin)*
J Verification of minimum contrast.
The second, right trace is not output because the minimum contrast is not fulfilled.

Figure 8.6: Application of the "Minimum contrast" filter using a dark trace as an example

8.6.1 Teaching the minimum contrast

The value of the *TraceContrastTol* parameter (UART index 105_d and CAN index 2010_h [6_h]) is used for calculating a minimum threshold for the contrast using the contrast value measured during the teach. The value appears as a percentage [%] in the index.

Calculation in the sensor:

Contrast = | Amplitude_of_environment - Amplitude_of_trace |
TraceContrastMin = Contrast - (Contrast * *TraceContrastTol* /100)

8.6.2 Manual configuration of the minimum contrast

If the minimum contrast is to be configured manually, it can be written directly to the *TraceContrastMin* parameter (UART index 103_d and CAN index 2010_h [4_h]) as a value in [LSB].

⚠ CAUTION!
If a minimum contrast teach is performed, a manually configured minimum contrast value is overwritten.

8.6.3 Warning for minimum contrast

The warning threshold corresponds to a percentage deviation from the minimum contrast *TraceContrastMin* (UART index 103_d and CAN index 2010_h [4_h]). The warning threshold for the minimum contrast is calculated using the *TraceContrastWarning* factor (UART index 104_d and CAN Index 2010_h [5_h]). There is no index for calling up this value directly.

Calculation:

TraceContrastWarning_threshold = *TraceContrastMin* + (*TraceContrastMin* * *TraceContrastWarning*)

8.6.4 Process data information for the "Minimum contrast" filter

In the status byte of the process data, there are two bits for information relating to the minimum contrast:

- Bit 1: *Minimum contrast warning*
- Bit 4: *Minimum contrast error*

Bit 1 *Minimum contrast warning* is set if the number of detected traces for which the minimum contrast is less then the warning threshold, is greater than or equal to one.

Bit 4 *Minimum contrast error* is set if the number of detected traces for which the contrast is less than *TraceContrastMin*, is greater than or equal to one.

8.6.5 Index overview for the "Minimum contrast" filter

Bit counting method: bit0 … bit15.

Name	Index UART	Index [sub-index] CANopen	Index length [byte]	Ac-cess	Data / (de-fault value)	Info
Activate filter	2_d	2000_h [0_h]	2	W	231_d	System command
Deactivate filter	2_d	2000_h [0_h]	2	W	232_d	System command
Teaching the minimum contrast	2_d	2000_h [0_h]	2	W	195_d	System command
TraceContrast-Min	103_d	2010_h [4_h]	2	RW	(5500_d)	Result from teach or manual entry, unit: [LSB]
TraceContrast-Warning	104_d	2010_h [5_h]	2	RW	(20_d)	Factor for calculation of warning threshold, unit [%]
TraceContrastTol	105_d	2010_h [6_h]	2	RW	(30_d)	Tolerance is used in teach event, unit: [%]
Status	200_d	2020_h [1_h]	2	R	Bit no. 6	1 = Minimum contrast error See also process data status byte, bit no. 4 (Chapter 7.1.4.1)
Status	200_d	2020_h [1_h]	2	R	Bit no. 3	1 = Minimum contrast warning See also process data status byte, bit no. 1 (Chapter 7.1.4.1)
UserMode	75_d	2002_h [0_h]	2	R	Bit no. 3	If bit = 1, then "Minimum contrast" filter is active.

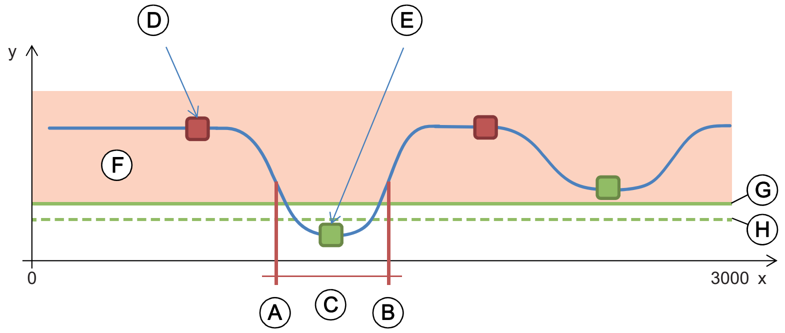
Table 8.6: Index accesses for the "Minimum contrast" filter

8.7 "Trace amplitude" filter

The filter is based on the assumption that the guide trace tape processed in a system is the same throughout. If this is the case, then the guide trace tape is a known constant. As a result, all other markings with a different amplitude can be filtered out.

It is therefore recommended to select a guide trace tape which is as light as possible (white) or as dark as possible (black) so that there are no markings which are darker or lighter than the optical trace.

The filter for trace amplitude is the limit value *TraceAmplitudeMin* (UART index 106_d and CAN index 2010_h [7_h]) which marks as incorrect all traces for which the amplitude of the trace signal is greater than the limit value. There is a warning threshold which is set using the *TraceAmplitudeWarning* parameter (UART index 107_d and CAN index 2010_h [8_h]).



x Edge position
y Reception signal amplitude
A Left edge of guide trace
B Right edge of guide trace
C Trace width
D Amplitude of background
E Amplitude of guide trace
F Invalid trace amplitudes (red area)
G *TraceAmplitudeMin* value
H *TraceAmplitudeWarning* value

Figure 8.7: Application of the "Trace amplitude" filter using a dark trace as an example

8.7.1 Teaching the trace amplitude

The *TraceAmplitudeTol* value (UART index 108_d and CAN index 2010_h [9_h]) is used for setting the *TraceAmplitudeMin* limit value (UART index 106_d and CAN index 2010_h [7_h]) for the "Trace amplitude" filter during the teach.

