Manual
Software SPECTRO3-MSM-SLE-Scope V1.2
(PC software for Microsoft® Windows® 7, 8, 10)

for color sensors of SPECTRO-3-MSM-SLE Series
with internal temperature compensation and white light balancing

This manual describes the installation of the PC software for the SPECTRO-3-MSM-SLE color sensor. As a support for commissioning of the color sensor this manual explains the functional elements of the Windows® user interface.

The SPECTRO-3-MSM-SLE provides highly flexible signal acquisition. For example, the sensor can be operated in alternating-light mode (AC mode), which makes the sensor insensitive to extraneous light. It also can be set to constant-light mode (DC mode), which makes the sensor extremely fast. With the stepless adjustment of the integrated light source as well as the selectable gain of the receiver signal and an INTEGRAL function the sensor can be set to almost any surface or any "self-luminous object".

When the integrated light source of the SPECTRO-3-MSM-SLE color sensor is activated, the sensor detects the radiation that is diffusely reflected from the object. As a light source the SPECTRO-3-MSM-SLE color sensor uses LEDs with adjustable transmitter power. An integrated 3-fold receiver for the red, green, and blue content of the light that is reflected from the object, or the light that is emitted by a "self-luminous object", is used as a receiver.

The color sensors of the SPECTRO-3-MSM-SLE series can be taught up to 31 colors that are provided directly at the outputs, and up to 48 colors that are output by means of grouping. For each of these taught colors it is possible to set tolerances. Evaluation of the taught colors is performed either with the "FIRST HIT" or with the "BEST HIT" mode. With "FIRST HIT" the first hit in the teach table is output, and with "BEST HIT" the best hit in the teach table is output. Raw data are represented with 12 bit resolution.

Color detection either operates continuously or is started through an external PLC trigger signal. The respective detected color either is provided as a binary code at the 5 digital outputs or can be sent directly to the outputs, if only up to 5 colors are to be detected.

Through input IN0 the color sensor can be taught up to 31 colors (max. 48 colors in group mode). For this purpose the corresponding evaluation mode must be set with the software.

Parameters and measurement values can be exchanged between a PC and the SPECTRO-3-MSM-SLE color sensor through the serial RS232 interface. All the parameters for color detection also can be saved to the non-volatile EEPROM of the SPECTRO-3-MSM-SLE color sensor through this serial RS232 interface. When parameterisation is finished, the color sensor continues to operate with the current parameters in STAND-ALONE mode without a PC.

Possible firmware updates can be easily performed through the RS232 interface, even with the sensor system in installed condition. (→ cf. Annex „Firmware update via software Firmware Loader").

The SPECTRO-3-MSM-SLE series sensors are factory calibrated. When establishing the connection, the date of the next calibration is displayed in the status window in the **CONNECT** tab. For continuously accurate measurement results, we recommend having the calibration renewed every 18 months and sending in the sensors for calibration.

The sensors of the SPECTRO-3-MSM-SLE series can be individually calibrated. Calibration can be performed to any surface with known color coordinates. A ColorChecker[TM] table is available as an alternative. This table has 24 color fields according to the CIE STANDARD that can be used for calibration.

0 Contents

Shortcuts:	
SEND	F9
GET	F10
GO	F11
STOP	F12

1. Installation of the SPECTRO3-MSM-SLE-Scope software

The following requirements must be fulfilled for successful installation of the software:

* Microsoft® Windows® 7, 8, 10
* IBM PC AT or compatible
* VGA graphics
* Microsoft-compatible mouse
* Serial RS232 interface at the PC or USB slot or RJ45 connector
* Cable *cab-las4/PC* for the RS232 interface or *cab-4/USB* USB converter or *cab-4/ETH* Ethernet converter

Please install the software as described below:

|	
1.	The software can be installed directly from the installation DVD. To install the software, start the SETUP program in the SOFTWARE folder of the DVD.
2.	The installation program displays a dialog and suggests to install the software in the C:\"FILENAME" directory on the hard disk. You may accept this suggestion with OK or [ENTER], or you may change the path as desired. Installation is then performed automatically.
3.	During the installation process a new program group for the software is created in the Windows Program Manager. In the program group an icon for starting the software is created automatically. When installation is successfully completed the installation program displays "Setup OK".
4.	After successful installation the software can be started with a left mouse button double-click on the icon.

Windows^{TM} is a registered trademark of Microsoft Corp.
VGA^{TM} is a trademark of International Business Machines Corp.

2.   Operation of the SPECTRO3-MSM-SLE-Scope software

Please read this chapter first before you start to adjust and parameterise the SPECTRO-3-MSM-SLE color sensor.

When the SPECTRO3-MSM-SLE-Scope software is started, the following window appears on the Windows interface:

TIP! To avoid problems with the handling of the file path, it is advisable to run the software as administrator. You can either set this in the **Properties** under **Compatibility** or you start the software with a right click and choose **"Run as administrator"**.
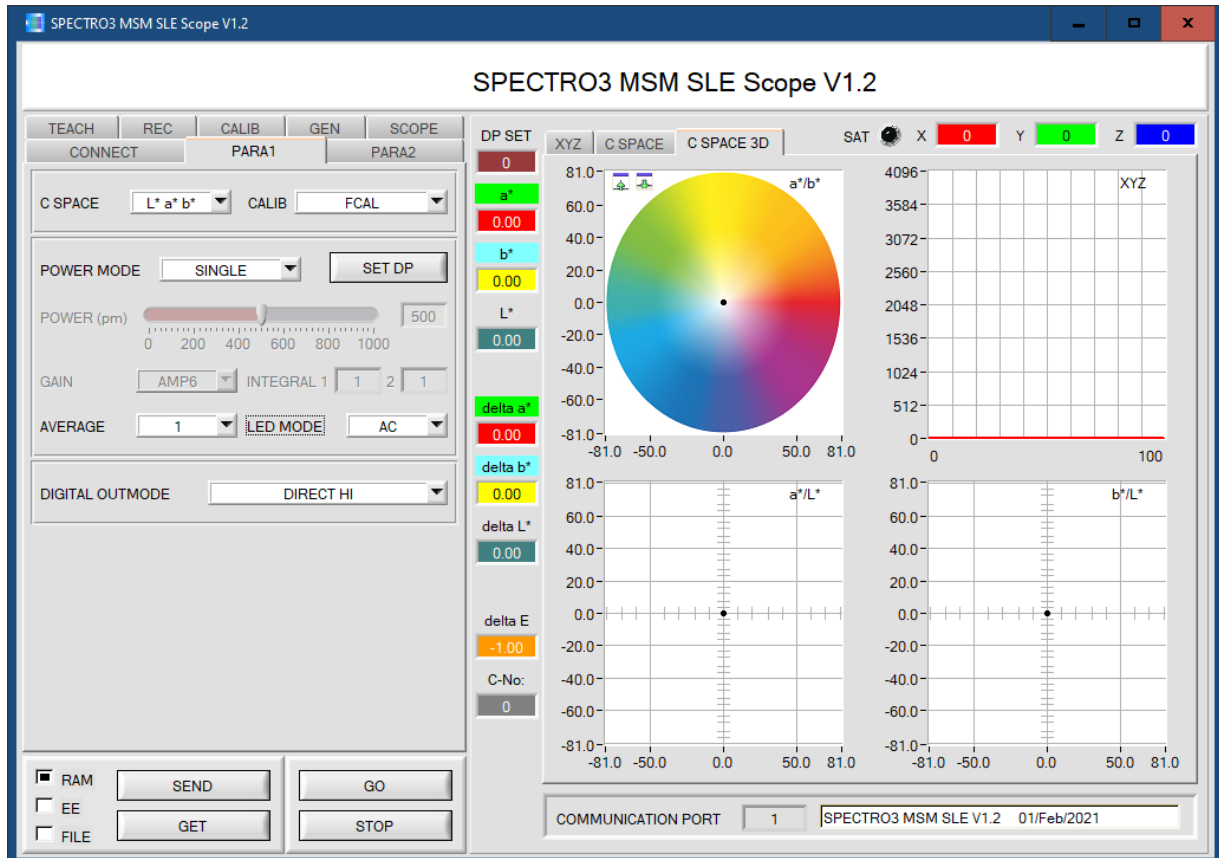
The window size and position will be the same as when the software was last closed. A double-click with the right mouse button e.g. under the minimise symbol places the window centrally in its original size.

If a connection is not established automatically, e.g. if no sensor is connected, the software can be run in OFFLINE mode. In offline mode it only is possible to exchange parameters with a file on a storage medium, which often is helpful for the purpose of analysing parameter files.

If a sensor is connected and a connection still cannot be established, either the SCOPE version (program at the PC) and the firmware version (program in the sensor) do not match, or the interface to the sensor must be correctly configured.
If different Scope and firmware versions should be the problem, please get the Scope version that matches the firmware from your supplier.
The interface configuration is described in the CONNECT tab chapter.

Pressing the right mouse button on an individual element will call up a short help text.

Due to a better overview, parameters that are not required, displays, graphs, etc., are greyed out or invisible depending on the parameterization.

2.1 Tab CONNECT

CONNECT:
Pressing the **CONNECT** tab opens a window for selecting and configuring the interface.

The **COMMUNICATION PROTOCOL** function field is used for selecting either an **RS232** or a **TCP/IP** protocol.
If **RS232** is selected, a port from 1 to 256 can be selected with **SELECT COM PORT**, depending on which port the sensor is connected to. The sensor operates with a set baudrate that can be modified with **CHANGE BAUDRATE** (see below). The sensor and the user interface both must operate with the same baudrate. At the user interface the baudrate is set with **SELECT BAUDRATE**. If after starting the software should not automatically establish a connection, the correct baudrate can be found with **SELECT BAUDRATE**.
If an converter is used, the **COM PORT** number can be determined by way of the hardware manager in the system control panel.
A click on the magnifier symbol opens a list with all the possible COM ports in the display.

An RS232 to Ethernet converter **(cab-4/ETH)** is needed if the sensor should communicate through a local network. With this converter a connection to the sensor can be established using the **TCP/IP** protocol.
Parameterisation of the **cab-4/ETH** converter (assigning of IP address, baudrate setting, ...) can be done with the **SensorFinder software** that is available free of charge on the internet.

In order to establish a connection to the converter, its IP address or HOST name must be entered in the field **IP ADDRESS (xxx.xxx.xxx.xxx) OR HOST NAME**. The DROP DOWN menu (down arrow) shows the last 10 IP addresses that were used. An address from this list can be directly selected by clicking on the respective item. The DROP DOWN list is saved and is thus always available when the software is closed.
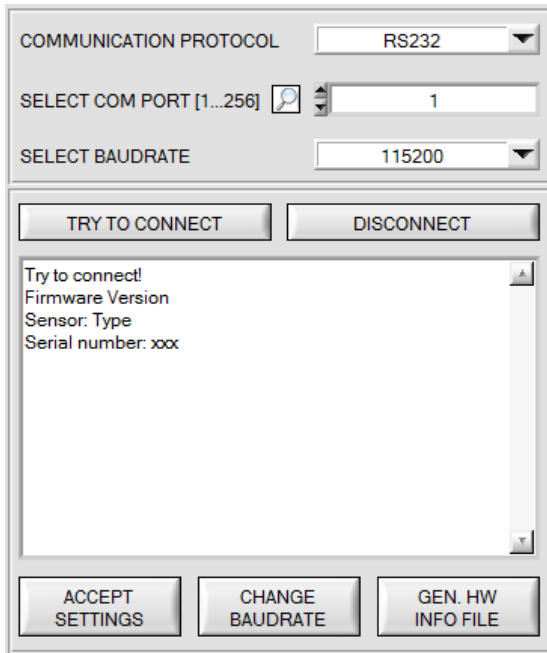The **PORT NUMBER** for the **cab-4/ETH** is 5000. This port number must not be changed.
When you press the **TRY TO CONNECT** button, the software tries to establish a connection with the set parameters. The communication status is shown in the display field. If the sensor answers with its FIRMWARE ID, the set connection type can be accepted by pressing **ACCEPT SETTINGS**. You will then be returned to the **PARA** tab. If you get a **TIMEOUT** message, the software could not establish a connection to the sensor. In this case please check if the interface cable is correctly connected, if the sensor is supplied with power, and if the set parameters are correct. If a connection has been accepted by pressing **ACCEPT SETTINGS**, the software starts automatically with these settings when called the next time.

DISCONNECT disconnects the connection between sensor and PC. The software then switches to OFFLINE mode, where it is only possible to exchange parameters with a file on a storage medium.

Under **PANEL ID** a name can be entered that will be displayed at different points in the program window, and that will be recorded in different files (e.g. Record File) as well. With the input field **LANGUAGE** a language can be set with which the individual controls are displayed on the surface. This also applies to the help function that is actuated with the right mouse button.

| Please note:	The stable function of the interface is a basic prerequisite for measured value transfer from the PC to the sensor.
⚠️ **ATTENTION !**	Due to the limited data transfer rate through the serial RS232 interface only slow changes of the raw signals at the sensor front end can be observed in the graphic output window of the PC. For maintaining maximum switching frequency at the sensor data communication with the PC must be stopped (press the STOP button).

The baudrate for data transfer through the RS232 interface can be set by means of the **SELECT BAUDRATE** drop down menu and **CHANGE BAUDRATE** function field.

If the baudrate should be changed, a connection must first be established by clicking on **TRY TO CONNECT**.
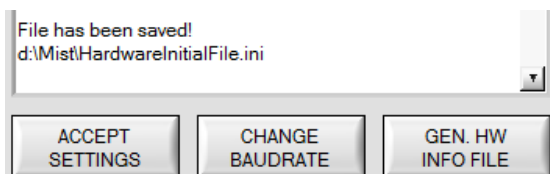The **CHANGE BAUDRATE** button will then be active.

COMMUNICATION PROTOCOL	RS232
SELECT COM PORT [1...256]	1
SELECT BAUDRATE	115200

TRY TO CONNECT   DISCONNECT

Try to connect!
Firmware Version
Sensor: Type
Serial number: xxx

ACCEPT SETTINGS   CHANGE BAUDRATE   GEN. HW INFO FILE

Now a new baudrate can be selected under **SELECT BAUDRATE**.
A click on **CHANGE BAUDRATE** sends the new baudrate information to the sensor.

|	
SELECT BAUDRATE	19200

CHANGE BAUDRATE
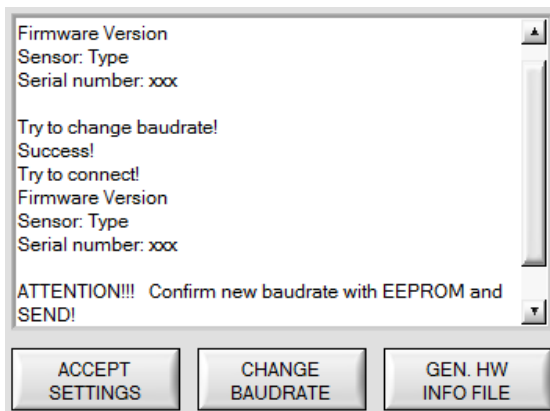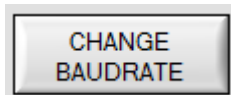
When the new baudrate information has been successfully sent, the sensor operates with the new baudrate. A window will pop up, prompting you to select **EEPROM** and then to press **SEND**. After a hardware reset the new baudrate only will be used when **EEPROM** and **SEND** have been pressed.

A click on **ACCEPT SETTINGS** saves the current interface settings, which will then be automatically set when the software is restarted.

Firmware Version
Sensor: Type
Serial number: xxx

Try to change baudrate!
Success!
Try to connect!
Firmware Version
Sensor: Type
Serial number: xxx

ATTENTION!!! Confirm new baudrate with EEPROM and SEND!

ACCEPT SETTINGS   CHANGE BAUDRATE   GEN. HW INFO FILE

A click on the **GEN. HW INFO FILE** generates a file in which all the important sensor data are stored in encrypted form. This file can be sent to the manufacturer for diagnostic purposes.

File has been saved!
d:\Mist\HardwareInitialFile.ini

ACCEPT SETTINGS   CHANGE BAUDRATE   GEN. HW INFO FILE

2.2 Tab PARA1, button SEND, GET, GO, STOP
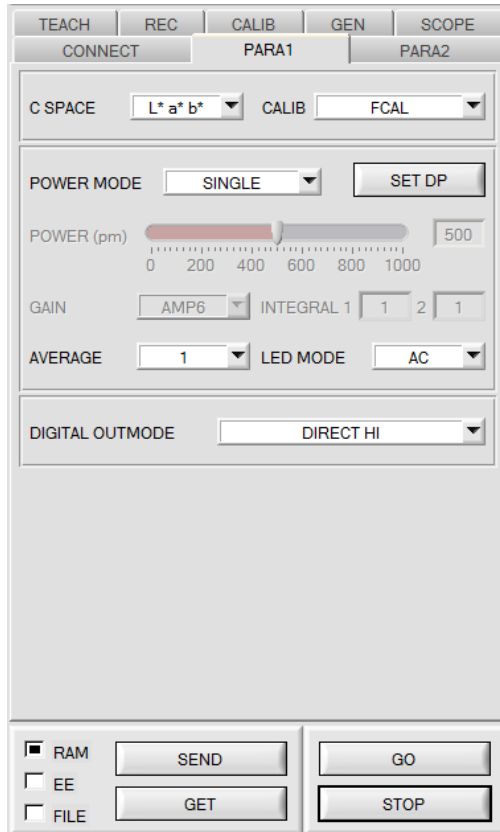
PARA1:
Pressing the **PARA1** tab opens a window for setting the sensor parameters.

ATTENTION!
A change of the parameter function groups only becomes effective at the sensor after actuation of the SEND button in the MEM function field!

SEND [F9]:
When the **SEND** button is clicked (or shortcut key button F9 is pressed), all the currently set parameters are transferred between PC and sensor. The target of the respective parameter transfer is determined by the selected button (**RAM, EEPROM,** or **FILE**).

GET [F10]:
The currently set values can be interrogated from the sensor by clicking on the **GET** button (or with shortcut key button F10). The source of data exchange is determined by the selected button (**RAM**, **EEPROM**, or **FILE**).

RAM:
After a click on the **SEND** button the current parameters are written into the **RAM** memory of the sensor, or they are read from the **RAM** by clicking on the **GET** button, i.e. these parameters are lost when the voltage at the sensor is switched off.

EEPROM:
After a click on the **SEND** button the current parameters are written into the non-volatile memory of the **EEPROM** in the sensor, or they are read from the **EEPROM** by clicking on the **GET** button, i.e. the parameters in the internal **EEPROM** are stored when the voltage at the sensor is switched off.

FILE:
After pressing **SEND**, the current parameters can be written to a selectable file on the hard disk. With **GET** parameters can be read from such a file. When the **SEND** or **GET** button is pressed, a dialog box opens for selecting the desired file.
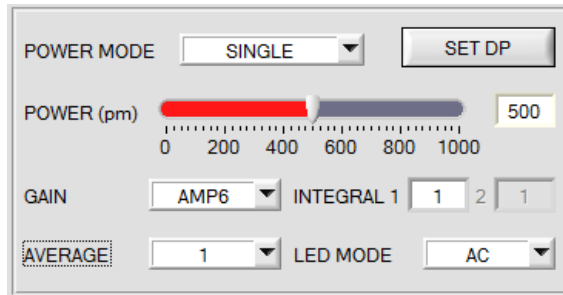
GO [F11]:
A click on this button (or with shortcut key button F11) starts data transfer from the sensor to the PC through the serial RS232 interface.
SOURCE is used to select which signals should be shown in the displays and graphs.

