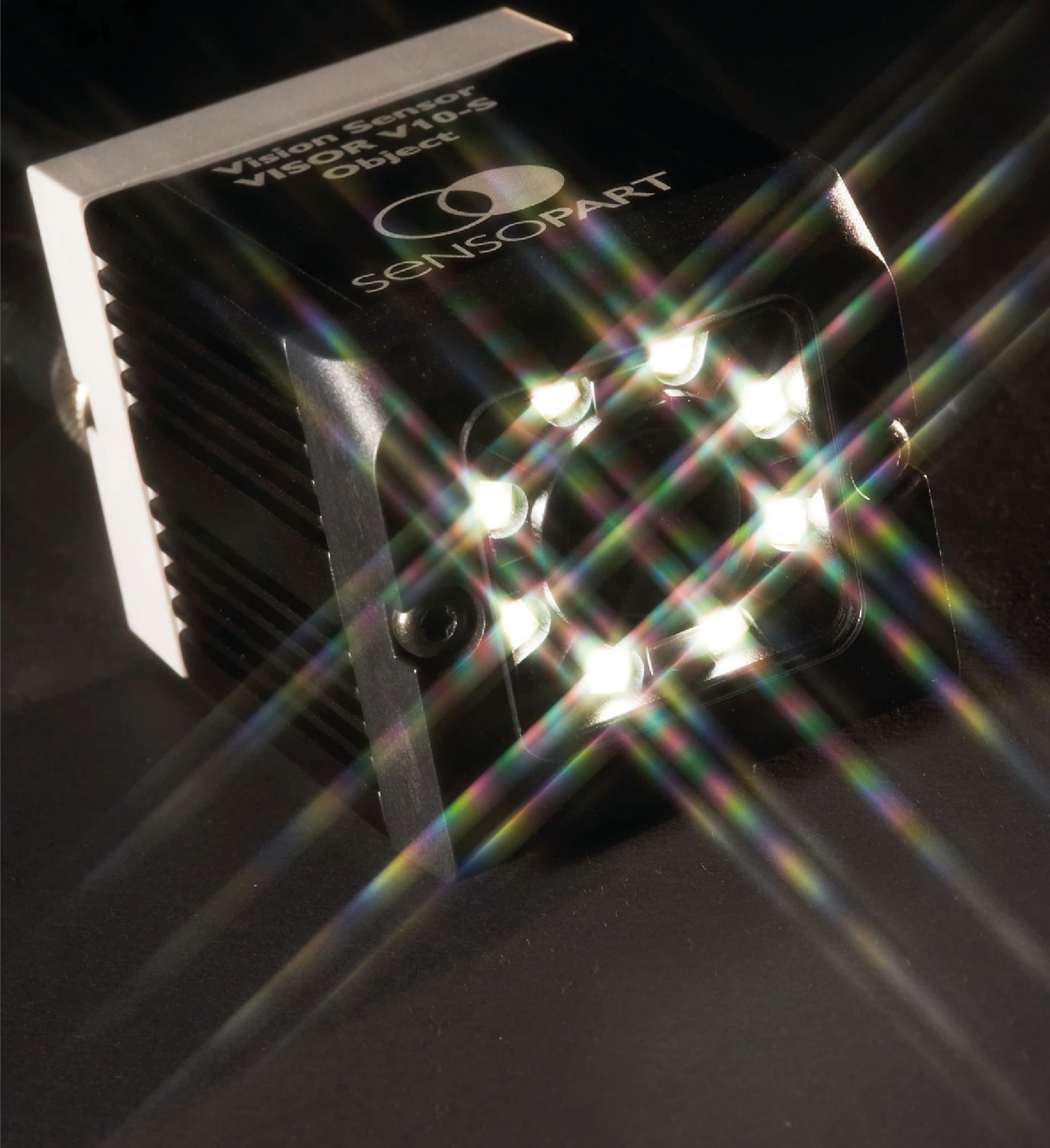
SENSOPART

VISOR® Benutzerhandbuch
VISOR® user manual
Manuel d‘utilisation VISOR®



Copyright (English)

We accept no responsibility for printing errors and mistakes which occurred in drafting these document. Subject to delivery and technical alterations.

First publication February 2011

SensoPart Industriesensorik GmbH

Nägelseestraße 16

D-79288 Gottenheim

IND. CONT. EQ.
77HA

For use in class 2 circuits

For use in NFPA 79 Applications only

Open Source Licences

The VISOR® software makes use of a couple of third party software packages that come with various licenses. This section is meant to list all these packages and to give credit to those whose code helped in the creation of the VISOR® software.

For components that reference the GNU General Public License (GPL) or the GNU Lesser General Public License (LGPL), please find these licenses and the written offer for source code in this software installation in \SensoPart \VISOR® vision sensor\Eula\OpenSourceLicenses.

The VISOR® firmware makes use of Linux Version 2.6.33 (Website: www.kernel.org), which is distributed under the GNU GPL version 2.

The VISOR® firmware makes use of x-loader, an initial program loader for Embedded boards based on OMAP processors (Website: http://arago-project.org/git/projects/?p=x-load-omap3.-git; a=summary) which is distributed under the GNU GPL version 2 or higher.

The VISOR® firmware makes use of u-boot, an initial program loader for Embedded boards based on OMAP processors (Website: http://arago-project.org/git/projects/?p=x-load-omap3.-git; a=summary) which is distributed under the GNU GPL version 2 or higher.

The VISOR® firmware makes use of spike Version 0.2, a SPI-driver (Website: https://-github.com/scottellis/spike/blob/master/spike.c), which is distributed under the GNU GPL version 2 or higher.

The VISOR® firmware makes use of Busy-Box Version 1.18.1 (Website: http://www.busybox.net/), which is distributed under the GNU GPL version 2 or higher.

The VISOR® firmware makes use of vsftpd Version 2.0.3 (Website: https://security.appspot.com/vsftpd.html), which is distributed under the GNU GPL version 2 or higher.

The VISOR® firmware makes use of mtd-utils Version 1.5.0 (Website: http://www.linux-mtd.infradead.org/doc/general.html), which is distributed under the GNU GPL version 2 or higher.

The VISOR® firmware makes use of Boa Webserver Version 0.94.13 (Website: http://www.-boa.org/), which is distributed under the GNU GPL version 2 or higher.

The VISOR® firmware makes use of Procps Version 3.2.8 (Website http://-procps.sourceforge.net/download.html), which is distributed under the GNU GPL version 2 or higher and GNU LGPL version 2.1 or higher.

The VISOR® firmware makes use of GnuPG Version 1.4.10 (Website: https://www.gnupg.org/), which is distributed under the GNU GPL version 3 or higher.

The VISOR® firmware makes use of glibc, which is distributed under GNU LGPL version 2.1 or higher.

The VISOR® firmware makes use of Dropbear - a SSH2 server Version 2012.55 (Website: https://matt.ucc.asn.au/ dropbear/dropbear.html). The Dropbear SSH2 server is distributed under the terms of the Dropbear License which is a MIT/X Consortium style open source license.

Please find this license in this software installation in \SensoPart \VISOR® vision sensor\Eula\OpenSourceLicenses.

SensoConfig software is based in part on the work of the Qwt project (http://qwt.sf.net).

Table of Contents

1 General Information and Safety

1.1 Safety notes

Before starting the VISOR® vision sensor, read these instructions carefully, ensure that you have understood them and comply with them at all times.
The VISOR® vision sensor should only be connected by a qualified electrician.
Do not tamper with or make alterations on the unit!
The VISOR® vision sensor is not a safety-critical component and its use is prohibited under conditions where the safety of persons may depend on its function.
The IP address set for the VISOR® vision sensor should be marked on the enclosed label. After installation, stick the label on the sensor in a clearly visible position.
The IP address of the VISOR® vision sensor must be used once only in any network.
For use with any listed (CYJV) cable assembly.

1.2 Components supplied

- VISOR® vision sensor including integrated illumination (or as version with C-Mount lens without illumination)
- CD-ROM with Computer software and Operating instructions
- Data sheet, mounting clamp, allen key, screwdriver and protective cap for Ethernet plug.

1.2.1 Software Setup Download

The PC Software Setup is also available at www.sensopart.com in section Download/Software.

In order to reduce the size of the installation files two setup files are provided:

1. VISOR®_PC-Software_VX_X_X_X.exe:
   This setup contains the current release version. Vision sensors with immediate previous firmware versions can be parameterized with this PC software. Earlier versions can be installed with the legacy setup. For further information, please refer to the corresponding download information.
2. VISOR®_PC-Software_VX_X_X_X_Legacy.exe:
   This setup serves as an extension and can be installed to the PC software to parameterize older vision sensors. For further information, please refer to the corresponding download information.

1.3 Requirements for use

Configuration of the VISOR® vision sensor requires a standard PC/notebook (at least Pentium 4, 1GHz and 1 GB RAM, with Microsoft Windows 7 or Windows 10) with network connection or a network with TCP-IP protocol. We recommend a Pentium 4 Dual Core > 2GHz and 2GB RAM, for Windows 7 or Windows 10. We recommend a screen resolution of min. 1024 x 768 pixels. A basic knowledge of computers is also required. The VISOR® vision sensor is supplied with the IP address 192.168.100.100 and a subnet mask 255.255.255.0. The VISOR® vision sensor is operated independently of a PC or PLC. A PC/notebook is only necessary for configuration of the VISOR® vision sensor. Attention must be paid to sufficient and constant object illumination to ensure reproducible results and avoid malfunction. Reflections or varying incident light may affect detection results. If necessary, use an external light source and/or light-screening / shrouding devices to exclude incident light.

2 Intended Use

2.1 Field of application

The VISOR® vision sensor is an optical sensor and uses several evaluation methods according to the version: pattern recognition, contrast detection, brightness, BLOB, caliper, gray level, contour detection, barcode or Data Matrix code reading, Optical Character Recognition as well as wafer detection. The product is designed for industrial use only. In residential areas it is possible that additional measures for noise suppression must be done.

Object:
The VISOR® vision sensor precisely detects faulty parts, parts in the wrong place, at the wrong angle or in the wrong order or a combination of all of these. Several detectors are available for inspection tasks and interpretation: e.g. pattern matching, contour detection, brightness, gray level, contrast detection, caliper or BLOB. The advanced version of the VISOR® vision sensor also offers alignment: it is thus now also possible to reliably detect those features which do not appear with repeated accuracy in the taught position. All interpretation is carried out relative to the actual position and angle of the part without having to define an independent characteristic for each possible position. This high capacity tool also enables you to solve demanding pick and place applications.
The advanced version offers also the calibration in world coordinates for measurement- and robot applications.

Code Reader:
Identification of products, components or packaging from printed or directly marked – punched or laser-etched – codes is common practice in many sectors of industry today. The VISOR® Code Reader from SensoPart immediately detects which part is in front of it: it can easily read numerous types of barcodes as well as printed and directly marked data matrix codes according to ECC 200 standard and read characters directly via Optical Character Recognition (OCR), and this on any base (metal, plastic, paper, glass). The sensor can even routinely decipher askew or warped codes or codes on convex, reflective or transparent surfaces. The VISOR® Code Reader assesses the quality of your printed or directly marked data matrix codes using standardised ISO and AIM quality parameters. This enables you to introduce early correctional measures and thus avoid rejects due to illegible codes.

Solar:
The VISOR® Solar sensor offers an optimised inspection algorithm for a sound, process-concurrent quality control of sensitive silicon wafers. The relevant functions for wafer and cell inspection, from the detection of the size and shape of the wafer to the location of defects and the setting of processing speed and inspection accuracy, are already preconfigured so that the sensor is ready for operation in just a few mouse clicks.

Color:
The VISOR® Color offers powerful object detection in combination with color detection. This leads to an increased stability in several object detection applications as well as the possibility to

sort colored parts which would have a similar look in gray image. Beside this even active objects (like e.g. lighting LED´s) or “non colors” like black and white can be detected.

Allround:
In the VISOR® Allround all functions of VISOR® Object, Code Reader and Color are available in combination in one device. The Professional version offers also the Multishot function to detect smallest surface defects.

The VISOR® vision sensor range is an economic alternative to conventional image processing systems.

2.2 Functions overview

2.2.1 Characteristics VISOR®: Object / Code Reader / Solar

Function	**Object Std.**	**Object Adv.**	**Code Reader Std.**	**Code Reader Adv.**	Code Reader Prof.	**Solar Std.**	**Solar Adv.**
Frames per second	50	50	50	50	50	50	50
Number of Jobs	8	255	8	255	255	8	255
Alignment	Contour only	X		X	X		X
Calibration in world coordinates		X					
• Measurement: Scaling		X					
• Measurement: Calibration plate		X					
• Robotic: Point pair list		X					
Number of detectors	32	255	2	255	255	32	255
• Pattern matching (X-, Y- translation)	X	X		X	X		X
• Contour matching (X-, Y- translation and rotation)	X	X					
• Gray level	X	X		X	X	X	X
• Contrast	X	X		X	X	X	X
• Brightness	X	X		X	X	X	X
• Caliper		X					X

Function	**Object Std.**	**Object Adv.**	**Code Reader Std.**	**Code Reader Adv.**	Code Reader Prof.	**Solar Std.**	**Solar Adv.**
• BLOB		X					X
• Data code			X	X	X		
• Barcode			X	X	X		
• OCR					X		
• Wafer						X	X
• Busbar							X
4 digital outputs, 2 inputs, PNP or NPN	X	X	X	X	X	X	X
Free definable digital In- / Outputs, PNP or NPN	2	4	2	4	4	2	4
Free shape of ROI	contour only	X		X	X		X
Timeout, specified time response	X	X	X	X	X	X	X
Variable resolutions	X	X	X	X	X	X	X
Illumination quadrant controlled	X	X	X	X	X	X	X
Image recorder	X	X	X	X	X	X	X
Encoder input		X		X	X		X
Ethernet interface	X	X	X	X	X	X	X
PROFINET	X	X	X	X	X	X	X
RS422 / RS232 interface		X	X	X	X		X
EtherNet/IP interface	X	X	X	X	X	X	X
Sensor monitoring by Viewer, Job-Upload	X	X	X	X	X	X	X

Function	**Object Std.**	**Object Adv.**	**Code Reader Std.**	**Code Reader Adv.**	Code Reader Prof.	**Solar Std.**	**Solar Adv.**
Sensor monitoring by SensoWeb (Webviewer)	X	X	X	X	X	X	X
I/O- Extension (with Encoder-control / Profibus- Inter-face)		X	X	X	X		X
V10 integrated 6 / 12 / 25 mm	X / X / -	X / X / X	X / X / X	X / X /X	- / - / -	X / - / -	X / X / -
V20 integrated 12 mm		X	X	X	X		X
Version with C-Mount		X		X	X		X

2.2.2 Characteristics VISOR®: Color, Allround

Function	**Color Standard**	**Color Advanced**	**Monochrome Allround Advanced**	**Monochrome Allround Professional**	**Color Allround Advanced**
Frames per second	40	40	40	40	40
Number of Jobs	8	255	255	255	255
Alignment	Contour only	X	X	X	X
Calibration in world coordinates		X	X	X	X
• Measurement: Scaling		X	X	X	X
• Measurement: Calibration plate		X	X	X	X
• Robotic: Point pair list		X	X	X	X

Function	Color Standard	Color Advanced	Monochrome Allround Advanced	Monochrome Allround Professional	Color Allround Advanced
Number of detectors	32	255	255	255	255
• Pattern matching (X-, Y- translation)		X	X	X	X
• Contour matching (X-, Y- translation and rotation)		X	X	X	X
• Gray level		X	X	X	X
• Contrast		X	X	X	X
• Brightness		X	X	X	X
• Caliper		X	X	X	X
• BLOB		X	X	X	X
• Data code			X	X	X
• Barcode			X	X	X
• OCR			X	X	X
• Color value		X			X
• Color area	X	X			X
• Color List		X			X
• Multishot				X	
4 digital outputs, 2 inputs,	X	X	X	X	X

Function	Color Standard	Color Advanced	Monochrome Allround Advanced	Monochrome Allround Professional	Color Allround Advanced
PNP or NPN					
Free definable digital In- / Outputs, PNP or NPN	2	4	4	4	4
Free shape of ROI	Contour only	X	X	X	X
Timeout, specified time response	X	X	X	X	X
Variable res-olutions	X	X	X	X	X
Illumination quad-rant controlled	X	X	X	X	X
Image recorder	X	X	X	X	X
Encoder input		X	X	X	X
Ethernet interface	X	X	X	X	X
PROFINET	X	X	X	X	X
RS422 / RS232 interface		X	X	X	X
EtherNet/IP inter-face	X	X	X	X	X
Sensor monitoring by Viewer, Job-Upload	X	X	X	X	X
Sensor monitoring by SensoWeb (Webviewer)	X	X	X	X	X
I/O- Extension (with Encoder-con-trol / Profibus- Inter-face)		X	X	X	X

Function	Color Standard	Color Advanced	Monochrome Allround Advanced	Monochrome Allround Professional	Color Allround Advanced
V10 integrated 6 / 12 / 25 mm	X / X / -	X / X / X	- / - / -	- / - / -	- / - / -
V20 integrated 12 mm		X	X	X	X
Version with C-Mount		X	X	X	X

2.3 Sensor types

2.3.1 Object

Part no.	**Type**	**Focal length**	**Depth of focus**	**Internal illumination**	**min. operating distance / mm *1**	**min. Field of view mm x mm**
V10 Advanced White						
535-91001	V10-OB-A1-W6	6	Normal	White	6	5 x 4
535-91002	V10-OB-A1-W12	12	Normal	White	30	8 x 6
535-91012	V10-OB-A1-W25	25	Normal	White	140	18 x 14
535-91013	V10-OB-A1-W6D	6	Enhanced	White	6	5 x 4
535-91014	V10-OB-A1-W12D	12	Enhanced	White	30	8 x 6
V10 Advanced Red						
535-91003	V10-OB-A1-R6	6	Normal	Red	6	5 x 4
535-91004	V10-OB-A1-R12	12	Normal	Red	30	8 x 6
535-91015	V10-OB-A1-R25	25	Normal	Red	140	18 x 14
535-91016	V10-OB-A1-R6D	6	Enhanced	Red	6	5 x 4
535-91017	V10-OB-A1-R12D	12	Enhanced	Red	30	8 x 6
V10 Advanced IR						
535-91006	V10-OB-A1-I6 *3	6	Normal	InfraRed	6	5 x 4

Part no.	Type	Focal length	Depth of focus	Internal illumination	min. operating distance / mm *1	min. Field of view mm x mm
535-91007	V10-OB-A1-I12 *3	12	Normal	InfraRed	30	8 x 6
535-91018	V10-OB-A1-I25 *3	25	Normal	InfraRed	140	18 x 14
535-91019	V10-OB-A1-I6D *3	6	Enhanced	InfraRed	6	5 x 4
535-91020	V10-OB-A1-I12D *3	12	Enhanced	InfraRed	30	8 x 6
V10 Advanced C-Mount						
535-91005	V10-OB-A1-C *2,3	C-Mount		External	lens dependant	lens dependant
V10 Standard White						
535-91008	V10-OB-S1-W6	6	Normal	White	6	5 x 4
535-91009	V10-OB-S1-W12	12	Normal	White	30	8 x 6
V10 Standard Red						
535-91010	V10-OB-S1-R6	6	Normal	Red	6	5 x 4
535-91011	V10-OB-S1-R12	12	Normal	Red	30	8 x 6
V10 Standard IR						
535-91046	V10-OB-S1-I6 *3	6	Normal	InfraRed	6	5 x 4
535-91047	V10-OB-S1-I12 *3	12	Normal	InfraRed	30	8 x 6
V20 Advanced White						

Part no.	Type	Focal length	Depth of focus	Internal illumination	min. operating distance / mm *1	min. Field of view mm x mm
536-91011	V20-OB-A2-W12	12	Normal	White	30	16 x 13
V20 Advanced Red						
536-91012	V20-OB-A2-R12	12	Normal	Red	30	16 x 13
V20 Advanced IR						
536-91013	V20-OB-A2-I12*3	12	Normal	InfraRed	30	16 x 13
V20 Advanced C-Mount						
536-91010	V20-OB-A2-C*2,3	C-Mount		External	lens dependant	lens dependant

*1 For longer operating distances (from approx. 200 mm) external illumination may be necessary.

*2 When the C-Mount version of VISOR® is in use, a C-Mount lens with a 5 mm intermediate ring (delivered separately) or a C-Mount protective case is required.

*3 External IR illumination is only possible with IR sensors or C-Mount sensors.

2.3.2 Code Reader

Part no.	Type	Focal length	Depth of focus	Internal illumination	min. operating distance / mm *1	min. Field of view mm x mm
V10 Advanced White						
535-91021	V10-CR-A1-W6	6	Normal	White	6	5 x 4
535-91022	V10-CR-A1-W12	12	Normal	White	30	8 x 6
535-91084	V10-CR-A2-W25	25	Normal	White	140	18 x 14

Part no.	Type	Focal length	Depth of focus	Internal illumination	min. operating distance / mm [*1]	min. Field of view mm x mm
535-91023	V10-CR-A1-W6D	6	Enhanced	White	6	5 x 4
535-91024	V10-CR-A1-W12D	12	Enhanced	White	30	8 x 6
V10 Advanced Red						
535-91025	V10-CR-A1-R6	6	Normal	Red	6	5 x 4
535-91026	V10-CR-A1-R12	12	Normal	Red	30	8 x 6
535-91085	V10-CR-A2-R25	25	Normal	Red	140	18 x 14
535-91027	V10-CR-A1-R6D	6	Enhanced	Red	6	5 x 4
535-91028	V10-CR-A1-R12D	12	Enhanced	Red	30	8 x 6
V10 Advanced IR						
535-91029	V10-CR-A1-I6 [*3]	6	Normal	InfraRed	6	5 x 4
535-91030	V10-CR-A1-I12 [*3]	12	Normal	InfraRed	30	8 x 6
535-91086	V10-CR-A2-I25 [*3]	25	Normal	InfrarRed	140	18 x 14
535-91031	V10-CR-A1-I6D [*3]	6	Enhanced	InfraRed	6	5 x 4
535-91032	V10-CR-A1-I12D [*3]	12	Enhanced	InfraRed	30	8 x 6
V10 Advanced C-Mount						
535-	V10-CR-A1-C [*2,3]	C-		External	lens dependant	lens

Part no.	Type	Focal length	Depth of focus	Internal illumination	min. operating distance / mm [*1]	min. Field of view mm x mm
91033		Mount				dependant
V10 Standard White						
535-91034	V10-CR-S1-W6	6	Normal	White	6	5 x 4
535-91035	V10-CR-S1-W12	12	Normal	White	30	8 x 6
535-91088	V10-CR-S2-W25	25	Normal	White	140	18 x 14
535-91036	V10-CR-S1-W6D	6	Enhanced	White	6	5 x 4
535-91037	V10-CR-S1-W12D	12	Enhanced	White	30	8 x 6
V10 Standard Red						
535-91038	V10-CR-S1-R6	6	Normal	Red	6	5 x 4
535-91039	V10-CR-S1-R12	12	Normal	Red	30	8 x 6
535-91089	V10-CR-S2-R25	25	Normal	Red	140	18 x 14
535-91040	V10-CR-S1-R6D	6	Enhanced	Red	6	5 x 4
535-91041	V10-CR-S1-R12D	12	Enhanced	Red	30	8 x 6
V10 Standard IR						
535-91042	V10-CR-S1-I6 [*3]	6	Normal	InfraRed	6	5 x 4
535-91043	V10-CR-S1-I12 [*3]	12	Normal	InfraRed	30	8 x 6

Part no.	Type	Focal length	Depth of focus	Internal illumination	min. operating distance / mm *1	min. Field of view mm x mm
535-91090	V10-CR-S2-I25*3	25	Normal	InfraRed	140	18 x 14
535-91044	V10-CR-S1-I6D *3	6	Enhanced	InfraRed	6	5 x 4
535-91045	V10-CR-S1-I12D *3	12	Enhanced	InfraRed	30	8 x 6
V20 Advanced White						
536-91001	V20-CR-A2-W12	12	Normal	White	30	16 x 13
536-91026	V20C-CR-A2-W12	12	Normal	White	30	16 x 13
V20 Advanced Red						
536-91002	V20-CR-A2-R12	12	Normal	Red	30	16 x 13
V20 Advanced IR						
536-91003	V20-CR-A2-I12*3	12	Normal	InfraRed	30	16 x 13
V20 Advanced UV						
536-91019	V20-CR-A2-U12	12	Normal	UV	30	16 x 13
V20 Advanced C-Mount						
536-91000	V20-CR-A2-C*2,3	C-Mount		External	lens dependant	lens dependant
V20 Professional White						
536-91005	V20-CR-P2-W12	12	Normal	White	30	16 x 13
536-91027	V20C-CR-P2-W12	12	Normal	White	30	16 x 13

Part no.	Type	Focal length	Depth of focus	Internal illumination	min. operating distance / mm *1	min. Field of view mm x mm
V20 Professional Red						
536-91006	V20-CR-P2-R12	12	Normal	Red	30	16 x 13
V20 Professional IR						
536-91007	V20-CR-P2-I12*3	12	Normal	InfraRed	30	16 x 13
V20 Professional C-Mount						
536-91004	V20-CR-P2-C*2,3	C-Mount		External	lens dependant	lens dependant

*1 For longer operating distances (from approx. 200 mm) external illumination may be necessary.

*2 When the C-Mount version of VISOR® is in use, a C-Mount lens with a 5 mm intermediate ring (delivered separately) or a C-Mount protective case is required.

*3 External IR illumination is only possible with IR sensors or C-Mount sensors.

2.3.3 Solar

Part no.	Type	Focal length	Depth of focus	Internal illumination	min. operating distance / mm *1	min. Field of view mm x mm
V10 Advanced White						
535-91051	V10-SO-A1-W6	6	normal	White	361	170x261
535-91052	V10-SO-A1-W12	12	normal	White	706	170x261
V10 Advanced IR						
535-91053	V10-SO-A1-I6 *3	6	normal	InfraRed	361	170x261
535-91054	V10-SO-A1-I12 *3	12	normal	InfraRed	706	170x261
V10 Advanced C-Mount						
535-91050	V10-SO-A1-C *2,3	C-		External	lens	lens

Part no.	**Type**	**Focal length**	**Depth of focus**	**Internal illumination**	**min. operating distance / mm** [*1]	**min. Field of view mm x mm**
		Mount			dependant	dependant
V10 Standard White						
535-91049	V10-SO-S1-W6	6	normal	White	361	170x261
V20 Advanced White						
536-91028	V20-SO-A2-W12	12	normal	White		
V20 Advanced Red						
536-91029	V20-SO-A2-R12	12	normal	Red		
V20 Advanced IR						
536-91030	V20-SO-A2-I12	12	normal	InfraRed		
V20 Advanced C-Mount						
536-91031	V20-SO-A2-C	12	normal	External	lens dependant	lens dependant

*1 for inspection of 6"-Wafers. The typical focus range is operating distance ± 5%.

*2 When the C-Mount version of VISOR® is in use, a C-Mount lens with a 5 mm intermediate ring (delivered separately) or a C-Mount protective case is required.

*3 External IR illumination is only possible with IR sensors.

2.3.4 Color Sensor

Part no.	**Type**	**Focal length**	**Depth of focus**	**Internal illumination**	**min. operating distance / mm** [*1]	**min. Field of view mm x mm**
V10 Advanced White						
535-91073	V10C-CO-A2-W6	6	Normal	White	6	5 x 4
535-91074	V10C-CO-A2-W12	12	Normal	White	30	8 x 6

Part no.	**Type**	**Focal length**	**Depth of focus**	**Internal illumination**	**min. operating distance / mm** *1	**min. Field of view mm x mm**
535-91075	V10C-CO-A2-W25	25	Normal	White	140	18 x 14
V10 Advanced C-Mount						
535-91076	V10C-CO-A2-C *2	C-Mount		External	lens dependant	lens dependant
V10 Standard White						
535-91071	V10C-CO-S2-W6	6	Normal	White	6	5 x 4
535-91072	V10C-CO-S2-W12	12	Normal	White	30	8 x 6
V20 Advanced White						
536-91020	V20C-CO-A2-W12	12	Normal	White	30	8 x 6
V20 Advanced C-Mount						
536-91021	V20C-CO-A2-C *2	C-Mount		External	lens dependant	lens dependant

*1 For longer operating distances (from approx. 200 mm) external illumination may be necessary.

*2 When the C-Mount version of VISOR® is in use, a C-Mount lens with a 5 mm intermediate ring (delivered separately) or a C-Mount protective case is required.

2.3.5 Allround

Part no.	**Type**	**Focal length**	**Depth of focus**	**Internal illumination**	**min. operating distance / mm** *1	**min. Field of view mm x mm**
V20 Allround White						
536-91032	V20-ALL-A2-W12	12	Normal	White	30	16 x 13

Part no.	Type	Focal length	Depth of focus	Internal illumination	min. operating distance / mm *1	min. Field of view mm x mm
V20 Allround Red						
536-91033	V20-ALL-A2-R12	12	Normal	Red	30	16 x 13
V20 Allround IR						
536-91034	V20-ALL-A2-I12*3	12	Normal	InfraRed	30	16 x 13
V20 Allround C-Mount						
536-91035	V20-All-A2-C*2,3	C-Mount		External	lens dependant	lens dependant
V20 Allround Color White						
536-91036	V20C-ALL-A2-W12	12	Normal	White	30	16 x 13
V20 Allround Color C-Mount						
536-91037	V20C-All-A2-C*2,3	C-Mount		External	lens dependant	lens dependant

*1 For longer operating distances (from approx. 200 mm) external illumination may be necessary.

*2 When the C-Mount version of VISOR® is in use, a C-Mount lens with a 5 mm intermediate ring (delivered separately) or a C-Mount protective case is required.

*3 External IR illumination is only possible with IR sensors or C-Mount sensors.

2.4 Field of view / Depth of view

Field of view V10 6mm lens, internal

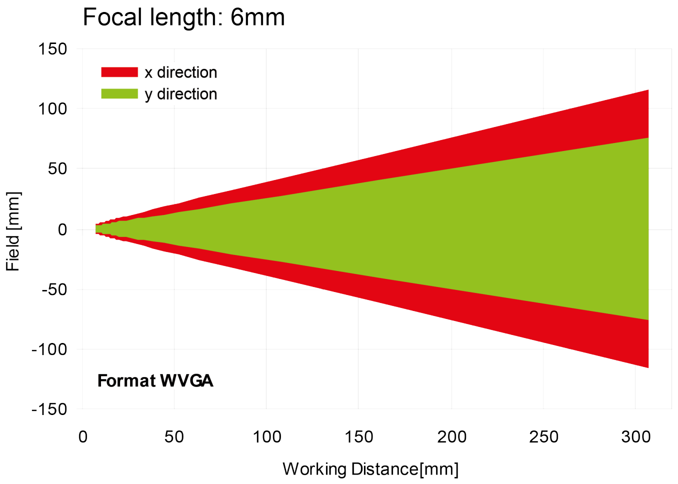


Fig. 1: Field of view V10 6mm lens, internal

Field of view V10 12mm lens, internal

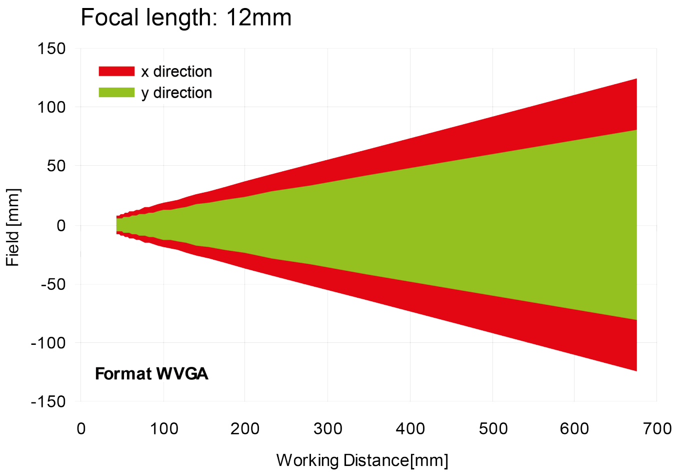


Fig. 2: Field of view V10 12mm lens, internal

Field of view V10 25mm lens, internal

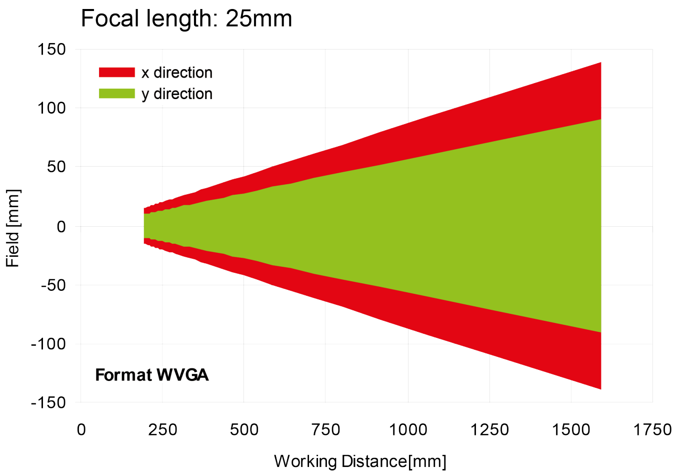


Fig. 3: Field of view V10 25mm lens, internal

Field of view V20 12mm lens, internal

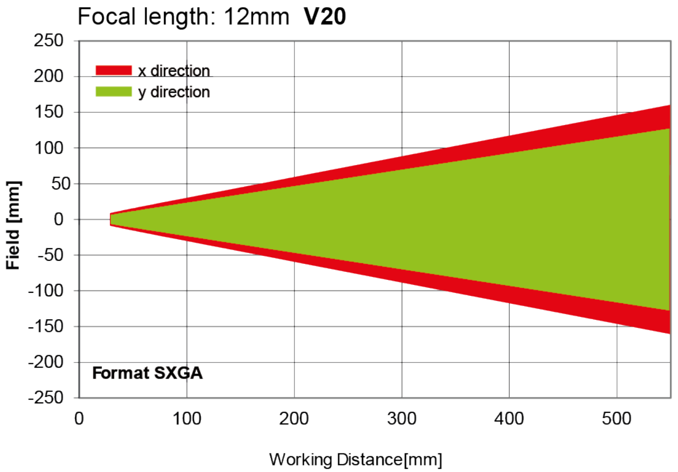


Fig. 4: Field of view V20 12mm lens, internal

Depth of view V10 6mm lens internal, normal

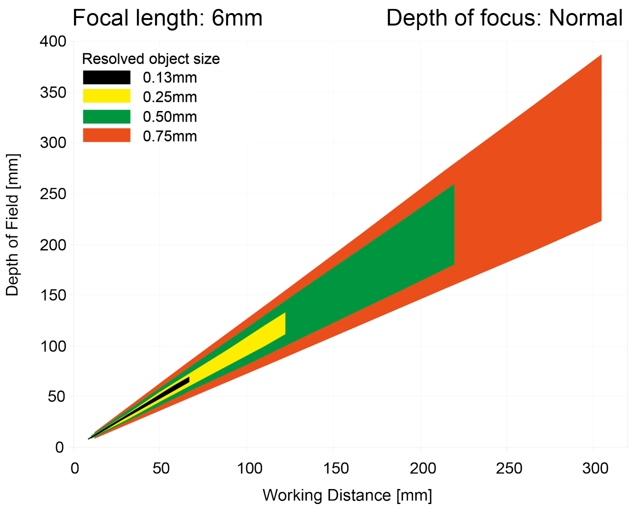


Fig. 5: Depth of view V10 6mm lens internal, normal

Depth of view V10 6mm lens internal, enhanced

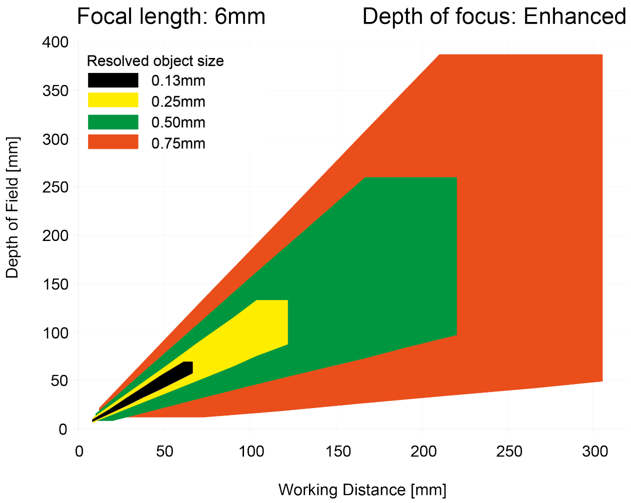


Fig. 6: Depth of view V10 6mm lens internal, enhanced

Depth of view V10 12mm lens internal, normal

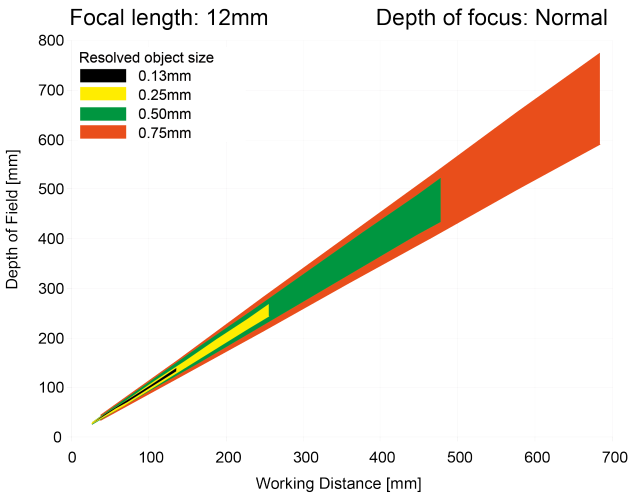


Fig. 7: Depth of view V10 12mm lens internal, normal

Depth of view V10 12mm lens internal enhanced

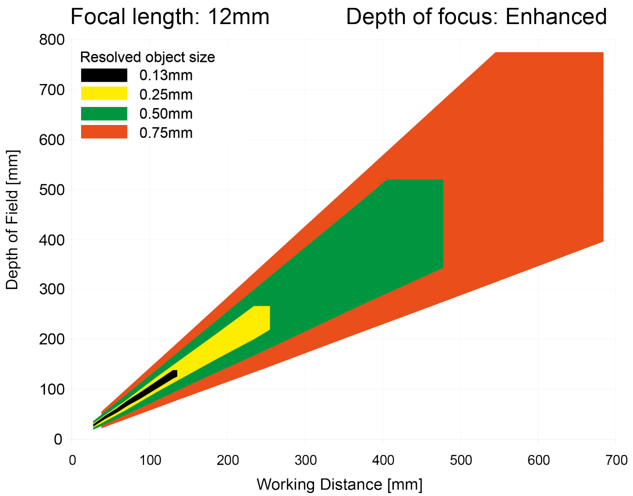


Fig. 8: Depth of view V10 12mm lens internal, enhanced

Depth of view V10 25mm lens internal, normal

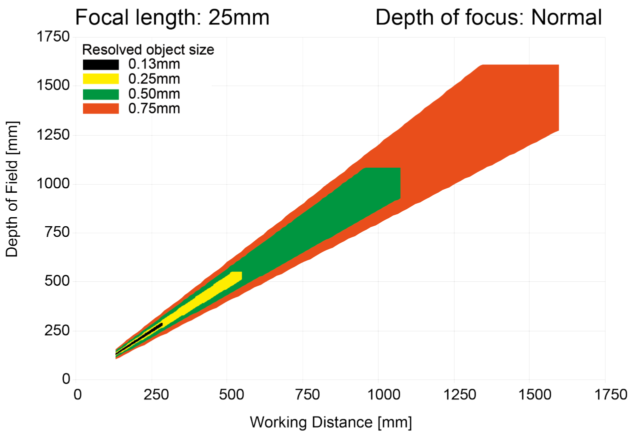


Fig. 9: Depth of view V10 25mm lens internal, normal

Depth of view V20 12mm lens internal, normal

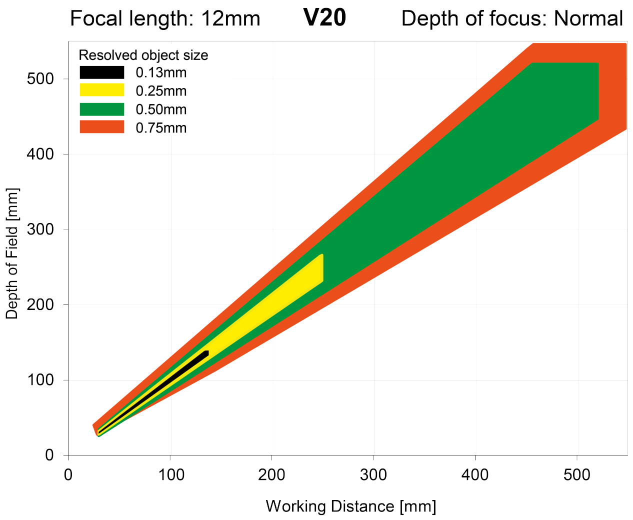


Fig. 10: Depth of view V20 12mm lens internal, normal

3 Installation

3.1 Mechanical Installation

To ensure maximum accuracy of detection, the VISOR® vision sensor should be protected from vibration. Secure the supply and I/O cables with cable binders to prevent crushing or slipping. Select a position for the VISOR® vision sensor in which interfering factors such as slight differences in the position of the object or variations in illumination have little or no effect.
Screw the VISOR® vision sensor onto the mounting clamp (supplied with the unit) and then onto a suitable object. Use only the Mounting bracket MK 45 (no. 543-11000) or the mounting hinge MG2A (no. 543-11023).

