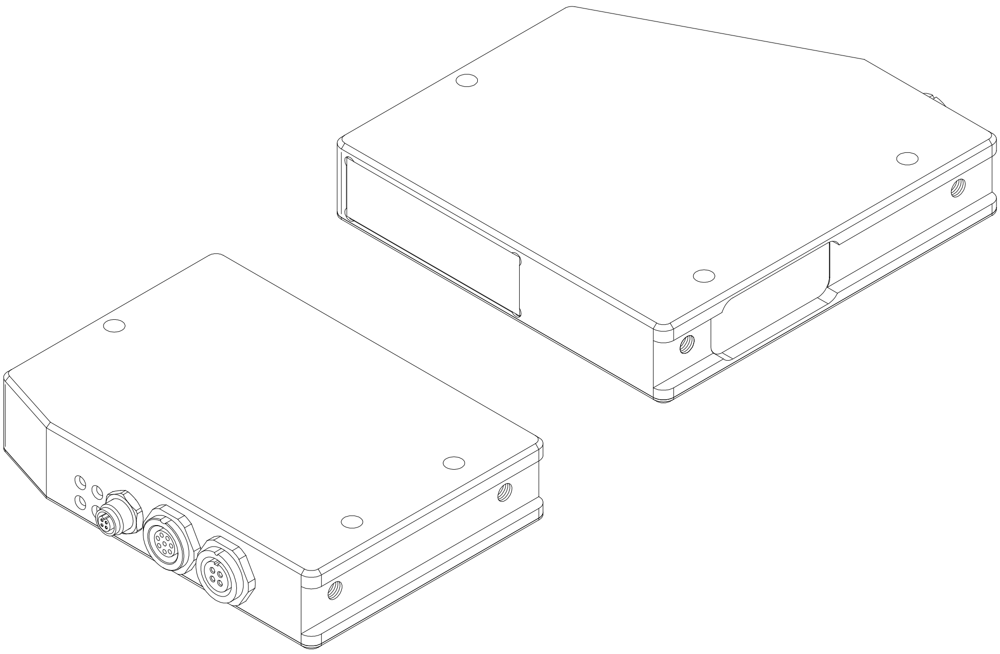
Manual / Start-Up Guide
L-LAS-TB-…-AL-SC
Spray Control System

with

L-LAS-Spray-Control Scope V2.0

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



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

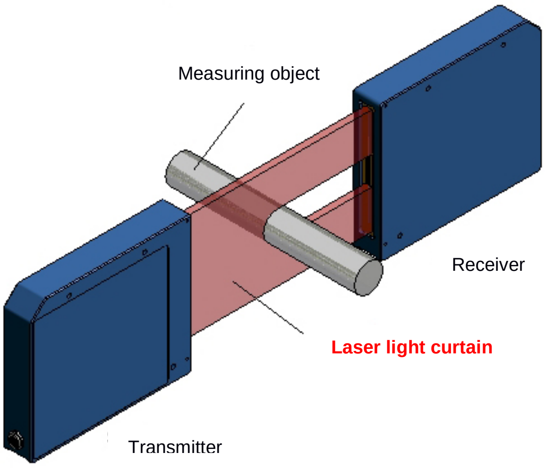
0 Contents

1 Functional principle of the *L-LAS-TB-…* laser line sensor

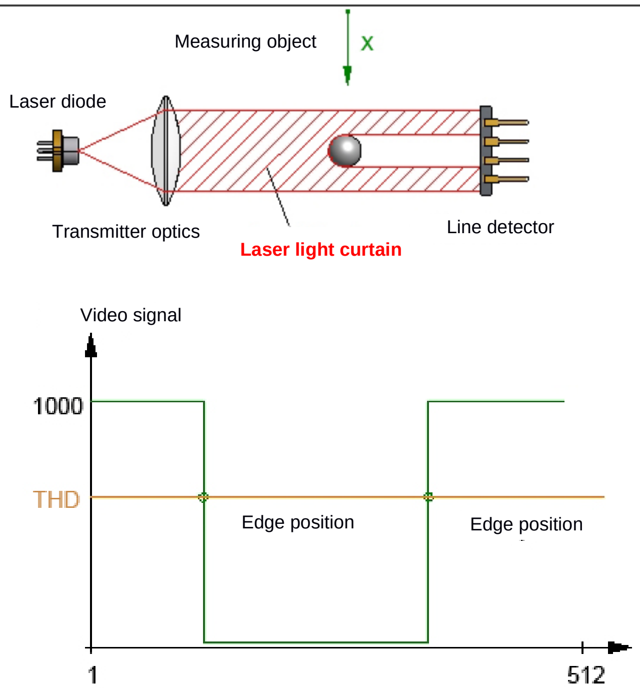
1.1 Technical description

In the laser line sensors of the *L-LAS-TB-... series* the laser beam of a laser diode (»=670nm, 0.4mW opt. power, laser class 1) through suitable collimators and apertures is emitted from the optical transmitter unit as a laser line, i.e. as parallel laser light with homogeneous light distribution. In the optical receiver unit the laser line impinges on a CMOS line receiver. This CMOS line comprises many closely adjacent individual receiver elements (pixels) that are arranged in a line. The light quantity of each of these receiver elements that is collected during the integration time is separately read out as an analog voltage (video signal) and, after analog/digital conversion, is stored in a data field as a digital value.

When there is a non-transparent measuring object in the laser line, the parallel laser light only illuminates those receiver elements (pixels) of the line that lie outside the shadow zone of the measuring object. As a result the pixels within the shadow zone give off a considerably lower analog voltage compared to the illuminated pixels (cf. pic. 1). By way of suitable software algorithms the areas of the shadow zones can be determined from the previously stored data field. Since the distance of the pixels on the line detector is known, the size and position of the measuring object can thus be determined. The micro-controller of the *L-LAS-TB-… sensor* can be parameterized through the serial RS232 interface by means of a Windows PC software. The sensor can be set to operate with various evaluation modes. Switching states are visualized by means of four LEDs (1x green, 1x yellow, and 2x red) that are integrated in the housing of the *L-LAS-TB-… sensor*. The has three digital outputs (OUT0, OUT1, OUT2), the output polarity of which can be set with the software. Two digital inputs (IN0, IN1) make it possible to realize an external WHITE-BALANCE-OFFSET (IN1) functionality and an external TRIGGER (IN0) functionality through a PLC. In addition the control unit features a high-speed analog output (0...+10V or 4…20mA) with 12-bit digital/analog resolution.