Calculation of **dark guide trace**:

TraceAmplitudeMin = Amplitude_trace [LSB] + *TraceAmplitudeTol* [LSB]

Calculation of **light guide trace**:

TraceAmplitudeMin = Amplitude_trace [LSB] - *TraceAmplitudeTol* [LSB]

8.7.2 Manual configuration of the trace amplitude

If the trace amplitude limit value is to be configured manually, it can be written directly to the *TraceAmplitudeMin* parameter (UART index 106_d and CAN index 2010_h [7_h]) as a value in [LSB].

⚠ CAUTION!	
⚠	If a trace amplitude teach is performed, a manually configured trace amplitude limit value is overwritten.

8.7.3 Warning for trace amplitude

The warning threshold corresponds to a percentage deviation from the trace amplitude limit value *TraceAmplitudeMin* (UART index 106_d and CAN index 2010_h [7_h])
The warning threshold for the trace amplitude is calculated using the *TraceAmplitudeWarning* factor (UART index 107_d and CAN index 2010_h [8_h]). There is no index for calling up the calculated value directly.

Calculation of **dark guide trace**:

TraceAmplitudeWarning_threshold = *TraceAmplitudeMin* [LSB] - (*TraceAmplitudeMin* [LSB] * *TraceAmplitudeWarning*)

Calculation of **light guide trace**:

TraceAmplitudeWarning_threshold = *TraceAmplitudeMin* [LSB] + (*TraceAmplitudeMin* [LSB] * *TraceAmplitudeWarning*)

8.7.4 Process data information for the "Trace amplitude" filter

In the status byte of the process data, there are two bits for information relating to the trace amplitude:

- Bit 2: *Trace amplitude warning*
- Bit 5: *Trace amplitude error*

The *Trace amplitude warning* bit is set if the number of detected traces for which the trace amplitude is greater (dark guide trace) or less (light guide trace) than the warning threshold, is greater than or equal to one.

The *Trace amplitude error* bit is set if the number of detected traces for which the contrast is greater (dark guide trace) or less (light guide trace) than *TraceAmplitudeMin*, is greater than or equal to one.

8.7.5 Index overview for the "Trace amplitude" filter

Bit counting method: bit0 … bit15.

Name	Index UART	Index [sub-index] CANopen	Index length [byte]	Access	Data / (default value)	Info
Activate filter	2_d	2000_h [0_h]	2	W	233_d	System command
Deactivate filter	2_d	2000_h [0_h]	2	W	234_d	System command
Teaching the minimum contrast	2_d	2000_h [0_h]	2	W	196_d	System command
TraceAmplitude-Min	106_d	2010_h [7_h]	2	RW	(2500_d)	Result from teach or manual change, unit [LSB]
TraceAmplitude-Warning	107_d	2010_h [8_h]	2	RW	(20_d)	Factor for calculation of warning threshold, unit [%]
TraceAmplitu-deTol	108_d	2010_h [9_h]	2	RW	(1000_d)	During teach: Tolerance for calculation of minimum threshold, unit [LSB]
Status	200_d	2020_h [1_h]	2	R	Bit no. 7	1 = Trace amplitude error See also process data status byte, bit no. 5 (Chapter 7.1.4.1)
Status	200_d	2020_h [1_h]	2	R	Bit no. 4	1 = Trace amplitude warning See also process data status byte, bit no. 2 (Chapter 7.1.4.1)
UserMode	75_d	2002_h [0_h]	2	R	Bit no. 4	If bit = 1, then "Trace amplitude" filter is active

Table 8.7: Index accesses for the "Trace amplitude" filter

8.8 Index overview – More data on correct and incorrect traces

It is also possible to access the detected and filtered traces without accessing the process data. In doing so, additional information on the traces can be retrieved:

- With valid traces, the warning (if available) is shown in the *TraceValidStatus* index (UART index 210_d and CAN index 2025_h [01…06]) for each trace.
- With filtered-out traces, the error why the trace was filtered out is shown in the *TraceInvalidStatus* index (UART index 215_d and CAN index 2029_h [01…06]).
- The amplitudes which are used for calculating the filters can be read in the *TraceValidAmp* index (UART index 209_d and CAN index 2023_h [01…12]).
- The data for filtered-out traces can be read in the *TraceInvalidAmp* index (UART index 214_d and CAN index 2028_h [01…12]).

The data is always sorted by edge/trace in ascending order.

Direct access to all data of valid traces

Name	Index UART	Index [sub-index] CANopen	Index length [byte]	Ac-cess	Data	Info
TraceValidSub-Pixel	207_d	2022_h $[1_h]…[C_h]$	24	R	[LeftEdge1 LowByte, LeftEdge1 HighByte, RightEdge1 LowByte, RightEdge1 HighByte, LeftEdge2 LowByte, LeftEdge2 HighByte, RightEdge2 LowByte, RightEdge2 HighByte, …]	Contains the edge positions of the valid traces: • 16 bit for each edge • Divided into LowByte and HighByte • Only traces are displayed • A trace always consists of two consecutive edges
TraceValidAmp	208_d	2023_h $[1_h]…[C_h]$	24	R	[Environment1 Low-Byte, Environment1 HighByte, Trace1 Low-Byte, Trace1 HighByte, Environment2 LowByte, Environment2 High-Byte, Trace2 LowByte, Trace2 HighByte, …]	Contains the amplitude of the environment and of the trace: • 16 bit amplitude value • Divided into LowByte and HighByte • Sorted in ascending order consistent with the traces in index 207 and 2022_h
TraceValidSta-tus	210_d	2025_h $[1_h]…[6_h]$	12	R	[Trace1, Trace2, Trace3, …]	The warning is shown for each trace. Data: 1_h: contrast warning 2_h: trace amplitude warning

