Software Description

Walther Optic Sensor WOS

Article Number*:* 979905

IMPORTANT
Before unpacking or start of operation of this device, please read and follow this software description carefully!

The device will only be operated, serviced or repaired by qualified personnel who are familiar with the software description as well as the guiding rules and regulations on job safety and accident prevention.

Always keep this "Original software description" close to the device!
The manual should always be easily available!

WALTHER SYSTEMTECHNIK

Walther Systemtechnik GmbH – D 76726 Germersheim
Telefon: +49 (0)7274-7022-0 Telefax: +49 (0)7274-7022-91
http://www.walther-2000.de – info@walther-2000.de

Table of Contents Page

1 Software „WOS-ADJUST”

The color sensors can be extensively configured by PC software. Therefore software tools are shipped with the sensors for individually adjusting the sensors to any operational environment. The software functions are as follows:

- Parameter settings (signal gain, scanning frequency, processing mode etc.).
- Color sampling („Teach In“) procedure
- Color evaluation/processing (recognition and classification)
- Displaying color values

The functions of the parameter program are described below.

1.1 Installation of the Software

Installing and operating the software require the following minimum system parameters:

- · PC with 300 MHz CPU
- · 40Mbyte free hard disc space
- · mouse
- · RS232 interface
- · USB interface for sensors with USB option
- · CD-ROM drive
- · VGA graphic display with minimum resolution 800x600
- · MS Windows ® operating system (WIN98, WIN ME, WIN2000, WIN XP)

For installing the software, insert the CD-ROM in the CD-ROM drive and follow the installation instructions on the screen. Or, alternatively, start the program “SETUP.EXE” from the folder: *[CD drive]:\WOS-ADJUST*.

1.2 Programm Operation

1.2.1 Selecting Communication Interface

The sensor system can be connected via RS232 or USB to the computer. The actual interface will be selected after the start of the program (Fig. 2).

SERIAL CONNECTION
The settings for the serial interface can be seen on the left. The communication port can be chosen in the field *“COM PORT SELECT“* (Figure). Use *“ADDRESS“* to choose the address of an individual sensor when several sensors are connected to the serial port (see also *“SENSOR ADDRESS“* further below in section *“SERVICE“*).

IMPORTANT
The software is limited to the COM ports 1 to COM 9.

By default the communication port is configured as follows:

- · Baud Rate: 28800
- · Data bits: 8
- · Parity: none
- · Stop bits: 1
- · Flow control: none

Status and error information will be displayed in the text box “STATUS SERIAL“ below the button “GET ID“.

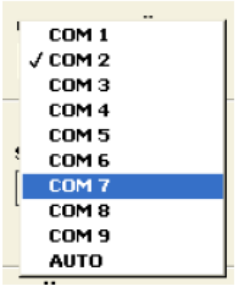


Fig. 1: *Selecting the RS232 interface port*

Fig. 2: *Start window and interface selection*

USB CONNECTION

The program can be started via USB on the right side if a USB interface is available. It is possible to connect several devices to the USB at the same time. The number of detected devices will be shown in the field “DEVICES DETECTED”. The desired USB can be selected with the switch in the field “SELECT USB DEVICES” (Figure 3).

Fig. 3: *Selecting the desired USB device*

SERVICE

The switch in the field *“SERVICE“* starts a service program which allows you to perform special hardware functions and basic settings. The functions of this service tool are described in section 1.2.5.

1.2.2 Sensor Settings

After starting a selected connection, the parameter window for setting the sensor system will display. See Figure 4. The sensor parameters are read out and displayed in the program window.

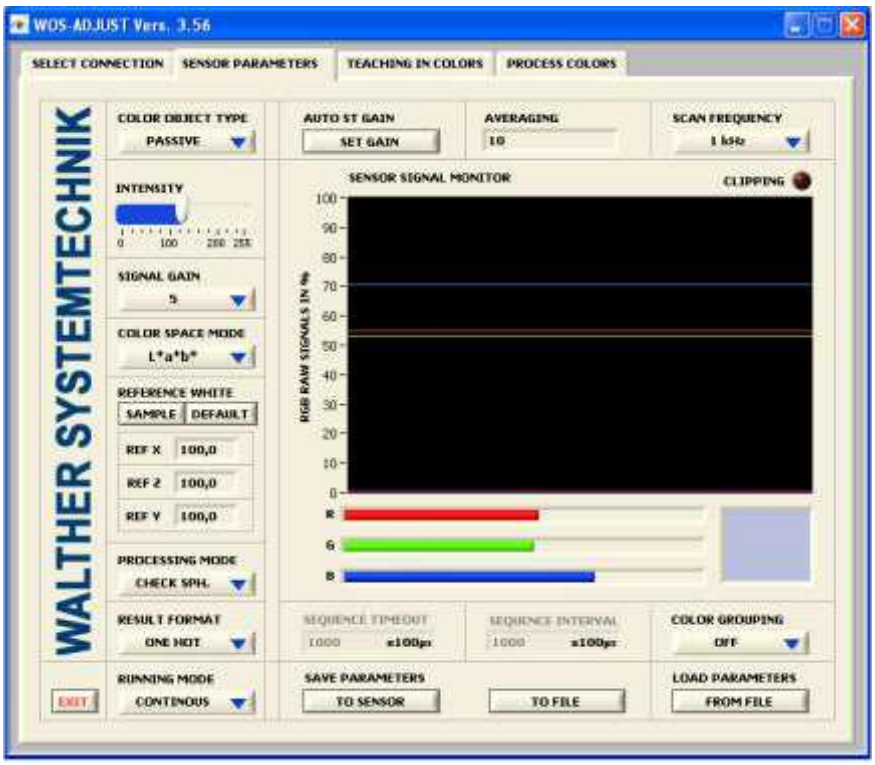


Fig. 4: *Window for parameter setting*

COLOR OBJECT TYPE
At first the type of object to be measured will be selected in the field „COLOR OBJECT TYPE“ (Figure 5). The setting “ACQUISITION OFF” turns the signal gain (acquisition) of the sensor off.

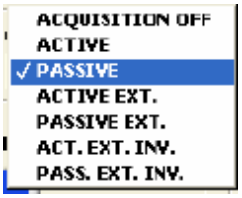


Fig. 5: *Selecting the object type*

For non self-shining objects, so-called body colors, choose the setting “PASSIVE”. This will activate an installed light source for illuminating the object. For self-shining objects, such as lamps, LEDs, etc., choose the “ACTIVE“ mode. This will turn off the installed light source for the actual measuring.

There are 4 additional “EXT:” modes which are intended for use with an external trigger source to be provided on the TRG1 input. With this mode the sensor can be synchronized with an external frequency. The modes with an „INV.“ addition invert the TRG1 signal.
For self-shining objects an ambient light compensation can be realized. For non-self-shining objects a ‚master–slave’ operation can be realized in such a way that one sensor is switched to PASSIVE mode and all other sensors are adjusted to the PASSIVE EXT (or PASSIVE EXT.INV.) mode. The master-slave operation is particularly useful for avoiding light interfering effects if several sensors are used in close proximity.

INTENSITY
Select *„INTENSITY“* to adjust the brightness of the built-in light source with the intensity regulator.

SIGNAL GAIN / REINFORCEMENT
The amplification gain (out of 4 steps) of the color signals from the primary color sensor can be selected in the field „SIGNAL GAIN“ (Figure 6).

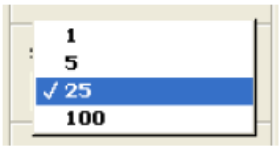


Fig. 6: *Signal gain*

AUTOMATIC REINFORCEMENT / GAIN
The amplification gain and illumination intensity can be automatically adjusted when the switch in the „AUTO SET GAIN“ field is set on. The signal range will be set to approximately 90% of the entire range.

IMPORTANT
Before successfully applying the function „AUTO SET GAIN“ you have to verify that the sensor system has been placed in its operating position and the distance between sensor and objects has been fixed. It is recommended that a white colored object or an object with the highest signal amplitude (brightest object to be measured) will be used for adjusting the signal amplification gain and further for preventing any overdrive.

COLOR SPACE MODE
Select „COLOR SPACE MODE“ for setting and determining the color space to be used. Depending on the light control setting, one of the following color spaces shown in the table can be selected. The figure shows the selection field in “COLOR SPACE MODE”.

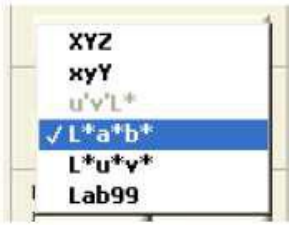


Fig. 7: *Color space modes*

Table 1: *Color space modes depending on PASSIVE or ACTIVE*

PASSIVE (Body colors)	ACTIVE (Self-shining)
XYZ	XYZ
xyY	xyY
L*a*b*	u'v'L*
L*u*v*	
Lab99	

Note 1:
Due to the limited hardware accuracy of the sensor and the used non-standard illumination source (white light LED), the obtained color values in the corresponding color spaces are not accurate for colorimetry!

