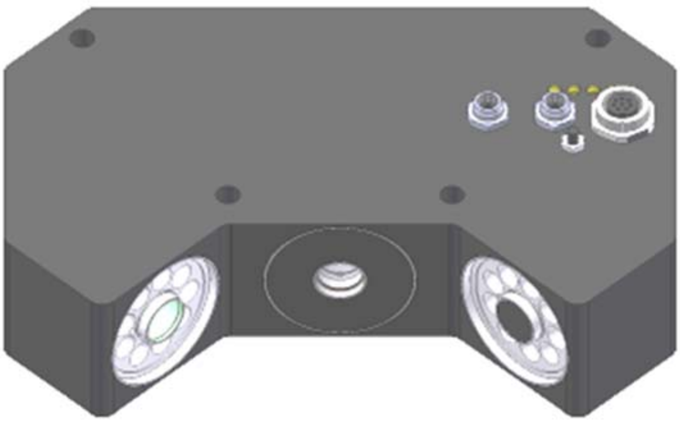
Short Instructions

Software COAST-STRUCT-Scope V1.1

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

Structure part of sensors of COAST (Color and Structure) series



Design versions:

COAST-52-45°

Sensor Instruments GmbH - Schlinding 11 - D-94169 Thurmansbang
Tel.: +49 (0)8544 / 9719-0 - Fax: +49 (0)8544 / 9719-13
E-mail: info@sensorinstruments.de - www.sensorinstruments.de

0 Contents

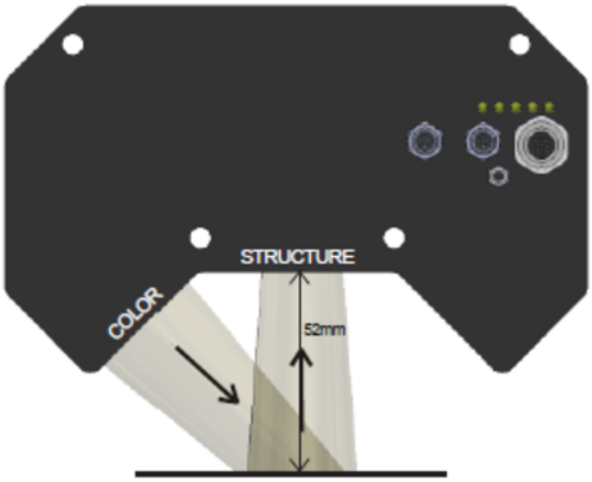
1 Functional principle: COAST color gloss and structure sensor

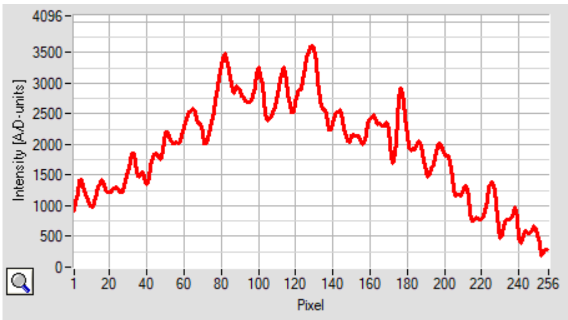
1.1 Technical description

The sensors of the ***COAST** series* (***CO**lor **A**nd **ST**ructure*) are hybrid sensors comprising a color sensor and a structure sensor, both of which are contained in one housing and operate independently of each other. The *COAST sensor* has two LED ring lights that can be operated alternately (forward reflection, backward reflection). The *COAST sensor* is equipped with an 8-pole PLC interface and two RS232 interface (one for the color sensor and the other for the structure sensor). The color sensor evaluates the color and gloss of the object to be inspected, the structure sensor evaluates the surface quality of the object.

<u>Function of the STRUCTURE sensor:</u>

The evaluation unit for structure detection (see. fig. 1) is arranged under an angle of 45° relative to the LED ring lights in the *COAST* sensor housing. By way of an optical aperture system that is positioned at the centre of one ring light the surface quality (structure) of the measuring object is represented on a line sensor. This optical representation of a line-shaped range (approx. 20 mm) of the surface onto the pixels of the line sensor results in an intensity characteristic across the line that is typical of the respective surface quality (structure). By way of fast Fourier transformation (FFT) a frequency spectrum can thus be calculated from the intensity characteristic of the line sensor. From this frequency spectrum the typical features of the surface quality (structure) can be learned by means of suitable evaluation algorithms and can thus be monitored and recognised. The evaluation values calculated in this way are transmitted to the higher level evaluation unit of the color sensor via an internal digital interface. The microcontroller of the *COAST sensor* can be parameterised through the serial RS232 interface by means of a Windows PC software. The software also can be used to set various evaluation modes. The housing of the control unit features a button for switching over the ring light. This function also can be realised through the digital input (IN0) of the 8-pole PLC connection socket.





Intensity characteristic across pixel position on line detector

Fig.1: Functional principle of the COAST sensor

2 Installation of the *COAST-STRUCT-Scope* software

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

- 1GHz Pentium-compatible processor or better
- CD-ROM or DVD-ROM drive
- Approx. 400 MByte of free hard disk space
- SVGA graphics card with at least 800x600 pixel resolution and 256 colors or better.
- Windows® 7, 8 or Windows® 10 operating system
- Free serial RS232 interface or USB port with USB-RS/232 adaptor at the PC

Please install the *COAST-STRUCT-Scope* software as described below:

1.

Insert the installation CD-ROM in your CD-ROM drive. In our example we suppose that this is drive "D".

2.

Start the Windows Explorer, and in the folder tree of your CD-ROM drive go to the installation folder D:\ … \ SOFTWARE\ ... .
Then start the installation program by double-clicking on the SETUP.EXE symbol.

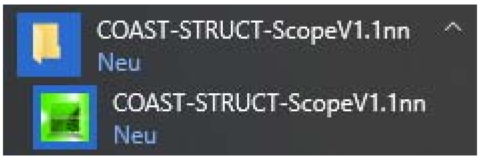
As an alternative, software installation also can be started by clicking on **START-Run**... and then entering "D:\ … \ SOFTWARE \ …\ setup.exe", which must be confirmed by pressing the **OK** button.

3.

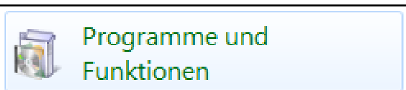
During the installation process a new program group for the software is created in the Windows Program Manager. In this program group an icon for starting the software is created automatically. When installation is successfully completed the installation program displays a "Setup OK" message.

4.

