

# Manual

## Software COAST-Scope V1.1

(PC software for Microsoft® Windows® 7, 8, 10)

### for sensors of the COAST series with internal temperature compensation

This manual describes the installation of the PC software for the COAST sensor. As a support for commissioning of the sensor this manual explains the functional elements of the Windows® user interface.

The sensors of the **COAST** series (**CO**lor **And** **ST**ructure) are hybrid sensors comprising a color sensor and a structure sensor.

The color sensor detects the color of the object to be inspected.

The structure sensor detects data concerning the surface structure of the object, and through an internal interface sends these data to the color sensor.

The color sensor then evaluates the color data together with the data it receives from the structure sensor. It also controls the digital inputs and outputs and communicates with the **COAST-Scope** PC software.

The available interfaces include a PLC interface, a RS232 interface for the color sensor, and a RS232 interface for the structure sensor. The structure sensor is factory-preconfigured in such a way that it only requires parameterisation in exceptional cases. For this purpose a special **COAST-STRUCT-Scope** software is available.

The COAST sensor 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 and allows a scan-frequency of up to 35 kHz. With the stepless adjustment of the two integrated LED light sources as well as the selectable gain of the receiver signal and an INTEGRAL function the sensor can be set to almost any surface.

The sensor detects the radiation that is diffusely reflected from the object. As light sources the COAST sensor uses two white-light LED ring lights with adjustable transmitter power. An integrated 3-fold receiver for the red, green, and blue component of the light that is reflected from the object is used as a receiver. The surface structure of the measurement object is mapped on a line sensor by means of an optical aperture system.

The sensors of the COAST series can be taught up to 31 surfaces, hereinafter referred to as **teach vectors** or **vectors**, that can be directly provided at the outputs, and up to 48 vectors that can be output by way of grouping. The vector that shows the smallest deviation from the current surface (minimum distance) is provided at the output.

The respective detected vector 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 vectors are to be detected. At the same time the detected vector is visualised by means of 5 LEDs at the housing of the COAST sensor.

With a TEACH button at the sensor housing the sensor can be taught up to 31 vectors (max. 64 vectors in group mode). For this purpose the corresponding evaluation mode must be set with the software. The TEACH button is connected in parallel to the input IN0 (green wire at cable cab-las8/SPS).

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

## 0 Contents

	Page
1. Installation of the COAST-Scope software .....	3
2. Operation of the COAST-Scope software.....	4
2.1 Tab CONNECT .....	5
2.2 Tab PARA1, button SEND, GET, GO, STOP (parameterization, data exchange) .....	7
2.3 Tab PARA2 (vector groups and HOLD time settings) .....	14
2.4 Tab TEACH (Teach Table).....	15
2.5 Tab GEN (vector specifications).....	17
2.6 Tab REC (data recording) .....	18
2.7 Tab Offset Calibration .....	20
2.8 Tab SCOPE .....	21
2.9 Graphic display elements .....	22
3. Operation of the TEMPCOMP-Scope software .....	24
4. Function of the LED display.....	25
5. Connector assignment of the COAST sensor.....	27
6. RS232 communication protocol.....	28
A. Firmware update via software Firmware Loader .....	39

**Shortcuts:**

SEND	F9
GET	F10
GO	F11
STOP	F12

## 1. Installation of the COAST-Scope software

Hardware requirements for successful installation of the COAST-Scope software:

- Microsoft® Windows® 7, 8, 10
- IBM PC AT or compatible
- VGA graphics
- Microsoft-compatible mouse
- CD-ROM drive
- Serial RS232 interface at the PC or USB slot
- Cable **cab-las4/PC** for the RS232 interface or **cab-4/USB** for USB slot

The COAST-Scope software can only be installed under Windows. Windows must therefore be started first if it is not yet running.

Please install the software as described below:

1. The software can be installed directly from the installation CD-ROM. To install the software, start the SETUP program in the SOFTWARE folder of the CD-ROM.
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.

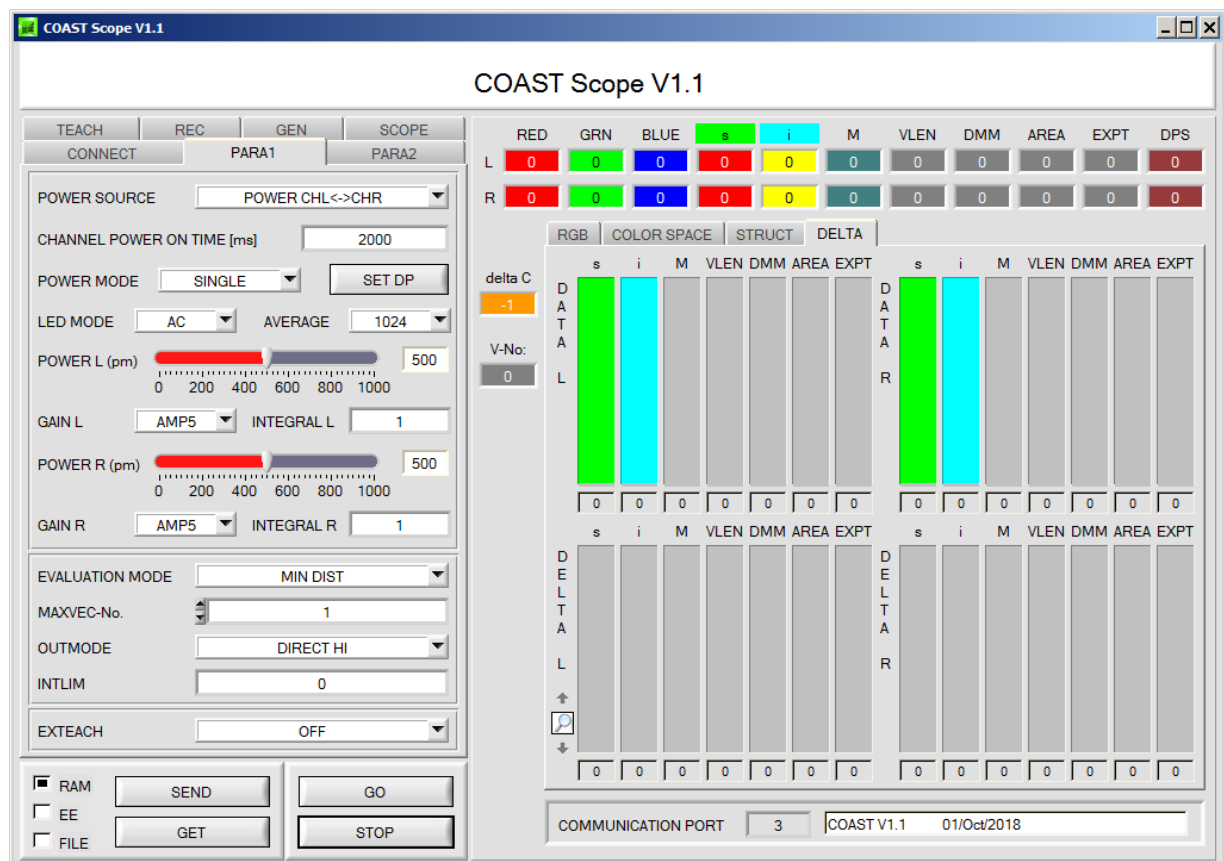
Windows® is a trademark of the Microsoft Corp.  
VGA™ is a trademark of the International Business Machines Corp.

## 2. Operation of the COAST-Scope software

Please read this chapter first before you start to adjust and parameterise the COAST sensor.

When the COAST-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.**

## 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 adaptor 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 adaptor (**cab-4/ETH**) is needed if the sensor should communicate through a local network. With this adaptor a connection to the sensor can be established using the **TCP/IP** protocol.

Parameterisation of the **cab-4/ETH** adaptor (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 adaptor, 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.

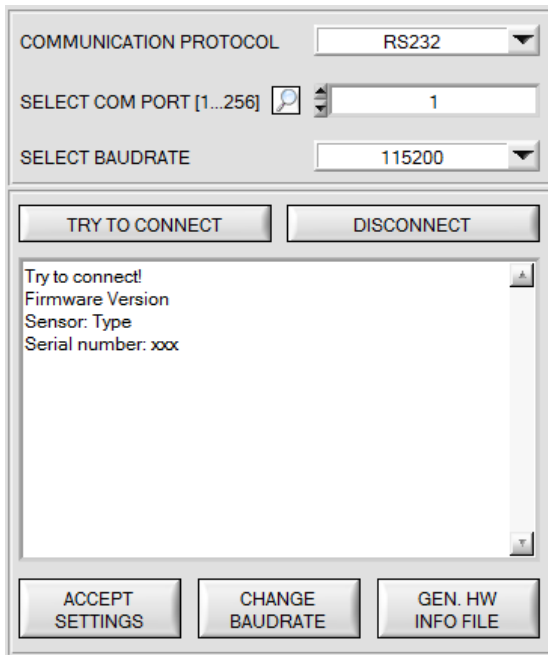
When you press the **TRY TO CONNECT** button, the software tries to establish a connection with the set parameters. The communication status is shown in the display field. If the sensor answers with its FIRMWARE ID, the set connection type can be accepted by pressing **ACCEPT SETTINGS**. You will then 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.

<p><b>Please note:</b></p>  <p><b>ATTENTION !</b></p>	<p><b>The stable function of the interface is a basic prerequisite for measured value transfer from the PC to the sensor.</b></p> <p><b>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.</b></p> <p><b>For maintaining maximum switching frequency at the sensor data communication with the PC must be stopped (press the STOP button).</b></p>
--	---



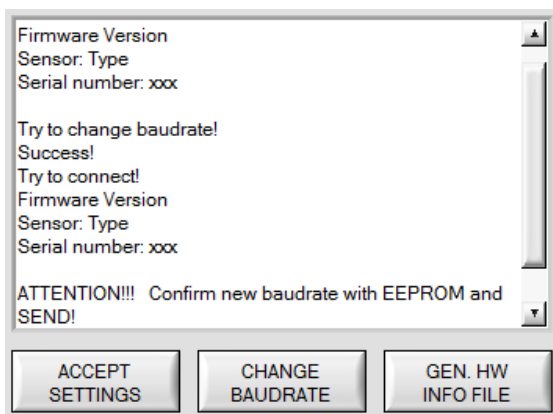
The baudrate for data transfer through the RS232 interface can be set by means of the **SELECT BAUDRATE** drop down menu and **CHANGE BAUDRATE** function field.

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.



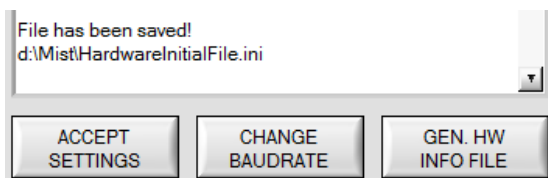
Now a new baudrate can be selected under **SELECT BAUDRATE**.

A click on **CHANGE BAUDRATE** sends the new baudrate information to the sensor.