Note 2:
The ranges of the color values in the parameterization software differ from the common ranges of color values! Table 2 shows the range of the color values for the several color space modes of the software in comparison to the common ranges. **The visible range of color diagrams is pre-defined to a suitable value. In order to adapt the color diagram to the desired range, the axes can be adjusted by simply editing the numbers on the corresponding axis.**

Table 2: *Axis scaling for the parameterization software*

Color space	common range	WOS-Adjust range	Multiple
XYZ	X: 0…100 Y: 0…100 Z: 0…100	X: 0…100 Y: 0…100 Z: 0…100	1 1 1
xyY	x: 0…1 y: 0…1 Y: 0…100	x: 0…100 y: 0…100 Y: 0…100	100 100 1
L*a*b*	L*: 0…100 a*: -500…+500 b*: -200…+200	L*: 0…100 a*: -500…+500 b*: -200…+200	1 1 1
L*u*v*	L*: 0…100 u*: -1300…+1300 v*: -1300…+1300	L*: 0…100 u*: -1300…+1300 v*: -1300…+1300	1 1 1
u'v'L	L: 0…100 u': 0…1 v': 0…1	L: 0…100 u': 0…100 v': 0…100	1 100 100
DIN99	L: 0…100 a: -500…+500 b: -200…+200	L: 0…100 a: -500…+500 b: -200…+200	1 1 1

REFERENCE WHITE

For storing a color value as reference white, the actual sensor signals can be transferred by clicking *„SAMPLE“* in the field *„REFERENCE WHITE“.* Sampling the reference values is helpful only if a white colored object is used and if the signal levels were set with the GAIN and INTENSITY function before. The function AUTO SET GAIN can simplify adjusting the signal range. The button DEFAULT sets the color values for the reference white to 100.

The reference values can be manually edited by clicking on the number and then change the value.

Note 1:
The reference white is crucial for color measurements and display; therefore it has to be re-sampled every time the signal settings (such as AUTO SET GAIN or LIGHT CONTROL – INTENSITY) change.

Note 2:
The reference white is required for properly displaying the color on the monitor and not needed for further signal processing in the color spaces XYZ and xyY. On the other hand, the reference white is also needed for a correct color transformation into the color spaces L*a*b*, L*u*v* and Lab99. Therefore sampling of real sensor values is recommended for the color spaces L*a*b*, L*u*v* and Lab99. However the sensor will work acceptable even though a reference white has not been sampled as long as the color values do not exceed the pre-defined reference white. In general the sensor works well when using the DEFAULT reference white.

PROCESSING MODE

The sensor system provides up to 5 different processing modes. The desired mode can be selected from „PROCESSING MODE“, as shown in Figure 8 below. Table 3 shows all processing modes and their description in detail.

Fig. 8: *Processing modes*

Table 3: *Processing modes and description*

Processing mode	Description
CLASSIFY	The actual measured color value is being classified among pre-sampled color values. A classification is always done independently if the colors match or not. As an example shows, if there is only one color RED stored, all colors will be later classified as RED. Figure 9 shows the classification of the actual color value Cm to the stored color C3 because the distance between Cm and C3 is the smallest one.
CHECK SPH.	In the mode CHECK SPH. (spherical) it is to be checked if a measured color is within a spherical tolerance room. If the measured color is within such a spherical room, the check is successful (color recognized), otherwise the check was unsuccessful (color not recognized). Figure 10 shows spherical tolerances and a measured color Cm that is within the tolerance C3 and hence was recognized as the color C3.
CHECK CYL.	The mode CHECK CYL. (cylindrical) provides a method for assigning cylindrical tolerance spaces. Tolerance parameters can be configured separately for color and brightness. The measurement principle is shown in Figure 11. Two tolerance parameters (color and brightness tolerance) are necessary. This mode can not be applied in the XYZ color space because there is no coordinate for brightness.

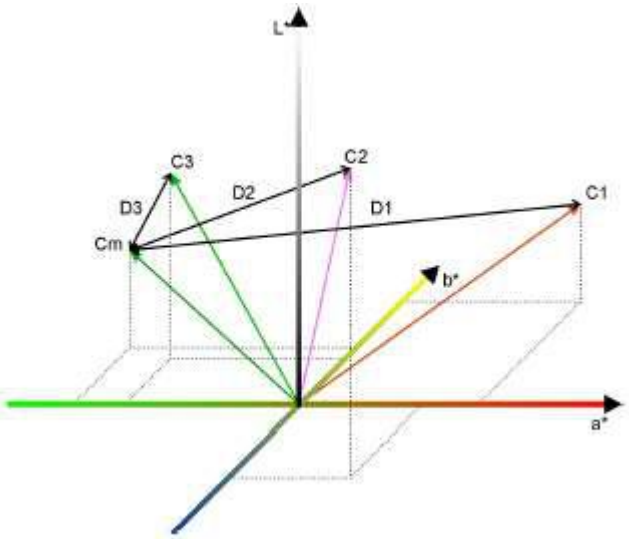


Fig. 9: *Color classification diagram*

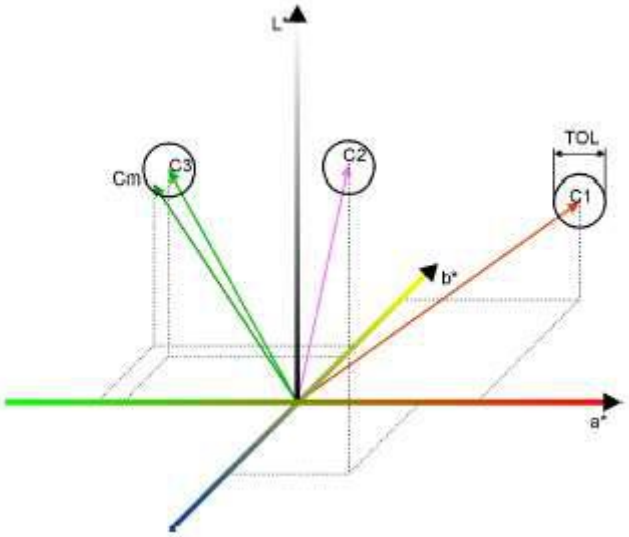


Fig. 10: *Color space with spherical tolerances*

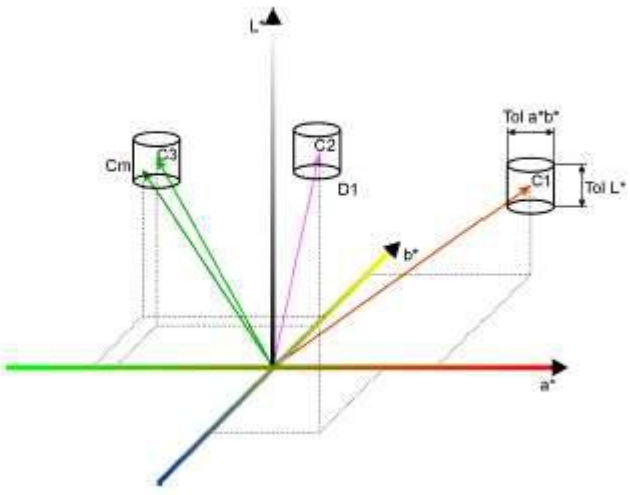


Fig. 11: *Color space with cylindrical tolerances*

RESULT FORMAT

In the field “RESULT FORMAT“ the user defines how the results of the color checking or classification are represented at the outputs of the color sensor. Figure 12 shows the 6 possible output formats and table 4 shows their description in detail.

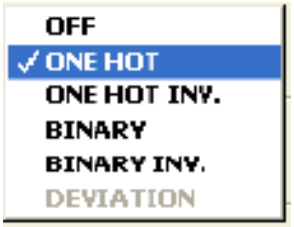


Fig. 12: *Formats of color processing results at the sensor outputs*

Table 4: *Output formats for color checking and classification*

Selection	Description
OFF	The outputs keep their last result and are switched off.
ONE HOT	This is a “1-out-of-N” encoding. Each color value in the internal color table (please refer to sect. 1.2.3, subsect. “COLOR TABLE”) is directly assigned to one output signal. Thus there are as many colors possible as sensor outputs are available.
ONE HOT INV.	Refer to “ONE HOT“, but output signals are inverted
BINARY	The result is encoded binary and thus 2^n (n = number of sensor outputs) states are possible. The index 255 is used as “Not recognized” or “Out of tolerance” in the CHECK mode.
BINARY INV.	Refer to “BINARY“, but output signals are inverted
DEVIATION[4]	If no matching color can be found, this result format indicates the color-deviation from the first entry of the color-table. The meaning of the different output ports is shown in Table . This mode is only available with the color-space-modes L*a*b* or LAB99 and processing mode CHECK CYL.

Table 5: *Deviation mode signal assignment*

Ausgang	Anzeigensegment
OUT0	1 = color detected, 0 = No color detected
OUT1	Darker
OUT2	Brighter
OUT3	Deviation to red
OUT4	Deviation to green
OUT5	Deviation to yellow
OUT6	Deviation to blue
OUT7	Not used

RUNNING MODE

The **sensor ports** can be used continuously or triggered („EXTERN TRG.“). The selection can be made in “RUNNING MODE” (Figure 13).

When continuously used („CONTINUOUS“), the outputs are always update das fast as the current adjusted response time.

In the „EXTERN TRG.“ mode, the outputs are updated when a rising edge at the input TRG 0 is detected and will remain in this state until the next edge occurs.

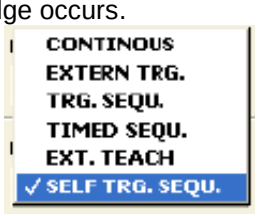


Fig. 13: *Output signals running mode*

Additionally the sensor supports three different sequential modes. This makes it possible to detect a color sequence defined by the color table in a trigger based mode or in a time based mode.

The triggered sequence mode “TRG.SEQU.“ compares the actual result with an entry in the color table (please refer to sect. 1.2.3, subsection “COLOR TABLE”) after each appearing trigger on TRG 0. It starts with entry 0 and goes to the next entry after every successful comparison (correct color). If a detected color does not match its table entry, or the time between two trigger events is too long (timeout), the sequence will reset to entry 0. The adjustment of the timeout can be done in the field “SEQUENCE TIMEOUT” (Figure 14).