Pic.1: Functional principle of the laser line sensor



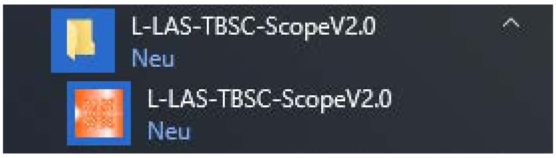
Intensity curve over pixel position at the line detector

2 Installation of the *L-LAS-Spray-Control* software

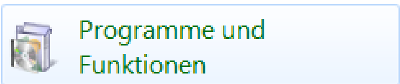
Hardware requirements for successful installation of the *L-LAS-Spray-Control-Scope* software:

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

Please install the *L-LAS-Spray-Control-Scope* software as described below:

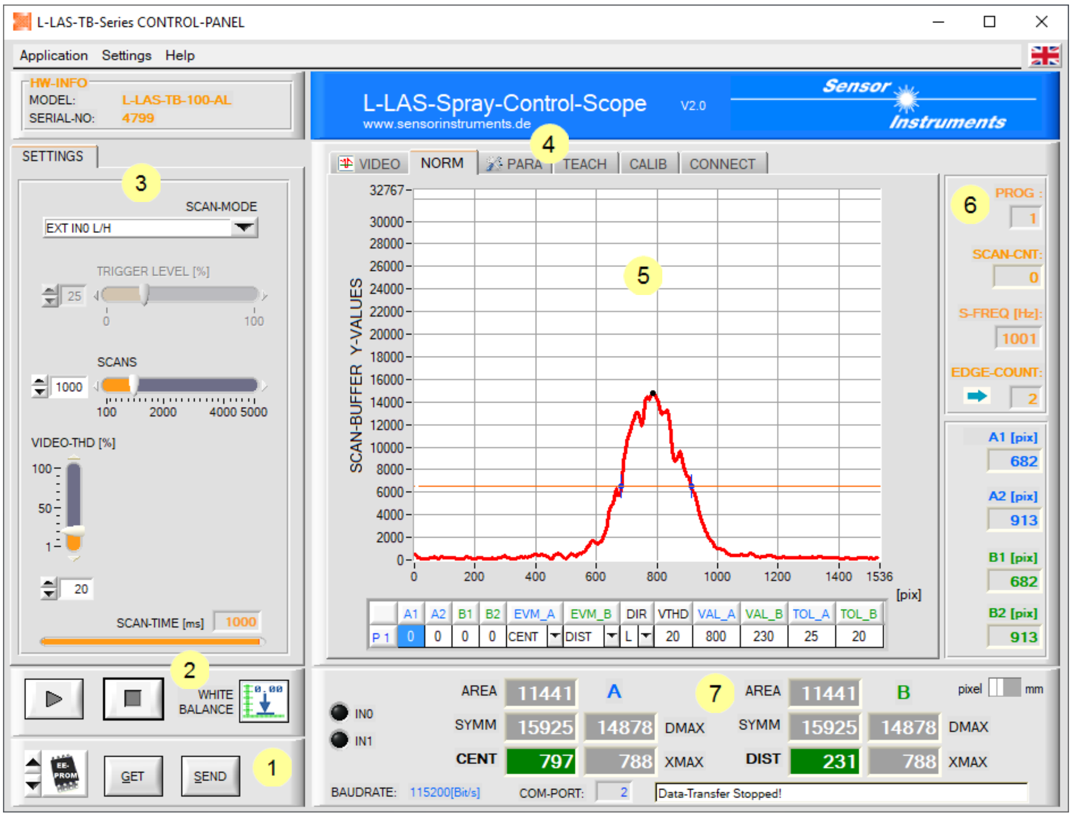
1. 2. 3.	The software can be installed directly from the installation DVD. To install the software, start the SETUP program in the SOFTWARE folder of the DVD. The installation program displays a dialog and suggests to install the software in the C:\"FILENAME" directory on the hard disk. You may accept this suggestion with OK or [ENTER], or you may change the path as desired. Installation is then performed automatically. 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.	The *L-LAS-Spray-Control-Scope* software can now be started by clicking on the respective icon in the newly created program group under: Start >All Programs > *L-LAS-TBSCl-ScopeV2.0* 

<u>Uninstalling the *L-LAS-Spray-Control-Scope* software:</u>

	Please use the Windows® uninstall tool to remove the software. The Windows® uninstall tool can be found under Start / Settings / Control Panel.

3 Control elements of the *L-LAS-Spray-Control-Scope* software

3.1 Short description of the *L-LAS-Spray-Control-Scope* user interface:



The *L-LAS-Spray-Control-Scope* user interface provides a great variety of functions:

- Visualization of measurement data in numeric and graphic output fields.
- Setting of the light source.
- Setting of the polarity of the digital switching outputs OUT0, OUT1, OUT2
- Selection of a suitable evaluation mode.
- Saving of parameters to the RAM, EEPROM memory of the control unit, or to a configuration file on the hard disk of the PC.

1. Function fields for sending / reading the setting parameters (parameter transfer).
2. START / STOP function fields for the RS232 data exchange with the sensor.
3. Presetting of current parameters at the sensor (trigger mode, evaluation threshold...).
4. Tab row to switch between different tab graphic windows.
5. Graphic output (display of the measured value over time, with teach value and tolerance band).
6. Numeric display elements (measuring frequency, number of edges, program number, ...).
7. Measured value display in [mm] or [pixel].

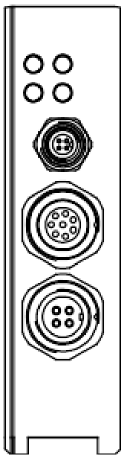
The following chapters provide explanations of the individual control elements of the *L-LAS-Spray-Control-Scope* software. Pressing the right mouse button on an individual element will call up a short help text.

3.2 Procedure for start-up:

A) Adjusting transmitter and receiver:

The L-LAS transmitter and receiver should be optimally aligned to each other, which should be done by means of a traverse or by way of fixed mounting of transmitter and receiver.

B) Establishing power supply / PLC connection:



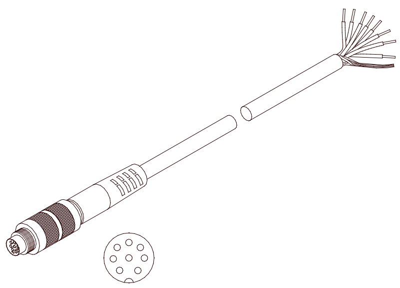
There are three connecting sockets at the housing of the *L-LAS-TB-…-AL-SC control unit (receiver unit).*

A 4-pole M5 socket type Binder 707 is used to connect the serial RS232 interface.