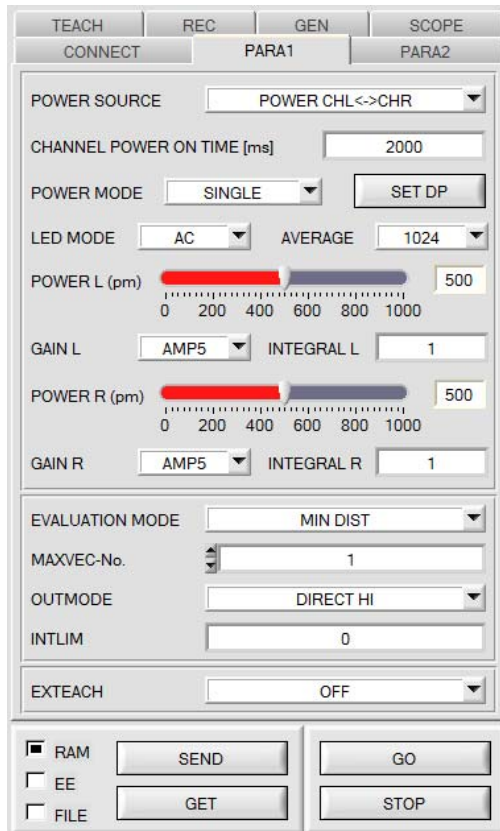
When the new baudrate information has been successfully sent, the sensor operates with the new baudrate. A window will pop up, prompting you to select **EEPROM** and then to press **SEND**. After a hardware reset the new baudrate only will be used when **EEPROM** and **SEND** have been pressed.

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



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.

## 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:

The **RAM** is a **volatile** memory in the sensor's micro-controller, i.e. when the power at the sensor is turned off, these parameters will be lost again.

**The sensor always operates with the parameters in its RAM.**

If the **RAM** option is selected, a click on **SEND** writes the current parameters to the sensor's **RAM** memory, and a click on **GET** reads the parameters from the sensor's **RAM** memory.

### EEPROM:

The **EEPROM** is a **non-volatile** memory in the sensor's micro-controller. When the power at the sensor is turned off the parameters in the **EEPROM** will not be lost. When power is turned on again, the parameters are loaded from the **EEPROM** to the **RAM** memory. Figuratively speaking the **EEPROM** thus is a level lower than the **RAM**. Data exchange between **PC** and **EEPROM** automatically is performed through the **RAM** memory, which means that parameters that are written to the **EEPROM** automatically are also written to the **RAM**, and data that are read from the **EEPROM** automatically are also read to the **RAM**.

If the **EEPROM** option is selected, a click on **SEND** writes the current parameters to the sensor's non-volatile **EEPROM** memory, and a click on **GET** reads the parameters from the sensor's **EEPROM**.

The **RAM** memory should always be used for parameterising the sensor. When suitable parameters have been found for the respective application, these parameters must be written to the sensor's **EEPROM** so that after restarting the sensor these parameters can be loaded from the **EEPROM** into the **RAM** memory.

### 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.

**TIP!** Once suitable parameters have been found for a certain application, these should always be saved in a file on the PC.

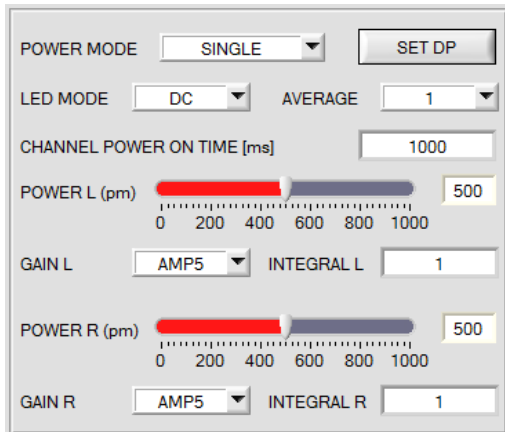
### GO [F11]:

A click on this button starts data transfer from the sensor to the PC through the serial RS232 interface.

### STOP [F12]:

A click on this button stops data transfer from the sensor to the PC through the serial RS232 interface.





**POWER MODE:** see SET DP

**POWER SOURCE:**

The sensor is equipped with two internal light sources.

**POWER SOURCE** is used to set how these are controlled.

**POWER CHL:** Only the left light source is active.

**POWER CHR:** Only the right light source is active.

**POWER CHL<->CHR:** The light sources are turned on and off one after the other. The time interval is defined by the parameter **CHANNEL POWER ON TIME [ms]**.

**INO CHL ON<->OFF:** The left light source is turned on and off through input **INO**. The right light source always is off.

**INO CHR ON<->OFF:** The right light source is turned on and off through input **INO**. The left light source always is off.

**INO CHL<->CHR:** Input **INO** is used to switch between the left (**INO=LO**) and right (**INO=HI**) light source.

**POWER:**

The transmitter power for the left and right light source is constantly kept at the value that is set at the respective slider **POWER L (or R) [pm]**. The **POWER** value 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 means the smallest intensity is set at the transmitter.

**LED MODE:**

This item serves for setting the control mode for the integrated light sources of the sensor.

**DC:** In this mode the sensor operates extremely fast and reaches a scan frequency of up to 35 kHz. 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:**

**GAIN** is used to set the amplification of the color receiver for every channel. 8 different amplification stages can be set (AMP1 to AMP8). **GAIN** should be set such that with a medium **POWER** value the sensor operates in its dynamic range (red, green, blue between 2750 and 3750).

In **AC** mode, **GAIN** directly influences the scan frequency. The scan-frequency table on the software CD shows the different scan frequencies.

**AVERAGE:**

This function field is used for adjusting the number of scanning values (measurement values) over which the raw signal measured at the color 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:**

In this function field the number of scan values (measurement values) over which the raw signal measured at the color receiver is summed up can be set for every channel. This integral function allows the reliable detection even of extremely weak signals. A higher **INTEGRAL** value increases the noise of the raw signals of the receiver unit, and simultaneously decreases the maximum achievable switching frequency of the sensor.

**INFO:**

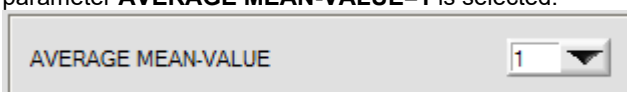
The **POWER** slider, **GAIN**, and **INTEGRAL** are only effective in **POWER MODE = SINGLE**.

**CHANNEL POWER ON TIME [ms]** only is effective with **POWER SOURCE = POWER CHL<->CHR**.

If a mode with only one channel is selected under **POWER SOURCE**, the respective other channel is inactive.

If input **INO** is used under **POWER SOURCE**, it is no longer available for **EXTEACH**.

With **POWER CHL<->CHR** it must be ensured for timing reasons that in the **COAST-STRUCT-SCOPE** software the parameter **AVERAGE MEAN-VALUE=1** is selected.



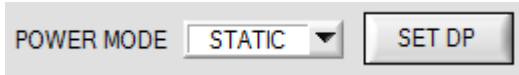
With all the other **POWER SOURCE** modes this parameter can be used for smoothing of the signal.

→ see manual **COAST-STRUCT-Scope**.



**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 vector calculation. To minimise this effect, the sensor in **PMODE = 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. The two 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 parameter sets. A parameter set is defined by **POWER, GAIN and INTEGRAL**.

**DOUBLE PARAMETER L (DP)**

	POW	GAIN	INT
1	500	6	1
2	900	6	1

**DOUBLE PARAMETER R (DP)**

	POW	GAIN	INT
1	800	5	2
2	800	7	3

**CORRECTION VALUES L**

R	G	B
1.68	1.66	1.71

**CORRECTION VALUES R**

R	G	B
6.87	6.61	4.71

**RGB VALUES L**

	R	G	B
BRIGHT DP1	3265	2944	2704
DARK DP1	2010	1706	1576
DARK DP2	3375	2832	2694

**RGB VALUES R**

	R	G	B
BRIGHT DP1	3148	2847	2758
DARK DP1	443	294	319
DARK DP2	3043	1944	1501

**DOUBLE PARAMETER SET L**     2     **DOUBLE PARAMETER SET R**     2

**? DOUBLE PARAMETER**

Push **START ASSISTANT** to adjust automatically proper **DOUBLE PARAMETERS**.

Follow these steps for manual setup:

Place the sensor to the brightest target.  
**POWER MODE** must be **SINGLE**.

Push **GO** and adjust a proper **POWER, GAIN** and **INTEGRAL** value so that the highest value of **RED, GREEN, BLUE (RGB)** is approximately 3000.

Push **ASSIGN 1** to assign the **DOUBLE PARAMETER** and **RGB values** of the bright target to the tables.

Now place the darker target to the sensor.

Push **ASSIGN 2** to assign the **RGB values** of the dark target to the table.

**Automatic setting of the 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 the double parameter sets:**

Set **POWER MODE=SINGLE**.

Place the brightest surface before the sensor, and select suitable **POWER, GAIN** and **INTEGRAL** values for both channels so that the respective brightest R, G, or B value is at approx. 3000 digits.

Then click on **ASSIGN 1** to write the double parameters to row 1 in the **DP** tables.

The RGB values are written to the **RGB VALUES** tables.

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 RGB values are written to the **DARK DP1** row of the **RGB VALUES** tables.

Now set **POWER, GAIN** and **INTEGRAL** for both channels such that the respective highest R, G, or B value is at approx. 3000.

Click on **ASSIGN 3** to write the double parameters to row 2 in the **DP** tables, and the RGB values to the **RGB VALUES** tables.

The correction values are calculated automatically and are written to the **CORRECTION VALUES** tables.

**INFO!**

The double parameter sets only become active when you click on **SEND!**

The tables only are used to display the values. They cannot be edited.

The **DOUBLE PARAMETER SET** displays show which double parameter set the respective channel currently uses. When the display shows 0, the sensor operates with the settings in the **PARA1** tab.

EXTEACH OFF ▼

▼ OFF

ON

**EXTEACH:**

It is possible to teach a vector externally through IN0 or by means of the button at the sensor housing.

**OFF:** External teaching is deactivated.

**ON :** The sensor can be taught up to 48 vectors through input IN0 or by means of the button at the sensor housing

Activate the external teach function with **EXTERN TEACH = ON**.

Adjust the power in such a way that the arriving signal is neither too high nor too low.

Select how many vectors you wish to teach externally.

Click on the **TEACH TABLE** tab to change to the **TEACH TABLE**.

MAXVEC-No. 4

	0	1	2	3	4	5
s L	0	0	0	0	0	0
i L	0	0	0	0	0	0
M L	0	0	0	0	0	0
VLEN L	0	0	0	0	0	0
DMM L	0	0	0	0	0	0
AREA L	0	0	0	0	0	0
EXPT L	0	0	0	0	0	0
DP L	0	0	0	0	0	0
s R	0	0	0	0	0	0
i R	0	0	0	0	0	0
M R	0	0	0	0	0	0
VLEN R	0	0	0	0	0	0
DMM R	0	0	0	0	0	0
AREA R	0	0	0	0	0	0
EXPT R	0	0	0	0	0	0
DP R	0	0	0	0	0	0

In this example a value of **MAXVEC-No.** = 4 was selected, i.e. the sensor should find a vector that was saved in the first 4 rows of the **TEACH TABLE** by way of external teaching trough IN0.

Select **EEPROM** in the settings field, and click on **SEND**.


The PC then is no longer needed, provided that you only wish to teach colors up to **MAXVEC-No.**.

**INFO:**

The teach vectors of course can be displayed at the PC at any time.

Vectors that are taught with **EXTEACH = ON** are saved to the **EEPROM** of the sensor, i.e. the information is not lost when the sensor is turned off.

This function rather is intended for "re-teaching", because the sensor does not automatically adapt its **POWER** value.