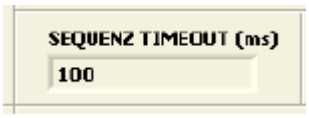


Fig. 14: *Timer configuration for triggered color sequence*

You can use this mode to recognize a color series of objects with undetermined object distance and appearance time (e.g. to trigger on colored objects on an assembly line that appear in the same order but undetermined in time).

The time based sequence mode („TIMED SEQU.“) basically works like the triggered sequence mode; only the beginning of a new sequence has to be indicated by a trigger event on TRG 0. After that the sensor automatically compares the colors after a programmable time (interval). In case a color does not match its corresponding table entry, the sequence will reset to entry 0. The adjustment of the timer interval can be done in the field “SEQUENCE INTERVAL” (Figure 15).

Fig. 15: *Interval configuration for timed color sequence*

Here too, the table entry will start with 0; in the case of a wrong color the sequence detection will start again.

Use this mode to recognize a color series of equidistant spaced and timed objects (e.g. a rotating wire with color stripes).

A third sequence mode is chosen by activating “SELF TRIGG SEQ”. For this mode no external trigger is required. The sequence starts by detecting a first color. If every color of the color table is recognized in the right order, the sequence is finished successfully. If the time adjusted in the field „SEQUENCE TIMEOUT“ to detect the next valid color of the color table is exceeded, an error signal occurs. The same applies for detecting the wrong color sequence order. The coding of the states is shown in Table 6. **This mode requires the “CHECK SPH” or “CHECK CYL” mode.**

Each state of the sequence detection is treated like an ordinary result of a single color detection and is encoded accordingly. The appearance of the sequence state on the output port can be adjusted by setting the proper “RESULT FORMAT” mode. The following modes are detected:

Table 6: *Sequence encoding*

State	Description
OFF	Waiting for start
0	Sequence active
1	Sequence finished successfully
2	Wrong color detected
3	Trigger Timeout (triggered sequence)
4	Trigger too early (timed sequence)

Use “PROCESSING MODE“ to activate another mode: “EXT. TEACH“. When you select this mode, you are able to remote-control the teaching-on of colors into the color table. Once a signal is received at the sensor input TRG 0, the current color value will be transferred into the table.

NOTE:
The color table on your computer will be updated only after a restart of the parameterization program.

AVERAGE VALUE
Use “AVERAGING“ to adjust the averaging values for the sensor signals in a range of 1…65535.

Large averaging values are recommended for poor signal quality. Keep in mind that large averaging values reduce the response time of the sensor (Table 7).

SCAN FREQUENCY
The scanning frequency of the sensor system in kHz can be defined in the field “SCAN FREQUENCY“ (figure 16).

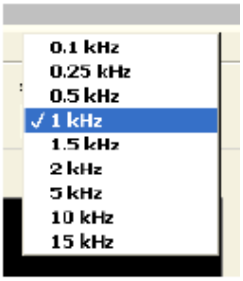


Fig. 16: *Scanning frequency of the sensor system*

Please notice the following relationship. If the scanning frequency is chosen low, the power consumption of the sensor is low, too (illuminant will get darker). But if a high scanning frequency is chosen, the power consumption rises (due to the increasing light intensity). The losses can an influence on the heat development.

Note 1:
The chosen frequency has also influence on the system’s ambient light compensation properties. In the case of artificial ambient light (modulated mainly at 100 Hz) a frequency greater than 1 kHz is recommended in order to achieve good compensation properties.

Note 2:
Due to hardware limitations, higher frequencies for higher signal gain settings (SIGNAL GAIN5 > 5) are not available.

Table 7: *Samples values for the correlation of response time on FREQUENCY and AVERAGING values*

FREQUENCY	AVERAGING	Response time
1kHz	1	1ms
10kHz	10	1ms
1kHz	100	100ms
10kHz	10000	1000ms

SENSOR SIGNAL MONITOR

The current sampled signals from the primary sensor are displayed in the graph “SIGNAL MONITOR“ as raw data. Their description is shown in the following table:

Table 8: *Description of signals on the sensor signal monitor*

Signal color	Description
Light red	Raw data of the red signal (bright phase)
Light green	Raw data of the green signal (bright phase)
Light blue	Raw data of the blue signal (bright phase)
Dark red	Raw data of the red signal (dark phase)
Dark green	Raw data of the green signal (dark phase)
Dark blue	Raw data of the blue signal (dark phase)

The three color bars red, green and blue are displayed underneath the sensor signal monitor and show the differential signals of the raw data. In the self-shining mode the signals are identical to the signals in the bright phase. On the right hand side a color sample is displayed which is continuously calculated from the actual three color signals.

Note 1:
The signal data from the dark phase are zero in the self-shining mode and thus not visible. If the signal data from the dark phase of body colors (passive mode) are very small, they are also not visible. Moreover the signal amplitudes from the bright and dark phase can possibly overlap and hence only one color can be seen at the same time.

Note 2:
The color sample on the right hand side of the three color bars reproduces a color which is similar to the measured object after setting a good reference white. However the color can be incorrect and not 100% identical and shall merely serve as orientation, e.g. during the color sampling process (“Teach-in”) or when displaying tolerance boundaries in color diagrams.

You will find a bar graph below the signal monitor diagram which displays the differencing signals of the raw data. If the difference of the signals from the light-and-dark-phase is zero, then also the bar graph will show zero.

COLOR GROUPING

In the right lower window region, you will find a switch for activating or deactivating the color grouping function (Figure 17).

Fig. 17: *Activating or deactivating the color grouping function*

With this function arbitrary colors of the color table can be pooled to groups. In this way complex class or tolerance boundaries can be realized. After activating the function in the color table, an additional column appears to indicate the group index. Colors with the same group index are mapped to the same sensor output correspondingly to the adjusted “RESULT FORMAT“ mode.

SAVING PARAMETERS
There are two switches in the field “SAVE PARAMETERS“ for saving the current parameter settings.

The switch “TO SENSOR“ stores all parameters into the built-in nonvolatile Flash-memory of the sensor. The parameters will remain in the Flash-memory after power down.

The switch “TO FILE“ stores all parameters on a memory drive of a computer that is connected to the sensor system.

LOADING PARAMETERS
The switch “FROM FILE“ in the field “LOAD PARAMETERS“ loads a parameter set into the sensor system.

Note:
All parameter values from the sensor system will be automatically loaded when the program is started.

1.2.3 Teaching In Colors

The so-called “Teach-in“ procedure for sampling and storing colors into the color table is done with the tab “TEACHING IN COLORS“ of the program window (Figure 18).

COLOR SPACE MODE
The currently selected color mode is shown in the field “COLOR SPACE MODE“ in the left hand upper area of the program window.

PROCESSING MODE
The currently selected processing mode is shown in the field “PROCESSING MODE“ below the field “COLOR SPACE MODE“..

CURRENT COLOR VALUES
This field shows the current sensor data (color coordinates) as they are transformed within the current selected color space. Beneath these values the current color is shown as a visible color sample (a valid reference white is the basic precondition for proper color representation).

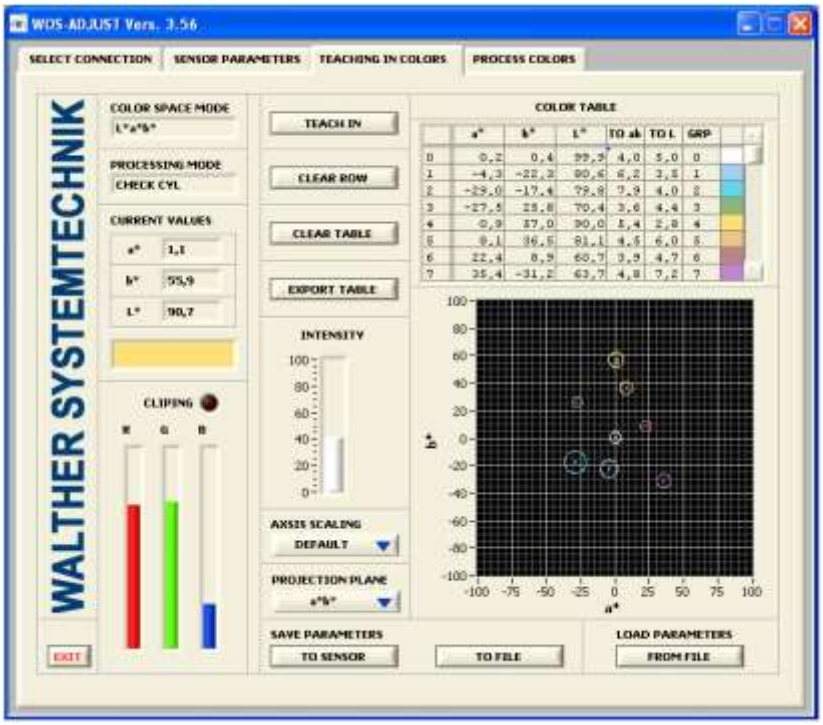


Fig. 18: *Window for entering color sample values*

COLOR TABLE
The color table is located on the right hand upper area of the program window. Its function is to store colors for recognition or classification. On the left hand side buttons for sampling and erasing colors are located.

The button “TEACH IN“ saves the current color space values into the next free row of the color table. For **overwriting** a row of the color table the PC cursor must be set into the corresponding row. After that, the button “TEACH IN” has to be pressed.

The button ”CLEAR ROW“ clears all color values of one row which has been selected by a mouse click. The then remaining data rows move up.

The button “CLEAR TABLE“ clears all entries in the color table.

EXPORT TABLE
The button “EXPORT TABLE” stores the current contents of the color table into a comma separated spreadsheet file (.csv) onto the disk of the PC. The file can be easily processed with common spreadsheet software.

AXIS SCALING
Use this selection switch to choose between 3 scaling methods (Figure19).