The *COAST-STRUCT-Scope* software can now be started with a mouse-click on the respective icon in the newly created program group under:
Start >All Programs > *COAST-STRUCT-ScopeV1.1*



Uninstalling the *COAST-STRUCT-Scope* software:



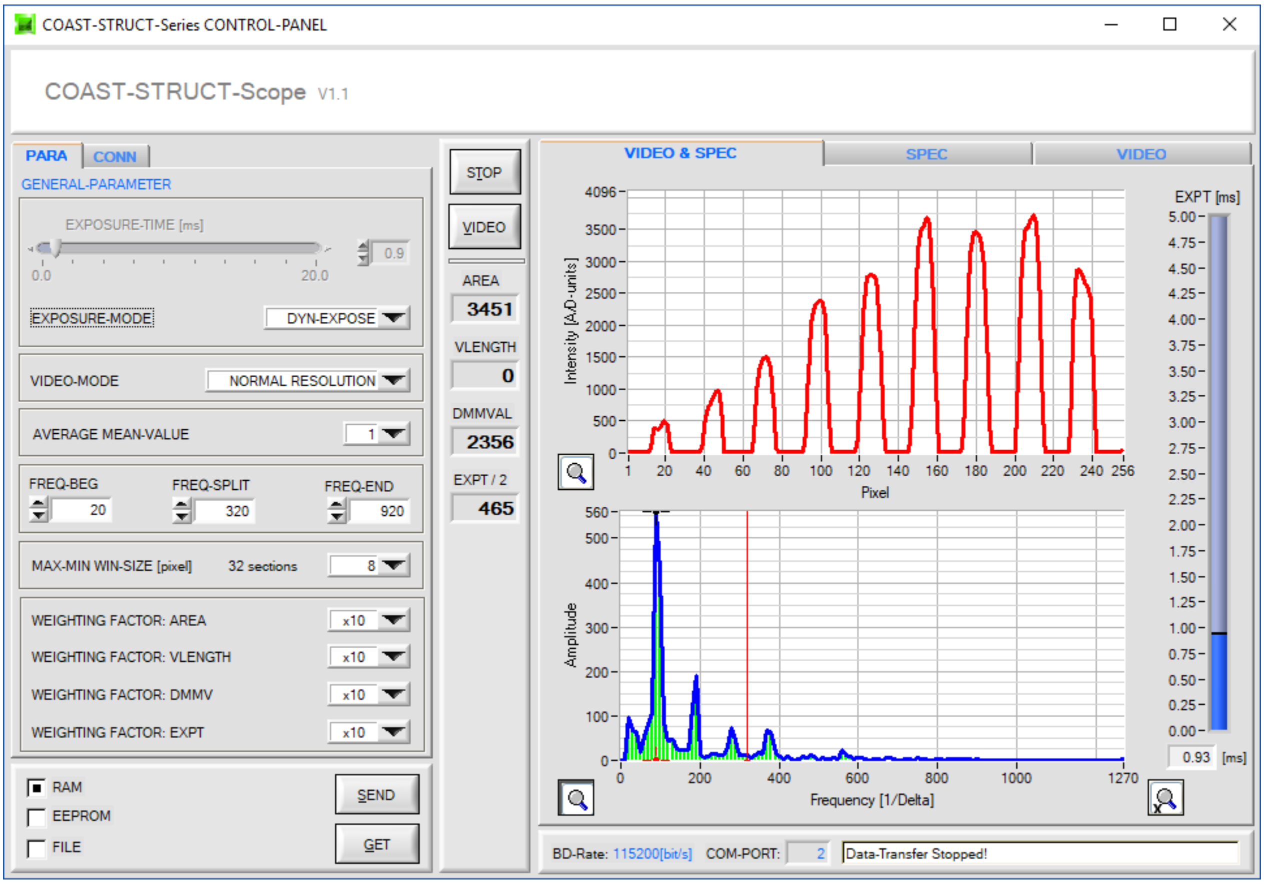
Please use the Windows® uninstall tool to remove the software.
The Windows® uninstall tool can be found under

Start / Settings / Control Panel.

3 Operation of the *COAST-STRUCT-Scope* software

The *COAST-STRUCT-Scope* software is used to parameterise the control unit for the control/evaluation of the *COAST (STRUCT) sensor*. The PC software visualises the measurement values that are provided by the sensor. It can therefore, among others, be used to select a suitable evaluation algorithm and to set tolerance limits for the inspection of the measurement object.

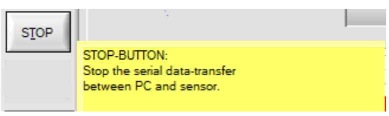
Data exchange between the PC user interface and the sensor system is effected through a standard RS232 interface. For this purpose the sensor is connected to the PC with the serial interface cable cab-las-4/PC or with the USB cable cab-4/USB. When parameterisation is finished, the setting values can be permanently saved in an EEPROM memory of the *COAST (STRUCT) control unit*. The *COAST (STRUCT) sensor* then continues to operate in "STAND-ALONE" mode without the PC.
When the *COAST-STRUCT-Scope* software is started, the following Windows® user interface will be displayed:



The **COAST-STRUCT-Scope CONTROL-PANEL** provides a large variety of functions:

- Visualisation of measurement data in numeric and graphic output fields.
- Setting of the light source/exposure-time.
- Selection of a suitable evaluation mode.
- Saving of parameters in the RAM / EEPROM memory of the control unit or in a configuration file on the PC's hard disk.

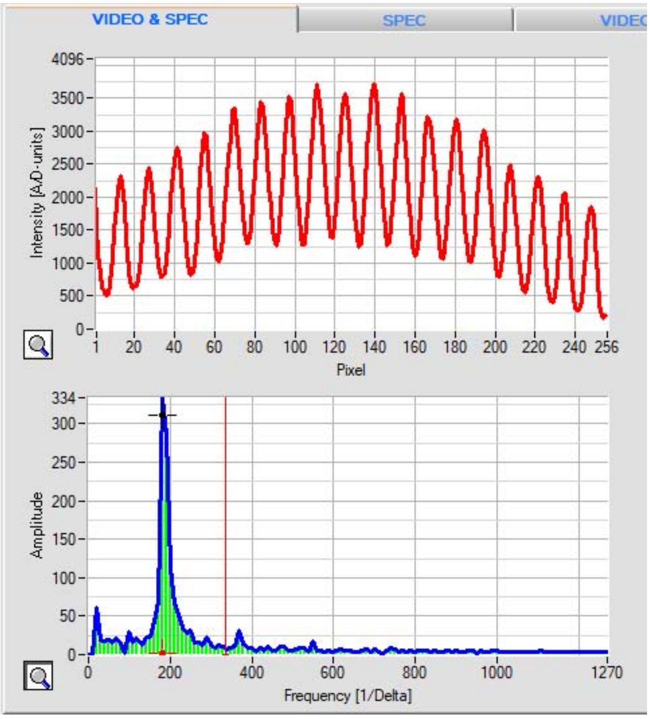
The following chapters provide explanations of the individual control elements of the *COAST-STRUCT-Scope* software.