RAM      
       
 FILE

**EXTEACH=ON:**

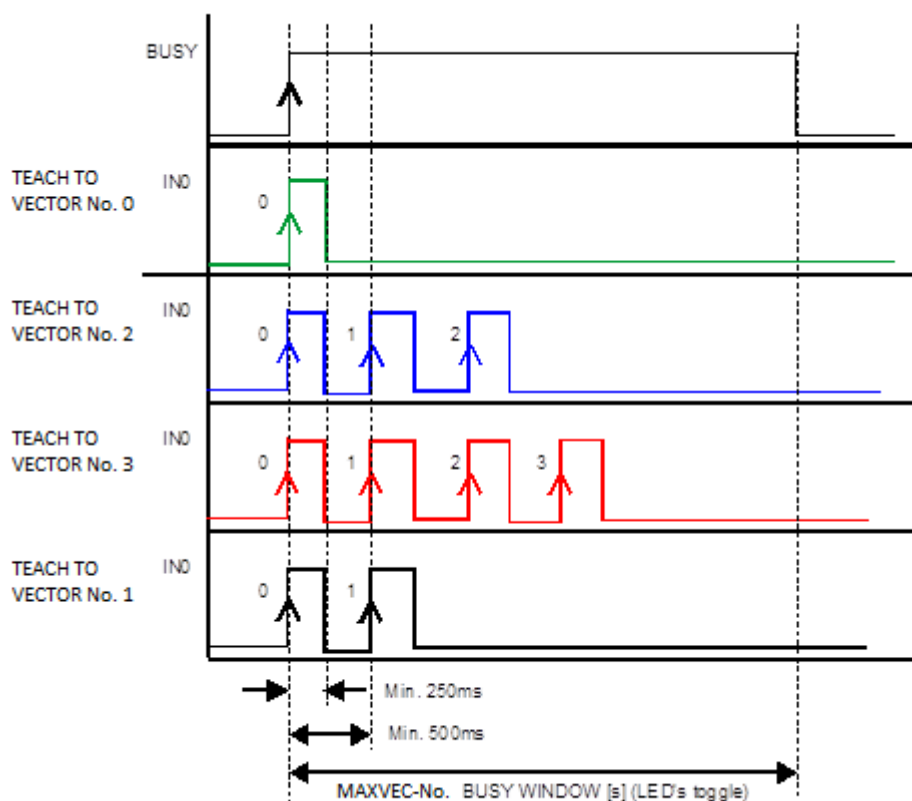
Before the external teach-process can be started, the surface (vector) to be taught must be presented to the sensor. The external teach process is started with a positive edge at IN0 (green wire). When such a positive edge is applied, the output LEDs (OUT0 ... OUT4) start to blink. From now on, the user has a certain number of seconds time to inform the sensor about the position at which the vector should be placed in the TEACH TABLE. The period of time, in which the user can teach the color information to the sensor depends on the value that is set in MAXVEC-No. (e.g. approx. 5 seconds when MAXVEC-No. = 5).

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 surface (vector) to position 3 in the TEACH TABLE, the following steps have to be performed:

1. Start of the external TEACH process with a positive edge (0) at IN0 → Position 0 is selected, the LEDs start to blink.
2. An additional positive edge (1) selects position 1 in the TEACH TABLE. This position 1 is indicated by the LEDs. The LED with binary value 1 remains on HIGH level all the time, while the other 3 LEDs continue to blink.
3. An additional positive edge (2) selects position 2 in the TEACH TABLE. This position 2 is indicated by the LEDs. The LED with binary value 2 remains on HIGH level all the time, while the other 3 LEDs continue to blink.
4. An additional positive edge (3) selects position 3 in the TEACH TABLE. This position 3 is indicated by the LEDs. The LEDs with binary value 3 remain on HIGH level all the time, while the other 2 LEDs continue to blink.
5. The desired position is now selected.
6. When the BUSY window is over (the LEDs stop to blink), the sensor starts the evaluation.
7. To teach another vector → go to pos. 1



INTLIM

**INTLIM:**

This edit box is used for setting an intensity limit. If the current intensity **M** arriving at the receiver unit falls below this limit, evaluation will no longer be performed, and an error state will be output.

**Attention: ERROR STATE if:  $M < INTLIM$**

MAXVEC-No.

**MAXVEC-No.:**

This function field is used to set the number of vectors that should be checked.

With **OUTMODE = BINARY HI** or **BINARY LO** a maximum of 31 vectors or vector groups can be taught.  
 With **OUTMODE = DIRECT HI** or **DIRECT LO** a maximum of 5 vectors or vector groups can be taught.  
 If **VECTOR GROUPS = ON**, the sensor can be taught up to 48 vectors.  
 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 sensor. If the number of vectors to be checked is lower, the sensor operates faster. The numerical value that is defined here refers to the number of columns (starting with column 0) in the **TEACH TABLE** (→ Tab TEACH).

OUTMODE

**OUTMODE:**

This function field is used to determine the operating mode of the digital outputs.

**DIRECT HI:**

In this mode a maximum of 5 teach vectors or 5 vector groups is permitted.  
 If in the row-by-row comparison the current vector values correspond with the teach parameters entered in the **TEACH TABLE**, the respective digital output (OUT0 to OUT4) is switched to HIGH.  
 If no vector is detected the digital outputs are in LOW state.

**DIRECT LO:**

In this mode a maximum of 5 teach vectors or 5 vector groups is permitted.  
 If in the row-by-row comparison the current vector values correspond with the teach parameters entered in the **TEACH TABLE**, the respective digital output (OUT0 to OUT4) is switched to LOW.  
 If no vector is detected the digital outputs are in HIGH state

**BINARY HI:**

In this mode a maximum of 31 teach vectors or 31 vector groups is permitted.  
 If in the row-by-row comparison the current vector values correspond with the teach parameters entered in the **TEACH TABLE**, this "hit" in the **TEACH TABLE** is provided as a **bit pattern** at the digital outputs (OUT0 to OUT4).

**BINARY LO:**

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

EVALUATION MODE MIN DIST

	0	1	2	3	4	5
s L	5167	5203	5394	5130	0	0
i L	2073	2040	2239	2126	0	0
M L	801	875	252	498	0	0
VLEN L	275	280	264	264	0	0
DMM L	377	383	370	370	0	0
AREA L	2806	2753	3263	3263	0	0
EXPT L	924	917	876	876	0	0
DPL	0	0	0	0	0	0
s R	5415	5435	5258	5258	0	0
i R	2158	2193	2168	2168	0	0
M R	294	314	211	211	0	0
VLEN R	181	182	174	174	0	0
DMM R	243	239	228	228	0	0
AREA R	2687	2746	2779	2779	0	0
EXPT R	1070	1051	1055	1055	0	0
DPR	0	0	0	0	0	0
GRP	0	0	1	1	0	0

**EVALUATION MODE:**

This display is used to set the evaluation mode.

**MIN DIST:**

Of the current surface the data **s**, **i**, **M**, **VLEN**, **DMM**, **AREA**, **EXPT** are provided for the left and right channel each.

With these data and with the corresponding data in the **TEACH TABLE** the respective **deltas(Δ)** are then determined.

These are then entered in the formula, and the result is a **deltaC** for every teach vector.

The vector with the smallest **delta C** value (hit) is then output.

$$\text{deltaC} = \sqrt{\begin{matrix} \Delta L_s^2 + \Delta L_i^2 + \Delta L_M^2 + \Delta L_{VLEN}^2 + \\ \Delta L_{DMM}^2 + \Delta L_{AREA}^2 + \Delta L_{EXPT}^2 + \\ \Delta R_s^2 + \Delta R_i^2 + \Delta R_M^2 + \Delta R_{VLEN}^2 + \\ \Delta R_{DMM}^2 + \Delta R_{AREA}^2 + \Delta R_{EXPT}^2 \end{matrix}}$$

It is a basic prerequisite, however, that the current intensity (**M**) is not lower than **INTLIM**, and that each of the two channels operates with the correct double parameter set **DPL** and **DPR**.

The double parameter set for the left and right channel also is saved in the **TEACH TABLE**.

**MIN DIST GRP:**

With **MIN DIST GRP** an additional **GRP** row appears in the teach table. In this row various teach vectors can be assigned to a group.

Evaluation then is performed as follows:

Of the current surface the color coordinates **s**, **i**, **M** are provided for the left and right channel each.

With these data and with the corresponding data in the **TEACH TABLE** the respective **deltas(Δ)** are then determined.

These are then entered in the formula

$$\text{deltaC Farbe} = \sqrt{\Delta L_s^2 + \Delta L_i^2 + \Delta L_M^2 + \Delta R_s^2 + \Delta R_i^2 + \Delta L R_M^2}$$

and the result is a **deltaC color** for every teach vector.

The vector with the shortest distance then is a color hit.

This color hit belongs to the group that was defined in the teach table.

Evaluation then is performed as with **MIN DIST**.

In contrast to **MIN DIST**, however, only those teach vectors are used that belong to the group that was determined as color hit.

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

**Tip!** The teach vectors only are activated when **SEND** is pressed!

## 2.3 Tab PARA2

**VECTOR GROUPS**

ON

	GROUP
0	0
1	0
2	1
3	1
4	2
5	0
6	0
7	0
8	0
9	0
10	0
11	0
12	0
13	0
14	0
15	0
16	0

RESET

**SELECT HOLD [ms]  
FOR EACH ROW**

	HOLD
255	10
0	10
1	10
2	10
3	10
4	10
5	10
6	10
7	10
8	10
9	10
10	10
11	10
12	10
13	10
14	10
15	10

RESET    10

V-No:

3

GRP

1

### PARA2:

A click on **PARA2** opens a view where vectors in the **TEACH TABLE** can be assigned to certain groups, and for each vector and the error status an explicit **HOLD** time can be set.

#### Please note:

**The sensor must be informed about the vector groups and HOLD settings by pressing SEND.**

#### VECTOR GROUPS:

If **VECTOR GROUPS = ON** it is possible to form vector groups. This means that in a special table the individual teach-in vectors are assigned to a group.

In this example, **VECTOR GROUPS** has been set to **ON**, i.e. group evaluation is activated.

Columns 0 and 1 have been assigned to group 0.

Columns 2 and 3 have been assigned to group 1, and column 4 to group 2.

A **GRP** display is shown under the **V-No:** display.

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

The group number will be output through outputs OUT0 to OUT4.

A total number of 48 different vectors 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.

## 2.4 Tab TEACH

Data transfer from the sensor to the PC is started when the **GO** button is pressed. The respective data of the left (L) and right (R) channel are visualised by means of the bars and in the display fields.

The screenshot shows the TEACH software interface. On the left is a data table with columns 0-5 and rows for parameters sL, iL, mL, VLEN L, DMM L, AREA L, EXPT L, DP L, sR, iR, MR, VLEN R, DMM R, AREA R, EXPT R, and DPR. A green box highlights the first column (0) and the first two rows (sL, iL). On the right, a summary bar chart shows data for L and R channels across parameters s, i, M, VLEN, DMM, AREA, and EXPT. Below this are two stacked bar charts labeled 'DELTA L' and 'DELTA R' with numerical values at the bottom.

	0	1	2	3	4	5
s L	5207	5175	5214	0	0	0
i L	1960	2103	1994	0	0	0
M L	862	469	910	0	0	0
VLEN L	165	272	134	0	0	0
DMM L	237	327	197	0	0	0
AREA L	2773	3010	2737	0	0	0
EXPT L	661	120	447	0	0	0
DP L	0	0	0	0	0	0
s R	5466	5459	5481	0	0	0
i R	2183	2185	2199	0	0	0
M R	352	339	361	0	0	0
VLEN R	147	263	219	0	0	0
DMM R	184	323	267	0	0	0
AREA R	2601	3037	3072	0	0	0
EXPT R	561	120	348	0	0	0
DPR	0	0	0	0	0	0

s	i	M
5197	2086	828
5256	2053	675

VLEN	DMM	AREA	EXPT
286	719	645	517
315	669	642	517

DPS
0
0

The upper data represent the measurement values of the left channel, the bottom data are the values of the right channel.