Fig. 19: *Scaling options*

“MANUAL“ allows manual editing of the diagram axes (by marking and editing the numbers on the axes). “DEFAULT“ resets the axes to default values. The setting “AUTO“ activates an auto scaling of the axes.

PROJECTION LEVEL
The X and Y axes of the **color diagram** on the right hand side can be selected in field “PROJECTION PLANE“ in order to display the desired color plane out of all three possible 2-dimensional planes of the 3-dimensional color space. All colors and tolerances can be completely viewed by this projection plane. **For fitting the color diagram scales to the desired range, the axes can be adjusted by editing the numbers on the axes**.

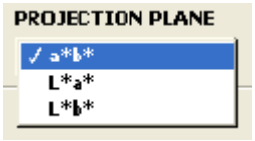


Fig. 20: *Selecting the plane for the color diagram*

All values stored in the color table (except the index row) can be manually modified. By clicking on a table row the input mode becomes active and numbers can be simply modified by using the keyboard. By entering the RETURN key or clicking on another area within the program window, the modified values will be stored into the color table. There is a scroll bar on the right hand side at the table for scrolling the table up and down.

COLOR TABLE

	a*	b*	L*			GRP
0	0,2	0,4	99,9			0
1	-4,3	-22,3	80,6			1
2	-29,0	-17,4	79,8			2
3	-27,5	25,8	70,4			3
4	0,9	57,0	90,0			4
5	8,1	36,5	81,1			5
6	22,4	8,9	60,7			6
7	35,4	-31,2	63,7			7

Figure : classify mode

COLOR TABLE

	a*	b*	L*	TOL		GRP
0	0,2	0,4	99,9	4,0		0
1	-4,3	-22,3	80,6	6,2		1
2	-29,0	-17,4	79,8	7,9		2
3	-27,5	25,8	70,4	3,6		3
4	0,9	57,0	90,0	5,4		4
5	8,1	36,5	81,1	4,5		5
6	22,4	8,9	60,7	3,9		6
7	35,4	-31,2	63,7	4,8		7

Figure : recognition with spherical tolerances

COLOR TABLE

	a*	b*	L*	TO ab	TO L	GRP
0	0,2	0,4	99,9	4,0	5,0	0
1	-4,3	-22,3	80,6	6,2	3,5	1
2	-29,0	-17,4	79,8	7,9	4,0	2
3	-27,5	25,8	70,4	3,6	4,4	3
4	0,9	57,0	90,0	5,4	2,8	4
5	8,1	36,5	81,1	4,5	6,0	5
6	22,4	8,9	60,7	3,9	4,7	6
7	35,4	-31,2	63,7	4,8	7,2	7

Figure : recognition with cylindrical tolerances

COLOR TABLE

	a*	b*	L*			GRP
0	0,2	0,4	99,9			0
1	-4,3	-22,3	80,6			1
2	-29,0	-17,4	79,8			1
3	-27,5	25,8	70,4			2
4	0,9	57,0	90,0			3
5	8,1	36,5	81,1			3
6	22,4	8,9	60,7			3
7	35,4	-31,2	63,7			4

Figure : grouping of colors

The assigned columns depend on the selected processing mode and upon the activation of the grouping function (see figures above).

There are no tolerance target values needed in the classification mode. The recognition mode needs target values: one parameter, spherical radius (= column TOL in figures above), is required for spherical tolerances and two parameters, color (column 'TO ab' in figure above) and brightness (column 'TO L' in figure above) tolerance, are required for cylindrical tolerances. The latter is preferred in applications where the color brightness plays a less important role than the actual color. If the 'TO L' tolerance is set to a high value. the influence of the color brightness is correspondingly low.

In the right column "GRP" a group index (0…255) can be assigned if the grouping function is activated. The assigned index is encoded according to the adjusted output format ("RESULT FORMAT"). Equal group index numbers activate the same sensor output. In this way different colors can be mapped to the same output (grouping).

The color diagram is located in the right lower area of the program window.

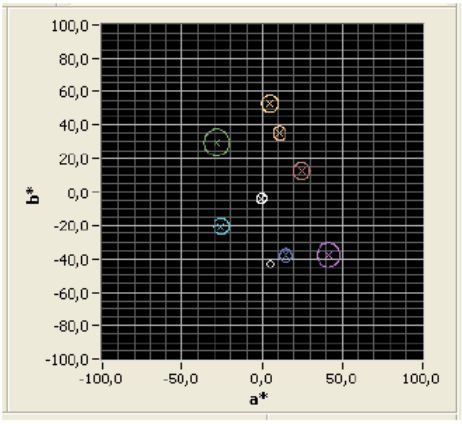


Figure : Color diagram with tolerance cylinders in the a*b* plane

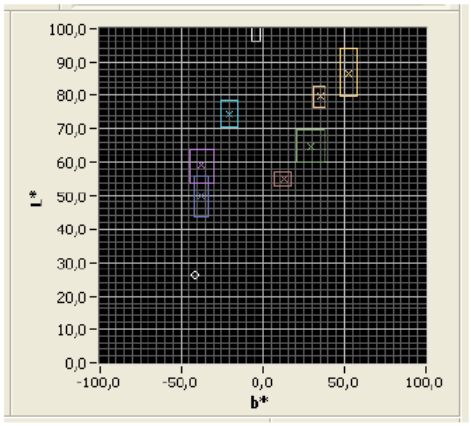


Figure : L*b* plane

The figures above show an example color diagram in the L*a*b* color space in the processing mode with cylindrical tolerances.

The sample colors in the right column of the color table correspond to the colors resulting from the color values/coordinates in each row. The displayed color complies with the real color only if a valid reference white was successfully sampled before. The visible colors in the color table are also used for displaying the tolerance boundaries in the color diagram and hence support the user at defining the tolerance boundaries in the diagram.

Note 1:
Although tolerance rooms/boundaries may randomly overlap, the color inspection and classification is done always uniquely. The order of colors stored as rows in the color table does not affect in any way the color inspection.

Note 2:
The tolerance parameters are used as ΔE-like units. Table shows how the human color perception commonly recognizes color variations in the L*a*b* color space. Due to the dependencies of the obtained color values on the used illumination source and the accuracy of the sensor, the table only serves as an orientation. Practical tolerance values must be found individually for the sensor.

Table 9: *Common values of human perception of color deviations*

color variation ΔE	human perception
<1	very small color variation that can not be seen by the human eye
1...2	small color variation that can be seen by trained human eye
2...3,5	medium color variation that can be seen by average human eye
3,5...5	considerable color variation
>5	high color variation

Note 3:
If the spherical tolerances processing mode is selected, the tolerance circles become ellipses in some projection planes due to different scaling of the diagram axes. This is, however, merely a displaying effect.

SAVING PARAMETERS
The switch “TO SENSOR“ stores all parameters into the built-in nonvolatile Flash-memory of the sensor. The parameters will remain in the Flash-memory after power down. The switch “TO FILE“ stores all parameters on a memory drive of a computer that is connected to the sensor system.

LOADING PARAMETERS
The switch “FROM FILE“ of the field “LOAD PARAMETERS“ loads a parameter set into the sensor system.

Note:
The buttons in the fields “SAVE PARAMETERS“ and “LOAD PARAMETERS“ are located in each program tab and they all perform the same function. An entire parameter set is always stored or loaded. After starting the program all parameters are read out from the built-in Flash-memory and are displayed in the program.

1.2.4 Color Processing

Figure 21 shows the program window for color recognition and classification which can be found under the tab “PROCESS COLORS“.

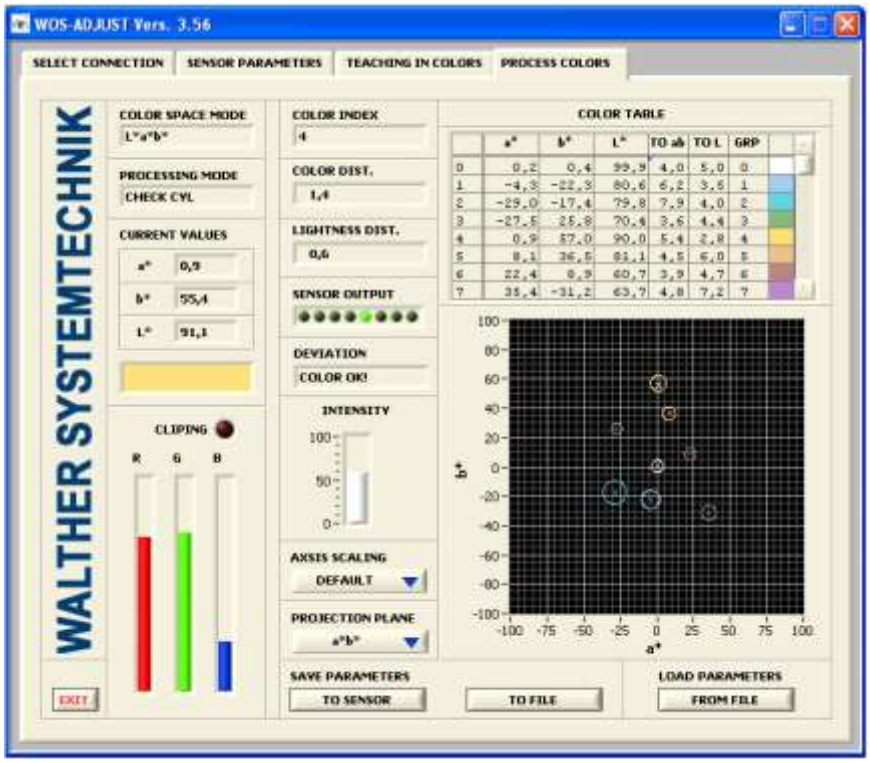


Fig. 21: *Program window for color recognition and classification program*

The window arrangement and operation is similar to the program window “TEACHING IN COLORS“ except for the three buttons on the left hand side of the color table which are substituted by three number boxes.