Table 8.8: Index overview: direct access to all data of valid traces

Direct access to all data of invalid traces

Name	Index UART	Index [sub-index] CANopen	Index length [byte]	Access	Data	Info
TraceInvalid-SubPixel	213_d	2027_h $[1_h]…[C_h]$	24	R	[LeftEdge1 LowByte, LeftEdge1 HighByte, RightEdge1 LowByte, RightEdge1 HighByte, LeftEdge2 LowByte, LeftEdge2 HighByte, RightEdge2 LowByte, RightEdge2 HighByte, …]	Contains the edge positions of the valid traces: 16 bit for each edge Divided into LowByte and HighByte Only traces are displayed A trace always consists of two consecutive edges
TraceInvalidAmp	214_d	2028_h $[1_h]…[C_h]$	24	R	[Environment1 LowByte, Environment1 HighByte, Trace1 LowByte, Trace1 HighByte, Environment2 LowByte, Environment2 HighByte, Trace2 LowByte, Trace2 HighByte, …]	Contains the amplitude of the environment and of the trace: 16 bit amplitude value Divided into LowByte and HighByte
TraceInvalidStatus	215_d	2029_h $[1_h]…[6_h]$	12	R	[Trace1, Trace2, Trace3, …]	The error is shown for each trace. Data: 1_h: contrast error 2_h: trace amplitude error 4_h: trace width error

Table 8.9: Index overview: direct access to all data of invalid traces

9 Tips for initial commissioning

To obtain a quick impression of how the sensor operates, you will need:

- USB <-> UART interface adapter (RS232, RS422, RS485)
- PC software (see Chapter 6)
- Mounting bracket for the device

9.1 Configuration of the sensor according to the trace

9.1.1 Variant: All filters ON

The aim of this procedure is to detect as few incorrect traces as possible.

↳ Reset the sensor to the factory settings (system command).

↳ Switch on all filters.

↳ Position the sensor or the vehicle with the sensor over the trace.

↳ Perform teach mode 4. This mode teaches all three filters at once.

The trace output is now very restrictive. If a point is reached at which the sensor stops outputting traces, it is possible to check the status bit in the process data to find out which filter is responsible for this. Alternatively, *Status* (UART index 200 and CAN index 2020_h [1_h]) can be evaluated.
By evaluating the warning bits and error bits, a corresponding action can be triggered in the vehicle's control unit.
The warning can help in detecting gradual soiling of the guide trace. Alternatively, the contrast information from the process data response can be evaluated.
The system user can then be informed of any location-dependent recommendation such as "clean trace" or "replace trace".
If is recommended to clean the guide trace if the contrast has continuously reduced over a long period.
Replacement of the guide trace is recommended if the trace amplitude deviates from the taught value or if, owing to the "Trace width" filter, a trace is no longer detected because it has become detached or has widened.
Before a type 2 switch is reached, the maximum trace width must be increased using the switch function so that the midpoint of the switch is output. To enable this, the currently used trace number must also be transferred.

9.1.2 Changeover between different traces

If the system consists of various traces with different width and/or trace type (light/dark), the following procedure is recommended.
For each trace, a parameter set with the appropriately adjusted variables must be stored in the control unit.
The following settings should be stored:

Index name	Comment
TraceWidthMax	
TraceWidthMin	
TraceTeachThr	Taught threshold (affects the measured trace width)
TraceContrastMin	
TraceAmplitudeMin	
SwitchTraceWidthFactor	For changeover between a system with 2-way and 3-way switches.

Table 9.1: Parameters for trace-specific parameter set

9.2 Floor markings

The following approaches are conceivable in order to provide the vehicle with location-dependent information by means of the trace or additional markings.

Trace width

The width of the trace can be varied. The sensor always outputs the left edge and the right edge of the trace. The difference between these two values is the width.

The trace width information can be used to inform the vehicle whether e.g. it should move slower or faster.

Markings next to the trace

Additional markings can be attached next to the trace in order to e.g. create a code.

For example, a 4 bit code can be realized by the presence of detected traces (which satisfy the filters) at a certain position.

From the process data, the vehicle's control unit recognizes if traces have been discovered which, owing to the filters, are not output via the process data.

It is therefore possible to design the markings such that they are not detected as a trace. The positions of the markings can be read out by evaluating *TraceInvalidSubPixel*. In this way, it is possible to implement a code for system control.

"Trace width" filter	Active	Inactive
Width of marking	…< trace width or …> trace width	Not relevant
Information that marking has been detected	Process data byte 2 *StatusPD* via bit 3 (trace width error)	Process data output
Number of invalid traces	TraceInvalidNum UART index 211_d CAN index 2026_h	
Position of trace	TraceInvalidSubPixel UART index 213_d CAN index 2027_h	

Table 9.2: Influence of "Trace width" filter

How is the trace width influenced?

- The distance of the sensor from the marking influences the trace width by up to ± 5 mm (vehicle loaded, unloaded).
- The linearity error (see figure 11.6).

Marking requirements

- In the simplest instance, same color as the guide trace.
- The marking trace is permitted to be narrower than the guide trace.
- To ensure reliable operation, the distance between two markings or between a guide trace and a marking must comply with the specifications given in Chapter 3.3.

Example:

Settings:

- The marking trace is considerably narrower than the main trace, but must comply with the requirements given in Chapter 3.3. Furthermore, the distance specified in Chapter 3.3 must also be observed.
- The trace width filter is active and is set to the main trace (teach).

If the sensor is positioned above the marking, the marking trace is indicated as an invalid trace:

Process data byte 2 *StatusPD* via bit 3 (trace width error)

The position of this invalid trace is read out via *TraceInvalidSubPixel* (UART index 213_d; CAN index 2027_h). This can be used to establish the side of the main trace on which the marking is located. The number of invalid traces can be read via *TraceInvalidNum* (UART index 211_d; CAN index 2026_h).

Amplitude of the trace

By reading out the parameter *TraceValidAmp* or *TraceInvalidAmp*, it is possible to make a distinction between traces.

Additionally, a distinction between markings next to the trace could be made on the basis of their amplitude and, in this way, it would be possible to implement system control.

9.3 Basic settings for the filters

The basic settings for the filters were determined using a black, 40mm wide guide trace on a white background. The distance between the trace and the lower edge of the sensor was 35mm.