### s i M:

These numeric output fields show the RED-GREEN component **s**, the BLUE-YELLOW component **i** and the intensity **M** of the scattered light currently arriving at the color receiver.

### VLEN DMM AREA EXPT:

These numeric output fields show the data coming from the structure sensor.

### DP SET:

This display field shows which **DOUBLE PARAMETER SET** the sensor currently uses. With **POWER MODE = SINGLE** this value always is 0, i.e. the sensor operates with the values for **POWER**, **INTEGRAL** and **GAIN** that are set in the **PARA1** tab. With **POWER MODE = DOUBLE** the sensor switches between two parameter sets (1 and 2) (see **SET DP**).

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



	0	1	2	3	4	5
s L	5207	5175	5214	0	0	0
i L	1960	2103	1994	0	0	0
M L	862	469	910	0	0	0
VLEN L	165	272	134	0	0	0
DMM L	237	327	197	0	0	0
AREA L	2773	3010	2737	0	0	0
EXPT L	661	120	447	0	0	0
DP L	0	0	0	0	0	0
s R	5466	5459	5481	0	0	0
i R	2183	2185	2199	0	0	0
M R	352	339	361	0	0	0
VLEN R	147	263	219	0	0	0
DMM R	184	323	267	0	0	0
AREA R	2601	3037	3072	0	0	0
EXPT R	561	120	348	0	0	0
DP R	0	0	0	0	0	0

TEACH DATA TO No.: 0 Inc

**TEACH:**

A click on the **TEACH** tab opens a view where vectors can be taught to the **TEACH TABLE**.

**Please note: The sensor must be informed about the teach-in vectors by pressing SEND.**

After a left mouse button double click (or a click on shortcut key button F2) on the respective field the default values can be changed by entering numerical values with the PC keyboard.

The **TEACH TABLE** is organized in columns, i.e. the individual parameters for the teach-in vectors are arranged one below the other in the respective column.

The sensor is able to check up to 48 teach-in vectors. The number of the respective teach-in vector is given in the upper row of the table.

Only the green columns are used for evaluation in the sensor. The number of columns to be inspected is set with **MAXVEC-No.**

When the **TEACH DATA TO** button is pressed the currently displayed data are written to the column in the **TEACH TABLE** that is selected under **No.:**

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 column in the **TEACH TABLE** is selected.

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

	0	1	2	3	4
s L	5197	5088	5086	0	0
i L	2086	2005	2074	0	0
M L	828	424			
VLEN L	286	288			
DMM L	719	586	1235	0	0
AREA L	645	489	684	0	0

With **Set selection to** several cells in the **TEACH TABLE** can be filled with a value.

For this purpose the cells that should be changed must first be selected.

A right mouse click then opens a popup menu.

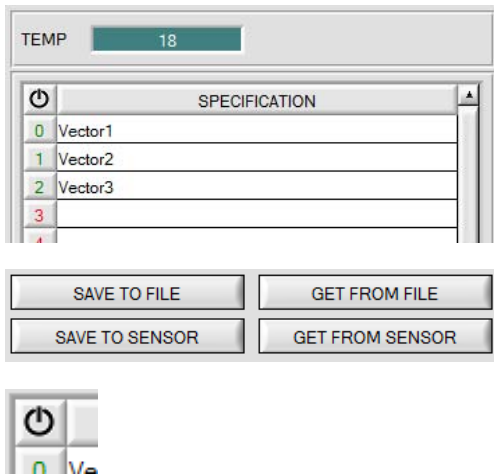
If you click on **Set selection to** there, a popup window will open where you can enter the value to which the cells should be changed.

A click on **Reset selection** resets the selected cells to 0.

**VALUE!** X

Insert a cell value!

## 2.5 Tab GEN



### GEN:

A click on **GEN** opens a display that shows the current temperature value **TEMP** in the sensor housing.

This display is **NOT** in degrees Celsius or Fahrenheit

In the GEN tab designations for the individual vectors can be entered in the **SPECIFICATION** table.

With **SAVE TO FILE** and **GET FROM FILE** the **SPECIFICATION** table can be saved to hard disk or loaded from there.

**SAVE TO SENSOR** and **GET FROM SENSOR** writes it to the sensor or loads it from the sensor.

The Reset button can be used to reset 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.

## 2.6 Tab REC

The COAST-Scope software features a data recorder that makes it possible to save the data that are acquired and calculated by the sensor. The recorded file is saved to the hard disk of the PC and can then be evaluated with a spreadsheet program.

The file that is created has as many rows as data frames were recorded.

A row is structured as follows:

**Date, time, DATA**

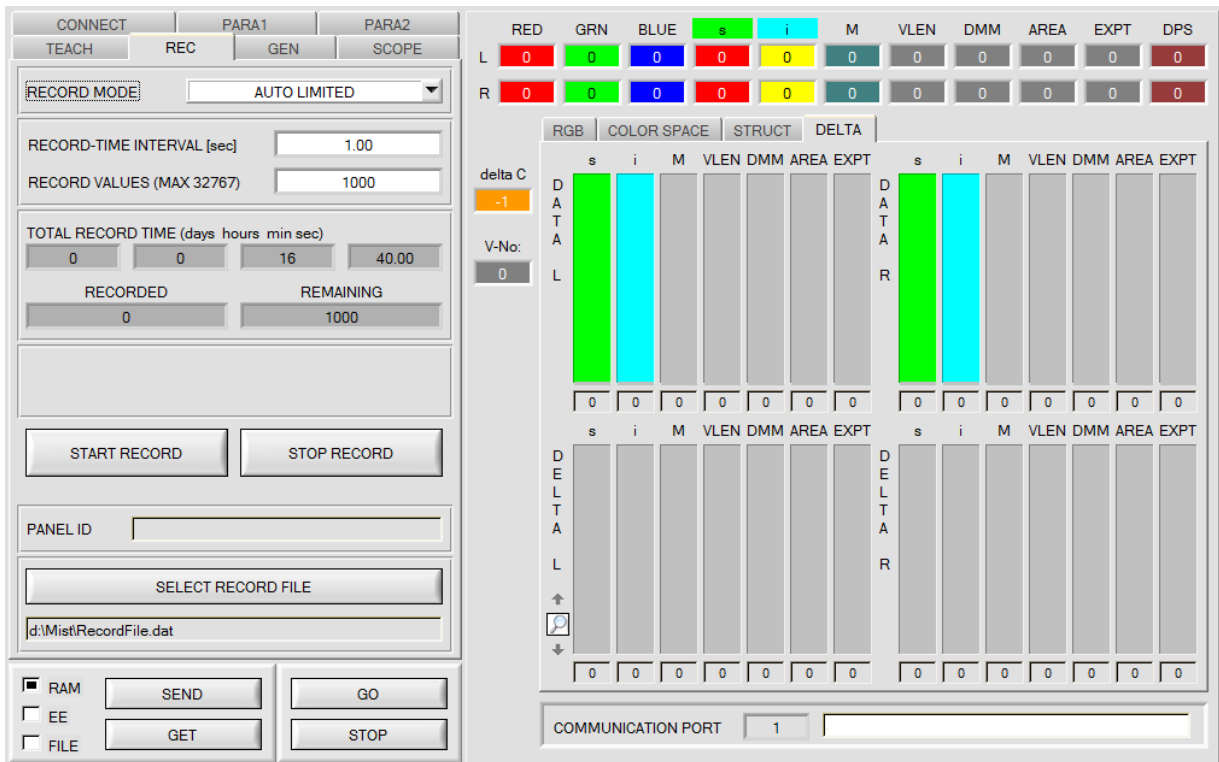
The following steps describe how data frames are recorded with the recorder:

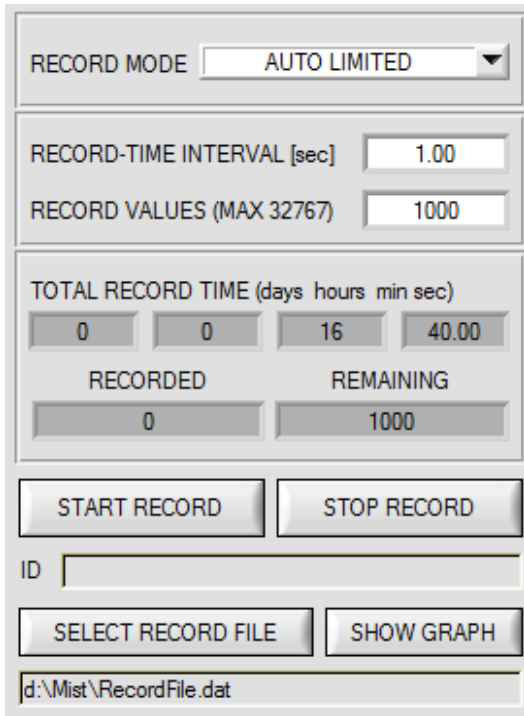
**Please note:**

Recording depends on the **EVALUATION MODE** that is chosen. Several **EVALUATION MODES** do not need certain data, therefore these data will be set to 0, i.e. for these data the value 0 will be recorded.

**Step 1:**

When the **REC** button is pressed, the following window will be displayed:





**Step 2:**

If you want to automatically record several data frames, please select **AUTO LIMITED** under **RECORD MORE**. 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 dimmed 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 MORE**. Then select the desired recording interval and press **START RECORD**.

If you want to record data "manually", select the **MANUAL RECORDING** function under **RECORD MORE**. 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.

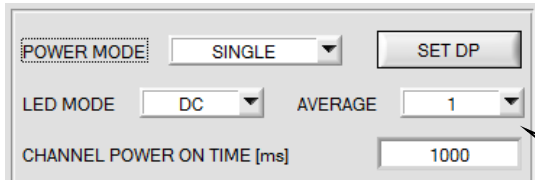
Pressing the **STOP RECORD** button terminates the automatic sending function of the sensor again.

**Please note:**

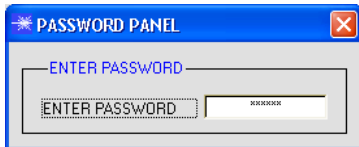
When you press **START RECORD**, the file that is selected under **SELECT RECORD FILE** will be deleted. With **RECORD FRAME MANUALLY**, the file will be created if it does not already exist. If the file already exists, the data are added to the existing file.

## 2.7 Tab Offset Calibration

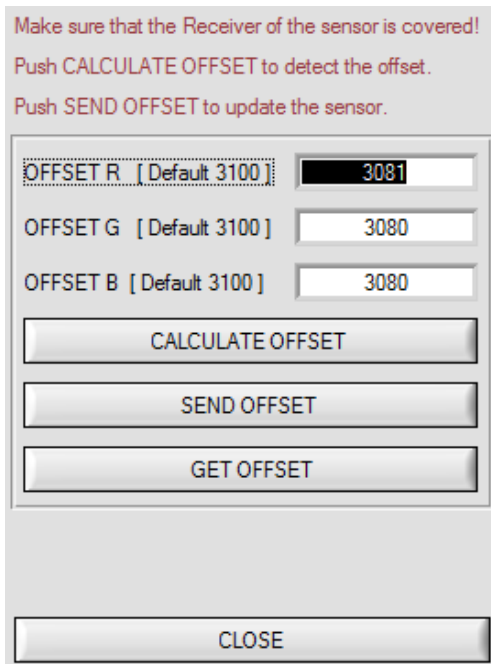
To avoid an increase of the electronic offset when using the integral function (**INTEGRAL** parameter), this offset can be eliminated by way of offset calibration or zero-point calibration. The corresponding tab is password-protected to prevent inadvertent incorrect settings.