COLOR NUMBER/INDEX
The color number of the recognized or classified color is shown in the field ”COLOR INDEX“ and represents the result of the entire sensor system’s signal processing path. This number is equivalent to the row number in the color table. If in the recognition (checking) processing mode the tolerance boundaries were exceeded, the color number becomes 255. In addition the result is also available at the sensor system’s communication interface.

COLOR DISTANCE
The Euclidean distance between the actual measured color and the recognized or classified (matched) color value is displayed in the field ”COLOR DIST“. If the cylindrical tolerance mode is selected the Euclidean distance becomes a 2-dimensional distance vector (2-dimensional color vector). All other processing modes show a 3-dimensional Euclidean distance vector.

Note:
If in the processing modes “CHECK CYL“ and “CHECK SHP“ the tolerance room was exceeded or the color was not recognized (index 255), the distance to the next closest color is calculated and shown in the field ”COLOR DIST“.

LIGHTNESS DISTANCE
The field ”LIGHTNESS DIST“ shows the magnitude of the brightness distance in the cylindrical tolerances processing mode (equal to an “altitude difference” in the color space).

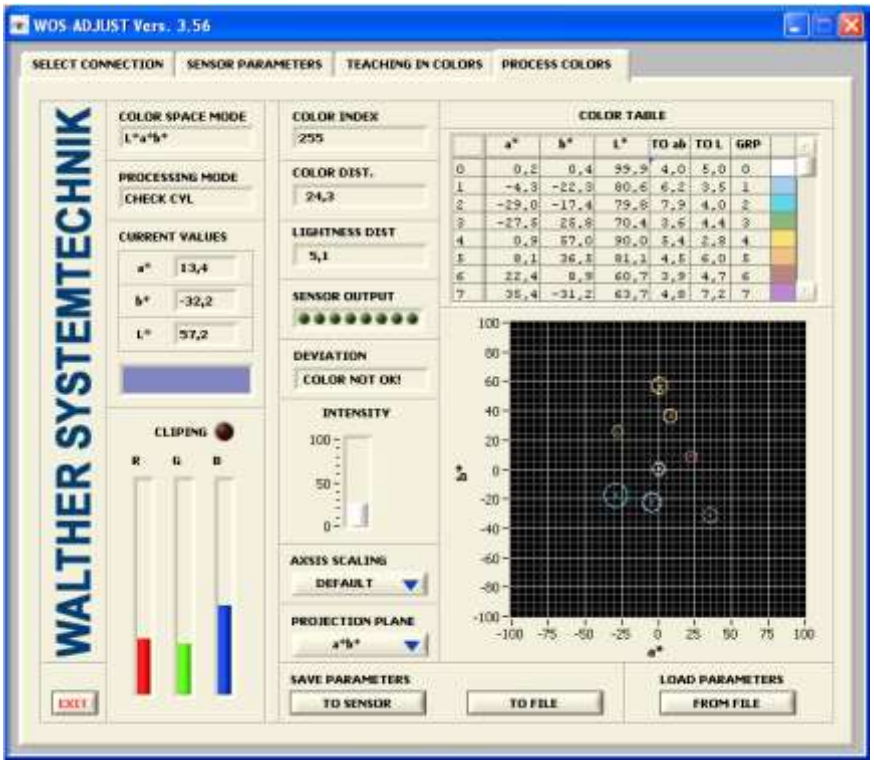


Fig. 22: *Program window for: tolerance exceeded or color not recognized*

The color distance values can be read out by the RS232 or USB interface.

Note:
If in the processing mode “CHECK CYL“ the tolerance room was exceeded or the color was not recognized, respectively, the brightness distance to the next closest color is calculated and displayed in the field ”LIGHTNESS DIST“.

SENSOR OUTPUT
This display signalizes the switching states of the sensor outputs corresponding to the „RESULT FORMAT“ and „COLOR GROUPING“ settings*“.*

DEVIATION
If the result format “DEVIATION” is selected, this display interprets the sensors outputs and displays the color-deviation in plain text. If one of the other result modes is selected, this display just shows whether a color is detected or not.

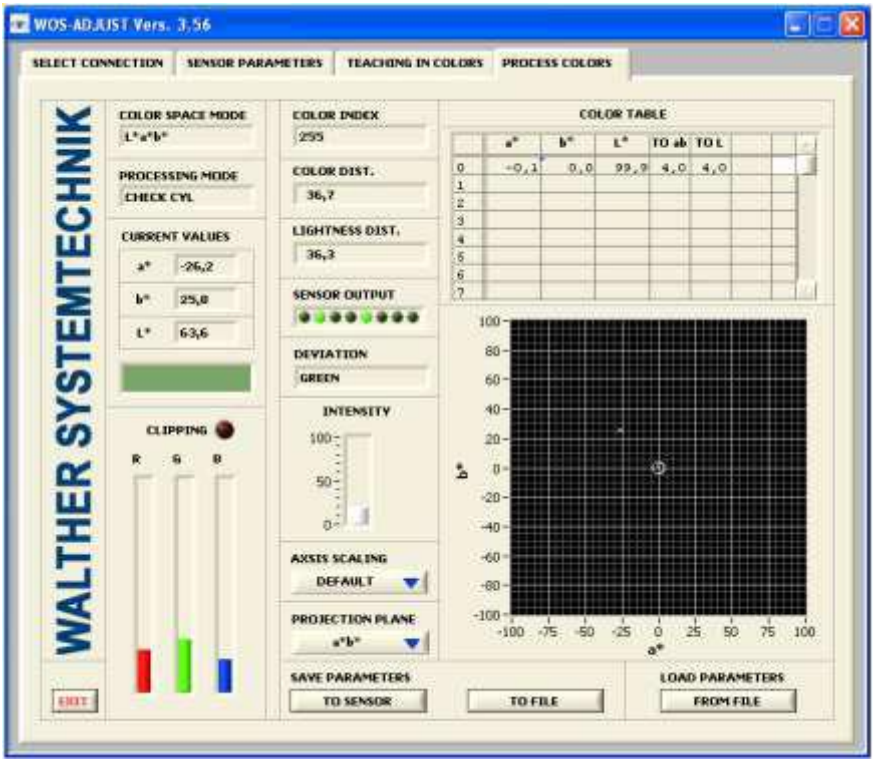


Fig. 23: *Display of deviation in DEVIATION mode*

AXIS SCALING
With this selection switch, 3 scaling methods can be chosen (ref. to above section).

1.2.5 Service

With the switch the field “SERVICE“ at the startup-screen a service tool is started, which allows certain special hardware functions and settings. Figure 24 shows the program window.

Fig. 24: *“SERVICE“ program window*

KEY LOCK
Use this switch to lock or unlock the sensor keys respectively.

GET SENSOR ID
With “GET ID“ the device serial number and version of the sensor firmware is read from the device and displayed on the right textbox.

SENSOR ADDRESS
With “GET ADDRESS“ in the area of “SENSOR ADDRESS“ the programmed address of the sensor is read out and displayed on the right textbox. Underneath this switch you will find a field to set the particular sensor address (scope: 0…255). With “SET ADDRESS” the sensor is programmed to listen to the chosen address.

BAUD RATE
With the “SELECT BAUD RATE“ switch, the sensor communication is reconfigured to the chosen baudrate.

OFFSET CORRECTION
Press this button to start a routine which ascertains and corrects the sensor offsets. Offsets are the sensors dark signals that are noticeable in the active mode for self-shining objects. For correct operation, the hole of the sensor light input must be darkened.

FACTORY SETTINGS
With the „LOAD FACTORY SETT.” switch the factory settings are loaded into the RAM of the sensor.

Auto Gain Level
Use this control to change the factory preset level of 70%.

Button Tolerances
With these controls the default tolerance values for the 5 possible steps can be changed. The assignment to the 5 blinking impulses is as follows.

Tolerance Step (T.S.)	Blinking impulses	Factory Tolerance Value
0	1	3
1	2	6
2	3	9
3	4	15
4	5	20

The left tolerance values operate in the processing mode “CHECK SPH”. In the operating mode “CHECK CYL.” the right tolerance controls are additionally active. The can be used to change the brightness tolerances.

To adopt the adjusted tolerances the “SET” button must be pushed.

Note:
In the processing mode “CLASSIFY” the tolerance controls are inactive.

EXT. TEACH BEHAVIOUR
There are different ways of using the running mode “EXT. TEACH” (see also 1.2.2). If the checkbox “AUTO INCREMENT” is activated, the new color will be added as new entry of the internal color table.
If it’s disabled, the last valid color-value will be overwritten.

The function “KEEP PREVIOUS TOLERANCES” is only useful with disabled “AUTO INCREMENT”. It overwrites the color-values and keeps the old tolerance-values of the table entry.
If this checkbox is disabled, default tolerances are used.

The button “SET” sends the new properties to the sensor.

SETTING HYSTERESIS
This function guarantees a stable operation of the sensor system. It is recommended to chose a high hysteresis value for poor signal quality in order to avoid instability (‘flutter’) in the signal processing. The value for hysteresis can be set by pressing “SET HYSTERESIS“. The value adjusted in the field “HYSTERESIS VALUE“ will be set. The values are percentages of the pre-defined tolerances in the color table (refer to respective section).

OUTPUT BEHAVIOUR

„Hold time (ms)“
This function sets a hold time for the sensor outputs. Using the running modes “CONTINOUS” and “EXT.TEACH” this function holds the output-values for the selected number of milliseconds. The max. adjustable time is 65535 ms. With the Selector C.S. the corresponding color channel can be adjusted.