STOP [F12]:
A click on this button (or with shortcut key button F12) stops data transfer from the sensor to the PC through the serial RS232 interface.

POWER MODE	SINGLE ▼		SET DP
POWER (pm)	[slider] 500		
	0 200 400 600 800 1000		
GAIN	AMP6 ▼	INTEGRAL 1	1 2 1
AVERAGE	1 ▼	LED MODE	AC ▼

POWER MODE: siehe **SET DP**

POWER:
The transmitter power is constantly kept at the value set with the **POWER [pm]** slider (recommended operation mode). The **POWER** can be set with the slider or by entering a value in the edit-box. A value of 1000 means full intensity at the transmitter unit, a value of 0 sets the lowest intensity at the transmitter.
POWER only is available with **CALIB=OFF**.

LED MODE:
This item serves for setting the control mode for the integrated light source of the sensor.
DC: In this mode the sensor operates extremely fast. Unfortunately the sensor is somewhat sensitive to extraneous light in DC mode, but if the extraneous light source does not directly shine into the sensor's receiver, the signal only is influenced to a very small extent.
AC: In this mode the sensor is insensitive to extraneous light, which is achieved by "modulating" the integrated light source, i.e. by turning the light on and off. The extraneous content in the signal is determined in off status and is simply subtracted from the on status.

GAIN:
This item is used for setting the gain of the receiver in 8 different gain stages (AMP1 to AMP8). **GAIN** should be set such that with a medium **POWER** value the sensor operates in its dynamic range (X,Y,Z between 2750 and 3750). In **AC** mode, **GAIN** directly influences the scan frequency. The current scan frequency is displayed in the **SCOPE** tab.
GAIN only is available with **CALIB=OFF**.

AVERAGE:
This function field is used for adjusting the number of scanning values (measurement values) over which the raw signal measured at the receiver is averaged. A higher **AVERAGE** default value reduces noise of the raw signals at the receiver unit and there will be a decrease of the maximal available switching frequency of the sensor

INTEGRAL 1:
This function field is used to set the number of scan values (measurement values) over which the raw signal measured at the receiver is summed up. This integral function allows the reliable detection even of extremely weak signals. A higher **INTEGRAL 1** value increases the noise of the raw signals of the receiver unit, and simultaneously decreases the maximum achievable switching frequency of the sensor.
INTEGRAL 1 only is available with **CALIB=OFF**.

INTEGRAL 2:
This function field is used to set the number of scan values (measurement values) over which the raw signal measured **and averaged** at the receiver is summed up. This integral function allows the reliable detection even of extremely weak signals. A higher **INTEGRAL 2** preset value increases the noise of the raw signals of the receiver unit, and simultaneously decreases the maximum achievable switching frequency of the sensor.
INTEGRAL 2 only is available with **CALIB=OFF** or **XYZ OFFSET**.
In addition, **AVERAGE** must be higher than 1, so that a summation is possible at all.

INFO:
The **POWER** slider, **GAIN** and **INTEGRAL** only are effective with **CALIB = OFF** and **POWER MODE = SINGLE.** Otherwise the sensor operates with the values set in calibration.

SET DP:

If an application involves both very bright and very dark surfaces, and the sensor is set such that it is not in saturation at the bright surfaces, the dark surfaces frequently provide only a very low signal.

Little variations with a very low signal lead to a large change in color space calculation **(C-SPACE).** To minimise this effect, the sensor in **POWER MODE = DOUBLE** automatically switches back and forth between 2 **DOUBLE PARAMETER sets**. Set 2 must be parameterised such that the signal is maximally amplified.

When the signal is too high (saturation), it automatically switches to set 1.

To get correct color space coordinates with set 2, a correction factor must be used for the signal.

The two **DOUBLE PARAMETER sets** must be set at the sensor in advance.

When you click on **SET DP** (Set Double Parameter) a window will open at the right side, where you can set two sets. A Double Parameter set is defined by **POWER, GAIN** and **INTEGRAL**.

Automatic setting of DOUBLE parameter sets:
Pressing START ASSISTANT starts a menu guidance for automatically setting of the Double Parameter sets.
Just follow the instructions.

Manual setting of DOUBLE parameter sets:
Set **CALIB=OFF** and **POWER MODE=SINGLE**.
Place the brightest surface before the sensor, and select suitable **POWER, GAIN** and **INTEGRAL** values so that the brightest channel is at approx. 3000 digits.
Then click on **ASSIGN 1** to write the power parameters to row 1 in the **PP** table.
The X Y Z values are entered in the **XYZ VALUES** table.
Now place the darkest surface before the sensor.
The dark surface should be such that the lowest channel provides a signal higher than 100 digits.
Click on **ASSIGN 2**, and the X Y Z values are written to the DARK PP1 row in the **XYZ VALUES** table.
Now set **POWER, GAIN** and **INTEGRAL** such that the highest value is at approx. 3000.
Click on **ASSIGN 3** to write the power parameters to row 2 in the **PP** table, and the X Y Z values to the **XYZ VALUES** table.
The correction values are calculated automatically and are written to the **CORRECTION VALUES** table.

INFO!
The double parameter sets are first activated after pressing **SEND**!
The tables are for display purposes only. They cannot be edited.
The **DOUBLE PARAMETER SET** and **DP SET** displays show which DOUBLE PARAMETER set the sensor currently uses.
When the display shows 0, the sensor in **POWER MODE = SINGLE** operates with the settings in the **PARA1** tab.


To understand with which **X, Y** and **Z** values the sensor works in parameter set 2, you can display this raw data in the display **XYZ** using the **DP RAW DATA** check box.
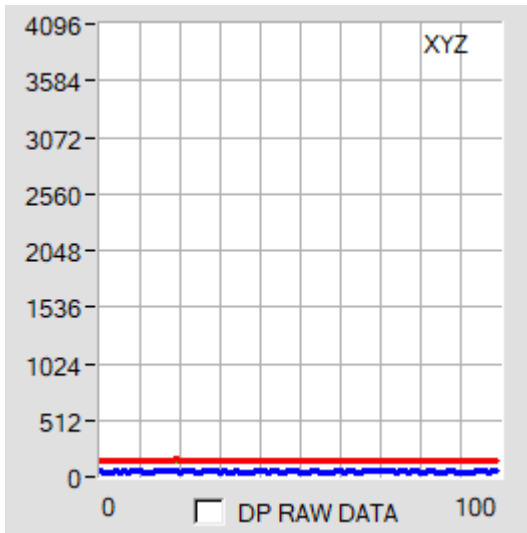
As mentioned above, the color space coordinates are calculated using the raw data **X, Y** and **Z** received with parameter set 2.
At the end of the calculation, they are divided using the correction factor.
This method avoids rounding errors in the calculation and thus maintains accuracy.

C SPACE	L* a* b* ▼
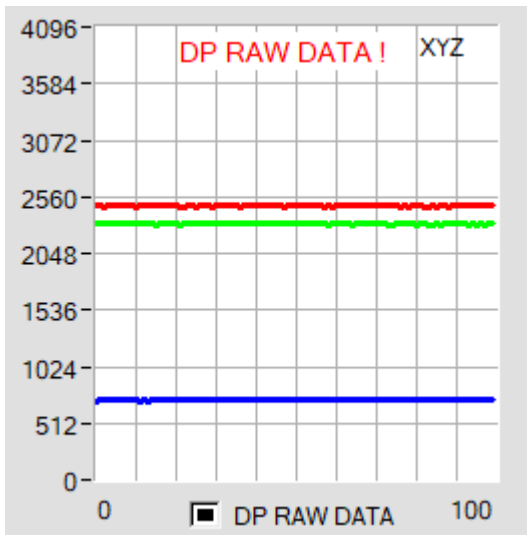
C SPACE:
Selection of a color space **(COLOR SPACE)** for the calculation of the color stimulus specification.

x y Y:
The chromaticity values (chromaticity coordinates) **x, y** are calculated from the tristimulus values X, Y, Z. **x** and **y** define the color, and **Y** is used as a brightness reference (lightness variable).

Lightness variable	$Y = \dfrac{Y}{4096}$	
Chromaticity coordinates	$x = \dfrac{X}{X + Y + Z}$	$y = \dfrac{Y}{X + Y + Z}$
Color difference	$\Delta E^* = \sqrt{(\Delta x)^2 + (\Delta y)^2 + (\Delta Y)^2}$	

L*a*b*:
The L*a*b* color space (also referred to as the CIELAB space) is one of the uniform color spaces defined by the CIE in 1976.

Lightness variable	$L^* = 116\left(\dfrac{Y}{Y_n}\right)^{1/3} - 16$	
Chromaticity coordinates	$a^* = 500\left[\left(\dfrac{X}{X_n}\right)^{1/3} - \left(\dfrac{Y}{Y_n}\right)^{1/3}\right]$	$b^* = 200\left[\left(\dfrac{Y}{Y_n}\right)^{1/3} - \left(\dfrac{Z}{Z_n}\right)^{1/3}\right]$
Color difference	$\Delta E^* = \sqrt{(\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2}$	

L*C*h*:
The L*C*h* color system is based on the same diagram as the L*a*b* system, but polar coordinates are calculated here. In the L*C*h* color system it is not possible to determine teach vectors, which is why **DIGITAL OUTMODE = OFF** is set and disabled.

Lightness variable	$L^* = 116\left(\dfrac{Y}{Y_n}\right)^{1/3} - 16$
Chroma	$C^* = \sqrt{(a^*)^2 + (b^*)^2}$
Hue angle	$h^* = tan^{-1}\dfrac{b^*}{a^*}$

L*u*v* and **L*u'v':**
The L*u*v* color space (also referred to as the CIELUV space) is one of the uniform color spaces defined by the CIE in 1976.

Lightness variable	$L^* = 116\left(\dfrac{Y}{Y_n}\right)^{1/3} - 16$	
Chromaticity coordinates	$u^* = 13L^*(u' - u_n')$ $\qquad$ $u' = \dfrac{4X}{X + 15Y + 3Z} = \dfrac{4x}{3 - 2x + 12y}$ $\qquad$ $u_n' = \dfrac{4X_n}{X_n + 15Y_n + 3Z_n} = \dfrac{4x_n}{3 - 2x_n + 12y_n}$	$v^* = 13L^*(v' - v_n')$ $\qquad$ $v' = \dfrac{9Y}{X + 15Y + 3Z} = \dfrac{9y}{3 - 2x + 12y}$ $\qquad$ $v_n' = \dfrac{9Y_n}{X_n + 15Y_n + 3Z_n} = \dfrac{9y_n}{3 - 2x_n + 12y_n}$
Color difference	$\Delta E^* = \sqrt{(\Delta L^*)^2 + (\Delta u^*)^2 + (\Delta v^*)^2}$	

X Y Z	Tristimulus values XYZ of the specimen.
Xn Yn Zn	Tristimulus values of a perfect reflecting diffuser.
$\Delta L^* \Delta a^* \Delta b^*$	Difference in L∗, a∗, and b∗ values between the specimen color and the target color.
$\Delta L^* \Delta u^* \Delta v^*$	Difference in L∗, u∗, and v∗ values between the specimen color and the target color.
u' v'	Chromaticity coordinates from the CIE 1976 UCS diagram.

u'n v'n	Chromaticity coordinates of a perfect reflecting diffuser.

CALIB OFF ▼	**CALIB:** The sensors are factory-calibrated.

CALIB=OFF can be selected if the sensor should operate in "uncalibrated" mode**.** This can be done if the sensor should operate fast. Both **FCAL** and **UCAL** will slow down the sensor.

If the **CALIB** parameter is set to **FCAL (F**actory **CAL**ibration), the sensor operates with this factory calibration.
As an alternative the user can himself calibrate the sensor to different surfaces (**U**ser **CAL**ibration).
If **CALIB=UCAL** is selected, the sensor operates with the calibration made by the user.

If you select **CALIB=UCAL WB** or **FCAL WB**, white light balancing can be performed through input IN0. For this purpose the reference surface used for calibration must be placed before the sensor, and input IN0 must then be activated.
The calculated white light calibration factors are displayed in the **CALIB FACTORS OF WHITE BALANCE** table in the **SENSOR DATA** tab.
To get the white light calibration factors for **CALIB=UCAL WB,** press **GET CF** in the **CALIB** tab.
To get the white light calibration factors for **CALIB=FCAL WB,** double-click with the right mouse button on the frame in which **GET CF** is contained, and enter GETCF as password.
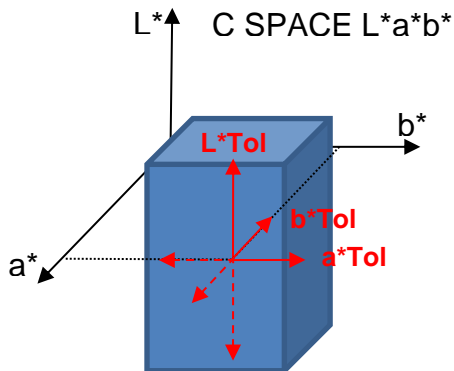With **CALIB=OFF** the sensor operates in "uncalibrated" mode**.** In this mode the sensor is very fast. Both **FCAL**

With **CALIB = XYZ OFFSET** and **XYZ OFFSET IN0** one can determine an offset for X Y Z.
See tab **XYZOFF**.

SHAPE MODE:
This function field is used to specify whether the color space coordinates **(e.g. L* a* b*)** should be viewed as a block, cylinder, or sphere in space.

When viewed as a block, each color space coordinate has its own tolerance **(a*Tol, b*Tol, L*Tol)**.
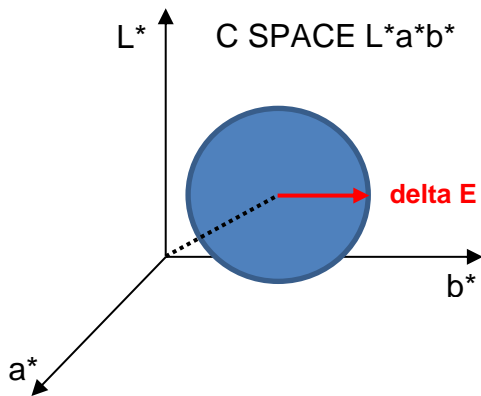The color distance **delta E** is calculated from the color coordinates **(a* b*)**.
A color is recognised if the current color space coordinates lie in the color block.
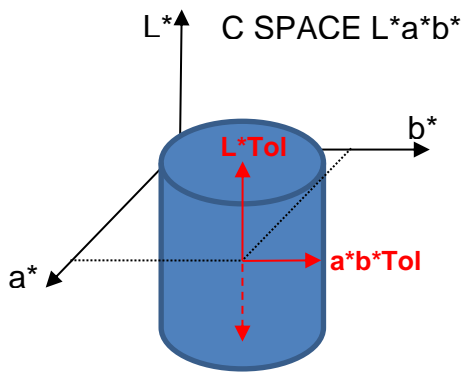
When viewed as a cylinder, the color coordinates **(a* b*)** have their own tolerance **(a*b*Tol),** and the brightness value **(L*)** has its own tolerance **(L*Tol)**.
The color distance **delta E** is calculated from the color coordinates **(a* b*)**.
A color is recognised if the current color space coordinates lie in the color cylinder.

In sphere mode there only is one tolerance for the color space coordinates.
The color distance **delta E** is calculated from the color space coordinates **(L* a* b*)**.
A color is recognised if the current color space coordinates lie in the color sphere.

EVALUATION MODE:
In this function field the evaluation mode at the SPECTRO-3-MSM-SLE color sensor can be set.

In the text below the **SHAPE MODE "SPHERE"** is used to explain the evaluation modes **FIRST HIT** and **BEST HIT**.

	a*	b*	L*	deltaE			
0	-11.35	13.99	50.85	5.00	0.00	0.00	🟥
1	-11.35	13.99	50.85	10.00	0.00	0.00	🟩
2	-11.35	13.99	50.85	15.00	0.00	0.00	🟦
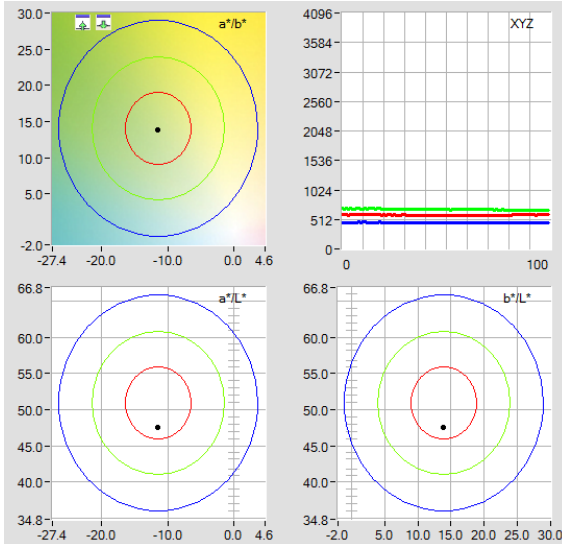
FIRST HIT:
The currently measured color values are compared with the default values in the **TEACH TABLE**, starting with teach-color 0. If in the row-by-row comparison the current color values correspond with the teach-parameters entered in the TEACH TABLE, this first "hit" in the TEACH TABLE is displayed as a color number (**C-No.**) and is output at the digital outputs according to the settings of the **OUTMODE** parameter (see **OUTMODE**).
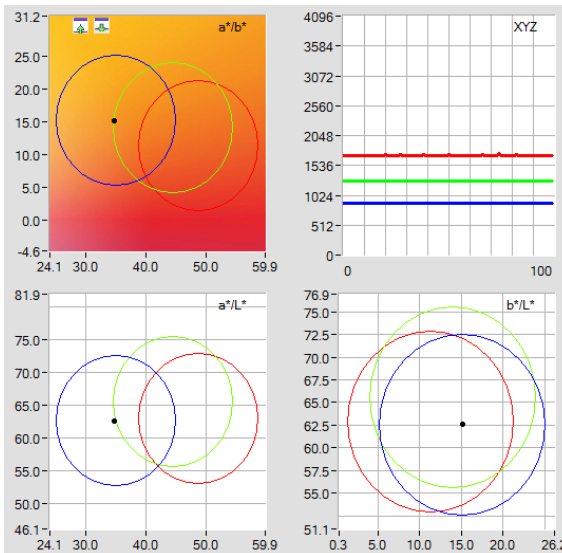If the current color does not correspond with any of the teach-colors, the color code **C-No.** = 255 will be set ("error status").

Tip! This mode is used if only one color is taught, and this color must be checked to avoid "drifting away". With the rising tolerance windows this can be detected very well, and countermeasures can be initiated, if necessary.

Tip! Values can be entered in a cell of the table either by double-clicking on the respective cell, or by highlighting the respective cell and then pressing F2 .

Tip! The teach colors are only activated when the **SEND** button is pressed!

	a*	b*	L*	deltaE			
0	48.89	11.34	62.84	10.00	0.00	0.00	🟥
1	44.65	14.01	65.48	10.00	0.00	0.00	🟩
2	35.10	15.17	62.53	10.00	0.00	0.00	🟦

BEST HIT:
The currently measured color values are compared with the default values in the **TEACH TABLE**, starting with teach-color 0. If in the row-by-row comparison the current color values correspond with several of the teach parameters entered in the color table, the teach parameter that has the shortest distance from the current color value will be a hit.
This "hit" in the TEACH TABLE is displayed as a color number (**C-No.**) and is output at the digital outputs according to the settings of the **OUTMODE** parameter (see **OUTMODE**).
If the current color does not correspond with any of the teach-in colors, the color code **C-No.** = 255 will be set ("error status").