3.1.1 Arrangement for dark-field illumination

For the prevention of direct reflections and accentuation of edges etc.

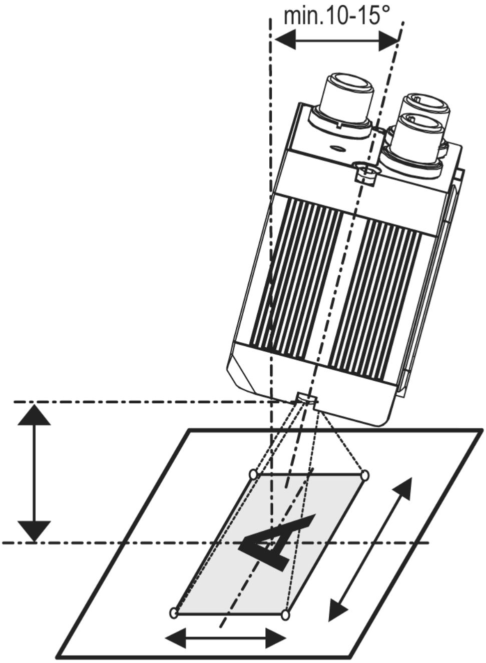


Fig. 11: Arrangement for dark-field illumination

3.1.2 Arrangement for bright-field illumination

For transmitted light/measuring tasks or for the accentuation of highly-reflective objects

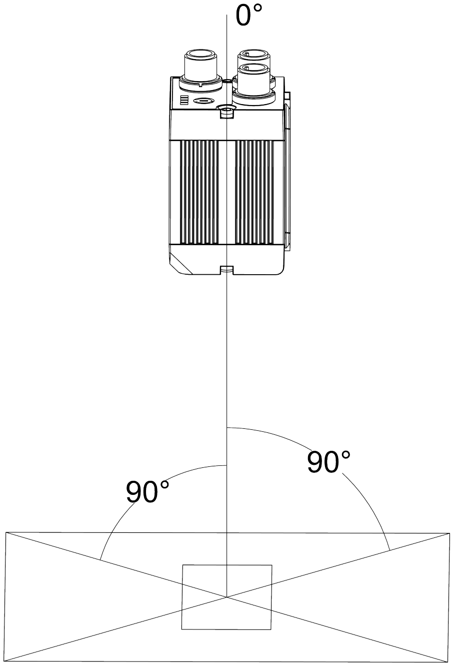


Fig. 12: Arrangement for bright-field illumination

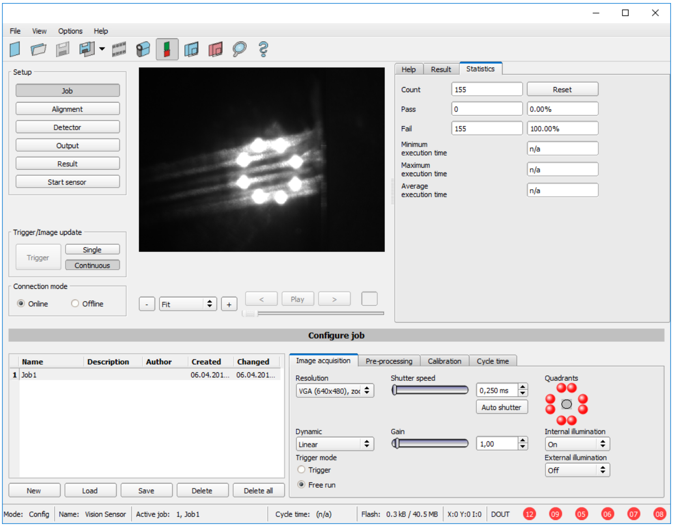
Observe the object clearance given in the table Field of View / Working Distance. To avoid interfering reflection from the detection object, align the VISOR® vision sensor at an angle of approx. 10°- 15° with reference to the optical axis.

Fine adjustment

Important: Fine adjustment of the VISOR® vision sensor should not be carried out until after electrical connection and start-up (PC software installation).

3.1.3 Alignment for a vertical illumination

In order to assure the absolutely vertical alignment of the VISOR® to the object surface, put a piece of reflective foil or a mirror on top of the object and start the VISOR® operating software. For an image that is continually updated, select trigger mode “Free run” and image update: “Continuous”. Then align the sensor to the reflective surface / the mirror as vertical as possible until the integrated illumination LEDs are directly dazzling in the image of the user interface (Arrangement for bright-field illumination (Page 37)).



Count	155	
Pass	0	0.00%
Fail	155	100.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Name	Description	Author	Created	Changed
1	Job1			06.04.201...	06.04.201...

Resolution: VGA (640x480), zoo

Shutter speed: 0,250 ms

Dynamic: Linear

Gain: 1,00

Trigger mode: ○ Trigger ● Free run

Internal illumination: On

External illumination: Off

Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 0.3 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT

Fig. 13: Alignment for a vertical illumination

3.1.4 Assembly VISOR® - Mounting bracket MK 45

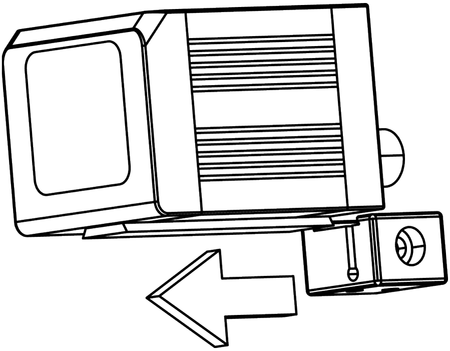


Fig. 14: Assembly VISOR® - Mounting bracket MK 45

For fixing the VISOR® on a fixing system / machine housing, slide the provided dovetail Mounting bracket MK 45 on the dovetail guide at the bottom side of the VISOR® and fix it at the desired position with the hexagon socket in the cross hole of the mounting bracket. Then additional SensoPart mounting accessories may be attached to the mounting bracket or any other attachments may be fixed by using the tapped holes in the Mounting bracket MK 45.

3.2 Electrical installation

The electrical installation of the VISOR® vision sensor must be carried out by a qualified person. When installing the VISOR® vision sensor, disconnect all electrical components from the power supply. When the unit is being used in a network, ensure that the network address (IP address) of the VISOR® vision sensor set by the manufacturer at 192.168.100.100 is free and is not in use for any other unit connected to the system. If necessary, re-set the IP address of the VISOR® vision sensor as described in the section “Network settings”. When the VISOR® vision sensor is in use, the protective caps supplied must be pushed onto the M12 sockets (data and LAN) which are not in use. For error free operation the length of the connecting cables must not be longer than 30 m. Failure to do this may cause malfunction.

3.2.1 Connection possibilities

For stand-alone operation (independent of PC /PLC) only connection 24 V DC is required after-start-up.

For electrical installation, connect wires as follows:
***A: LED display**
***B: Focussing screw**
***C: 24 VDC, I/O- M12 connection socket**
***D: Data (RS422) M12 socket**
***E: LAN M12 connection socket**

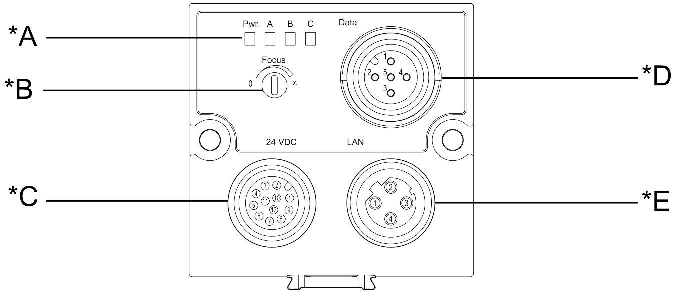


Fig. 15: Connectors VISOR®

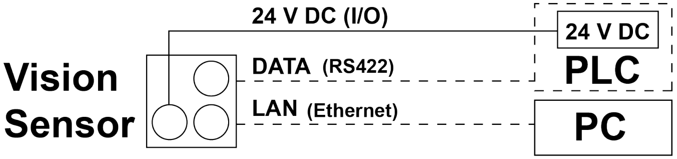


Fig. 16: Connection VISOR®

3.2.1.1 LED Display

Name	Color	Meaning
Pwr.	green	Operating voltage
A	yellow	Result 1
B	yellow	Result 2
C	yellow	Result 3

All LED´s are set without taking into account any timing function (e.g. Trigger delay)

3.2.1.2 Focussing screw

Focussing screw to adjust focus.
Focus: Clockwise = higher distance
Counter Clockwise = lower distance

3.2.1.3 24 V DC Connection

M12 Connection socket for 24 V DC voltage supply and digital I/O.
For the exact plug connection see PIN assignment, connection 24 V DC

3.2.1.4 LAN Connection

M12 Connection socket for Ethernet connection.
For the exact plug connection see PIN assignment, connection LAN .
Use only the correct network cables.

3.2.1.4.1 Direct connection of the VISOR® vision sensor to a PC (recommended)

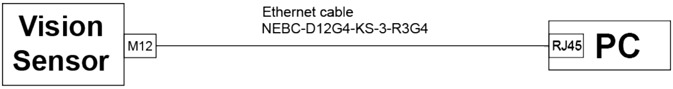


Fig. 17: Direct connection VISOR® ↔ PC

3.2.1.4.2 Connection of the VISOR® vision sensor to a PC via a network:

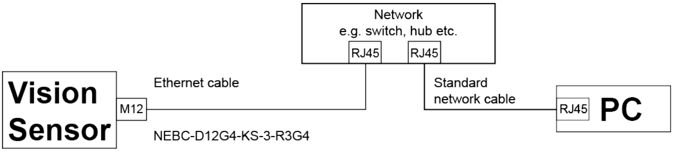


Fig. 18: Connection via a network

3.2.1.5 Data Connection

M12 Connection socket for DATA serial interface, RS422 / RS232.
s. PIN assignment DATA *A) (Page 44)

3.2.1.6 Plug connections

All pin assignments and signals are referring to the view from the sensor.

3.2.1.6.1 PIN assignment, connection 24 V DC

PIN	Color	Use
1	BN	+ Ub (24V DC)
2	BU	GND
3	WH	IN (external trigger)
4	GN	READY *1
5 *2, *5	PK	IN/OUT (advanced: encoder B+)
6 *2, *5	YE	IN/OUT
7 *2	BK	IN/OUT, LED B*4
8 *2	GY	IN/OUT, LED C*4
9	RD	OUT (external illumination)
10	VT	IN (advanced: encoder A+)
11	GYPK	VALID *3
12	RDBU	OUT (ejector, max. 100mA), LED A*4

*1 Ready: Ready for next ext. trigger
*2 Switchable input- output
*3 VALID: shows available results
*4 All LED´s are set without taking into account any timing function (e.g. Trigger delay)
*5 Not available with all Standard types

For shielded cables use shield, extensively connected.

3.2.1.6.2 PIN assignment, connection LAN

(M12) 4 pin	Signal
1	TxD+

(M12) 4 pin	Signal
2	RxD+
3	TxD-
4	RxD-

3.2.1.6.3 PIN assignment DATA *A)

PIN	Color	Use RS422	Use RS232
1	brown	RxD+	Rx
2	white	RxD-	NC
3	blue	TxD+	NC
4	black	TxD-	Tx
5	gray	GND	GND

*A) Not with Object- , Color-, Solar- Standard version

For shielded cables use shield.

3.2.1.6.4 Exemplary connection plan and software settings for the following setup:

- Power supply
- Trigger
- 1x digital output
- Encoder
- Ethernet to PC or PLC

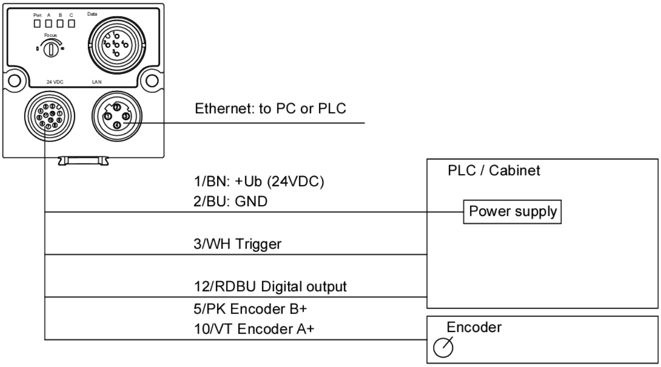


Fig. 19: Exemplary connection plan

3.2.1.6.5 Electrical connection supply voltage and shield

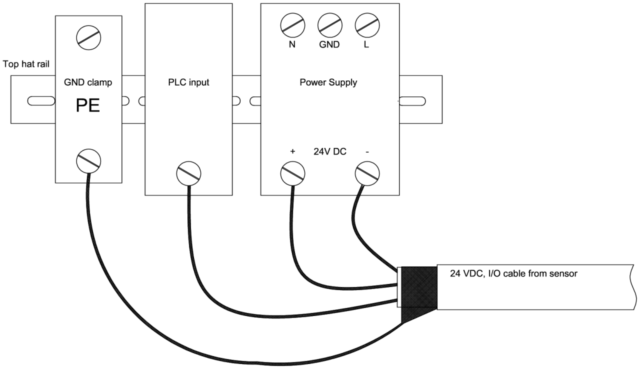


Fig. 20: Electrical connection, supply voltage 24VDC in cabinet with shield

3.2.1.6.6 Electrical connection PNP / NPN

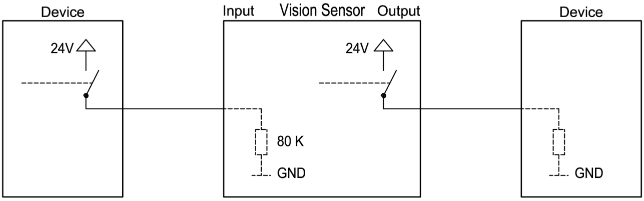


Fig. 21: Connection example VISOR® in PNP mode. In-/outputs switch to +24V

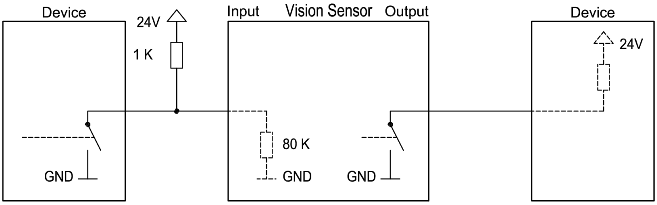


Fig. 22: Connection example VISOR® in NPN mode

As the inputs refer to ground, an additional pull-up resistor may be required in order to increase the input voltage to 24V when unswitched. The outputs switch to ground.

3.3 Network settings, Short reference

The following instructions indicate how to change the network configuration of the PC and the VISOR® vision sensor. If incorrect settings are used, the network connections in the computer may be lost. To be on the safe side, note the former settings for later use if required. Following this procedure, it may be necessary to re-start the system. In order to determine which IP addresses are allowed in your network or locally in your PC, and to carry out the necessary settings on your PC, contact the system administrator beforehand. The illustrations, dialogues and menus originate from the operating system Microsoft WindowsXP™. The illustrations are similar in other operating systems.

See chapter: Basic settings for PC and VISOR® vision sensor

3.3.1 Basic settings for PC and VISOR® vision sensor

To configure the VISOR® vision sensor with a PC it is essential that a network board and the TCP/IP LAN- connection is installed on the PC (This also applies when the PC is not connected to a network). The VISOR® supports the automatic recognition of the Ethernet transmission rate, but 100 MBit at the most. The internet protocol IPv4 must be activated. There are two alternatives to configure and parameterize the VISOR® vision sensor.

Also see chapter Network connection

1. **Direct Connection**
2. **Network Connection**

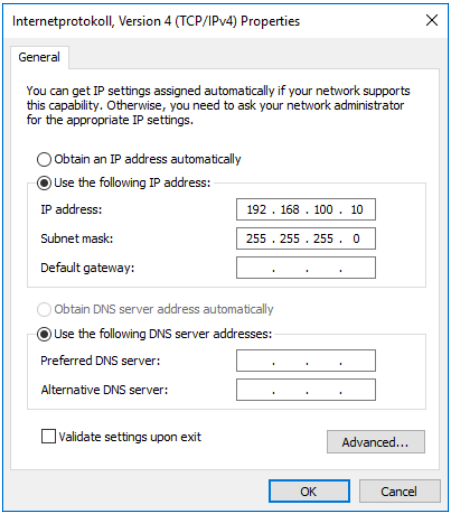
3.3.2 Direct Connection - Setting the IP Address of the PC

To connect the VISOR® vision sensor to a PC via Ethernet the IP addresses of both devices have to correspond. The default IP of the VISOR® is 192.168.100.100 with Subnet mask = 255.255.255.0. To establish a direct connection, the PC must be set to a corresponding, fixed IP address like follows.

1. Click on Start / Control Panel / Network Connection / LAN Connection / Properties, the window "Local Area Connection Properties" opens.
2. In the list "This connection requires following elements" select the option "Internet Protocol (TCP/IP)" and then click the button "Properties".
3. In the following window (see fig. 7) set the desired IP address of the PC and the sub-network data.
4. Confirm entries with OK.

Example:

The VISOR® vision sensor is pre-set to IP address 192.168.100.100 and subnet mask 255.255.255.0.
In this case, the IP address may be set to any value between 192.168.100.1 and 192.168.100.254, with a subnet mask 255.255.255.0, with the exception of the sensor IP address (192.168.100.100).
To alter the sensor's IP address, see chapter Sensor's network settings (Page 73). Please do also not use the addresses .0 and .255 as these addresses are reserved for network infrastructure devices such as servers, gateways, etc.



Internetprotokoll, Version 4 (TCP/IPv4) Properties

General

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically

● Use the following IP address:

IP address: 192 . 168 . 100 . 10

Subnet mask: 255 . 255 . 255 . 0

Default gateway: . . .

○ Obtain DNS server address automatically

● Use the following DNS server addresses:

Preferred DNS server: . . .

Alternative DNS server: . . .

□ Validate settings upon exit

Fig. 23: PC IP Setup

3.3.3 Network Connection - Setting the IP address of the VISOR® vision sensor

Before connecting the sensor in the network, check with the network administrator whether the sensor's address has already been assigned (default: 192.168.100.100 with subnet mask 255.255.255.0). This can otherwise cause network failure. The set IP address is to be noted on the enclosed label. The label is then to be stuck on the sensor in a clearly visible place after installation.

Network connection speed:

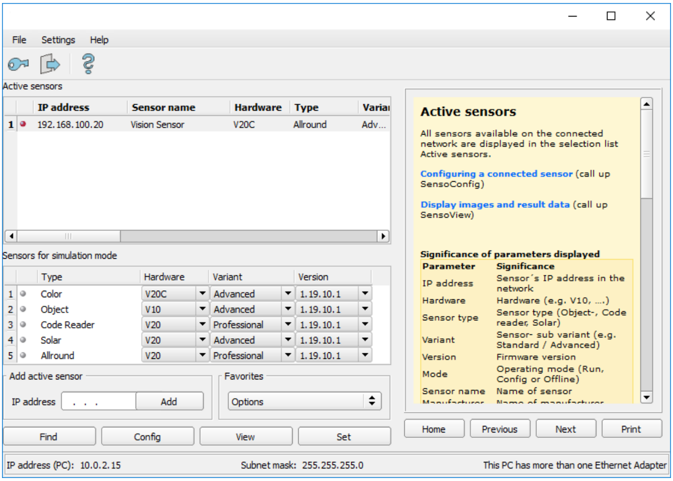
The sensor must only be operated with 100MBit/full-duplex when using VGA resolution (or higher) and SensoView.

Sensor's IP still free:

Connect sensor to network and then set the sensor's IP to match the PC according to the administrator's specifications, as follows, beginning with 2.

Sensor IP already assigned:

1. First connect sensor and PC directly and set an authorised IP address in the sensor.
2. Connection via the network can then be carried out. First ensure electrical connection and installation of PC software has been completed. To set the IP address on the VISOR® vision sensor, the following steps are to be carried out in the PC software:
   a. Start SensoFind software.
   b. Select the required VISOR® vision sensor from the active sensor list (single left mouse click).
   c. Set sensor's new IP address with the "Set" button. Follow the on screen prompts. The IP address is assigned by your system administrator. The PC's IP address is shown in the status bar under the buttons. (Please note some pc's have more than one Ethernet connection i.e. wireless and wired LAN connections.
   d. When the new IP address has been set, Re-select the sensor and connect via SensoConfig or SensoView.



Active sensors

	IP address	Sensor name	Hardware	Type	Variai
1	192.168.100.20	Vision Sensor	V20C	Allround	Adv...

Sensors for simulation mode

	Type	Hardware	Variant	Version
1	Color	V20C	Advanced	1.19.10.1
2	Object	V10	Advanced	1.19.10.1
3	Code Reader	V20	Professional	1.19.10.1
4	Solar	V20	Advanced	1.19.10.1
5	Allround	V20	Professional	1.19.10.1

Add active sensor

IP address . . .

Favorites

Options

Active sensors

All sensors available on the connected network are displayed in the selection list Active sensors.

Configuring a connected sensor (call up SensoConfig)

Display images and result data (call up SensoView)

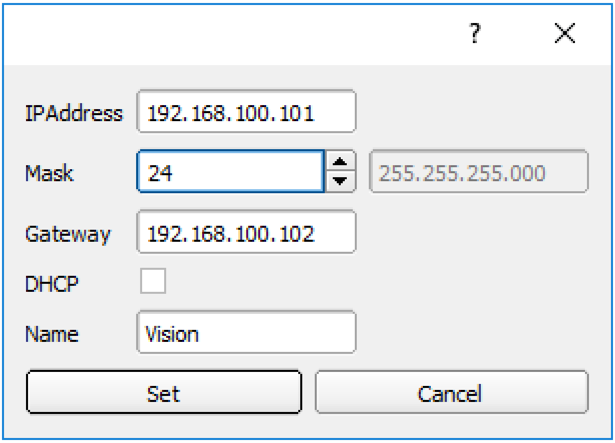
Significance of parameters displayed

Parameter	Significance
IP address	Sensor´s IP address in the network
Hardware	Hardware (e.g. V10, ...)
Sensor type	Sensor type (Object-, Code reader, Solar)
Variant	Sensor- sub variant (e.g. Standard / Advanced)
Version	Firmware version
Mode	Operating mode (Run, Config or Offline)
Sensor name	Name of sensor
Manufacturer	Name of manufacturer

IP address (PC): 10.0.2.15 Subnet mask: 255.255.255.0 This PC has more than one Ethernet Adapter

Fig. 24: SensoFind

Modification of the standard gateway enables operation in different sub-networks. Only alter this setting after consultation with your network administrator. Automatic integration of a new computer or sensor in the existing network without manual configuration is possible through DHCP. Normally, automatic supply of IP address must only be set on the sensor, the client. When the sensor is started in the network, it can obtain the IP address, net mask and gateway from a DHCP server. Activation of DHCP mode is carried out via the “Set” button by activating the checkbox “DHCP”. As one and the same VISOR® can thus have different IP addresses at different times, a sensor name must be attributed when activating the DHCP. Should several VISOR® be in one network, different names must be used.



IPAddress 192.168.100.101

Mask 24 255.255.255.000

Gateway 192.168.100.102

DHCP □

Name Vision

Fig. 25: VISOR® IP Setup

If a VISOR® with DHCP is switched on in a network without a DHCP server, the VISOR® automatically sets the IP address to 0.0.0.0. This can be the case, e.g. in the case of power/server failure or the restart of the system after shutdown as the DHCP server may boot slower than the VISOR®. Make sure that the VISOR® is only switched on when the DHCP server is available.

4 VISOR® – Operating- and configuration software

4.1 VISOR® – Operating- and configuration software - Overview

4.1.1 Structure of PC software

The PC software is organised into the following three sections:

- **SensoFind:**
  This module is for selection of a VISOR® sensor, or a sensor simulation model, for configuration with the “SensoConfig” tool,or display (monitoring) with the “SensoView” tool. Also system settings such as IP addresses, firmware updates can be modified with the “Set” tool.
- **SensoConfig:**
  Complete set of functions to configure and test VISOR® vision sensor for one or several inspection tasks (jobs) in six simple logical operating steps.
- **SensoView:**
  For the display and monitoring of images and results from connected sensors, as well as job switch and job upload.

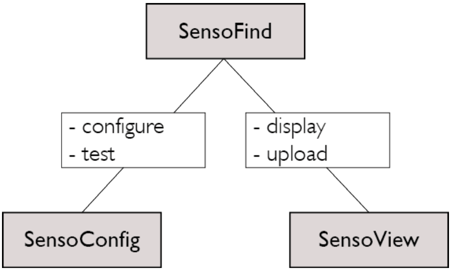


Fig. 26: Software structure

The latest software versions for free download are available at www.sensopart.com

4.1.2 Context help

For all software functions a context sensitive help page is available and displayed as soon as a function is selected.
All available help pages can be viewed by pressing the Help- button (“?” symbol) or by double click to the online help window. There you also can do a keyword search. In comparison to the context help the size of this help window can be enlarged to view longer text more comfortable.

Used open source software: Open Source Licences (Page 3)

4.2 VISOR®– Operating- and configuration software – Short introduction

(Example: Object sensor)

4.2.1 VISOR®, Short introduction, Starting the software

This short guide explains step by step the procedure for setting an example inspection task on the VISOR® vision sensor

To start the VISOR® application click to the desktop icon “VISOR® vision sensor”.

Fig. 27: Icon VISOR®

4.2.2 SensoFind: Open sensors or sensor simulation / Passwords

In this program, you can select a sensor or a sensor simulation for configuration or display (monitoring) and carry out different basic settings.

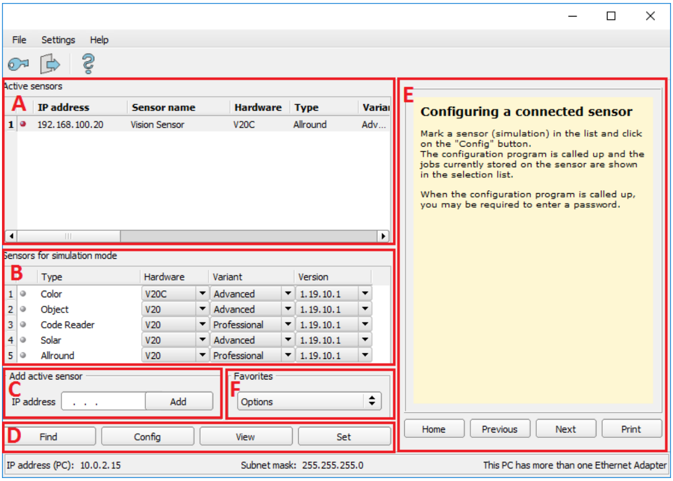
Next topic: SensoConfig: Setting sensor, Job (Page 59)

Configuring or displaying sensors

In order to open a sensor for configuration or display, select with a single left mouse click the required sensor in the “Active sensors” list, then click on the button “Config” to start the “SensoConfig” software, or on the button “View” for the “SensoView” software.

Sensor simulation

To open a sensor for offline simulation, select the required sensor in the “Sensors for simulation mode” list, then click on the button “Config” to start the module “SensoConfig”. SensoView is not available for the simulation mode as there is no device to send the images for display.



	IP address	Sensor name	Hardware	Type	Variai
1	192.168.100.20	Vision Sensor	V20C	Allround	Adv...

	Type	Hardware	Variant	Version
1	Color	V20C	Advanced	1.19.10.1
2	Object	V20	Advanced	1.19.10.1
3	Code Reader	V20	Professional	1.19.10.1
4	Solar	V20	Advanced	1.19.10.1
5	Allround	V20	Professional	1.19.10.1

Configuring a connected sensor

Mark a sensor (simulation) in the list and click on the "Config" button.
The configuration program is called up and the jobs currently stored on the sensor are shown in the selection list.

When the configuration program is called up, you may be required to enter a password.

IP address (PC): 10.0.2.15 Subnet mask: 255.255.255.0 This PC has more than one Ethernet Adapter

Fig. 28: SensoFind Overview

A) Active sensors

This list displays all of the VISOR® vision sensors available on the network that can be controlled from the PC.

B) Sensors for simulation mode

All the sensors available for offline simulation are displayed here.

C) Add sensors via IP address

Sensors, which are not visible after starting the software or after clicking the “Find” button in SensoFind, can be add manually with their IP address, if they are available in the network (e.g. after a gateway) and if the IP address is well-known. Via clicking the button “Add” such sensor can be found and are added to the list of active sensors, in order to edit them.

D) Functions

- **Find**
  Activates another search procedure on the network to locate VISOR® products
- **Config**
  Configures a connected sensor or a sensor simulation
- **View**
  Displays image or result data from a connected sensor
- **Set**
  Edits network settings such as the sensor's IP address etc.

E) Context help

Context sensitive help

F) Favorites
The VISOR® vision sensors can be stored as favorites. The favorites are used for quick access and for managing the VISOR® vision sensors.

4.2.3 Passwords

When first started-up after installation, password entry is completely deactivated and auto login is preset to administrator.

If parameter settings are to be protected from unauthorised access, passwords should be given for the "Admin" and "User" password levels, see below. This can be called up via the menu bar File / User administration or via the button with the key symbol in the toolbar.

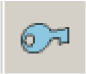


Fig. 29: Password button

4.2.4 Password levels:

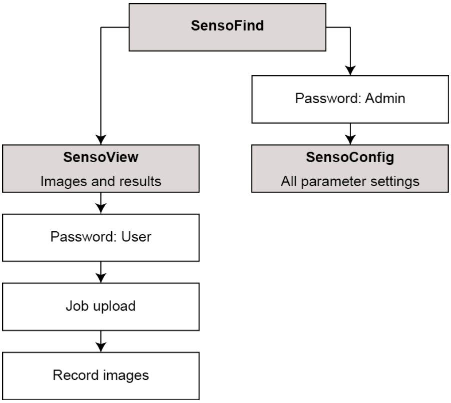


Fig. 30: Password levels

Password level	SensoFind	SensoConfig	SensoView
Administrator password	all functions	all functions	all functions
Worker password	all functions except • Config	none	all functions, including Job Upload and Image Recorder

Password level	SensoFind	SensoConfig	SensoView
	• Settings • Update		
User (without any password)	all functions except • Config • Settings • Update	none	only display of images, inspection results and statistics

In order to be able to use the function "Config" after the allocation of passwords, it is now necessary to login by clicking on the toolbar login button, and then entering the assigned password.

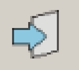


Fig. 31: Login button

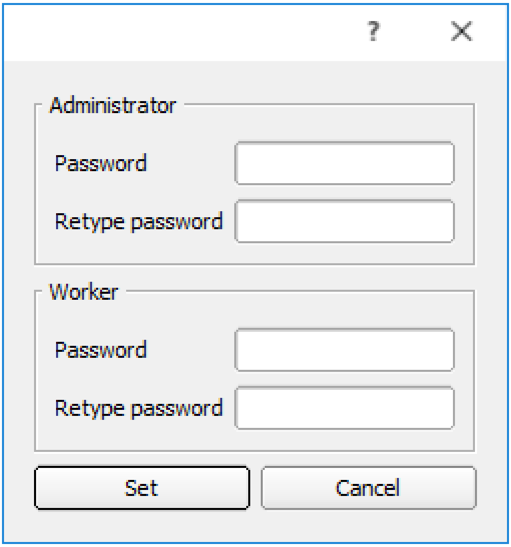


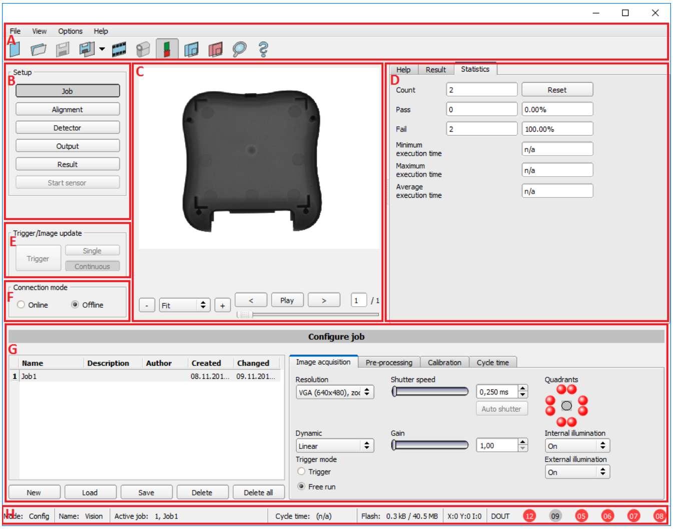
Fig. 32: Password input

Allocating an empty password means the password can be confirmed without any further entry. Activation of the "Deactivate password request" checkbox, permanently deactivates password request.

If passwords have been assigned and then forgotten, it is possible to reset passwords to delivery status by reinstalling the software on the local PC.

4.3 SensoConfig: Setting sensor, Job

With this program, you can configure your VISOR® vision sensor for one or several jobs in six simple logical operating steps.



Count 2

Pass 0 0.00%

Fail 2 100.00%

Minimum execution time n/a

Maximum execution time n/a

Average execution time n/a

○ Online ● Offline

Fit 1 / 1

	Name	Description	Author	Created	Changed
1	Job1			08.11.201...	09.11.201...

Resolution: VGA (640x480), zo(

Shutter speed: 0,250 ms

Dynamic: Linear

Gain: 1,00

Trigger mode: ○ Trigger ● Free run

Internal illumination: On

External illumination: On

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 0.3 kB / 40.5 MB | X:0 Y:0 I:0

Fig. 33: SensoConfig

The fields are:

A) Menu and tool bar

B) Setup Navigation / Operating steps

See next chapter for description

C) Image

Image output with graphically adjustable operating and search zones as well as zoom function also filmstrip navigation when in simulation mode

D) Context

Context-sensitive online help, automatically updated for each action.

E) Image acquisition mode

Switch-over between continuous (free run) and single image mode with trigger input (either from sensor or via onscreen button)

F) Connection mode

Switch-over between online and offline mode (sensor present or simulation without sensor)

G) Job selection

Changing variable content relating to action in set-up navigation, for setting of associated parameters.

H) Status bar

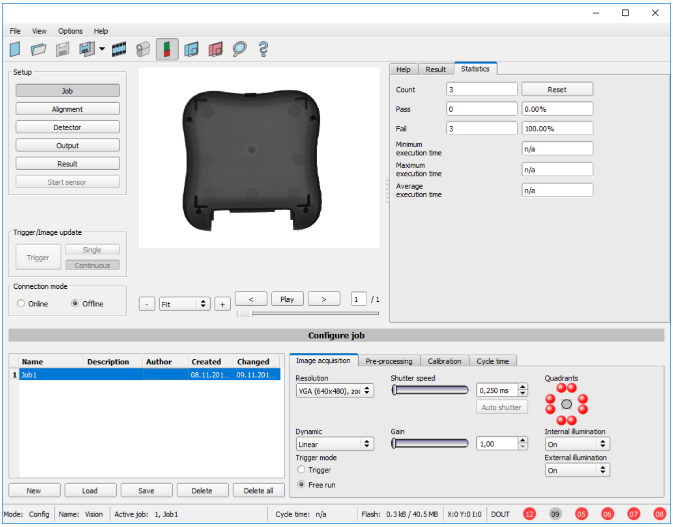
Different status information including Mode / Name of VISOR® / Active job. In Run Mode: Cycle time / cursor x/y location and pixel intensity / individual I/O on /off indication (like configured in "Output/Digital output").

4.3.1 Job Setup

Configuring a job

To configure a job, edit the job entry in the "Select job" (G) field or e.g. create a new job. Set global parameters here, such as shutter, exposure or the resolution which is valid for the entire job.

For Job- setup: in Setup/Job edit or generate a new job in field "Jobs" (G).



Connection mode: ○ Online ● Offline

1 / 1

Count	3	
Pass	0	0.00%
Fail	3	100.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Name	Description	Author	Created	Changed
1	Job1			08.11.201...	09.11.201...

Resolution: VGA (640x480), zoo

Shutter speed: 0,250 ms

Dynamic: Linear

Gain: 1,00

Trigger mode: ○ Trigger ● Free run

Internal illumination: On

External illumination: On

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: n/a | Flash: 0.3 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 34: SensoConfig Job

One job contains all settings and parameters necessary to perform a specific inspection task.

Jobs are created here, and several jobs can be stored in the VISOR®. All global settings, valid for each individual job, e.g. shutter, gain, illumination settings etc. are also carried out here.

- The following basic image settings should first be made to ensure a high-contrast and sharp image:
  - Image brightness: Set shutter or amplification, see Job/Image acquisition
  - Image sharpness: Focus setting via the screw on the back of the VISOR® camera itself

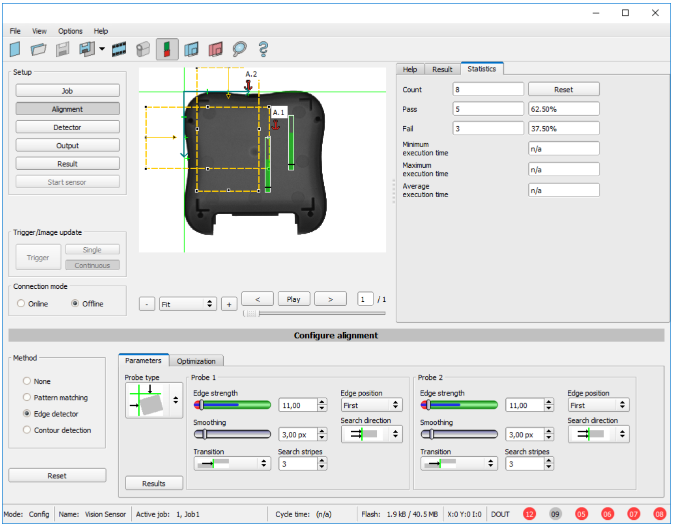
- When delivered, the factory settings are trigger mode = “free run” (see Job/Image acquisition) and image acquisition mode = “continuous”. A new image is continuously displayed for easier focus and brightness set up.
- The subsequent setting of alignment and detectors should preferably be carried out in single image mode, as all settings are then based on a master image and image collection is not continuously carried out.
- Alignment and multiple different detectors (depending on the type of sensor max. 32 or 255 detectors) can subsequently be defined within one job to solve an inspection task.

There is the possibility to save a job as a template. To do this, right-click on the job in the job list and select “Save as template”. For each new job, the settings and detectors are then copied from the job template. In the job list, the job template is identified with a “T” (Template). The job template cannot be edited. To remove the job template, right-click on the template and select “Remove”.

4.3.2 Alignment settings

Alignment compensation can be necessary for objects whose position varies on the screen.