The values were selected such that

- the trace is still detected with a change in vehicle height of ±30mm.
- the trace is still detected with a change in the angle between the trace/floor and the sensor's longitudinal axis of up to 5°.

The diffuse reflection (coefficient of luminous intensity (CIL) of the light) was:

- 90% for the background.
- 6% for the guide trace.

10 Service and support

Service hotline

You can find the contact information for the hotline in your country on our website www.leuze.com under **Contact & Support**.

Repair service and returns

Defective devices are repaired in our service centers competently and quickly. We offer you an extensive service packet to keep any system downtimes to a minimum. Our service center requires the following information:

- Your customer number
- Product description or part description
- Serial number and batch number
- Reason for requesting support together with a description

Please register the merchandise concerned. Simply register return of the merchandise on our website www.leuze.com under **Contact & Support > Repair Service & Returns**.

To ensure quick and easy processing of your request, we will send you a returns order with the returns address in digital form.

What to do should servicing be required?

	NOTE
!	**Please use this chapter as a master copy should servicing be required.** ↳ Enter the contact information and fax this form together with your service order to the fax number given below.

Customer data (please complete)

Device type:	
Serial number:	
Firmware:	
Display messages	
Status of LEDs:	
Error description	
Company:	
Contact person/department:	
Phone (direct dial):	
Fax:	
Street / no.:	
ZIP code / City:	
Country:	

Leuze Service fax number:

+49 7021 573 - 199

11 Technical data

11.1 General technical data OGS 600

Operating voltage		18 … 30VDC (PELV [1]) , Class 2)
Average current consumption		Approx. 180 mA at 24 V DC (no load at switching output)
Integrated LED illumination		Red, wavelength 634nm, risk group 0 (exempt group) in acc. with EN 62471:2008
Transmitter/receiver		49 transmitter and 49 receiver elements
Sensor field width	OGS 600-280/… OGS 600-140/…	300mm 150mm
Distance between sensor and floor		10 … 70mm, nominal: 30mm optimal: 20 … 40mm
Measurement time		10ms
Linearity error		Typ. 5mm (at a sensor-floor distance of 30mm)
Measurement value resolution		Typ. 1mm (at a sensor-floor distance of 30mm)
Guide trace width		Ideally 40mm, at least 10mm
Guide trace color		Light trace on dark floor, dark trace on light floor
Branches		Switch filter
Interface type	OGS 600-…/CN… OGS 600-…/D3… OGS 600-…/D2…	CANopen and RS232 RS485 RS422
Switching inputs/outputs		1 adjustable switching output (all OGS 600s), 1 configurable switching input/output (only OGS 600s with RS485 or RS422)
Degree of protection		IP 65 [2])
VDE protection class		III
Housing		Diecast aluminum
Optics cover		Polycarbonate [3])
Weight	OGS 600-280/… OGS 600-140/…	Approx. 405g Approx. 245g
Ambient temp. (operation/storage)		-15°C … +50°C / -30°C … +60°C
Rel. air humidity		Max. 90% (non-condensing)
Standards applied		EN 60947-5-2:2007+A1:2012
Conformity		CE

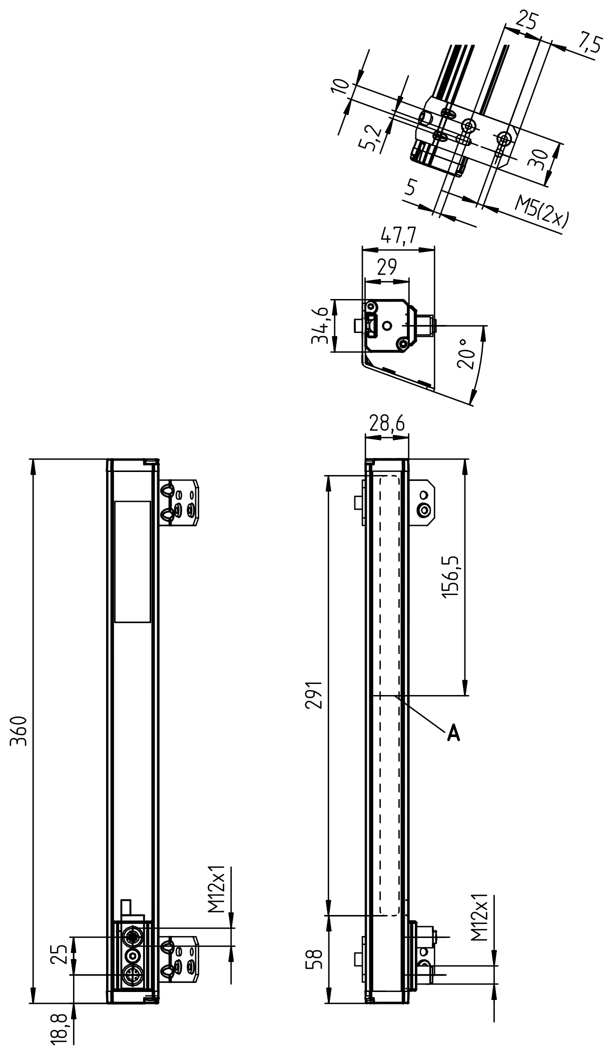
1) Protective Extra Low Voltage (PELV) - protective extra-low voltage.

2) Only with screwed-on M12 connectors or mounted caps

3) Only lint-free cloths may be used to clean the lens covers. Tips and hard objects damage the lens.