Offset calibration can be accessed by double-clicking with the right mouse button exactly at the frame in the **PARA 1** 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. with 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.

## 2.8 Tab SCOPE

The SCOPE tab visualises an oscilloscope.

With **TRIG MODE** either the **R G B** signals or the color space coordinates (**Color Space**) or the structure values, as well as the states of the digital outputs and of digital input IN0 are displayed.

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]**. Please give the sensor 8 seconds to determine the correct scan frequency before you click on **GET CYCLE TIME**.

In **TRIG MODE = SINGLE SHOT** a click on **SCAN** records a data frame and displays it in the graph.

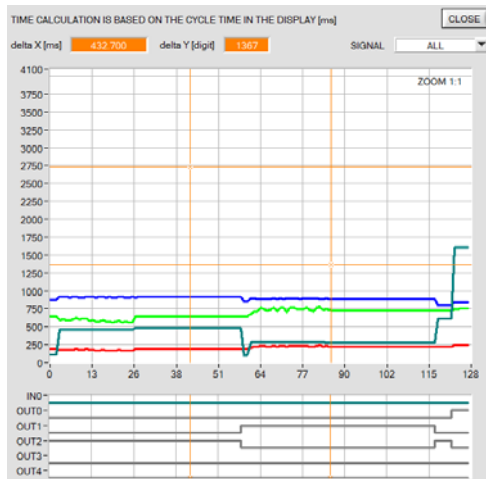
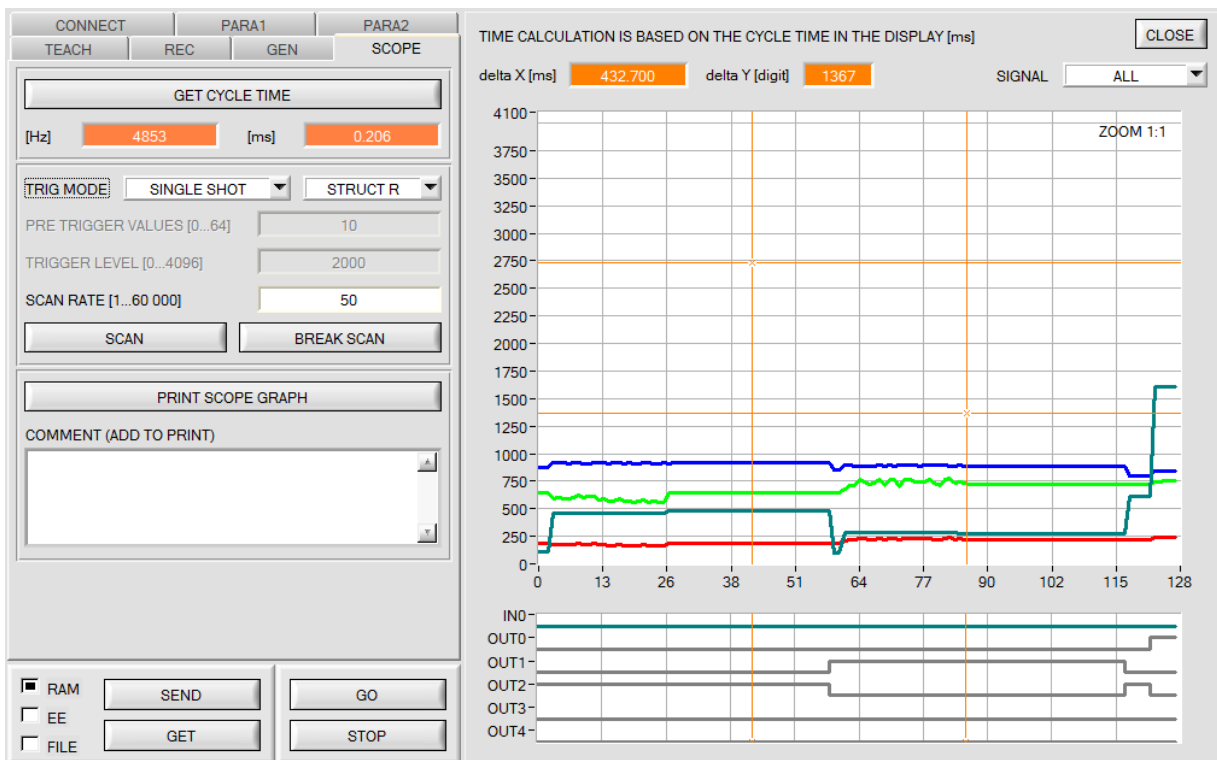
In **TRIG MODE = FALLING EDGE** and **RIISING EDGE** a click on **SCAN** starts triggered recording. A trigger start can be defined with **TRIGGER LEVEL**.

Triggering is done either to **BLUE**, **M** or **AREA**, depending on which signal shall be recorded. In the graph this is the blue line.

In **TRIG MODE= INTERN V-No.0** recording starts stand-alone once V-No. 0 is detected.

With **TRIG MODE= EXTERN IN0** recording can be started external via 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]**, **deltaY[digit]** and **deltaY[V]** will be updated.

**deltaX[ms]** shows the time between the cursors in X direction. **deltaY[digit]** or **[V]** show the difference between the two cursors in digits or in Volt in Y direction.

**SIGNAL** can be used to display individual curves.

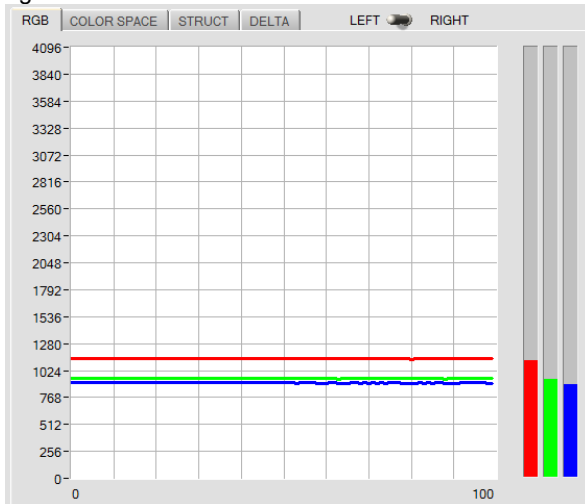
**PRINT SCOPE GRAPH** prints the current screen together with the text in the **COMMENT** text field.

## 2.9 Graphic display elements

### RGB tab:

Display of the current raw signals of the 3-fold receiver (red, green, blue).

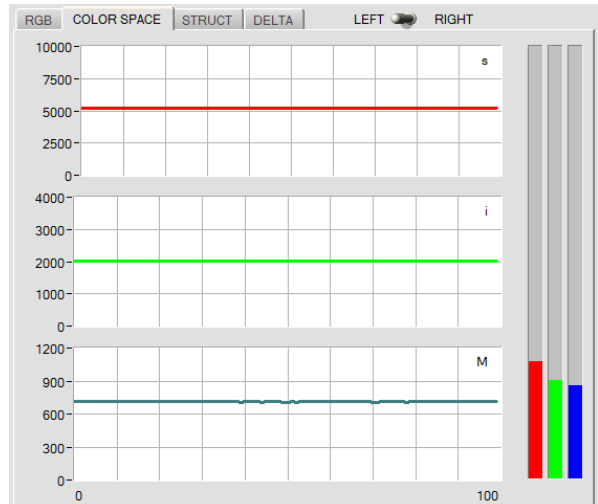
The switch can be used to select the data of the left or right channel.



### COLOR SPACE tab:

Display of the current color space coordinates (s, i, M).

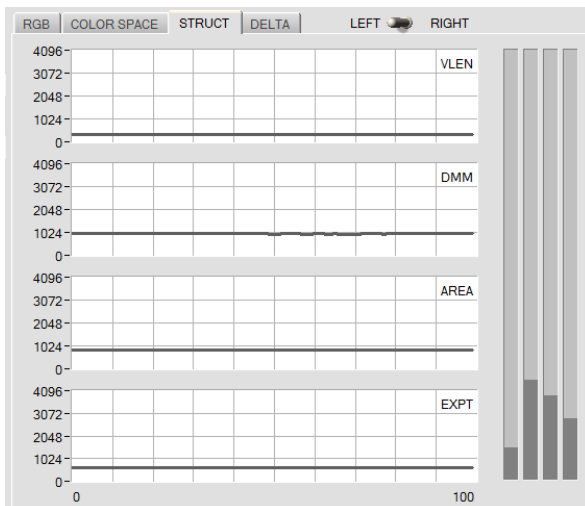
The switch can be used to select the data of the left or right channel.



### STRUCT tab:

Display of the current data of the structure sensor.

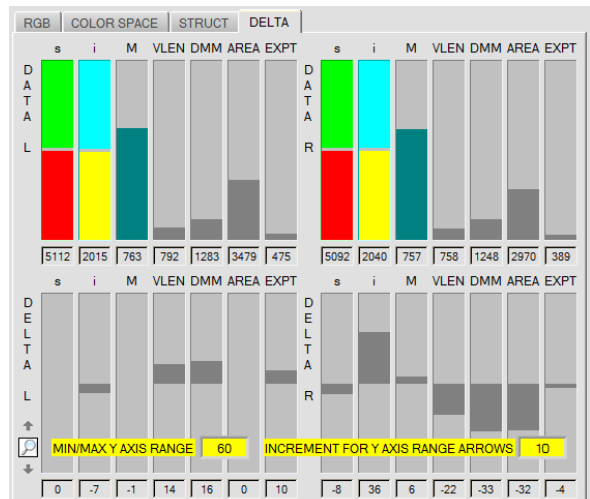
The switch can be used to select the data of the left or right channel.



### DELTA tab:

All data of the left and right channel that are needed for evaluation are shown in bars and displays.

The deviations from the detected surface (hits) also are displayed.



With a click on the magnifying glass, two input fields are displayed, which make it possible to set the value range of the Y axis of the bars.

**MIN / MAX Y AXIS RANGE** sets the range.

With the arrows up and down, this range is increased or decreased by the value **INCREMENT FOR Y AXIS RANGE ARROW**.

By double-clicking the windows are hidden again.



TEMP 27

**TEMP:**

This display shows the temperature prevailing in the sensor housing.  
The display **DOES NOT** show degrees Centigrade or Fahrenheit.