Three different detection methods (alignment detectors) are available for this purpose, pattern matching, edge detection and contour. Alignment is optional. After selection of the alignment method, set the working zones on the parameter to be used for alignment tracking by adjusting the graphic frame to the appropriate position and size on the image. The associated parameters are displayed on the bottom right-hand side and can also be adjusted there. Alignment, when used, affects the positions of all the detectors subsequently defined in this job. In this example, the outside contour is used for alignment and the plug can be found either by contour or by pattern matching. If the angular rotation of the object can vary also, the contour method must be used.



Count: 8
Pass: 5 — 62.50%
Fail: 3 — 37.50%
Minimum execution time: n/a
Maximum execution time: n/a
Average execution time: n/a

Connection mode: ○ Online ● Offline

Fit — 1 / 1

Method: ○ None ○ Pattern matching ● Edge detector ○ Contour detection

Parameters | Optimization

	Probe 1	Probe 2
Edge strength	11,00	11,00
Smoothing	3,00 px	3,00 px
Search stripes	3	3
Edge position	First	First

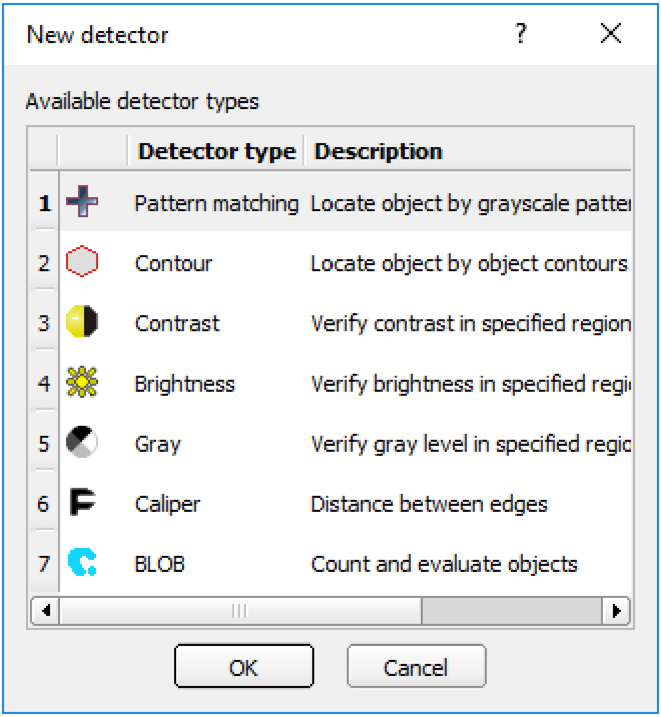
Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 1.9 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 35: SensoConfig Alignment

SensoConfig Alignment

4.3.3 Detector settings

Different detectors can be selected and adjusted to solve an inspection task. First the required detector is selected in the dialog box shown below.



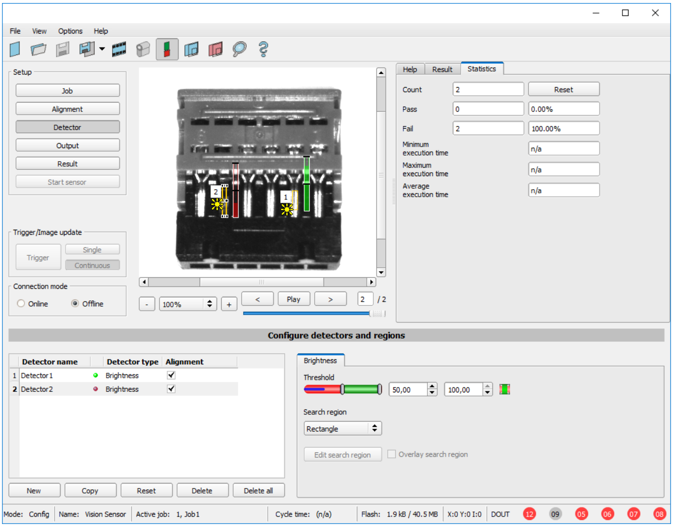
Available detector types

		Detector type	Description
1	✚	Pattern matching	Locate object by grayscale patte
2	⬡	Contour	Locate object by object contours
3	◐	Contrast	Verify contrast in specified region
4	✲	Brightness	Verify brightness in specified regi
5	◑	Gray	Verify gray level in specified regic
6	F	Caliper	Distance between edges
7	C	BLOB	Count and evaluate objects

Fig. 36: Detector list, Object sensor

Then the working and search zones are graphically set on the screen. If “teach zones” (red outline) exist, they are taught immediately after completion of the settings. All the detectors defined in this job are shown in the bottom left-hand corner. The parameters of the currently selected detector are shown in the bottom right-hand corner and can be adjusted there. If other parameters are to be checked on the same part, many other detectors can be created as described above by clicking on “New”. In the example two brightness detectors are defined to check the presence of metal contacts in a plastic connector housing.

- Detector 1: contact found (brightness value is in defined range as the shiny metal contact is mounted) result positive.
- Detector 2: contact not found (brightness value out of defined range, as only weak reflection from the black plastic housing background) result negative.



Count	2
Pass	0 / 0.00%
Fail	2 / 100.00%
Minimum execution time	n/a
Maximum execution time	n/a
Average execution time	n/a

	Detector name		Detector type	Alignment
1	Detector1	●	Brightness	☑
2	Detector2	●	Brightness	☑

Threshold: 50,00 – 100,00

Search region: Rectangle

Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 1.9 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

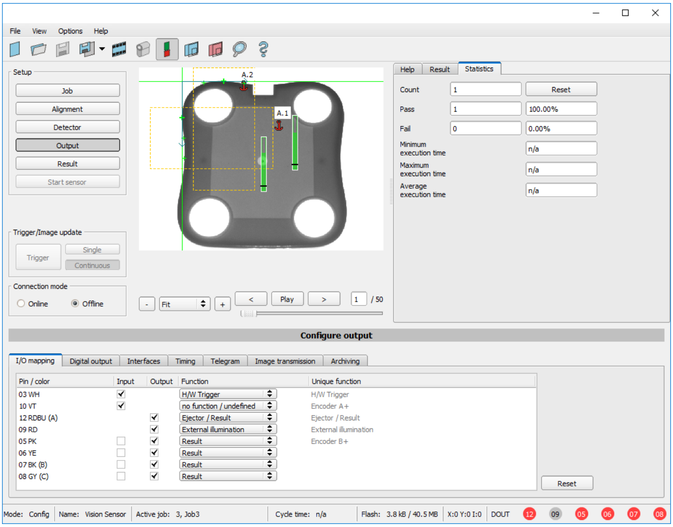
Fig. 37: Detector settings

4.3.4 Output, I/O and data output

The output module enables different settings of digital inputs/outputs and data output.

Select and activate the interfaces in the different tabs. Logically connect detector results and assign to the available I/O´s.

In order to enable the output of serial result data, select the required interface and compose data string.



Count	1	
Pass	1	100.00%
Fail	0	0.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

Pin / color	Input	Output	Function	Unique function
03 WH	☑		H/W Trigger	H/W Trigger
10 VT	☑		no function / undefined	Encoder A+
12 RDBU (A)		☑	Ejector / Result	Ejector / Result
09 RD		☑	External illumination	External illumination
05 PK	□	☑	Result	Encoder B+
06 YE	□	☑	Result	
07 BK (B)	□	☑	Result	
08 GY (C)	□	☑	Result	

Mode: Config | Name: Vision Sensor | Active job: 3, Job3 | Cycle time: n/a | Flash: 3.8 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

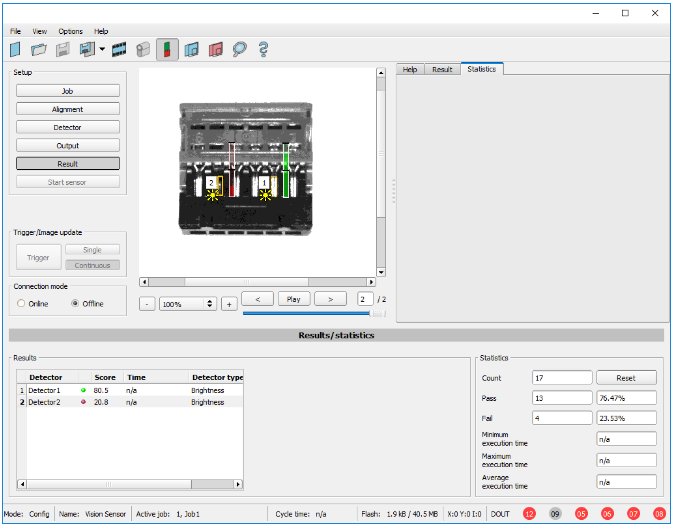
Fig. 38: Output, digital and data

Setting possibilities in the different tabs:

- **I/O mapping**
  Settings for the I/O Hardware configuration
- **Digital Output**
  Selection of digital signal outputs and definition and assignment of logical connection using the Boolean results of all detectors. Definition of complex logic connections via table or via input of a logical formula.
  A different logical connection can be assigned to each available digital output.
- **Interfaces**
  Selection, setting and activation of the individual interfaces
- **Timing**
  Setting of delay times: trigger delay, result delay and duration of result
- **Telegram**
  Setting and preview of data output string via RS422 or Ethernet
  Selection of: binary or ASCII protocol, header and/or trailer, standard contents and/or flexible, combinable, special individual data from the individual detectors.
  Any number of individual results from all the defined detectors can be freely arranged in an output string.

4.3.5 Result

With this function, an inspection is carried out on the PC for control purposes, using all the settings made. All the results are produced and displayed just as on the sensor. However e.g. execution times will not be updated as these values are only informative when implemented on the sensor itself. See next step: “Start Sensor”.



Connection mode: ○ Online ● Offline

100% 2 / 2

Results

	Detector		Score	Time	Detector type
1	Detector1	●	80.5	n/a	Brightness
2	Detector2	●	20.8	n/a	Brightness

Statistics

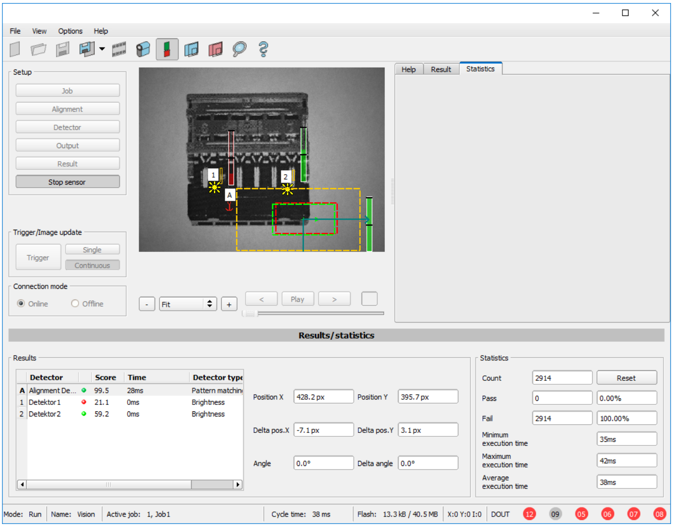
Count	17	Reset
Pass	13	76.47%
Fail	4	23.53%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: n/a | Flash: 1.9 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 39: Result display

4.3.6 Start sensor

When this function is activated, all settings are transferred to the sensor, stored in the flash memory and carried out in e.g. in free run or in triggered mode according to the settings made. All information in the list of detectors, result field or under “Statistics” is updated here. If using “triggered mode” then a trigger will be required from the external control system, alternatively a ‘software’ trigger can be sent using the “Trigger” button the left hand side of the image area.



Connection mode: ● Online ○ Offline

Results

	Detector		Score	Time	Detector type
A	Alignment De...	🟢	99.5	28ms	Pattern matchin
1	Detektor1	🔴	21.1	0ms	Brightness
2	Detektor2	🟢	59.2	0ms	Brightness

Position X: 428.2 px
Position Y: 395.7 px
Delta pos.X: -7.1 px
Delta pos.Y: 3.1 px
Angle: 0.0°
Delta angle: 0.0°

Statistics

Count	2914	Reset
Pass	0	0.00%
Fail	2914	100.00%
Minimum execution time		35ms
Maximum execution time		42ms
Average execution time		38ms

Mode: Run | Name: Vision | Active job: 1, Job1 | Cycle time: 38 ms | Flash: 13.3 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 40: Start sensor

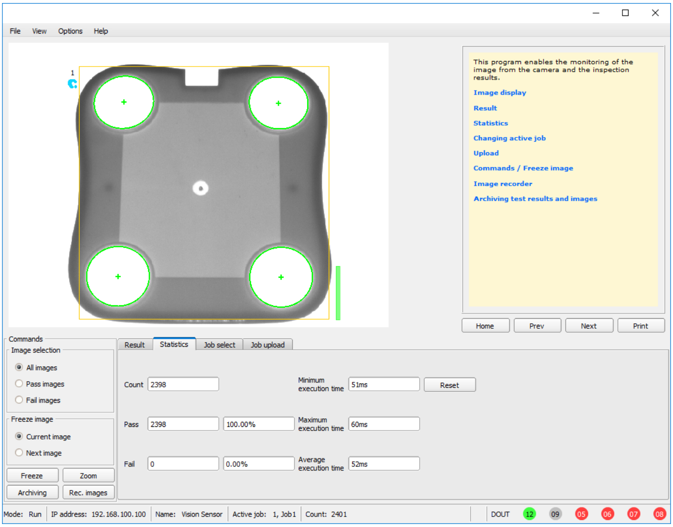
4.4 SensoView, display images and results

This program enables the monitoring/inspection of the connected sensor and the analysis of inspection results.

Click the "**View**" button in the SensoFind software to start the SensoView module. (You can open multiple copies of this software if you are using multiple cameras on the system, however only one 'connection' is allowed to each VISOR® vision sensor).

The current image is displayed with the drawings for alignment and the detectors (if "image transmission = active" is activated in the configuration module under Job/General).

The tab "**Result**" shows the individual detectors with their results and the overall result.
The tab "**Statistics**" shows further statistical results.
The "**Freeze image**" button enables result-controlled images (e.g.: bad part) to be kept on the display.
"**Zoom**" enlarges images.
With "**Archive images**", images and result data, as previously set under "File/Configure archiving", can be archived on the hard disk of a connected PC, with or without numerical result data.
With "**Rec. images**" the last 10 images can be retrieved from the VISOR® vision sensor.
In the tab "**Job**", it is possible to switch between jobs present on the sensor.
In the tab "**Upload**", further, previously defined jobs or whole job sets can be loaded from the viewer on to the sensor.



This program enables the monitoring of the image from the camera and the inspection results.

Count: 2398
Pass: 2398, 100.00%
Fail: 0, 0.00%
Minimum execution time: 51ms
Maximum execution time: 60ms
Average execution time: 52ms

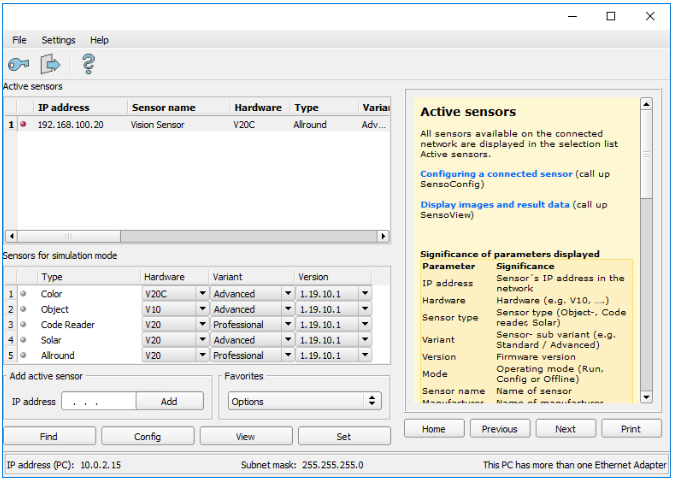
Mode: Run | IP address: 192.168.100.100 | Name: Vision Sensor | Active job: 1, Job1 | Count: 2401

Fig. 41: SensoView

4.5 VISOR® – Operating- and configuration software – SensoFind, all functions

In this program you can select a sensor or sensor simulation for configuration or display (monitoring) and carry out different basic settings:

- Active sensors (Page 69)
- Sensors for simulation mode (Page 71)
- Find / Add active sensor (Page 72)
- Configuring a connected sensor (Page 72)
- Display images and result data (Page 72)
- Sensor's network settings (Page 73)
- Update / Firmware update (Page 73)
- User administration / Passwords (Page 74)
- Favorites (Page 76)
- Auto Start Up (Page 79)



Active sensors

		IP address	Sensor name	Hardware	Type	Variai
1	●	192.168.100.20	Vision Sensor	V20C	Allround	Adv...

Sensors for simulation mode

		Type	Hardware	Variant	Version
1	●	Color	V20C	Advanced	1.19.10.1
2	●	Object	V10	Advanced	1.19.10.1
3	●	Code Reader	V20	Professional	1.19.10.1
4	●	Solar	V20	Advanced	1.19.10.1
5	●	Allround	V20	Professional	1.19.10.1

Add active sensor

IP address . . . Add

Favorites

Options

Active sensors

All sensors available on the connected network are displayed in the selection list Active sensors.

Configuring a connected sensor (call up SensoConfig)

Display images and result data (call up SensoView)

Significance of parameters displayed

Parameter	Significance
IP address	Sensor´s IP address in the network
Hardware	Hardware (e.g. V10, ...)
Sensor type	Sensor type (Object-, Code reader, Solar)
Variant	Sensor- sub variant (e.g. Standard / Advanced)
Version	Firmware version
Mode	Operating mode (Run, Config or Offline)
Sensor name	Name of sensor
Manufacturer	Name of manufacturer

IP address (PC): 10.0.2.15 Subnet mask: 255.255.255.0 This PC has more than one Ethernet Adapter

Fig. 42: SensoFind

If the “Configure” function is not accessible (button inactive), login (button with door- / arrow- symbol) with password entry is required. If you do not know the password, please contact the administrator.

4.5.1 Active sensors

All sensors available on the connected network are displayed in the selection list Active sensors.

Configuring a connected sensor (Page 72) (call up SensoConfig)

Display images and result data (Page 72) (call up SensoView)

Significance of parameters displayed

Parameter	**Significance**
IP address	Sensor´s IP address in the network
Hardware	Hardware (e.g. V10, ….)
Sensor type	Sensor type (Object-, Code reader, Solar)
Variant	Sensor- sub variant (e.g. Standard / Advanced)
Version	Firmware version
Mode	Operating mode (Run, Config or Offline)
Sensor name	Name of sensor
Manufacturer	Name of manufacturer
Mac-Address	Sensor´s Mac address
Subnet mask	Sensor´s subnet mask
Gateway	Standard gateway
DHCP	DHCP active / inactive
Operating system	Type of operating system
Operating System Version	Version of operating system
Platform	e.g. VISOR®
Hardware version	Hardware version
RAM	RAM size
Flash	Flash size

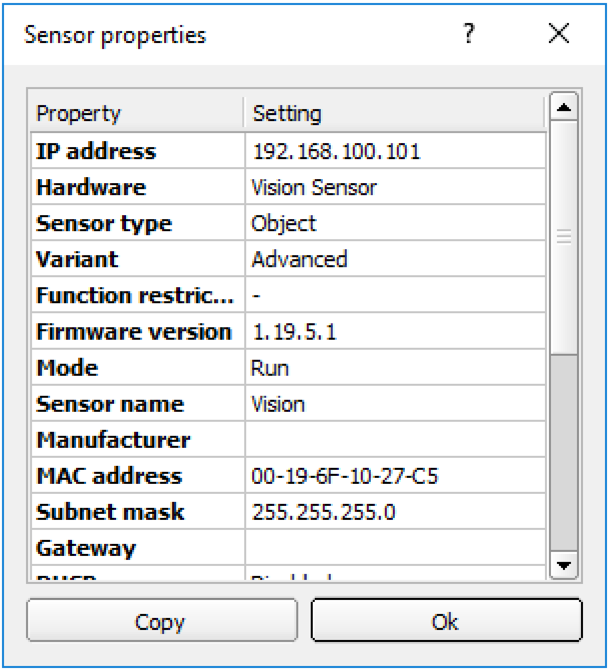
If the “Configure” function is not accessible (button inactive, grayed out), login with password entry is required. If you do not know the password, please contact your site system administrator.

Information:

- If no entries are shown in the list, even though a sensor is connected, you can refresh the list with the “Find” button or manually “Add” the IP address of the VISOR® product.

- If no sensor is connected, simulations of different sensor applications are available in the Sensors for simulation mode (Page 71) list such as 'Object' sensor.

Via the button “details” (at the right, upper corner of the parameter list of “Active Sensors”) a detailed list of all VISOR® parameters is accessible.



Sensor properties

Property	Setting
IP address	192.168.100.101
Hardware	Vision Sensor
Sensor type	Object
Variant	Advanced
Function restric...	-
Firmware version	1.19.5.1
Mode	Run
Sensor name	Vision
Manufacturer	
MAC address	00-19-6F-10-27-C5
Subnet mask	255.255.255.0
Gateway	

Fig. 43: Sensor properties

4.5.2 Sensors for simulation mode

In order to access the simulation mode, select the required sensor type with a double click and press Configuring a connected sensor (Page 72)button (call up SensoConfig).

Significance of parameters displayed

Parameter	Significance
Type	Sensor type (e.g. Object , Code reader, Solar...)
Hardware	Hardware type (e.g. resolution, monochrome or color version)

Parameter	Significance
Version	Firmware version
Variant	Sensor- sub variant (e.g. Advanced ...)

If the function “Config” is not accessible (button inactive) a Login (button with door / arrow symbol) with password input is necessary. If you do not know the password please contact your administrator.

4.5.3 Find / Add active sensor

If no sensors are shown in the list Active sensors, even though a sensor is connected, please follow these steps:

Find / search sensor:

To search for sensors which are connected directly to the PC, or which are available in the network, click button “Find”. Basic understanding of PC networking is required this is not covered within the scope of supply from SensoPart .

Add active sensor:

If you know the IP-address of a sensor, please enter it into the field IP-address and click button “Add”.

Now the sensor appears in the list and can be accessed for e.g. Config or View.

If the function “Config” is not accessible (button not active / grayed out) a Login with password input is necessary. If you do not know the password please contact your site systems administrator.

4.5.4 Configuring a connected sensor

Mark a sensor (simulation) in the list and click on the “Config” button. The configuration program SensoConfig is called up and the jobs currently stored on the sensor are shown in the selection list. When SensoConfig is called up, you may be required to enter a password. See User administration / Passwords (Page 74) for defining passwords.

See chapter: VISOR® – Operating- and configuration software – SensoConfig, all functions

4.5.5 Display images and result data

Mark a sensor in the list and click on the “View” button. The SensoView program is opened up and images and measurement results from the active jobs are displayed on screen.

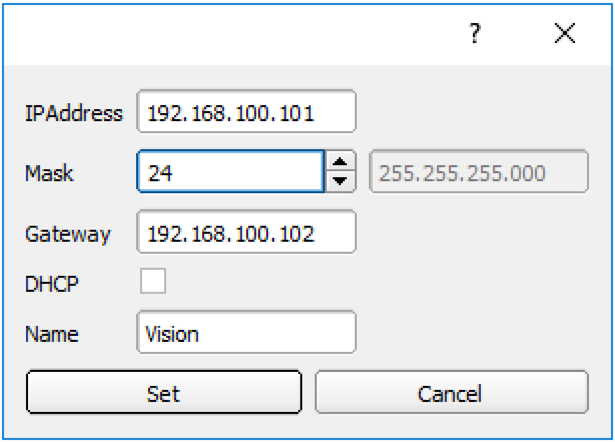
Information:

Calling up SensoView does not affect operation of the selected sensor.

See chapter: VISOR® – Operating- and configuration software – SensoView, all functions

4.5.6 Sensor's network settings

You can change the network settings of the selected sensor with the Set button. The IP address, subnet mask, standard gateway, DHCP and sensor name can be set here. The PC's IP address and subnet mask are displayed below in the SensoFind status bar. The address structure must be correct in order to be able to connect the sensor to the PC. The sensor's IP address etc. can therefore be modified accordingly here if necessary. Please contact your site administrator for the definition of network parameters. Further information on this subject can be found in the chapter Network settings, Short reference (Page 48) and Network connection (Page 377). If "DHCP = active" is selected, a unique name must be given for the sensor as the IP address is newly assigned each time the sensor starts up and can thus change. You require administrator authorisation for these functions (see user administration).



IPAddress: 192.168.100.101

Mask: 24 — 255.255.255.000

Gateway: 192.168.100.102

DHCP: □

Name: Vision

Fig. 44: SensoFind, IP- Setup

See chapter: Network settings, Short reference and Network connection

4.5.7 Update / Firmware update

You can update the firmware of the selected sensor through the menu item "SensoFind/File/Update" (see following figure). The appropriate firmware update file must first have been obtained via download from the SensoPart website or from SensoPart Support. Select the appropriate firmware file in the file dialogue box that opens and follow the instructions. Do not disconnect the power to the sensor during this process unless prompted by the onscreen instructions.

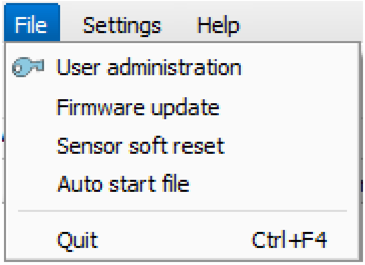


Fig. 45: SensoFind, Firmware update

Please note: Before executing the firmware update please create a current backup! Use the menu item “SensoConfig/File/Save job set (Backup) ...” to save the job sets (see following figure).

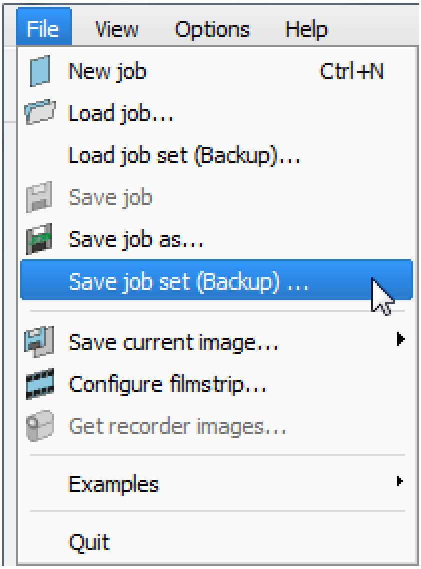
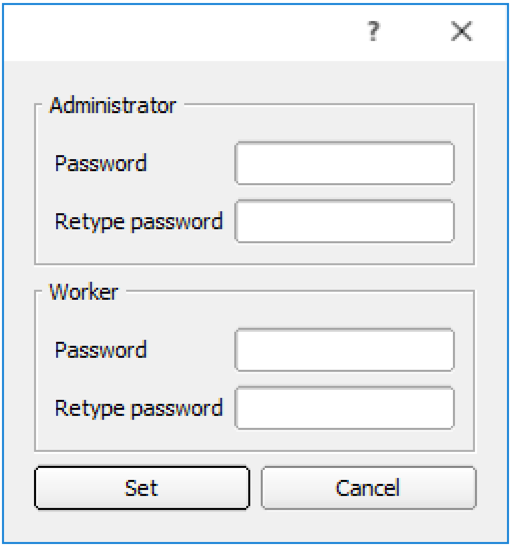


Fig. 46: Backup, save job set

4.5.8 User administration / Passwords

The VISOR® configuration distinguishes between three user groups, which have different authorisations:



Administrator

Password

Retype password

Worker

Password

Retype password

Fig. 47: SensoFind, Password input

Password level	SensoFind	SensoConfig	SensoView
Administrator password	all functions	all functions	all functions
Worker password	all functions except - Config - Settings - Update	none	all functions, including Job Upload and Image Recorder
User (without any password)	all functions except - Config - Settings - Update	none	only display of images, inspection results and statistics

After software installation, login is automatically carried-out when the application is called-up, without password request. No passwords are assigned.

Define passwords:

Select file user administration in the File menu or click on in the toolbar to assign passwords for the administrator and user categories. Once a password has been entered, a logout is automatically carried out, i.e. input of the new password is now necessary. Assigning an “empty” password, enables entry by simply confirming with OK.

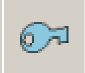


Fig. 48: Password button

Login

Once passwords have been assigned and automatic logout has taken place, a login is required e.g. for sensor configuration. Click on the Login-button in the tool bar to login and / or (after password entry) to deactivate password entry for the next session for the selected user group.

If the “deactivate password request” box is ticked, the password will not be requested when the application is next started.

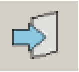
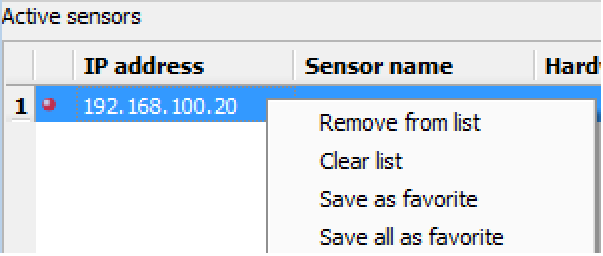


Fig. 49: Login-button

4.5.9 Favorites

The favorites are used to quickly access and manage the VISOR® vision sensors. The following parameters can be selected for the favorites.

Right-click on active sensors in SensoFind:



Active sensors

		IP address	Sensor name	Hard
1	●	192.168.100.20		

Fig. 50: Parameter favorites with right-click on active sensors

Parameter	Function
Remove from list	Removes the selected sensor from the “Active sensors” list.

Parameter	Function
Clear list	Clears the complete list “Active sensors”.
Save as favorite	The selected sensor is saved as a favorite.
Save all as favorite	Saves all sensors in the “Active sensors” list as favorites.

“Favorites” in SensoFind:

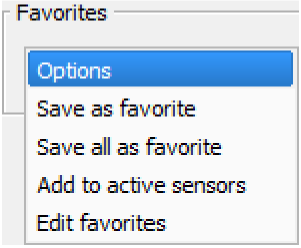
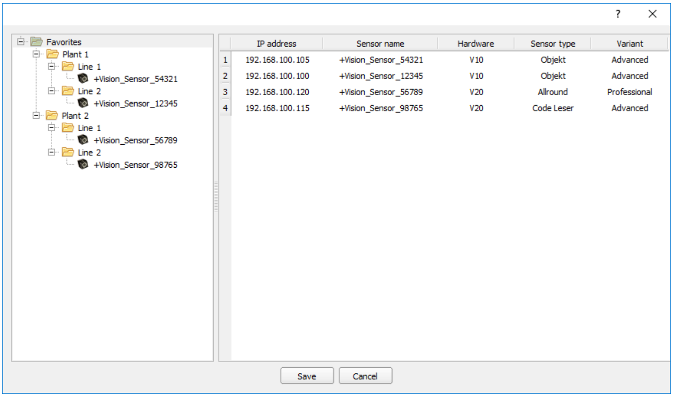


Fig. 51: Options Favorites

Parameter	Function
Save as favorite	Opens the “Save as favorite” window where a desired location can be selected in the tree structure in which the sensor from the “Active sensors” list is to be saved as favorite.
Save all as favorite	Opens the “Save all as favorite” window where a desired location can be selected in the tree structure in which all the sensors from the “Active sensors” list are to be saved as favorite.
Add to active sensors	Opens the “Add to active sensors” window where a sensor/ sensor group can be selected that is to be added to the “Active sensors” list.
Edit favorites	Opens the “Edit favorites” window in which the sensor groups can be edited.

Edit favorites - create groups

In the left window area, the sensors are divided into groups via a tree structure, e.g. according to production sites and production lines. In the right window area, the sensors below a selected group are listed in tabular form, e.g. group “Favorites” shows all sensors.



- Favorites
  - Plant 1
    - Line 1
      - +Vision_Sensor_54321
    - Line 2
      - +Vision_Sensor_12345
  - Plant 2
    - Line 1
      - +Vision_Sensor_56789
    - Line 2
      - +Vision_Sensor_98765

	IP address	Sensor name	Hardware	Sensor type	Variant
1	192.168.100.105	+Vision_Sensor_54321	V10	Objekt	Advanced
2	192.168.100.100	+Vision_Sensor_12345	V10	Objekt	Advanced
3	192.168.100.120	+Vision_Sensor_56789	V20	Allround	Professional
4	192.168.100.115	+Vision_Sensor_98765	V20	Code Leser	Advanced

Fig. 52: Group configuration

The favorites are stored in the installation path of the VISOR® vision sensor on the PC as an XML file. The file is located under: “SensoPart /VISOR® vision sensor/SensoFind/Data”. It can be exchanged between different PCs.

Examples of applying the favorites:

Example 1:

VISOR® vision sensors which are integrated in different networks can be viewed and managed locally in SensoFind (see also the following figure). The sensors can be added to the “Active sensors” list by entering the IP address in the field “Add active sensor”. The sensors are subsequently managed via the favorites. The sensors can be added to favorites by “SensoFind/Favorites/Saves as favorite”. Within the favorites the sensors can be assigned to different groups.

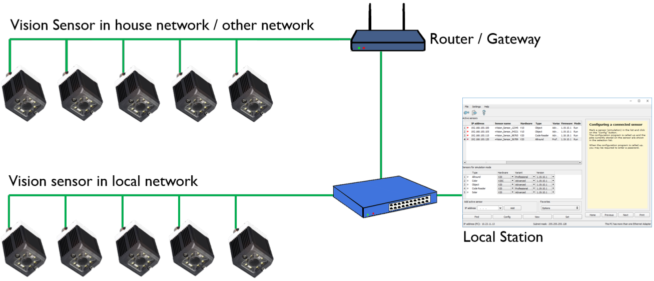

Vision Sensor in house network / other network
Router / Gateway
Vision sensor in local network
Local Station

Fig. 53: Example 1 - VISOR® vision sensors in different networks

Example 2:

Multiple stations are on the same local network. All users have access to all VISOR® vision sensors, although only a few VISOR® vision sensors are relevant for their work (see also the following figure). In conjunction with the “Auto Start Up” function (see also Auto Start Up (Page 79)), it is possible that only a certain selection of VISOR® vision sensors (favorites) is displayed. To do so, the sensors must be added to the favorites and divided into groups. Subsequently a group of favorites can be selected in the Auto Start Up file. The users now only have access to the relevant sensors when opening SensoFind via the Auto Start Up file.

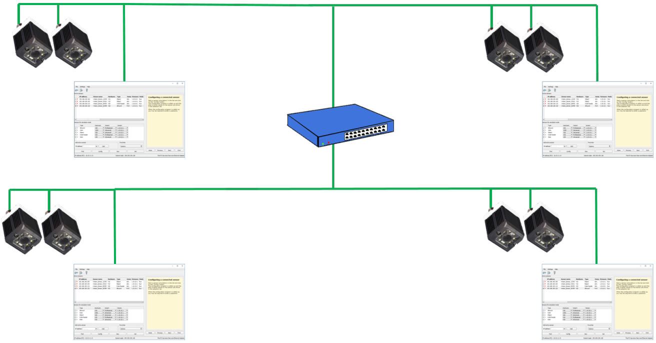


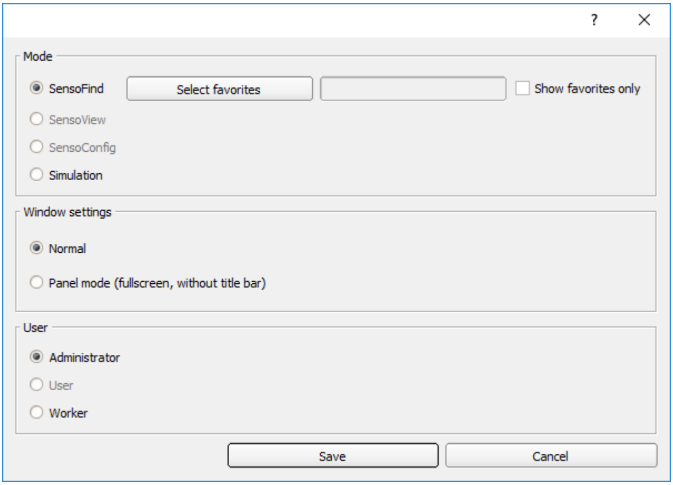
Fig. 54: Example 2 - Favorites in the Auto Start Up file

4.5.10 Auto Start Up

Auto Start Up enables the automatic start of the VISOR® vision software. For this purpose, a batch file is created, which can be stored in the windows system folder “Startup” so that it can be called up automatically every time the PC is started. The Auto Start up file window is divided into the areas: mode, window settings and user.

Sequence

1. Open the Auto Start Up file in the SensoFind module with the file path: SensoFind/File/Auto Start Up file.
2. In the “Mode” area, determine the modules of the VISOR® vision software that are to be started automatically.
3. In the window settings, select the view of the module: Normal or panel mode (fullscreen, without title bar).
4. In the “User” area, define the user for the Auto Start Up file. For more information about the authorization function see User administration / Passwords (Page 74).
5. Select the “Save” button and save the batch file (.bat) to the desired destination. For an automatic start when the PC boots, the file must be stored in the windows system folder “Startup”.
6. Close the VISOR® vision software.
7. Execute the batch file. The VISOR® vision software is started according to the settings.



Mode

● SensoFind [Select favorites] [ ] □ Show favorites only

○ SensoView

○ SensoConfig

○ Simulation

Window settings

● Normal

○ Panel mode (fullscreen, without title bar)

User

● Administrator

○ User

○ Worker

Fig. 55: Auto Start Up file

The following parameters can be configured in the “Auto Start Up” window:

Mode	
Parameter	**Function**
SensoFind	Modules of the VISOR® vision software which are to be opened automatically in the Auto Start Up file. For the start of the simulation mode, the model variant which is currently selected in SensoFind (marked in blue) is used.
SensoView	
SensoConfig	
Simulation	
Select favorites	With this parameter, a favorite group can be added to the Auto Start Up file.
Show favorites only	If the parameter “Select favorites” is selected, the “Active sensors” list is cleared and then only filled with the selected favorites.

Window settings	
Parameter	**Function**
Normal	The selected VISOR® vision software module is opened normally in the Auto Start Up file with the title bar.

Window settings	
Parameter	**Function**
Panel Mode (full-screen, without title bar)	The selected VISOR® vision software module is opened in the Auto Start Up file in fullscreen without a title bar. Typical application for touch screen panel PCs.

User	
Parameter	**Function**
Administrator	The selection of the user depends on the rights that the user should have within the Auto Start Up file. For more information about the authorization function see User administration / Passwords (Page 74).
User	
Worker	

4.6 VISOR® – Operating- and configuration software – SensoConfig, all functions

With this programme, you can configure your VISOR® vision sensor for one or several jobs in six logical operating steps.

- Jobs (Inspection tasks) (Page 82)
- Alignment (Page 120)
- Detectors (Page 145)
- Output of inspection results (Page 275)
- Result (Page 304)
- Start sensor (Page 314)

Other program functions:

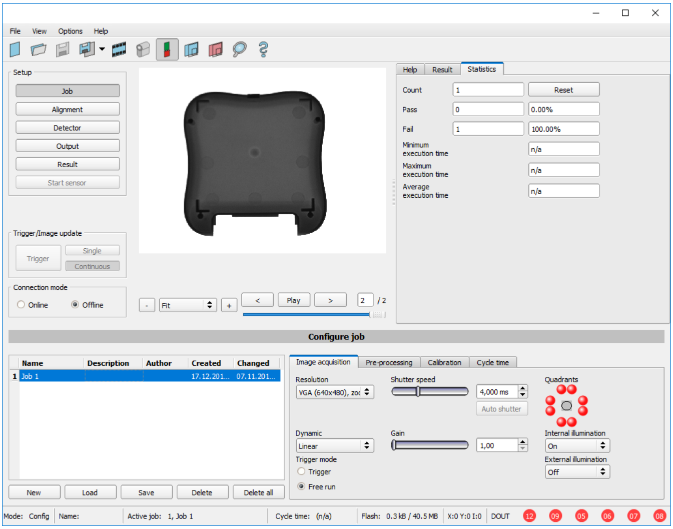
- Trigger settings (Page 315)
- Connection mode: Switching between Online and Offline mode (Page 316)
- Simulation of jobs (offline mode) (Page 316) using series of images
- Creating filmstrips (Page 316) Image recording for analysis or simulation purposes. Use of SensoConfig may require password entry (administrator user group). See User administration / Passwords (Page 74)
- Image recorder (Page 328)

To obtain a continuously updated live image even without trigger, carry out the following (if necessary temporary) settings:

- Set to **free run** in “Job/Image acquisition”
- Set to **continuous** in “Trigger / collect image” User interface and operating procedure

4.6.1 Jobs (Inspection tasks)

A job contains all the settings and parameters required to carry out a certain inspection task.