Tip! This mode is used if several colors must be separated from each other, and only certain surface fluctuations are allowed.

Tip! Since what you are looking for here with several "hits" is the shortest distance of the current color to the centres of the taught colors, the individual tolerance windows (circles) may overlap here. The sensor detects the "**best hit**".

Tip! Values can be entered in a cell of the table either by double-clicking on the respective cell, or by highlighting the respective cell and then pressing F2.

Tip! The teach colors are only activated when the **SEND** button is pressed!
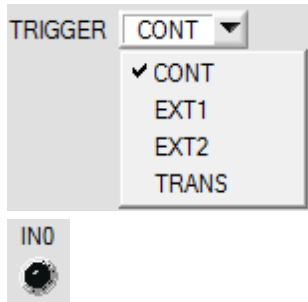
TRIGGER:
This function field serves for setting the trigger mode at the sensor. If **TRIGGER** is not **CONT**, the LED **IN0** shows a trigger event.

CONT: Continuous color detection (no trigger event required).

EXT1: Color detection is started through the external trigger input (IN0 pin3 green of cable cab-las8/SPS). A trigger event is recognized as long as +24V is present at the IN0 input (HIGH-active).
After the trigger input goes to LOW again, the state (C-No.) that was last detected will be held at the outputs.

EXT2: Same behaviour as in EXT1 mode, with the difference that an error state (color no. = 255) will be output after the trigger input goes to LOW again.

TRANS:
Same behaviour as in **CONT.** However, in contrast to **CONT** in this mode the transmitter source will be switched off if no trigger event (IN0) is given.
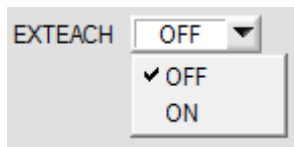
EXTEACH:
In all the evaluation modes teaching of a color can be performed externally through IN0.

OFF: The external TEACH feature is deactivated.
ON: The sensor can be taught up to 48 colors through IN0.

In **BEST HIT** evaluation mode teaching through IN0 can be performed to every individual row in the **TEACH TABLE**. With **FIRST HIT** the currently present color is taught to all the rows that are activated with **MAXCOL-No.**
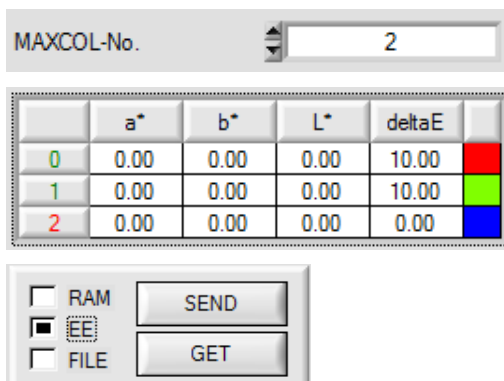
Select the function **EXTERN TEACH = ON**.
Set the power such that the sensor is not overdriven and that the signal also is not too low.
Select how many colors you want to teach externally.
Click on the **TEACH** tab to change to the **TEACH TABLE**.
Then enter the respective tolerances for the colors you want to teach.

	a*	b*	L*	deltaE	
0	0.00	0.00	0.00	10.00	
1	0.00	0.00	0.00	10.00	
2	0.00	0.00	0.00	0.00	

In this example **MAXCOL-No.** = 2 was selected. This means that the sensor should find the color information that is saved in the first 2 rows of the **TEACH TABLE** by way of external teaching through IN0. Since the sensor cannot calculate its **deltaE** tolerances itself, these values must be entered once (in this example 10 in all fields) and together with **MAXCOL-No.** and **EXTEACH = ON** must be saved in the **EEPROM**.

Then select **EE** in the function field, and click on **SEND**.
From now on, the PC is no longer necessary, as long as you only wish to teach colors up to the **MAXCOL-No.** and do not want to change the tolerances.

INFO:
Of course the taught colours can be viewed at any time with the PC.
Colors that are taught are stored in the **EEPROM** of the sensor, i.e. the information will <u>not</u> be lost when the system is turned off.
This function rather is intended for "after-teaching", because the sensor does not automatically adjust its **POWER** value.

EXTEACH=ON:

Before the external teach-process can be started, the color to be taught must be presented to the sensor.
The external teach process is started with a positive edge at IN0 (green wire). The outputs alternately change their state between 0V and 24V. From now on, the user has a certain number of seconds time to inform the sensor about the position at which the color information should be placed in the **TEACH TABLE**. The BUSY window is the time that the user has for informing the sensor about the teach rows. It depends on the value that is set in **MAXCOL-No.** (e.g. approx. 2 seconds when **MAXCOL-No.** = 2).

The first positive edge (start edge 0) selects position 0 in the **TEACH TABLE**.
Every additional positive edge selects one position higher (see table below).

Example:
If you wish to save the current color to position 3 in the **TEACH TABLE**, the following steps have to be performed:

1. Start the external TEACH process with a positive edge (0) at IN0 → Position 0 is selected.
2. Another positive edge (1) selects position 1 in the TEACH TABLE.
3. Another positive edge (2) selects position 2 in the TEACH TABLE.
4. Another positive edge (3) selects position 3 in the TEACH TABLE.
5. The desired position is now selected.
6. When the BUSY window is over, the sensor starts the evaluation.
7. To teach another color → go to pos. 1
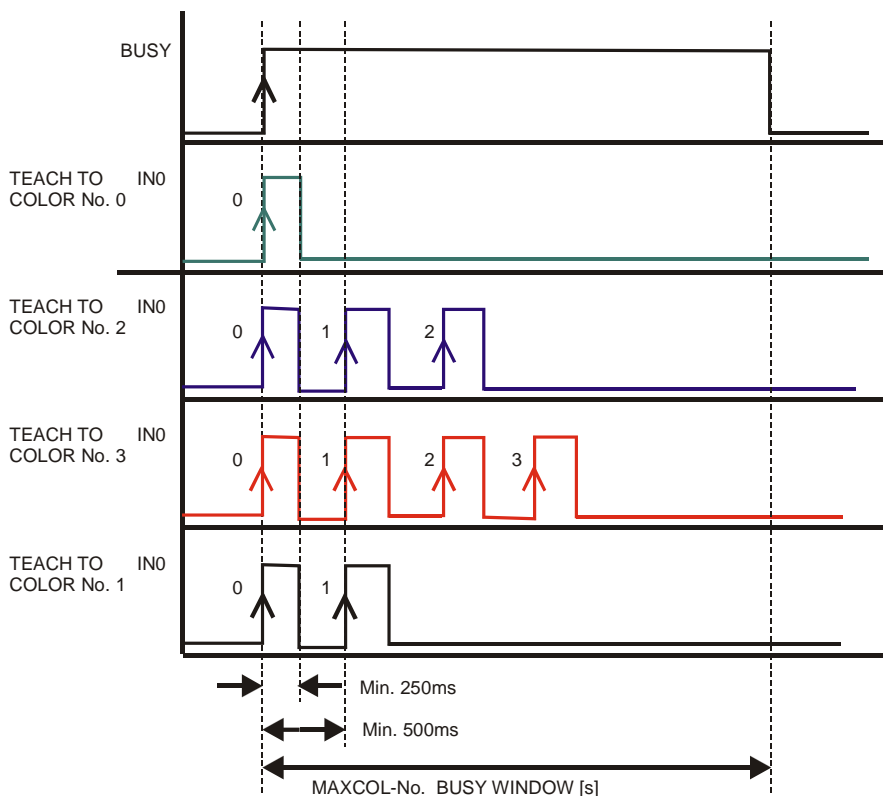
INTLIM	0

INTLIM:
An intensity limit can be set in this edit box. If the current intensity **(X+Y+Z)/3** arriving at the receiver unit falls below this limit, color evaluation will no longer be performed, and an error state will be output.

Attention: ERROR STATE if: (X+Y+Z)/3 < INTLIM

MAXCOL-No.	1

MAXCOL-No.:
This function field serves for setting the number of colors to be checked.

With **OUTMODE = BINARY HI** or **BINARY LO** a maximum of 31 colors or color groups can be taught.
With **OUTMODE = DIRECT HI** or **DIRECT LO** a maximum of 5 colors or color groups can be taught.
If **COLOR GROUPS = ON**, the sensor can be taught up to 48 colors.
However, only 31 groups can be formed with **OUTMODE = BINARY** and 5 groups with **DIRECT HI** or **LO**. In the **TEACH TABLE** rows 31-47 therefore have gray background.
The numerical value that is set here determines the current possible scan rate of the color sensor. If the number of colors to be checked is lower, the sensor operates faster. The numerical value that is defined here refers to the number of rows (starting with row 0) in the **TEACH TABLE** ($\rightarrow$ TEACH TABLE tab).

DIGITAL OUTMODE	DIRECT HI ▼

DIGITAL OUTMODE:
This group of buttons offers the method of how to control the 5 digital outputs.

OFF:
No digital signal is output. All the corresponding function fields are hidden.

DIRECT:
In this mode a maximum of 5 teach colors or 5 color groups is permitted.
If in this row-by-row comparison the current color values correspond with the teach-in parameters entered in the TEACH TABLE, this color in the TEACH TABLE is displayed as a color number (C-No.) and is sent **direct** to the digital outputs.

DIRECT HI:
If **DIRECT HI** is activated, the respective digital output is set to HIGH. If the current color does not correspond with any of the teach-in colors, all digital outputs are set to LOW.

DIRECT LO:
If **DIRECT LO** is activated, the specially digital output is set to LOW, while the other ones are set to HIGH. If the current color does not correspond with any of the teach-in colors, all digital outputs are set to HIGH.

BINARY HI:
If in the row-by-row comparison the current color values correspond with the teach-parameters entered in the color table, this "hit" in the color table is displayed as a color number (C-No.) and is provided at the digital outputs as a **bit pattern**.
In this mode a maximum of 31 teach colors or 31 color groups is permitted.

BINARY LO:
Same function as **BINARY HI,** but the **bit pattern** is output inversely.

2.3 Tab PARA2 (color groups and HOLD time settings)
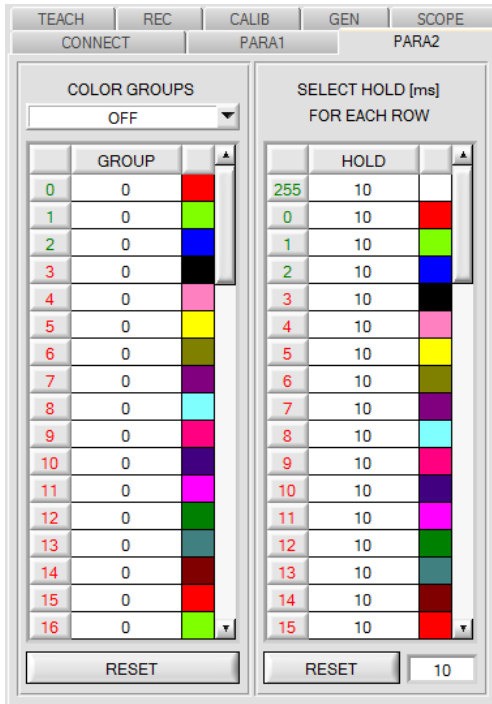
PARA2:
A click on **PARA2** opens a view where colors in the **TEACH TABLE** can be assigned to certain groups, and for each color and the error status an explicit **HOLD** time can be set.

Please note:
The sensor must be informed about the color groups and HOLD settings by pressing SEND.

COLOR GROUPS:
By means of the GROUP table individual rows can be assigned to a group.

In this example, **COLOR GROUPS** has been set to **ON**, i.e. group evaluation is activated.
Rows 0 and 1 have been assigned to group 0.
Rows 2 and 3 have been assigned to group 1, and row 4 to group 2.
A **GRP** display is shown under the **C-No:** display.

If, as in this example, row 3 is detected in the evaluation, this row and the corresponding group will be visualised.

The group number will be output through the outputs.

A total number of 48 different colors can be taught.
In evaluation mode **DIRECT HI** and **DIRECT LO** a maximum of 5 groups can be formed (group 0 to 4).
In evaluation mode **BINARY HI** and **BINARY LO** 31 groups can be formed.

All the cell values can be set to 0 by pressing **RESET**.
A double-click on the **GRP** display opens a larger display window.

HOLD in milliseconds [ms]:
The sensor operates with minimum scanning times in the magnitude of less than 100µs. This is why most of the PLCs that are connected to the digital outputs have difficulties with the safe detection of the resulting short switching state changes. Entering values in the table provides pulse lengthening at the digital outputs of the sensor system up to 100 ms. Each individual row can be assigned its own HOLD time.
When the **RESET** button is pressed, the complete table is set to the reset value indicated beside the **RESET** button.

Example:
Print marks are transported at a very high speed. The print marks are positioned in the sensor's range only for a very short time (milliseconds or even shorter). A minimum pulse of 10 ms is required for the reliable detection of a mark.
The background is taught to row 0, and the print mark to row 1. This is desirable because it allows the use of the **BEST HIT** evaluation mode.
If row 0 and row 1 overlap seamlessly, the sensor operates perfectly. The output is immediately activated and remains active at least for the time specified in the corresponding row.
If row 0 and row 1 do not overlap, the sensor detects an error at the transition from row 0 to row 1 (print mark), which immediately is sent to the output and remains active there at least for the **HOLD** time. An error inevitably would occur here if **HOLD** for the error state (255) was chosen too high, because the sensor only outputs a status change again after **HOLD** has elapsed (this may be the print mark, but it does not have to be).
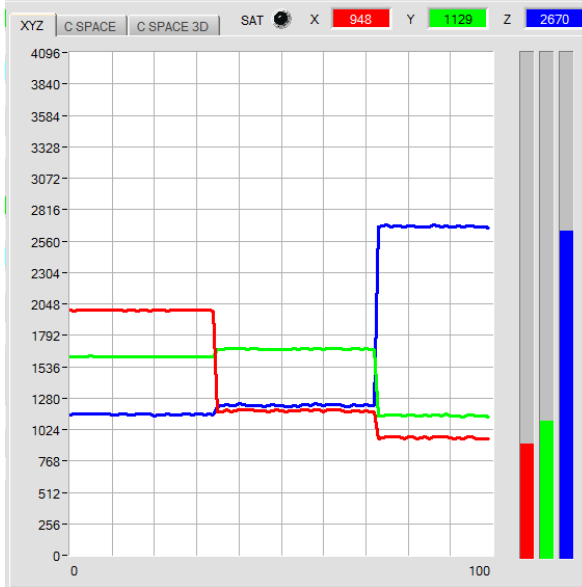It is absolutely necessary here to select a **HOLD** time of 0 for error state 255.

2.4 Graphical display elements

Tab **XYZ:**
Display of the current raw signals (tristimulus values) **X, Y, Z** of the 3-fold receiver.
The LED **SAT** indicates if one of the channels is in saturation. In this case, it lights up red.
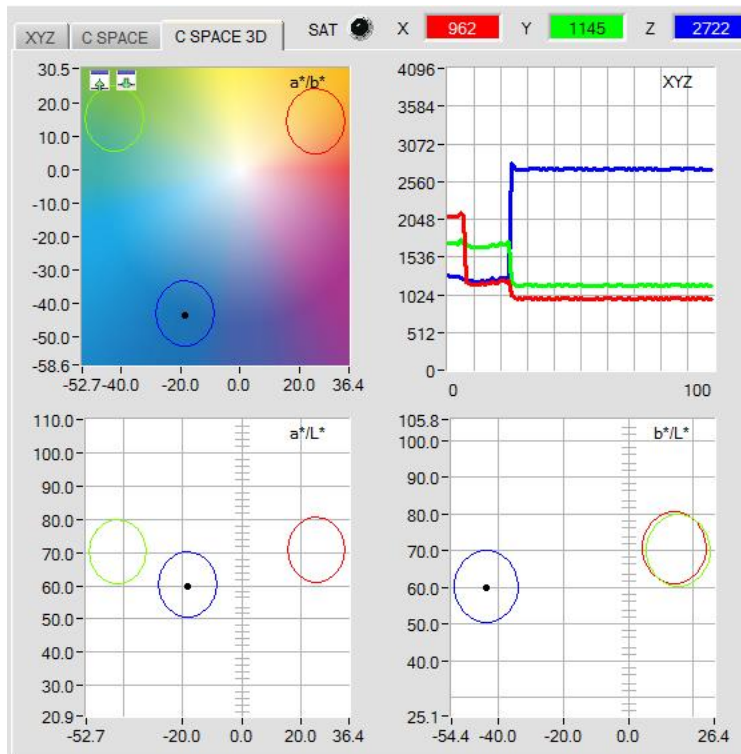
Tab **C SPACE:**
Line graph display of the color coordinates of the color space selected under **C SPACE**.

Tab **C SPACE 3D:**
Three-side view of the taught color in space.
The taught color space coordinates with their tolerances and the current color position are displayed. A three-side view in space was chosen to provide a clearer representation.

A left double-click in one of the graphs causes all the circles to be displayed in grey, only the circle with the recognised color is displayed in color.
A single left click returns to the standard view.

A left double-click in the XYZ graph starts an automatic zoom function.
A single left click closes this zoom function again

The graph with the color chart in the background can be zoomed in and out.

These numerical value displays show the color coordinates and brightness values that are calculated from the tristimulus values **X, Y, Z**.
Calculation is performed as described in **C SPACE.**

a*	u*	u'	C*	x
-12.98	-17.22	0.3196	35.17	0.2645
b*	v*	v'	h*	y
-8.18	-15.34	0.4736	14.43	0.3029
L*	L*	L*	L*	Y
67.63	54.01	50.10	32.31	0.2007

These numeric value displays show the deviations of the individual color coordinates and of the brightness value of the current surface from the "color hit" (C-No:) in the **TEACH TABLE**.
The values are calculated on the PC user interface and not in the sensor, and are only shown in these displays.

delta a*	delta u*	delta x
0.10	0.01	0.0004
delta b*	delta v*	delta y
0.08	0.15	0.0097
delta L*	delta L*	delta Y
0.05	0.06	0.0007

delta E
0.12

delta E:
This display shows the color distance from a color hit.
Calculation is performed as described in **C SPACE**.
In **FIRST HIT** evaluation mode **delta E** is equal to the distance from a color hit. If there was no color hit, **delta E** is calculated in relation to the last valid color in the color table (depending on **MAXCOL-No**.)
In **BEST HIT** evaluation mode **delta E** also is equal to the distance from a color hit. However, if there was no color hit, **delta E** has the value -1.00.
This also applies if the current intensity **(X+Y+Z)/3** falls below the **INTLIM** parameter.

C-No:
0

C-No.:
This numerical value output field shows the currently detected color number according to the entry in the **TEACH TABLE**. The currently detected color number is sent to the digital outputs as a corresponding bit pattern.
A value of 255 means that none of the taught colors was detected.
A double-click on the display will open a larger display window.

IN0

IN0:
This LED display visualises the status of input IN0.
If 0 V is applied at the input, the LED is black.
If 24 V is applied at the input, the LED is green.
This display only is visible if the input is used.

TEMP	27

TEMP:
This display shows the temperature in the sensor housing.
The display does **NOT** show degrees Celsius or Fahrenheit.