A click with the right mouse button on a control element displays a short help text on the respective element.

3.1 General control elements of the *COAST-STRUCT-Scope* software

VIDEO button:
After a click on the VIDEO button the intensity profile (video image) that is measured at the CMOS line receiver and the frequency spectrum that is calculated from it are transferred to the PC.



When the **<VIDEO&SPEC>** tab is selected, both the intensity profile (red curve) and the frequency spectrum calculated from it (blue curve) are displayed.

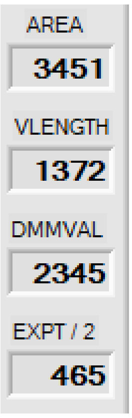
Intensity characteristic at the line sensor:
Y-axis: Amplitude at the respective pixel
X-axis: Pixel of the line sensor

The picture on the left shows the typical representation of a strip-shaped structure with periodically repeated, different reflectance (bright/dark transitions).

Frequency spectrum:
Y-axis: Amplitude of the respective frequency
X-axis: Frequency

The frequency spectrum is calculated from the video image (intensity characteristic) by way of a FFT algorithm. The frequency spectrum displays the frequency distribution of the frequency components contained in the video image.

Numeric display elements:
Various numeric evaluation values are derived from the frequency spectrum or from the video image:

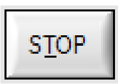


AREA:
Normed area ratio calculated from the frequency spectrum.

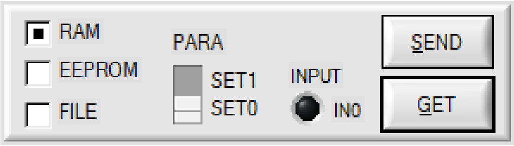
VLENGTH:
Video length, the path integral which is calculated from the video image.

DMMVAL:
Delta-max-min-value, calculated from the video image. In this case, the difference of the maxima / minima values over several, adjustable pixel ranges is calculated.

EXPT / 2:
Exposure-time / 2, this value is transferred to the color sensor.



STOP button:
A click on the STOP button stops the data transfer from the *COAST (STRUCT) sensor* to the PC through the serial interface.



PARAMETER TRANSFER:

This group of buttons is used for the transfer of parameters between the PC and the *COAST (STRUCT) control unit* through the serial RS232 interface.

SEND:

When the SEND button is clicked, the parameters currently set in the user interface are transferred to the *COAST (STRUCT) control unit*.

The target of the respective parameter transfer is determined by the selected radio button (RAM, EEPROM, or FILE).

GET:

When the GET button is clicked, the setting parameters are transferred from the *COAST (STRUCT) control unit* to the PC and are refreshed in the user interface. The source of parameter transfer again is determined by the selected radio button:

RAM:

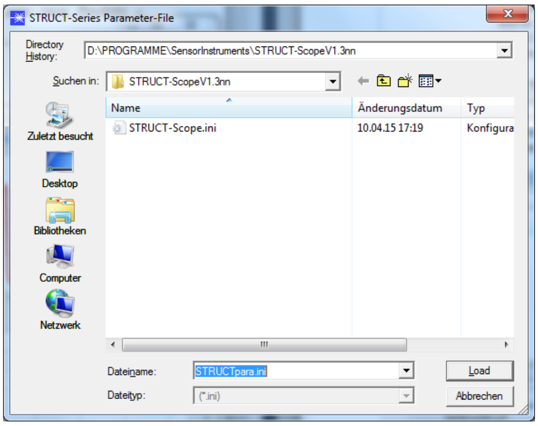
The currently set parameters are written to the volatile RAM memory of the *COAST (STRUCT) control unit*, or they are read from there and transferred to the PC.

EEPROM:

The currently set parameters are written to the non-volatile EEPROM memory of the *COAST (STRUCT) control unit*, or they are read from there and transferred to the PC. Parameters that have been saved in the EEPROM will not be lost when the sensor's power supply is turned off. When parameters are loaded from the EEPROM of the *COAST (STRUCT) control unit*, they must then be written to the RAM of the *COAST* (STRUCT) *control unit* by selecting the RAM button and pressing the SEND button. The *COAST (STRUCT) control unit* then continues to operate with the set RAM parameters.

FILE:

When the FILE radio button is selected, a click on the SEND/GET button will open a new file dialog on the user interface.

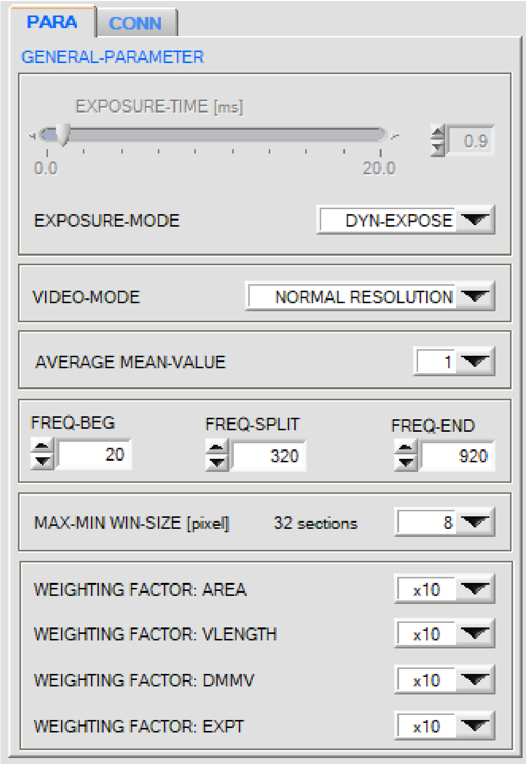


FILE dialog:

The current parameters can be written to a selectable file on the PC's hard disk, or they can be read from such a file. The file name of the standard output file for parameter values is "STRUCTpara.ini".

The output file for example can be opened with the standard Windows text editor program "EDITOR".

3.2 PARAMETER tab

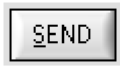


PARA TAB:

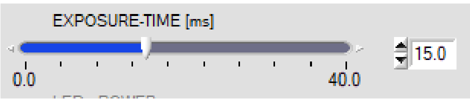
A click on the PARA tab opens the GENERAL-PARAMETER window.

In this window various general parameters can be set at the control unit.

Please note !



Any changes that are made in the function fields described below only become active at the control unit of the *COAST (STRUCT) sensor* when the SEND button is pressed!