11.2 Dimensioned drawings

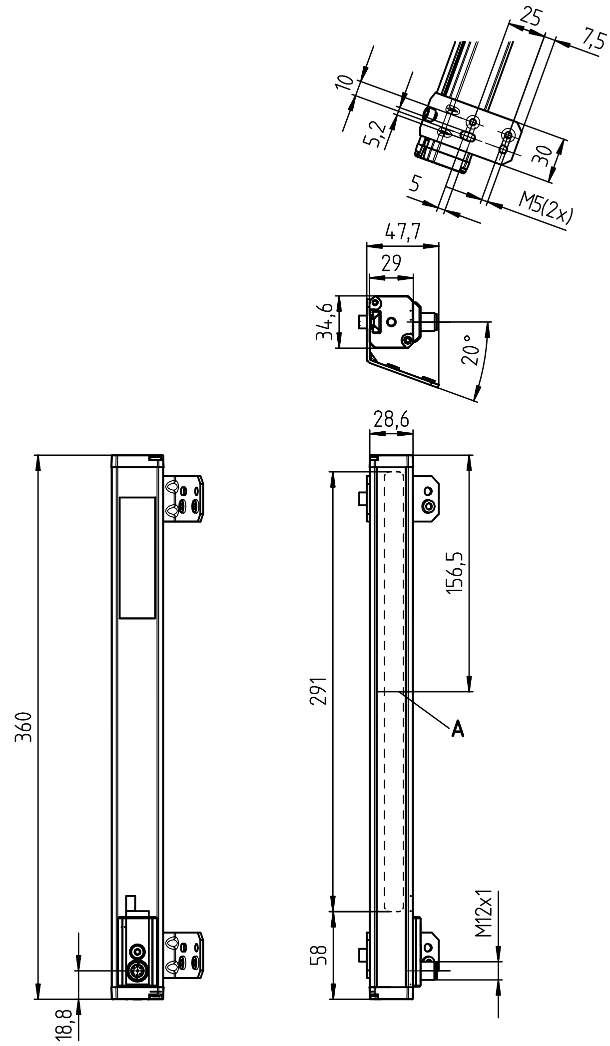
11.2.1 Dimensioned drawing OGS 600-280/CN-M12 – long version



A Center of sensor's measurement field

Figure 11.1: Dimensioned drawing OGS 600-280/CN-M12 – long version

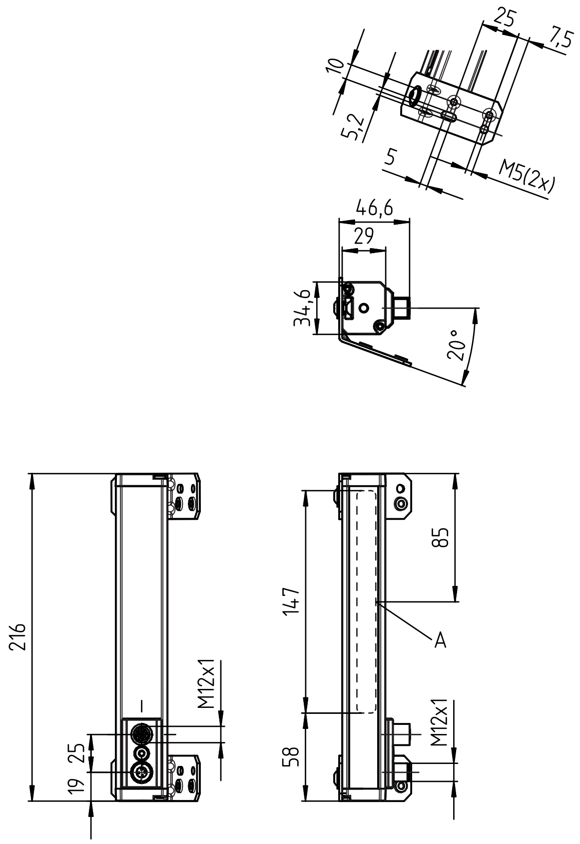
11.2.2 Dimensioned drawing OGS 600-280/D…-M12.8 – long version



A Center of sensor's measurement field

Figure 11.2: Dimensioned drawing OGS 600-280/D…-M12.8 – long version

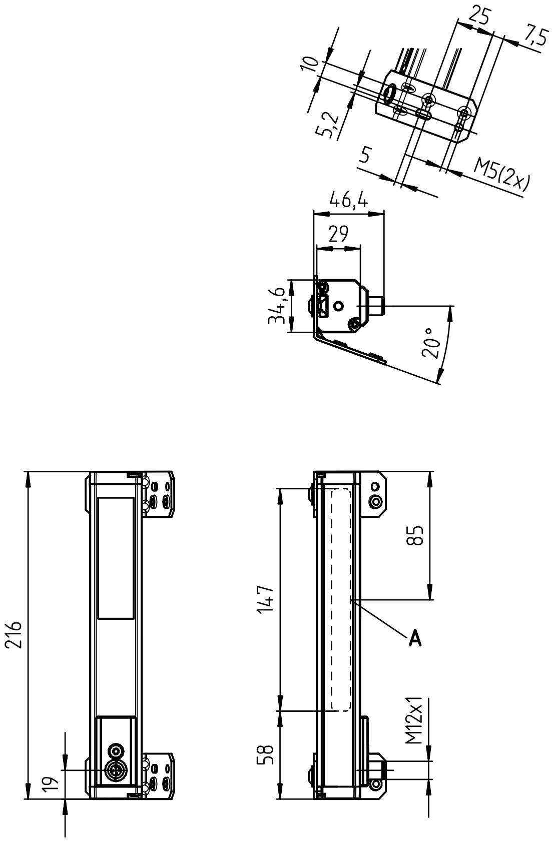
11.2.3 Dimensioned drawing OGS 600-140/CN-M12 – short version