An 8-pole M9 socket type Binder 712 is used to connect the sensor with the PLC / power supply.

A 4-pole M9 socket type Binder 712 is used to connect the *L-LAS-TB-…-AL-SC transmitter unit*.

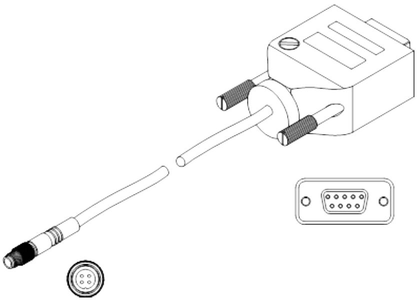
8-pole female connector type Binder 712
Connecting cable: cab-las8/SPS (standard length 2m, cable sheath: PUR)



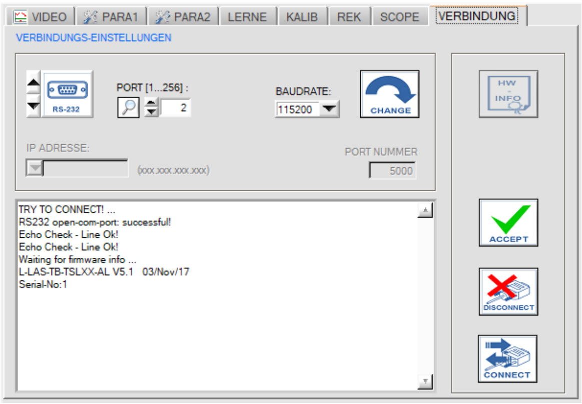
Pin	Color	Assignment L-LAS-TB-…-AL-SC (control unit)
1	White	0V (GND)
2	Brown	+24 VDC ± 10%
3	Green	IN0 (EXT TRIGGER)
4	Yellow	IN1 (WHT BALANCE)
5	Grey	OUT0 (BUSY)
6	Pink	OUT1 (ERR EVAL A)
7	Blue	OUT2 (ERR EVAL B)
8	Red	Analog (voltage 0…+10V or current 4…20mA)

C) Connecting the RS-232 interface:

4-pole female M5 connector type Binder 707
Connecting cable: cab-las4/PC (standard length 2m, cable jacket: PUR), also available: cab-4/USB, cab-4/ETH-500

	Pin	Assignment L-LAS-TB-…-AL-SC (control unit)
	1	+Ub
	2	0V (GND)
	3	RxD
	4	TxD

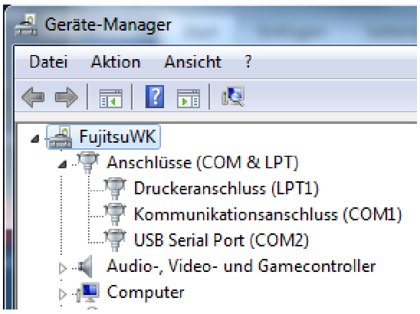
D) Establishing the connection through RS-232:



CONNECTION tab:

Click on this tab to open the CONNECTION window, where you can set various parameters for data exchange through the serial RS232 interface.
Basically the following default values are used for communication:

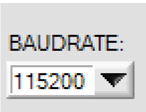
- Standard RS232 serial interface, no hardware handshake
- 3-wire-connection: GND, TXD, RXD
- Baud rates from 9600 baud to 115200 baud
- 8 DATA bits, 0 PARITY bit, 1 STOP bit
- Highest-order byte first (MSB first)
- **Standard baud rate = 115200 baud**



PORT [1…256]:

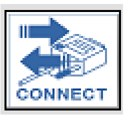
The number of the communication port can be set in this function field. Possible values are COM 1 to 255.
The communication port number can be found in the Windows® operating system under START/Control Panel/Device Manager.

As an alternative the communication port numbers that are available on the PC can be searched by clicking on the magnifier symbol.
The available COM ports are displayed in the status text field.



BAUDRATE:

The baud rate of the serial interface can be set in this function field:
Possible values: 9600 baud, 19200 baud, 38400 baud, 57600 baud, 115200 baud, 230600 baud, 460800 baud or 921600 baud. (**Setting when delivered = 115200 baud**).



CONNECT:

When you click on this button, the system attempts to establish a connection to the sensor with the set communication parameters. Feedback about the progress of connection establishment is shown in the status display field.

DISCONNECT:

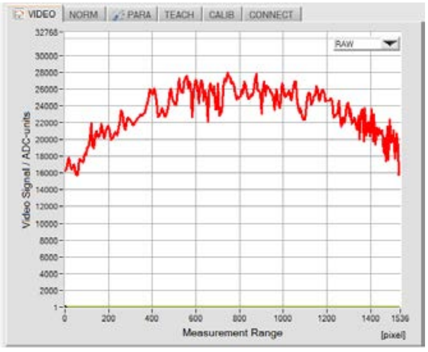
Click on this button to disconnect the connection with the sensor. The opened communication port becomes free again.

ACCEPT:

When you click on the ACCEPT button, the current communication settings are saved in the *TB-Scope.ini* file. When the *L-LAS-Spray-Control-Scope* software is started again, communication is established with the parameters saved in the *TB-Scope.ini* file.

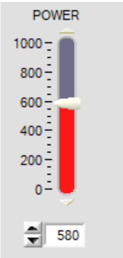
E) Adjusting the laser power / white balance:

Prerequisite: Successfully established connection ->> corresponding status line is displayed in the CONNECT tab.

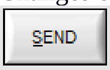


The laser power is adjusted in the VIDEO tab.
The function element for graphic selection must be set to RAW, which means that the raw data of the video signal are sent from the line sensor to the PC.

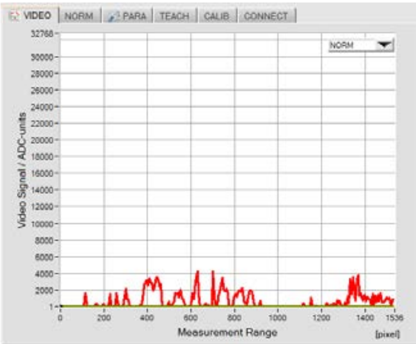
Please note: The graphic only is refreshed when data transfer is active, which means that data transfer must first be started.



POWER:
By way of adapting the laser power the VIDEO intensity profile should be adjusted in such a way that the curve lies in the upper third of the dynamic range.
Please note:
Changes only become active at the sensor when the SEND button is pressed!