EXPOSURE-TIME [ms]:

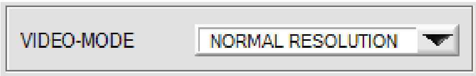
In this function field, the exposure time can be set on the receiving unit of the *COAST (STRUCT) sensor*. This is only possible in EXPOSURE-MODE = STATIC. In EXPOSURE-MODE = DYN-EXPOSE mode, this field is grayed out and the exposure time is automatically adjusted to the respective brightness by the structure-sensor side.

EXPOSURE-MODE:

List box for setting the operating mode of the receiver unit of the COAST (STRUCT) sensor.

STATIC:
Fixed shutter speed with the time specified in the EXPOSURE-TIME [ms] slider.

DYNAMIC:
Automatic adjustment of the exposure time via the amplitude of the received video signal (default setting).

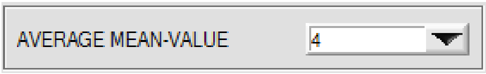


VIDEO-MODE:

List box for specifying the resolution of the linear sensor array receiver.

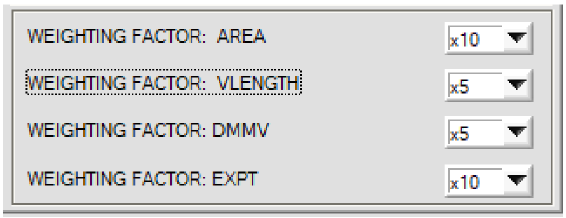
NORMAL RESOLUTION:
Every second pixel on the line sensor is evaluated, the full measuring range (typically 20mm) is available.

HIGH RESOLUTION:
Each pixel on the line sensor is evaluated, only half the measuring range (typically 10mm) is available.List box for setting the resolution of the linear sensor array receiver.



AVERAGE MEAN-VALUE:

List box to specify averaging on the *COAST (STRUCT) control electronics.*
Possible values: N = 1, 2,4,8,16,32 or 64.
For measuring value aquisition, a number N which can be set here is specified. After each video image, the calculated values are put into the respective ring buffer of size N. With each main program run, the mean value from the ring buffer is used for further calculation.



WEIGHTING FACTOR:

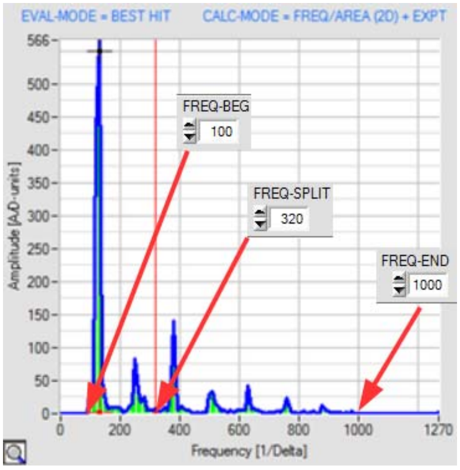
Selection fields for specifying weighting factors for the evaluation quantities *AREA*, *VLENGTH*, *DELTA-MAX-MIN* and *EXPOSURE-TIME*.

With the aid of the weighting factors (*WF*), the evaluation quantities can be adapted before being transferred to the COLOR sensor.

The evaluation values are multiplied by the weighting factor.

$$MeasVal\_new = WF * \{MeasVal\}$$

The weighting factors *WF* can be preset in the range of ***x1*** to ***x10***. The default is ***x10***.



FREQ-BEG, FREQ-SPLIT, FREQ-END:

These numeric input fields are used to set evaluation limits in the frequency spectrum.

FREQ-BEG:
Evaluation beginning in the frequency spectrum.

FREQ-END:
Evaluation end in the frequency spectrum.

FREQ-SPLIT:
This is the dividing line that is used for the calculation of the normed area ratio. It is displayed as a red vertical auxiliary line in the frequency spectrum.

The following evaluation variable is derived from the frequency spectrum:

AREA1/(AREA1+AREA2):
A normed area ratio is used for calculation. The area ratio is calculated from the frequency spectrum. The FREQ-SPLIT parameter is used as a dividing line for ratio calculation. All frequencies lower than FREQ-SPLIT are assigned to AREA1, all frequencies higher than FREQ-SPLIT are assigned to AREA2. It is then checked whether the area ratio determined this way lies within a set tolerance TOL.

$$AREA = 4095 * \{\frac{AREA1}{AREA1 + AREA2}\}$$

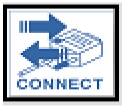
3.3 CONNECTION tab

RS232 COMMUNICATION:

- Standard RS232 serial interface without hardware-handshake.
- 3-line connection: GND, TXD, RXD.
- Speed: Can be set from 9600 Baud to 115200 Baud, 8 data bits, no parity bit, 1 stop bit in binary mode, MSB first.

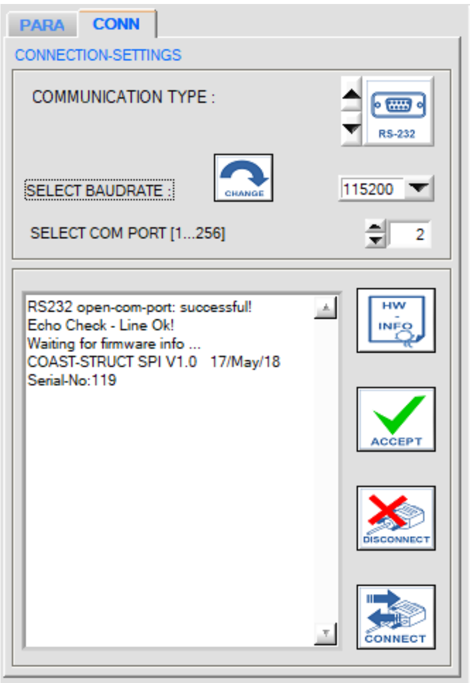
Attention !

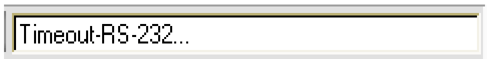
The stable function of the RS232 interface (status message after program start) is a basic prerequisite for a successful parameter transfer between PC and *COAST (STRUCT) control unit*. Due to the low data transfer rate of the serial interface only slow changes of the analog values can be observed in the graphic display at the PC. In order to guarantee the maximum switching frequency of the *COAST (STRUCT) control unit* it is therefore necessary to stop the data exchange during the normal monitoring process (click on the STOP button).



CONNECT:

When the software is started it attempts to establish a connection to the *COAST (STRUCT) senso* through the COM interface that was last used. If connection could be established successfully the current firmware version and the number of the COM port are displayed in the status line.

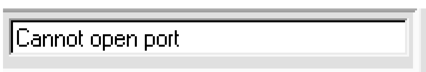




The serial connection between PC and *COAST (STRUCT) control unit* could not be established, or the connection is faulty.