Count: 1

Pass: 0 — 0.00%

Fail: 1 — 100.00%

Minimum execution time: n/a

Maximum execution time: n/a

Average execution time: n/a

	Name	Description	Author	Created	Changed
1	Job 1			17.12.201...	07.11.201...

Resolution: VGA (640x480), zo(

Shutter speed: 4,000 ms

Dynamic: Linear

Gain: 1,00

Trigger mode: ○ Trigger ● Free run

Internal illumination: On

External illumination: Off

Mode: Config | Name: | Active job: 1, Job 1 | Cycle time: (n/a) | Flash: 0.3 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 56: SensoConfig Job

4.6.1.1 Creation, modification and administration of jobs

A selected job (marked in the list) can be modified by entering parameters in both tabs of the configuration window:

If there is no job entry in the list, you must create a new job first.

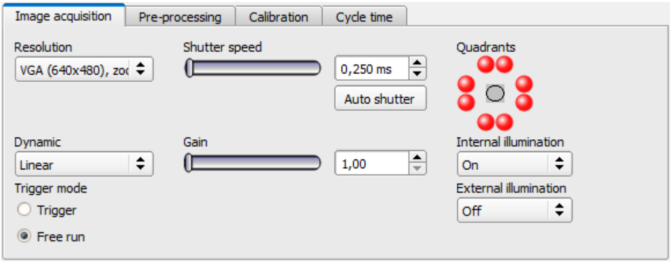
Creating a new job:

1. Click on the button “New” underneath the job selection list. A new job entry appears in the list.
2. Edit the entry with a double click on the respective line (Name, Description, Author):

Further functions:

Function	Description
New	Defines a new job
Load	Loads a job from the PC
Save	Saves the selected job on the PC
Delete	Deletes the selected job from the list
Delete all	Deletes all the jobs in the list

All the functions described can also be carried out using the File menu.



Resolution: VGA (640x480), zo(

Shutter speed: 0,250 ms

Dynamic: Linear

Gain: 1,00

Internal illumination: On

External illumination: Off

Trigger mode: ○ Trigger ● Free run

Fig. 57: SensoConfig job list

If the sensor's memory capacity is exhausted and no further jobs can be loaded on to the sensor, the color of the remaining memory display in the status bar changes to red.

4.6.1.2 Loading and saving jobs and job sets

Jobs can be loaded and stored individually or as a whole set of jobs in a job set. If several jobs are stored on the sensor, they form a job set, which you can store as an XML file on your PC or on an external storage medium just like an individual job.

Saving a job / job set:

1. Select Save job as ... from the File menu.
2. Select Save job set (Backup) ... from the File menu.

Loading a job / job set:

1. Select “Load job ...” or “Load job set (Backup) ...” from the File menu.
2. Activate the button “Start Sensor” to transfer jobs to the sensor.

All the jobs stored on the sensor are deleted when a new job / job set is loaded !

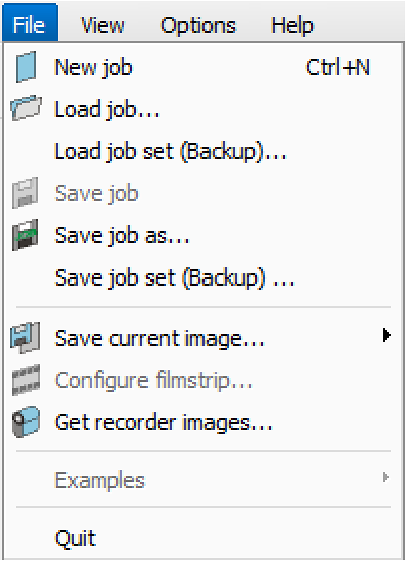


Fig. 58: SensoConfig, Load / save job

4.6.1.3 Parameters for image acquisition

The basic parameters for image acquisition are determined in the tab Image acquisition.

Set image sharpness with the focus setting screw on the back of the VISOR®.

Parameters	**Functions and setting possibilities**
Resolution	Standard resolution is VGA (640x480), but a lower resolution (QVGA) can be selected with time-critical applications or for compatibility reasons. Available resolutions: V10: WVGA (736x480), VGA (640x480), QVGA (320x240), QQVGA (160x120) V10C: WVGA (736x480), VGA (640x480), QVGA (320x240) V20: SXGA (1280x1024), VGA (640x480), QVGA (320x240) V20C: SXGA (1280x1024), VGA (640x480) When the resolution is altered, all the detectors previously defined are deleted!
Zoom (V20 only)	Via the Zoom function different fields of view / image zones can be selected
Dynamic	Optimization of characteristics of image capturing: “Linear” means

Parameters	**Functions and setting possibilities**
	linear response curve(behaves like VISOR®-products with no dynamic image capturing), “High” means better graduation in bright areas of the image (avoids override).
Trigger mode	Select trigger mode (triggered or free run). In case of triggered mode, trigger can be done by hardware-trigger (Pin 03 WH) or over one of the data interfaces. In free run the VISOR® continuously captures images and processes evaluations.
Shutter speed	Parameter for control of image brightness. Image brightness preferably should be set with “Shutter speed”, only in case that it´s not possible to achieve the required image brightness this way use the slider “Gain” (Default value of Gain = 1). With fast moving objects a high shutter value can cause blurring of the image. Exposure can be set automatically with the Auto-Shutter button. Maximum shutter value is 100ms. Maximum duration of internal illumination pulse is 8ms. Shutter timers longer than 8 ms just make sense, if internal and external illuminations are used.
Gain	Set image brightness preferably with shutter speed first, and only if necessary in a second step with gain. (Default value of Gain = 1).
Quadrants (illumination)	By click on the LED single quadrants of illumination can be switched off. This function may avoid reflections at low working distances.
Internal illumination	Switch internal illumination (on, off).
External illumination	Switch external illumination (on, off, permanent). External illumination is switched over Pin 09 RD.
Multishot (with version “Allround Professional” only)	Activation of function/tab “Multishot”. With this function very small e.g. surface defects, like scratches etc., can be detected. See also: Function Multishot, selection (Page 85)

To obtain a continuously updated live image even without trigger, carry out the following (if necessary temporary) settings:

- Set to **free run** under “Job/Image acquisition”
- Set to **continuous** under “Trigger / collect image”

4.6.1.3.1 Function Multishot, selection

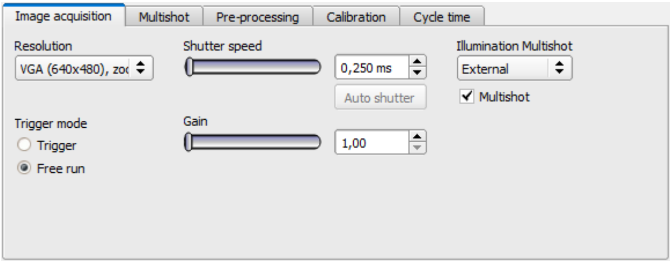
Version “Allround” only!

With the function Multishot four images are taken in one sequence, where each of them are illuminated from a different direction, and then combined into one single image. This way smallest defects in a surface like fine scratches can be detected.

The following conditions must be considered:

1. Lights source should be located from the measurement object so that the maximum shadows are created on the product.
2. Dark regions or shadows should not occur the same in all four images.
3. Overexposed or blurred regions should not occur in all four images.

The parameters of this function can be set in tab “Multishot”.



Resolution: VGA (640x480), zoo

Shutter speed: 0,250 ms

Illumination Multishot: External

☑ Multishot

Trigger mode: ○ Trigger ● Free run

Gain: 1,00

Fig. 59: Selection: Function “Multishot”

4.6.1.3.2 Function Multishot, parameter

Via the function “Multishot” very small defects / smallest height deviations such as scratches or difficult to read DPM Datacodes can be detected.

For the duration of the image capturing of the images (four images in one sequence) the test object must be stationary in relation to the camera sensor.

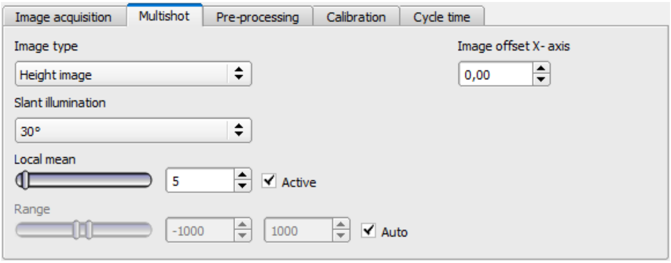


Image type: Height image

Slant illumination: 30°

Local mean: 5 ☑ Active

Range: -1000 1000 ☑ Auto

Image offset X- axis: 0,00

Fig. 60: Function “Multishot”, Parameter

Select “Image type”

Image type	Description
Curvature	Image shows virtual curvature values scaled to gray values
Curvature, absolute	Image shows virtual curvature values, here absolute values only, scaled to gray values
Height	Image shows virtual height values scaled to gray values. Note: “Hight image” causes longer execution time.
Albedo	Image shows virtual reflectivity scaled to gray values
Mean	Average calculated from 4 single images
Combined, absolute	All four single images combined in one image, Use this function to adjust illumination: avoid dark spots or shadows in all 4 single images, avoid overexposed sections with blurring in all 4 single images
Combined, horizontal	All four single images with horizontal image section one over the other displayed in one image . If there was a slight continuous movement of the test object during the four images have been taken, this is visible in the four horizontally combined images and can be compensated with below mentioned function “Image offset X-axis”
East	Single image illumination from east
North	Single image illumination from north
West	Single image illumination from west
South	Single image illumination from south

Image type parameter.

Parameter	Function
Image type	Select Image type, see above.
Slant illumination	Angle of the illumination in relation to measurment plane (0° = flat from side, 90° = perpendicular from top)
Local mean	Local smoothing of height differences. Aid to adjust the height values, if the sensor is not adjusted perpendicular measurement plane.
Range	Range of virtual height ant angle values. Scaling of differences to gray values. With “Auto” this value is automatically determined from the fund minimum and maximum values in the image.

Parameter	Function
Image offset X-axis	If the object was continuously moving during the four single images, , while the image sequence was made, with this, in conjunction with the function “Combined image, horizontal”, the position shift can be compensated.

4.6.1.3.3 Function Multishot, illumination

With Multishot four images are taken of the object in one sequence.

Each of this four images is illuminated from a different direction.

The object must stand still for the duration of this four image sequnence. On the basis of the different reflections across edges a “virtual height image” can be calculated, this pseudo image contains information which is not visible with in the single images.

This technology is especially suitable for:

- detection of defects on flat surfaces like scratches
- reading of punched letters and signs via OCR
- punched Datacodes
- detection of Braille letters

This technology is not suitable for:

- moving objects
- curved surfaces
- 3D-applications - calculations of absolute heights
- Detection of details, which are hidden by other parts of the object, as they can not be illuminated from all four directions.

Important for the correct function is the correct illumination, The object must be illuminated from all four cardinal)North, East, South. West) directions. The sequence of the four image capturing is automatically controlled by the VISOR® vision sensor.

To simplify description the four directions are named North / East / South / West (North at top of the image)

Orientation of illumination:

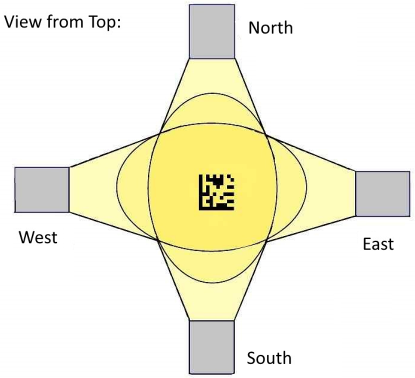


Fig. 61: Multishot, Orientation of illumination

Illumination / connections

Direction	**Output pin (old)**	**Output pin (new)**
East	09	09
South	0 7	06
West	06	07
North	05	08

The correct connection of the illuminations can be checked by function “Combined image, quadrants”, in which all four single images are shown in one image. Place a object in the image which causes a clear shadow (e.g. a screw upright). The images are combined like displayed below.

Left above: illumination from north shadow to south	**Right above:** illumination from east shadow to west
Left below: illumination from west shadow to east	**Right below:** illumination from south shadow to north

Image looks like follows.

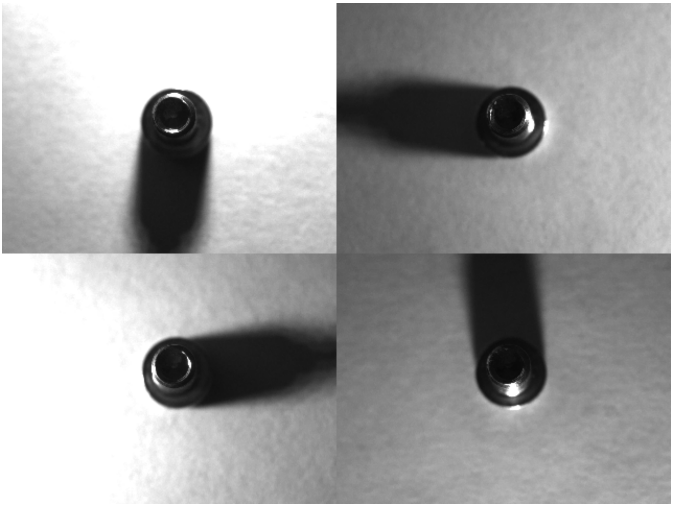


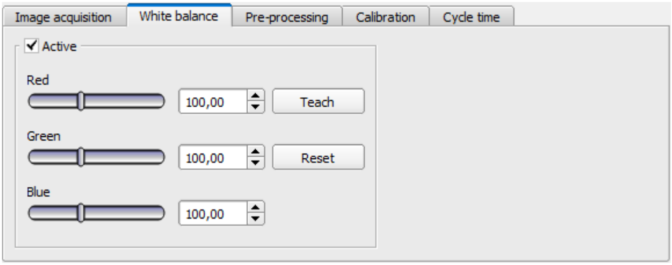
Fig. 62: Multishot, single images

Further advice for illumination

- avoid overexposed regions also as dark shadows
- the SensoPart illumination can be mounted in angle of 30° or 60°
- use 30° angle to illuminate part flat from side (avoids reflections)
- use 60° angle to illuminate parts from above (more reflections)

4.6.1.4 Job, tab White balance

White balance is necessary for compensation of image colors.



☑ Active

Red: 100,00

Green: 100,00

Blue: 100,00

Fig. 63: White balance

Parameter	Function
Red	Mean value of red channel in image
Green	Mean value of green channel in image
Blue	Mean value of blue channel in image
Teach	Execution of white balance, for white balance there has to be a homogeneous, white area below the camera.
Reset	Reset values

4.6.1.5 Preprocessing, Filter for image improvement.

In the Preprocessing tab you can filter and rearrange the images taken by the sensor before analysis.

- Up to 5 filters and one arrangement filter can be used, which are processed in the selected sequence.
- All detectors (alignment and standard detectors) will work with the preprocessed image (not with the original image)
- Especially morphological operations (Dilation and Erosion) can lead to improvements by combining them e.g. by processing Erosion and Dilation one after another – or in reverse order.

Example:- Black points in front of a bright background can be eliminated, if a sequence of dilation and erosion is processed.

The following arrangements are available for image improvement:

Arrangement type	Effect
Rotation 180°	Rotation of image for 180°
Mirror	Vertical mirroring
Flip	Horizontal mirroring

The following filters are available for image improvement:

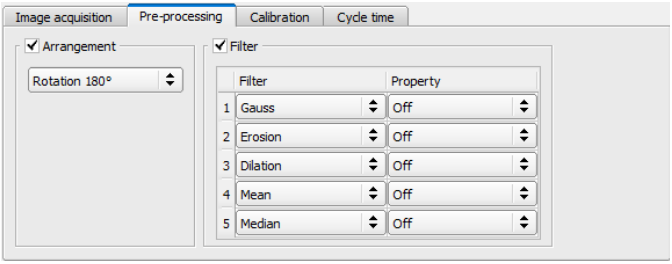
Filter type	Effect
Gauss	Image is smoothed using a gaussian filter mask. This can be applied for reduction of disturbances, suppression of disturbing details and artifacts and smoothing the image.
Erosion	Extension of dark zones, elimination of light pixels in dark zones, elimination of artifacts, division of bright objects. Each

Filter type	Effect
	gray value is replaced by the minimum gray level found inside the filter mask (e.g. 3x3).
Dilation	Extension of light zones, elimination of dark pixels in light zones, elimination of artifacts, division of dark objects. Each gray value is replaced by the maximum gray level found inside the filter mask (e.g. 3x3).
Median	Each gray value is replaced by the median value of the pixels found inside the filter mask (e.g. 3x3). Typical applications include noise reduction, especially for local bright or dark pixels (“salt-and-pepper”-noise).
Mean	Each gray value is replaced by the average gray value of the pixels found inside the filter mask (e.g. 3x3). This can be applied for reduction of disturbances, suppression of disturbing details and artifacts and smoothing the image.
Range	Each gray value is replaced by the range value (maximum gray level − minimum gray level) of the pixels found inside the filter mask (e.g. 3x3). Typical applications include the detection and enhancement of edges and the improvement of local image contrasts. (starting with firmware 1.5.x.x)
Standard deviation	Each gray value is replaced by the standard deviation of the pixels found inside the filter mask (e.g. 3x3). Typical applications include the highlighting of surface defects or edges.
Edge detection (Sobel)	Result image contains edges detected using the Sobel-algorithm (compare image processing literature also). Typical applications include the detection and enhancement of edges and the improvement of local image contrasts or the detection of surface defects.
Multiplication	The gray value of each pixel is multiplied by the choosen multiplier (2x, 4x, 8x, 16x). Values are clipped to 255.
Inversion	Inversion of image

The effect of an active filter is immediately visible in the image. The larger the filter core is selected, the stronger the effect of the filter. The filters are used in the order listed from top to bottom.

Configuring filters:

1. Select the filters in the required order, via the pop-up menus in the column Filter.
2. Enter the size of the filter kernel in the pop-up menu in the column Property. If the setting is "Off", the respective filter is deactivated.



☑ Arrangement

Rotation 180°

☑ Filter

	Filter	Property
1	Gauss	Off
2	Erosion	Off
3	Dilation	Off
4	Mean	Off
5	Median	Off

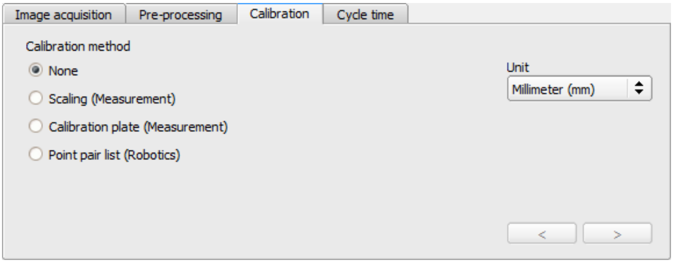
Fig. 64: Tab Job / Preprocessing

4.6.1.6 Calibration

The function "Calibration" transforms the image coordinates (pixel) into world coordinates (e.g. millimeter). When activated all position and distance data is calculated in the selected unit.

4.6.1.6.1 Select the calibration method

The calibration methods are divided into two application areas: "Measurement" and "Robotics".



Calibration method

● None

○ Scaling (Measurement)

○ Calibration plate (Measurement)

○ Point pair list (Robotics)

Unit

Millimeter (mm)

Fig. 65: Select the calibration method

Parameter	Function
Calibration method	Selection of a calibration method: • None: Calibration not active, coordinate determination, display and output in pixels (px) • Measurement: Calibration methods for applications in the field of measurement and testing

Parameter	Function
	• Robotics: Calibration methods for applications in the field of robotics
Unit	Desired unit for world coordinates. The following units are available: • mm (millimeter) • cm (centimeter) • m (meter) • in (inch)
">" / "<"	Go to next / previous step

Calibration method measurement

Method	Functions
Scaling (Measurement) • Relative calculation of distances in world coordinates • limited accuracy	The calibration method "Scaling" serves relative calculation of distances in world coordinates (mm). This is realized with a simple factor. There is only one factor for both coordinate axis X and Y. The advantage is the very simple function, but accuracy is limited. Errors caused by tilt angle against perpendicular view to the measurement plane or by lens distortion are not corrected by this method. World coordinates are not absolute. The coordinate values refer to the principal point in left, upper corner or the field of view. **Example:** Determination of distances between two objects in mm. (Limited accuracy) **Additional information:** Calibration method Scaling (Measurement) (Page 97)
Calibration plate (Measurement) • Relative calculation of distances in world coordinates • high accuracy	The calibration method "Calibration plate (Measurement)" serves relative calculations of distances in world coordinates (e.g. mm). This is done by image capturing of a calibration plate. By using a large number of points, the known, exact relative position of the points on the plate, this method provides a high accuracy. Errors caused by scaling, x- and y- axis separately, tilt angle against perpendicular view to the measurement plane or by lens distortion are all corrected by this method.

Method	Functions
	World coordinates are not absolute. The coordinate values refer to the principal point in left, upper corner or the field of view. Beside coordinates, distances are also calculated in world frame. **Example:** Determination of distances between two object in mm. **Additional information:** Calibration method Calibration plate (Measurement) (Page 99)

Calibration method robotics

Method	Functions
Calibration plate (Robotics) Absolute calculation in world coordinates, in a user defined reference system, e.g. robot coordinate system	The calibration method “Calibration plate (Robotics)” is used to determine absolute positions in world coordinates (e.g., mm). The scaling in x and y is separated. The tilt of the sensor towards the field of view, and the lens distortion are corrected. In contrast to the “Calibration plate (Measurement)”, the transformation into the absolute coordinate system of the robot is made possible with the “Calibration plate (Robotics)”, by teaching the fiducials. **Example:** Determination of absolute positions of objects in world coordinates (for example, robot coordinate system) in millimeters. This takes place via the image acquisition of the calibration plate and the additional teach-in of the fiducials. For each fiducial, the world coordinate are transmitted e.g. by the robot controller or entered numerically. **Additional information:** Calibration method Calibration plate (Robotics) (Page 101)
Point pair list (Robotics) Absolute calculation in world coordinates, in a user defined reference system, e.g. robot coordinate system	The calibration method “Point pair list” serves absolute calculation of positions in world coordinates (e.g. mm). Errors caused by scaling, x- and y- axis separately, tilt angle against perpendicular view to the measurement plane or by lens distortion are all corrected by this method. **Example:** Determination of absolute positions of objects in world coordinates in millimeter (e.g. robot coordinate system) This is realized by the image capturing of a calibration

Method	Functions
	part which is placed by the robot in the field of view. A point pair is set by: - Image coordinate by graphical input in the image, or by numerical input of a value - World coordinate by numerical input given from the robot controller This sequence is done till the desired number of point pairs is achieved in the list. **Additional information:** Calibration method Point pair list (Robotics) (Page 103)

Please note: All position values and measurement results are corrected. Not to cause longer cycle time the image data are not transformed / displayed rectified. This way a high execution speed, even with calibration active, is provided.

Activation of Calibration is done in two steps:

1. Selection of calibration method:
   Go to next / previous step with buttons **[<]**, **[>]** on the right hand side of the calibration tab
2. Execution of selected calibration method
   As soon as a calibration method is selected, on the left side in tab “Calibration” the status LED is shown.
   If calibration is active other functions like detectors can only be processed successfully, if calibration is valid.

Color significance status LED

Color	Status-LED	Point in image and in point pair list:
Green	Calibration valid	Points accurately positioned
Yellow	Calibration valid	Points not accurately positioned
Red	Calibration not valid	/

Please note:

- Scaling, only green is possible: Default- or input value result in scaling factor, no error determination possible.
- Point Pair list (Robtics): With a new job appears green. Default values (9 points) result in correct default calibration.
- Calibration plate (Measurement): With a new job appears red, as so far no calibration with calibration plate happened.

Calibration effects the following detectors / alignments

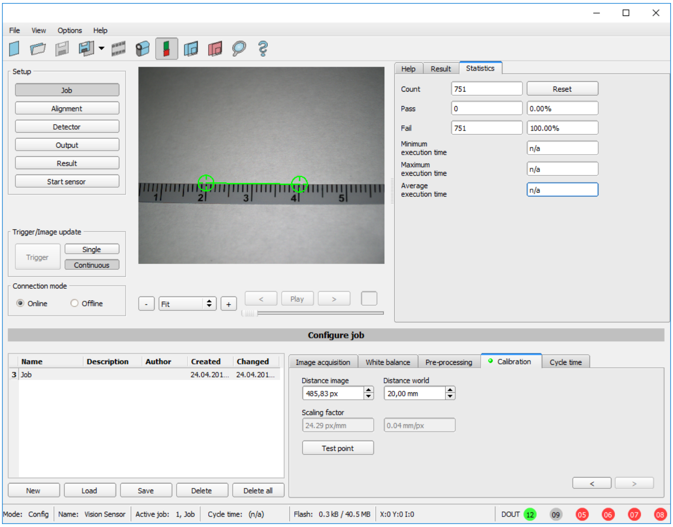
Detector	Result value
Contour	Center coordinate x, y, angle
Pattern matching	Center coordinate x, y, angle
Caliper	Center coordinate x, y, distance
BLOB	Center of gravity-/ center coordinate x, y; width, height, angle

Alignment	Result value
Contour	Center coordinate x, y, angle
Pattern matching	Center coordinate x, y, angle
Edge detection	Center coordinate x, y

4.6.1.6.2 Calibration measurement

4.6.1.6.2.1 Calibration method Scaling (Measurement)

The calibration method “Scaling (Measurement)” serves relative calculations of distances in world coordinates (mm). This is realized with a simple factor. There is only one factor for both coordinate axis X and Y. The advantage is the very simple function of the scaling process, although accuracy is limited.



Connection mode: ● Online ○ Offline

Count	751
Pass	0 — 0.00%
Fail	751 — 100.00%
Minimum execution time	n/a
Maximum execution time	n/a
Average execution time	n/a

	Name	Description	Author	Created	Changed
3	Job			24.04.201...	24.04.201...

Distance image: 485,83 px

Distance world: 20,00 mm

Scaling factor: 24.29 px/mm — 0.04 mm/px

Mode: Config | Name: Vision Sensor | Active job: 1, Job | Cycle time: (n/a) | Flash: 0.3 kB / 40.5 MB | X:0 Y:0 I:0

Fig. 66: Calibration method, “Scaling (Measurement)”

Example: Determination of distance between two objects in millimeter.

Parameter Scaling

Parameter	**Function**
Distance image	Distance in image in pixel (px), by graphical or numerical input
Distance world	Corresponding distance in world by numerical input (in previously selected unit, e.g. mm)
Scaling factor	From above mentioned settings “Distance image” and “Distance world” resulting scaling factor e.g.. [x] px/mm or. [y] mm/px
Test point	Test point (graphically or values input) is for the user to check calibration of known points / dimensions around the image to confirm satisfactory setting of the scaling factor.
“>” / “<”	Go to next / previous step

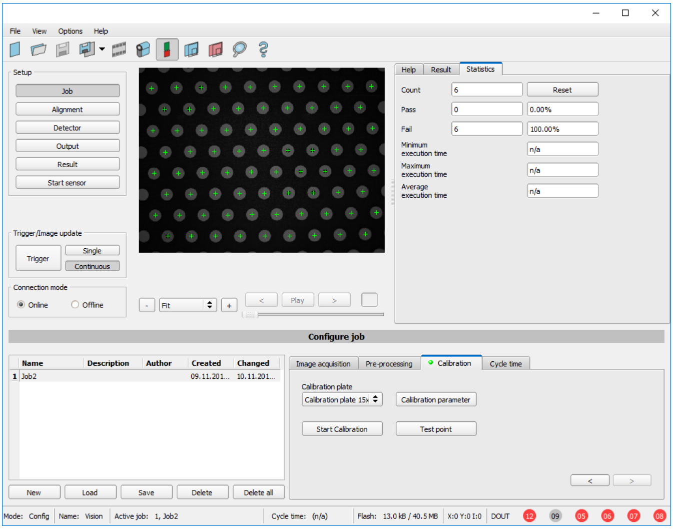
Note:

- Please take care that the optical axis of the sensor is aligned perpendicularly to the measurement plane. This avoids different distortion in x and y axis. Errors caused by tilt angle against perpendicular view to the measurement plane or by lens distortion are not corrected using this method.

- For setting up; place a object with known dimensions (e.g. gauge block) in the field of view. Position the both graphical, green crosshairs in the image to the points of the object with a known dimension / distance. The distance in image pixels between the both centres of the crosshairs is displayed in the field “Distance image”. Now type the known distance in world in field “Distance world” (e.g. in mm). The scaling factor is calculated and displayed. From now on positions and distances are displayed and transferred in world coordinates.
- The size of a crosshair in the field of view can be adjusted as desired. For this purpose, the desired crosshair point must be selected by mouse click, and the size can then be increased or decreased by means of the scroll wheel of the mouse.
- World coordinates are not absolute. The coordinate values refer to the principal point in left, upper corner or the field of view. Beside coordinates, distances are also calculated in world frame.
- This kind of calibration is suitable for standard lenses, integrated or C-mount. However it´s not suitable for telecentric lenses.

4.6.1.6.2.2 Calibration method Calibration plate (Measurement)

The calibration method “Calibration plate (Measurement)” serves relative determination of e.g. distances in world coordinates (e.g. mm). This is done by image capturing of a calibration plate with one single click!



Connection mode: ● Online ○ Offline

Count: 6

Pass: 0 0.00%

Fail: 6 100.00%

Minimum execution time: n/a

Maximum execution time: n/a

Average execution time: n/a

	Name	Description	Author	Created	Changed
1	Job2			09.11.201...	10.11.201...

Calibration plate: Calibration plate 15x

Mode: Config | Name: Vision | Active job: 1, Job2 | Cycle time: (n/a) | Flash: 13.0 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 67: Calibration method, Calibration plate (Measurement)

Example: Determination of distances between two object in mm (high accuracy).

Sequence calibration via calibration plate

Previously the focus and the shutter of the sensor, and the desired unit must be selected.

1. Use the "Calibration Parameter" button to set the z offset.
2. Place the calibration plate that the field of view is completely covered (see also: Advices on optimized use of the calibration plate / boundary conditions (Page 107), and Types / sizes of calibration plates (Page 101)).
3. Select the corresponding calibration plate (size and type) via the list box "Calibration plate".
4. With a click to the button "Start Calibration" all visible points of the calibration plate are determined, all detected are marked, and the calibration is calculated.

Parameter Calibration plate (Measurement)

Parameter	Function
Calibration plate	Here the used calibration plate (size and type / number of points) is selected (See also: Advices on optimized use of the calibration plate / boundary conditions (Page 107) and Types / sizes of calibration plates (Page 101)).
Calibration parameter	If given, here the z-offset between calibration and measurement plane can be set. Also different read only parameters, as well as deviation parameters are shown in this dialog (see also: Calibration, Calibration parameter (Page 109)).
Start Calibration	Calibration is started. All visible points of the calibration plate are determined, all detected are marked, and calibration is calculated.
Test point	A test point can be set in the image, whose world coordinate values for test and control purposes are displayed in the Test point window.
">" / "<"	Go to next / previous step

Note:

- The sensor can be mounted in any alignment / pose referred to the measurement plane. Anyway a close to perpendicular alignment should be preferred, as this causes less distortion and this way less error correction is needed.
- World coordinates are not absolute. The coordinate values refer to the principal point in left, upper corner or the field of view. Beside coordinates, distances are also calculated in world frame.
- This kind of calibration is suitable for standard lenses, integrated or C-mount. It's not suitable for telecentric lenses.

- Normalization: Independent of the position and orientation in which the calibration plate is detected, the coordinate origin always lies in the upper left corner of the field of view (image and world). The zero-degree direction and the positive X-axis point to the east.

Advices on the optimized use of the calibration plate / boundary conditions can be found at: Advices on optimized use of the calibration plate / boundary conditions (Page 107).

Types / sizes of calibration plates

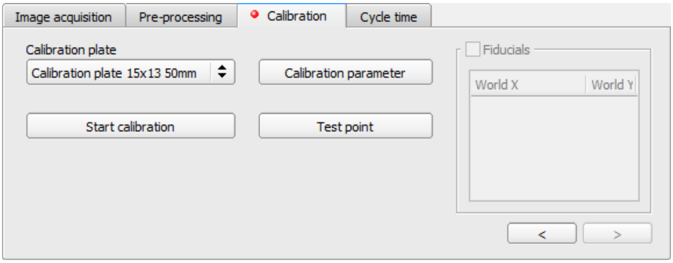
Sizes of calibration pattern	Number of points
50 mm x 37.9 mm	15 x 13
100 mm x 75.8 mm	15 x 13
200 mm x 151.7 mm	15 x 13

In the installation folder: SensoPart /VISOR® vision sensor/Documentation/... the available calibration plates can be found as .pdf-file. This can be printed on paper or any other medium. Please consider the setting “actual size”, that print out is not scaled. The length of the long edge of the plate must correspond exactly to the number in the name of the plate.

4.6.1.6.3 Calibration robotics

4.6.1.6.3.1 Calibration method Calibration plate (Robotics)

The calibration method “Calibration plate (Robotics)” is used to determine absolute positions in world coordinates (e.g., mm). This takes place via the image acquisition of the calibration plate and the teach-in of the four fiducials. With the teach-in of the fiducials, the transformation into the absolute coordinate system of the robot is made possible.



Calibration plate: Calibration plate 15x13 50mm

□ Fiducials

Fig. 68: Calibration method calibration plate (robotics)

Sequence calibration via calibration plate (Robotics)

Previously the focus and the shutter of the sensor must be selected as well as the desired unit.

1. Use the "Calibration Parameter" button to set the z offset.
2. Place the calibration plate as far as possible in the field of view (see also "Advices on optimized use of the calibration plate / boundary conditions (Page 107)" and "Types / sizes of calibration plates").
3. Select the corresponding calibration plate (size and type) in the "Calibration plate" selection field.
4. Start calibration.
5. If the working range of the robot deviates from the field of view, transfer the calibration plate into the working range of the robot, e.g. over a conveyor belt.
6. Enable listbox "Fiducials".
7. Select line 1 in Listbox "Fiducials".
8. Get the first reference mark with the robot.
9. Now type in the corresponding, known world coordinates in the field "World X" and "World Y" (with e.g. robot: the values displayed in the robot controller).
10. Repeat steps 7-10 until all fiducials have been entered.

Parameter calibration plate (robotics)

Parameter	Function
Calibration plate	Here the used calibration plate (size and type / number of points) is selected. (see also: Advices on optimized use of the calibration plate / boundary conditions (Page 107) and Types / sizes of calibration plates (Page 102)
Calibration parameter	If given, here the z-offset between calibration and measurement plane can be set. Also different read only paremters, as well as deviation parameters are shown in this dialog (see also: Calibration, Calibration parameter (Page 109)
Start Calibration	Calibration is started. All visible points of the calibration plate are determined, all detected are marked, and calibration is calculated.
Test point	A test point can be set in the image, whose world coordinate values for test and control purposes are displayed in the Test point window.
Fiducial - World X - World Y	Coordinate values in selected unit (e.g. mm), by direct numerical input of the values in the fiducials list. In case of e.g. Robotics Pick&Place this values can be taken from the robot controller when placing the calibration part in the field of view.
">" / "<"	Go to next / previous step

Advices on the optimized use of the calibration plate / boundary conditions can be found at: Advices on optimized use of the calibration plate / boundary conditions (Page 107)

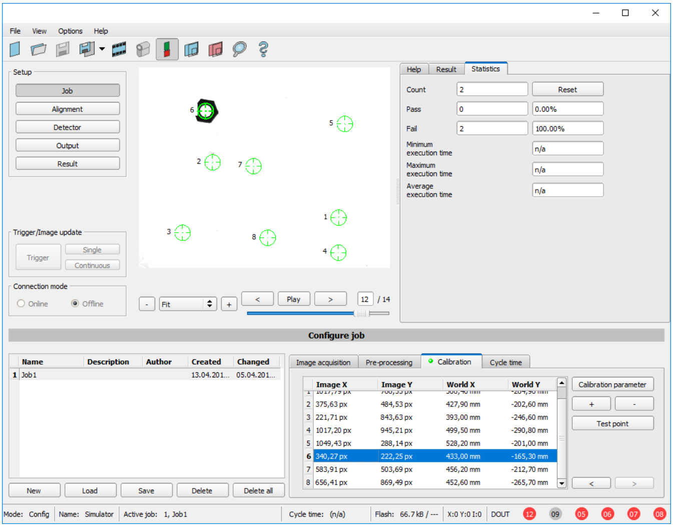
Types / sizes of calibration plates

Sizes of calibration plates	Sizes of calibration pattern	Number of points
98 mm x 54 mm	50 mm x 37.9 mm	15 x 13
180 mm x 100 mm	100 mm x 75.8 mm	15 x 13
340 mm x 176 mm	200 mm x 151.7 mm	15 x 13
820 mm x 403 mm	500 mm x 379.2 mm	15 x 13

In the installation folder: SensoPart /VISOR® vision sensor/Documentation/... the available calibration plates can be found as .pdf-file. This can be printed on paper or any other medium. Please consider the setting “actual size”, that print out is not scaled. The length of the long edge of the plate must correspond exactly to the number in the name of the plate.

4.6.1.6.3.2 Calibration method Point pair list (Robotics)

The calibration method “Point pair list (Robotics)” is used to determine absolute positions in world coordinates (e.g. mm).



Count	2
Pass	0 / 0.00%
Fail	2 / 100.00%
Minimum execution time	n/a
Maximum execution time	n/a
Average execution time	n/a

	Name	Description	Author	Created	Changed
1	Job1			13.04.201...	05.04.201...

	Image X	Image Y	World X	World Y
1	1017,79 px	[illegible]	[illegible]	[illegible]
2	375,63 px	484,53 px	427,90 mm	-202,60 mm
3	221,71 px	843,63 px	393,00 mm	-246,60 mm
4	1017,20 px	945,21 px	499,50 mm	-290,80 mm
5	1049,43 px	288,14 px	528,20 mm	-201,00 mm
6	340,27 px	222,25 px	433,00 mm	-165,30 mm
7	583,91 px	503,69 px	456,20 mm	-212,70 mm
8	656,41 px	869,49 px	452,60 mm	-265,70 mm

Mode: Config | Name: Simulator | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 66.7 kB / --- | X:0 Y:0 I:0

Fig. 69: Calibration method, Point pair list (Robotics)

Example: Determination of absolute positions, and orientation of objects in world coordinates in mm (e.g. robot coordinate system).

Motivation / Benefit

After calibration of the sensor via point pair list, the position of the part to 'pick' is available directly in the absolute coordinate system of the robot!
All errors like scaling, perspective and lens distortion are corrected. In robotics pick and place applications, the robot can now pick the part with the sensor provided robot coordinate values.

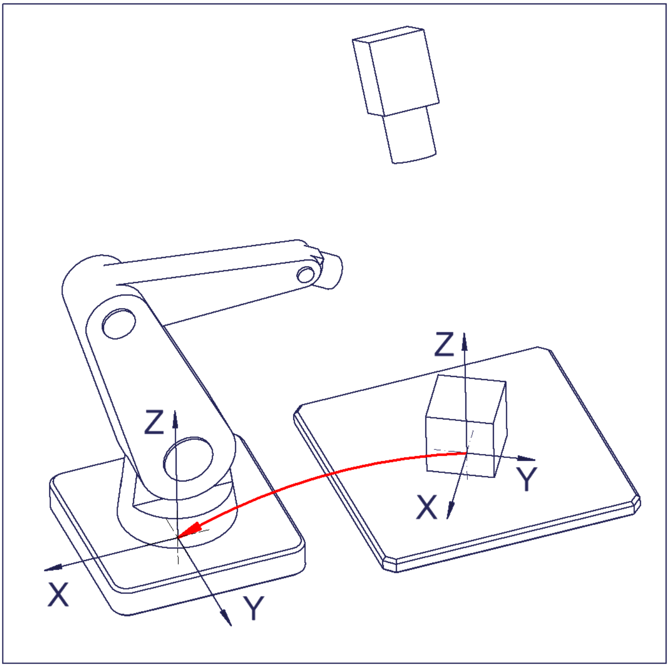


Fig. 70: Position of part to pick directly in robot coordinate system!