„Fall off.“
If this function is used with an external triggered mode and the checkbox “Fall off” is enabled, the results fall back to zero after the expiration of the desired number of milliseconds.

„CLK out modes“
With this switch the output function of the CLK output (PIN 5 on SB1) can be determined.

Mode	Function
„User“	Can be set with command 0x73
„CLK Out“	Output of Illumination clock
„Color Out“	Additional color channel

SAVING TO SENSOR
With this switch the chosen settings will be stored in flash memory permanently.

Note:
If the program is ended without storing the settings to flash memory permanently, all information is lost after a restart.

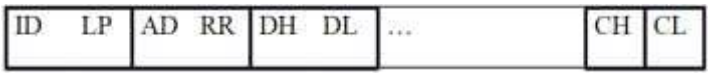
2 Command Reference and Protocol

The sensors can be fully controlled via RS232 or USB9. All commands are also available as LabVIEWR VIs and DLL function library. All software is shipped with the sensor system.

2.1 Communication Protocol Structure

There is a uniform protocol for sending and receiving data, respectively. The smallest communication unit is a byte (8 bits), however two bytes are always combined to one word and transferred together:

Protocol structure



The high-order byte is followed by a low-order byte. After sending a command word and receiving by the sensor system, a status response including error codes is sent back after executing the command.

Table 10: *Protocol set-up*

Word	Byte	Name	Description
0	0	ID	Command ID, higher 8 Bit of word 0
0	1	LP	Payload Length in Words, lower 8 bit of word 0
1	2	AD	Programmable Sensor Address, higher 8 bit of word 1
1	3	RR	Reserved (always 0), lower 8 bit of word 1
...	...	DH	Data, higher 8 bit
...	...	DL	Data, lower 8 bit
N	...	CH	Frame checksum, higher 8 bit
N	...	CL	Frame checksum, lower 8 bit

The payload length is limited to 255 words. Bigger frames will lead to unpredictable behavior of the device.

Every sensor can have an 8 bit wide address (AD). The sensor will only respond to commands with this exact address but always to commands with address '0'. This address can be set by command 0x11.

The checksum is a simple addition of all transmitted 16 bit words including the header. After adding all values only the lower 16 Bit will be taken.

Table 11: *Response values and codes*

Response	Data Value	Description
0xF0	dep. on com.	Commando executed correctly
0xF1	0x00-0x03	Internal status
	0x04	Frame: wrong checksum
	0x05	Frame: wrong length
	0x06	Frame: receive buffer not ready
	0x07	Frame: unknown error (fatal)
	0x08	Frame: timeout (1s)
0xF2	0x00	Illegal command
0xF3	0x00	Command not enabled
0xF4	0x00	Invalid parameter
0xE0	dep. on com.	Burst mode

If a frame is not transmitted completely within one second, the sensor responds with a timeout code (0xF1, Payload: 0x08)

Communication sequence example 1:
Setting the light intensity to 0x20*: „2401 0000 0020 2421"*
Response: *"F000 0000 F000"* (no error)

Communication sequence example 2:
Reading actual value of the light intensity: *„2500 0000 2500"*
Response: *"F001 0000 0020 F021"* (no error)

2.1.1 Burst Mode

For getting values from the sensor system as fast as possible without querying them, a burst mode has been implemented. When a burst mode is set by the burst mode command (0x12), this initial command is acknowledged as usual. After that, the system will send frames of the following structure continuously, until stopped by another burst mode command, switching the burst mode off again.

Burst Frame Structure

ID LP	AD RR	TH TL	DH DL	…	CH	CL

Table 12: *Burst Frame set-up*

Word	Byte	Name	Description
0	0	ID	Burst Mode response Code, always "0xE0"
0	1	LP	Payload Length in Words, lower 8 bit of word 0
1	2	AD	Programmable Sensor Address, higher 8 bit of word 1
1	3	RR	Reserved (always 0), lower 8 bit of word 1
2	4	TH	Timing information, higher 8 Bit of word 2
2	5	TL	Timing information, lower 8 Bit of word 2
…	…	DH	Begin of burst data. Format according to requested mode.
…	…	DL	Begin of burst data. Format according to requested mode.
N	…	CH	Frame checksum, higher 8 bit
N	…	CL	Frame checksum, lower 8 bit

The burst frame has the same structure like all other frames. Directly after the header a timing counter (TH/TL) is inserted. This counter indicates the time between two successive burst frames in µs. After that, the requested data follows as if it was the payload of the requested burst mode.

Other commands received by the system during a burst will be accepted and acknowledged accordingly.

Burst Mode Example:
Step 1: activate Burst Mode (2F-Mode):
"*1201 0000 0001 1202*". Response: *"F000 0000 F000".*

Step 2: receive Burst Frames:
"*E00A 0000 22D2* ***0112 00F9 0075 0EED 0F06 0F8A 0FFF 0FFF 0FFF*** *62D6".*
The system will go on sending frames similar to the above. The delay between this and the last frame is 8.914ms (0x22D2). The bold values represent the payload as if requested by a 0x2F-command.

Step 3: stop Burst Mode:
"1201 0000 0000 1201" : Response (after the last Burst Frame) *"F000 0000 F000"*

2.1.2 Fixed Point Format

In many cases it is necessary to use a more precise numbering system than the 16-bit integer number. For this reason the sensor system uses a 32-bit fixed point format number which has the following structure:

Table 13: *Fixed-Point Format (two's complement)*

Bit 31-21	Bit 20-0
Integer part	Fractional part

This format has a range +/-1024 with a resolution of 0.000 000 477. The representation corresponds to the two's complement. The conversion is described by the following formula:

$$F_{fixed} = F_{real} \cdot 2^{21}$$

Conversion Example 1: from Float to Fixed Point
a) Number: +1.123
Because the value is positive, the formula can be used directly.

$$1.123 \cdot 2^{21} = 2355101.696 \rightarrow 2355102 = 0x0023EF9E$$

b) Number: -1.123
Because the value is negative, the above result has to be converted to a negative two's complement value by inverting and adding "1":

0x0023EF9E → 0xFFDC1061 → 0xFFDC1062

Conversion Example 2: from Fixed Point to Float
a) Number: 0x0023EF9E
As the most significant bit is "0", this is a positive value. Simply convert by using the above formula.

$$\frac{0x0023EF9E}{2^{21}} = \frac{2355102}{2^{21}} = +1.123000$$

b) Number: 0xFFDC1062
As the most significant bit is "1", this is a negative value. Before conversion, invert and add "1":

0xFFDC1062 → 0x0023EF9E = 2355102 → -1.123000

Important:
Please pay attention to the word order of the low and high word of the 32 bit number. First, the lower part (Bit 15-0) is transmitted, then the higher part (Bit 31-16.

Communication example:
Setting reference white to 0.9/0.8/0.75 (command 0x42)
0.90 → 0x001CCCCC
0.80 → 0x00199999
0.75 → 0x00180000

→ *"4206 0000 CCCC 001C 9999 0019 0000 0018 A6B8"*

2.2 Command Reference

The following section describes all available commands. Data types other then 16 Bit integer (a word) and 32 Bit fixed point are described at the command description directly.
The leading hex-number represents the command id, the rest of the header (for example the length) and the checksum have to be set appropriately to build a complete communication frame.

2.2.1 System Identification

0x01: ID-String
Response: ASCII-String with 16 Bit character width

Note:
As the communication is 16-Bit based, only the lower 8 Bit of each word contain ASCII code. The higher 8 Bit are always zero.

0x02: Device production date
Response: 2 words

Word 1: Bit 7-0:	day
Word 1: Bit 15-8:	month
Word 2:	year

Note:
To make the date readable the value must be interpreted hexadecimal.

0x03: Device type
Response: 1 word

Bit 15-0: device type

0x04: Hardware revisions
Response: 2 words

0x05: Device software revision
Response: 1 word

Bit 15-8: major revision
Bit 7-0: minor revision

0x06: Device unique number
Response: 2 words

Word 1: device number (high)
Word 2: device number (low)

2.2.2 Communication

0x10: Set baudrate
Send: 1 word

0: 9.600
1: 14.400
2: 28.800 (default)
3: 38.400
4: 57.600
5: 115.200

0x11: Set sensor address
Send: 1 word

Note:
Note: Values other than 0-255 are ignored.

2.2.3 Data Acquisition

0x20: Set acquisition mode
Send: 1 word

0: no acquisition
1: active light internal triggered
2: passive light internal triggered
3: active light external triggered
4: passive light external triggered
5: active light external triggered inverted
6: passive light external triggered inverted

0x21: Read acquisition mode
Response: 1 word (as 0x20)

0x22: Set transimpedance amplification
Send: 1 word

0: Amplification factor 1
1: Amplification factor 5
2: Amplification factor 25
3: Amplification factor 100

0x23: Read amplification
Response: 1 word (as 0x22)

0x24: Set illuminating intensity
Send: 1 word

Note:
Values other than 0-255 are ignored.

0x25: Read illuminating intensity
Response: 1 word (as 0x24)

0x26: Set light compensation frequency
Send: 1 word

Note:
This value represents the time between bright and dark sample phases. The value 0xFFFF represents a frequency of 100Hz, a value of 0x0270 a frequency of 10kHz. Values lower than 0x1D0 (>15kHz) may not give the expected results. A value of 0x0270 is approx. 100µs.

.
0x27: Read light compensation frequency
Response: 1 word (as 0x26)

0x28: Automatic gain set-up
Response: 2 words

Word 1:	autom. found gain
Word 2:	autom. found illumination intensity

0x29: Set average for cycles
Send: 1 word

0x2A: Read average for cycles
Response: 1 word (as 0x29)

0x2D: Read raw color values (ADC channels)
Response: 4 words

Word 1:	[X] red value
Word 2:	[Y] green value
Word 3:	[Z] blue value
Word 4:	reserved / temperature sensor

Note:
Ranges are 0x0000 - 0xFFF0. The values are 12 Bit, left justified. 0xFFF0 represents the darkest, 0x0000 the brightest intensity (inverted).