In this case it should first be checked whether the *COAST (STRUCT) control unit* is connected to the power supply, and whether the serial interface cable is correctly connected to PC and control unit.

If there is an "Invalid port number" status message, the selected interface, e.g. COM PORT 2, is not available at your PC.



If there is a "Cannot open port" status message, the selected interface, e.g. COM PORT 2, may already be used by another device.

DISCONNECT:

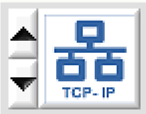
The connection to the sensor hardware is disconnected. The previously opened communication port is released again.

ACCEPT SETTINGS:

With a click on the ACCEPT SETTINGS button the current setting values of the *STRUCT-Scope* PC software are saved in the TB-Scope.ini file. The popup window will then be closed. When the *STRUCT-Scope* software is restarted, the parameters saved in the INI file will be loaded.

GEN. HW INFO FILE:

A click on this button generates a file in which all the important sensor data are stored in encrypted form. This file can be sent to the sensor manufacturer for diagnostic purposes.



COMMUNICATION TYPE:

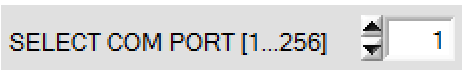
The type of data communication can be set in this function field:

RS232:
Data communication through the standard RS232 interface.

TCP/IP:
Data communication through a RS232-TCP/IP Ethernet converter module.

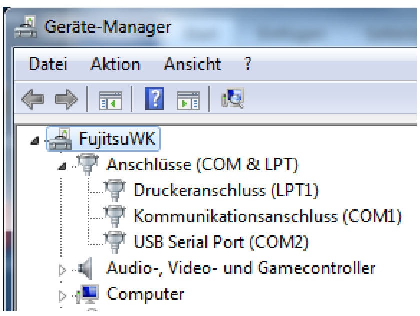
SELECT BAUDRATE:

The baud rate of the serial interface can be set in this function field:
Possible values: 9600Baud, 19200Baud, 38400Baud, 57600Baud or 115200Baud (setting when delivered = 115200Baud).

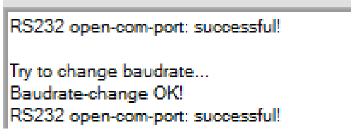


SELECT COM PORT [1…256]:

The number of the communication port can be set in this function field.
Possible values are COM PORT 1 to 255.
The communication port number can be found in the Windows® operating system under
START/Control Panel/Device Manager.



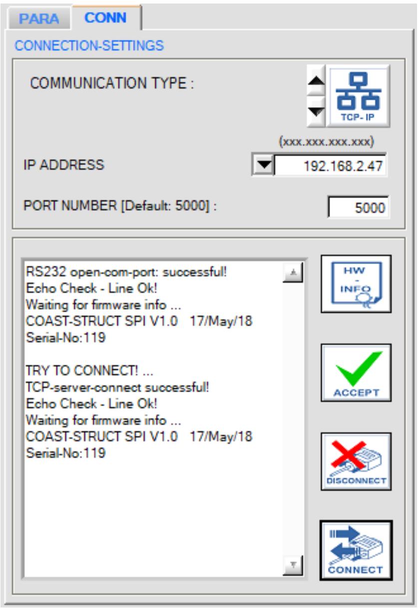
Note!



CHANGE BAUDRATE:

With a click on this button the baud rate of the serial interface at the sensor hardware is changed to the value selected in the SELECT-BAUDRATE list field. A corresponding status message will be displayed when the change of the baud rate at the sensor was successful.
The baud rate change only is performed in the volatile RAM memory of the *COAST (STRUCT) sensor*. If the baud rate should be changed permanently, the new baud rate value must be saved to the EEPROM by clicking on the SEND + EE button!

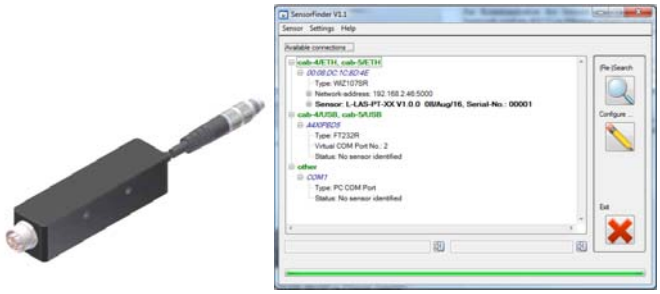
3.4 Data transfer through the external RS232 to Ethernet adapter



An RS232 to Ethernet adapter (***cab-4/ETH-500***) is needed if the sensor should communicate through a local network. With this adapter a connection to the sensor can be established using the **TCP/IP** protocol.

The network adapter converts the standard RS232 signals of the sensor and provides an interface for a LAN network. The RS232 interface can be operated with a Baudrate of 11200Baud.

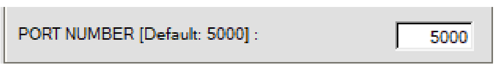
A software (*SensorFinder*) that is supplied with the adapter can be used to find the adapter in the network – and to then configure it:



cab-4/ETH-500 RS232/Ethernet converter and *SensorFinder V1.1 software*.

IP ADDRESS:

Input mask for entering the IP address.