Sequence calibration via point pair list

Prior to this, the sensor must be focused, have the correct shutter speed set and the desired unit must be selected.

1. Select the calibration model (with / without correction of the lens distortion), and adjust the z offset if necessary.
2. Select line 1 in list box “point pair list”.
3. Place calibration part (preferably flat, symmetric, e.g. similar plain washer) at the exact known world coordinate (e.g. with robot). Place graphically the corresponding crosshair in the image (no. “n” corresponding to line “n” in point pair list) exactly in the center of the calibration object (if necessary zoom image).
   Alternatively: use “Snap- Function”, that means: right click somewhere inside the calibration part. This way the center of gravity of the calibration part is automatically

determined.
Preferably use point symmetric calibration parts, as then the center of gravity is independent from orientation. With calibration parts which are not point symmetric please take care that the same orientation is always used (not available with color sensors).
Result: Values of image coordinates in pixel “Image X” and “Image Y” are automatically set in line “n”.

4. Place graphically the corresponding crosshair in the image (no. “n” corresponding to line “n” in point pair list) exactly in the center of the calibration object (if necessary zoom image).
   Alternatively: use “Snap- Function”, that means: right click somewhere inside the calibration part. This way the center of gravity of the calibration part is automatically determined.
   Preferably use point symmetric calibration parts, as then the center of gravity is independent from orientation. With calibration parts which are not point symmetric please take care that the same orientation is always used (not available with color sensors).
   Result: Values of image coordinates in pixel “Image X” and “Image Y” are automatically set in line “n”.
5. Now type in the corresponding, known world coordinates in the field “World X” and “World Y” (with e.g. robot: the values displayed in the robot controller).
6. Repeat steps 2-5 as long as the desired number of point pairs is achieved. If more lines are necessary press “+”, to delete lines press “-” (min. 6 points, recommended >10 points).

Automated calibration, see also:Sequence calibration point pair list (robotics) (Page 114)

Parameter point pair list (robotics)

Method	**Functions**
- Image X **- Image Y** Values in point list	Coordinate values in pixels (px) in the image, via exact graphical positioning of the crosshair to the center point of the calibration part which is placed exactly in world coordinates. Or: use “Snap- Function”, that means: right click somewhere inside the calibration part. This way the center of gravity of the calibration part is automatically determined (recommended).
- World X **- World Y** Values in point list	Coordinate values in selected unit (e.g. mm), by direct numerical input of the values in the point pair list. In case of e.g. Robotics Pick & Place these values can be taken from the robot controller when placing the calibration part in the field of view.
Calibration parameter	**Calibration model:** With or without correction of lens distortion.

Method	Functions
	Z- Offset: (if offset is given) Offset between calibration plane and measurement plane **Different read only parameters** Of the regression calculation and error values. See also: Calibration, Calibration parameter (Page 109).
"+" / "-"	Add or delete one line / point. Delete affects the highlighted line.
Test point	A test point can be set in the image, whose world coordinate values for test and control purposes are displayed in the Test point window.
">" / "<"	Go to next / previous step

Note:

- The sensor can be mounted in any alignment / pose referred to as the measurement plane. Anyway a close to perpendicular alignment should be preferred, as this causes less distortion and this way less error correction is needed.
- The accuracy of the calibration first depends on the quality / accuracy of the point position and secondly on the sufficient number of points. If the calibration is not accurate (yellow points) this can be improved by better precision of position input of the single points.
- This kind of calibration is suitable for standard lenses, integrated or C-mount. It´s not suitable for telecentric lenses.
- The size of a crosshair in the field of view can be adjusted as desired. For this purpose, the desired crosshair point must be selected by mouse click, and the size can then be increased or decreased by means of the scroll wheel of the mouse.

Minimum required number of point pairs is "6" points.

The minimum necessary number of points for calibration via point pair list is 6 points. With minimum this number of points false inputs (like x and y interchanged) can be found by high error values in dialog "Calibration parameters" Calibration, Calibration parameter (Page 109), (if <= 5 points the error values are always = 0, as no errors can be calculated).

To show the quality of point position (how good point positions match with calculated point positions) the points are displayed in the following colors (only meaningful if there is a minimum of 6 points).

Color significance of graphical points in image and lines in Point pair list:

Color	Significance
Green	Calibration valid, points accurately positioned

Color	Significance
Yellow	Calibration valid, points not accurately positioned
Red	Calibration not valid

In the case of a yellow point color a yellow line is visible starting in the center of the point. It’s lenght and direction is a measure for the absolute value and orientation of the error in relation to the position accuracy of point input in the world frame.

If there are big errors, potentially x- and y- coordinate are interchanged with one or multiple points, or some points are interchanged completely with others.

In the dialog: Calibration, Calibration parameter (Page 109) the deviation values / errors: “Mean”, “Min”imum error and “Max”imum error are displayed. With these values the exact input positions of the existing points can be optimised.

This calibration method serves alongside with the absolute coordinate values as well as the orientation of the part to pick (if Contour or Pattern matching is used as detector).

The result coordinates of the target object are present in the reference system of the robot!

4.6.1.6.4 Advices on optimized use of the calibration plate / boundary conditions

- The calibration plate must be clean and plain.
- The plate must be illuminated homogeneously over the entire field of view and must not be overexposed. The bright regions should have a gray value of at least 100 and below 255. The contrast between bright and dark regions should be at least 100 gray values. That means, the image must not be under- or overexposed.
- The calibration pattern should cover the entire field of view of the VISOR® vision sensor. For a successful, precise calibration it’s not necessary that the entire calibration plate is visible. To perform a calibration, at least one search pattern must be found.
- For small calibration patterns, it may be necessary to use two search patterns.
- Calibration works correctly only if the focus and position of the sensor does not change in relation to the measurement plane.

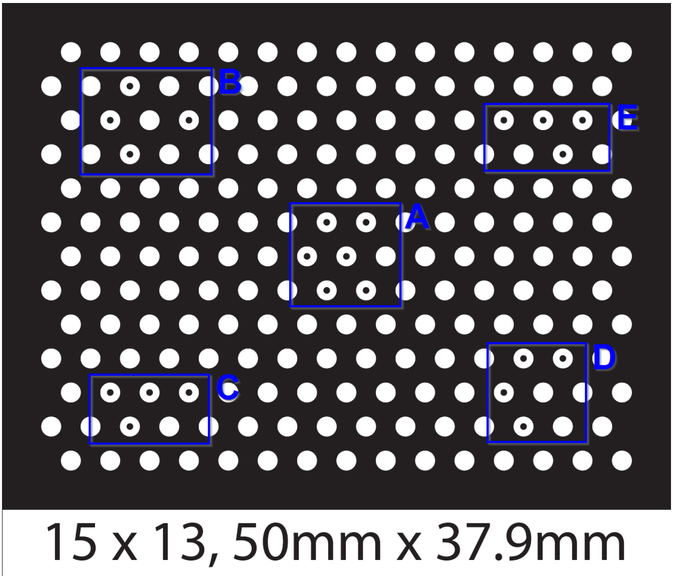


Fig. 71: Calibration plate, blue = search pattern.

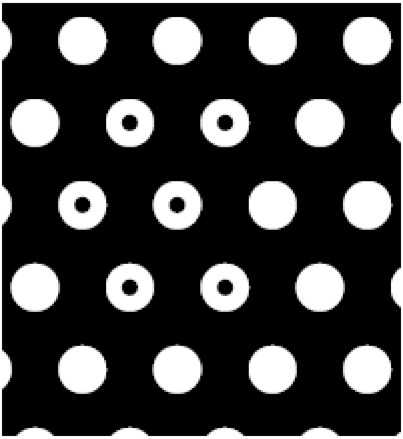
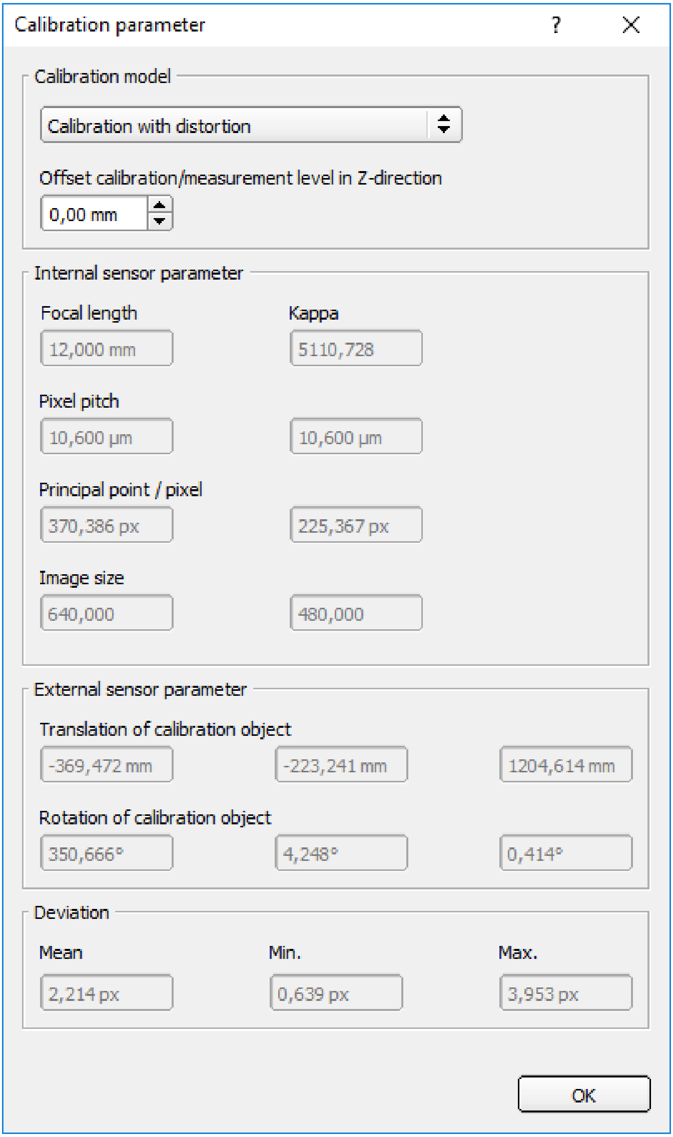


Fig. 72: Calibration plate, detail with smaller black points in the center (see above: blue regions).

4.6.1.6.5 Calibration, Calibration parameter

Here, if required, the Z-offset between calibration- and measurement level in Z-direction can be set and compensated. Also, if desired, the calibration- and deviation parameters for optimization can be displayed.

This kind of calibration is suitable for standard lenses, integrated or C-mount. However it does not work for telecentric lenses.



Calibration parameter

Calibration model

Calibration with distortion

Offset calibration/measurement level in Z-direction

0,00 mm

Internal sensor parameter

Focal length	Kappa
12,000 mm	5110,728

Pixel pitch	
10,600 µm	10,600 µm

Principal point / pixel	
370,386 px	225,367 px

Image size	
640,000	480,000

External sensor parameter

Translation of calibration object		
-369,472 mm	-223,241 mm	1204,614 mm

Rotation of calibration object		
350,666°	4,248°	0,414°

Deviation

Mean	Min.	Max.
2,214 px	0,639 px	3,953 px

Fig. 73: Calibration, Calibration parameter

Parameter	Function
Calibration model	
Calibration model: Standard lens, with distortion	Correction of: • Scaling, x and y separately • Tilt angle against perpendicular view to the measurement plane • Lens distortion
Offset calibration/measurement level in Z-direction [*1)]	For Z=0 the calibration and the measurement plane are identical. For Z≠0 the calibration plane is shifted against the measurement plane. The two planes are always parallel. The sign of the deviation results from the right hand world system (thumb = x, index finger = y, middle finger = z, see below) Note: The depth of focus of the sensor must cover the calibration and the measurement plane. See also: Offset calibration/measurement level in Z-direction (Page 112)
Internal sensor parameter	
Focus	Focus of the lens • With integrated lens: value of the built in lens [*2)] • With C-Mount lens: Take value written on the used lens and type in. Option: to check plausibility of e.g. z- value with below mentioned “Translation of calibration object” no malfunction if not used. [*1)]
Kappa (x10E-6) [*2)]	Calculated kappa (distortion) value of the lens.
Pixel pitch [*2)]	Calculated pitch / axial distance from pixel to pixel on the sensor chip. Reduction of resolution in tab “Image acquisition” effects this value.
Principal point / pixel [*2)]	Point where the optical axis penetrates the measurement plane in the center of the sensor chip, compared with the ideal center point. This values refer to left , upper corner in pixel.

Parameter	**Function**
Image size $^{*2)}$	Image size in pixel
External sensor parameter	
Translation of calibration object $^{*2)}$	All three calculated values of translation of the calibration object. I.e. in x-, y- and z-direction.
Rotation of calibration object $^{*2)}$	All three calculated values of rotation of the calibration object. I.e. the angles: alpha, beta and gamma.
Deviation	
Mean $^{*2)}$	Average error of calculated positions against input.
Min. $^{*2)}$	Maximum error of calculated positions against input.
Max. $^{*2)}$	Minimum error of calculated positions against input.
Center of calibration plate (world coordinates)	
X $^{*2)}$	X position of the center of the calibration plate in the world coordinate system.
Y $^{*2)}$	Y position of the center of the calibration plate in the world coordinate system.
Angle $^{*2)}$	Angle from the center of the calibration plate to the zero point of the world coordinate system.
Deviation fiducials	
Mean $^{*2)}$	Average error of calculated positions against input.
Min. $^{*2)}$	Maximum error of calculated positions against input.
Max. $^{*2)}$	Minimum error of calculated positions against input.

$^{*1)}$ Input parameter, $^{*2)}$ Read only parameter

Offset calibration/measurement level in Z-direction

Sign of “Z” value depending on the world coordinate system / “right hand world system” (thumb = x, index finger = y, middle finger = z)

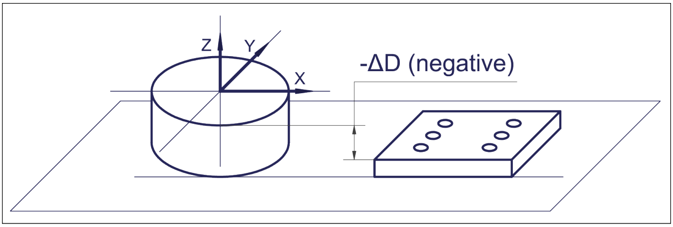


Fig. 74: Delta “D” / Z- Offset = negative! In case of: Z-to top, and calibration plane lower than measurement plane!

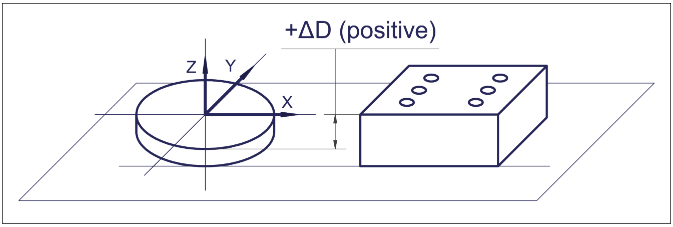


Fig. 75: Delta “D” / Z- Offset = positive! In case of: Z-to top, and calibration plane higher than measurement plane!

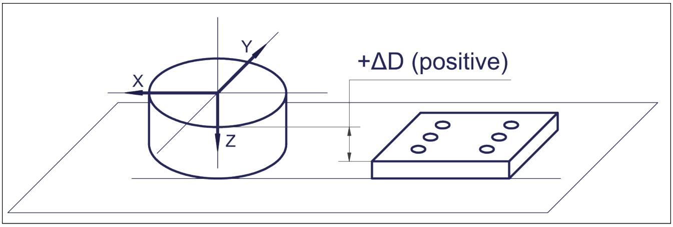


Fig. 76: Delta “D” / Z- Offset = positive! In case of: Z-to bottom, and calibration plane lower than measurement plane!

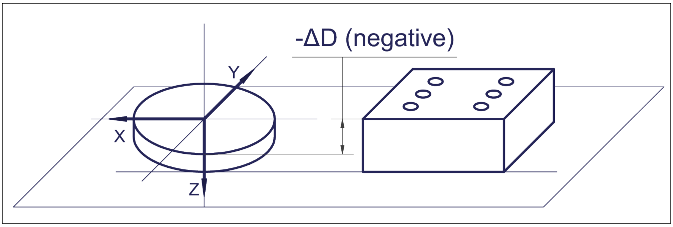


Fig. 77: Delta “D” / Z- Offset = negative! In case of: Z-to top, and calibration plane higher than measurement plane!

4.6.1.6.6 Calibration via telegrams

Various interface telegrams are available for the calibration, see chapter: Overview VISOR® vision sensor telegram: (Page 456).

The telegrams can be used for recalibration if a drift in the production process has occurred or if the mounting position of the sensor has changed. The calibration process can be executed automatically, e.g. from the robot controller.

4.6.1.6.6.1 Sequence calibration point pair list (robotics)

An example shows the sequence for automated calibration with interface telegrams via the point pair list.

Sequence / flow chart

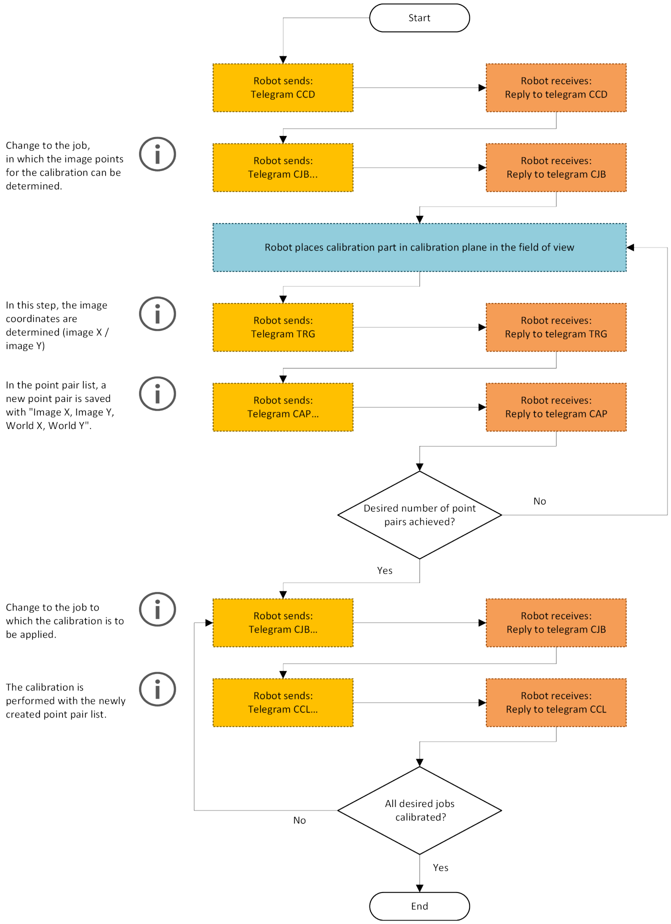

Start
Robot sends: Telegram CCD
Robot receives: Reply to telegram CCD
Change to the job, in which the image points for the calibration can be determined.
Robot sends: Telegram CJB...
Robot receives: Reply to telegram CJB
Robot places calibration part in calibration plane in the field of view
In this step, the image coordinates are determined (image X / image Y)
Robot sends: Telegram TRG
Robot receives: Reply to telegram TRG
In the point pair list, a new point pair is saved with "Image X, Image Y, World X, World Y".
Robot sends: Telegram CAP...
Robot receives: Reply to telegram CAP
Desired number of point pairs achieved?
No
Yes
Change to the job to which the calibration is to be applied.
Robot sends: Telegram CJB...
Robot receives: Reply to telegram CJB
The calibration is performed with the newly created point pair list.
Robot sends: Telegram CCL...
Robot receives: Reply to telegram CCL
All desired jobs calibrated?
No
Yes
End

Fig. 78: Automated sequence for calibration via point pair list

4.6.1.6.7 Calibration methods, location of the world system

Legend:

Abbreviation	Meaning
WF	World Frame

Abbreviation	Meaning
IF	Image Frame
CPF	Calibration Plate Frame

Scaling (Measurement)

In the calibration method Scaling, the origin of the world frame (WF) corresponds to the origin of the image frame (IF). The zero point is located in the upper left corner of the field of view.

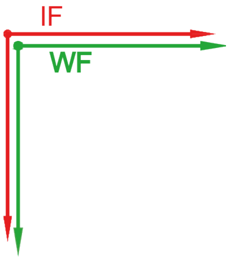


Fig. 79: Origin WF ≙ Origin IF

Calibration plate (Measurement)

In the calibration method Calibration plate (measurement), the origin of the world frame (WF) corresponds to the origin of the image frame (IF). The zero point is located in the upper left corner of the field of view.

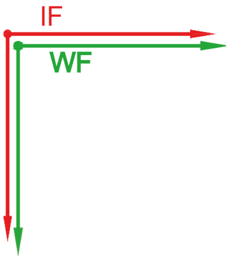


Fig. 80: Origin WF ≙ Origin IF

Calibration plate (Robotics)

In the calibration method “Calibration plate (Robotics)”, the origin of the world frame (WF) can be located in two places, depending on the calibration. Usually, the origin of the world system (WF1) is given by the world coordinates for the fiducials.

If the coordinate determination of the fiducials is omitted and only the calibration plate is taught-in, the origin of the world frame (WF2) corresponds to the origin of the calibration plate frame (CPF). In contrast to the “Calibration plate (Measurement)”, the zero point of the “Calibration plate (Robotics)” is not in the upper left corner of the field of view but in the center of the calibration plate.

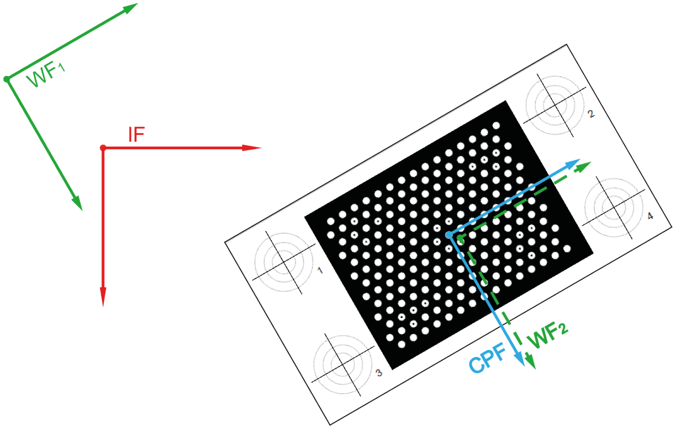


Fig. 81: Origin WF = Given by the coordinates of the fiducials or origin WF ≙ origin CPF

Point pair list (Robotics)

The origin of the world system (WF) is given by the coordinates (WF and IF) for the crosshair points.

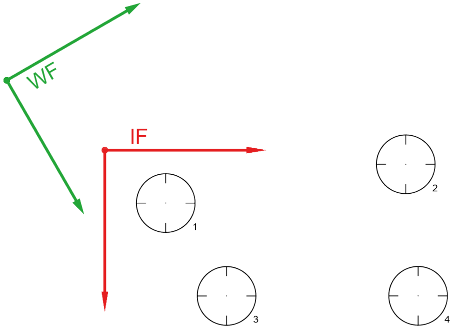
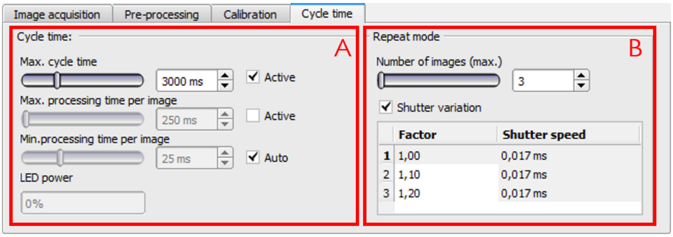


Fig. 82: Origin WF = Given by the coordinates of the crosshair points

4.6.1.7 Tab Cycle time

In tab Cycle time the timing conditions of the VISOR® vision sensor can be defined.



Cycle time:

Max. cycle time 3000 ms ☑ Active

Max. processing time per image 250 ms □ Active

Min.processing time per image 25 ms ☑ Auto

LED power 0%

Repeat mode

Number of images (max.) 3

☑ Shutter variation

	Factor	Shutter speed
1	1,00	0,017 ms
2	1,10	0,017 ms
3	1,20	0,017 ms

Fig. 83: Setup Job, tab Cycle time

(A) Cycle time	
Parameter	**Function and possibilities**
Max. cycle time	Parameter to control the minimum and maximum time of a cycle. Inside a cycle some images can be evaluated (in case of “Number of images (max)” >1) Maximum processing time per image interrupts a job after a defined time. The result of a cycle after a timeout is always “not ok”. Maximum processing time should be selected higher than the time demand for one execution. The processing time is the time elapsed from trigger till the setting of the digital outputs. If this cycle time should be limited (e.g. if the machine cycle should not be exceeded) this function can be used. The result of all detectors which are not processed / finished after this processing time has elapsed are set to “failed”. As the currently processed detector will still be finished, please consider that the adjusted job time may not be met a 100% exactly, and it may last a few milliseconds longer till the job is interrupted. It’s recommended to test the real cycle time and to choose a value for this parameter which is a bit smaller / shorter.
Max. processing time per image	Maximum duration of one evaluation inside a cycle including image capturing.
Min. processing time per image	Minimum duration of one evaluation inside cycle including image capturing. Minimum processing time blocks trigger signals which are coming before the minimum processing time is reached.
LED-Power	This value is calculated automatically. Standard Value is 100%. LED-power may be reduced, if shutter time is quite long and minimum job time is quite short, because the recovery time for the LEDs may be to short in this case. To obtain 100% LED power, minimum job time should be factor of 10 longer than shutter time.

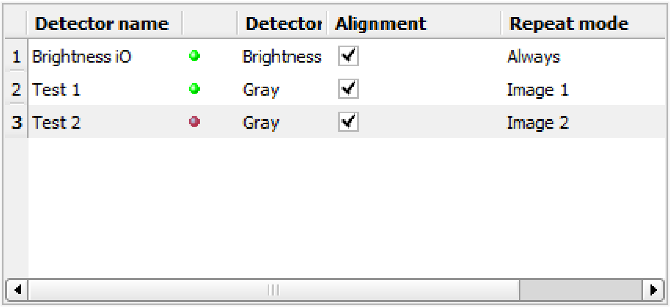
(A) Cycle time	
Parameter	**Function and possibilities**
Auto	If “Auto” is selected the minimum cycle time is automatically adjusted in a way that the LED-power is 100%

(B) Repeat mode	
Parameter	**Function and possibilities**
Number of images (max.)	Maximum number of image capturings, which are processed after one trigger, if the stop criteria is not fulfilled. The stop criteria is the: • “Overall job result”= positiv (access via Output/Digital output) • “Max. processing time per image” is not fulfilled (if activated)
Shutter variation	When Shutter variation is “active”, a variation from several different shutter speeds can be set up over a table. Per configured shutter speed an image is aquired, this means that first image is taken with shutter value 1, second image is taken with shutter value 2, third image is taken with shutter value 3, etc. Default setting for “Shutter variation” is off. In this case, the table is not displayed.
Factor and Shutter speed	Default value for “Factor” is: First value = 1.00 (the first factor is always identical to 1.00 and read-only). Subsequent default values are increased by 0.1, eg. 1.10, 1.20, ... The user can modify the factor in the table, thereby the “Shutter speed” (second column, read-only) will automatically be updated and an image acquired. By a click into one row of the table, an image with the settings of these table row will be acquired. Please note: When changing the “Shutter speed” parameter in “Image acquisition” tab, the shutter speed in the “Shutter variation” listbox is recalculated.

Repeat Mode: Assign the detector to an image

In the setup “Detector” all selected detectors are listed. If the “Number of images (max)” parameter of the repeat mode is greater than 1, the option of assigning a detector to an image acquisition is obtained. In the “Repeat mode” column, this setting can be made for each detector.

- Always: Executed in all image acquisitions
- Recording n: Executed in the corresponding image acquisition



	Detector name		Detector	Alignment	Repeat mode
1	Brightness iO	🟢	Brightness	☑	Always
2	Test 1	🟢	Gray	☑	Image 1
3	Test 2	🔴	Gray	☑	Image 2

Fig. 84: Open the selection table by double-clicking.

4.6.2 Alignment

Alignment compensation can be necessary for objects or characteristics whose position varies in the image. Three different detection methods (alignment detectors) are available for this purpose.

Mode of function of an alignment detector

An alignment detector is a tracking coordinate system, which is anchored to one selected characteristic. All subsequently defined detectors are aligned in relation to this coordinate system. The tracked coordinate system is drawn in dark blue (for information on the meaning and adjustment of the different frames see chapter: Search and parameter zones).

Please note:

- Maximum of one alignment detector can be defined for each job.
- For each detector in the job, it can be selected whether the detector is to be tracked with the alignment or not.
- As alignment requires an extra calculation step, it should only be used if required by the application.

4.6.2.1 Selection and configuration of an Alignment

Select alignment detector:

1. Click on the button Alignment.
2. Select a detection method in the configuration window “Method”:

Detection method	Description, Selection
None	Alignment deactivated

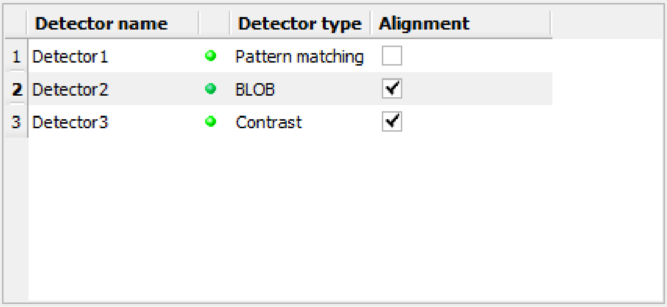
Detection method	Description, Selection
Pattern matching	Detection of any pattern Pattern matching can be used preferably if... • ... there are only marginal edges, parallel to axis or with strong contrast, but zones with gray pattern in the image. Pattern matching cannot be used if there is an angular deviation / rotation of the part.
Edge detection	The detection of edge should be used: • if an offset of the position occurs in X- and / or Y- direction. • at a maximum angle offset (rotational offset compared to the teach-in position) of approx. ± 20° (depending on object and application). • if there are edges with strong contrast, parallel to the axis. If above mentioned criteria are fulfilled, edge detection is a very quick method of alignment.
Contour detection	Detection of contours and edges at any angle Contour detection must always be used if... • ... there can be an angular offset (rotation against teach in position). It can be used preferably if there are edges of any shape but with good contrast.

Configuration of alignment detector:

1. Adapt the position and size of the search and parameter zones displayed on the screen if necessary.
2. Configure the alignment detector in the Parameters tab .

Activation of alignment for detectors

In the “Detector” setup, all selected detectors are listed. In the “Alignment” column, it is possible to select for each detector whether it is to be aligned by the adjusted alignment or not. Default value is “Active”.



	Detector name		Detector type	Alignment
1	Detector1	●	Pattern matching	□
2	Detector2	●	BLOB	☑
3	Detector3	●	Contrast	☑

Fig. 85: Detector list, alignment active / inactive

Reset

The “Reset” button can be used to restore the factory settings for the selected alignment detectors.

4.6.2.2 Alignment Pattern matching

This alignment detector is suitable for the detection of any patterns, even without clear edges and/or contours. The pattern is taught in and placed over the image during the subsequent check. A match is made at the largest similarity value.

4.6.2.2.1 Tab Color channel

In the color channel tab, a color image (3 channel) can be converted to a gray value image (1 channel). In contrast to the gray value image of a monochrome VISOR® vision sensor, contrasts can be significantly increased. The highlighting of a color can be set individually for each detector. Thus, the flexibility compared to the use of optical color filters is significantly higher.

The image displayed is dependent on the selected detector.

- Color detectors: Display always colored.
- Object detectors: Monochrome image, display dependent upon selected color model and color channels.

The following parameters can be configured in the Color channel tab:

Parameter	Function
Color model	Color models: RGB, Color model RGB (Page 324) HSV, Color model HSV (Page 325) LAB, Color model LAB (Page 325)
Selection color filter	Depending on the color space, all or part of the following color filters

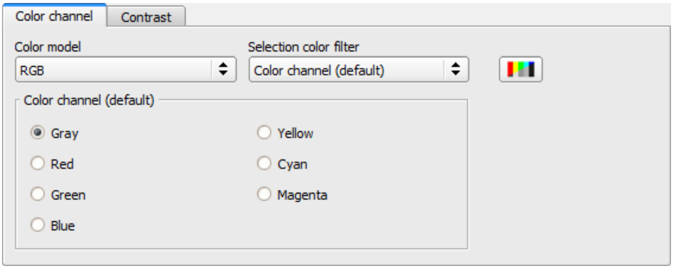
Parameter	Function
	are available: Color channel (default) Color distance Binarization
▥	Switching the image between color and monochrome.

4.6.2.2.1.1 Selection color filter

The following color filters are available:

Color channel (default)

The selected color channel is used as a gray value image.



Color channel | Contrast

Color model: RGB

Selection color filter: Color channel (default)

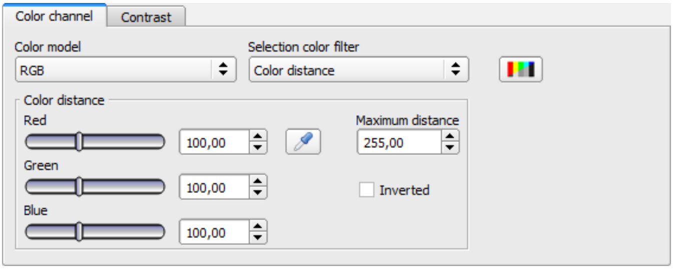
Color channel (default)

- ● Gray
- ○ Red
- ○ Green
- ○ Blue
- ○ Yellow
- ○ Cyan
- ○ Magenta

Fig. 86: Color filter, Color channel (default)

Color distance

A color is selected as reference color by specifying the color model values or by pipette. The gray value image indicates the distance of each pixel to this reference color. Typical application: Segmentation of characters for OCR.



Color channel | Contrast

Color model: RGB

Selection color filter: Color distance

Color distance

Red: 100,00

Green: 100,00

Blue: 100,00

Maximum distance: 255,00

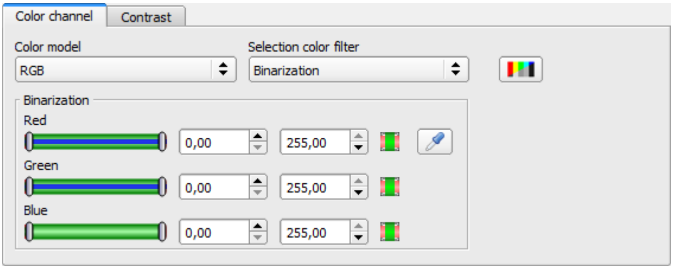
□ Inverted

Fig. 87: Color filter, Color distance

Parameter		Function
Red Green Blue	Lightness A B	Color channels: The color channel can be set via the slider or by entering a value (default 0).
Pipette button		With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.
Maximum distance		Distance of the current color versus the taught-in color. Colors that will exceed the maximum color distance will be black or white depending on the setting of "Inverted".
Inverted		Inversion of the color distance image.

Binarization

A color range is selected. All pixels within this color range become white. Pixels with deviating color values become black.



Color model: RGB

Selection color filter: Binarization

Binarization

Red: 0,00 – 255,00

Green: 0,00 – 255,00

Blue: 0,00 – 255,00

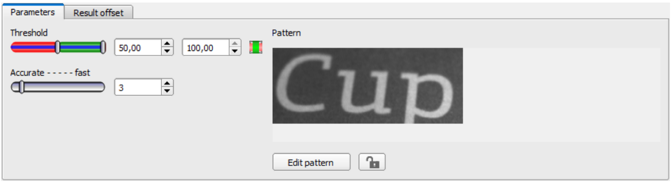
Fig. 88: Color filter, Binarization

Parameter			Function
Red Green Blue	Hue Saturation Value	Lightness A B	Determination of the color range. The color ranges can be set via the slider or by entering a value.
Inverting button			The current setting is inverted when selecting the button.
Pipette button			With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.

4.6.2.2.2 Alignment Pattern matching, tab Parameters

The following parameters can be configured in the Parameters tab:

Parameters	Functions
Threshold	Zone for the required concordance of the found sample with the taught sample
Accurate - fast	Number of search levels / coarsening levels 0 = automatic selection Higher value: faster = riskier (overlook candidates) Smaller value: slower = less risky (all candidates)
Pattern	Shows the taught sample = red frame
Edit pattern	By editing the ROI parts can be masked out of the search area. The parts which are not relevant for the examination can be painted out like using an eraser. Masks can also be inverted, meaning that parts that are of interest can be marked.
Lock	Lock / Unlock Pattern: In locked status the taugth pattern is protected against (unintentional / accidential) changing, e.g. modification of the teach region. Unlock to modify taught pattern.



Threshold 50,00 100,00

Accurate - - - - - fast 3

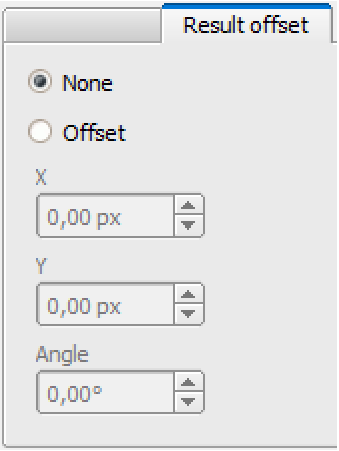
Fig. 89: Alignment Pattern matching, tab Parameters

4.6.2.2.3 Result offset

With the Result offset, the final position of a found object can be modified. This can be useful when working with a robotic coordinate systems and needing to define a 'pick point' for example.

Settings in result offset tab:

Parameters	Functions
None	Automatically determines the of the Region Of Interest or ROI.
Offset	Free selectable position (graphically or by value input, e.g. for robot gripper use) • X: Offset in X- direction (ref. ROI center) • Y: Offset in Y- direction (ref. ROI center) • Angle: angle offset (ref. ROI orientation)



Result offset

● None

○ Offset

X

0,00 px

Y

0,00 px

Angle

0,00°

Fig. 90: Result offset

4.6.2.3 Alignment Edge detector

The alignment determines the object position and thus the tracked coordinate system based on the intersection point of edges in the image. Angle positions up to approx. ±20% deviation (depending on the object) can be compensated.

4.6.2.3.1 Structure of alignment method Edge detector

The alignment “Edge detector” is carried out via “Probes”. Depending on the probe type, there are between one and three probe/s. The search region of a probe is indicated by the yellow frame (ROI). Within this ROI the object is searched and the edge of the object is scanned. The scanning is performed in the direction of the yellow arrow, the “search direction”. This yellow arrow can also be used to turn the search region of the detector. From the starting point of the ROI, search stripes (number can be set as desired) are sent out in search direction. If the search stripe touches the edge of the object, the “touching point” of the search stripe is marked with a cross at this point. Depending on the number and the setting, there may be a “winner search stripe”, the touching point of which is shown in bold. Which edge of an object is touched is indicated at the “scanning line” in the search direction. If an object is not scanned from both X and Y direction but only from one direction, the second scanning line is at the center of the search region. The arrows with the origin at the intersection of the scanning lines form the aligned coordinate system. In the following figure, the structure of the alignment Edge detector is visualized.

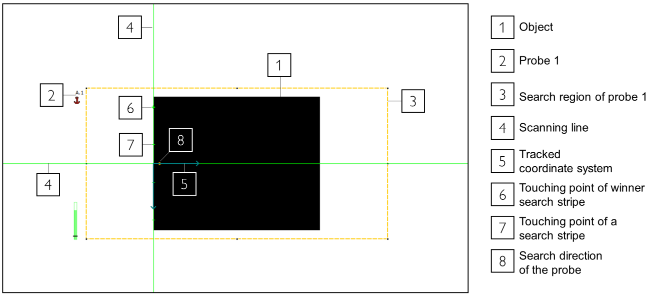

4
1
2
6
3
7
8
4
5
1 Object
2 Probe 1
3 Search region of probe 1
4 Scanning line
5 Tracked coordinate system
6 Touching point of winner search stripe
7 Touching point of a search stripe
8 Search direction of the probe

Fig. 91: Structure of Edge detector

4.6.2.3.2 Tab Color channel

In the color channel tab, a color image (3 channel) can be converted to a gray value image (1 channel). In contrast to the gray value image of a monochrome VISOR® vision sensor, contrasts can be significantly increased. The highlighting of a color can be set individually for each detector. Thus, the flexibility compared to the use of optical color filters is significantly higher.