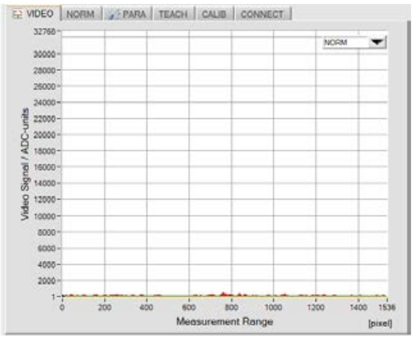
BEFORE WHITE BALANCE:



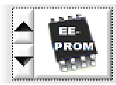
White balancing is performed in the VIDEO tab, the function element for graphic selection must be set to NORM.
White balancing must be performed after every change of parameters, e.g. of the laser POWER, or after a change of the integration time EXPOSE-TIME[ms].

White balancing compensates disturbing extraneous light influences or slight mechanical misalignments between transmitter and receiver.
It is recommended to perform white balancing in regular intervals. In an automated process white balancing can be started with the PLC directly before the measurement. For this purpose a short HIGH pulse (10ms < T < 750ms) is applied through the external input IN1/pin4/yellow.

AFTER WHITE BALANCE:

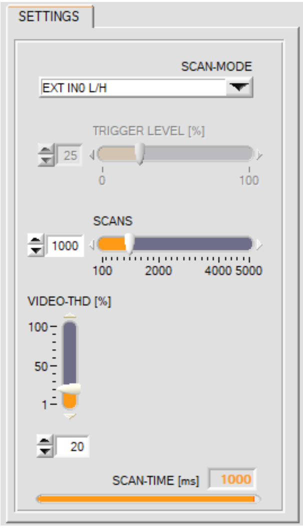


When you click on this software button, white balancing is automatically performed at the control unit. After white balancing the Y-values of the NORM video image should lie close to the X-axis over the complete measurement range of the line sensor.

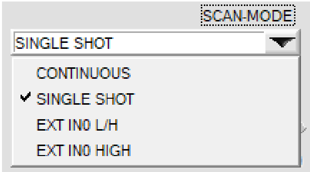


If the selection field / storage location is set to EEPROM the current white balance information is written to the non-volatile EEPROM of the control unit.

F) Recording of spray events:

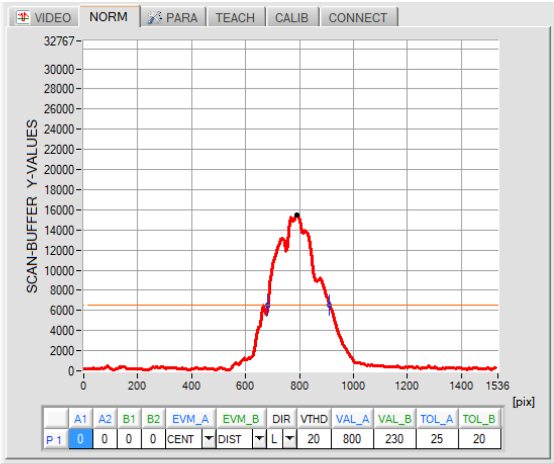


Spray events can be recorded in various ways.
A scan-mode must therefore be set first:



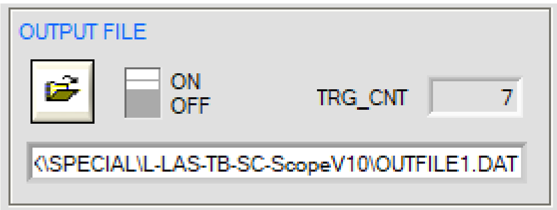
CONTINUOUS: Continuous spray jet scanning (for test purposes).
SINGLE SHOT: Single spray jet scan when the START button is pressed.
EXT IN0 L/H: Spray jet scan triggered with PLC through a L/H edge.
EXT IN0 HIGH: Spray jet scan triggered with PLC through a HIGH level.

Please note: The graphic only is refreshed when data transfer is active, which means that data transfer must first be started.



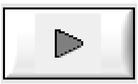
The picture opposite shows a typical picture of a spray recording. Above the X axis (pixels of the line sensor) the spray density distribution is visible as a red curve. Using an adjustable search threshold THD [%] (orange horizontal line), intersection points (edges) can be derived from the density profile. A black cursor indicates the position / height of the density maxima.

In the microcontroller, during the measured value recording, individual images of the line sensor are constantly accumulated and normalized to the spray density distribution after the recording with the SCAN counter reading.
X-axis: Pixel position (measurement range)
Y-axis: Normalised "spray density" information



OUTPUT FILE:
With the aid of these function fields, the evaluation results can be stored in an output file

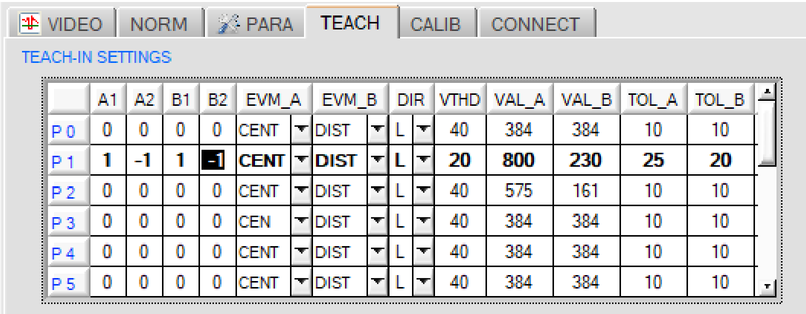
After clicking the File-Open button, a file name can be specified via a dialog box. The storage process can be enabled via the binary switch.
To save to the output file, the [NORM] –TAB must first be selected.



The measurement values are saved in the output file after clicking den [RUN] button (data-request).