0x2F: Read compensated color values (ADC channels)
Response: 9 words

Word 1:	[X] red compensated
Word 2:	[Y] green compensated
Word 3:	[Z] blue compensated
Word 4:	[X] red value (Phase I)
Word 5:	[Y] green value (Phase I)
Word 6:	[Z] blue value (Phase I)
Word 7:	[X] red value (Phase II)
Word 8:	[Y] green value (Phase II)
Word 9:	[Z] blue value (Phase II)

Phase I: light phase
Phase II: dark phase

Note:
Ranges are 0x0000 - 0x0FFF. The values are averaged according to 0x29. 0x0000 represents the darkest, 0x0FFF the brightest value).

2.2.4 Sensor Correction

0x30: Set matrix correction mode
Send: 1 word

0: off
1: turn on sensor level correction
2: turn off sensor matrix correction

0x31: Read matrix correction mode
Response: 1 word (as 0x30)

0x32: Set correction matrix
Send: 9 x Fixed Point

Matrix formula:
Xc = a00*R + a01*G + a02*B
Yc = a10*R + a11*G + a12*B
Zc = a20*R + a21*G + a22*B

Sending order:
a00, a01, a02, a10, a11, a12, a20, a21, a22

Note:
The values of the correction matrix are only limited within the range of the fixed point numbers, but to let the system perform reasonable color transformations, the result of each channel should be between 0.0-1.0. The settings can be verified by using command 0x3F.

0x33: Read correction matrix
Response: 9 x Fixed Point (as 0x32)

0x3F: Read corrected color values (channels)
Response: 3 x Fixed Point

Fixed Point 1: corrected X value (red)
Fixed Point 2: corrected Y value (green)
Fixed Point 3: corrected Z value (blue)

2.2.5 Color Transformation

0x40: Set color space mode
Send: 1 word

0: XYZ
1: xyY
2: L*u'v'
3: L*a*b*
4: L*u*v*
5: Lab99

0x41: Read color space mode
Response: 1 word (as 0x40)

0x42: Write reference white value
Send: 3 x Fixed Point

Fixed Point 1: X reference
Fixed Point 2: Y reference

Fixed Point 3: Z reference

0x43: Read reference white value
Response: 3 x Fixed Point (as 0x42)

0x44: Record reference white value and save
Response: 3 x Fixed Point (as 0x43)

Note:
This command takes the current input light value and sets this values as reference white.

0x4F: Read transformed color values (channels)
Response: 3 x Fixed Point

Table 14: *Sequence and ranges in different transformation modes:*

Mode	Read order	Range 1	Range 2	Range 3
0	X, Z, Y	0-1	0-1	0-1
1	x, y, Y	0-1	0-1	0-1
2	u', v', L*	0-1	0-1	0-1
3	a*, b*, L*	+/- 5	+/- 2	0-1
4	u*, v*, L*	+/- 5	+/- 5	0-1
5	a99, b99, L99	+/-5	+/- 2	0-1

2.2.6 Color Table, Color Recognition/Classification

0x50: Write color value table entry
Send: 9 x Fixed Point

Fixed Point 1: Entry number (integer as Fixed Point)
Fixed Point 2: Value 1
Fixed Point 3: Value 2
Fixed Point 4: Value 3
Fixed Point 4: Tolerance 1 (Quadratic value of tolerance in spherical & cylindrical mode)
Fixed Point 5: Tolerance 2 (only cylindrical mode, corresp. to lightness distance)
Fixed Point 7: Raw value 1 (X)
Fixed Point 8: Raw value 2 (Y)
Fixed Point 9: Raw value 3 (Z)

Note:
This command writes a complete entry to the 255 entries long color table. The first value is the index of this entry. The next three entries are the transformed color values of the color to be recognized. The two tolerances depend on the classification mode. Please see chapter 1.2.4 for more detailed information. The last three values are the corresponding raw values to the transformed values. A more practical way to create a new entry is to use command 0x56.

0x51: Read color value from color table
Send entry number: 1 x Fixed Point

Response:

Fixed Point 2: Value 1
Fixed Point 3: Value 2
Fixed Point 4: Value 3
Fixed Point 4: Tolerance 1 (Quadratic value of tolerance in spherical & cylindrical mode)
Fixed Point 5: Tolerance 2 (brightness tolerance in cylindrical mode)
Fixed Point 7: Raw value 1 (X)
Fixed Point 8: Raw value 2 (Y)
Fixed Point 9: Raw value 3 (Z)

0x52: Set processing / classification mode
Send: 1 word

Bit 15-8: Table size (number of valid entries)
Bit 7-0: 0: no classification
1: spherical check (with radius check)
2: cylindrical check (with radius check)
3: spherical classification (without radius check

0x53: Read processing / classification mode
Response: 1 word (as 0x52)

0x54: Set hysteresis
Send: 1 word (value in percent)

Note:
This command sets a classification hysteresis (in percent) for filtering small disturbances within the color recognition. The value enlarges the radius of the actual recognized color and makes it more difficult to leave.

0x55: Read hysteresis
Response: 1 word (as 0x54)

0x56: Recording color table entry and saving
Response: 8 x Fixed Point

Fixed Point 1: Value 1
Fixed Point 2: Value 2
Fixed Point 3: Value 3
Fixed Point 4: Tolerance 1 (Quadratic value of tolerance in spherical & cylindrical mode)
Fixed Point 5: Tolerance 2 (only cylindrical mode, corresp. to lightness distance)
Fixed Point 6: Raw value 1 (X)
Fixed Point 7: Raw value 2 (Y)
Fixed Point 8: Raw value 3 (Z)

Note:
This command takes the actual light input as entry and auto-increments the table-size. Tolerances are set by default to 4 units.

0x57: Set tolerance defaults
Send: 2x Fixed Point

Fixed Point 1: default value color tolerance = (value/100)2
Fixed Point 2: default value brightness tolerance = (value/100)

0x58: Read tolerance defaults
Response: 2 x Fixed Point

Fixed Point 1: default value color tolerance = (√value) * 100
Fixed Point 2: default value brightness tolerance = value * 100

0x5F: Read classified value
Response: 3 x Fixed Point

Fixed Point 1: Index of recognized/classified color (closest color)
Fixed Point 2: Quadratic value of distance 1 (spherical & cylindrical mode)
Fixed Point 3: Value of Distance 2 (only cylindrical mode, corresp. to lightness distance)

2.2.7 Output Encoding

0x60: Set encoding configuration
Send: 1 word

Bit 15-12: Output Mode
0 = [0000]: continuous
1 = [0001]: external triggered
2 = [0010]: trigger controlled sequence
3 = [0011]: timer controlled sequence
4 = [0100]: external triggered teach in
5 = [0101]: self triggered sequence

Bit 11-8: Mapping/grouping Mode
0: no mapping (direct)
1: mapping enabled (use 0x62 to set entries)

Bit 7: External Teach Mode: Auto increment enable
0: Auto increment disabled
1: Auto increment enabled

Bit 6: External Teach Mode: Keep previous tolerance values
0: Set default tolerances
1: Keep previous tolerances (only useful in combination with Bit 7)

Bit 5: External Trigger Modes: Automatic fall-off enable
0: Automatic fall-off disabled
1: Automatic fall-off enabled (set time with 0x68)

Bit 4: Encoding Polarity
0: not inverted outputs
1: inverted outputs

Bit 3-0: Encoding Mode
0 = [0000]: off (output pins remain unchanged)
1 = [0001]: direct, binary
2 = [0010]: 1-hot
3 = [0011]: 7-segment
4 = [0100]: color deviation vector

0x61: Read output mode configuration
Response: 1 Word (as 0x60)

0x62: Write group encoding map entry
Send: 2 words

Word 1: Table entry (table index) (range 0..255)
Word 2: Table value (group number) (range 0..255)

0x63: Read group encoding entries
Send: 1 word (table entry 0..255)
Response: 1 word (table value 0..255)

0x64: Set sequence timeout
Send: 1 word

Note:
Sequence Timeout for triggered mode in ms: 10^{-4}s (multiple of 100 μ s).

0x65: Read sequence timeout
Response: 1 word (as 0x64)

0x66: Set sequence interval
Send: 1 word

Note:
Time between two automatic samples for timed mode in ms.

0x67: Read sequence interval
Response: 1 word (as 0x66)

0x68: Set output hold time
Send: 1 word

Note:
This command sets the time for which the resulting output signal will be held constant on the output pins. The time is indicated in ms.

0x69: Read output hold
Response: 1 word (as 0x68)

0x6F: Read output (encoded) signals
Response: 1 word

2.2.8 Flash Operations and Parameters

0xA0: Saving parameters
Store all parameters from RAM to Flash.

Note:
Flash write operations can last up to several seconds. The device must not be interrupted and power supply has to be stable during this time. Otherwise the flash may be corrupted and the device may start with an unknown or even a dangerous configuration

0xA1: Read parameters
Copy stored parameters from Flash to RAM.

0xA2: Read factory settings
Copy factory setting into RAM

3 LabVIEW® VIs

All commands described in chapter 4 are implemented as LabVIEW VIs and are shipped with this color sensor system. The following VIs are saved as LabVIEW Version 6.1 libraries on CD:

Table 15: *LabVIEW libraries*

Library	Description
libpcs_cmd.llb	Main library with all commands
libtools.llb	Help library for libpcs_cmd. Internal, shared VIs.
libft.llb	Help library for controlling the USB module.
libpcs_com	general functions for communication

3.1 Structure of Command VIs

All command VIs have the same standard structure (Figure 25).