Please note:	The above-mentioned output fields are only updated when data transfer between PC and sensor is active (GO button pressed).

2.5 Tab TEACH

When the **GO** button is pressed, data transfer from the sensor to the PC is started.
The tristimulus values **X, Y, Z** and the calculated color coordinates and brightness values are shown in the displays and in the graphic windows.
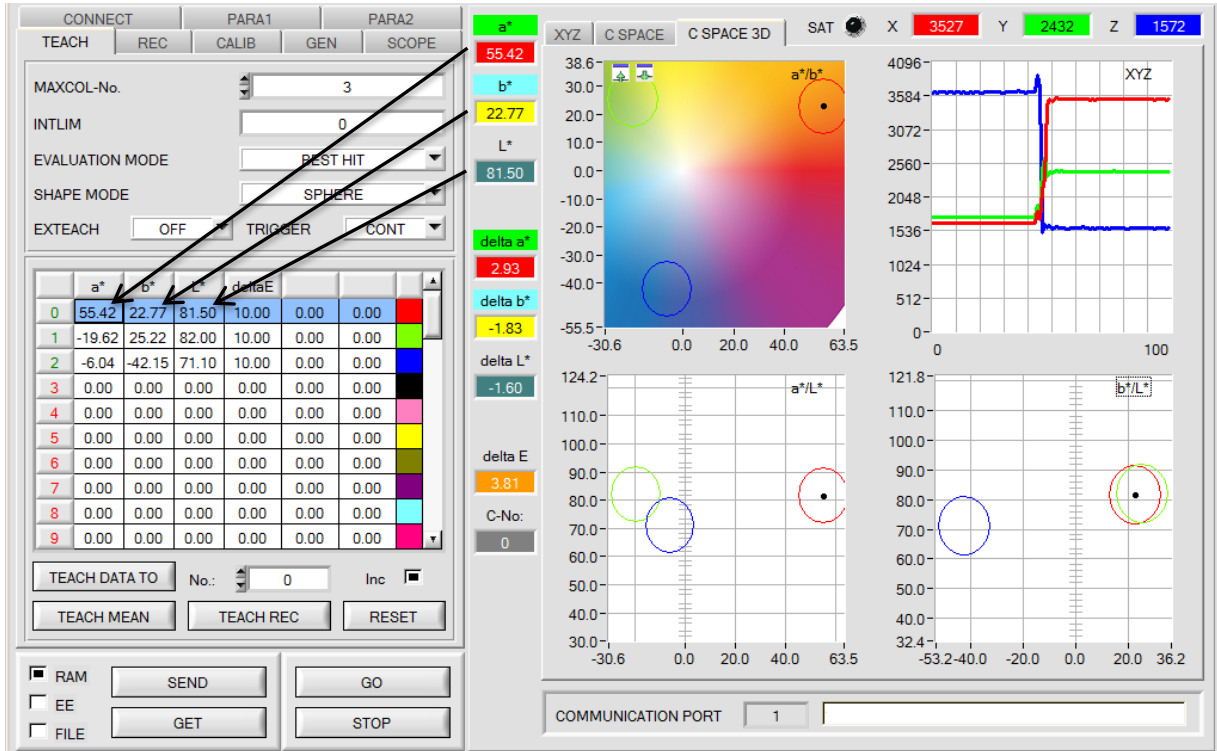
	a*	b*	L*	deltaE			
0	-20.14	50.32	92.16	10.00	0.00	0.00	
1	18.46	34.11	84.40	10.00	0.00	0.00	
2	16.36	56.65	72.29	10.00	0.00	0.00	
3	0.00	0.00	0.00	0.00	0.00	0.00	
4	0.00	0.00	0.00	0.00	0.00	0.00	
5	0.00	0.00	0.00	0.00	0.00	0.00	
6	0.00	0.00	0.00	0.00	0.00	0.00	
7	0.00	0.00	0.00	0.00	0.00	0.00	
8	0.00	0.00	0.00	0.00	0.00	0.00	
9	0.00	0.00	0.00	0.00	0.00	0.00	

MAXCOL-No.: 3
INTLIM: 0
EVALUATION MODE: BEST HIT
SHAPE MODE: SPHERE
EXTEACH: OFF TRIGGER: CONT

TEACH DATA TO   No.: 0   Inc:
TEACH MEAN   TEACH REC   RESET

TEACH:
Click on the **TEACH** tab to open a window where you can teach colors to the **TEACH TABLE**.

Please note: The teach colors must be sent to the sensor by clicking on SEND.

The **TEACH TABLE** is organised in rows, i.e. the individual parameters for the teach colors are listed side by side in the respective row.
The sensor can work with up to 48 teach colors. The number of the respective teach color is shown in the left column of the table.
Only rows with a green number are used for evaluation by the sensor. The number of rows to be used is defined with **MAXCOL-No.**.

After a click on **TEACH DATA TO** the currently displayed color coordinates and the brightness value are written to the row in the **TEACH TABLE** that is selected under **No.:**.
A color tolerance **(delta E)** also is defined. If necessary, the tolerances and the teach values can be changed by double-clicking on the respective field with the left mouse button (or by pressing F2) and then entering a numerical value with the PC keyboard.

When **Inc:** is activated, and the **TEACH DATA TO** button is pressed, the **No.:** input field is automatically incremented (increased) by 1, i.e. the next row in the **TEACH TABLE** is selected.

A click on the **RESET TABLE** button resets the **TEACH TABLE** (RESET value = 0).

	a*	b*	L*	deltaE			
0	-20.14	50.32	92.16	12.00	0.00	0.00	
1	18.46	34.11	84.40	12.00	0.00	0.00	
2	16.36	56.65	72.29	12.00	0.00	0.00	

It is possible to edit several cells at the same time.
For this purpose the cells that should be changed must be selected.

Help

Set selection to

Increment selection with

Reset selection

Teach tolerance off

Teach tolerance on

Right-click with the mouse to open a pop-up menu.

Use **Set selection to** if you wish to fill several cells in the **TEACH TABLE** with a certain value.

Use **Increment selection with** if you wish to enter increasing tolerance values.
The upper left cell is the starting value.
Starting from this cell the following cells will be increased by the value that is entered in the pop-up menu.

VALUE!

Insert a cell value!

12
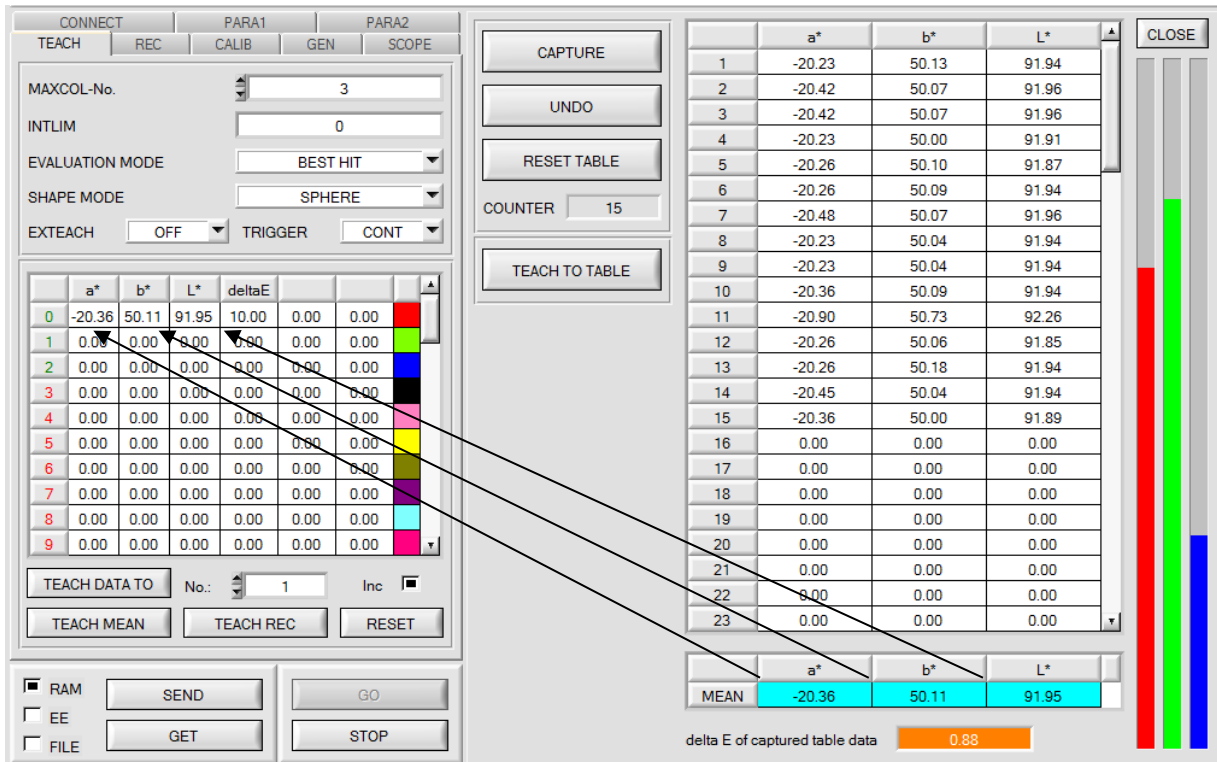
OK

Use **Reset selection** to set the selected cells to 0.

When **TEACH DATA TO** is used the software suggests tolerance values and writes these values to the **deltaE** column.
This function can be switched on and off with **Teach tolerance off** and **Teach tolerance on**.

A click on **Help** will open a pop-up window that provides explanations of the individual functions.

The following panel will be displayed after a click on **TEACH MEAN VAL**.

After a click on **GO,** data are automatically obtained from the sensor and displayed.
A click on the **CAPTURE** button enters a parameter frame in the table.
The **COUNTER** display field shown how many frames have already been recorded.
The last frames that were entered in the table can be deleted again by pressing **UNDO**.
RESET TABLE sets the whole table back to 0.
After every activation of **CAPTURE**, **UNDO**, or **RESET TABLE**, the mean values for the individual parameters are calculated automatically and are displayed in the mean value table.
Furthermore, a maximum color tolerance **delta E** is formed with respect to the mean values.

When the **TEACH TO TABLE** button is pressed, the respective mean values are taught to the row in the **TEACH TABLE** that is selected under **No.:**.

A click on the **CLOSE** button will take you back to the main panel.

A click on the button **TEACH REC VAL** (**teach recorded values**) opens a panel which, after a click on the **START** button, starts to record data and display them in the three graphs. This function is useful if the material to be detected cannot be directly placed in front of the sensor, for example because it is transported on a conveyor belt that cannot be stopped or cannot be exactly stopped at the position required for teaching. After some time, area centres will become apparent that can be assumed to be the objects to be taught. The screenshot below shows 3 of these positions.

After a click on **STOP** the cursor can then be positioned on such an area centre in any of the graphs. The cursor will be automatically positioned in the other two graphs.

With **TEACH TO TABLE** the current cursor position can then be transferred to the **TEACH TABLE**. Teaching will be performed to the row that is set under **No.:**.

RESET can be used to reset the graphs and the counter.
CLOSE takes you back to the main panel.

2.6 Tab GEN

TEMP	9

SET ROW COLORS **MANUALLY**

SELECT ROW 0   ROW COLOR

ASSIGN TO ROW	RESET
SAVE TO FILE	GET FROM FILE
SAVE TO SENSOR	GET FROM SENSOR

	SPECIFICATION	
0	Color 0	
1	Color 1	
2	Color 2	

More...
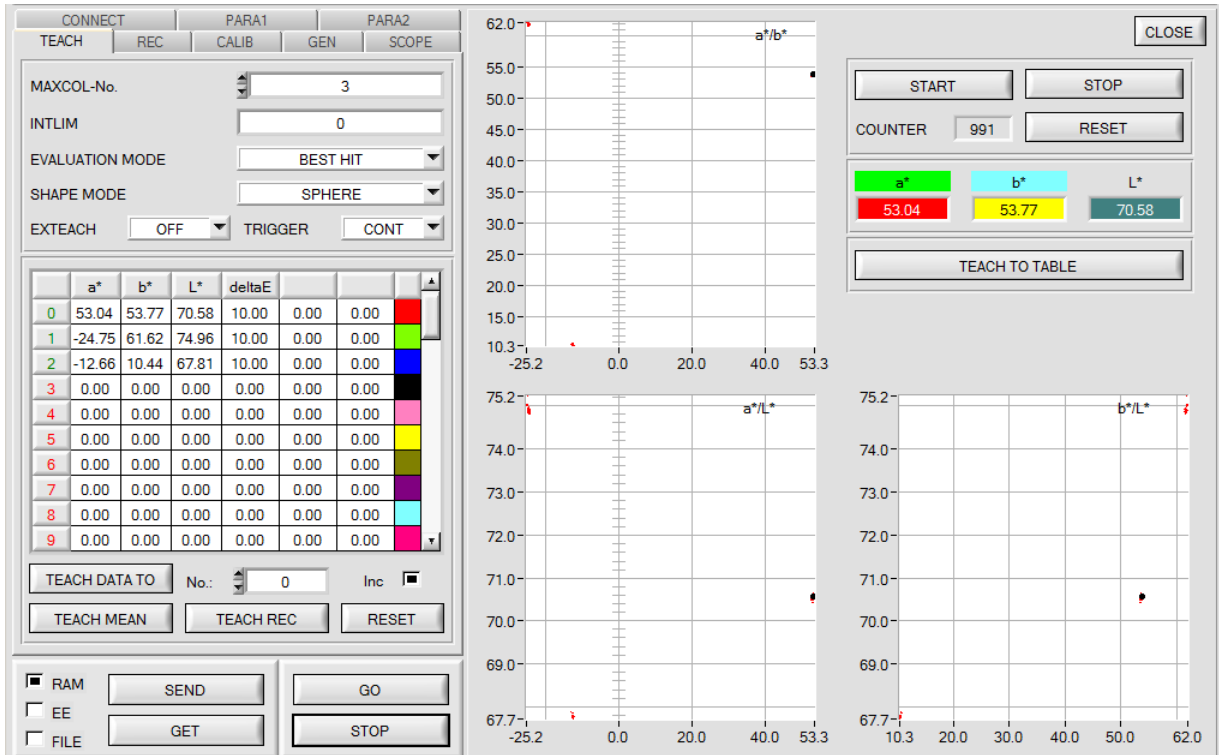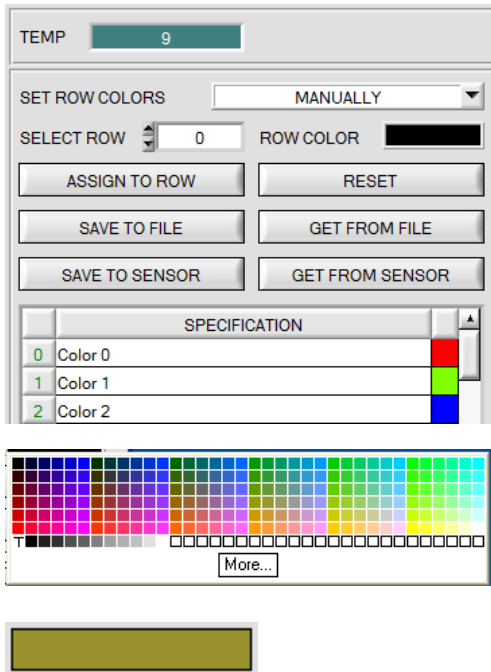
GEN:
A click on **GEN** opens a display of the current temperature value **TEMP** in the sensor housing.
The display is **NOT** in degrees Celsius or Fahrenheit.

In the **GEN** tab you also can manually select the row colors in which the individual tolerance circles are displayed, or let the system set these colors automatically based on the colors detected by the system.

Furthermore, designations for the individual rows can be entered in the **SPECIFICATION** table. These designations are saved on the PC's hard disk and can be loaded again, if necessary.

If **SET ROW COLORS** is set to **MANUALLY**, the row color that should be changed must be set under **SELECT ROW**.

A click on the colored area of **ROW COLOR** opens a color palette where the desired color can be selected.
After a click on the **ASSIGN TO ROW** button the color will be displayed in the 6th column and the selected row of the **TEACH TABLE**.

If **SET ROW COLORS** is set to **AUTOMATICALLY**, the system will automatically calculate the corresponding row color, display the color in a color display beside the graph and, after a click on **TEACH DATA TO**, automatically insert it in the corresponding row.

The **SAVE TO FILE** and **GET FROM FILE** functions allow you to save specified row color arrays and the **SPECIFICATION** table on the hard disk, or to load previously saved arrays.

SAVE TO SENSOR and **GET FROM SENSOR** writes these to the sensor or read these from the sensor.
For reasons of storage space, only the first 16 characters of **SPECIFICATION** are stored in the sensor.

A click on **RESET** resets the colors to a default value.

When the software is started again, the **ROW COLOR** table and the **SPECIFICATION** table that were active when the software was closed are automatically loaded again.

2.7 Tab REC

The SPECTRO3-MSM-SLE-Scope software includes a data recorder that makes it possible to save the data acquired and calculated by the sensor with a time stamp. The recorded file is saved to the hard disk of the PC and can then be evaluated with a spreadsheet program.
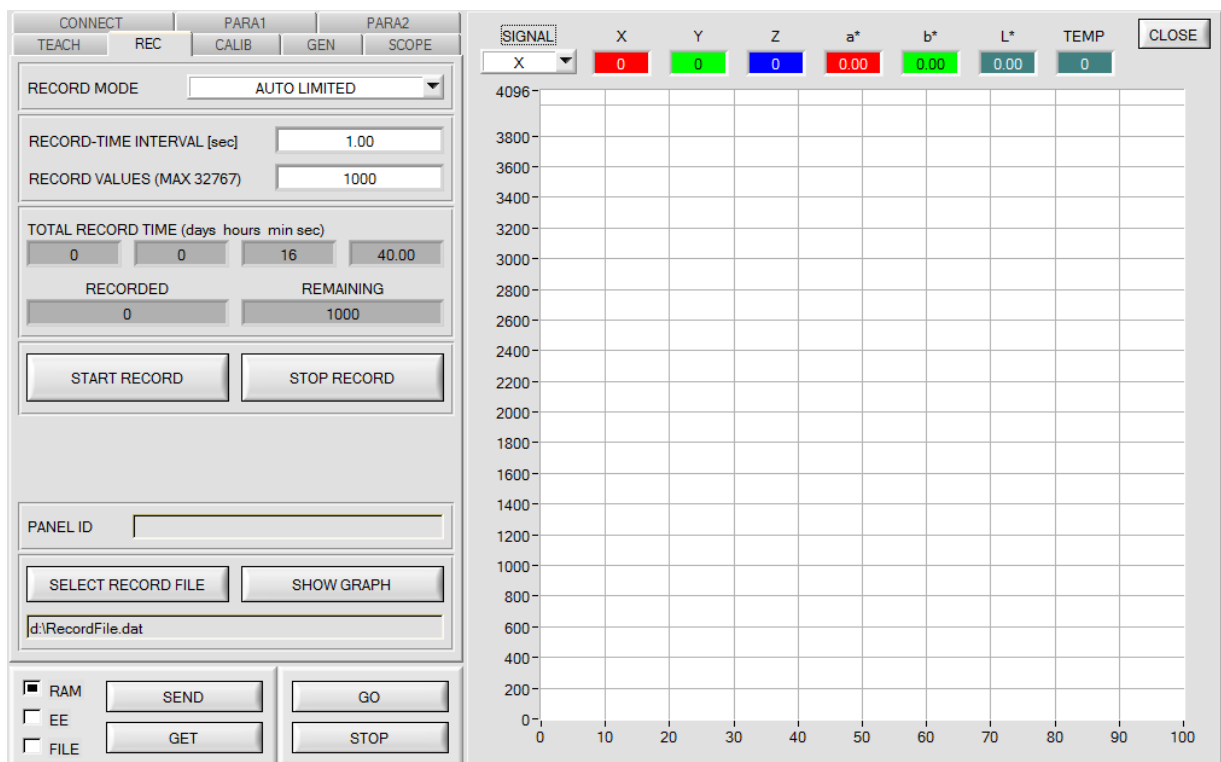
The following steps describe how data frames are recorded with the recorder:

Step 1:
When the **REC** button is pressed, the following window will be displayed:
When the **SHOW GRAPH** button is pressed, a panel will be displayed that allows the user to monitor the different signals.
The individual signals can be activated from the **SIGNAL** drop-down menu.