Attention:
The data output to the output file is only possible in the following working (trigger) modes:
- SINGLE SHOT,
- EXT-IN0-L/H
- EXT IN0-HIGH

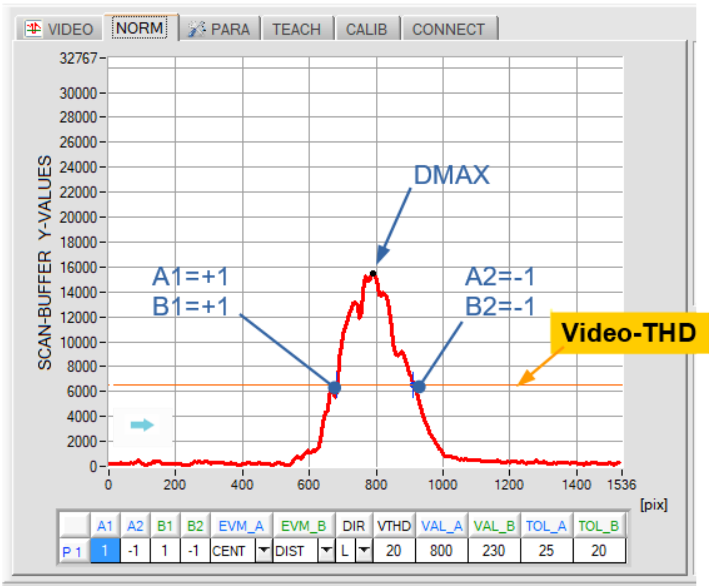
G) Working with the teach table:



	A1	A2	B1	B2	EVM_A	EVM_B	DIR	VTHD	VAL_A	VAL_B	TOL_A	TOL_B
P 0	0	0	0	0	CENT	DIST	L	40	384	384	10	10
P 1	1	-1	1	-1	CENT	DIST	L	20	800	230	25	20
P 2	0	0	0	0	CENT	DIST	L	40	575	161	10	10
P 3	0	0	0	0	CEN	DIST	L	40	384	384	10	10
P 4	0	0	0	0	CENT	DIST	L	40	384	384	10	10
P 5	0	0	0	0	CENT	DIST	L	40	384	384	10	10

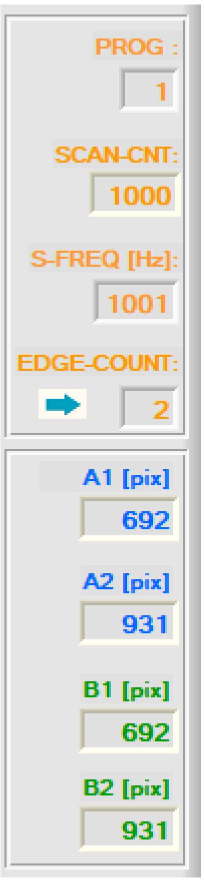
The learning table can store 16 programs. For each program, two independent edge evaluations EVM_A and EVM_B, as well as one video threshold VTHD and one edge search direction DIR can be specified. Furthermore, for each edge evaluation A and B, a separate tolerance band TOL_A and TOL_B can be preset for the respective teach-in value.

The teach table is used for searching edges in the normalised SCAN data field.
A1:= Edge index position (+1 = first rising edge, evaluation A)
A2:= Edge index position (-1 = first falling edge, evaluation A)
B1:= Edge index position (+1 = first rising edge, evaluation B)
B2:= Edge index position (-1 = first falling edge, evaluation B)
DIR:= Edge search direction L = from left to right (from pixel 1)
EVM_A / EVM_B: = EVALMODE A / B := OFF, POS, CENTER,DISTANCE, DMAX, AREA or SYMMETRY
VTHD:= Video threshold for edge search
VAL_A:= Teach-in value evaluation A
VAL_B:= Teach-in value evaluation B
TOL_A:= Tolerance value evaluation A (+ / - tolerance specification around teach-in value)
TOL_B:= Tolerance value evaluation B (+ / - tolerance specification around teach-in value)



Evaluation A:
Edge search from left to right, DIR=L, VTHD=20% (orange threshold)
EVM_A: CENTER position between the first rising edge [A1=+1] and the first falling edge [A2= -1].

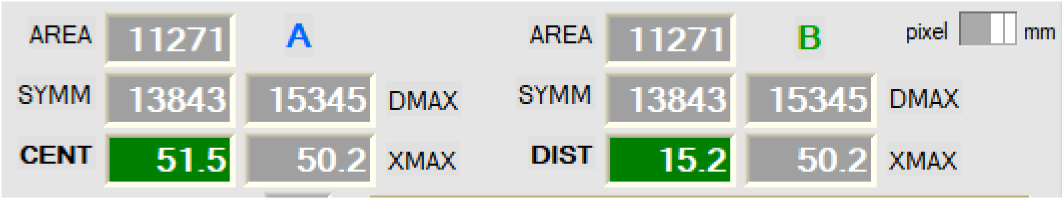
Evaluation B:
Edge search from left to right DIR=L , VTHD=20% (orange threshold)
EVM_B: DISTANCE, distance between first rising edge [B1=+1] to the first falling edge [B2= -1].



These display elements provide information about the detected edges from the normalised data field. The measurement values for the jet positions, based on the known pixel distances of the receiver line, can be calculated with the following formula:

Basic condition: 63.5[µm/pixel] = Pixel-pitch line sensor

$$mm_{Value} = (Pixel_{Value} * 63.5)/1000$$



The evaluation quantities AREA, SYMMETRY and DMAX are standardized quantities and refer to the maximum Y value range from 0 to 32767.

DMAX := density-maxima (0 … 32767)

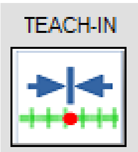
XMAX := x-position of the density-maxima

AREA := area under curve between edges A1 and A2, or B1 and B2

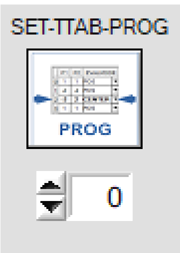
SYMM:= area ration:

$$SYMM = 16384 * (AREA1)/(AREA1 + AREA2)$$

SYMM [%]:= in percent $SYMM[\%] = 100 * (SYMM)/(32767)$



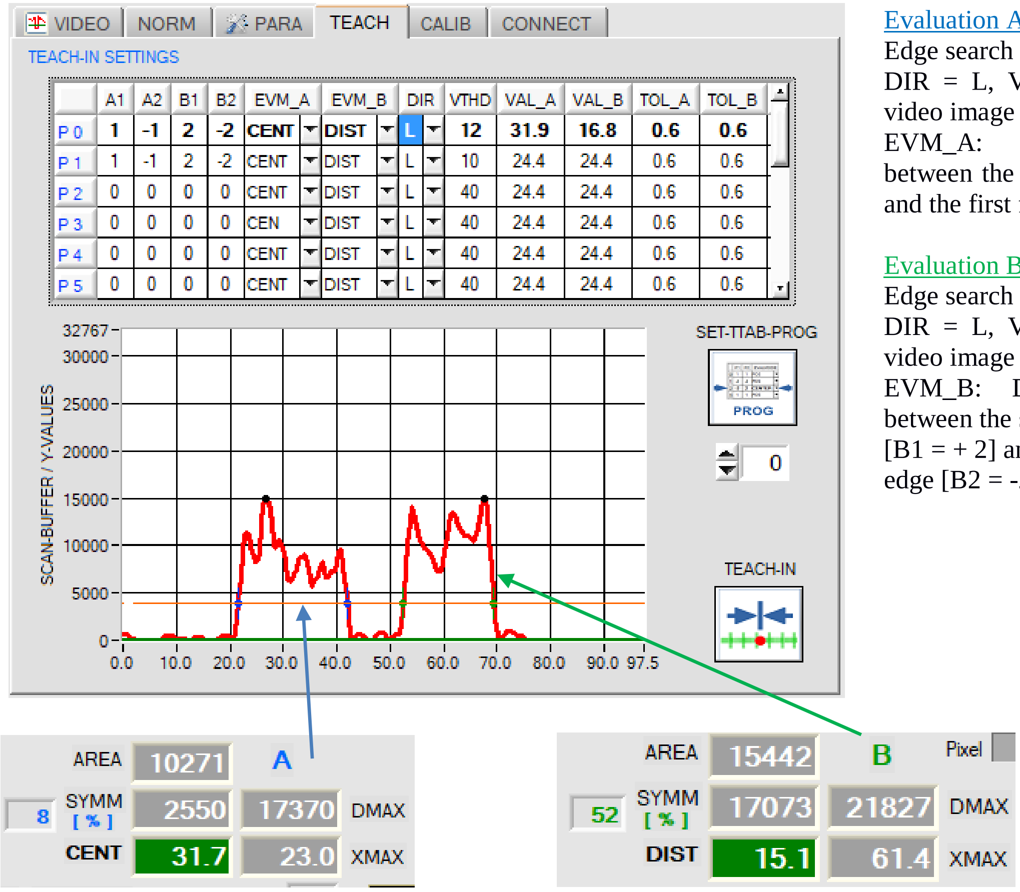
TEACH-IN FUNCTION:
Click the software-button to start the TEACH-IN function on the sensor. For the edge search at position A and position B, the specifications for the edge search (A1, A2, or B1, B2) of the currently activated row of the LEARNING TABLE are used.
The current edge positions and measured values of the respective evaluation modes (AREA, SYMM, DMAX and CENTER) are filled automatically into the LEARNING TABLE. The multifunction LED (orange) on the sensor housing flashes twice.



SET-TTAB-PROG:
Click the software button SET-TTAB-PROG to transfer the selected LEARN VECTOR of the TEACH-TABLE to the sensor. The sensor then works with the values of the selected line (program).

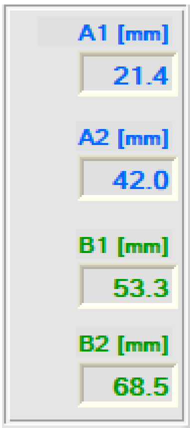
The program to be activated (= line) of the TEACH-TABLE can be selected using the numeric input field. Alternatively, the program can be selected by clicking on the first column in the TEACH-TABLE.

Evaluation of two spray jet positions A and B:

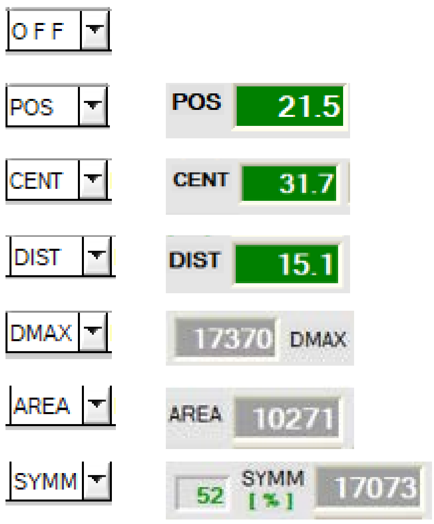


Evaluation A:
Edge search from left to right
DIR = L, VTHD = 12% (orange video image threshold)
EVM_A: CENTER position between the first rising [A1 = + 1] and the first falling edge [A2 = -1].

Evaluation B:
Edge search from left to right
DIR = L, VTHD = 12% (orange video image threshold)
EVM_B: DISTANCE, distance between the second rising [B1 = + 2] and the second falling edge [B2 = -2].



Beispiele für Auswerte-Optionen bei Position A und B:



Evaluation of position A or B is deactivated. The corresponding fields in the LEARNING TABLE are "greyed out".

Evaluation of an edge position. e.g. POS A1 = +1 first rising edge.
Edges result from the intersection of video threshold / density curve.

Evaluation of the center position POSITION A between the first rising (A1 = +1) and the first falling edge (A2 = - 1).

Evaluation of the distance at POSITION B between the second rising edge (B1 = +2) and the second falling edge (B2 = - 2).

Evaluation of density maxima at POSITION A between the two edges (A1 = + 1, A2 = - 1)

Evaluation of the area under the density curve at POSITION A between the two edges (A1 = + 1, A2 = -1).

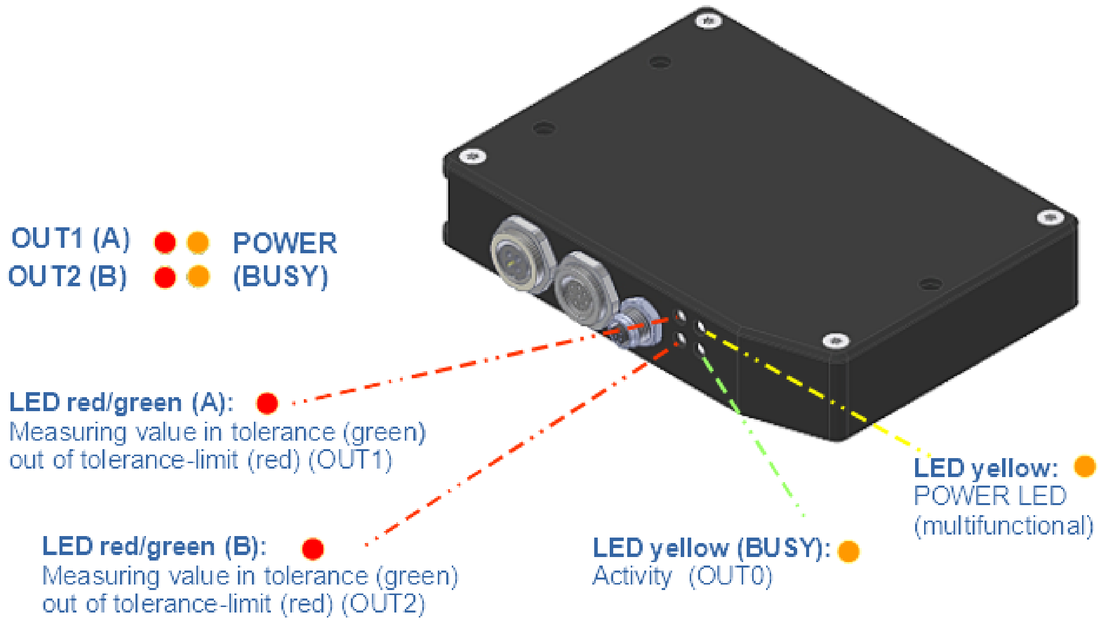
Evaluation of the symmetry at POSITION B between the second rising edge (B2 = +2) and the second falling edge (B2 = - 2).