The image displayed is dependent on the selected detector.

- Color detectors: Display always colored.
- Object detectors: Monochrome image, display dependent upon selected color model and color channels.

The following parameters can be configured in the Color channel tab:

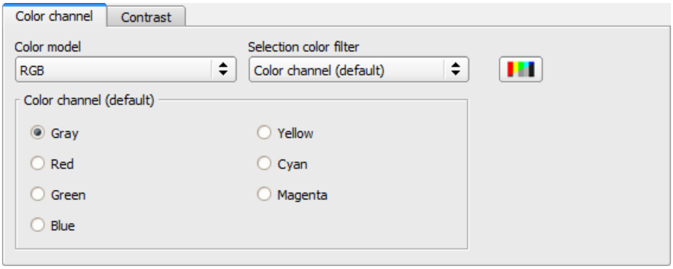
Parameter	Function
Color model	Color models: RGB, Color model RGB (Page 324) HSV, Color model HSV (Page 325) LAB, Color model LAB (Page 325)
Selection color filter	Depending on the color space, all or part of the following color filters are available: Color channel (default) Color distance Binarization
▨	Switching the image between color and monochrome.

4.6.2.3.2.1 Selection color filter

The following color filters are available:

Color channel (default)

The selected color channel is used as a gray value image.



Color model: RGB

Selection color filter: Color channel (default)

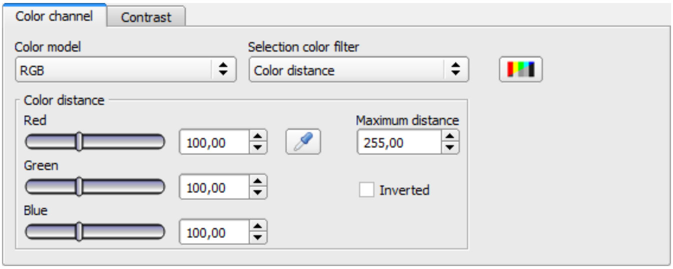
Color channel (default)

● Gray
○ Red
○ Green
○ Blue
○ Yellow
○ Cyan
○ Magenta

Fig. 92: Color filter, Color channel (default)

Color distance

A color is selected as reference color by specifying the color model values or by pipette. The gray value image indicates the distance of each pixel to this reference color. Typical application: Segmentation of characters for OCR.



Color model: RGB

Selection color filter: Color distance

Color distance

Red: 100,00

Green: 100,00

Blue: 100,00

Maximum distance: 255,00

□ Inverted

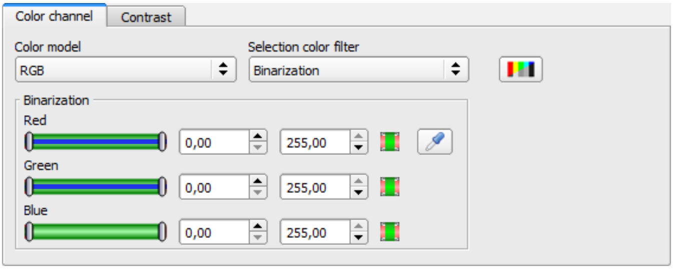
Fig. 93: Color filter, Color distance

Parameter		**Function**
Red Green Blue	Lightness A B	Color channels: The color channel can be set via the slider or by entering a value (default 0).
Pipette button		With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.
Maximum dis-		Distance of the current color versus the taught-in color. Colors that will

Parameter	Function
tance	exceed the maximum color distance will be black or white depending on the setting of “Inverted”.
Inverted	Inversion of the color distance image.

Binarization

A color range is selected. All pixels within this color range become white. Pixels with deviating color values become black.



Color model: RGB

Selection color filter: Binarization

Binarization

Red: 0,00 – 255,00

Green: 0,00 – 255,00

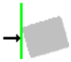
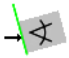
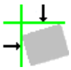
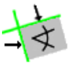
Blue: 0,00 – 255,00

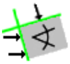
Fig. 94: Color filter, Binarization

Parameter			Function
Red Green Blue	Hue Saturation Value	Lightness A B	Determination of the color range. The color ranges can be set via the slider or by entering a value.
Inverting button			The current setting is inverted when selecting the button.
Pipette button			With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.

4.6.2.3.3 Alignment Edge detector, tab Parameters

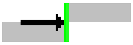
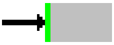
The probe type must be selected to perform the edge detection. The probe type determines which change in position of the object should be aligned: Shift in one or two directions, rotation. The following probe types are available:

Probe type		Function	Recommended use with varying object position		
			... in one direction	... in two directions	... with rotation
1		**One probe: Alignment by shift in one direction** Alignment of object by shift in one direction. The position of the scanning line is determined by the search direction of the detector. The other scanning line is in the middle of the search region (ROI).	✓		
2		**One probe: Alignment by shift in one direction and rotation** Alignment of object by shift in one direction and rotation. The position of the scanning line is determined by the search direction of the detector. The other scanning line is in the middle of the search region (ROI).	✓		✓
3		**Two probes: Alignment by shift in two directions** Alignment of object by shift in two directions. The position of the scanning line in X direction of the coordinate system is determined by probe 1. The position of the scanning line in Y direction of the coordinate system is determined by probe 2. The origin of the coordinate system lies at the intersection of the two scanning lines.		✓	
4		**Two probes: Alignment by shift in two directions and rotation** Alignment of object by shift in two directions and rotation. The position of the scanning line in X direction of the coordinate system is determined by probe 1. The position of the scanning line in Y dir-		✓	✓

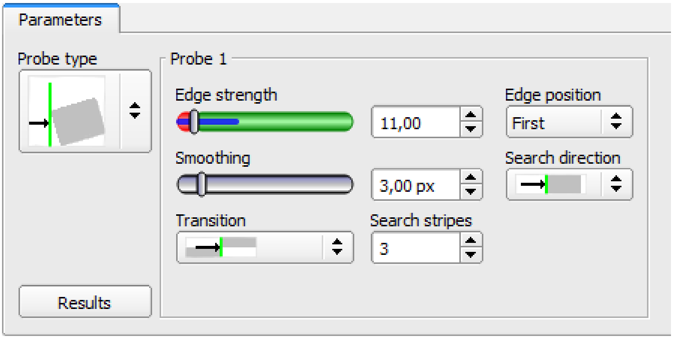
Probe type		Function	Recommended use with varying object position		
			... in one direction	... in two directions	... with rotation
		ection of the coordinate system is determined by probe 2. The origin of the coordinate system lies at the intersection of the two scanning lines. In addition, the orientation of the object is determined. Probe 2 is rotated and moved according to the object movement. The position of probe 2 is aligned relative to the position and orientation of the scanning line of probe 1.			
5		**Three probes: Alignment by shift in two directions and rotation** Alignment of object by shift in two directions and rotation. A straight line is drawn through the touching points of the winner search stripes of probe 1 and 2. This scanning line (12) determines the position and orientation of the coordinate system. The origin of the coordinate system lies at the intersection of scanning line 12 and scanning line 3. Probe 3 is rotated and moved according to the object movement. The position of probe 3 is aligned relative to the position and orientation of the scanning line 12.		✓	✓

After selecting the probe type, the corresponding parameters must be determined.
The following parameters can be configured in the Parameters tab:

Parameter	Function
Edge strength	Edge strength / contrast at which an edge should be detected as an edge.
Smoothing	The edge contour is smoothed in search direction. With larger val-

Parameter	Function
	ues, noisy edges, blurred edges or edges that are not perpendicular to the search direction are detected more reliably. In addition, light-dark-light or dark-light-dark transitions which are close together can be ignored with larger values. Thus, interfering edges, e.g. scratches, can be hidden. The effect of smoothing can be displayed graphically using the button “Results”.
Transition	With the “Transition” parameter the edge transition can be determined.
• Both directions 	Edge transition from light to dark and vice versa.
• Light → Dark 	Edge transition from light to dark.
• Dark → Light	Edge transition from dark to light.
Search stripes	Number of parallel search stripes into which the width of the search region is divided. Edge detection is carried out in each search stripe and the first edge is decisive.
Edge position	The parameter “Edge position” determines which edge is to be detected from the search direction. It is determined how the winner search stripes and thus the edge position are determined.
• First	The first edge in search direction is detected. The distances from the beginning of the search region to the touching points of all search stripes in search direction are determined. The winner search stripe is the one with the shortest distance to the beginning of the search region.
• Last	The last edge in the search direction is detected. The distances from the beginning of the search region to the touching points of all search stripes in search direction are determined. The winner search stripe is the one with the longest distance to the beginning of the search region.
• Median	The distances in search direction from the beginning of the search region to the touching points of all search strips are determined. Then the median value of these distances is formed.

Parameter	Function
• Mean	The distances in search direction from the beginning of the search region to the touching points of all search strips are determined. Then the mean value of these distances is formed.
Orientation	The "Orientation" parameter defines the type of the scanning line determination.
• Best-fit line	In this setting, the scanning line is determined by placing a best-fit line through all search stripes.
• Edge guide	In this setting, a scanning line is determined which acts like a mechanical edge stop. This makes it possible to achieve more robust results for convex-shaped edges than with a simple best-fit line.
Search direction	This parameter determines the search direction of the probes. From this direction the object edge is aligned. All probes can be rotated with the small black arrow.
• →|	The search direction takes place in only one direction, the direction of the yellow arrow (ROI). The touching points and thus the origin of the coordinates lie at an edge of the object.
• →|←	For each search stripe, a touching point is determined from both directions of the probe. Then the center between these touching points is determined. The origin of the coordinate system is at the center of the winner search stripe, i.e. in the object.
Results	Opens the results and histogram window. For more information, see Caliper results / Histogram display (Page 218)



Edge strength: 11,00

Edge position: First

Smoothing: 3,00 px

Search direction: →|

Transition: →|

Search stripes: 3

Fig. 95: Alignment Edge detector, tab Parameters

Optimization of execution speed:

- Search zone for position (yellow frame) only as large as required
- Reduce search stripes
- Reduce smoothing value
- Reduce resolution to QQVGA, QVGA or VGA instead of WVGA (Attention: global parameter, affects all detectors!)

Improve robust detection:

- If edges are blurred: increase smoothing value
- If interfering edges such as scratches are detected: increase switching threshold or / and smoothing value
- If edge is not vertical to search direction: increase search stripes

Effect of Number "search stripes"

"Search stripes" represent the how many search stripes the width of the search area is divided into. Edge detection is processed in each search stripe over the whole width. The first edge which is detected is the overall result of all search stripes. By increasing the number of search stripes it's assures that the first edge in the search area is found. By increasing search stripes it may happen that the found edge strength fluctuates, e.g. if only the half width of the search area is covered with an edge. This is because the first – not the strongest – edge which is above the threshold is detected.

Further information on edge detection see chapter: Further explanations to Edge detector (alignment) (Page 573).

4.6.2.4 Alignment Contour detection

This detector is suitable for detecting contours by means of edges. The contours of an object in the search area are taught and stored in the sensor. In Run mode the sensor searches the position of the best fit with the taught contour. If the fit is higher than the selected threshold the result is positive. The function contour detection can work incomplete 360° angular detection mode. So the object can be rotated in any angle. (The angular settings must be set accordingly!)

4.6.2.4.1 Tab Color channel

In the color channel tab, a color image (3 channel) can be converted to a gray value image (1 channel). In contrast to the gray value image of a monochrome VISOR® vision sensor, contrasts can be significantly increased. The highlighting of a color can be set individually for each detector. Thus, the flexibility compared to the use of optical color filters is significantly higher.

The image displayed is dependent on the selected detector.

- Color detectors: Display always colored.
- Object detectors: Monochrome image, display dependent upon selected color model and color channels.

The following parameters can be configured in the Color channel tab:

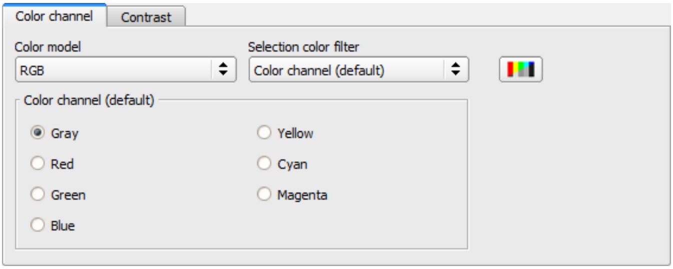
Parameter	Function
Color model	Color models: RGB, Color model RGB (Page 324) HSV, Color model HSV (Page 325) LAB, Color model LAB (Page 325)
Selection color filter	Depending on the color space, all or part of the following color filters are available: Color channel (default) Color distance Binarization
▮▮▮	Switching the image between color and monochrome.

4.6.2.4.1.1 Selection color filter

The following color filters are available:

Color channel (default)

The selected color channel is used as a gray value image.



Color channel | Contrast

Color model: RGB

Selection color filter: Color channel (default)

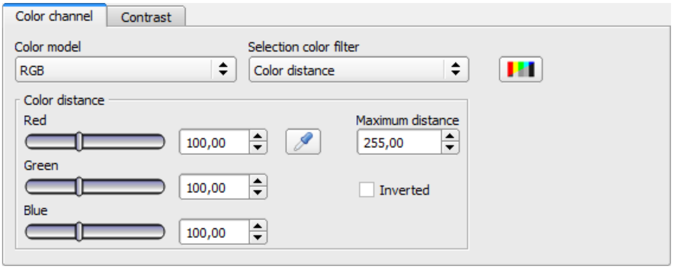
Color channel (default)

● Gray
○ Red
○ Green
○ Blue
○ Yellow
○ Cyan
○ Magenta

Fig. 96: Color filter, Color channel (default)

Color distance

A color is selected as reference color by specifying the color model values or by pipette. The gray value image indicates the distance of each pixel to this reference color. Typical application: Segmentation of characters for OCR.



Color model: RGB

Selection color filter: Color distance

Color distance

Red: 100,00

Green: 100,00

Blue: 100,00

Maximum distance: 255,00

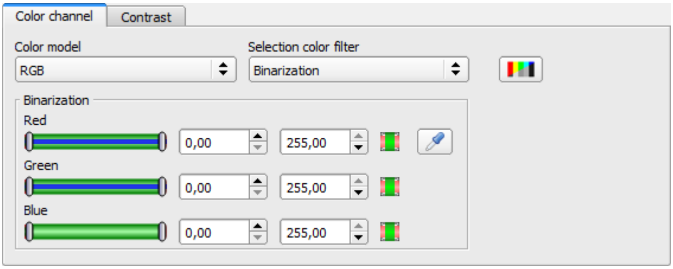
□ Inverted

Fig. 97: Color filter, Color distance

Parameter		Function
Red Green Blue	Lightness A B	Color channels: The color channel can be set via the slider or by entering a value (default 0).
Pipette button		With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.
Maximum distance		Distance of the current color versus the taught-in color. Colors that will exceed the maximum color distance will be black or white depending on the setting of "Inverted".
Inverted		Inversion of the color distance image.

Binarization

A color range is selected. All pixels within this color range become white. Pixels with deviating color values become black.



Color model: RGB

Selection color filter: Binarization

Binarization

Red: 0,00 255,00

Green: 0,00 255,00

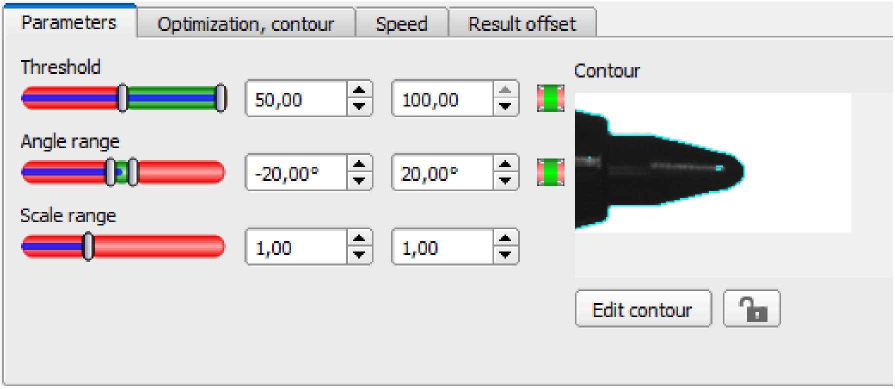
Blue: 0,00 255,00

Fig. 98: Color filter, Binarization

Parameter			Function
Red Green Blue	Hue Saturation Value	Lightness A B	Determination of the color range. The color ranges can be set via the slider or by entering a value.
Inverting button			The current setting is inverted when selecting the button.
Pipette button			With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.

4.6.2.4.2 Alignment method Contour detection, tab Parameters

The most important parameters for contour detection can be set in the Parameters tab.



Threshold: 50,00 – 100,00

Angle range: -20,00° – 20,00°

Scale range: 1,00 – 1,00

Fig. 99: Alignment Contour detection, tab Parameters

The pale blue edges in the lower right corner (high contrast changes in the image) have been identified and marked based on the parameter settings. The found edges / contour can be influenced by changing these parameters, or by the function “Edit contour”. The VISOR® now searches this contour in the search area (yellow frame).

The following parameters can set in the Parameters tab:

Parameters	Function and setting possibilities
Threshold	Zone for required match of found contour with taught contour.
Angle range	Angle range in which search is carried out (large range means longer process time)

Parameters	Function and setting possibilities
Scale range	Detection also of enlarged or reduced objects in a given scale range.
Edit contour	By editing contour, parts of the search area can be masked out. The parts which are not relevant for an examination can removed like using an eraser. Masks can also be inverted.
Lock	Lock / Unlock Contour: In locked status the taught contour is protected against (unintentional / accidential) changing, such as modification of the teach region. Unlock to modify taught contour.

Optimization of the execution speed:

- Search zone for position (yellow frame) only as large as necessary. Please note: The contour is found as long as the center point of the pattern is inside the search area!
- Search zone for angle only as large as necessary
- Search zone for scale only as large as necessary
- Reduce resolution to CGA instead of VGA (Attention: Global parameter, affects all detectors!)
- Set “accurate – fast” to fast
- Increase value “Min. contrast pattern”. Take care that the relevant contours are still visible in the display.
- Increase value “Min. contrast image”.
- Use alternate reference pattern, e.g. with higher contrast, so that “Min. contrast pattern” and “Min. contrast image” can be increased.

Robust detection:

- Search zone for position (yellow frame) sufficiently large?
- Search zone for angle sufficiently large?
- Search zone for scale sufficiently large?
- Contrasts for model and image suitably set? (for model visible in sample)
- Set “accurate – fast” to accurate
- Are there several overlapping objects in the image?
- Distinctive edges available? Re-teach if necessary .
- “Min. contrast pattern” set to a suitable value? If in the taught pattern the relevant contour lines are not shown completely: decrease “Min. contrast pattern”. If there are too many contour lines shown: increase “Min. contrast pattern”.

- “Min. contrast image” set to a suitable value for the current image? If the current image(s) do have a higher / lower contrast than the taught reference image /pattern please increase / decrease the value of “Min. contrast image” accordingly.
- If in the taught pattern the relevant contour lines are not shown completely: decrease “Min. contrast pattern”. If there are too many contour lines shown: increase “Min. contrast pattern”.
- If found at wrong position: use more distinct sample, re-teach if necessary.
- If the result value is fluctuating strongly from image to image? Take care that there are no “false” edges taught (edges because of shadows, or fragments of contours, which are not desired in the contour model): This can be achieved by increasing “Min. contrast pattern” or by eliminating those false edges by function “Edit contour”.

Parameter Angle range: Rotational direction of angle

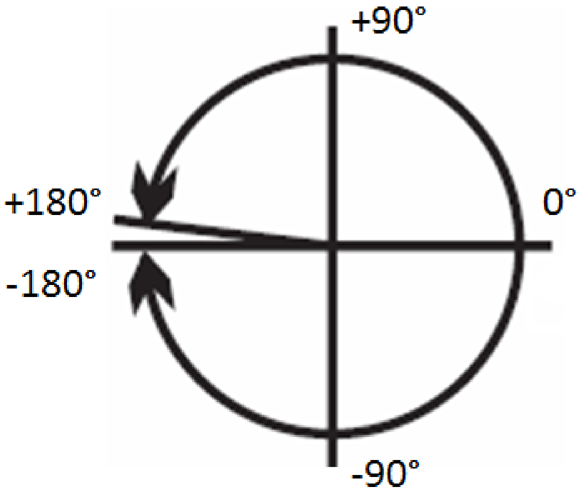
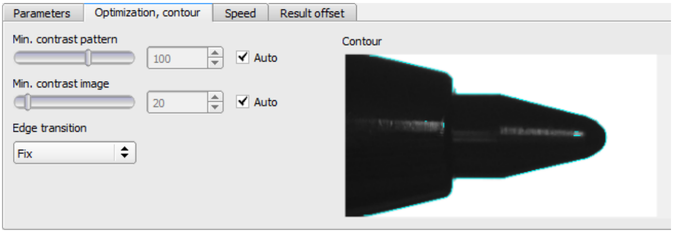


Fig. 100: Rotational direction of angle

4.6.2.4.3 Alignment Contour detection, tab Optimization, contour

In the “Optimization, contour” tab further settings for the edge transition and the contrast can be made.



Min. contrast pattern 100 ☑ Auto

Min. contrast image 20 ☑ Auto

Edge transition Fix

Fig. 101: Alignment Contour detection, tab Optimization, contour

The following parameters can be configured in the Optimization, contour tab:

Parameter	Function
Min. contrast pattern	Minimum contrast required with taught model for an edge to be detected as one.
Min. contrast image	Minimum contrast required in current image for an edge to be accepted as one.
Edge transition • Fix • Fix + inverted • Flexible	The parameter “edge transition” can be used to determine the transition between object/contour and background. Select whether the contour is to be detected only on the taught-in background (“Fix”), on the taught-in and inverted background (“Fix + inverted”) or on any background (“Flexible”). See also “Additional information:”
Auto	Automatic selection

Edge transition

Example:

A gray object is taught-in in front of a brighter background, as shown in the following figure.

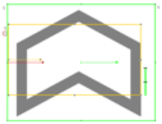
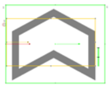
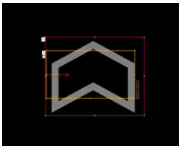
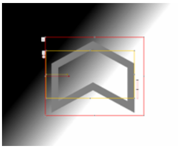
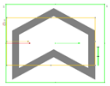
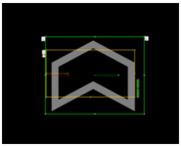
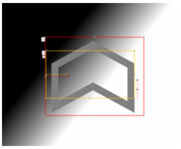
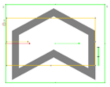
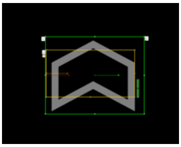
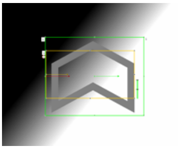


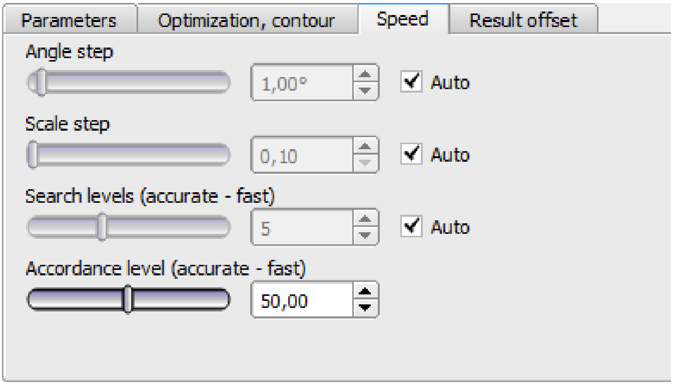
Fig. 102: Taught-in contour in front of a brighter background

The following table shows the results of the contour detector with the respective settings for the edge transition.

Settings for parameter "Edge transition"	**Bright background**	**Dark background**	**Inconsistent background**
Fix			
	Contour detector: OK	Contour detector: not OK	Contour detector: not OK
Fix + inverted			
	Contour detector: OK	Contour detector: OK	Contour detector: not OK.
Flexible			
	Contour detector: OK	Contour detector: OK	Contour detector: OK

4.6.2.4.4 Alignment Contour detection, tab Speed

Using the adjustable parameters in the Speed tab, execution time of the sensor can be altered. Adjusting the Search levels of the Speed tab alters the level of detail and corresponding time applied to a given search.



Angle step 1,00° ☑ Auto

Scale step 0,10 ☑ Auto

Search levels (accurate - fast) 5 ☑ Auto

Accordance level (accurate - fast) 50,00

Fig. 103: Alignment Contour detection, tab Speed

The following parameters can be configured in the Speed tab:

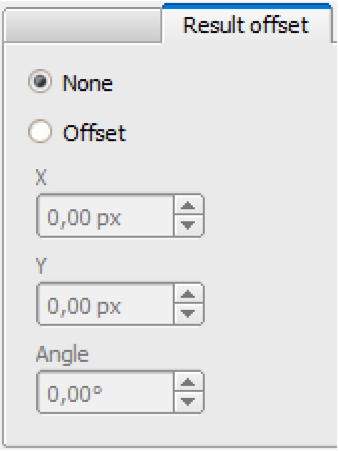
Parameter	**Function**
Angle step	Sensitivity of search throughout the selected angle range in degrees [°]
Scale step	Sensitivity of search throughout the selected scale range
Search levels (accurate - fast)	Number of search levels • High value: faster = riskier (candidates may be overlooked) • Small value: slower = less risky (all candidates)
Accordance level (accurate - fast)	Candidates that score less than indicated will automatically be rejected during the search. • High value: early rejection = quicker = riskier • Small value: late rejection = slower = less risky In case of false results this value can be decreased (more accurate).
Auto	Automatic selection

4.6.2.4.5 Result offset

With the Result offset, the final position of a found object can be modified. This can be useful when working with a robotic coordinate systems and needing to define a 'pick point' for example.

Settings in result offset tab:

Parameters	Functions
None	Automatically determines the of the Region Of Interest or ROI.
Offset	Free selectable position (graphically or by value input, e.g. for robot gripper use) • X: Offset in X- direction (ref. ROI center) • Y: Offset in Y- direction (ref. ROI center) • Angle: angle offset (ref. ROI orientation)



● None

○ Offset

X: 0,00 px

Y: 0,00 px

Angle: 0,00°

Fig. 104: Result offset

4.6.2.4.6 Alignment Contour detection, tab Gripping space

Robots grip objects, e.g. with a twin-jaw gripper, on the outer contour of the objects. Gripping with the robot may not be possible if the objects touch or overlap. The VISOR® gripping space function can be used to check whether the gripping positions on the object are available in the required size. The position of the first found object is output, in which its tracking detectors (gripping regions) are OK (according to the logical links in the overall result).

The gripping space function is available for contour alignment.

Sequence:

1. The contour alignment identifies those objects as candidates whose contour matches the taught-in contour.
2. These candidates will be sorted. The sorting takes place according to the values of “Sorting criteria” and “Sorting order” set in the “Gripping space” tab.

3. According to this order, the candidates will be checked to make sure that the tracked detectors that are assigned by alignment (e.g. clearance check) all comply. This happens under consideration of the logical links in the overall result. In the “Digital output” tab of the “Output” setup, logical links can be used to evaluate the objects. For example, free spaces for different gripping positions can be defined here. The gripping positions X-X and Y-Y are possible for the object shown in the following figure. Of these gripping possibilities, only those that are necessary for one grip can then be checked for “free”.

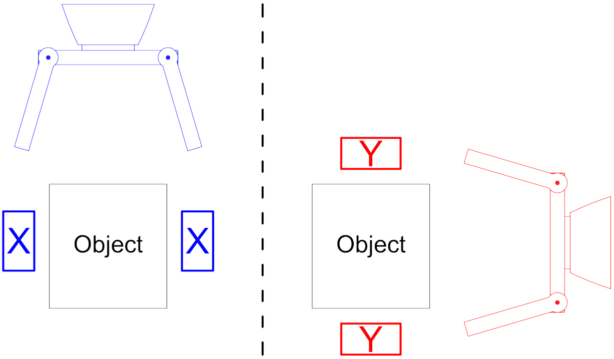
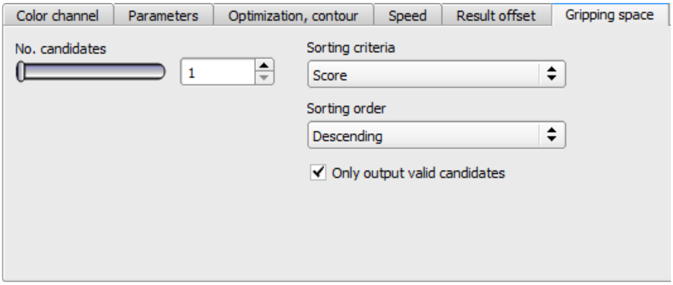
   

   Fig. 105: Possible gripping position X-X (left) and possible gripping position Y-Y (right).

   Please note: The alignment per detector can be activated or deactivated in the detector list (Default: Active). Only detectors activated here are effective for the gripping space check.

4. The position data of the first object that meets all these criteria are output and the search is terminated at this point.

The precondition for the successful finding of an object is at least one object per image/evaluation for which the total result, i.e. also aligned detectors are “i.O.”!



No. candidates: 1

Sorting criteria: Score

Sorting order: Descending

☑ Only output valid candidates

Fig. 106: Alignment method Contour detection, tab Gripping space

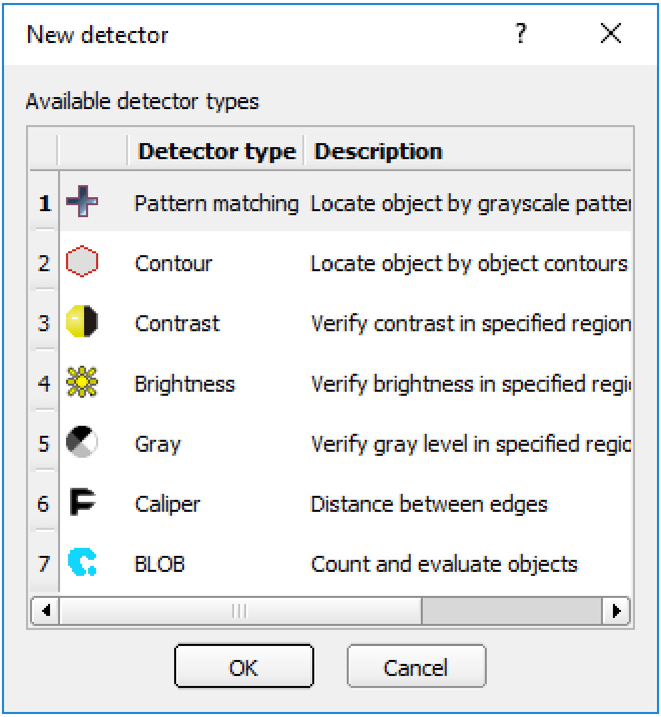
Following parameters can be set in the Gripping space tab:

Parameter	**Function**
No. candidates	With the parameter “No. candidates”, the maximum number of target objects can be determined. If there are more objects in the field of view than specified in the parameter, only as many objects as specified in the parameter are evaluated. Please note: Only one object will be output - the first one that, is found according to the following settings!
Sorting criteria	Sorting criteria according to which objects are “pre-sorted”.
• Score	Score
• Position X	Position X
• Position Y	Position Y
• Angle	Angle
• Scale	Scale
Sorting order	Sorting order for the selected sorting criteria.
• Ascending	The values of the sorting criteria are sorted in ascending order.
• Descending	The values of the sorting criteria are sorted in descending order.
Only output valid candidates	If this checkbox is activated, only objects whose score value is above the set threshold (tab “Parameters”) are displayed and output. It can be used e.g. for parameter optimization.

4.6.3 Detectors

Each job contains one or several inspection steps (detectors), which you can define here. By clicking on the "Defector" button, or the "New" button under the Detector list, a window with a list of all available detectors opens. Drawings in the image (yellow, red frames etc.) can be activated or deactivated for any detector or category in the menu item “View/all drawings”. With “View/drawings of current detector only”, all drawings on the screen can be deactivated with the exception of the detector currently being processed.

For information to the meaning and adjustment of the different frames see chapter: Search and parameter zones .



Available detector types

		Detector type	Description
1	✚	Pattern matching	Locate object by grayscale patte
2	⬡	Contour	Locate object by object contours
3	◐	Contrast	Verify contrast in specified region
4	✻	Brightness	Verify brightness in specified regi
5	◐	Gray	Verify gray level in specified regic
6	F	Caliper	Distance between edges
7	C	BLOB	Count and evaluate objects

Fig. 107: Detector list for selection (here Object sensor)

4.6.3.1 Creating and adjusting detectors

Types of detector:

- Detector Pattern matching (Page 149)
- Detector Contour, tab Contour (Page 163)
- Detector Contrast (Page 171)
- Detector Brightness (Page 183)
- Detector Gray (Page 177)
- Detector BLOB, Introduction (Page 189)
- Detector Caliper (Page 211)
- Detector Barcode (Page 219)
- Detector Datacode (Page 229)
- Detector OCR (Page 239)

- Detector Color area (Page 268)
- Detector Color list (Page 272)
- Detector Color value (Page 266)
- Detector Wafer (Page 254)
- Detector Busbar (Page 262)

Create new detector:

1. Click on “New” button under the selection list in the configuration window and select the type of detector required. A new detector entry appears in the selection list.
2. Edit the name of the detector by double clicking on “Name”

Configure detector:

1. Activate the detector in the selection list.
2. Graphically define the appropriate search and parameter zone within the image.
3. Configure the detector by entering / adjusting the parameters in the Parameters and if necessary Advanced tabs in the configuration window. Which tabs are shown depends on the type of detector selected.

Functions for administration of detectors:

Control panel	Functions
New	Adds new detector > dialogue box with above-mentioned detector selection list appears
Copy	Copies all parameters from one detector to one or several others. The parameter zones are not copied. All detectors must be from the same type. Copy process: Create all desired destination detectors; they must be of the same type as the source detector. Mark source detector in the list Click to button “copy” A list will appear, mark all desired destination detectors. (To select several press “Ctrl” key) Click “Copy” to confirm
Reset	Resets parameters, search and parameter zones of selected detector to standard values
Delete	Deletes the selected detector
Delete all	Deletes all of the detectors in the list

Information:

“Flash x.x/yyyy.y kB” appears in the bottom corner of the screen, indicating first the memory used by the current configuration x.x), and the memory available on the sensor (yyyy.y) in kB. Should the memory used exceed the available memory, this indicator switches to red as there is not enough space for the current settings on the sensor. In this case you can delete other jobs from the sensor before transfer.

Drawings in the image (yellow, red frames etc.) can be activated or deactivated for any detector or category in the menu item “View/all drawings”. With “View/drawings of current detector only”, all drawings on the screen can be deactivated with the exception of the detector currently being processed.

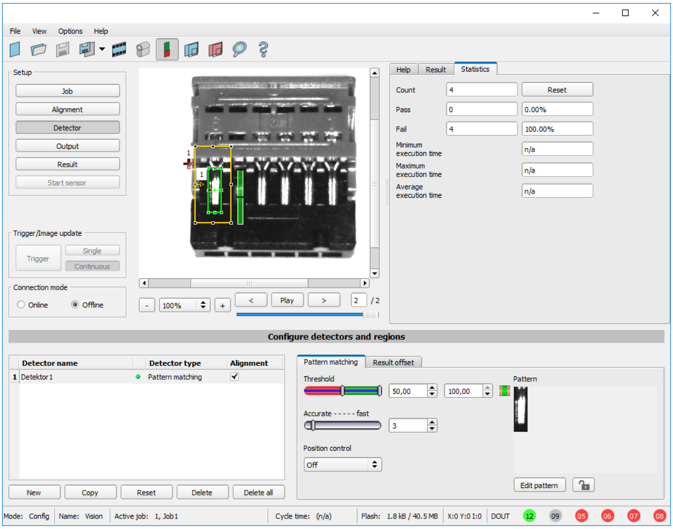
4.6.3.2 Selecting a suitable detector

The following detectors are available in SensoConfig

Type of detectors	**Description**
Pattern matching	Part detection using pattern matching, X- and Y- translational
Contour detection	Part detection using object contour, up to 360° rotation
Contrast	Evaluation of contrast in selected search zone
Brightness	Evaluation of brightness in selected search zone
Gray level	Evaluation of gray values in selected search zone
BLOB	Count and evaluate objects
Caliper	Distance between edges
Barcode	Barcode reading 1D Codes (Code reader)
Datacode	Data code reading Datacodes (Code reader)
Optical Character Recognition (OCR)	Optical character recognition (Code reader)
Color Area	Color verification inside area
Color List	Color verification inside list
Color Value	Output of color values
Wafer	Position check and control of wafers (Solar sensor)
Busbar	Position check and control of busbars (Solar sensor)

4.6.3.3 Detector Pattern matching

This detector is suitable for the detection of patterns of any shape, even with shapes that do not have distinctive edges or contours.



Count	4	
Pass	0	0.00%
Fail	4	100.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Detector name	Detector type	Alignment
1	Detektor 1	Pattern matching	☑

Threshold 50,00 100,00

Accurate - - - - - fast 3

Position control Off

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 1.8 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 108: Detector Pattern matching

4.6.3.3.1 Detector Pattern matching, tab pattern matching

Parameters	Functions
Switching threshold min/max	Zone for the required concordance of the pattern found with the pattern taught.
Accurate - fast	Number of search levels / coarsening levels. 0 = automatic selection Higher value: faster = riskier (overlook candidates) Smaller value: slower = less risky (all candidates)
Position check	Checks whether the pattern found is in the right position. If position check is activated, the position frame is shown in blue (either rectangular or elliptic).
Pattern	Shows the taught pattern = contents of the red frame
Edit pattern	By editing the mask you can mask out regions of the search area. The regions which are not relevant for this examination can be painted out like using an eraser. Masks can also be inverted.

Parameters	Functions
Lock	Lock / Unlock Pattern: In locked status the taugth pattern is protected against (unintentional / accidential) changing. by e.g. modification of the teach region. Unlock to modify taught pattern.

For newly generated detectors, all parameters are preset as standard values, suitable for many applications.

Optimisation Pattern matching:

Execution speed:

- Search zone for position (yellow frame) is only as large as necessary: Attention: The search area marks the area where the center point of the pattern is searched!
- Reduce resolution to QVGA instead of VGA (Attention: Global parameter, affects all detectors!)
- Set "accurate – fast" to fast

Robust pattern detection:

- Select the search region for position (yellow frame) sufficiently large.
- Reduce search levels.
- Select a distinctive gray value pattern, re-teach if necessary.
- If found at wrong position: use distinct sample, re-teach if necessary.

If, directly after teach, the found position (green frame) is not identical with teach area (red frame) the slider "Accurate – fast" should be set to "Accurate".

4.6.3.3.2 Tab Color channel

In the color channel tab, a color image (3 channel) can be converted to a gray value image (1 channel). In contrast to the gray value image of a monochrome VISOR® vision sensor, contrasts can be significantly increased. The highlighting of a color can be set individually for each detector. Thus, the flexibility compared to the use of optical color filters is significantly higher.

The image displayed is dependent on the selected detector.

- Color detectors: Display always colored.
- Object detectors: Monochrome image, display dependent upon selected color model and color channels.

The following parameters can be configured in the Color channel tab:

Parameter	Function
Color model	Color models:

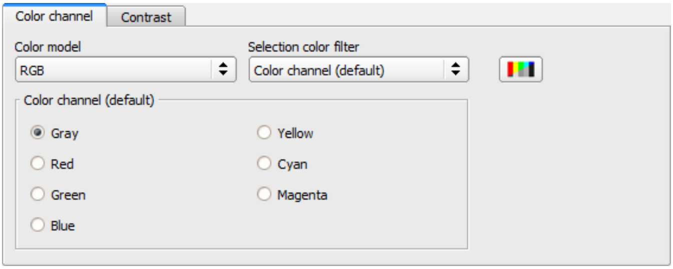
Parameter	Function
	RGB, Color model RGB (Page 324) HSV, Color model HSV (Page 325) LAB, Color model LAB (Page 325)
Selection color filter	Depending on the color space, all or part of the following color filters are available: Color channel (default) Color distance Binarization
🟥🟨🟩⬛	Switching the image between color and monochrome.