V-No:  
0

**V-No.:**

This numerical value output field shows the currently detected vector number according to the entry in the **TEACH TABLE**. The currently detected vector number is sent to the digital outputs OUT0 ... OUT4 as a corresponding bit pattern.  
A value of 255 means that none of the taught vectors was detected.  
A double-click on the display will open a larger display window

delta C  
62

**delta C:**

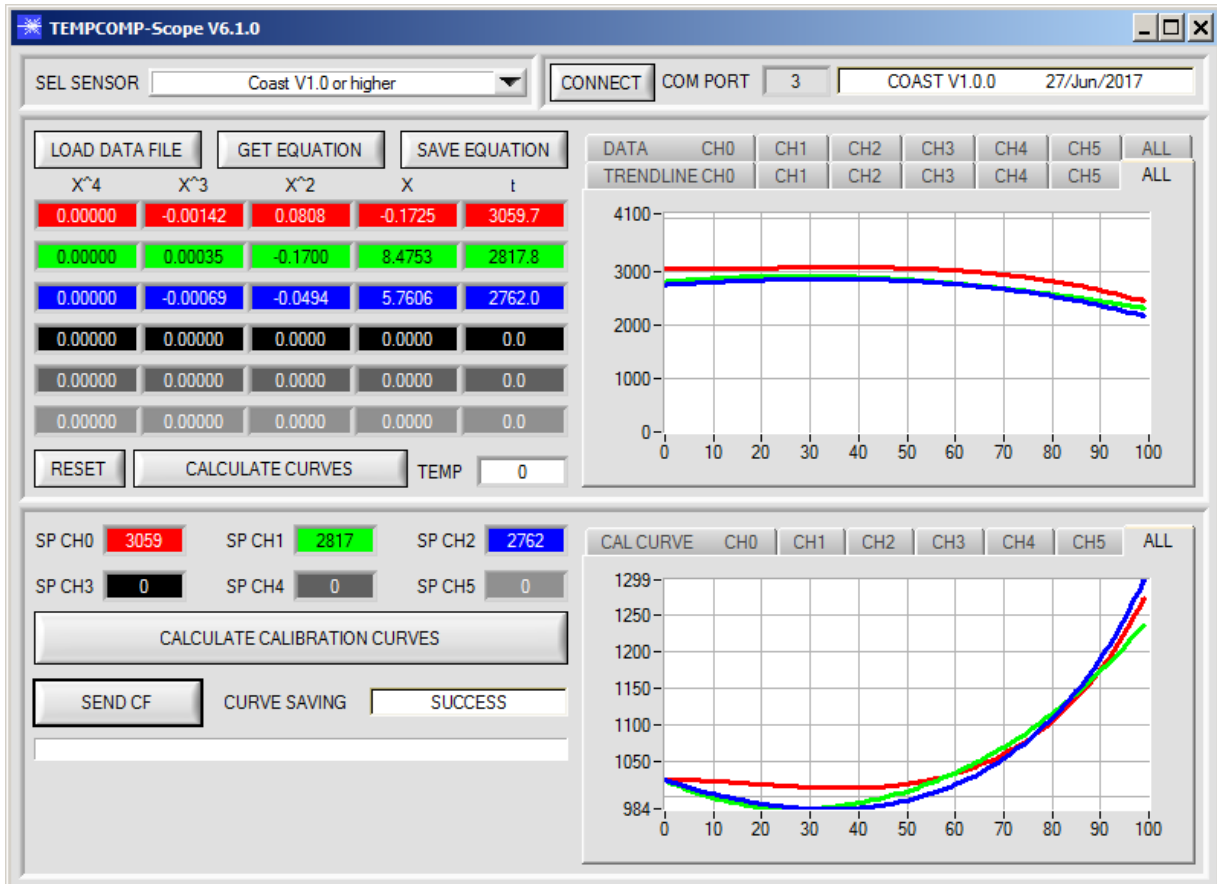
This display shows the deviation from a vector hit.

**delta C** is calculated as described in **EVALUATION MODE**.

### 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.

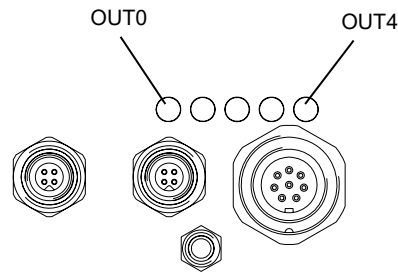


- Step 1: Load the temperature compensation file that you have received from your supplier with **GET EQUATION** or **LOAD DATA FILE**.
- Step 2: Press **CALCULATE CURVES** to display the data in the graph.
- Step 3: Select the sensor-internal operating temperature (not in °C) that the sensor has at 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. Function of the LED display

LED display:

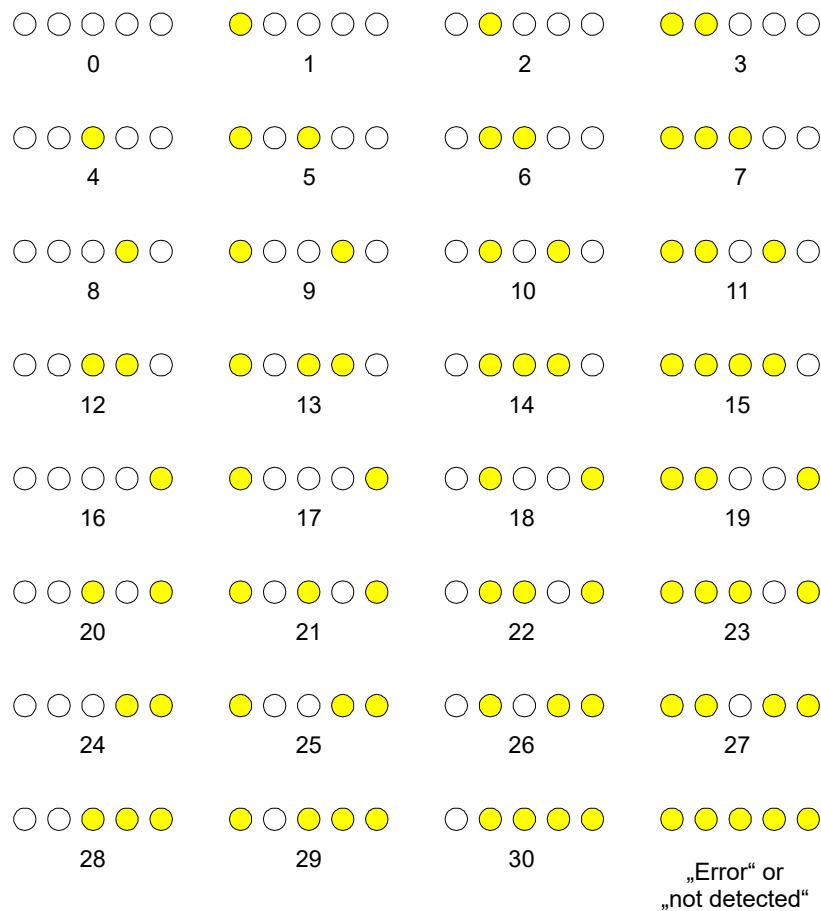


### BINARY

The vector code is visualised by way of 5 yellow LEDs at the housing of the COAST sensor. At the same time in the binary mode (OUT BINARY HI) the vector code indicated on the LED display is output as 5-bit binary information at the digital outputs OUT0 to OUT4 of the 8-pin COAST/PLC socket.

The COAST sensor is able to process a maximum of 31 vectors or vector groups (0 ... 30) in accordance with the corresponding columns in the TEACH TABLE. An "error" respectively a "not detected vector" is displayed by the lighting of all LED (OUT0 ... OUT4 digital outputs are set to HIGH-level).

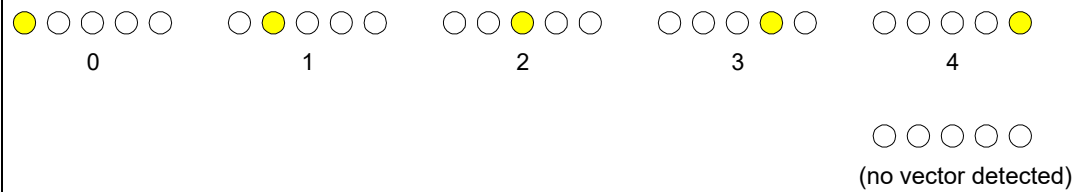
With OUTMODE BINARY LO the output is inverse.



**DIRECT HI:**

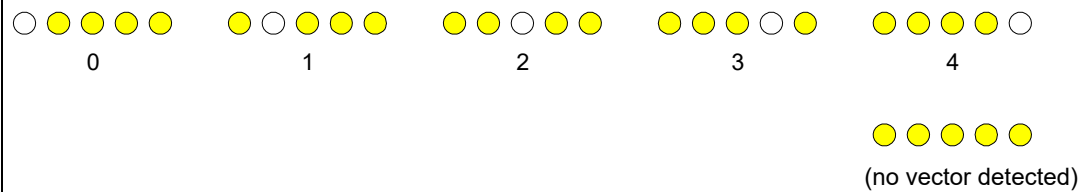
In the DIRECT mode (OUT DIRECT HI or OUT DIRECT LO) the maximum numbers of vectors to be taught is 5 (no. 0, 1, 2, 3, 4).

If the selector switch is set to DIRECT HI, the respective digital output is HI, and the other four are LO. If no vector was detected the digital outputs are in LO state (no LED lights up).




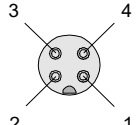
**DIRECT LO:**

If the selector switch is set to DIRECT LO, the respective digital output is LO, and the other four are HI. If no vector was detected the digital outputs are in HI state (all LEDs light up).



## 5. Connector assignment of the COAST sensor


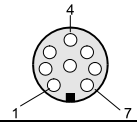
### Connection of COAST to PC:

<b>4-pole fem. M5 connector (type Binder 707)</b> <b>COAST/PC-RS232</b>		 
Pin-No.:		Assignment:
1		+24VDC (+Ub)
2		0V (GND)
3		Rx0
4		Tx0

#### Connecting cables to choose from:

cab-las4/PC-...  
 cab-4/USB-...  
 cab-4/ETH-...

### Connection of COAST to PLC:

<b>8-pole female connector (type Binder 712)</b> <b>COAST/PLC</b>		 
Pin No.:	Color of wire: (cab-las8/SPS)	Assignment:
1	white	0V (GND)
2	brown	+24VDC ( $\pm 10\%$ ) (+Ub)
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-las8/SPS-...