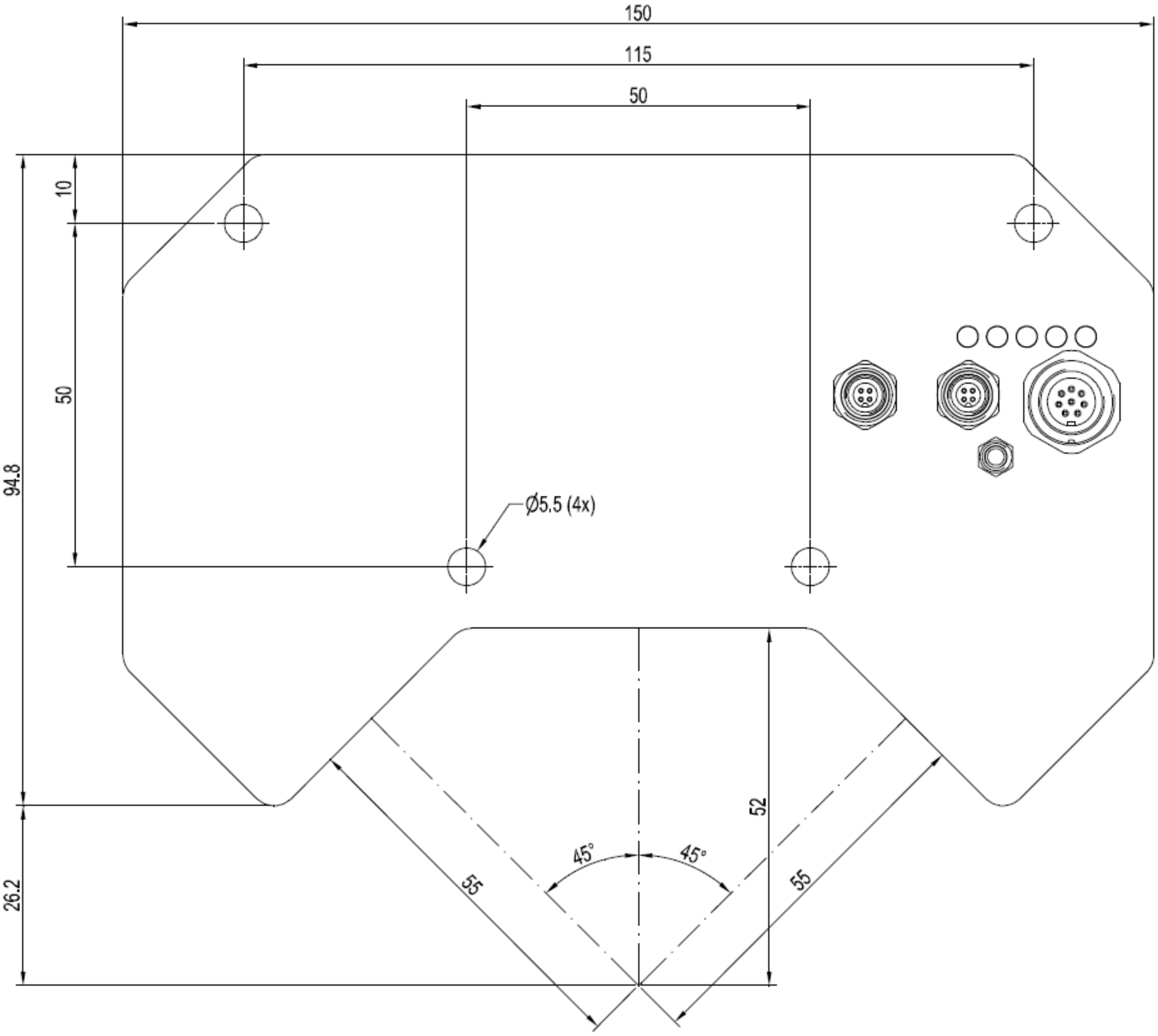
PORT NUMBER:

The **PORT NUMBER** for the network adapter based on Lantronix-XPort is set to 5000. This value must not be changed.

4 Annex

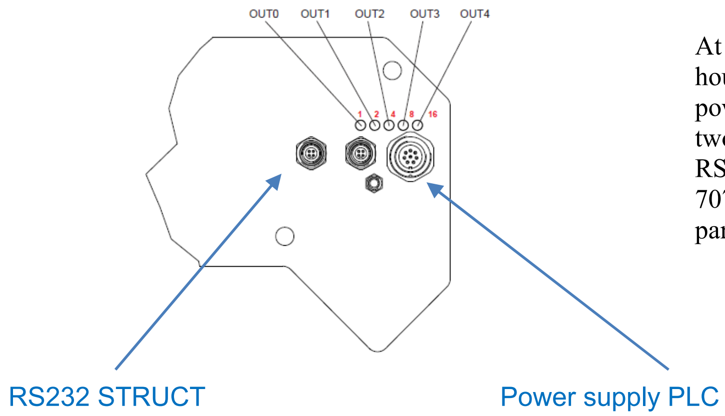
4.1 Dimensions / adjustment

All data in mm



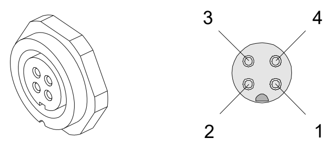
Sensortyp: COAST-52-45°

4.2 Connection sockets



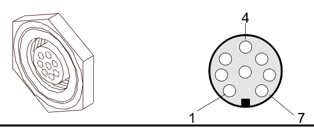
At the side of the *COAST (STRUCT) sensor* housing there is a socket for connecting the power supply (8-pole M12 type Binder 712) and two more sockets for connecting the serial RS232 connection cable (4-pole type M5 Binder 707) to the structure part as well as to the color part of the sensor.

Connection to PC (RS232):

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

Connection cables:
cab-las4/PC-…
cab-4/USB-…
cab-4/ETH-…

Connection COAST to PLC (power supply):

8-pole socket (type Binder 712) ***COAST / PLC***		
Pin No.:	**Color of wire: (cab-las8/SPS)**	**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%)

Connection cable:
cab-las8/SPS-…

4.3 RS232 interface protocol

- Standard RS232 serial interface, no hardware handshake
- 3-wire-connection: GND, TXD, RXD
- Speed: 9600 baud, 19200 baud, 38400 baud, 57600 baud or 115200 baud
- 8 data-bits
- NO parity-bit
- 1 STOP-bit
- binary-mode.

METHOD:

The sensor control unit always behaves passively. Data exchange therefore is initiated by the PC (or PLC). The PC sends a data package ("frame") either with or without appended data, to which the sensor control unit responds with a frame that matches the request. The data package comprises a **HEADER** and the optional **DATA**.

HEADER

1. Byte : Synchronisation byte <SYNC> (85dez = 0x55hex)
2. Byte: Order byte <ORDER>
3. Byte : Argument <ARG LO>
4. Byte : Argument <ARG HI>
5. Byte : Data length <LEN LO>
6. Byte : Data length <LEN HI>
7. Byte : Checksum Header <CRC8 HEAD>
8. Byte : Checksum Data <CRC8 DATA>

The first byte is a synchronisation byte and always is 85_{dez} (55_{hex}). The second byte is the so-called order byte <ORDER>, it determines the action that should be performed (send data, save data, etc.).
A 16-bit value <ARG> 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 <LEN>. This value states the number of appended data bytes. Without appended data <LEN=0>, the maximum data length is 512 bytes <LEN=512>. The seventh byte is formed with the CRC8 checksum over all data bytes.
The eight byte is the CRC8 checksum for the header and is formed from bytes 0 up to and incl. 7.
The header always has a total length of 8 bytes. The complete frame may contain between 8 and 520 bytes.

Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header	Byte8 Header	Byte9 Data	Byte10 Data	...	Byte n+7 Data	Byte n+8 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 commands (orders) can be processed by the structure sensor:

<ORDER>	Meaning of the 2.nd byte <order>:	ORDER-TABLE
0	NOP	no operation
1	Send parameter from PC to RAM of the STRUCT-sensor.	PC ⇒ STRUCT-sensor RAM
2	Get parameter from STRUCT-sensor RAM	STRUCT-sensor-RAM ⇒ PC
3	Send parameter from PC to EEPROM	PC ⇒ STRUCT-sensor EEPROM
4	Get parameter from EEPROM of STRUCT-sensor	STRUCT-sensor EEPROM ⇒ PC
5	Echo check: Get echo of STRUCT-sensor	first word=0x00AA=170dec
7	Get software version info of STRUCT-sensor	STRUCT-sensor ⇒ PC
8	**Get measured values from** STRUCT-sensor RAM	**STRUCT-sensor ⇒ PC**
9	Get video/spectra-buffer from STRUCT-sensor	STRUCT-sensor ⇒ PC
190	Change RS232-baud-rate STRUCT-sensor RAM	PC ⇒ STRUCT-sensor RAM

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

Attention:

For the HEADER bytes as well as for the DATA bytes an own checksum has to be calculated!

4.3.1 Parameter set format

The sensors of the COAST series are working on the side of the STRUCT sensor system with the following 16 STRUCT parameters, which have to be transmitted in the specified order in the data attachment to the STRUCT sensor:

DATA-FRAME: <parameter-set>		
Para	**Meaning**	**Comment**
0	POWER	Laser intensity (0 … 1000) not used!
1	INTEGRATION-TIME	Integration time 0.1ms … 40ms (= 100 … 40000)
2	POWER-MODE	Power mode: (0 = STATIC), (1 = DYN-EXPOSE-TIME)
3	VIDEO-MODE	Video readout-mode: (0:= NORMAL-RES), (1:=HIGH-RES)
4	AVERAGE	Average setting (1,2,4,8,16,32 or 64) size of ring buffer
5	DMM-WINDOW	Delta-Max-Min window size (pixel) (4,8,16,32 or 64)
6	FFT-BEG	Evaluation end -pixel (E_BEG+1 … SUBPIXEL)
7	FFT-END	Teach-value (1 … SUBPIXEL)
8	FFT-SPLIT	Upper-tolerance (0 ... SUBPIXEL/2)
9	RS232-MODE	RS232 mode: (0=STAT,1=IN0-L/H,2=IN0-HI[6-byte],3=CONT[6-byte]
10	RS232-BAUDRATE	Baudrate: (0=9600,1=19200,2=38400,3=57600,4=115200) baud
11	WF-AREA	Weighting factor area-value: 1,2,3,4,…,10 (default=10)
12	WF-VECT-LENGTH	Weighting factor vector-length: 1,2,3,4,…,10 (default=10)
13	WF-DELTA-MAX-MIN	Weighting factor delta-max-min: 1,2,3,4,…,10 (default=10)
14	WF-EXPOSE-TIME	Weighting factor exposure-time: 1,2,3,4,…,10 (default=10)
15	PARA15	Free use (default: 0)

4.3.2 RS232 Data transfer examples

< ORDER = 5 > : ECHO-CHECK, READ LINE OK from sensor.

DATA FRAME PC → Sensor (8 Bytes)

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	**5**	0	0	0	0	170	60
		ARG=0		LEN=0			

DATA FRAME Sensor → PC (8 Bytes)

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	**5**	170	0	0	0	170	178
		ARG=170		LEN=0			

Serial number of sensor = <ARG> value

< ORDER = 7 > : Read FIRMWARE-VERSION STRING from 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)	7	0	0	0	0	170	82
		ARG=0		LEN=0			

DATA FRAME Sensor → PC (8 + 72) Bytes

Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header	Byte8 Header	Byte9 Data	Byte10 Data	Byte11 Data	Byte12 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	1	2	72	0	XXX	82	C	O	A	S
		ARG=513 (Ser.-No)		LEN=72							

Byte13 Data	Byte14 Data	Byte15 Data	Byte16 Data	Byte17 Data	Byte18 Data	Byte19 Data	Byte20 Data	Byte21 Data	Byte22 Data	Byte23 Data	Byte24 Data
ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII
T	-	S	T	U	C	T		S	P	I	

Byte25 Data	Byte26 Data	Byte27 Data	Byte28 Data	Byte29 Data	Byte30 Data	Byte31 Data	Byte32 Data	Byte33 Data	Byte34 Data	Byte35 Data	Byte36 Data
ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII
V	1	.	0	.	0		1	7	/	M	A

Byte37 Data	Byte38 Data	Byte39 Data	Byte40 Data	Byte41 Data	Byte42 Data	Byte43 Data	Byte44 Data	Byte45 Data	Byte46 Data	Byte47 Data	Byte48 Data
ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII
Y	/	1	8								

Byte49 Data	Byte50 Data	Byte51 Data	Byte52 Data	Byte53 Data	Byte54 Data	Byte55 Data	Byte56 Data	Byte57 Data	Byte58 Data	Byte59 Data	Byte60 Data
ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII

Byte61 Data	Byte62 Data	Byte63 Data	Byte64 Data	Byte65 Data	Byte66 Data	Byte67 Data	Byte68 Data	Byte69 Data	Byte70 Data	Byte71 Data	Byte72 Data
ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII	ASCII

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

< ORDER = 1 > : SEND PARAMETER-SET TO RAM of the sensor

DATA FRAME PC → Sensor (8 + 32) Bytes
Attention: The complete parameter-set (32 Bytes) must be attached and sent to the header!

Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header	Byte8 Header	Byte9 Data	Byte10 Data	Byte11 Data	Byte12 Data
0x55	<order>	<ARG> (lo byte)	<ARG> (hi byte)	<LEN> (lo byte)	<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)	Para1 (lo byte)	Para1 (hi byte)	Para2 (lo byte)	Para2 (hi byte)
85 (dec)	**1**	0	0	42	0	xxx	203	0	0	144	1
		ARG=0		LEN=32				POWER=0		INT-TIME=400	

Byte13 Data	Byte14 Data	Byte15 Data	Byte16 Data	Byte17 Data	Byte18 Data	Byte19 Data	Byte20 Data	Byte21 Data	Byte22 Data	Byte23 Data	Byte24 Data
Para3	Para3	Para4	Para4	Para5	Para5	Para6	Para6	Para7	Para7	Para8	Para8
2	0	0	0	1	0	8	0	20	0	246	4
P-MODE=2		VMODE=0		AVERAGE=1		DMM-WIN=8		FFT-BEG=20		FFT-END=1270	

Byte25 Data	Byte26 Data	Byte27 Data	Byte28 Data	Byte29 Data	Byte30 Data	Byte31 Data	Byte32 Data	Byte33 Data	Byte34 Data	Byte35 Data	Byte36 Data
Para9	Para9	Para10	Para10	Para11	Para11	Para12	Para12	Para13	Para13	Para14	Para14
64	1	0	0	4	0	10	0	10	0	10	0
FFT-SPLIT=320		RS232-MODE=0		RS232-BAUD=4		WF_AREA=10		WF_VLEN=10		WF_DMMV=10	