4.6.3.3.2.1 Selection color filter

The following color filters are available:

Color channel (default)

The selected color channel is used as a gray value image.



Color model: RGB

Selection color filter: Color channel (default)

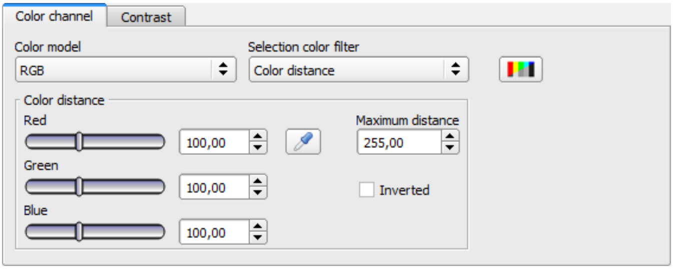
Color channel (default)

- ● Gray
- ○ Red
- ○ Green
- ○ Blue
- ○ Yellow
- ○ Cyan
- ○ Magenta

Fig. 109: Color filter, Color channel (default)

Color distance

A color is selected as reference color by specifying the color model values or by pipette. The gray value image indicates the distance of each pixel to this reference color. Typical application: Segmentation of characters for OCR.



Color model: RGB

Selection color filter: Color distance

Color distance

Red: 100,00

Green: 100,00

Blue: 100,00

Maximum distance: 255,00

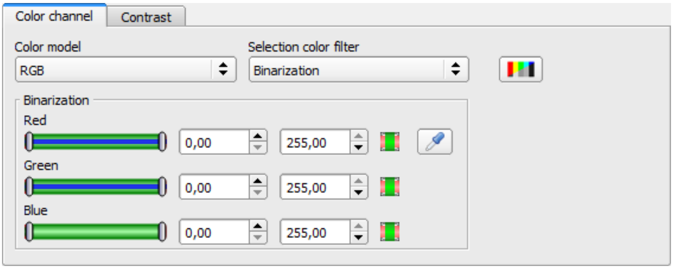
□ Inverted

Fig. 110: Color filter, Color distance

Parameter		Function
Red Green Blue	Lightness A B	Color channels: The color channel can be set via the slider or by entering a value (default 0).
Pipette button		With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.
Maximum distance		Distance of the current color versus the taught-in color. Colors that will exceed the maximum color distance will be black or white depending on the setting of “Inverted”.
Inverted		Inversion of the color distance image.

Binarization

A color range is selected. All pixels within this color range become white. Pixels with deviating color values become black.



Color model: RGB

Selection color filter: Binarization

Binarization

Red: 0,00 255,00

Green: 0,00 255,00

Blue: 0,00 255,00

Fig. 111: Color filter, Binarization

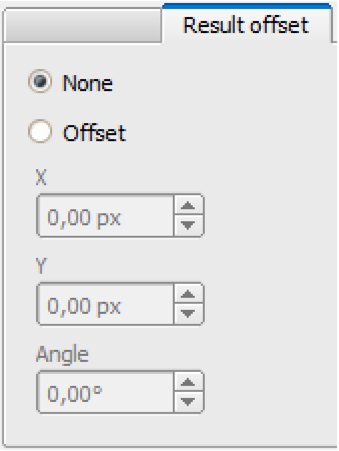
Parameter			Function
Red Green Blue	Hue Saturation Value	Lightness A B	Determination of the color range. The color ranges can be set via the slider or by entering a value.
Inverting button			The current setting is inverted when selecting the button.
Pipette button			With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.

4.6.3.3.3 Result offset

With the Result offset, the final position of a found object can be modified. This can be useful when working with a robotic coordinate systems and needing to define a 'pick point' for example.

Settings in result offset tab:

Parameters	Functions
None	Automatically determines the of the Region Of Interest or ROI.
Offset	Free selectable position (graphically or by value input, e.g. for robot gripper use) • X: Offset in X- direction (ref. ROI center) • Y: Offset in Y- direction (ref. ROI center) • Angle: angle offset (ref. ROI orientation)



Result offset

● None

○ Offset

X

0,00 px

Y

0,00 px

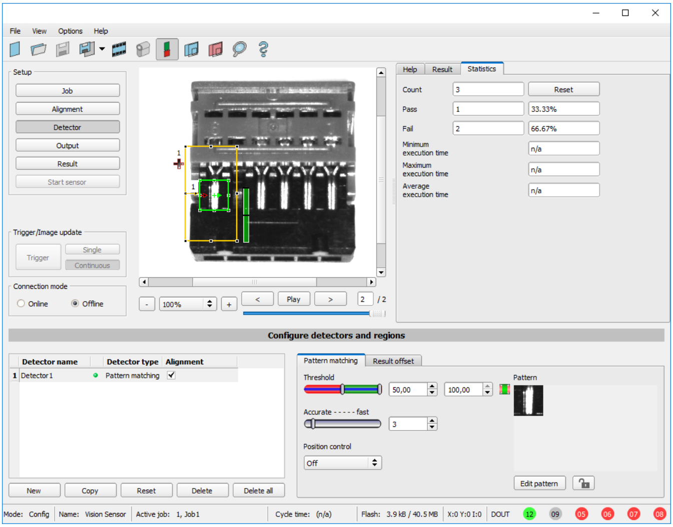
Angle

0,00°

Fig. 112: Result offset

4.6.3.3.4 Pattern matching application

In this example a metal contact (left side) in a black plastic part is taught as pattern. It is detected with a high score value, as the metal contact is mounted. (Threshold near 100%)



Count	3	Reset
Pass	1	33.33%
Fail	2	66.67%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Detector name	Detector type	Alignment
1	Detector 1	● Pattern matching	☑

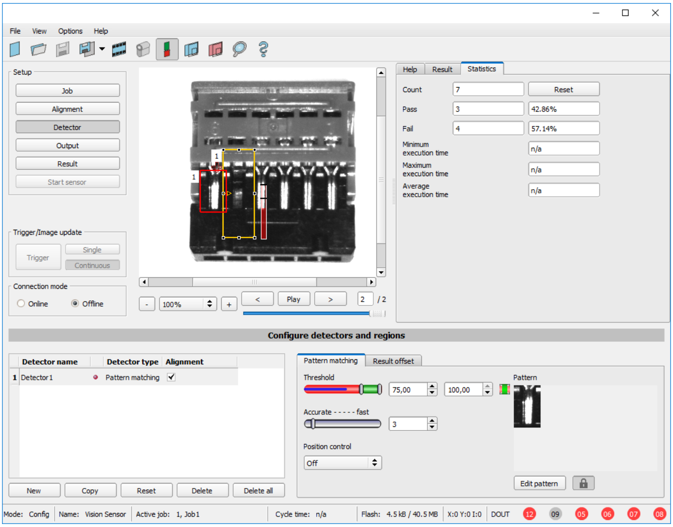
Threshold: 50,00 / 100,00

Accurate - - - - - fast: 3

Position control: Off

Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 3.9 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 113: Pattern matching, application example, positive result



Count	7	Reset
Pass	3	42.86%
Fail	4	57.14%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Detector name	Detector type	Alignment
1	Detector 1	● Pattern matching	☑

Threshold: 75,00 / 100,00

Accurate - - - - - fast: 3

Position control: Off

Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: n/a | Flash: 4.5 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 114: Pattern matching, application example, negative result

If the same pattern matching is performed at a position, where the metal contact searched for is not mounted, the score value does not reach the threshold and a negative result is given. With the function “Pattern matching”, the detection is made by the gray values of the pixels at the corresponding position in the image. As here the inner, shiny and therefore bright region does not exist, and instead of this the gray values of the pixels in the corresponding position do have lower (darker) values, the score value is significantly lower than with the contact mounted.

But, as big regions of the search area are matching (the outer dark frame of the black plastic) the score value is not zero, but approx. 70%.

The settings in these examples are just to illustrate the function of the detector pattern matching. In real operation, these settings should be optimized further (e.g. by smaller search and / or feature regions >> relevant pattern gets more significant, etc.).

By Teaching the pattern inside the red frame, it gets stored in the sensor as reference pattern. Size and position of the reference pattern is defined by the red frame. In Run mode the ACR 300i tries the find the best fit of the reference with any region in the image. Depending on the settings of the threshold, the object / feature is detected or not. The function pattern matching does not work with rotated images; it´s tolerance is limited to an angle of approx. +/- 5°. Patterns with higher angular deviation are not detected. This behaviour can be used to detect if a part is in correct orientation in input process.

Example:

The following pattern was taught:

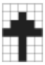


Fig. 115: Pattern, reference

With the following three examples, the object is detected with 100% concordance, as the taught pattern is exactly the same, even though it is in another place on the image.

Only offset in X or Y direction and not rotated.

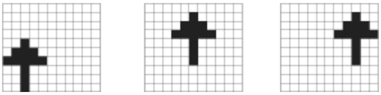


Fig. 116: Pattern, positive result

With the three now following examples in the second row, the object is also detected, but with less than 100% concordance, as it deviates from the taught pattern in some pixels. Good or bad results are supplied according to the setting of the threshold value (degree of concordance).

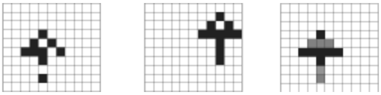


Fig. 117: Pattern, limit case

Pattern detection tolerates a ± 5 degree rotation. This means, the images in the bottom row were also detected, although the actual degree of concordance with the sample image is less than 100%, despite 100% pixel concordance.

Patterns with a larger degree of rotation are not detected.

This can be used as a function e.g. for detection of the correct alignment of parts on feeding units.

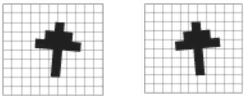


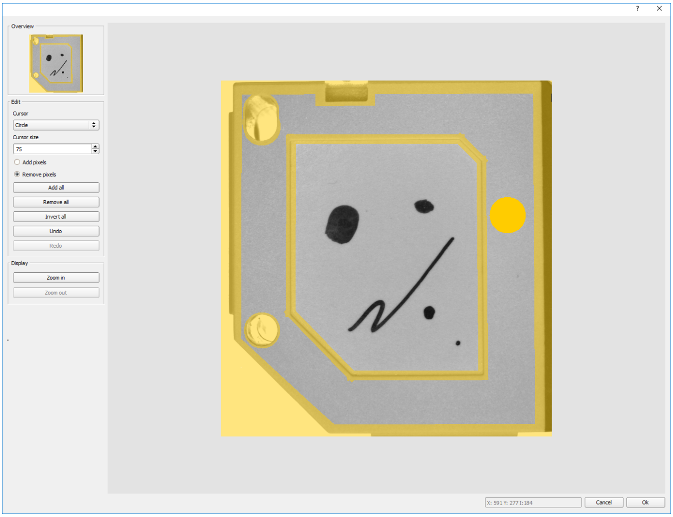
Fig. 118: Pattern, rotation

4.6.3.3.5 Function: Mask

With “Mask” function the search region can be modified. Inside the search and feature areas of the different detectors, regions can be included or excluded.

Application example:

Outer and inner contour lines as well as holes will not be considered, but all defects in the surface of the object are relevant. In this example only the non-marked regions inside of the ROI of the detector are relevant. The yellow masked regions are no longer relevant for the evaluation.



Cursor: Circle

Cursor size: 75

○ Add pixels

● Remove pixels

X: 591 Y: 277 I:184

Fig. 119: Mask

Parameter	Function
Cursor (shape)	Changing shape (square, circle or line) of the cursor
Cursor size	Changing size of the cursor
Add pixels / Remove pixels	Select if the cursor adds or removes pixels
Add all	Adds all pixels
Remove all	Removes all pixels
Invert all	Inverts all pixels
Undo	Undo function – last action
Redo	Redo function – for last undo action
Display	Select a display mode (Zoom in / Zoom out)

By the flexible selection of cursor- shape and size, as well as if an action adds or removes pixels, complex geometric or free shaped search regions can be defined in a simple and quick manner. These regions are included = relevant, or excluded (yellow) in the search area.

To use the function “Mask” the following settings are necessary for the different detector types.

Detector type	Necessary setting to use the function "Mask"
Pattern matching, Contour	Generally possible with "Edit pattern"
Contrast, Brightness, Gray, BLOB, Color value, Color area, Color list	Search region "Free shape"

Function "Mask" of search regions, examples

For the above mentioned detectors three different shapes of search regions: Circle, Rectangle and Free shape are available. The shapes: Circle and Rectangle can be rotated by picking and moving the tip of the arrow. If the shape of the search area cannot be fit in a satisfying manner to the shape of the object, the "Free shape" function can be used. With this feature, any geometry can be designed for a search area. To design the search area, the cursor can be set to a square, circle or line of any size.

In the following examples the creation of a masked search region is shown.

Example 1

Logo with relevant zones.

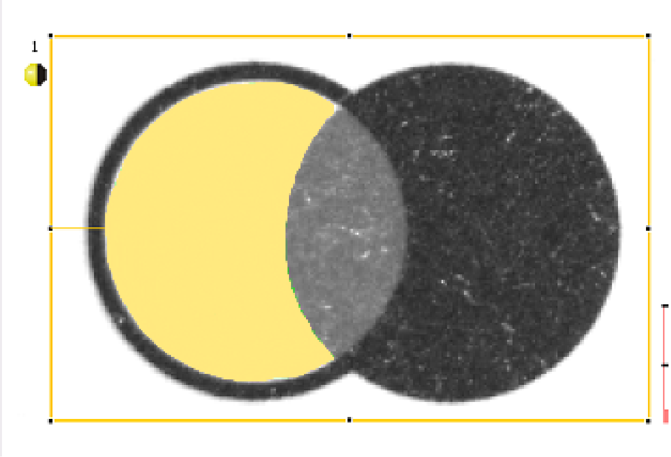


Fig. 120: Mask pattern 2

Created by one adding and one removing circle in front of the before reset mask.

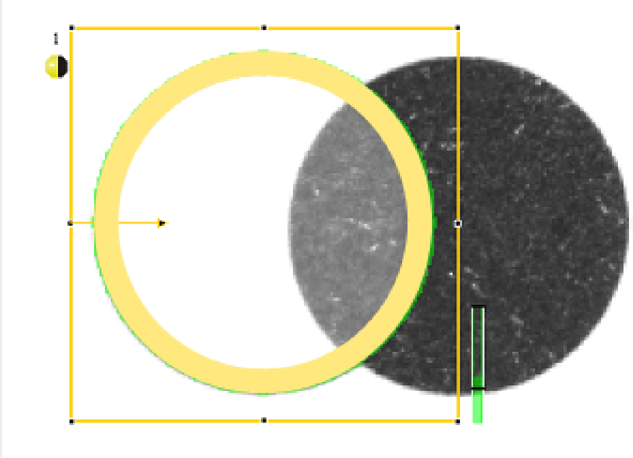


Fig. 121: Mask pattern 3

Created be one adding and one removing circle in front of the before reset mask.

Example 2

Only surface defects are relevant, object contour lines have to be masked.

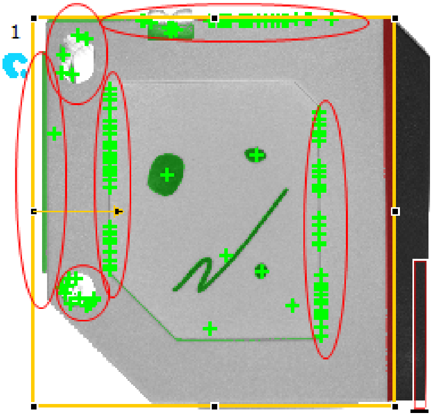
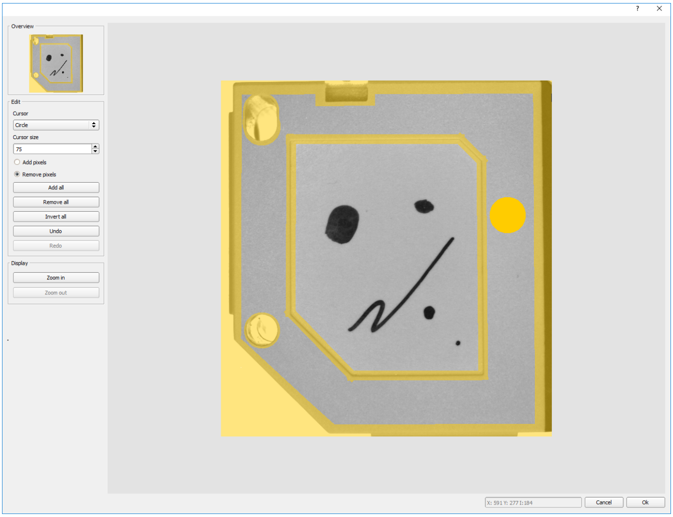


Fig. 122: BLOB without function Mask, with a BLOB detector the surface defects and the outer and inner contour lines are detected.



Cursor: Circle

Cursor size: 75

○ Add pixels

● Remove pixels

X: 591 Y: 277 I:184

Fig. 123: Function Mask: masking contour lines shall not be considered.

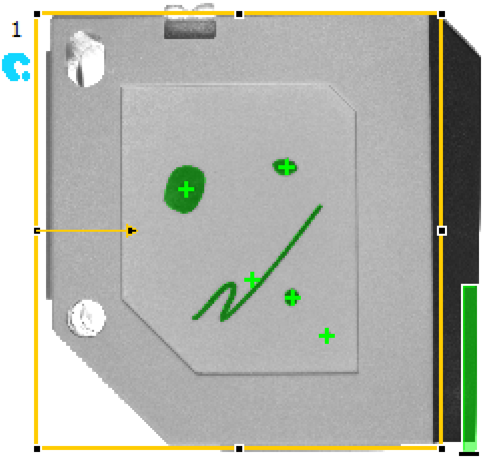


Fig. 124: BLOB with function Mask, only surface defects are detected, all contour lines are not relevant as they are now masked.

4.6.3.4 Detector Contour

This detector is suitable for detecting contours by means of edges. The contours of an object in the search area are taught and stored in the sensor. In Run mode the sensor searches the position of the best fit with the taught contour. If the fit is higher than the selected threshold the result is positive. The function contour detection can work in complete 360° angular detection mode. Thus, the object can be rotated in any angle (The angular settings must be set accordingly!).

4.6.3.4.1 Tab Color channel

In the color channel tab, a color image (3 channel) can be converted to a gray value image (1 channel). In contrast to the gray value image of a monochrome VISOR® vision sensor, contrasts can be significantly increased. The highlighting of a color can be set individually for each detector. Thus, the flexibility compared to the use of optical color filters is significantly higher.

The image displayed is dependent on the selected detector.

- Color detectors: Display always colored.
- Object detectors: Monochrome image, display dependent upon selected color model and color channels.

The following parameters can be configured in the Color channel tab:

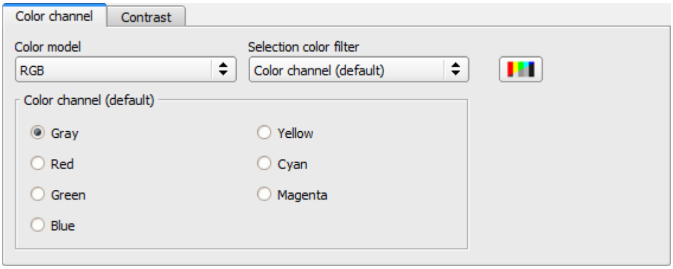
Parameter	**Function**
Color model	Color models: RGB, Color model RGB (Page 324) HSV, Color model HSV (Page 325) LAB, Color model LAB (Page 325)
Selection color filter	Depending on the color space, all or part of the following color filters are available: Color channel (default) Color distance Binarization
▨	Switching the image between color and monochrome.

4.6.3.4.1.1 Selection color filter

The following color filters are available:

Color channel (default)

The selected color channel is used as a gray value image.



Color channel | Contrast

Color model: RGB

Selection color filter: Color channel (default)

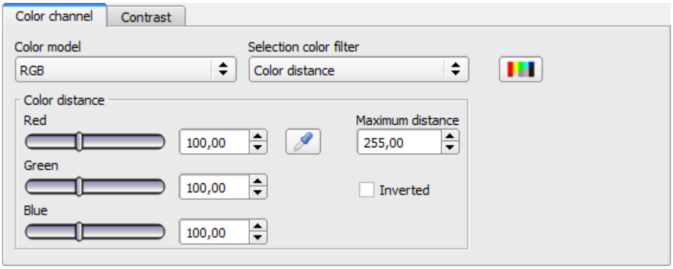
Color channel (default)

● Gray
○ Red
○ Green
○ Blue
○ Yellow
○ Cyan
○ Magenta

Fig. 125: Color filter, Color channel (default)

Color distance

A color is selected as reference color by specifying the color model values or by pipette. The gray value image indicates the distance of each pixel to this reference color. Typical application: Segmentation of characters for OCR.



Color channel | Contrast

Color model: RGB

Selection color filter: Color distance

Color distance

Red: 100,00

Green: 100,00

Blue: 100,00

Maximum distance: 255,00

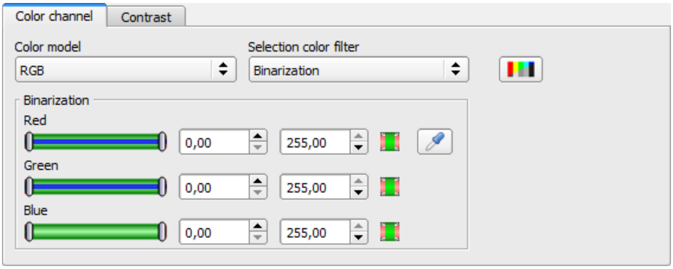
□ Inverted

Fig. 126: Color filter, Color distance

Parameter		Function
Red Green Blue	Lightness A B	Color channels: The color channel can be set via the slider or by entering a value (default 0).
Pipette button		With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.
Maximum distance		Distance of the current color versus the taught-in color. Colors that will exceed the maximum color distance will be black or white depending on the setting of "Inverted".
Inverted		Inversion of the color distance image.

Binarization

A color range is selected. All pixels within this color range become white. Pixels with deviating color values become black.



Color model: RGB
Selection color filter: Binarization

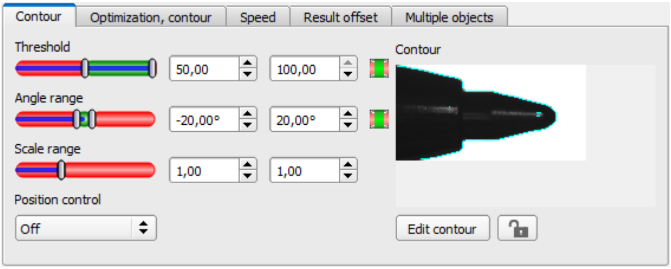
Binarization
Red: 0,00 – 255,00
Green: 0,00 – 255,00
Blue: 0,00 – 255,00

Fig. 127: Color filter, Binarization

Parameter			Function
Red Green Blue	Hue Saturation Value	Lightness A B	Determination of the color range. The color ranges can be set via the slider or by entering a value.
Inverting button			The current setting is inverted when selecting the button.
Pipette button			With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.

4.6.3.4.2 Detector Contour, tab Contour

The most important parameters for contour detection can be set in the “Contour” tab.



Threshold: 50,00 – 100,00
Angle range: -20,00° – 20,00°
Scale range: 1,00 – 1,00
Position control: Off

Fig. 128: Detector contour, tab Contour

The pale blue edges in the lower right corner (high contrast changes in the image) have been identified and marked based on the parameter settings. The found edges / contour can be influenced by changing these parameters, or by the function “Edit contour”. The VISOR® now searches this contour in the search area (yellow frame).

The following parameters can be configured in the “Contour” tab:

Parameters	**Function**
Threshold	Zone for required concordance of found contour with taught contour.
Angle range	Angle range in which search is carried out (large range means longer process time)
Scale range	Detection also of enlarged or reduced objects in a given scale range.
Position control	Checks whether the found sample is in the right position. If position control is activated, the authorised zone for the position of the found parameter is shown in a blue frame (either rectangular or elliptic). The center (green cross) of the found parameter must be situated inside the blue frame.
Contour	Shows the taught contour (red frame in field of view).
Edit contour	By edit contour, parts of the search area can be masked out. The parts which are not relevant for this examination can removed like using an eraser. Masks can also be inverted. S. also chap Function: Mask (Page 156).
Lock	Lock / Unlock Contour: In locked status the taught contour is protected against (unintentional / accidential) changing, such as modification of the teach region. Unlock to modify taught contour.

Optimization of the execution speed:

- Search zone for position (yellow frame) only as large as necessary. Please note: The contour is found as long as the center point of the pattern is inside the search area!
- Search zone for angle only as large as necessary
- Search zone for scale only as large as necessary
- Reduce resolution to CGA instead of VGA (Attention: Global parameter, affects all detectors!)
- Set “accurate – fast” to fast
- Increase value “Min. contrast pattern”. Take care that the relevant contours are still visible in the display.
- Increase value “Min. contrast image”.
- Use alternate reference pattern, e.g. with higher contrast, so that “Min. contrast pattern” and “Min. contrast image” can be increased.

Robust detection:

- Search zone for position (yellow frame) sufficiently large?
- Search zone for angle sufficiently large?
- Search zone for scale sufficiently large?
- Contrasts for model and image suitably set? (for model visible in sample)
- Set “accurate – fast” to accurate
- Are there several overlapping objects in the image?
- Distinctive edges available?, Re-teach if necessary.
- “Min. contrast pattern” set to a suitable value? If in the taught pattern the relevant contour lines are not shown completely: decrease “Min. contrast pattern”. If there are too many contour lines shown: increase “Min. contrast pattern”.
- “Min. contrast image” set to a suitable value for the current image? If the current image(s) do have a higher / lower contrast than the taught reference image /pattern please increase / decrease the value of “Min. contrast image” accordingly.
- If in the taught pattern the relevant contour lines are not shown completely: decrease “Min. contrast pattern”. If there are too many contour lines shown: increase “Min. contrast pattern”.
- If found at wrong position: use more distinct sample, re-teach if necessary.
- If the result value is fluctuating strongly from image to image: Take care that there are no “false” edges taught: This can be achieved by increasing “Min. contrast pattern” or by eliminating those false edges by function “Edit contour”.

Parameter Angle range: Rotational direction of angle

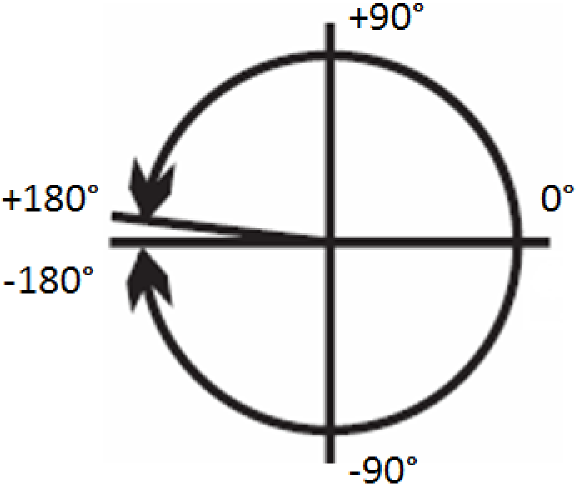
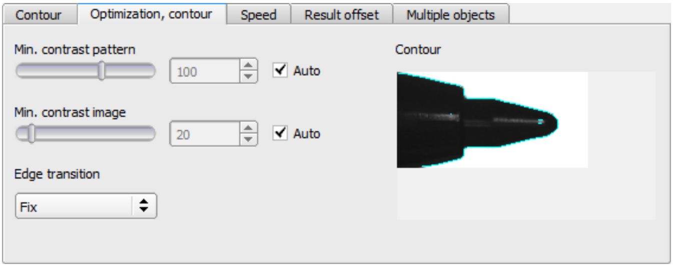


Fig. 129: Rotational direction of angle

4.6.3.4.3 Detector Contour, tab Optimization, contour

In the “Optimization, contour” tab further settings for the edge transition and the contrast can be made.



Min. contrast pattern 100 ☑ Auto

Min. contrast image 20 ☑ Auto

Edge transition Fix

Fig. 130: Detector Contour, tab Optimization, contour

The following parameters can be configured in the “Optimization, contour” tab:

Parameter	**Function**
Min. contrast pattern	Minimum contrast required with taught model for an edge to be detected as one.
Min. contrast image	Minimum contrast required in current image for an edge to be accepted as one.
Edge transition • Fix • Fix + inverted • Flexible	The parameter “edge transition” can be used to determine the transition between object/contour and background. Select whether the contour is to be detected only on the taught-in background (“Fix”), on the taught-in and inverted background (“Fix + inverted”) or on any background (“Flexible”). See also “Additional information:”
Auto	Automatic selection

Edge transition

Example:

A gray object is taught-in in front of a brighter background, as shown in the following figure.

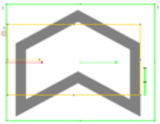
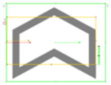
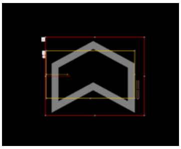
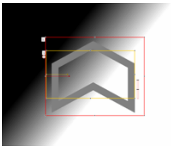
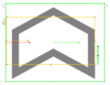
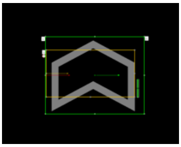
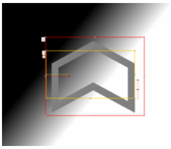
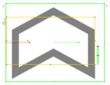
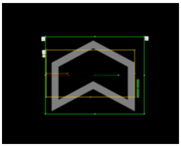
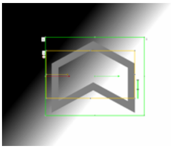


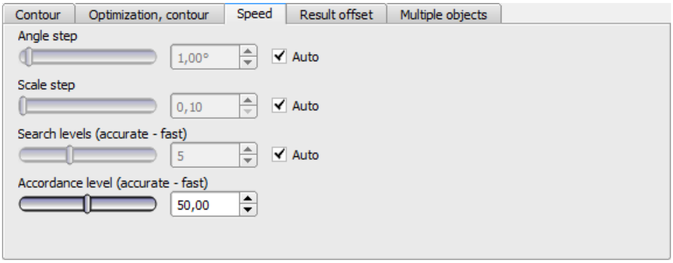
Fig. 131: Taught-in contour in front of a brighter background

The following table shows the results of the contour detector with the respective settings for the edge transition.

Settings for parameter “Edge transition”	Bright background	Dark background	Inconsistent background
Fix			
	Contour detector: OK	Contour detector: not OK	Contour detector: not OK
Fix + inverted			
	Contour detector: OK	Contour detector: OK	Contour detector: not OK
Flexible			
	Contour detector: OK	Contour detector: OK	Contour detector: OK

4.6.3.4.4 Detector Contour, tab Speed

With these adjustable parameters the execution speed can be influenced. The search is processed either less detailed, that means it is stopped earlier and is thus quicker, or it’s processed more detailed, that means search lasts longer and is thus slower.



Angle step: 1,00° ☑ Auto

Scale step: 0,10 ☑ Auto

Search levels (accurate - fast): 5 ☑ Auto

Accordance level (accurate - fast): 50,00

Fig. 132: Detector contour, tab speed

The following parameters can be configured in the “Speed” tab:

Parameter	Function
Angle step	Sensitivity of search throughout the selected angle range in degrees [°].
Scale step	Sensitivity of search throughout the selected scale range.
Search levels (accurate - fast)	Number of search levels. • High value: faster = riskier (candidates may be overlooked) • Small value: slower = less risky (all candidates)
Accordance level (accurate - fast)	Candidates that score less than indicated will already be rejected during the search. • High value: early rejection = quicker = riskier • Small value: late rejection = slower = less risky In case of false results this value can be decreased (more accurate).
Auto	Automatic selection

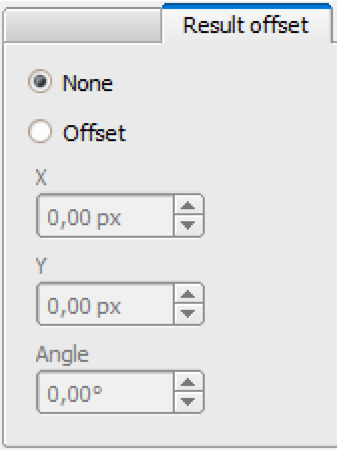
4.6.3.4.5 Result offset

With the Result offset, the final position of a found object can be modified. This can be useful when working with a robotic coordinate systems and needing to define a 'pick point' for example.

Settings in result offset tab:

Parameters	Functions
None	Automatically determines the of the Region Of Interest or ROI.
Offset	Free selectable position (graphically or by value input, e.g. for robot gripper use)

Parameters	Functions
	- X: Offset in X- direction (ref. ROI center) - Y: Offset in Y- direction (ref. ROI center) - Angle: angle offset (ref. ROI orientation)



Result offset

● None

○ Offset

X: 0,00 px

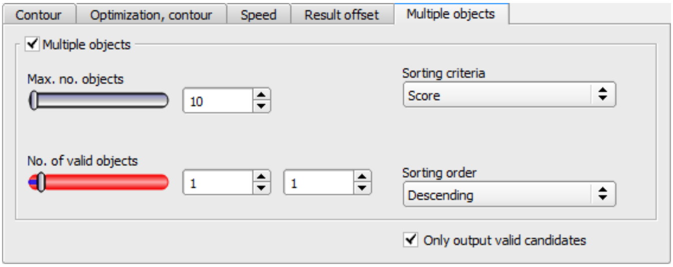
Y: 0,00 px

Angle: 0,00°

Fig. 133: Result offset

4.6.3.4.6 Detector Contour, tab Multiple objects

The “Multiple objects” function identifies objects whose contour matches the taught-in contour. Only as many objects as specified in the parameter “Max. no. objects” will be identified and output. The output of the object results is sorted according to the set criteria in ascending or descending order.



☑ Multiple objects

Max. no. objects: 10

No. of valid objects: 1 1

Sorting criteria: Score

Sorting order: Descending

☑ Only output valid candidates

Fig. 134: Detector Contour, tab Multiple objects

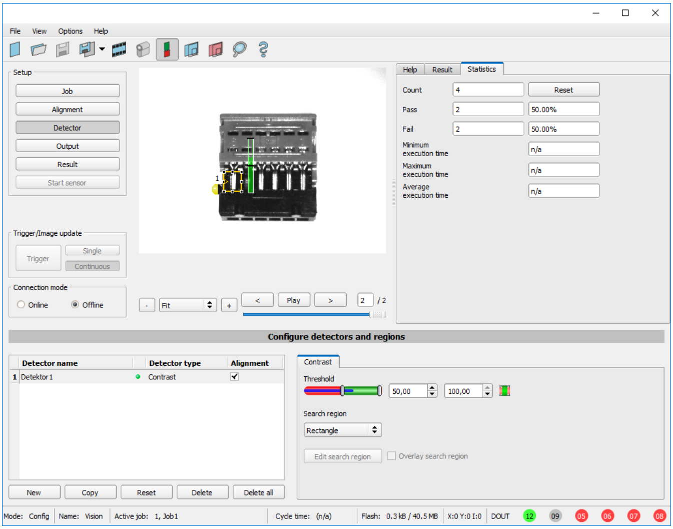
It is also possible to use this function to count objects in the image. The number of objects found can be output as a telegram. The minimum and maximum number of tolerated objects can be specified with the “No. objects” parameter. If the number of objects found is outside this range, the detector result is not OK.

The following parameters can be configured in the “Multiple objects” tab:

Parameter	**Function**
Max. no. objects	With the parameter “Max. no. objects”, the maximum number of target objects can be determined.
No. of valid objects	This parameter makes it possible to check whether the number of objects found is within a specified range. If it is within the range, the detector result is ok, otherwise not ok.
Sorting criteria	Sorting criteria according to which objects are “pre-sorted”.
• Score	Score
• Position X	X-Position
• Position Y	Y-Position
• Angle	Angle
• Scale	Scale
Sorting order	Sorting order for the selected sorting criteria.
• Ascending	The values of the sorting criteria are sorted in ascending order.
• Descending	The values of the sorting criteria are sorted in descending order.
Only output valid candidates	If this checkbox is activated, only objects whose score value is above the set threshold (tab “Contour”) are displayed and output. It can be used e.g. for parameter optimization.

4.6.3.5 Detector Contrast

This detector determines the contrast in the selected search area. Therefore all pixels inside the search area are evaluated with its gray value and the contrast value is calculated. If the contrast value is inside the limits set in parameter threshold the result is positive. The position of the single bright or dark pixels here is not relevant. The contrast is just depending on the bandwidth between darkest and brightest pixels and their quantity. Highest contrast value with 50% gray value “0” (= black) AND 50% gray value “255” (=white).



Count 4
Pass 2 50.00%
Fail 2 50.00%
Minimum execution time n/a
Maximum execution time n/a
Average execution time n/a

Connection mode: ○ Online ● Offline

Fit | 2 / 2

Detector name	Detector type	Alignment
1 Detektor 1	Contrast	☑

Contrast
Threshold 50,00 100,00
Search region: Rectangle
□ Overlay search region

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 0.3 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 135: Detector Contrast

Settings in tab Contrast:

Parameters	Functions
Threshold min/max	Range of contrast accepted.
Search region	Shape of search region can be set as Rectangle, Circle or Free shape. In mode Free shape “Edit search region” is active.
Edit search region	With Edit search region there can be masked out parts of the search area. The parts which are not relevant for this examination can be painted out like using an eraser. Masks can also be inverted, means that parts which are interesting can be marked. Also see chapter: Function: Mask
Overlay search region	On- / Off of display of edited search region.

For newly generated detectors, all parameters are present as standard values, suitable for many applications.

4.6.3.5.1 Tab Color channel

In the color channel tab, a color image (3 channel) can be converted to a gray value image (1 channel). In contrast to the gray value image of a monochrome VISOR® vision sensor, contrasts can be significantly increased. The highlighting of a color can be set individually for each detector. Thus, the flexibility compared to the use of optical color filters is significantly higher.

The image displayed is dependent on the selected detector.

- Color detectors: Display always colored.
- Object detectors: Monochrome image, display dependent upon selected color model and color channels.

The following parameters can be configured in the Color channel tab:

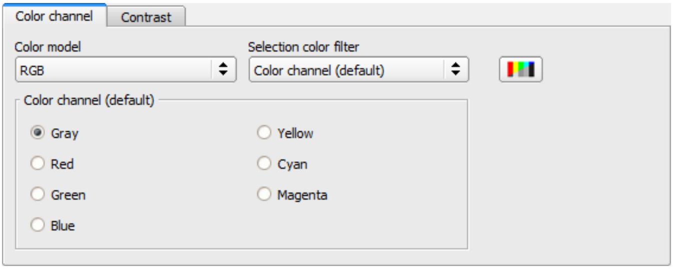
Parameter	**Function**
Color model	Color models: RGB, Color model RGB (Page 324) HSV, Color model HSV (Page 325) LAB, Color model LAB (Page 325)
Selection color filter	Depending on the color space, all or part of the following color filters are available: Color channel (default) Color distance Binarization
▥	Switching the image between color and monochrome.

4.6.3.5.1.1 Selection color filter

The following color filters are available:

Color channel (default)

The selected color channel is used as a gray value image.



Color channel | Contrast

Color model: RGB

Selection color filter: Color channel (default)

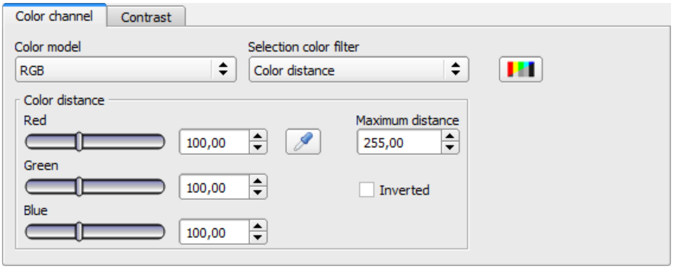
Color channel (default)

● Gray
○ Red
○ Green
○ Blue
○ Yellow
○ Cyan
○ Magenta

Fig. 136: Color filter, Color channel (default)

Color distance

A color is selected as reference color by specifying the color model values or by pipette. The gray value image indicates the distance of each pixel to this reference color. Typical application: Segmentation of characters for OCR.