4 Annex

4.1 Display elements

At the housing of the *L-LAS-TB-…-AL-SC control unit* there are 4 two-color LEDs that visualise system states.

L-LAS-TB- … - R-AL-SC (control unit / receiver)



4.2 Function of digital input IN0, IN1

PLC CONNECTION:

Trigger input IN0/pin3/green:

Measurement value recording in the control unit can be started simultaneously with the spray process. Triggering at the control unit is performed with a HIGH level at digital input IN0/pin3/green. Data recording starts with a L/H edge at IN0, and stops with the following H/L level change. The evaluation result is available after typ. 1ms and can be read out through the RS-232 interface (order <ORDER=8>. A maximum of 5000 scans can be performed, which with a scan rate of 1kHz means a time of 5 seconds.

The "triggered" operating mode (SCAN-MODE) must first be set with the *L-LAS-TB-Spray-Control-Scope* software.

TRIGG-IN0 HIGH:
External triggering of measurement value recording through a HIGH level (+24VDC) at digital input IN0.

Input IN1/pin4/yellow – RESET PULSE:

This digital input at the control unit is used for "white balancing" at the line sensor. When a short HIGH pulse (0ms < T < 750ms) is applied immediately before the start of the spray process, the pixel line can be balanced anew to the current ambient conditions (ambient light, slight misalignment of the sensor, etc.). For achieving optimal evaluation results this white balance should be performed before every measurement.

Digital output OUT0/pin5/gray (BUSY):

Digital output OUT0 is used for the "Handshake" with the PLC. When data recording is started at the sensor, this is indicated by a level change at digital output OUT0/pin5/gray.

The evaluation result is available as soon as the BUSY output at the sensor changes back to its original level.

4.3 Laser warning

LASER WARNING
Solid state laser, λ=670 nm, 0.4mW max. optical power, Laser class 1 acc. to EN 60825-1 The use of these laser transmitters therefore requires no additional protective measures.

CLASS 1 Laser Product

DIN EN 60825-1: 2008-05

4.4 RS232 interface protocol

RS-232 data transmission:

- 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
- Default baud rate: 115200 baud

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)

<ORDER>	Meaning of the 2nd byte <order>:	ORDER-TABLE
0	NOP	no operation
1	Send parameter from PC to L-LAS-RAM	PC ⇒ L-LAS-RAM
2	Get parameter from L-LAS-RAM	L-LAS-RAM ⇒ PC
3	Send parameter from PC to L-LAS EEPROM	PC ⇒ L-LAS-EEPROM
4	Get parameter from L-LAS EEPROM	L-LAS-EEPROM ⇒ PC
5	Echo check: Get echo from L-LAS	first word=0x00AA=170dec
7	Get firmware version info from L-LAS	L-LAS ⇒ PC
8	**Get measured values from L-LAS**	L-LAS-RAM ⇒ PC
9	Get video image from L-LAS	L-LAS-RAM ⇒ PC
11	**Activate single measurement data scan (SINGLE SHOT)**	**PC ⇒ L-LAS-RAM**
12	**Activate white balance (WHITE-BALANCE)**	**L-LAS-RAM ⇒ PC**
16	**Activate evaluation program (PROG-NO)**	**PC ⇒ L-LAS-RAM**
26	Send teach vector (TEACH-TABLE VECTOR)	PC ⇒ L-LAS-RAM
27	Get teach vector (TEACH-TABLE VECTOR)	L-LAS-RAM ⇒ PC

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: $X^8+X^5+X^4+1$

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

RS-232 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 > : Get 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**	**170**	**0**	**72**	**0**	**XXX**	**82**	**L**	**-**	**L**	**A**
		ARG=170 (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
S	**-**	**T**	**B**	**-**	**S**	**C**	**-**	**T**	**S**	**L**	**X**

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
X	**-**	**A**	**L**		**V**	**1**	**.**	**0**	**.**	**0**	

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
	2	**6**	**/**	**M**	**A**	**R**	**/**	**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 = 11 > : ACTIVATE SINGLE MEASUREMENT DATA SCAN AT THE SENSOR

Activate a single measurement data scan at the sensor. The number of scans (single measurements) is set in the argument <ARG> of the header frame.

DATA FRAME PC → Sensor (8 bytes)
ARG = NUMBER OF SINGLE MEASUREMENTS (100 … 5000)

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)	**11**	232	3	0	0	170	67
		ARG =1000		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 (dec)	**11**	0	0	0	0	170	67
		ARG=0		LEN=0			

< ORDER = 12 > : ACTIVATE WHITE BALANCE AT THE SENSOR

Activate white balancing at the sensor. The storage location is set in the argument <ARG> of the header frame.

DATA FRAME PC → Sensor (8 bytes)
ARG: 0=RAM, 1=EEPROM

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)	**12**	0	0	0	0	170	67
		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 (dec)	**12**	0	0	0	0	170	67
		ARG=0		LEN=0			

< ORDER = 16 > : ACTIVATE EVALUATION PROGRAM AT THE SENSOR

Activate the current evaluation program at the sensor. The program number (0 to 15) is sent in the argument.