A Center of sensor's measurement field

Figure 11.3: Dimensioned drawing OGS 600-140/CN-M12 – short version

11.2.4 Dimensioned drawing OGS 600-140/D…-M12.8 – short version

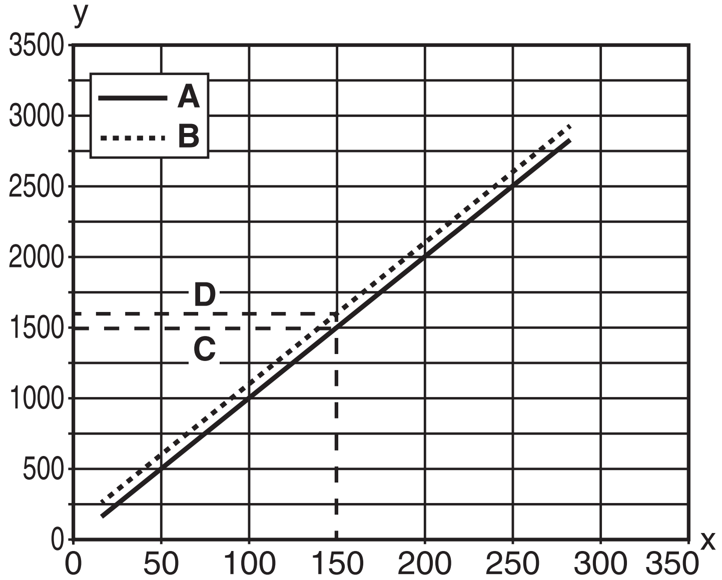


A Center of sensor's measurement field

Figure 11.4: Dimensioned drawing OGS 600-140/D…-M12.8 – short version

11.3 Diagrams

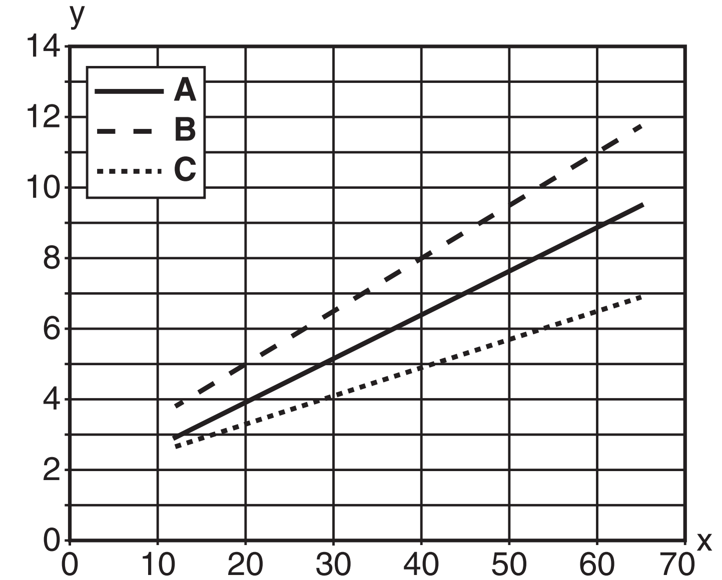
11.3.1 Sensor characteristic curve with one guide trace



A Measurement value for left edge of guide trace
B Measurement value for right edge of guide trace
C Output value: Position of left edge
D Output value: Position of right edge
x Position of left edge of guide trace under sensor
y Output value: Edge positions

Figure 11.5: Sensor characteristic curve with one trace

11.3.2 Linearity error



A Characteristic curve: typical linearity error
B Characteristic curve: maximum linearity error
C Characteristic curve: minimum linearity error
x Distance of sensor to ground
y Linearity error

Figure 11.6: Linearity error as a function of distance between sensor and ground

12 Order guide and accessories

12.1 Sensor part number code

OGS 600-	XXX	/YY	-M12	.8	
				N/A	2x 5-pin
				.8	1x 8-pin
					M12 connection technology
				/CN	CANopen and RS232 interface
				/D3	RS485 interface
				/D2	RS422 interface
				280	Long version
				140	Short version
					Optical guidance sensor, OGS 600 series (Optical Guidance Sensor)

Table 12.1: OGS 600 part number code

12.2 Order guide for sensor

Part no.	Type designation	Description
50137472	OGS 600-280/CN-M12	Optical guidance sensor OGS 600, long version, CANopen and RS232 interface, 2x M12 connector, 5-pin
50137473	OGS 600-140/CN-M12	Optical guidance sensor OGS 600, short version, CANopen and RS232 interface, 2x M12 connector, 5-pin
50137474	OGS 600-280/D3-M12.8	Optical guidance sensor OGS 600, long version, RS485 interface, 1x M12 connector, 8-pin
50137475	OGS 600-140/D3-M12.8	Optical guidance sensor OGS 600, short version, RS485 interface, 1x M12 connector, 8-pin
50137476	OGS 600-280/D2-M12.8	Optical guidance sensor OGS 600, long version, RS422 interface, 1x M12 connector, 8-pin
50137477	OGS 600-140/D2-M12.8	Optical guidance sensor OGS 600, short version, RS422 interface, 1x M12 connector, 8-pin

12.3 Accessories

12.3.1 Connection cables for CANopen/RS232 devices

Connection cables

Part no.	Type designation	Description
50114692	KB DN/CAN-2000 BA	CANopen connection cable, length 2m, PUR black, M12 socket, 5-pin, A-coded, axial, open end
50114693	KB DN/CAN-2000 SA	CANopen connection cable, length 2m, PUR black, M12 plug, 5-pin, A-coded, axial, open end
50114696	KB DN/CAN-5000 BA	CANopen connection cable, length 5m, PUR black, M12 socket, 5-pin, A-coded, axial, open end
50114697	KB DN/CAN-5000 SA	CANopen connection cable, length 5m, PUR black, M12 plug, 5-pin, A-coded, axial, open end