## 6. RS232 Communication protocol

The sensors of the COAST 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**

Parameter	Type	Meaning
<b>Para1:</b> POWER SOURCE	Word	Select Transmitter source POWER CHL, POWER CHR, POWER CHL<->CHR, IN0 CHL ON<->OFF, IN0 CHR ON<->OFF, IN0 CHL<->CHR code to (0, 1, 2, 3, 4, 5)
<b>Para2:</b> CHANNEL POWER ON TIME [ms]	Word	Select the switch-on time for the light sources. (500...10000) [ms]
<b>Para3:</b> POWER MODE	Word	Transmitter mode: SINGLE, DOUBLE coded to (0, 1)
<b>Para4:</b> LED MODE	Word	Control for the internal light source DC, AC coded to (0,1)
<b>Para5:</b> AVERAGE	Word	Signal averaging 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096, 8192, 16384 or 32768
<b>Para6:</b> POWER L (pm)	Word	Left channel transmitter intensity (0...1000). Intensity in thousandth!
<b>Para7:</b> POWER R (pm)	Word	Right channel transmitter intensity (0...1000). Intensity in thousandth!
<b>Para8:</b> GAIN L	Word	Left channel amplification of the integrated receiver AMP1, AMP2, AMP3, AMP4, AMP5, AMP6, AMP7, AMP8 coded to (1, 2, 3, 4, 5, 6, 7, 8)
<b>Para9:</b> GAIN R	Word	Right channel amplification of the integrated receiver AMP1, AMP2, AMP3, AMP4, AMP5, AMP6, AMP7, AMP8 coded to (1, 2, 3, 4, 5, 6, 7, 8)
<b>Para10:</b> INTEGRAL L	Word	Left channel signal integration (1...250)
<b>Para11:</b> INTEGRAL R	Word	Right channel signal integration (1...250)
<b>Para12:</b> EVALUATION MODE	Word	Evaluation mode: MIN DIST, MIN DIST GRP coded to (0,1)
<b>Para13:</b> MAXVEC-No.	Word	Number of the vectors (1,2,3,...,48)
<b>Para14:</b> OUTMODE	Word	Function of the digital output: DIRECT HI, DIRECT LO, BINARY HI, BINARY LO coded to (0,1,2,3)
<b>Para15:</b> INTLIM	Word	Intensity limit (0 ... 4095)
<b>Para16:</b> EXTEACH	Word	External teach mode: OFF, ON coded to (0,1)
<b>Para17:</b> VECTOR GROUPS	Word	Vector groups enable: OFF, ON coded to (0,1)
<b>Para18:</b> HOLD for V-No: 255	Word	Hold time for failure condition (V-No: 255) coded to (0...100) [ms]
<b>Para19:</b> POWER DP1 L	word	Left channel transmitter intensity of Double Parameter Set 1 (0 ... 1000) Attention intensity in thousandth!
<b>Para20:</b> POWER DP1 R	word	Right channel transmitter intensity of Double Parameter Set 1 (0 ... 1000) Attention intensity in thousandth!
<b>Para21:</b> GAIN DP1 L	word	Left channel 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)
<b>Para22:</b> GAIN DP1 R	word	Right channel 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)
<b>Para23:</b> INTEGRAL DP1 L	word	Left channel signal integration of Double Parameter Set 1 (1...250)
<b>Para24:</b> INTEGRAL DP1 R	word	Right channel signal integration of Double Parameter Set 1 (1...250)
<b>Para25:</b> POWER DP2 L	word	Left channel transmitter intensity of Double Parameter Set 2 (0 ... 1000) Attention intensity in thousandth!
<b>Para26:</b> POWER DP2 R	word	Right channel transmitter intensity of Double Parameter Set 2 (0 ... 1000) Attention intensity in thousandth!
<b>Para27:</b> GAIN DP2 L	word	Left channel 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)
<b>Para28:</b> GAIN DP2 R	word	Right channel 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)
<b>Para29:</b> INTEGRAL DP2 L	word	Left channel signal integration of Double Parameter Set 2 (1...250)
<b>Para30:</b> INTEGRAL DP2 R	word	Right channel signal integration of Double Parameter Set 2 (1...250)
<b>Para31:</b> COR VAL R L	word	Left channel Correction Value for R
<b>Para32:</b> COR VAL R R	word	Right channel Correction Value for R
<b>Para33:</b> COR VAL G L	word	Left channel Correction Value for G
<b>Para34:</b> COR VAL G R	word	Right channel Correction Value for G
<b>Para35:</b> COR VAL B L	word	Left channel Correction Value for B
<b>Para36:</b> COR VAL B R	word	Right channel Correction Value for B

One Column in the Teach Table and one row in the Group Table and Hold Table determine a **TEACH VECTOR**. The sensors of the COAST 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**

TEACH VECTOR		Type	Meaning
<b>TeachVal1:</b>	Teach Table Column 0 Row 0	Word	Left channel s of Column 0
<b>TeachVal2:</b>	Teach Table Column 0 Row 1	Word	Left channel i of Column 0
<b>TeachVal3:</b>	Teach Table Column 0 Row 2	Word	Left channel M of Column 0
<b>TeachVal4:</b>	Teach Table Column 0 Row 3	Word	Left channel VLEN of Column 0
<b>TeachVal5:</b>	Teach Table Column 0 Row 4	Word	Left channel DMM of Column 0
<b>TeachVal6:</b>	Teach Table Column 0 Row 5	Word	Left channel AREA of Column 0
<b>TeachVal7:</b>	Teach Table Column 0 Row 6	Word	Left channel EXPT of Column 0
<b>TeachVal8:</b>	Teach Table Column 0 Row 7	Word	Left channel DP of Column 0
<b>TeachVal9:</b>	Teach Table Column 0 Row 8	Word	Right channel s of Column 0
<b>TeachVal10:</b>	Teach Table Column 0 Row 9	Word	Right channel i of Column 0
<b>TeachVal12:</b>	Teach Table Column 0 Row 10	Word	Right channel M of Column 0
<b>TeachVal12:</b>	Teach Table Column 0 Row 11	Word	Right channel VLEN of Column 0
<b>TeachVal13:</b>	Teach Table Column 0 Row 12	Word	Right channel DMM of Column 0
<b>TeachVal14:</b>	Teach Table Column 0 Row 13	Word	Right channel AREA of Column 0
<b>TeachVal15:</b>	Teach Table Column 0 Row 14	Word	Right channel EXPT of Column 0
<b>TeachVal16:</b>	Teach Table Column 0 Row 15	Word	Right channel DP of Column 0
<b>TeachVal17:</b>	Teach Table Column 0 Row 16	Word	Group for color hit if EVALUATION MODE = MIN DIST GRP
<b>TeachVal18:</b>	Free	Word	Send 0 as dummy
<b>TeachVal19:</b>	Free	Word	Send 0 as dummy
<b>TeachVal20:</b>	Group Table Row 0	Word	Group to which Column 0 belongs if enabled
<b>TeachVal21:</b>	Hold Table Row 0	Word	Hold time for Column 0
<b>TeachVal22:</b>	Teach Table Column 1 Row 0	Word	Left channel s of Column 1
<b>TeachVal23:</b>	Teach Table Column 1 Row 1	Word	Left channel i of Column 1
<b>TeachVal24:</b>	Teach Table Column 1 Row 2	Word	Left channel M of Column 1
<b>TeachVal25:</b>	Teach Table Column 1 Row 3	Word	Left channel VLEN of Column 1
...	...	...	...
<b>TeachVal1003:</b>	Teach Table Column 47 Row 15	Word	Right channel DP of Column 47
<b>TeachVal1004:</b>	Teach Table Column 47 Row 16	Word	Group for color hit if EVALUATION MODE = MIN DIST GRP
<b>TeachVal1005:</b>	Free	Word	Send 0 as dummy
<b>TeachVal1006:</b>	Free	Word	Send 0 as dummy
<b>TeachVal1007:</b>	Group Table Row 47	Word	Group to which Column 47 belongs if enabled
<b>TeachVal1008</b>	Hold Table Row 47	Word	Hold time for Column 47



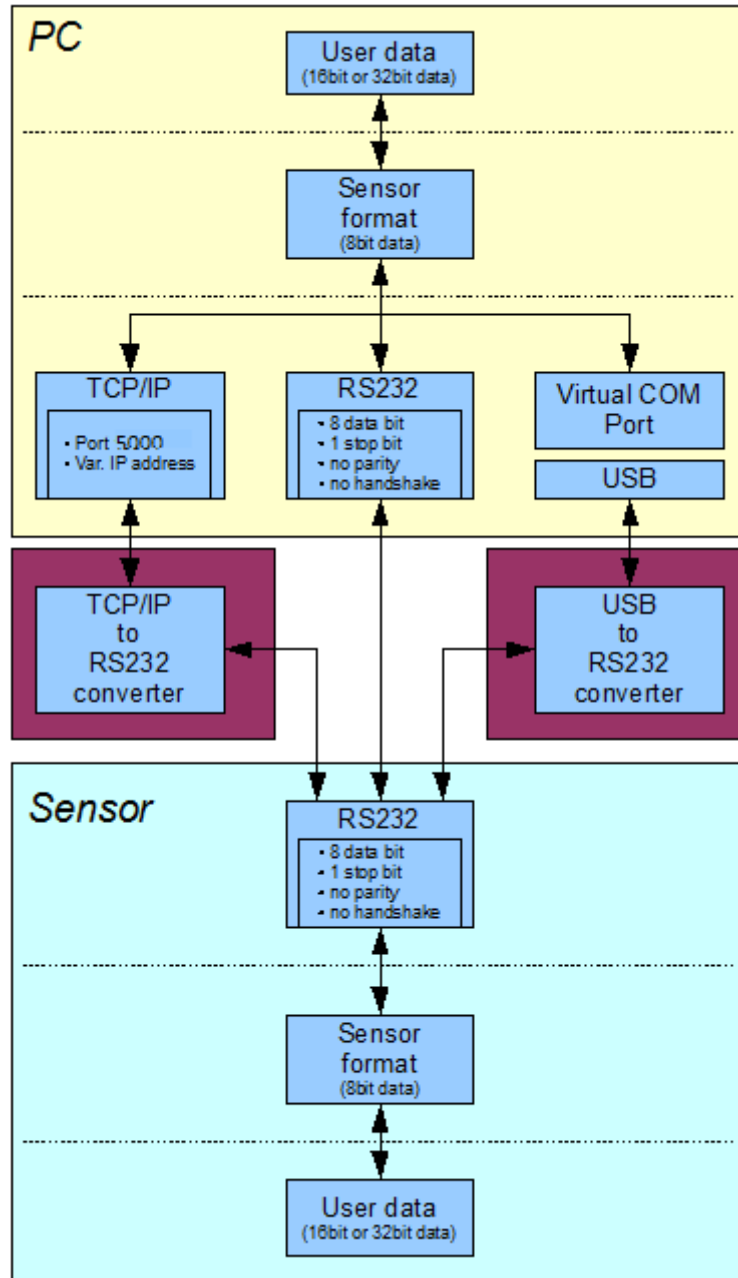
Upon request, the data acquired and processed by the sensor are sent by the sensor in the following sequence.  
 Info! 1 **bytes** = 8bit      1 **word** = 2 **byte**

DATA VALUE		Type	Meaning
<b>DatVal1:</b>	RED L	Word	Left channel calibrated and temperature compensated signal from channel red
<b>DatVal2:</b>	RED R	Word	Right channel calibrated and temperature compensated signal from channel red
<b>DatVal3:</b>	GREEN L	Word	Left channel calibrated and temperature compensated signal from channel green
<b>DatVal4:</b>	GREEN R	Word	Right channel calibrated and temperature compensated signal from channel green
<b>DatVal5:</b>	BLUE L	Word	Left channel calibrated and temperature compensated signal from channel blue
<b>DatVal6:</b>	BLUE R	Word	Right channel calibrated and temperature compensated signal from channel blue
<b>DatVal7:</b>	s L	Word	Left channel calculated s value
<b>DatVal8:</b>	s R	Word	Right channel calculated s value
<b>DatVal9:</b>	i L	Word	Left channel calculated i value
<b>DatVal10:</b>	i R	Word	Right channel calculated i value
<b>DatVal11:</b>	M L	Word	Left channel calculated M value
<b>DatVal12:</b>	M R	Word	Right channel calculated M value
<b>DatVal13:</b>	VLEN L	Word	Left channel VLEN value from structure sensor
<b>DatVal14:</b>	VLEN R	Word	Right channel VLEN value from structure sensor
<b>DatVal15:</b>	DMM L	Word	Left channel DMM value from structure sensor
<b>DatVal16:</b>	DMM R	Word	Right channel DMM value from structure sensor
<b>DatVal17:</b>	AREA L	Word	Left channel AREA value from structure sensor
<b>DatVal18:</b>	AREA R	Word	Right channel AREA value from structure sensor
<b>DatVal19:</b>	EXPT L	Word	Left channel EXPT value from structure sensor
<b>DatVal20:</b>	EXPT R	Word	Right channel EXPT value from structure sensor
<b>DatVal21:</b>	DP SET L	Word	Left channel Double Parameter Set
<b>DatVal22:</b>	DP SET R	Word	Right channel Double Parameter Set
<b>DatVal23:</b>	delta C	Word	Distance to a vector hit
<b>DatVal24:</b>	V-No:	Word	Detected vector
<b>DatVal25:</b>	GRP	Word	Detected group
<b>DatVal26:</b>	STATE IN0	Word	State of input IN0
<b>DatVal27:</b>	TEMP	Word	Temperature in the sensor (not in °C or °F)
<b>DatVal28:</b>	RAW RED L	Word	Left channel none calibrated and temperature compensated signal from channel red
<b>DatVal29:</b>	RAW RED R	Word	Right channel none calibrated and temperature compensated signal from channel red
<b>DatVal30:</b>	RAW GREEN L	Word	Left channel none calibrated and temperature compensated signal from channel green
<b>DatVal31:</b>	RAW GREEN R	Word	Right channel none calibrated and temperature compensated signal from channel green
<b>DatVal32:</b>	RAW BLUE L	Word	Left channel none calibrated and temperature compensated signal from channel blue
<b>DatVal33:</b>	RAW BLUE R	Word	Right channel none calibrated and temperature compensated signal from channel blue

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<sub>dez</sub> (55<sub>hex</sub>).