Byte37 Data	Byte38 Data	Byte39 Data	Byte40 Data
Para27	Para27	Para28	Para28
10	0	0	0
WF_EXPT=10		PARA15=0	

DATA FRAME Sensor → PC (8 Byte)

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	81
		ARG=0		LEN=0			

< ORDER = 2 > : READ PARAMETER-FROM RAM of the sensor

DATA FRAME PC → Sensor (8 Bytes)

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 (8+84) Bytes

Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header	Byte8 Header	Byte9 Data	Byte10 Data	Byte11 Data	Byte12 Data
0x55	<order>	<ARG> (lo byte)	<ARG> (hi byte)	<LEN> (lo byte)	<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)	Para1 (lo byte)	Para1 (hi byte)	Para2 (lo byte)	Para2 (hi byte)
85 (dec)	**2**	0	0	32	0	xxx	185	0	0	144	1
		ARG=0		LEN=32				POWER=0		INT-TIME=400	

The data-block is similar to < ORDER = 1 >:

Byte37 Data	Byte38 Data	Byte39 Data	Byte40 Data
Para39	Para39	Para40	Para40
10	0	0	0
WF_EXPT=10		PARA15=0	

< ORDER = 8 > : READ MEASUREMENT DATA from sensor

DATA FRAME PC → Sensor (8 Bytes)

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)	8	0	0	0	0	170	118
		ARG=0		LEN=0			

DATA FRAME Sensor → PC (8 + 48) Bytes

Byte1 Header	Byte2 Header	Byte3 Header	Byte4 Header	Byte5 Header	Byte6 Header	Byte7 Header	Byte8 Header	Byte9 Data	Byte10 Data	Byte11 Data	Byte12 Data
0x55	<order>	<ARG> (lo byte)	<ARG> (hi byte)	<LEN> (lo byte)	<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)	Raw1 (lo byte)	Raw1 (hi byte)	Raw2 (lo byte)	Raw2 (hi byte)
85 (dec)	8	0	0	60	0	XXX	118	160	0	101	1
		ARG=0		LEN=48				S_FREQ =160		S_AMP = 357	

Byte13 Data	Byte14 Data	Byte15 Data	Byte16 Data	Byte17 Data	Byte18 Data	Byte19 Data	Byte20 Data	Byte21 Data	Byte22 Data	Byte23 Data	Byte24 Data
Raw3	Raw3	Raw4	Raw4	Raw5	Raw5	Raw6	Raw6	Raw7	Raw7	Raw8	Raw8
118	11	100	7	251	10	0	0	246	2	211	13
S_AREA = 2934		V_VLEN = 1892		V-DMMV = 2811		DYNPOW=0		DYNTIME=758		R_STATE=3539	

raw	0x0052C7FA	raw_struct
raw.Sfreq	160	unsigned short
raw.Samp	357	unsigned short
raw.Sarea	2934	unsigned short
raw.Vvlen	1892	unsigned short
raw.Vdmmv	2811	unsigned short
raw.dynpow	0	unsigned short
raw.dyntime	758	unsigned short
raw.rstate	0x0052C808	rstate_struct
raw.rstate.val	3539	unsigned short
raw.rstate.ste	0	unsigned short

< ORDER = 9 > : GET VIDEO-DATA INFORMATION of sensor

ATTENTION: Only 256 pixel of the CMOS line-sensor are transferred!
The <ARG> value determines the source of the VIDEO-DATA-INFORMATION
ARG = 0 : CMOS-VIDEO-RAM-DATA, ARG = 1: SPECTRA-DATA

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)	**9**	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	Byte9 Data	Byte10 Data	Byte11 Data	Byte12 Data
0x55	<order>	<ARG> (lo byte)	<ARG> (hi byte)	<LEN> (lo byte)	<LEN> (hi byte)	CRC8 (Data)	CRC8 (Header)	Para1 (lo byte)	Para1 (hi byte)	Para2 (lo byte)	Para2 (hi byte)
85 (dec)	**9**	0	0	0	1	XXX	185	200	0	220	0
		ARG=0		LEN=256				PIX1=200		PIX2=220	

Byte13 Data	Byte14 Data	Byte15 Data	Byte16 Data	Byte17 Data	Byte18 Data	Byte19 Data	Byte20 Data	Byte21 Data	Byte22 Data	Byte23 Data	Byte24 Data
Para3	Para3	Para4	Para4	Para5	Para5	Para6	Para6	Para7	Para7	Para8	Para8
240	0	0	1	44	1	124	1	0	2	88	2
PIX3=240		PIX4=256		PIX5=300		PIX6=380		PIX7=512		PIX8=600	

Byte25 Data	Byte26 Data	Byte27 Data	Byte28 Data	Byte29 Data	Byte30 Data	Byte31 Data	Byte32 Data	Byte33 Data	Byte34 Data	Byte35 Data	Byte36 Data
Para9	Para9	Para10	Para10	Para11	Para11	Para12	Para12	Para13	Para13	Para14	Para14
168	2	170	2	188	2	188	2	198	2	208	2
PIX9=680		PIX10=682		PIX11=700		PIX12=700		PIX13=710		PIX14=720	

Byte37 Data	Byte38 Data	Byte39 Data	Byte40 Data	Byte41 Data	Byte42 Data	Byte43 Data	Byte44 Data	Byte45 Data	Byte46 Data	Byte47 Data	Byte48 Data
Para15	Para15	Para16	Para16	Para17	Para17	Para18	Para18	Para19	Para19	Para20	Para20
34	3	32	3	32	3	22	3	19	3	20	3
PIX15=802		PIX16=800		PIX17=800		PIX18=790		PIX19=787		PIX20=788	

●
●
●

Byte49 Data	Byte50 Data	Byte51 Data	Byte52 Data	Byte53 Data	Byte54 Data	Byte55 Data	Byte56 Data	Byte57 Data	Byte58 Data	Byte59 Data	Byte60 Data
Para251	Para251	Para252	Para252	Para253	Para253	Para254	Para254	Para255	Para255	Para256	Para256
124	1	44	1	0	1	240	0	220	0	200	0
PIX251=380		PIX252=300		PIX253=256		PIX254=240		PIX255=220		PIX256=200	

< ORDER = 190 > : CHANGE BAUDRATE at sensor (RAM)

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)	**190**	1	0	0	0	170	14
		ARG=1		LEN=0			

New baud rate is set by <ARG> value:
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

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)	190	0	0	0	0	170	195
		ARG=0		LEN=0			