DATA FRAME PC → Sensor (8 bytes)
ARG = PROG-NO

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)	**16**	1	0	0	0	170	65
		ARG =1		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 (dec)	**16**	1	0	0	0	170	65
		ARG=1		LEN=0			

< ORDER = 8 > : GET MEASURED VALUES [PIXEL] FROM THE L-LAS-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 + 64) 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	64	0	xxx	118	180	2	163	3
		ARG=0		LEN=64				PIX_A1 = 692		PIX_A2 = 931	

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
180	2	163	3	43	3	239	0	241	59	241	59
PIX_B1 = 692		PIX_B2 = 931		XVAL_A = 811		XVAL_B=239		DMAX_A=15345		DMAX_B=15345	

Byte25 Data	Byte26 Data	Byte27 Data	Byte28 Data	Byte29 Data	Byte30 Data	Byte31 Data	Byte32 Data	Byte33 Data	Byte34 Data	Byte35 Data	Byte36 Data
Raw9	Raw9	Raw10	Raw10	Raw11	Raw11	Raw12	Raw12	Raw13	Raw13	Raw14	Raw14
22	3	22	3	7	44	7	44	19	54	19	54
IDX_A=790		IDX_B=790		AREA_A=11271		AREA_B=11271		SYMM_A=13843		SYMM_B=13843	

°
°
°

Byte61 Data	Byte62 Data	Byte63 Data	Byte64 Data	Byte65 Data	Byte66 Data	Byte67 Data	Byte68 Data	Byte69 Data	Byte70 Data	Byte719 Data	Byte72 Data
Raw27	Raw27	Raw28	Raw28	Raw29	Raw29	Raw30	Raw30	Raw31	Raw31	Raw32	Raw32
231	3	232	3	231	3	0	0	0	0	0	0
DYNTIME=999		SCANCOUNT=1000		SCANTIME=999				RAW_31=0		RAW_32=0	

Name	Value	Type
raw	0x0073263C	raw_struct
raw.pixA1	692	unsigned short
raw.pixA2	931	unsigned short
raw.pixB1	692	unsigned short
raw.pixB2	931	unsigned short
raw.xvalA	811	unsigned short
raw.xvalB	239	unsigned short
raw.dmaxA	15345	unsigned short
raw.dmaxB	15345	unsigned short
raw.imaxA	790	unsigned short
raw.imaxB	790	unsigned short
raw.areaA	11271	unsigned short
raw.areaB	11271	unsigned short
raw.symmA	13843	unsigned short
raw.symmB	13843	unsigned short
raw.emodA	2	unsigned short
raw.emodB	3	unsigned short
raw.edcjet	2	unsigned short
raw.raw16	0	unsigned short
raw.eprog	1	unsigned short
raw.instate	0	unsigned short
raw.outstate	0	unsigned short
raw.runstate	1	short
raw.videomax	31964	unsigned short
raw.mvstart	0	unsigned short
raw.mvend	0	unsigned short
raw.dynpow	0	unsigned short
raw.dyntime	999	unsigned short
raw.scncnt	1000	unsigned short
raw.scntime	999	long int
raw.raw31	0	unsigned short
raw.raw32	0	unsigned short

< ORDER = 26 > : SEND JET TEACH VECTOR ENTRY TO L-LAS-SENSOR

The teach table of the control unit for the evaluation of jets contains up to 16 entries (programs). These entries in the teach table are used for edge searching by way of the intensity distribution over the individual pixels at the line sensor (see manual). Every entry in the teach table (teach vector) has a length of 16 words (32 bytes).
Currently only the first 7 entries in the teach vector are used, but column entries 8 to 16 of the teach vector nevertheless must be sent (total of 40 bytes = 8 header bytes + 32 data bytes).
<u>Example:</u>
Teach table program ARG = 1.
The first 7 columns of the teach table are processed in the L-LAS control unit.
The remaining columns 8-16 of the teach table nevertheless must be sent (assigned 0).

	A1	A2	B1	B2	EVM_A	EVM_B	DIR	VTHD	VAL_A	VAL_B	TOL_A	TOL_B
P 1	1	-1	1	-1	CENT	DIST	L	20	800	230	25	20

DIR: 0=LEFT, 1=RIGHT

EVALMODE: 0=OFF, 1=POS, 2=CENTER, 3=DISTANCE, 4=CENTER, 5=DMAX, 6=AREA, 7=SYMMETRY

DATA FRAME PC → Sensor (8 bytes +32 bytes)
ARG = TEACH VECTOR LINE INDEX 0 … 15

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)	Data1 (lo byte)	Data1 (hi byte)	Data2 (lo byte)	Data2 (hi byte)
85 (dec)	**26**	1	0	32	0	XXX	50	1	0	-1	-1
		ARG=1		LEN=32				A1 = 1		A2 = -1	

Byte13 Data	Byte14 Data	Byte15 Data	Byte16 Data	Byte17 Data	Byte18 Data	Byte19 Data	Byte20 Data	Byte21 Data	Byte22 Data	Byte23 Data	Byte24 Data
Data3	Data3	Data4	Data4	Data5	Data5	Data6	Data6	Data7	Data7	Data8	Data8
1	0	-1	-1	2	0	3	0	0	0	20	0
B1 = 1		B2 = -1		EVM_A = 2		EVM_B = 3		DIR = 0		VTHD=20	

Byte25 Data	Byte26 Data	Byte27 Data	Byte28 Data	Byte29 Data	Byte30 Data	Byte31 Data	Byte32 Data	Byte33 Data	Byte34 Data	Byte35 Data	Byte36 Data
Data9	Data9	Data10	Data10	Data11	Data11	Data12	Data12	Data13	Data13	Data14	Data14
32	3	230	0	25	0	20	0	0	0	0	0
VAL_A=800		VAL_B=230		TOL_A=25		TOL_B=20		0		0	

Byte37 Data	Byte38 Data	Byte39 Data	Byte40 Data
Data15	Data15	Data16	Data16
0	0	0	0
0		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 (dec)	**26**	0	0	0	0	170	50
		ARG=0		LEN=0			

The argument of the header frame that is sent back from the control unit shows the status of data exchange:
ARG = 0 , no error
ARG = - 105, data transmission error

< ORDER = 9 > : GET DATA BUFFER FROM SENSOR

ATTENTION: A maximum of 256 integer values = 512 bytes of data can be read out.
The argument <ARG> of the header frame determines which data memory is read out:

ARG = 0 : STATISTICS DATA BUFFER AFTER EVALUATION (NORM value)
ARG = 1 : RAW DATA CMOS VIDEO LINE (256 pixels distributed over full pixel number)
ARG = 2 : WHITE BALANCE BUFFER (256 values distributed over full pixel number)
ARG = 3 : CURRENT SCAN BUFFER (NORM value)

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	

●
●
●

Byte509 Data	Byte510 Data	Byte511 Data	Byte512 Data	Byte513 Data	Byte514 Data	Byte515 Data	Byte516 Data	Byte517 Data	Byte518 Data	Byte519 Data	Byte520 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	

With <ARG = 0> (STATISTICS DATA BUFFER) and <ARG = 3> (CURRENT SCAN BUFFER) the current counter (SCAN-COUNTER) also is sent in the last data word (byte 519, byte 520).