RECORD MODE	AUTO LIMITED ▼
RECORD-TIME INTERVAL [sec]	1.00
RECORD VALUES (MAX 32767)	1000

TOTAL RECORD TIME (days hours min sec)

0	0	16	40.00

RECORDED	REMAINING
0	1000

START RECORD   STOP RECORD

PANEL ID [   ]

SELECT RECORD FILE   SHOW GRAPH

d:\RecordFile.dat

Step 2:
If you want to automatically record several data frames, please select **AUTO LIMITED** under **RECORD MODE**. Enter a time interval for recording under **RECORD-TIME INTERVAL [sec]**, in this example: 1, i.e. a new value is called from the sensor every second).
Then enter the maximum number of values you wish to record in the **RECORD VALUES [MAX 32767]** field.
Please note: Recording can also be stopped earlier by clicking **STOP RECORD**, the data recorded so far will not be lost.

The **TOTAL RECORD TIME** field indicates how long recording will take (in days, hours, minutes, and seconds) if all data are recorded.

Step 3:
By pressing the button **SELECT RECORD FILE** a file can be selected in which the data frame will be stored.
If you select an already existing file name, you will be asked whether you want to overwrite the existing file or not.

Step 4:
Pressing the **START RECORD** button starts automatic data recording.

The recorder starts to record data, and the button **START RECORD** is red to indicate that recording is active. The respective data frames are shown in the display windows.

In the two display fields **RECORDED** and **REMAINING** you can check how many data frames have been recorded, and how many frames remain to be recorded.

Please note:
During recording the two input fields RECORD-TIME INTERVAL and VALUES TO BE RECORDED are inactive.

Step 5:
When as many data frames as set under **RECORD VALUES [MAX 32767]** have been recorded, or when the **STOP AUTO RECORD** button is pressed, a pop-up window will appear which confirms that the file is stored.

If you want to record an unlimited number of data, select the **AUTO UNLIMITED** function under **RECORD MODE**. Then select the desired recording interval and press **START RECORD**.

If you want to record data "manually", select the **MANUAL RECORDING** function under **RECORD MODE**.
You can start reading data from the sensor by pressing the **GO** button. These data are visualised in the display window. Pressing the **CAPTURE DATA FRAME** button saves a data frame in the file that was selected under **SELECT RECORD FILE**. The **RECORDED** field shows the sum of the frames already recorded.

If **AUTO TRIGGERED** is selected under **RECORD MODE**, and **TRIGGER = EXT1, EXT2,** or **TRANS** is selected, the sensor will automatically send a data frame after each drop of the trigger when **START RECORD** is pressed. This data frame is captured and recorded by the recorder.
Pressing the **STOP RECORD** button terminates the automatic sending function of the sensor again.

Please note:
When you press START RECORD, the file that is selected under SELECT RECORD FILE will be deleted. With RECORD FRAME MANUALLY, the file will be created if it does not already exist. If the file already exists, the data are added to the existing file.
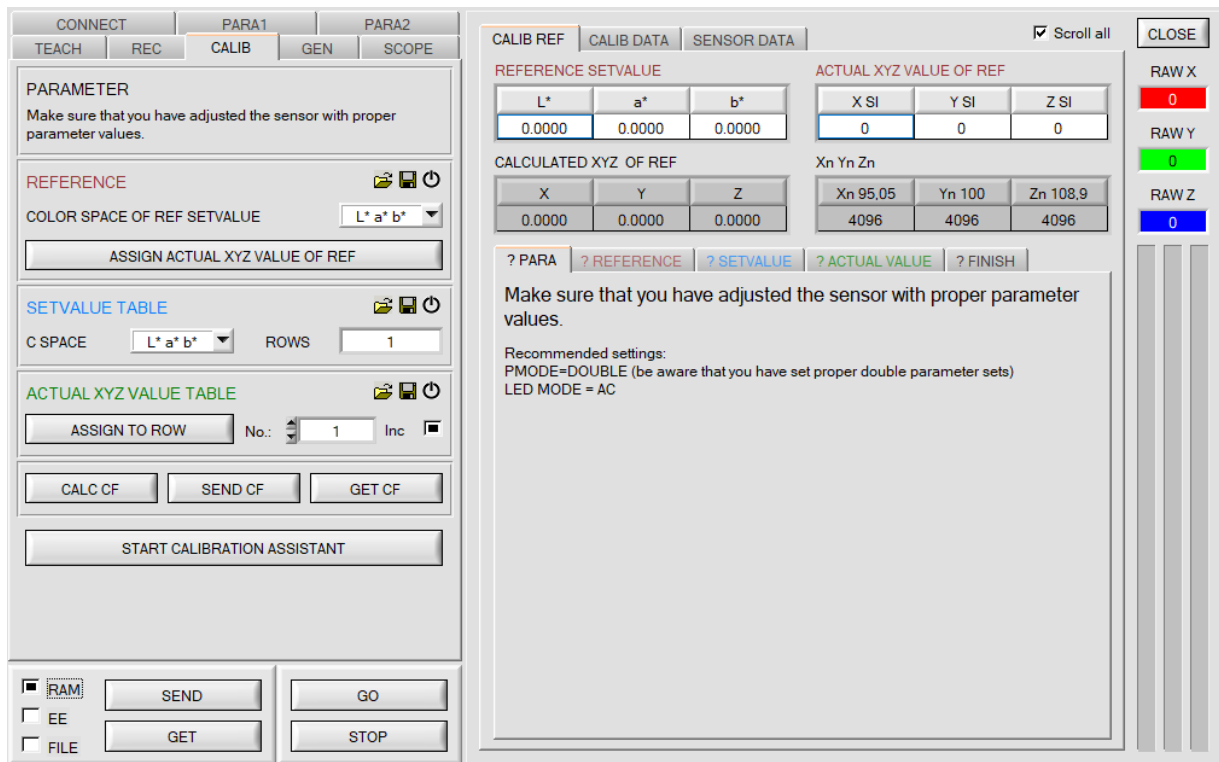
2.8 Tab CALIB

2.8.1 Calibration

The sensors are factory-calibrated. If the **CALIB** parameter is set to **FCAL (F**actory **CAL**ibration), the sensor operates with this factory calibration.
As an alternative the user can himself calibrate the sensor to different surfaces (**U**ser **CAL**ibration).
If **CALIB=UCAL** is selected, the sensor operates with the calibration made by the user.

Calibration is performed by using different surfaces with known color coordinates and brightness values. A ColorChecker™ table is available as an alternative. This table has 24 color fields according to the CIE STANDARD.

A click on **CALIB** will open the following window:

With **START CALIBRATION ASSISTANT** you can start an assistant that guides you through the calibration.

The following three conditions must be met:

1. You need **one** reference card. This is either the card RAL 9003-P, RAL 9010-P, or RAL 9016-P.

2. You need different RAL colors RAL from the *RAL CLASSIC* or *RAL DESIGN SYSTEM plus* series. These are available from the sensor supplier.

3. You need a file in which the L*a*b* values for the different RAL cards are stored. This file is also available from the supplier.

The following describes how the calibration is carried out without assistant on any surface of which the color coordinates and the brightness value are known.

REFERENCE		
COLOR SPACE OF REF SETVALUE	L* a* b*	▼
ASSIGN ACTUAL XYZ VALUE OF REF		

REFERENCE SETVALUE		
L*	a*	b*
0.0000	0.0000	0.0000

CALCULATED XYZ OF REF		
X	Y	Z
0.0000	0.0000	0.0000

ACTUAL XYZ VALUE OF REF		
X SI	Y SI	Z SI
0	0	0

Xn Yn Zn

Xn 95,05	Yn 100	Zn 108,9
4096	4096	4096

Please perform the following steps in sequence:

- Place a white reference surface before the sensor.
- Click on **GO** to start data exchange between sensor and PC software.
- **POWER, GAIN,** and **INTEGRAL** must be set such that the highest signal of **RAW X, Y, Z** is approx. 3000 digits.
- It is important to select the correct color space **C SPACE OF REF SETVALUE**.
- The color coordinates and the brightness value of the reference surface must be known and must be entered in the **REFERENCE SETVALUE** table. The corresponding tristimulus values **X, Y, Z** are calculated automatically and are written to the **CALCULATED XYZ OF REF** table.
- With **ASSIGN ACTUAL XYZ VALUE OF REF** the tristimulus values **RAW X, Y, Z** acquired by the sensor are written to the **ACTUAL XYZ VALUE OF REF** table, and the tristimulus values **Xn, Yn, Zn** of a completely matt white body are calculated automatically.
- **Xn, Yn, Zn** are needed for calculating the color coordinates and the brightness value in almost all color spaces.

Use the file and disk symbol if you want to write the REFERENCE settings to a file or read them from a file. A click on the reset symbol will reset the tables.

SETVALUE TABLE		
C SPACE	L*a*b* ▼	ROWS 24

SETVALUE TABLE

	L*	a*	b*
1	38.0800	12.0900	14.3900
2	66.3800	13.2200	17.1400
3	51.0600	0.3800	-22.0600
4	43.3000	-16.5200	21.4600
5	56.3600	12.8400	-25.2900
6	71.6000	-30.7100	1.1700
7	61.7000	27.5400	58.2300
8	41.2200	17.9500	-43.1600
9	51.5700	43.0000	14.7500
10	30.7700	25.7400	-23.3800
11	72.5800	-28.1700	58.6100
12	71.6000	12.4500	66.5800

Several different surfaces are needed for sensor calibration.

The number of surfaces must be entered in **ROWS**.

The color coordinates and brightness values must be known.

These must be entered in the **SETVALUE TABLE**.

The corresponding color space must be selected under **C SPACE**.

Use the file and disk symbol if you want to write the **SETVALUE** settings to a file or read them from a file. A click on the reset symbol will reset the tables.

ACTUAL XYZ VALUE TABLE

| ASSIGN TO ROW | No.: | 1 | Inc |

ACTUAL XYZ VALUE TABLE

	X	Y	Z
1	394	345	248
2	1290	1224	913
3	641	760	1173
4	389	454	237
5	853	911	1488
6	1166	1633	1492
7	1313	929	293
8	467	518	1338
9	901	607	524
10	280	248	495

Now the surfaces corresponding with the **SETVALUES** must be placed before the sensor.

With **ASSIGN TO ROW** the tristimulus values **RAW X, Y, Z** that are currently determined by the sensor are written to the row in the **ACTUAL XYZ VALUE TABLE** that is selected under **No.:**.

Use the file and disk symbol if you want to write the **ACTUAL XYZ VALUE TABLE** to a file or read it from a file. A click on the reset symbol will reset the tables.

| CALC CF | SEND CF | GET CF |

With a click on the **CALC CF** button the corresponding calibration factors that are determined from the tables **SETVALUE TABLE** and **ACTUAL XYZ VALUE TABLE** are calculated.

SEND CF saves the **ACTUAL XYZ VALUE TABLE** and the calculated calibration factors in the sensor's EEPROM.

A click on **GET CF** will display the tables saved in the sensor and the reference parameters in the **SENSOR DATA** tab.

With **CALIB=UCAL WB** or **FCAL WB** white light balancing can be performed through input IN0. For this purpose the reference surface used for calibration must be placed before the sensor, and input IN0 must then be activated. The calculated white light calibration factors are displayed in the **CALIB FACTORS OF WHITE BALANCE** table in the **SENSOR DATA** tab.
To get the white light calibration factors for **CALIB=UCAL WB,** press **GET CF** in the **CALIB** tab.
To get the white light calibration factors for **CALIB=FCAL WB,** double-click with the right mouse button on the frame in which **GET CF** is contained, and enter GETCF as password.

2.8.2 Offset calibration

To avoid an increase of the electronic offset when using the integral function (**INTEGRAL** parameter), this offset can be eliminated by way of offset calibration or zero-point calibration. The corresponding tab is password-protected to prevent inadvertent incorrect settings.
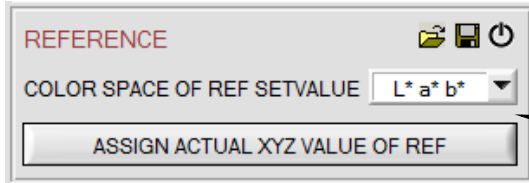
Offset calibration can be accessed by double-clicking with the right mouse button exactly at any frame in the **CALIB** tab.

e.g. here:
Double-click with the right mouse button
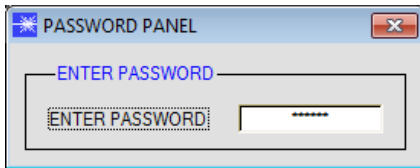
You will then be prompted to enter a password.
The password is: mellon
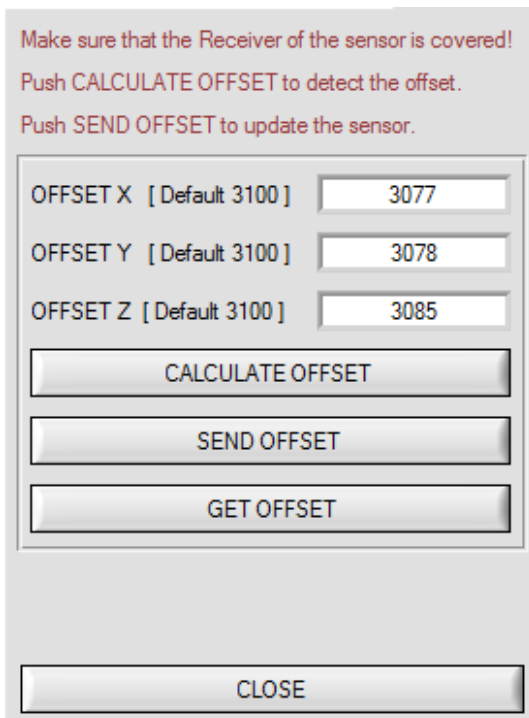
Now please follow the instructions provided in the tab.

ATTENTION!
It is extremely important for offset calibration that the receiver is not exposed to any extraneous light.
Please make sure that you cover the receiver of the sensor e.g. witch a black cloth that is impervious to light.

This is absolutely necessary for proper offset calculation.

Now press **CALCULATE OFFSET**. The offset values for red, green, and blue should be approximately 3080 plus/minus 40.

The offset values can then be sent to the sensor by pressing **SEND OFFSET**.
GET OFFSET can be used to check whether the data have been sent.

2.9 Tab SCOPE

The SCOPE tab visualises an oscilloscope.
Depending on **TRIG MODE,** the display shows the **XYZ** signals, the color space, the output analog signal
(depending on the **ANA OUT** settings), and the status of the digital outputs.
A click on **GET CYCLE TIME** displays the current sensor scan frequency in **[Hz]** and **[ms]**. The current scan
frequency must be determined for the correct calculation of **deltaX[ms]**. For determining the correct scan frequency
please give the sensor a time of 8 seconds before you click on **GET CYCLE TIME**.
In **TRIG MODE = SINGLE SHOT** a data frame is recorded and displayed in the graph when you click on **SCAN**.
In **TRIG MODE = FALLING EDGE** and **RISING EDGE** triggered recording can be started by clicking on **SCAN**. A
trigger start can be defined with **TRIGGER LEVEL**.
Triggering is performed either to **Z** or to the brightness value of the color space, depending on the signal that should
be recorded. In the graph this is the blue line.
In **TRIG MODE = INTERN C-No.0** recording starts automatically when C-No. 0 is detected. **TRIG MODE= EXTERN
IN0** can be used to start recording externally through input IN0.
SCAN-RATE can be used to delay or accelerate recording. This corresponds with the TIMEBASE function known
in oscilloscopes. **PRE TRIGGER VALUES** can be used to define how many values should still be displayed before
the actual trigger start.
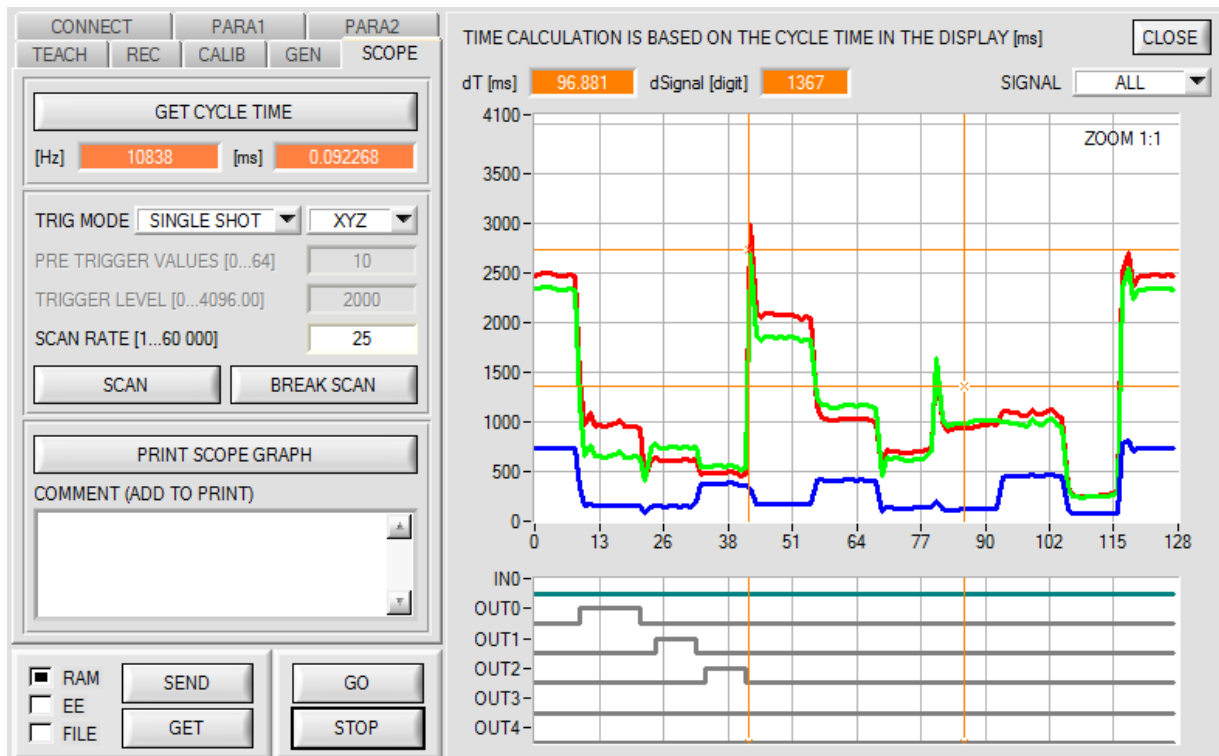
The zoom function in the graph can be activated by holding the control key (CTRL) and drawing a window with the
mouse.
A click on **ZOOM 1:1** cancels the zoom function again.