Interconnection cables

Part no.	Type designation	Description
50118184	K-YCN M12A-5m-M12A-S-PUR	CANopen Y-interconnection cable, PUR black, branch 1 length 0.25m, branch 2 length 5m, 2x M12 plug, 5-pin, A-coded, axial, 1x M12 socket, 5-pin, A-coded, axial
50118185	K-YCN M12A-M12A-S-PUR	CANopen Y-interconnection cable, PUR black, branch 1 length 0.25m, branch 2 length 0.35m, 2x M12 plug, 5-pin, A-coded, axial, 1x M12 socket, 5-pin, A-coded, axial
50114691	KB DN/CAN-1000 SBA	CANopen interconnection cable, length 1m, PUR black, 1x M12 plug, 5-pin, A-coded, axial, 1x M12 socket, 5-pin, A-coded, axial
50114694	KB DN/CAN-2000 SBA	CANopen interconnection cable, length 2m, PUR black, 1x M12 plug, 5-pin, A-coded, axial, 1x M12 socket, 5-pin, A-coded, axial
50129779	KDS DN-M12-5A-M12-5A-P3-010	CANopen interconnection cable, length 1m, PUR violet, 1x M12 plug, 5-pin, A-coded, axial, 1x M12 socket, 5-pin, A-coded, axial
50129780	KDS DN-M12-5A-M12-5A-P3-020	CANopen interconnection cable, length 2m, PUR violet, 1x M12 plug, 5-pin, A-coded, axial, 1x M12 socket, 5-pin, A-coded, axial
50129781	KDS DN-M12-5A-M12-5A-P3-050	CANopen interconnection cable, length 5m, PUR violet, 1x M12 plug, 5-pin, A-coded, axial, 1x M12 socket, 5-pin, A-coded, axial

12.3.2 Connection cables for RS485/RS422 devices

Connection cables

Part no.	Type designation	Description
50135120	KD U-M12-8A-P1-010	PWR/RS485/RS422 connection cable, length 1m, PUR black, M12 socket, 8-pin, A-coded, axial, open end
50135121	KD U-M12-8A-P1-020	PWR/RS485/RS422 connection cable, length 2m, PUR black, M12 socket, 8-pin, A-coded, axial, open end
50135122	KD U-M12-8A-P1-050	PWR/RS485/RS422 connection cable, length 5m, PUR black, M12 socket, 8-pin, A-coded, axial, open end

12.3.3 RS485-USB adapter set

Part no.	Type designation	Description
On request	RS485-USB adapter	RS485-USB converter
	Y-cable	Connection cable for connection of sensor, RS485-USB converter and supply voltage

12.3.4 Guide trace tapes, self-adhesive

Part no.	Type designation	Description
50137874	OTB 40-BK250	Black trace tape, width 40mm, self-adhesive, 25m roll
50137875	OTB 40-WH250	White trace tape, width 40mm, self-adhesive, 25m roll
50137873	OTB 40-GN250	Dark green trace tape, width 40mm, self-adhesive, 25m roll
50137876	OTB 30/100-BK/WH250	Black trace tape, width 30mm on white base material, width 100mm, self-adhesive, 25m roll
50137877	OTB SET-GN/BK/WH003	Set of 0.3m trace tapes • Black • White • Dark green • Black on white base material

13 Version history of device firmware

v1.3	Until July 2018
v1.4	Index 836 made accessible for users. Default value reduced from 450 to 100. Effect: Detection of lower contrasts.
v1.5	From August 2018. Index 70 (UART Node No): Value range extended to 1 -15
v1.6	Problem with retro-reflective trace rectified
v1.7	Problem with CanOpen object directory rectified
v1.8	New PD types 2, 5, 6 and 7
v1.9	New PD type 8: same as PD type 4 except that 1-6 traces are always output. Default is 3 traces;
V2.0	From September 2021 Addition of bit 15 to Index 200. Following error corrected: If LED illumination is deactivated and single pixels are read out via e.g. Index 202, the sensor no longer reacts and requires a voltage reset