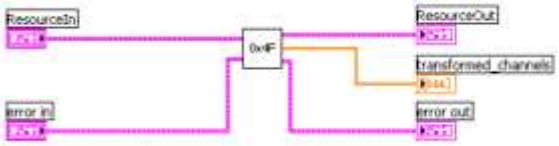


Fig. 25: *Connections of VIs*

Resource Signal
The **Resource** signal is a so-called cluster signal (Figure 26) and contains information and configuration about the communication interface.

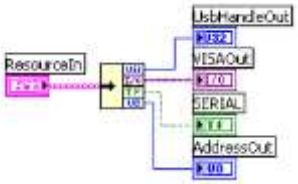


Fig. 26: *Resource cluster signal*

The signal **UsbHandleIn** contains the handle for the USB communication. The **VISAIn** signal stores all information for the serial communication interface. The communication channel is selected by the signal **SERIAL** (TRUE: RS232, FALSE: USB).

error Signal
The error signal is a cluster signal and contains error information (Figure 27).

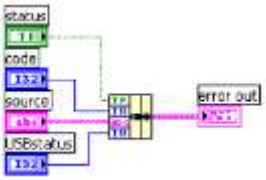


Fig. 27: *error cluster signal*

3.2 Open Resource VI and Close Resource VI

There are VIs for opening and closing the communication channel.

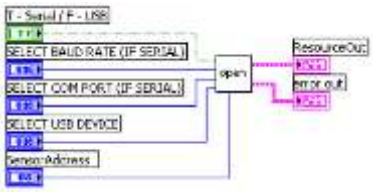


Fig. 28: *Open Resource VI*

The switch **T - Serial / F – USB** selects between RS232 or USB. With the selection switch **SELECT COM PORT (IF SERIAL)** the used COM-Port for the RS232 can be selected. Via the selection switch **SELECT BAUD RATE (IF SERIAL)** the used baud rate can be chosen.
With the control “SELECT USB DEVICE” the desired USB device can be chosen.

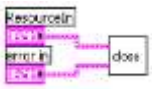


Fig. 29: *Close Resource VI*

For closing the used com port the VI **close resource** is provided.

3.3 Example

Figure 30 shows a block diagram for combination of LabVIEW VIs with the command 0x2F (Read Compensated ADC Channels.

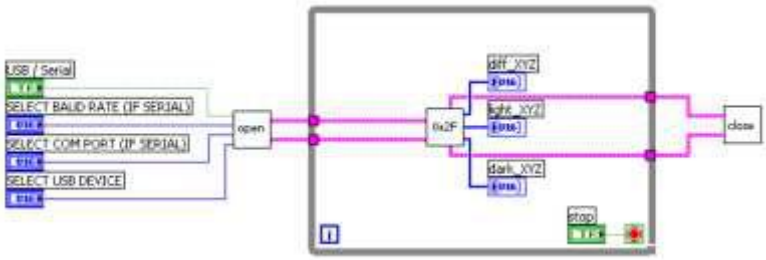


Fig. 30: *Combination with the command 0x2F (Read Compensated ADC Channels)*

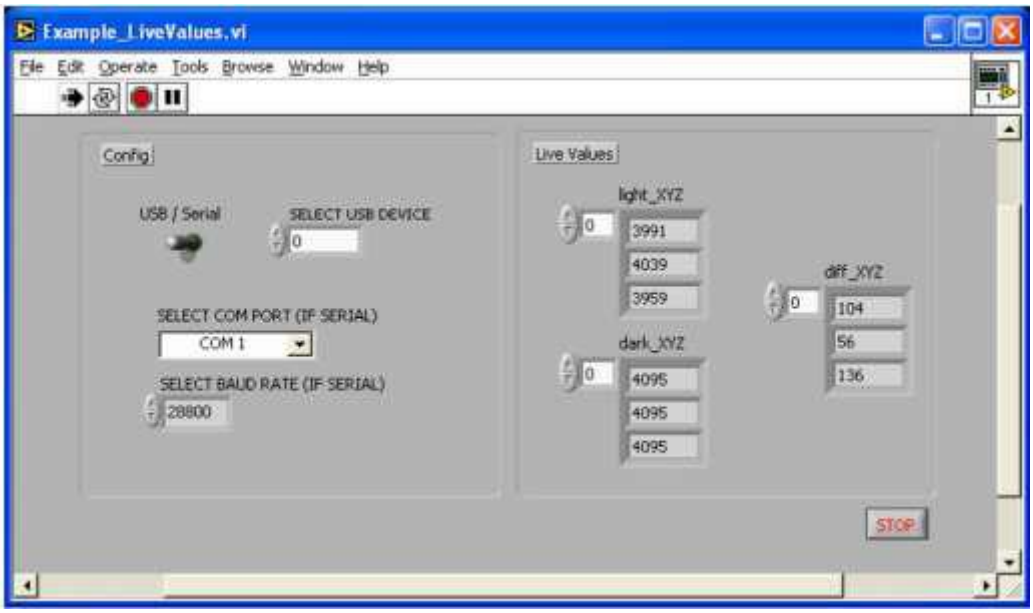


Fig. 31: *Front panel of the example VI for command 0x2F (Read Compensated ADC Channels)*

4 DLL Function Library

4.1 Introduction

The function library **PCSLib** contains the functionality to communicate and configure the sensors. For the communication two PC devices are supported: serial interface RS 232 (COM1, COM2, ...) and the USB interface.

This document describes the library functions, its usage and the delivered files.

4.2 Library Functions

Three function layers are supported:

- **PCS Library**:
  Information about the PCSLib (name, version number)
- **USB Library:**
  Initializing of the USB library
- **PCS devices:**
  Opening and closing of the interfaces and the communication

4.2.1 Library

PCS_VersionString
Delivers the PCSLib version string in the following format: *"PCSLIB 1.3"*

*UCHAR**
PCS_VersionString *(void);*

Parameter:
- none –

Return value:
string pointer to the **PCSLib** version string.

4.2.2 USB Library

PCS_initUSBLib
This function loads the USB library and initializes its functions. It is required to call this function, before additional USB functions can be used.

ULONG
PCS_initUSBLib *(void);*

Parameter:
- none –

Return value:

Status	0	successful initialization
	-1	if an error occurs

PCS_closeUSBLib
This function closes the USB library and frees the allocated memory.

void ***PCS_closeUSBLib*** *(void);*

Parameter:
- none –

Return value:
- none –

4.2.3 Devices

PCS_getUSBDeviceNum
Detects the number of sensors connected to the USB interface.

UCHAR
PCS_getUSBDeviceNum *(ULONG* num);*

Parameter:
num pointer which contains the device number.

Return value:
Communication error ***PCSCS_OK*** if command successful, otherwise return value has an error code.

PCS_openDevice
Opens a communication device (USB or RS232 interface) and initializes the internal device memory structure. Depending on the device type different configuration parameters are used.

UCHAR
PCS_openDevice (*struct PCS_Device_def* pcsdev, UCHAR devtype, UCHAR devnr, ULON baud);*

Parameter:
pcsdev pointer on device structure to be initialized
devtype device type: **PCSDT_USB, PCSDT_RS232**
devnr device number; depending on type (USB or RS232 device)
baud RS232 baudrate; default value: 28800 baud

Return value:
Communication error PCSCS_OK if command successful, otherwise return value has an error code

PCS_closeDevice
Closes a previously opened communication interface

void
PCS_closeDevice *(struct PCS_Device_def *pcsdev);*

Parameter:
pcsdev pointer to the device structure

Return value:
- none –

PCS_sendCommand
This function sends a command via the opened communication interface to the sensor and delivers the response.

UCHAR
PCS_sendCommand (struct PCS_Device_def pcsdev,*
UCHAR cmd, UCHAR devid,
USHORT txd, USHORT txd_len,*
USHORT rxd, USHORT* rxd_len,*
UCHAR responsecode);*

Parameters:
pcsdev pointer to device structure
cmd code of the command to be sent
devid device ID of the sensor for communication
0x00 (default) for broadcasting; all connected sensors will be addressed
txd pointer to send message
txd_len length of send message
rxd pointer to receive message
txd_len length of received message

responsecode response code

Return value:
Communication error **PCSCS_OK** if command successful; otherwise return value is error code.

Code example:

```
#include "libpcs.h"
...
USHORT      x, y, z;
UCHAR       devid = 0x00;                   // device id
USHORT      msg_tx[PCS_MAXMESSAGESIZE];     // transmit command data
USHORT      msg_rx[PCS_MAXMESSAGESIZE];     // received command data
USHORT      msg_tx_len, msg_rx_len;         // data length (rx, tx)
UCHAR       com_status;                     // communication status
UCHAR       responsecode;                   // protocol response code

msg_tx_len = 0;
com_status =    PCS_sendCommand(pcsdev, PCSCC_ReadRawADCChannels, devid, msg_tx, msg_tx_len, msg_rx,
                &msg_rx_len,&responsecode);

if (com_status != PCSCS_OK || responsecode != PCSRC_OK)
{
        printf("ReadRawADCChannels: communication error: %x, %x\n", com_status, responsecode);

} else {
        x = msg_rx[0];                  // value x
        y = msg_rx[1];                  // value y
        z = msg_rx[2];                  // value z
}
```

4.3 Files

File	Description
libpcs.dll *libpcs.h*	library and header file
Makefile	makefile for generating the example executables (in this case gcc with cygwin)
acquisition.c *acquisition.exe*	shows the sensor color values in Lab color space format
benchmark.c *benchmark.exe*	benchmark the USB and RS232 speed
identify.c *identify.exe*	reads the sensor identification string

4.4 Requirements

- Installed FTD2XX.DLL for USB communication
- Installed development environment