The two orange cursors can be moved with the mouse. The two displays **deltaX[ms]** and **deltaY[digit]** will be
updated.
deltaX[ms] shows the time between the cursors in X direction.
deltaY[digit] shows the difference between the two cursors in digits or in Volt in Y direction.

SIGNAL can be used to display individual curves.

PRINT SCOPE GRAPH prints the current screen together with the text in the **COMMENT** text field.

2.9 Tab XYZOFF

An offset for X Y Z can be defined with **CALIB = XYZ OFFSET** and **XYZ OFFSET IN0**.

For defining the offset, the surface that should be compensated must be placed under the sensor,
and **GO** mode must be activated.
Then the sensor parameters must be set.
In the **XYZOFF** tab the raw signal is shown in the corresponding displays and bars.
When you click on **ASSIGN XYZ OFFSET** the current raw signals are copied to the input fields.
With **SEND XYZ OFFSET** these values are sent to the sensor's EEPROM (non-volatile memory). With **GET XYZ OFFSET** the values can be loaded from the sensor.

The offset values are effective when they have been saved in the sensor and when **CALIB = XYZ OFFSET** has been selected.

If you select **CALIB = XYZ OFFSET IN0**, the offset values also can be updated through input IN0.
Attention! Through IN0 the offset values only are saved in the sensor's RAM (volatile memory).
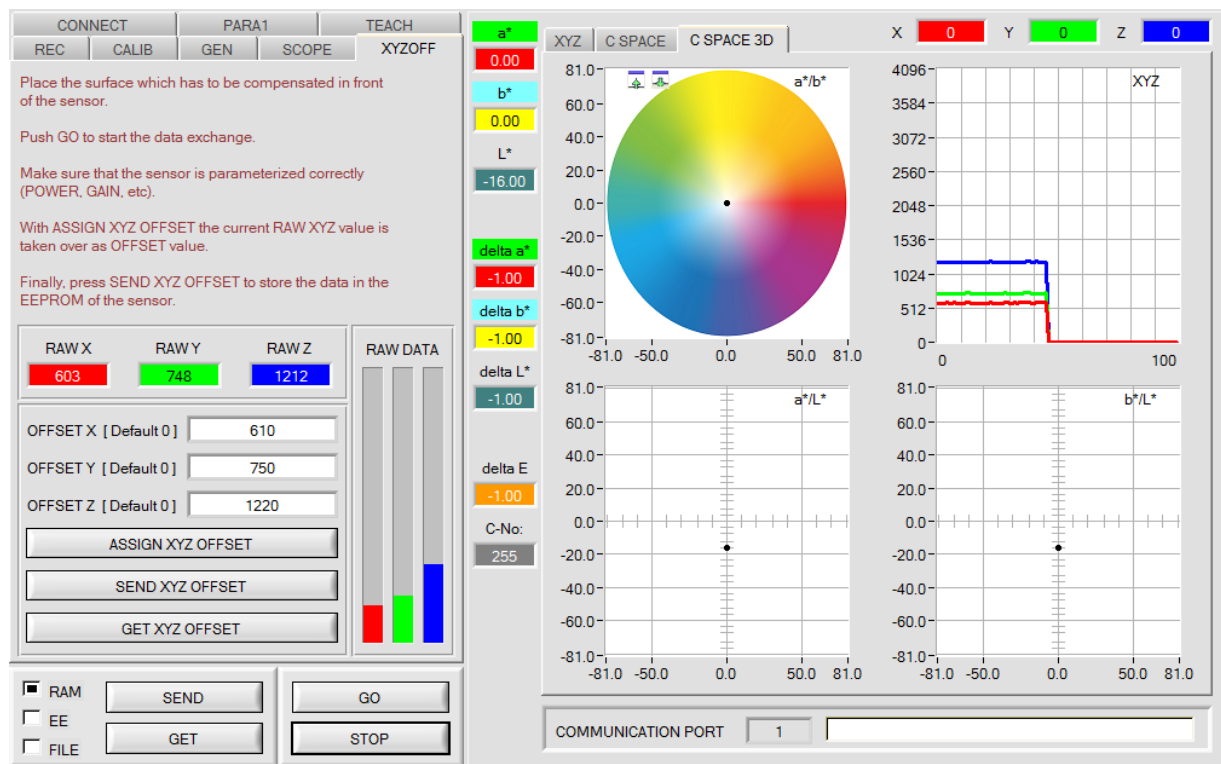
3. Operation of the TEMPCOMP-Scope software

If a firmware update should go wrong and the temperature characteristics that are stored in the EEPROM should be lost, these characteristics must be created anew. For this purpose you will need a file with the corresponding data. This file can be obtained from your supplier.
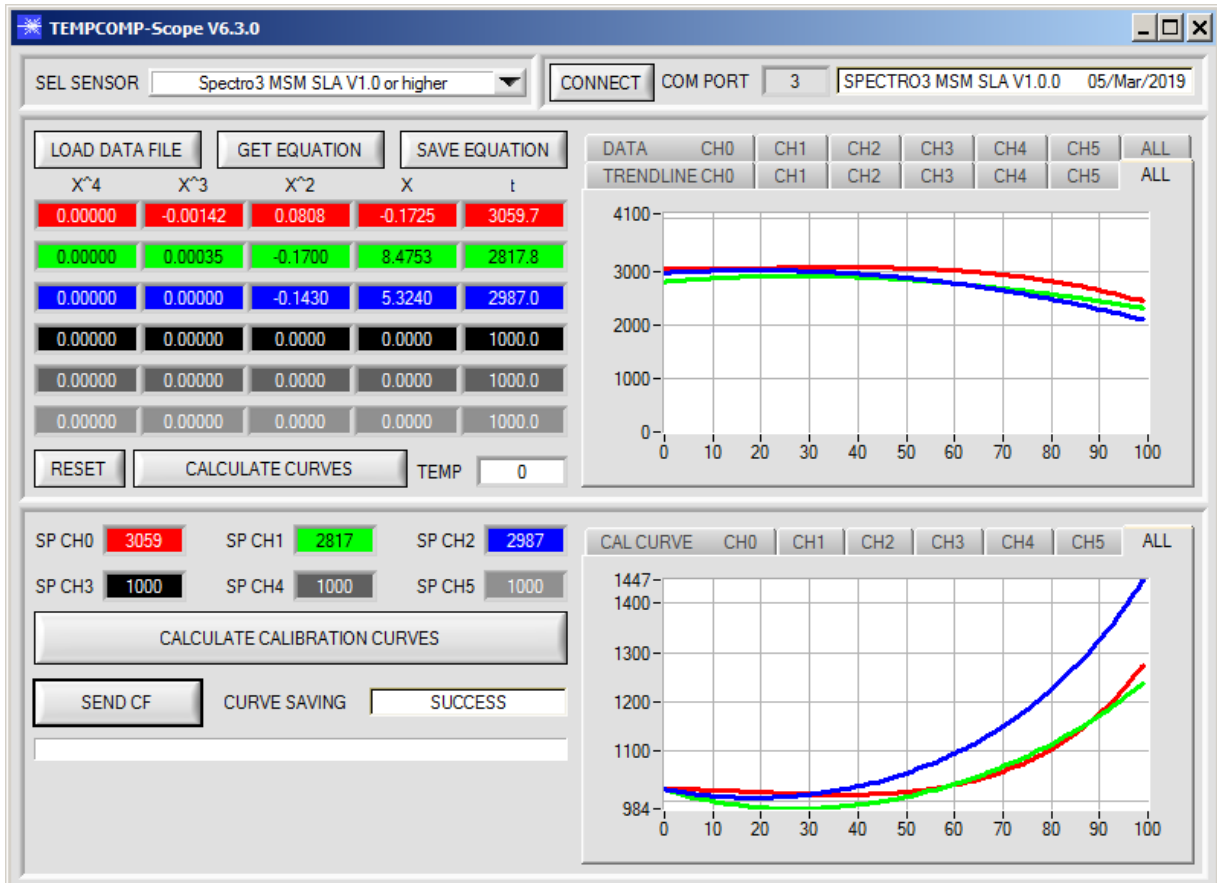
To perform temperature compensation please start the corresponding **TEMPCOMP-Scope software** that is included on the supplied CD. Please make sure that you have a functioning sensor connection. It may be necessary to select the connection with **CONNECT**. Set the correct sensor under **SELECT SENSOR**, if this is not done automatically.


Step 1: Load the temperature compensation file that you have received from your supplier with **GET EQUATION** or **LOAD DATA FILE**.

Step 2: Press **CALCULATE CURVES** to display the data in the graph.

Step 3: Select the sensor-internal operating temperature (not in °C) that the sensor has at an ambient temperature of 20°, if this has not already been done automatically. The value should be included in the file designation.

Step 4: Press **CALCULATE CALIBRATION CURVES** to calculate the mean straight line.

Step 5: Pressing the **SEND CF** button saves the mean straight lines in the **EEPROM** of the sensor.

Step 6: Temperature compensation is successfully finished if the **SUCCESS** status message is then displayed.

Comment! If you do not immediately have the temperature compensation file at hand, simply start the **TempComp-Scope software**. Establish a connection, if it is not already established, and simply press **SEND-CF**. The sensor then functions as before, but it is not temperature-compensated.

4. External triggering of the SPECTRO-3-MSM-SLE color sensor

External triggering is performed through pin no. 3 (grn) at the 8-pole M12 connector of the SPECTRO-3-MSM-SLE/PLC connection.
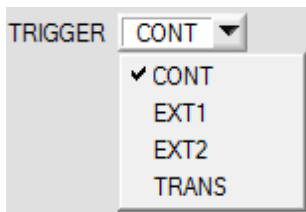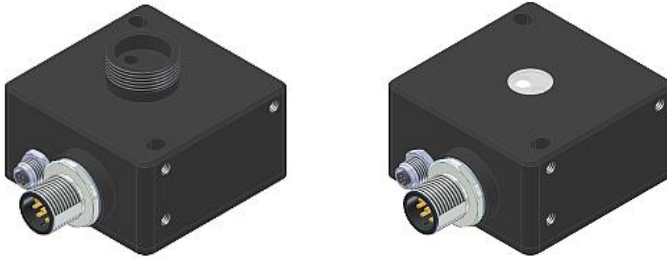
TRIGGER:
First the external trigger mode must be set at the color sensor. For this purpose option **EXT1, EXT2,** or **TRANS** must be selected in the **TRIGGER** selection field.
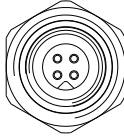
PLEASE NOTE:
The new setting is only activated at the sensor after a click on the SEND button!

Please note:	**The trigger input (IN0 PIN3 green of cable *cab-las8/SPS*) is HIGH-active, i.e. a trigger event is detected as long as IN0 = HIGH (+24V).**

IN0
(Pin3 grn)

HIGH (+24V)

LOW (0V, GND)

Triggering active !   No triggering !
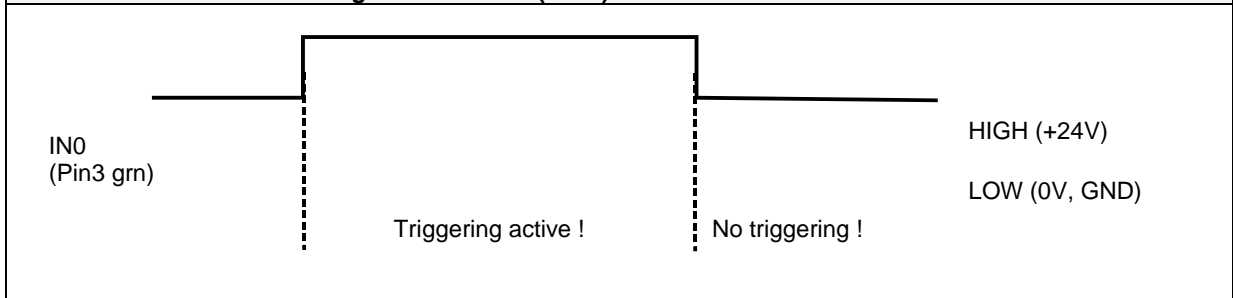
5. Connector assignment of the SPECTRO-3-MSM-SLE color sensor
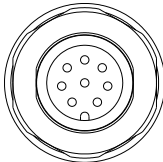
Connection of SPECTRO-3-MSM-SLE to PC:

4-pole fem. M5 connector (type Binder 707) *SPECTRO-3-MSM-SLE/PC-RS232*		

Pin-No.:		Assignment:
1		+24VDC (+Ub)
2		0V (GND)
3		Rx0
4		Tx0

Available connecting cables:
cab-las4/PC-…
cab-4/USB-…
cab-4/ETH-…

Connection of SPECTRO-3-MSM-SLE to PLC:

8-pole M12 connector (type Binder 763/713) *SPECTRO-3-MSM-SLE/PLC*		

Pin No.:	Color:	Assignment:
1	White	0V (GND)
2	Brown	+24VDC (± 10%)
3	Green	IN0
4	Yellow	OUT0 (Digital 0: typ. 0 … 1V, Digital 1: typ. +Ub – 10%)
5	Grey	OUT1 (Digital 0: typ. 0 … 1V, Digital 1: typ. +Ub – 10%)
6	Pink	OUT2 (Digital 0: typ. 0 … 1V, Digital 1: typ. +Ub – 10%)
7	Blue	OUT3 (Digital 0: typ. 0 … 1V, Digital 1: typ. +Ub – 10%)
8	Red	OUT4 (Digital 0: typ. 0 … 1V, Digital 1: typ. +Ub – 10%)

Connecting cable:
cab-M12/8-…-shd (shielded)

6.   RS232 communication protocol

The sensors of the SPECTRO-3-MSM-SLE series operate with the following **parameters** that are sent to the sensor or read from the sensor in the stated sequence.

Info! 1 **bytes** = 8bit   1 **word** = 2 **byte**   1 **long** = 2 **word** = 4 **byte**

Parameter		Type	Meaning
Para1:	POWER	word	Transmitter intensity (0 ... 1000) Attention intensity in thousandth!
Para2:	PMODE	word	Power Mode: SINGLE, DOUBLE coded to (0,1)
Para3:	GAIN	word	Amplification of the integrated receiver AMP1, AMP2, AMP3, AMP4, AMP5, AMP6, AMP7, AMP8 coded to (1, 2, 3, 4, 5, 6, 7, 8)
Para4:	INTEGRAL1	word	Raw signal integration (1…250)
Para5:	INTEGRAL2	word	Mean signal integration (1…250)
Para6:	AVERAGE	word	Signal averaging 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096, 8192, 16384 or 32768
Para7:	LED MODE	word	Control for the internal light source DC, AC coded to (0,1)
Para8:	C SPACE	word	Color Space: xyY, L*a*b*, L*u*v*, L*C*h*, L*u'v' coded to (0,1,2,3,4)
Para9:	CALIB	word	Calibration Mode: OFF, FCAL, UCAL, FCAL WB, UCAL WB, XYZ OFFSET, XYZ OFFSET IN0 coded to (0,1,2,3,4,5,6)
Para10:	DIGITAL OUTMODE	word	Function of the digital output: OFF, DIRECT HI, DIRECT LO, BINARY HI, BINARY LO coded to (0,1,2,3,4)
Para11:	MAXCOL-No.	word	Number of the colours (1,2,3,…,64)
Para12:	INTLIM	word	Intensity limit (0 ... 4095)
Para13:	EVALUATION MODE	word	Evaluation mode: FIRST HIT, BEST HIT coded to (0,1)
Para14:	SHAPE MODE	word	Shape mode: Block, Cylinder, Sphere codet to (0,1,2)
Para15:	EXTEACH	word	External teach mode: OFF, ON coded to (0,1)
Para16:	TRIGGER	word	Trigger mode: CONT, EXT1, EXT2, TRANS coded to (0, 1, 2, 3)
Para17:	COLOR GROUPS	word	Color groups enable: OFF, ON coded to (0,1)
Para18:	HOLD for C-No: 255	word	Hold time for failure condition (C-No: 255) coded to (0…100) [ms]
Para19:	POWER DP1	word	Transmitter intensity of Double Parameter Set 1 (0 ... 1000) Attention intensity in thousandth!
Para20:	GAIN DP1	word	Amplification of the integrated receiver of Double Parameter Set 1 AMP1, AMP2, AMP3, AMP4, AMP5, AMP6, AMP7, AMP8 coded to (1, 2, 3, 4, 5, 6, 7, 8)
Para21:	INTEGRAL DP1	word	Signal integration of Double Parameter Set1 (1…250)
Para22:	POWER DP2	word	Transmitter intensity of Double Parameter Set 2 (0 ... 1000) Attention intensity in thousandth!
Para23:	GAIN DP2	word	Amplification of the integrated receiver of Double Parameter Set 2 AMP1, AMP2, AMP3, AMP4, AMP5, AMP6, AMP7, AMP8 coded to (1, 2, 3, 4, 5, 6, 7, 8)
Para24:	INTEGRAL DP2	word	Signal integration of Double Parameter Set2 (1…250)
Para25:	COR VAL X	word	Correction Value for X multiplied by 128
Para26:	COR VAL Y	word	Correction Value for Y multiplied by 128
Para27:	COR VAL Z	word	Correction Value for Z multiplied by 128
Para28:	COR VAL X 3'rd root	word	3'rd root of COR VAL X = (COR VAL X * 128)^(1/3) * 1024
Para29:	COR VAL Y 3'rd root	word	3'rd root of COR VAL Y = (COR VAL Y * 128)^(1/3) * 1024
Para30:	COR VAL Z 3'rd root	word	3'rd root of COR VAL Z = (COR VAL Z * 128)^(1/3) * 1024