14 Appendix – Sensor measurement values for RAL colors

Overview RAL colors

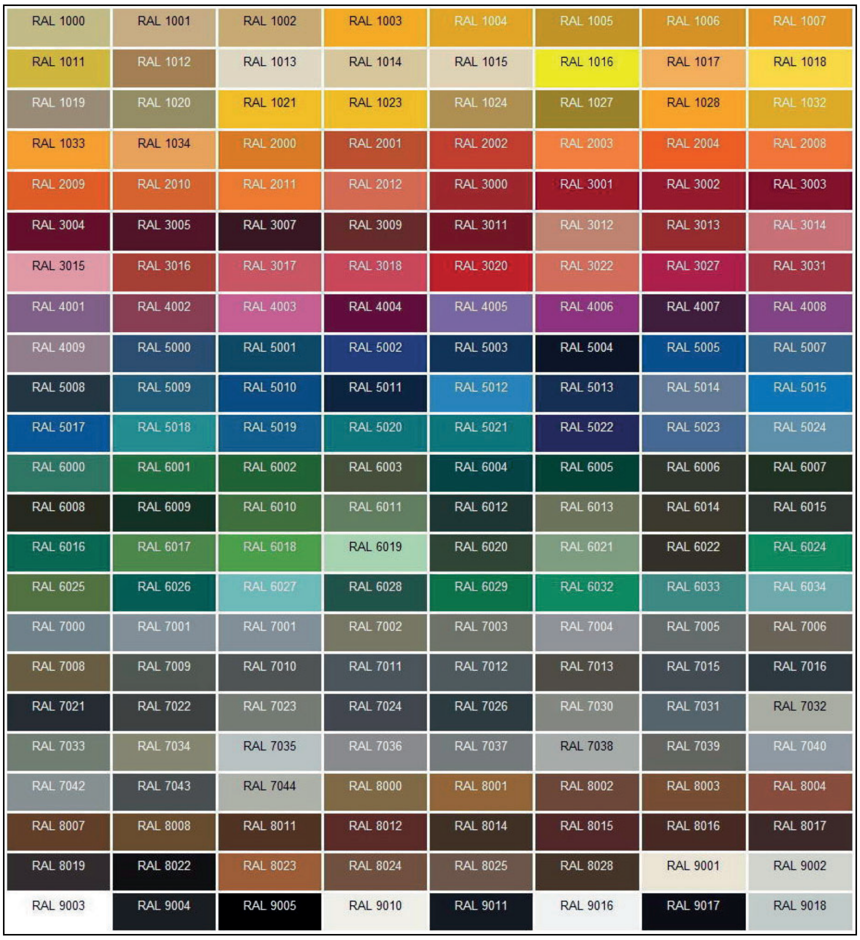


Figure 14.1: Excerpt from RAL color chart

Sensor measurement values for RAL colors

Color designation	RAL no.	Measurement value of the sensor: Amplitude [LSB]	Grayscale value	
Jet black	9005	400	0.01886792	
Opal green	6026	500	0.02358491	
Black blue	5004	600	0.02830189	
Graphite black	9011	600	0.02830189	
Ultramarine blue	5002	700	0.03301887	
Sapphire blue	5003	700	0.03301887	
Pearl green	6035	700	0.03301887	
Pearl opal green	6036	700	0.03301887	
Black brown	8022	700	0.03301887	
Black green	6012	800	0.03773585	
Signal black	9004	800	0.03773585	
Traffic black	9017	800	0.03773585	
Green blue	5001	900	0.04245283	
Signal blue	5005	900	0.04245283	
Gray blue	5008	1200	0.05660377	
Emerald green	6001	1200	0.05660377	
Granite gray	7026	1300	0.06132075	
Gray brown	8019	1300	0.06132075	
Violet blue	5000	1400	0.06603774	
Leaf green	6002	1400	0.06603774	
Black red	3007	1900	0.08962264	
Patina green	6000	1900	0.08962264	
Yellow olive	6014	1900	0.08962264	
Graphite gray	7024	2100	0.0990566	
Brilliant blue	5007	2400	0.11320755	
Olive green	6003	2400	0.11320755	
Traffic gray B	7043	2500	0.11792453	
Purple violet	4007	3100	0.14622642	
Green gray	7009	3100	0.14622642	
Tarpaulin gray	7010	3100	0.14622642	
Blue gray	7031	3600	0.16981132	
Pearl blackberry	4012	4100	0.19339623	
Quartz gray	7039	4400	0.20754717	
Oxide red	3009	4700	0.22169811	
Khaki gray	7008	4700	0.22169811	
Wine red	3005	4900	0.23113208	
Beige gray	7006	5000	0.23584906	

Color designation	RAL no.	Measurement value of the sensor: Amplitude [LSB]	Grayscale value	
Reseda green	6011	5100	0.24056604	
Pearl mouse gray	7048	5300	0.25	
Mouse gray	7005	5400	0.25471698	
Blue lilac	4005	6100	0.28773585	
Concrete gray	7023	6100	0.28773585	
Dusty gray	7037	6100	0.28773585	
Telegray 2	7046	6600	0.31132075	
Reed green	6013	6800	0.32075472	
Cement gray	7033	6800	0.32075472	
Squirrel gray	7000	7200	0.33962264	
Traffic gray A	7042	7500	0.35377358	
Brown red	3011	7800	0.36792453	
Curry	1027	7900	0.37264151	
Pearl orange	2013	7900	0.37264151	
Yellow gray	7034	8000	0.37735849	
Purple red	3004	8100	0.38207547	
Stone gray	7030	8100	0.38207547	
Telegray 1	7045	8200	0.38679245	
Platinum gray	7036	9200	0.43396226	
Olive yellow	1020	9400	0.44339623	
Window gray	7040	9400	0.44339623	
Signal violet	4008	9500	0.44811321	
Pastel violet	4009	9900	0.46698113	
Gray beige	1019	10200	0.48113208	
Brown beige	1011	10700	0.50471698	
Ruby red	3003	11000	0.51886792	
Traffic purple	4006	11100	0.52358491	
Coral red	3016	11600	0.54716981	
Tomato red	3013	11800	0.55660377	
Agate gray	7038	12400	0.58490566	
Pebble gray	7032	13000	0.61320755	
Honey yellow	1005	13400	0.63207547	
Signal red	3001	13500	0.63679245	
Silk gray	7044	13900	0.65566038	
Flame red	3000	14000	0.66037736	
Beige red	3012	14000	0.66037736	
Carmine red	3002	14500	0.68396226	
Maize yellow	1006	15200	0.71698113	

Color designation	RAL no.	Measurement value of the sensor: Amplitude [LSB]	Grayscale value
Green beige	1000	15300	0.72169811
Beige	1001	15400	0.72641509
Signal orange	2010	15400	0.72641509
Telegray 4	7047	15700	0.74056604
Sand yellow	1002	15900	0.75
Lemon yellow	1012	16000	0.75471698
Light gray	7035	16000	0.75471698
Daffodil yellow	1007	16300	0.76886792
Papyrus white	9018	16400	0.77358491
Telemagenta	4010	16500	0.77830189
Golden yellow	1004	16600	0.78301887
Ivory	1014	17200	0.81132075
Salmon orange	2012	17200	0.81132075
Deep orange	2011	17400	0.82075472
Gray white	9002	17400	0.82075472
Traffic orange	2009	17800	0.83962264
Rose	3017	17800	0.83962264
Colza yellow	1021	17900	0.84433962
Light pink	3015	17900	0.84433962
Antique pink	3014	18000	0.8490566
Bright red orange	2008	18400	0.86792453
Oyster white	1013	18500	0.87264151
Light ivory	1015	18600	0.87735849
Signal yellow	1003	18900	0.89150943
Zinc yellow	1018	19100	0.9009434
Strawberry red	3018	19100	0.9009434
Saffron yellow	1017	19300	0.91037736
Cream	9001	19600	0.9245283
Sulfur yellow	1016	19700	0.92924528
Melon yellow	1028	19800	0.93396226
Luminous yellow	1026	20100	0.94811321
Luminous orange	2005	20100	0.94811321
Signal white	9003	20100	0.94811321
Luminous bright orange	2007	20200	0.95283019
White aluminum	9010	20200	0.95283019
Traffic white	9016	21200	1

Table 14.1: Sensor measurement values for RAL colors