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<sub>dez</sub> or 00<sub>hex</sub>, 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 eighth 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
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 := AAhex
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 Set 0

ARG = 1 --> Teach Vector Column 0.....11

ARG = 2 --> Teach Vector Column 12...23

ARG = 3 --> Teach Vector Column 24...35

ARG = 4 --> Teach Vector Column 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 for Parameter Set 0

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 Set 0

ARG = 1 --> Teach Vector Column 0.....11

ARG = 2 --> Teach Vector Column 12...23

ARG = 3 --> Teach Vector Column 24...35

ARG = 4 --> Teach Vector Column 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=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			

$$\text{Cycle Time [Hz]} = \text{CYCLE COUNT} / (\text{COUNTER TIME} * 0,01)$$

$$\text{Cycle Time [ms]} = (\text{COUNTER TIME} * 0,01) / \text{CYCLE COUNT}$$

**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:

- IBM PC AT or compatible
- VGA graphics
- Microsoft® Windows® XP, VISTA, 7
- Serial RS232 interface at the PC
- Microsoft® compatible mouse
- Cable cab-las4/PC (cab-las5/PC) for the RS232 interface or cab-4/USB (cab-5/USB) for USB slot
- CD-ROM drive

The software can only be installed under Windows. Windows must therefore be started first, if it is not yet running.

Please install the software as described below:

1. The software can be installed directly from the installation CD-ROM. To install the software, start the SETUP program in the INSTALL folder of the CD-ROM.
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.**

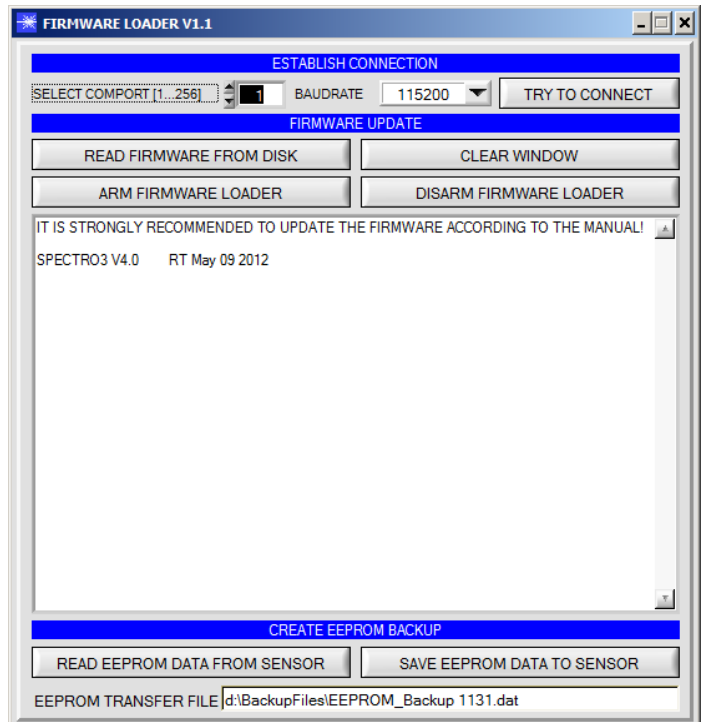
**Step 1:**

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.

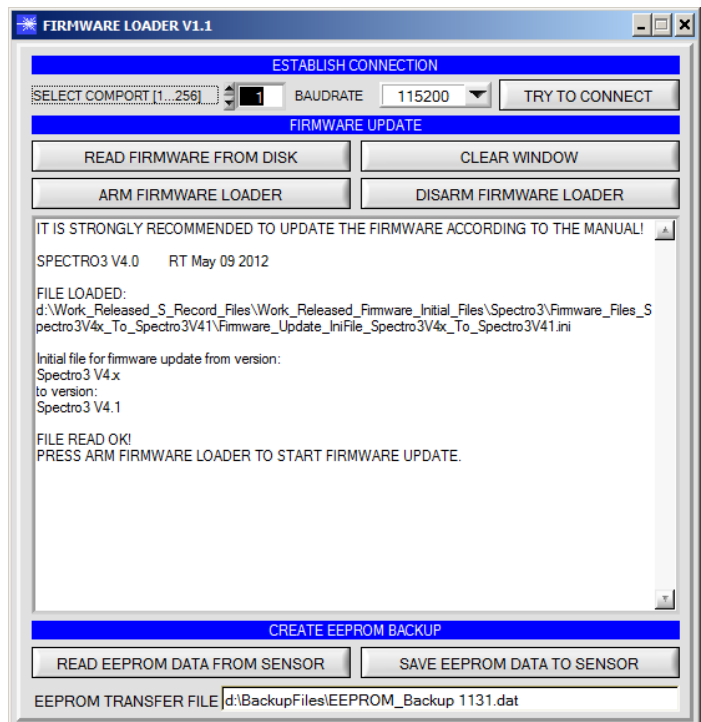
As described above, a plausibility check of the initialisation file will be performed first.

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.

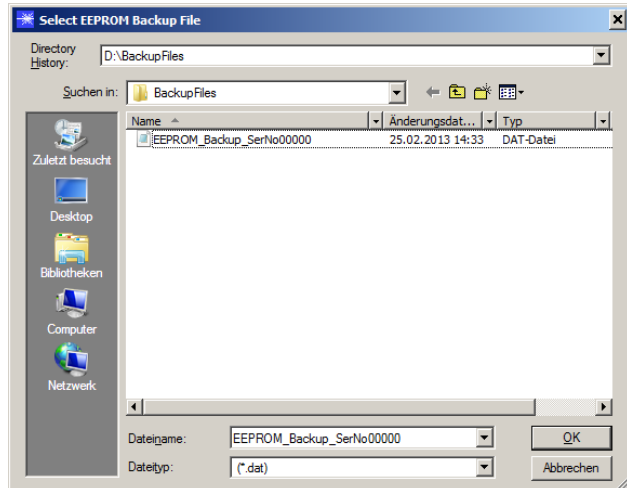
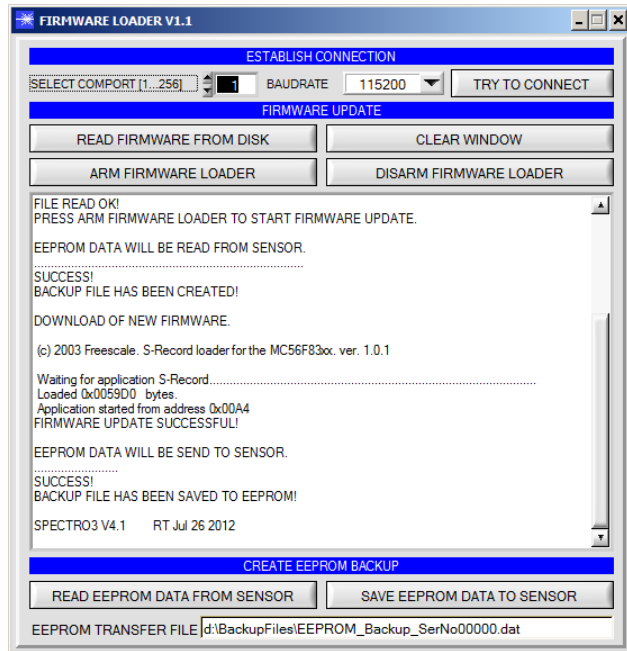
When you press the **ARM FIRMWARE LOADER** button the firmware update will be performed automatically.

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 EEPROM 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 choose an initialization if none has yet been loaded. Afterwards you will be asked to enter a file name. The selected name will be shown in the **EEPROM TRANSFER FILE** display.

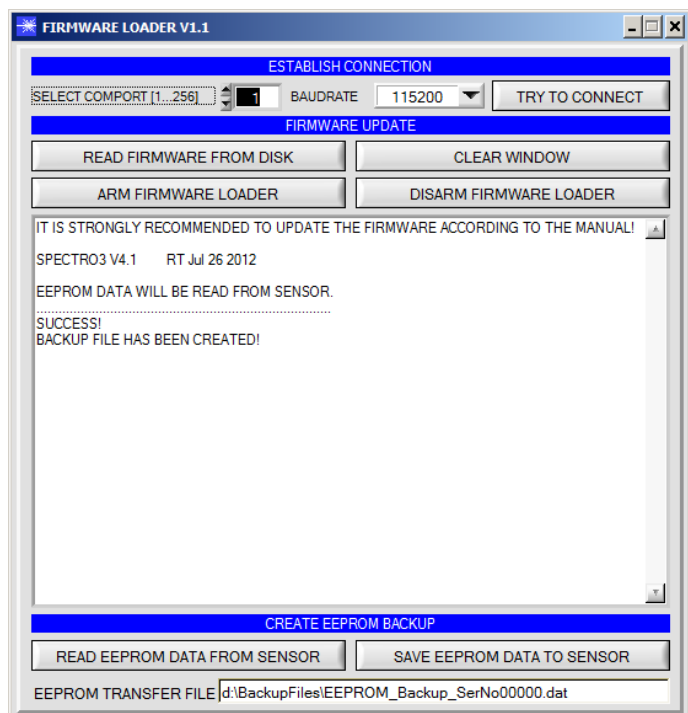
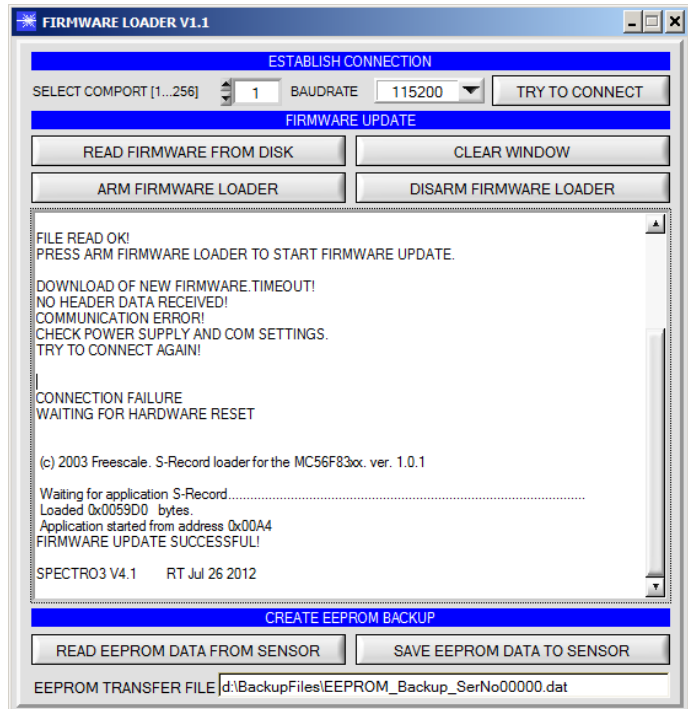
The file name for your **EEPROM backup file** should be chosen such that the names for several sensors cannot be mixed up. Using a file name that contains the sensor serial number might be advisable.

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, and 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.