One row in the Teach Table determines a **TEACH VECTOR**.
The sensors of the SPECTRO-3-MSM-SLE series operate with 48 **TEACH VECTORS** that are sent to the sensor or read from the sensor in four blocks in the stated sequence.
Info! 1 **bytes** = 8bit   1 **word** = 2 **byte**   1 **long** = 2 **word** = 4 **byte**

| TEACH VECTOR		Type	Meaning
TeachVal1:	Teach Table Row 0 Column 0	long	x, a*, u*, u' of row 0. Must be multiplied with 65536.
TeachVal2:	Teach Table Row 0 Column 1	long	y, b*, v*, v' of row 0. Must be multiplied with 65536.
TeachVal3:	Teach Table Row 0 Column 2	long	Y, L*, L*, L*of row 0. Must be multiplied with 65536.
TeachVal4:	Teach Table Row 0 Column 3	long	Shape mode Block: xTol or a*Tol or u*Tol or u' Tol of row 0. Shape mode Cylinder: xyTol or a*b*Tol or u*v*Tol or u'v' Tol of row 0. Shape mode Sphere: CTO or deltaE of row 0. Must be multiplied with 65536. Must be sent as a dummy if not needed.
TeachVal5:	Teach Table Row 0 Column 4	long	Shape mode Block: yTol or b*Tol or v*Tol or v' Tol of row 0. Shape mode Cylinder: YTol or L*Tol of row 0. Shape mode Sphere: no use. Must be multiplied with 65536. Must be sent as a dummy if not needed.
TeachVal6:	Teach Table Row 0 Column 5	long	Shape mode Block: YTol or L*Tol of row 0. Shape mode Cylinder: no use. Shape mode Sphere: no use. Must be multiplied with 65536. Must be sent as a dummy if not needed.
TeachVal7:	Group Table Row 0	word	Group to which row 0 belongs if enabled
TeachVal8:	Hold Table Row 0	word	Hold time for row 0
TeachVal9:	Teach Table Row 1 Column 0	long	x, a*, u*, u' of row 1. Must be multiplied with 65536.
TeachVal10:	Teach Table Row 1 Column 1	long	y, b*, v*, v' of row 1. Must be multiplied with 65536.
TeachVal11:	Teach Table Row 1 Column 2	long	Y, L*, L*, L*of row 1. Must be multiplied with 65536.
TeachVal12:	Teach Table Row 1 Column 3	long	Shape mode Block: xTol or a*Tol or u*Tol or u' Tol of row 1. Shape mode Cylinder: xyTol or a*b*Tol or u*v*Tol or u'v' Tol of row 1. Shape mode Sphere: CTO or deltaE of row 1. Must be multiplied with 65536. Must be sent as a dummy if not needed.
TeachVal13:	Teach Table Row 1 Column 4	long	Shape mode Block: yTol or b*Tol or v*Tol or v' Tol of row 1. Shape mode Cylinder: YTol or L*Tol of row 1. Shape mode Sphere: no use. Must be multiplied with 65536. Must be sent as a dummy if not needed.
TeachVal14:	Teach Table Row 1 Column 5	long	Shape mode Block: YTol or L*Tol of row 1. Shape mode Cylinder: no use. Shape mode Sphere: no use. Must be multiplied with 65536. Must be sent as a dummy if not needed.
TeachVal15	Group Table Row 1	word	Group to which row 1 belongs if enabled
TeachVal16	Hold Table Row 1	word	Hold time for row 1
…	…	…	…
TeachVal384	Hold Table Row 47	word	Hold time for row 47

Upon request, the data acquired and processed by the sensor are sent by the sensor in the following sequence.
Info! 1 **bytes** = 8bit   1 **word** = 2 **byte**   1 **long** = 2 **word** = 4 **byte**

DATA VALUE		Type	Meaning
DatVal1:	CSX	long	Color Space X chromaticity coordinate (x, a*, u*, C* or u'). Must be divided by 65536.
DatVal2:	CSY	long	Color Space Y chromaticity coordinate (y, b*, v*, h* or v'). Must be divided by 65536.
DatVal3:	CSI	long	Color Space Lightness variable (Y or L*). Must be divided by 65536.
DatVal4:	delta E	long	Color difference to a color hit. Must be divided by 65536.
DatVal5:	X	word	Calibrated and temperature compensated tristimulus value of channel X
DatVal6:	Y	word	Calibrated and temperature compensated tristimulus value of channel Y
DatVal7:	Z	word	Calibrated and temperature compensated tristimulus value of channel Z
DatVal8:	RAW X	word	None Calibrated tristimulus value of channel X
DatVal9:	RAW Y	word	None Calibrated tristimulus value of channel Y
DatVal10:	RAW Z	word	None Calibrated tristimulus value of channel Z
DatVal11:	TEMP	word	Temperature in the sensor (**not** in °C or °F)
DatVal12:	C-No:	word	Detected color
DatVal13:	GRP	word	Detected group
DatVal14:	DIG IN	word	DIG IN is 1 when input IN0 is HI
DatVal15:	DP SET	word	Double Parameter Set
DatVal16:	SAT	word	Saturation (SAT=0: no Saturation, SAT>0: Saturation of one or more channels)
DatVal17:	DP RAW DATA X	word	Double Parameter Set 2 Raw Data of X
DatVal18:	DP RAW DATA Y	word	Double Parameter Set 2 Raw Data of Y
DatVal19:	DP RAW DATA Z	word	Double Parameter Set 2 Raw Data of Z

Digital serial communication is used for the exchange of data between the software running on the PC and the sensor.

For this purpose the control unit features an EIA-232 compatible interface that operates with the (fixed) parameters "**8 data bits, 1 stop bit, no parity bit, no handshake**".

Five values are available for the baudrate: 9600baud, 19200baud, 38400baud, 57600baud and 115200baud. As an option the PC software also can communicate through TCP/IP or USB. In these cases transparent interface converters must be used that allow a connection to the RS232 interface.
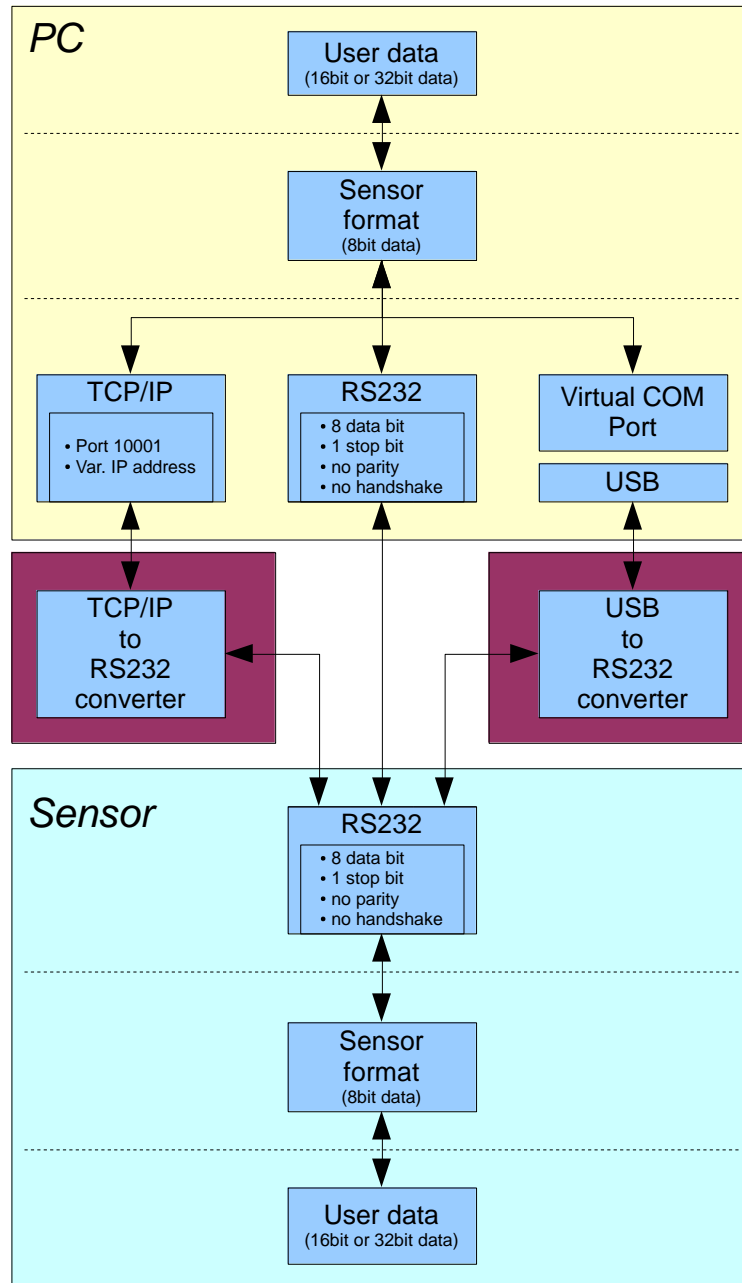
A proprietary protocol format that organises and bundles the desired data is used for all physical connection variants between PC software and control unit. Depending on their type and function the actual data are 16- or 32-bit variables and represent integer or floating-point values. The protocol format consists of 8-bit wide unsigned words ("bytes"). The actual data therefore sometimes must be distributed to several bytes.

The control unit always behaves passively (except if another behaviour has been specifically activated). Data exchange therefore always is initiated by the PC software. The PC sends a data package ("frame") corresponding to the protocol format, either with or without appended data, to which the control unit responds with a frame that matches the request.

The protocol format consists of two components:
A "header" and an optional appendant ("data").
The header always has the same structure.
The first byte is a synchronisation byte and always is 85_{dez} (55_{hex}).
The second byte is the so-called order byte. This byte determines the action that should be performed (send data, save data, etc.).
A 16-bit value (argument) follows as the third and fourth byte. Depending on the order, the argument is assigned a corresponding value.
The fifth and sixth byte again form a 16-bit value. This value states the number of appended data bytes. Without appended data both these bytes are 0_{dez} or 00_{hex}, the maximum number of bytes is 512.
The seventh byte contains the CRC8 checksum of all data bytes (data byte 0 up to and incl. data byte n).
The eight byte is the CRC8 checksum for the header and is formed from bytes 0 up to and incl. 6.
The header always has a total length of 8 bytes. The complete frame may contain between 8 and 520 bytes.

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header	Byte8 Data	Byte9 Data	…	Byte n+6 Data	Byte n+7 Data
0x55	<order>	<ARG> (lo byte)	<ARG> (hi byte)	<LEN> (lo byte)	<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)	Data1 (lo byte)	Data1 (hi byte)	…	Data n/2 (lo byte)	Data n/2 (hi byte)

The following **orders** can be sent to the sensor.

Number	ORDER (header byte no. 2)	Example
0	Sensor answers with order=0 if a communication error occurs. ARG=1: Invalid order number was sent to the sensor ARG=2: General communication error (wrong baudrate, overflow, …)	
1	Write parameter to the RAM of the sensor	order=1
2	Read parameter from the RAM of the sensor	order=2
3	Load parameter and actual Baudrate from RAM to EEPROM of the sensor	order=3
4	Load parameter from EEPROM to RAM of the sensor	order=4
5	Read CONNECTION OK and serial number from sensor	order=5
6	Free	
7	Read Firmware String and firmware number from sensor	order=7
8	Read data values from sensor	order=8
108	Read 3 data values from sensor	order=108
30	Start and Stop triggered sending of data frames	order=30
105	Get cycle time from sensor	order=105
190	Write new baud rate to the sensor	order=190

CRC8 checksum

The so-called "Cyclic Redundancy Check" or CRC is used to verify data integrity. This algorithm makes it possible to detect individual bit errors, missing bytes, and faulty frames. For this purpose a value - the so-called checksum - is calculated over the data (bytes) to be checked and is transmitted together with the data package. Calculation is performed according to an exactly specified method based on a generator polynomial. The length of the checksum is 8 bit (= 1 byte). The generator polynomial is:

$$X^8 + X^5 + X^4 + X^0$$

To verify the data after they have been received, CRC calculation is performed once again. If the sent and the newly calculated CRC values are identical, the data are without error.
The following pseudo code can be used for checksum calculation:

```
calcCRC8 (data[ ], table[ ])
Input:  data[ ], n data of unsigned 8bit
        table[ ], 256 table entries of unsigned 8bit
Output: crc8, unsigned 8bit

crc8  := AA_hex
for I := 1 to n do
        idx := crc8 EXOR data[ i ]
        crc8 := table[ idx ]
endfor
return crc8
```

table[ ]

0	94	188	226	97	63	221	131	194	156	126	32	163	253	31	65
157	195	33	127	252	162	64	30	95	1	227	189	62	96	130	220
35	125	159	193	66	28	254	160	225	191	93	3	128	222	60	98
190	224	2	92	223	129	99	61	124	34	192	158	29	67	161	255
70	24	250	164	39	121	155	197	132	218	56	102	229	187	89	7
219	133	103	57	186	228	6	88	25	71	165	251	120	38	196	154
101	59	217	135	4	90	184	230	167	249	27	69	198	152	122	36
248	166	68	26	153	199	37	123	58	100	134	216	91	5	231	185
140	210	48	110	237	179	81	15	78	16	242	172	47	113	147	205
17	79	173	243	112	46	204	146	211	141	111	49	178	236	14	80
175	241	19	77	206	144	114	44	109	51	209	143	12	82	176	238
50	108	142	208	83	13	239	177	240	174	76	18	145	207	45	115
202	148	118	40	171	245	23	73	8	86	180	234	105	55	213	139
87	9	235	181	54	104	138	212	149	203	41	119	244	170	72	22
233	183	85	11	136	214	52	106	43	117	151	201	74	20	246	168
116	42	200	150	21	75	169	247	182	232	10	84	215	137	107	53

Example order=1: Write parameter to the RAM of the sensor.
<ARG> determines whether you want to save parameter or teach vectors:
ARG = 0 --> Parameter
ARG = 1 --> Teach Vector Row   0…11
ARG = 2 --> Teach Vector Row 12…23
ARG = 3 --> Teach Vector Row 24…35
ARG = 4 --> Teach Vector Row 36…47

Example is with 5 parameter (Para1=500, Para2=0; Para3=3200, Para4=3300, Para5=1)
Have a look at the **TABLE PARAMETER** to check out how much parameter you have to send.

DATA FRAME PC → Sensor

Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header	Byte8 Header
0x55	<order>	<ARG> (lo byte)	<ARG> (hi byte)	<LEN> (lo byte)	<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	1	0	0	10	0	130	107
		ARG=0		LEN=10			

Byte9 Data	Byte10 Data	Byte11 Data	Byte12 Data	Byte13 Data	Byte14 Data	Byte15 Data	Byte16 Data	Byte17 Data	Byte18 Data
Para1 (lo byte)	Para1 (hi byte)	Para2 (lo byte)	Para2 (hi byte)	Para3 (lo byte)	Para3 (hi byte)	Para4 (lo byte)	Para4 (hi byte)	Para5 (lo byte)	Para5 (hi byte)
244	1	0	0	128	12	228	12	1	0
Para1=500		Para2=0		Para3=3200		Para4=3300		Para5=1	

DATA FRAME Sensor → PC

Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header	Byte8 Header
0x55	<order>	<ARG> (lo byte)	<ARG> (hi byte)	<LEN> (lo byte)	<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	1	0	0	0	0	170	224
		ARG=0		LEN=0			

If you receive an argument greater 0, ARG parameter where out of range and have been set to a default value.

Example order=2: Read parameter from the RAM of the sensor.

<ARG> determines whether you want to read parameter or teach vectors:

ARG = 0 --> Parameter

ARG = 1 --> Teach Vector Row 0…11

ARG = 2 --> Teach Vector Row 12…23

ARG = 3 --> Teach Vector Row 24…35

ARG = 4 --> Teach Vector Row 36…47

Example is with 5 parameter (Para1=500, Para2=0; Para3=3200, Para4=3300, Para5=1)

Have a look at the **TABLE PARAMETER** to check out how much parameter you will receive.

DATA FRAME PC → Sensor

Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header	Byte8 Header
0x55	<order>	<ARG> (lo byte)	<ARG> (hi byte)	<LEN> (lo byte)	<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	2	0	0	0	0	170	185
		ARG=0		LEN=0			

DATA FRAME Sensor → PC

Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header	Byte8 Header
0x55	<order>	<ARG> (lo byte)	<ARG> (hi byte)	<LEN> (lo byte)	<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	2	0	0	10	0	130	50
		ARG=0		LEN=10			

Byte9 Data	Byte10 Data	Byte11 Data	Byte12 Data	Byte13 Data	Byte14 Data	Byte15 Data	Byte16 Data	Byte17 Data	Byte18 Data
Para1 (lo byte)	Para1 (hi byte)	Para2 (lo byte)	Para2 (hi byte)	Para3 (lo byte)	Para3 (hi byte)	Para4 (lo byte)	Para4 (hi byte)	Para5 (lo byte)	Para5 (hi byte)
244	1	0	0	128	12	228	12	1	0
Para1=500		Para2=0		Para3=3200		Para4=3300		Para5=1	

Example order=3: Load parameter and actual Baudrate from RAM to EEPROM of the sensor.

DATA FRAME PC → Sensor

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header
0x55	\<order\>	\<ARG\> (lo byte)	\<ARG\> (hi byte)	\<LEN\> (lo byte)	\<LEN\> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	3	0	0	0	0	170	142
		ARG=0		LEN=0			

DATA FRAME Sensor → PC

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header
0x55	\<order\>	\<ARG\> (lo byte)	\<ARG\> (hi byte)	\<LEN\> (lo byte)	\<LEN\> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	3	0	0	0	0	170	142
		ARG=0		LEN=0			

Example order=4: Load parameter from EEPROM to RAM of the sensor.

DATA FRAME PC → Sensor

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header
0x55	\<order\>	\<ARG\> (lo byte)	\<ARG\> (hi byte)	\<LEN\> (lo byte)	\<LEN\> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	4	0	0	0	0	170	11
		ARG=0		LEN=0			

DATA FRAME Sensor → PC

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header
0x55	\<order\>	\<ARG\> (lo byte)	\<ARG\> (hi byte)	\<LEN\> (lo byte)	\<LEN\> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	4	0	0	0	0	170	11
		ARG=0		LEN=0			

Example order=5: Read CONNECTION OK from sensor.

DATA FRAME PC → Sensor

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header
0x55	\<order\>	\<ARG\> (lo byte)	\<ARG\> (hi byte)	\<LEN\> (lo byte)	\<LEN\> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	5	0	0	0	0	170	60
		ARG=0		LEN=0			

DATA FRAME Sensor → PC
ARG determines the serial number of the sensor

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header
0x55	\<order\>	\<ARG\> (lo byte)	\<ARG\> (hi byte)	\<LEN\> (lo byte)	\<LEN\> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	5	170	0	0	0	170	178
		ARG=170		LEN=0			

Example order=7: Read Firmware String from sensor

DATA FRAME PC → Sensor

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header
0x55	<order>	<ARG> (lo byte)	<ARG> (hi byte)	<LEN> (lo byte)	<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	7	0	0	0	0	170	82
		ARG=0		LEN=0			

DATA FRAME Sensor → PC
ARG determines the firmware number of the sensor

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header	Byte8 Data	Byte9 Data	Byte10 Data	Byte11 Data
0x55	<order>	<ARG> (lo byte)	<ARG> (hi byte)	<LEN> (lo byte)	<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)	ASCII	ASCII	ASCII	ASCII
85 (dec)	7	0	0	72	0	183	38	F	I	R	M
		ARG=0		LEN=72							

Byte12 Data	Byte13 Data	Byte14 Data	Byte15 Data	Byte16 Data	Byte17 Data	Byte18 Data	Byte19 Data	Byte20 Data	Byte21 Data	Byte22 Data	Byte23 Data
ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII
W	A	R	E		S	T	R	I	N	G	

Byte24 Data	Byte25 Data	Byte26 Data	Byte27 Data	Byte28 Data	Byte29 Data	Byte30 Data	Byte31 Data	Byte32 Data	Byte33 Data	Byte34 Data	Byte35 Data
ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII
											R

Byte36 Data	Byte37 Data	Byte38 Data	Byte39 Data	Byte40 Data	Byte41 Data	Byte42 Data	Byte43 Data	Byte44 Data	Byte45 Data	Byte46 Data	Byte47 Data
ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII
T	:	K	W	x	x	/	x	x			

Byte48 Data	Byte49 Data	Byte50 Data	Byte51 Data	Byte52 Data	Byte53 Data	Byte54 Data	Byte55 Data	Byte56 Data	Byte57 Data	Byte58 Data	Byte59 Data
ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII

Byte60 Data	Byte61 Data	Byte62 Data	Byte63 Data	Byte64 Data	Byte65 Data	Byte66 Data	Byte67 Data	Byte68 Data	Byte69 Data	Byte70 Data	Byte71 Data
ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII

Byte72 Data	Byte73 Data	Byte74 Data	Byte75 Data	Byte76 Data	Byte77 Data	Byte78 Data	Byte79 Data	Byte80 Data	Byte81 Data
ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII

Example order=8: Read data values from sensor.

DATA FRAME PC → Sensor

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header
0x55	<order>	<ARG> (lo byte)	<ARG> (hi byte)	<LEN> (lo byte)	<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	8	0	0	0	0	170	118
		ARG=0		LEN=0			

DATA FRAME Sensor → PC

8 Header Bytes and all **DATA VALUES** (See Table DATA VALUE)

Example order=108: Read 3 data values from sensor.

DATA FRAME PC → Sensor

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header
0x55	\<order>	\<ARG> (lo byte)	\<ARG> (hi byte)	\<LEN> (lo byte)	\<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	108	0	0	0	0	170	105
		ARG=0		LEN=0			

DATA FRAME Sensor → PC

8 Header Bytes and 3 **DATA VALUES** (DatVal1, DatVal2, DatVal3. See Table DATA VALUE)

Example order=30: Start and Stop triggered sending of data frames

Start triggered sending of data frames
DATA FRAME PC → Sensor
\<ARG> determines which data should be send when Trigger goes HI
ARG = 1 --> The HEADER and all DATA VALUES will be send
ARG = 2 --> The HEADER and only the color space coordinates CSX, CSY, CSY will be send

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header
0x55	\<order>	\<ARG> (lo byte)	\<ARG> (hi byte)	\<LEN> (lo byte)	\<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	30	1	0	0	0	170	82
		ARG=1		LEN=0			

DATA FRAME Sensor → PC

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header
0x55	\<order>	\<ARG> (lo byte)	\<ARG> (hi byte)	\<LEN> (lo byte)	\<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	30	1	0	0	0	170	82
		ARG=1		LEN=0			

Stop triggered sending of data frames
DATA FRAME PC → Sensor
ARG = 0 --> stops triggered sending.

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header
0x55	\<order>	\<ARG> (lo byte)	\<ARG> (hi byte)	\<LEN> (lo byte)	\<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	30	0	0	0	0	170	159
		ARG=0		LEN=0			

DATA FRAME Sensor → PC

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header
0x55	\<order>	\<ARG> (lo byte)	\<ARG> (hi byte)	\<LEN> (lo byte)	\<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	30	0	0	0	0	170	159
		ARG=0		LEN=0			

Example order=105: Get cycle time from sensor

DATA FRAME PC → Sensor

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header
0x55	<order>	<ARG> (lo byte)	<ARG> (hi byte)	<LEN> (lo byte)	<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	105	0	0	0	0	170	130
		ARG=0		LEN=0			

DATA FRAME Sensor → PC

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header	Byte8 Data	Byte9 Data	Byte10 Data	Byte11 Data
0x55	<order>	<ARG> (lo byte)	<ARG> (hi byte)	<LEN> (lo byte)	<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)	lo word lo byte	lo word hi byte	hi word lo byte	hi word hi byte
85 (dec)	105	0	0	8	0	206	163	40	28	2	0
		ARG=0		LEN=8				CYCLE COUNT = 138280			

Byte12 Data	Byte13 Data	Byte14 Data	Byte15 Data
lo word lo byte	lo word hi byte	hi word lo byte	hi word hi byte
144	1	0	0
COUNTER TIME = 400			

Cycle Time [Hz] = CYCLE COUNT / (COUNTER TIME * 0,01)

Cycle Time [ms] = (COUNTER TIME * 0,01) / CYCLE COUNT

Example order=190: Write new baud rate to the sensor.

DATA FRAME PC → Sensor

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header
0x55	<order>	<ARG> (lo byte)	<ARG> (hi byte)	<LEN> (lo byte)	<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	190	1	0	0	0	170	14
		ARG=1		LEN=0			

New baud rate is determined by argument.
ARG=0: baud rate = 9600
ARG=1: baud rate = 19200
ARG=2: baud rate = 38400
ARG=3: baud rate = 57600
ARG=4: baud rate = 115200
ARG=5: baud rate = 230400
ARG=6: baud rate = 460800

DATA FRAME Sensor → PC

Byte0 Header	Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header
0x55	<order>	<ARG> (lo byte)	<ARG> (hi byte)	<LEN> (lo byte)	<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)
85 (dec)	190	0	0	0	0	170	195
		ARG=0		LEN=0			

A. Firmware update via software Firmware Loader

A.1 Manual software Firmware Loader V1.1

This manual describes the installation of the Firmware Loader PC software and as a support for starting up the Firmware Loader software explains the individual functional elements of the graphic Windows® user interface.

The software allows the user to perform an automatic firmware update. The update will be carried out through the RS232 interface.

An initialisation file (xxx.ini) and a firmware file (xxx.elf.S) are required for performing a firmware update. These files can be obtained from your supplier. In some cases an additional firmware file for the program memory (xxx.elf.p.S) is also needed, and this file will be automatically provided together with the other two files.

Important! For a firmware update the two or three files must always be located in the same folder.

A plausibility check is performed after the initialisation file has been loaded with the Firmware Loader. If the initialisation file was changed or damaged, it will not be possible to perform a firmware update.

When the plausibility check is successfully completed, the instructions contained in the initialisation file will be carried out step by step.

The complete memory contents of the micro-controller in the sensor will be deleted in a firmware update. This means that both the program in the program memory and the data in the data memory will be lost.

The new firmware automatically writes the correct data to the program memory again.

However, the parameter settings, temperature curves, linearization curves, etc. that are stored in the data memory (EEPROM) will be deleted.

With the Firmware Loader V1.1 the data will be saved in the EEPROM, and can be written back again after successful firmware update.

For this purpose, the software creates an EEPROM backup file.

A.2   Installation of the software Firmware Loader V1.1

Hardware requirements for successful installation of the Firmware Loader software:

- Microsoft® Windows® 7, 8, 10
- IBM PC AT or compatible
- VGA graphics
- Microsoft® compatible mouse
- Serial RS232 interface at the PC or USB slot or RJ45 connector
- Cable *cab-las4/PC* (*cab-las5/PC)* for the RS232 interface or *cab-4/USB* (*cab-5/USB)* USB converter or *cab-4/ETH (cab-5/ETH)* Ethernet converter

Please install the software as described below:

1.	The software can be installed directly from the installation DVD. To install the software, start the SETUP program in the SOFTWARE folder of the DVD.
2.	The installation program displays a dialog and suggests to install the software in the C:\"FILENAME" directory on the hard disk. You may accept this suggestion with **OK** or [**ENTER**], or you may change the path as desired. Installation is then performed automatically.
3.	During the installation process a new program group for the software is created in the Windows Program Manager. In the program group an icon for starting the software is created automatically. When installation is successfully completed the installation program displays "Setup OK".
4.	After successful installation the software can be started with a left mouse button double-click on the icon.

Windows® is a trademark of the Microsoft Corp.
VGA™ is a trademark of the International Business Machines Corp.

Please read this chapter before you start!

In this example a software update is performed from SPECTRO3 V4.0 to SPECTRO3 V4.1.

Step 1:

When the Firmware Loader software is started, this window opens on the Windows® user interface.

Immediately after starting, the software attempts to establish a connection to the connected sensor. If the sensor should not be connected at **COM PORT 1,** please select the corresponding **COM PORT.**
Please make sure that the correct **BAUDRATE** is selected.
Now try to establish a connection by clicking on **TRY TO CONNECT**. When the connection has been established, the sensor sends back information about the current firmware.
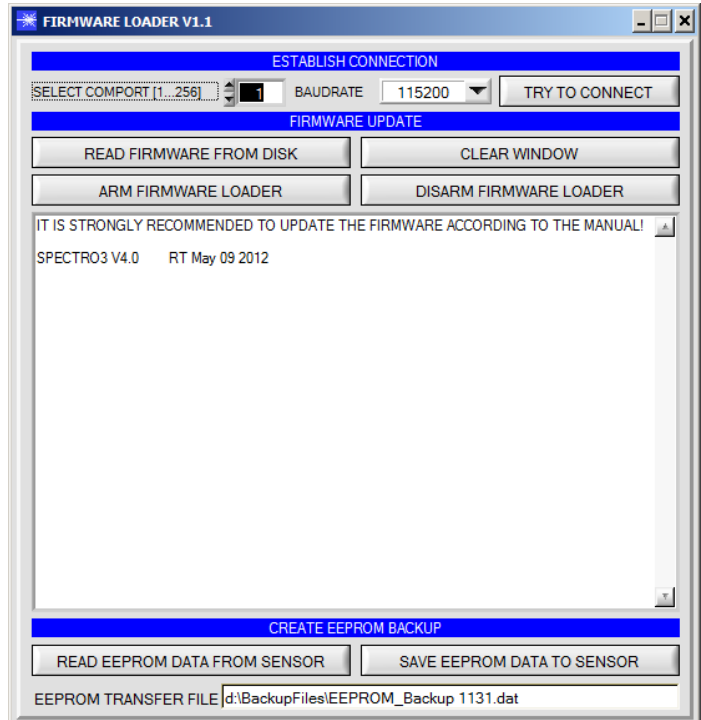

Step 2:
Press the **READ FIRMWARE FROM DISK** button and load the **xxx.ini** file.

The uploaded initialization file will be displayed in the status window.
As described above, a plausibility check of the initialisation file will be performed first.
If the file is OK, the following message will be displayed:
File read OK!
Press **ARM FIRMWARE LOADER** to start the firmware update.

Please read the comments that are shown in the display window. These comments allow you to make sure that you have loaded the correct initialisation file.
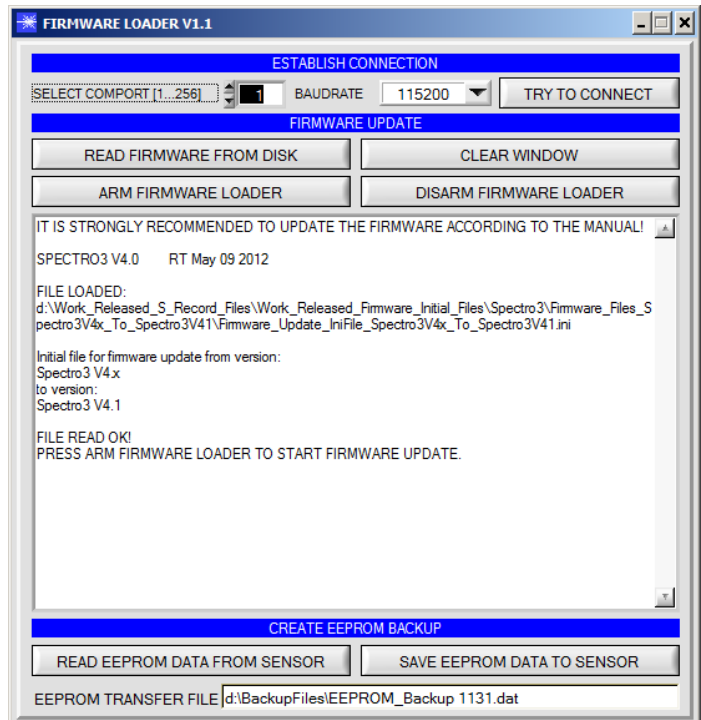
Step 3:

Now click on the **ARM FIRMWARE LOADER** button. The program now attempts to send a software command that interrupts the normal program run and jumps to the start address of the boot sector.

If this is successful, the sensor displays a prompt for loading the S-Record file to the sensor.

When you press the **ARM FIRMWARE LOADER** button the firmware update will be performed automatically.

In the course of the update process you will only be prompted to enter a name for the EEPROM backup file. If the firmware update should run perfectly until the EEPRM data are read out, but should then go wrong for any reason whatsoever, the EEPROM backup file can always be written back with **SAVE EEPROM DATA TO SENSOR**.

The file name for your **EEPROM backup file** should be chosen such that the names for several sensors cannot be mixed up. Using a file name that contains the sensor serial number might be advisable. Saving this file for future updates also might be a good idea.

After a successful update the sensor displays the status line of the new firmware.

The complete update process may take up to 1 minute.
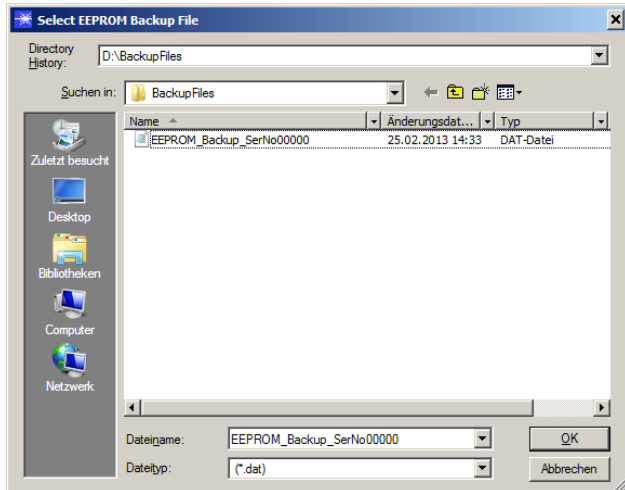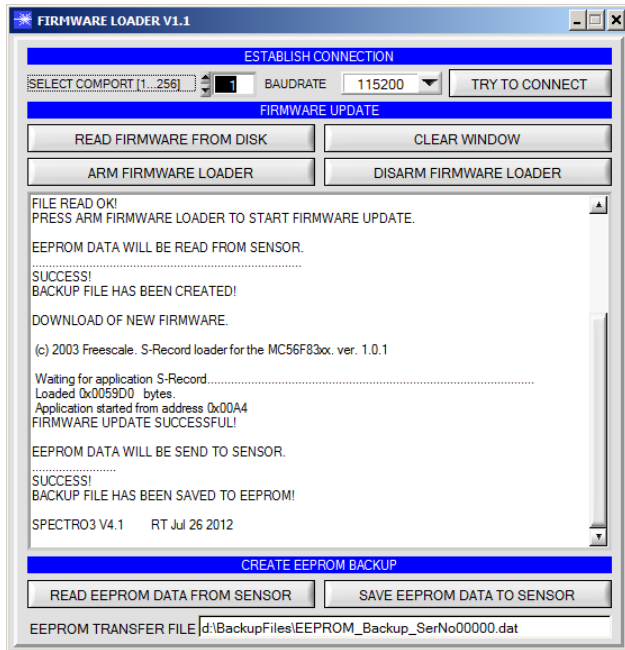
If, contrary to expectations, there should be any trouble with the update of the program memory, it will still be possible to perform an update, even though it may look like the sensor was "killed".

Please make sure that you have selected the correct **COM PORT** and the correct **BAUDRATE**.

You will not get any connection when you click on **TRY TO CONNECT**.

Load the corresponding **xxx.ini** file from the hard disk.

Then click on the **ARM FIRMWARE LOADER** button.

The program will try to send the software command for the update. This will not work, however, and you will get a **CONNECTION FAILURE** message.

However, the Firmware Loader software now is "armed" for 30 seconds.

If you perform a hardware reset within these 30 seconds, the firmware update will be performed.

After a successful update the sensor displays the status line of the new firmware.

The complete update process may take up to 1 minute.

INFO! In case that the sensor was "killed", the sensor will work with a BAUDRATE of 115200.

You may at any time create an EEPROM backup file for archiving it on your hard disk.

To do this, click on **READ EEPROM DATA FROM SENSOR.** You will be prompted to chose an initialization file in case that there has not yet been loaded any. Afterwards you will be asked to enter a file name. The selected name will be shown in the **EEPROM TRANSFER FILE** display.

The file name for your **EEPROM backup file** should be chosen such that the names for several sensors cannot be mixed up. Using a file name that contains the sensor serial number might be advisable.

The Firmware Loader then reads all the EEPROM data from the data memory and saves these data in the selected file.

Upon successful completion the following message will be displayed:

Success!

Backup File has been created!

If something should go wrong in a firmware update, any you have created the **backup file**, the saved **EEPROM backup file** can at any time be uploaded to the sensor again with **SAVE EEPROM DATA TO SENSOR.**
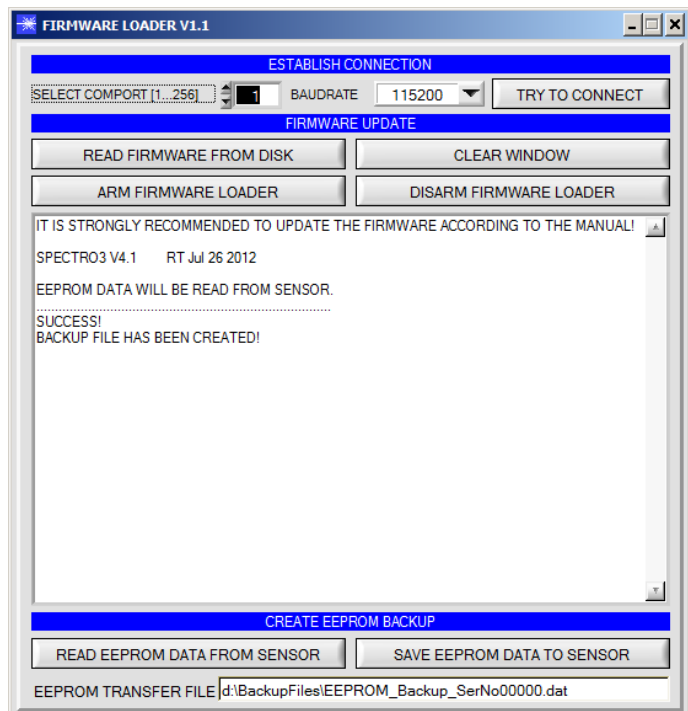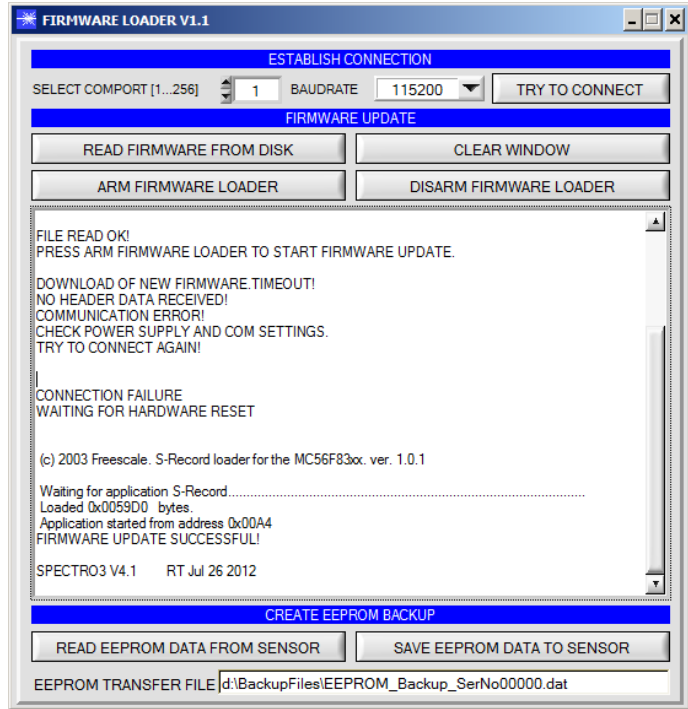
CLEAR WINDOW resets the display window.

If you should not get any response for a longer time, or if messages should be displayed in the status line, **DISARM FIRMWARE LOADER** can be used to cancel the firmware update process.

However, you should always wait for approx. 1 minute before you press this button.