Color channel | Contrast

Color model: RGB

Selection color filter: Color distance

Color distance

Red: 100,00

Green: 100,00

Blue: 100,00

Maximum distance: 255,00

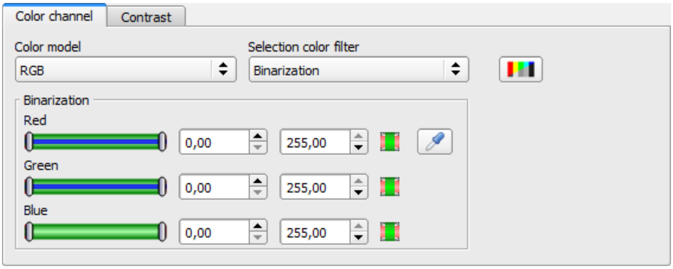
□ Inverted

Fig. 137: Color filter, Color distance

Parameter		Function
Red Green Blue	Lightness A B	Color channels: The color channel can be set via the slider or by entering a value (default 0).
Pipette button		With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.
Maximum distance		Distance of the current color versus the taught-in color. Colors that will exceed the maximum color distance will be black or white depending on the setting of “Inverted”.
Inverted		Inversion of the color distance image.

Binarization

A color range is selected. All pixels within this color range become white. Pixels with deviating color values become black.



Color model: RGB

Selection color filter: Binarization

Binarization

Red: 0,00 – 255,00

Green: 0,00 – 255,00

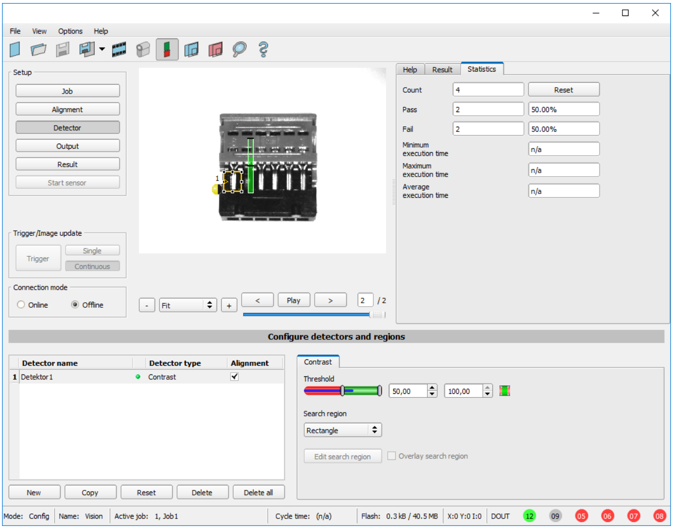
Blue: 0,00 – 255,00

Fig. 138: Color filter, Binarization

Parameter			**Function**
Red Green Blue	Hue Saturation Value	Lightness A B	Determination of the color range. The color ranges can be set via the slider or by entering a value.
Inverting button			The current setting is inverted when selecting the button.
Pipette button			With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.

4.6.3.5.2 Contrast application

In the example the presence of a metal contact is checked with a contrast detector.



Connection mode: ○ Online ● Offline

Fit | 2 / 2

Count	4	Reset
Pass	2	50.00%
Fail	2	50.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Detector name	Detector type	Alignment
1	Detektor1	● Contrast	✓

Contrast

Threshold 50,00 100,00

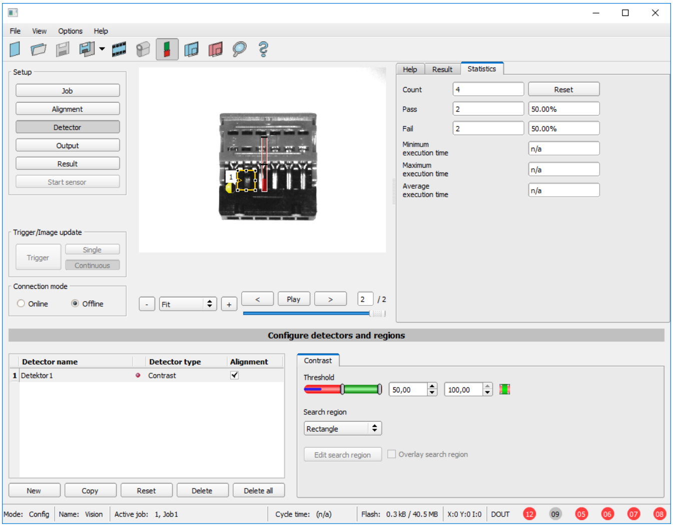
Search region: Rectangle

□ Overlay search region

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 0.3 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT

Fig. 139: Contrast, application example, positive result.

The presence of a shiny metal contact, in the middle of a surrounding black plastic housing, is checked with a contrast detector. As in this configuration contrast is pretty high, the contrast detector delivers a high score, and in combination with alignment the whole job works reliably.



Connection mode: ○ Online ● Offline

Fit | 2 / 2

Count	4	Reset
Pass	2	50.00%
Fail	2	50.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Detector name	Detector type	Alignment
1	Detektor1	● Contrast	✓

Contrast

Threshold 50,00 100,00

Search region: Rectangle

□ Overlay search region

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 0.3 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT

Fig. 140: Contrast, application example, negative result

If the same detector is placed now at a position where the metal contact is missing, it leads to a negative result. As, between the black surrounding and the now visible black background of the contact, the contrast value here is low.

Function detector Contrast

The dark and the bright pixels are evaluated according to the quantity and their intensity / brightness.

The position of the bright or dark pixels is not relevant.

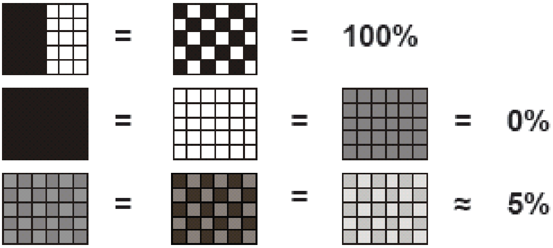
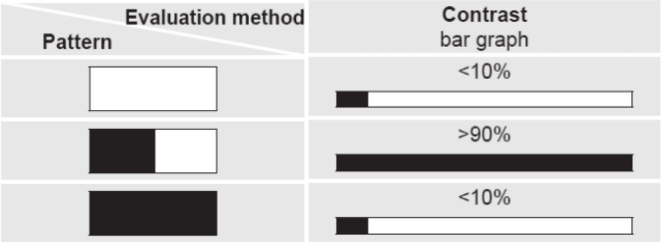


Fig. 141: Contrast examples



Evaluation method Pattern	Contrast bar graph
⬜	<10%
⬛⬜	>90%
⬛	<10%

Fig. 142: Contrast explanation

4.6.3.6 Detector Gray

With this detector in the first step is the range of accepted gray values is defined by setting the two limit sliders of parameter "Grey level".

In the second step the share of the search area (in %) which must be covered by pixels with the gray value inside the definition made in step 1, is defined with the parameter "Threshold" to achieve a positive result.

By the respective invert function all possible combinations can be defined, also those where the relevant gray values are only at the upper or lower border of the range of values. The position of the respective brought or dark pixels is not relevant.

With the parameter “Overlay” pixels can be marked in a certain color as an aid to select pixels / regions, which have a gray value inside (valid pixels), or outside (invalid pixels) the range set in “Grey level”. In this way pixels which are not covered with the settings / range of gray values can be detected very easily.

4.6.3.6.1 Settings in tab Gray:

Parameters	Functions
Gray level min/max	Range of gray values that are to be accepted
Threshold min/max	Percentage of the area, which must be in the selected gray value range
Search region	Shape of search region can be set as Rectangle, Circle or Free shape. In mode Free shape “Edit search region” is active.
Overlay	Selects which pixels are to be marked in color on the screen as an adjustment aid. “None” = no marking, or “Valid pixels” or “Invalid pixels” are marked in the image.
Edit search region	With Edit search region there can be masked out parts of the search area. The parts which are not relevant for this examination can be painted out like using an eraser. Masks can also be inverted, means that parts which are interesting can be marked. Also see chapter: Function: Mask
Overlay search region	On- / Off of display of edited search region.

For newly generated detectors, all parameters are preset as standard values, suitable for many applications.

4.6.3.6.2 Tab Color channel

In the color channel tab, a color image (3 channel) can be converted to a gray value image (1 channel). In contrast to the gray value image of a monochrome VISOR® vision sensor, contrasts can be significantly increased. The highlighting of a color can be set individually for each detector. Thus, the flexibility compared to the use of optical color filters is significantly higher.

The image displayed is dependent on the selected detector.

- Color detectors: Display always colored.
- Object detectors: Monochrome image, display dependent upon selected color model and color channels.

The following parameters can be configured in the Color channel tab:

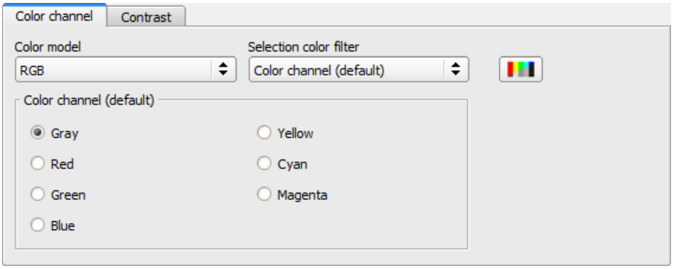
Parameter	Function
Color model	Color models: RGB, Color model RGB (Page 324) HSV, Color model HSV (Page 325) LAB, Color model LAB (Page 325)
Selection color filter	Depending on the color space, all or part of the following color filters are available: Color channel (default) Color distance Binarization
🟥🟨🟩🟦	Switching the image between color and monochrome.

4.6.3.6.2.1 Selection color filter

The following color filters are available:

Color channel (default)

The selected color channel is used as a gray value image.



Color model: RGB

Selection color filter: Color channel (default)

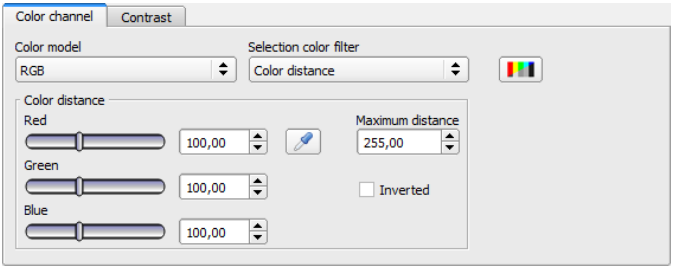
Color channel (default)

● Gray
○ Red
○ Green
○ Blue
○ Yellow
○ Cyan
○ Magenta

Fig. 143: Color filter, Color channel (default)

Color distance

A color is selected as reference color by specifying the color model values or by pipette. The gray value image indicates the distance of each pixel to this reference color. Typical application: Segmentation of characters for OCR.



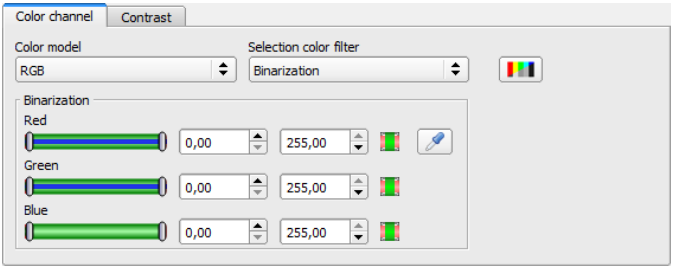
Color model: RGB
Selection color filter: Color distance
Color distance
Red: 100,00
Green: 100,00
Blue: 100,00
Maximum distance: 255,00
□ Inverted

Fig. 144: Color filter, Color distance

Parameter		Function
Red Green Blue	Lightness A B	Color channels: The color channel can be set via the slider or by entering a value (default 0).
Pipette button		With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.
Maximum distance		Distance of the current color versus the taught-in color. Colors that will exceed the maximum color distance will be black or white depending on the setting of "Inverted".
Inverted		Inversion of the color distance image.

Binarization

A color range is selected. All pixels within this color range become white. Pixels with deviating color values become black.

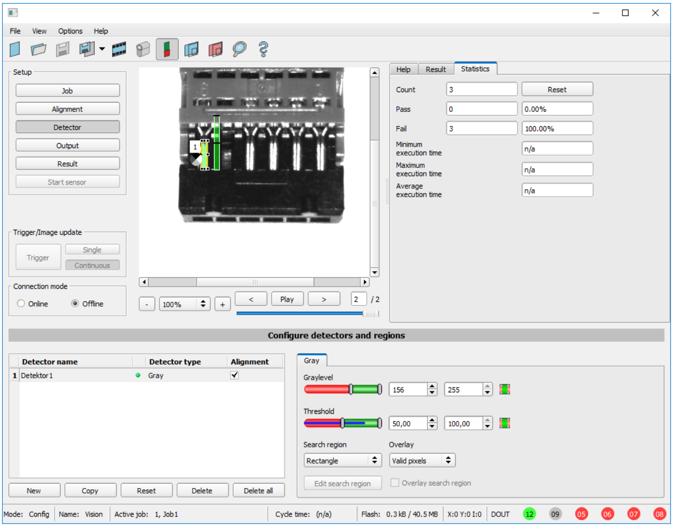


Color model: RGB
Selection color filter: Binarization
Binarization
Red: 0,00 255,00
Green: 0,00 255,00
Blue: 0,00 255,00

Fig. 145: Color filter, Binarization

Parameter			Function
Red Green Blue	Hue Saturation Value	Lightness A B	Determination of the color range. The color ranges can be set via the slider or by entering a value.
Inverting button			The current setting is inverted when selecting the button.
Pipette button			With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.

4.6.3.6.3 Gray level application



Connection mode: ○ Online ● Offline

100% 2 / 2

Count	3	
Pass	0	0.00%
Fail	3	100.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Detector name	Detector type	Alignment
1	Detektor 1	Gray	☑

Graylevel: 156 – 255

Threshold: 50,00 – 100,00

Search region: Rectangle

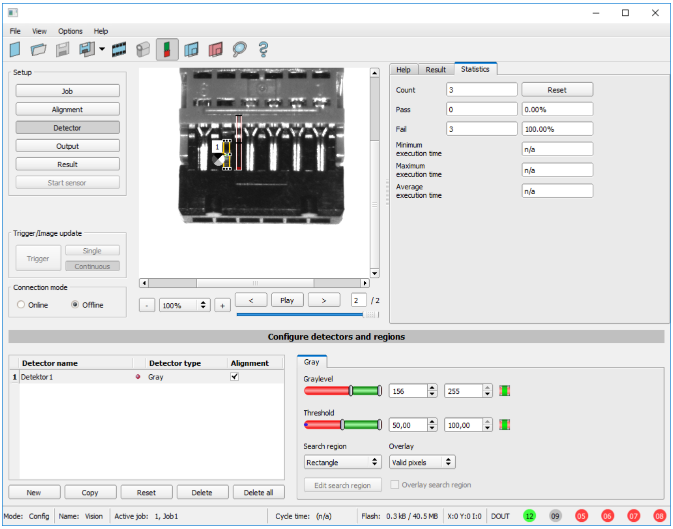
Overlay: Valid pixels

□ Overlay search region

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 0.3 kB / 40.5 MB | X:0 Y:0 I:0

Fig. 146: Gray level, application example, positive result.

The contact is present in search area. Shiny metal contact shows gray values > 192, that means inside the limits of threshold = result positive.



Connection mode: ○ Online ● Offline

100% | 2 / 2

Count	3	Reset
Pass	0	0.00%
Fail	3	100.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Detector name	Detector type	Alignment
1	Detektor1	Gray	✓

Graylevel: 156 | 255

Threshold: 50,00 | 100,00

Search region: Rectangle

Overlay: Valid pixels

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 0.3 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 147: Gray level, application example, negative result

Shiny metal contact is not present in the search area. That means average value of gray values in the search area is not inside the defined threshold limits. (Not inside gray value 192-255, but rather in range < 50). Result: negative = contact not found.

Aid to determine gray values:

By placing the cursor somewhere in the image the according X- and Y- coordinate and the gray value (“I” = Intensity) are displayed in the status line on the screen below in the next to last field at the right.

Function detector Gray level.

The authorised gray value range is defined by the two limits on the gray level slider.

All pixels within this gray value range and within the defined working zone (yellow frame) are added together. The proportion of the number of all the pixels in the working zone (yellow frame) and of the number of pixels in the authorised gray value range represents the result of this detector.

If this result is within the limits set on the switching threshold slider, the result is positive. The position of the gray value pixels on the screen is of no importance.

Example: (when the gray level slider is set to very dark values):

Both images produce exactly the same result with the gray level detector, as in each case 9 of the 25 pixels are detected as dark.

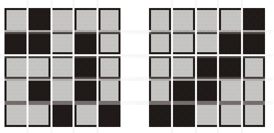


Fig. 148: Gray level, example 1

If the threshold value was set to 10 in this example, the following images would produce a positive result.

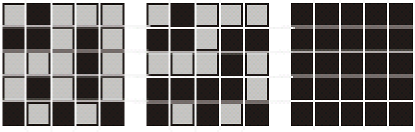


Fig. 149: Gray level, example 2

4.6.3.7 Detector Brightness

This detector determines the average value of the gray values in the search area. With the two limit sliders of the parameter “Threshold” the valid range of the brightness mean value is defined.

As soon as the calculated average value is within these two limits the result is positive. The result value is standardized to %. The position of the bright or dark pixels is not relevant. If there are position deviations from check to check the alignment function must be used.

Settings in tab Brightness:

Parameters	Functions
Brightness min/max	Range of gray values that are to be accepted
Search region	Shape of search region can be set as Rectangle, Circle or Free shape. In mode Free shape “Edit search region” is active.
Edit search region	With Edit search region there can be masked out parts of the search area. The parts which are not relevant for this examination can be painted out like using an eraser. Masks can also be inverted, meaning parts which are interesting can be marked. Also see chapter: Function: Mask
Overlay search region	On- / Off of display of edited search region.

For newly generated detectors, all parameters are preset as standard values, suitable for many applications.

4.6.3.7.1 Tab Color channel

In the color channel tab, a color image (3 channel) can be converted to a gray value image (1 channel). In contrast to the gray value image of a monochrome VISOR® vision sensor, contrasts can be significantly increased. The highlighting of a color can be set individually for each detector. Thus, the flexibility compared to the use of optical color filters is significantly higher.

The image displayed is dependent on the selected detector.

- Color detectors: Display always colored.
- Object detectors: Monochrome image, display dependent upon selected color model and color channels.

The following parameters can be configured in the Color channel tab:

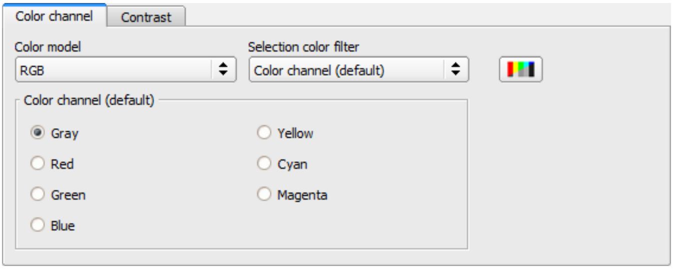
Parameter	**Function**
Color model	Color models: RGB, Color model RGB (Page 324) HSV, Color model HSV (Page 325) LAB, Color model LAB (Page 325)
Selection color filter	Depending on the color space, all or part of the following color filters are available: Color channel (default) Color distance Binarization
▨	Switching the image between color and monochrome.

4.6.3.7.1.1 Selection color filter

The following color filters are available:

Color channel (default)

The selected color channel is used as a gray value image.



Color model: RGB

Selection color filter: Color channel (default)

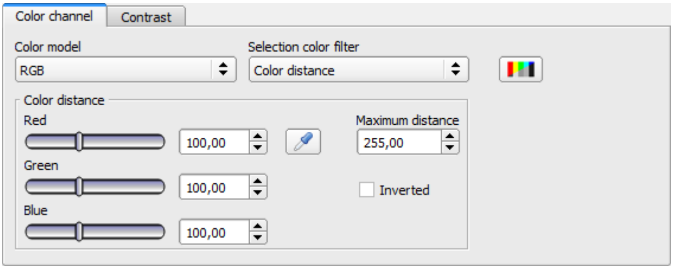
Color channel (default)

● Gray
○ Red
○ Green
○ Blue
○ Yellow
○ Cyan
○ Magenta

Fig. 150: Color filter, Color channel (default)

Color distance

A color is selected as reference color by specifying the color model values or by pipette. The gray value image indicates the distance of each pixel to this reference color. Typical application: Segmentation of characters for OCR.



Color model: RGB

Selection color filter: Color distance

Color distance

Red: 100,00

Green: 100,00

Blue: 100,00

Maximum distance: 255,00

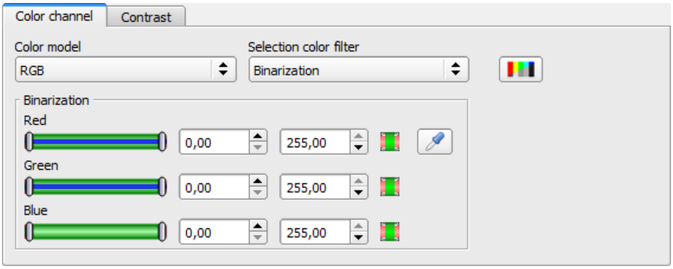
□ Inverted

Fig. 151: Color filter, Color distance

Parameter		Function
Red Green Blue	Lightness A B	Color channels: The color channel can be set via the slider or by entering a value (default 0).
Pipette button		With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.
Maximum distance		Distance of the current color versus the taught-in color. Colors that will exceed the maximum color distance will be black or white depending on the setting of “Inverted”.
Inverted		Inversion of the color distance image.

Binarization

A color range is selected. All pixels within this color range become white. Pixels with deviating color values become black.



Color model: RGB

Selection color filter: Binarization

Binarization

Red: 0,00 – 255,00

Green: 0,00 – 255,00

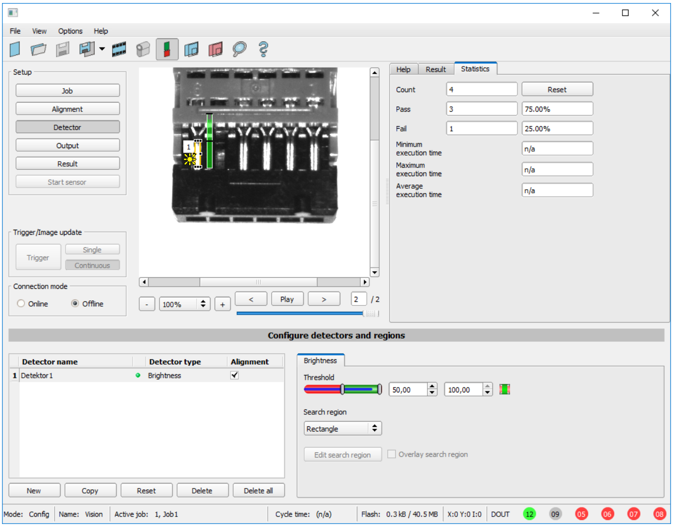
Blue: 0,00 – 255,00

Fig. 152: Color filter, Binarization

Parameter			**Function**
Red Green Blue	Hue Saturation Value	Lightness A B	Determination of the color range. The color ranges can be set via the slider or by entering a value.
Inverting button			The current setting is inverted when selecting the button.
Pipette button			With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.

4.6.3.7.2 Brightness application

The detector Brightness calculates the average value of the gray values of all pixels within the search area.



○ Online ● Offline

100% 2 /2

Count	4
Pass	3	75.00%
Fail	1	25.00%
Minimum execution time	n/a
Maximum execution time	n/a
Average execution time	n/a

Configure detectors and regions

	Detector name	Detector type	Alignment
1	Detektor 1	● Brightness	☑

Brightness

Threshold 50,00 100,00

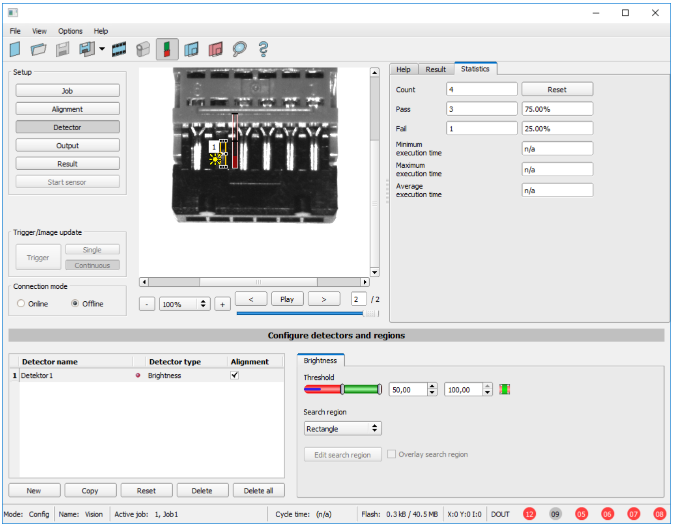
Search region: Rectangle

□ Overlay search region

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 0.3 kB / 40.5 MB | X:0 Y:0 I:0

Fig. 153: Brightness, application example, positive result.

The contact is present within the position searched for; therefore the average value of the gray values in the search area has a high score (near 100%). This means the current value is within the requested threshold limits and the result is positive = contact present.



○ Online ● Offline

100% 2 /2

Count	4
Pass	3	75.00%
Fail	1	25.00%
Minimum execution time	n/a
Maximum execution time	n/a
Average execution time	n/a

Configure detectors and regions

	Detector name	Detector type	Alignment
1	Detektor 1	● Brightness	☑

Brightness

Threshold 50,00 100,00

Search region: Rectangle

□ Overlay search region

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 0.3 kB / 40.5 MB | X:0 Y:0 I:0

Fig. 154: Brightness, application example, negative result.

The contact is not present within the position searched for; therefore the average value of the gray values in the search area delivers a low score (near 0%). This means the current value is not within the requested threshold limits and the result is negative = contact not present.

Examples: Brightness value as average value of the gray values.

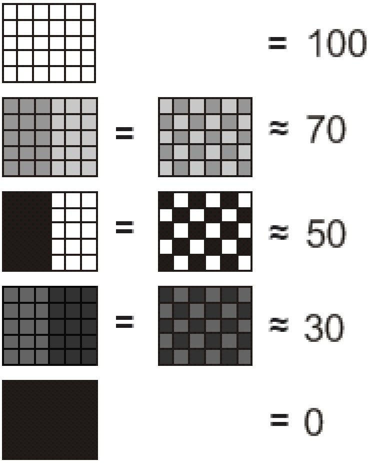


Fig. 155: Brightness, examples

4.6.3.8 Detector BLOB, Introduction

The BLOB detector is used to identify and count one or more objects with some common features like same gray value range, same area, same circumference, ...

Tab Color channel (Page 211)

Detector BLOB, tab Binarization, Absolute threshold (Page 193)

Detector BLOB, tab binarization, dynamic threshold (Page 194)

Detector BLOB, tab “Features” (Page 199)

Detector BLOB, tab sorting (Page 210)

BLOB, Introduction

- “BLOB” abbreviation for “Binary Large Object” or “Binary Labeled Object”.
- Basic function of machine vision for evaluation of connected areas / objects in an image.
- The single objects are distinguished by simple features like: area, width, height.

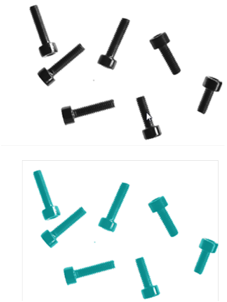


Fig. 156: Screws 1. Binarization, 2. detected as BLOB / object

Typical applications

- Count objects
- Differentiation / classification of objects in the image by:
- Size, area, contour
- Form, geometry
- Position, orientation
- Face up/ down
- Surface inspection

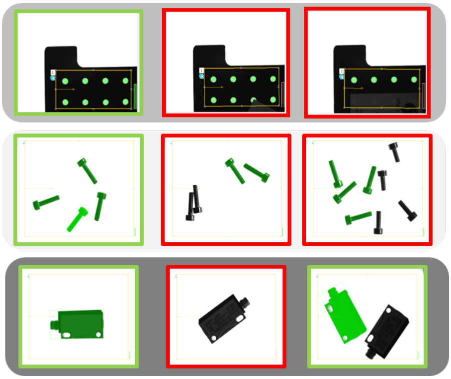
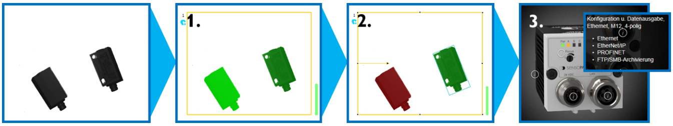


Fig. 157: Typical applications: count, classify / sort, orientation / face up / down

BLOB, simple configuration in 3 steps



1. Binarization

Distinguish between relevant objects and background.

Detector BLOB, tab Binarization, Absolute threshold (Page 193)

Detector BLOB, tab binarization, dynamic threshold (Page 194)

2. Filtering of detected BLOBs

Filtering by different features like: area, circumference, orientation, position, ...

Detector BLOB, tab “Features” (Page 199)

3. Data output

Definition of data output telegram and sorting of results.

Detector BLOB, tab sorting (Page 210)

Telegram, Data output (Page 296)

4.6.3.8.1 Tab Color channel

In the color channel tab, a color image (3 channel) can be converted to a gray value image (1 channel). In contrast to the gray value image of a monochrome VISOR® vision sensor, contrasts can be significantly increased. The highlighting of a color can be set individually for each detector. Thus, the flexibility compared to the use of optical color filters is significantly higher.

The image displayed is dependent on the selected detector.

- Color detectors: Display always colored.
- Object detectors: Monochrome image, display dependent upon selected color model and color channels.

The following parameters can be configured in the Color channel tab:

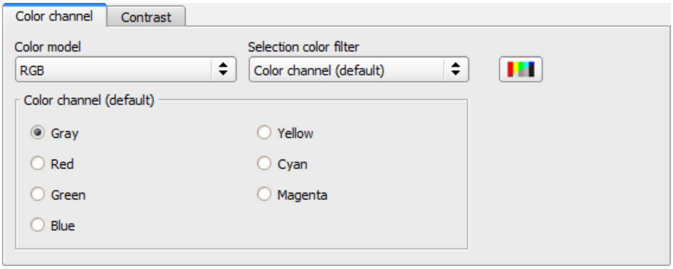
Parameter	Function
Color model	Color models: RGB, Color model RGB (Page 324) HSV, Color model HSV (Page 325) LAB, Color model LAB (Page 325)
Selection color filter	Depending on the color space, all or part of the following color filters are available: Color channel (default) Color distance Binarization
🟥🟨🟩⬛	Switching the image between color and monochrome.

4.6.3.8.1.1 Selection color filter

The following color filters are available:

Color channel (default)

The selected color channel is used as a gray value image.



Color model: RGB

Selection color filter: Color channel (default)

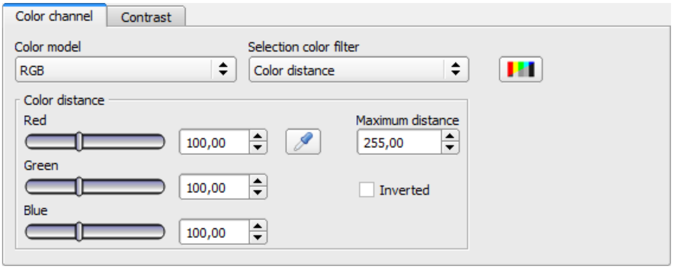
Color channel (default)

● Gray
○ Red
○ Green
○ Blue
○ Yellow
○ Cyan
○ Magenta

Fig. 158: Color filter, Color channel (default)

Color distance

A color is selected as reference color by specifying the color model values or by pipette. The gray value image indicates the distance of each pixel to this reference color. Typical application: Segmentation of characters for OCR.



Color model: RGB

Selection color filter: Color distance

Color distance

Red: 100,00

Green: 100,00

Blue: 100,00

Maximum distance: 255,00

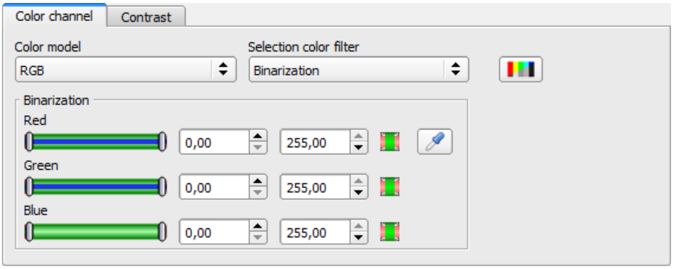
□ Inverted

Fig. 159: Color filter, Color distance

Parameter		Function
Red Green Blue	Lightness A B	Color channels: The color channel can be set via the slider or by entering a value (default 0).
Pipette button		With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.
Maximum distance		Distance of the current color versus the taught-in color. Colors that will exceed the maximum color distance will be black or white depending on the setting of "Inverted".
Inverted		Inversion of the color distance image.

Binarization

A color range is selected. All pixels within this color range become white. Pixels with deviating color values become black.



Color model: RGB

Selection color filter: Binarization

Binarization

Red: 0,00 – 255,00

Green: 0,00 – 255,00

Blue: 0,00 – 255,00

Fig. 160: Color filter, Binarization

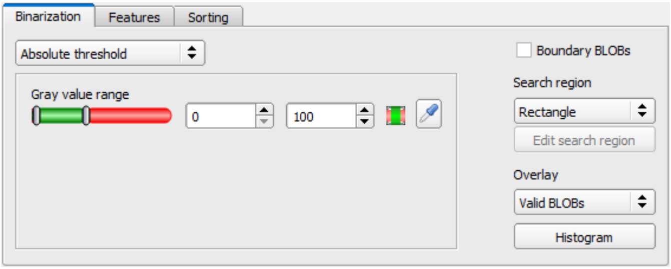
Parameter			Function
Red Green Blue	Hue Saturation Value	Lightness A B	Determination of the color range. The color ranges can be set via the slider or by entering a value.
Inverting button			The current setting is inverted when selecting the button.
Pipette button			With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.

4.6.3.8.2 Detector BLOB, tab Binarization, Absolute threshold

In this tab all parameters for binarization of a BLOB can be set.

Binarization is the first step of BLOB processing. It is used to distinguish relevant objects from the background of the image, by converting the gray image into a pure black and white / binary image.

Binarization can be done by two different binarization methods.



Absolute threshold

Gray value range 0 100

□ Boundary BLOBs

Search region: Rectangle

Overlay: Valid BLOBs

Fig. 161: Detector BLOB, tab Binarization

In the first combobox the threshold method for binarization is selected.

Parameter	Function
Absolute threshold	Binarization threshold is set to an absolute gray value in range of 0 .. 255.
Dynamic threshold	Threshold is automatically set to an statistically optimised position in order to distinguish between fore- and background. Detector BLOB, tab binarization, dynamic threshold (Page 194)

Parameters with selection “Absolute Threshold”

Parameter	Function
Absolute threshold	Setting the upper and lower limit defines the range of valid gray values of pixels belonging to the BLOB.
Gray value range	Adjusting the upper and lower limit of gray values for binarization / valid for the BLOB.
Invert button	With the "Invert button" (default: red/green/red) the logic of detection can be inverted. This way the relevant range can be included or excluded.
Pipette button	With a click to the "Pipette button" the cursor changes into a pipette symbol. By moving the cursor and clicking to any position (pixel) inside the image the gray value of this pixel is taken and the limits of "Absolute threshold" are set to +/- 10 gray values of the gray value of this pixel (values clipped at 0 or 255).

Boundary BLOB´s, Overlay and Histogram

Parameter	Function
Boundary BLOBs	Selected BLOBs (objects) are considered, if they are fulfilling the BLOB- / filter- criteria, even if they are not completely positioned within the yellow search region. Please note: BLOBs are also considered as Boundary BLOBs if they are touching a zone masked with the "Edit search region / Function: Mask" (even masked zones inside the image / search region). Detector BLOB, Boundary BLOBs (Page 195)
Search region	Search region can be set to: "Rectangle", "Circle" or "Free shape". In mode "Free shape" the function "Edit search region" is active.
Edit search region	Using the "Edit search region" button the dialog window to edit the search region can be opened. Function: Mask (Page 156)
Overlay BLOBs	"Valid BLOBs": all valid BLOBs which fulfill the filter criteria are marked in green. With this selection invalid BLOBs are marked in red. "BLOB contour": all valid BLOBs (all BLOBs fulfilling the filter criteria) are marked with a green contour line. With this selection invalid BLOBs are not marked.
Histogram	The Histogram button opens the Histogram window for the BLOB. Detector BLOB, tab Binarization, Histogram (Page 198)

4.6.3.8.3 Detector BLOB, tab binarization, dynamic threshold

In this tab all parameters for binarization with dynamic threshold can be set. The dynamic threshold can be used if BLOBs / objects and background do have clearly different gray value ranges, and illumination conditions are changing uniformly over the whole image.

If the brightness of the image changes uniformly, with the dynamic threshold limits are readjusted automatically. (With absolute threshold those limits must be readjusted manually.)

Please note:

- The Dynamic threshold is newly calculated with each new image / evaluation.
- Please consider that fluctuating illumination, surface- reflectivity, etc. may influence the result!

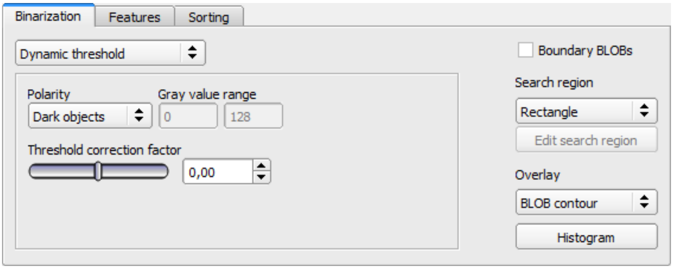


Fig. 162: Detector BLOB, tab Binarization, Dynamic threshold

Parameters with selection "Dynamic Threshold"

Parameter	Function
Dynamic threshold	Dynamic threshold is automatically set to an statistically optimised position in order to distinguish between fore- and background.
Polarity	Definition if BLOB is brighter or darker compared to the background.
Gray value range	Adjusted limits of gray values for binarization.
Threshold correction factor	With the Threshold correction factor the above automatically calculated binarization threshold can be moved / manipulated either towards the foreground- or background- brightness.

For illustration of the dynamic threshold see also: Detector BLOB, tab Binarization, Histogram (Page 198).

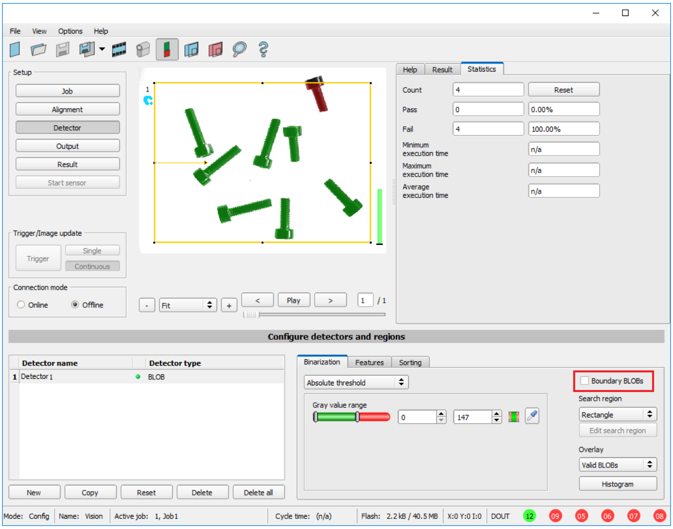
4.6.3.8.3.1 Detector BLOB, Boundary BLOBs

If the checkbox "Boundary BLOBs" is active, the selected BLOBs (objects) are considered, even if they are not completely positioned within the yellow search region. (Of course they have to fulfill the BLOB- / filter- criteria anyway).

Please note:

- BLOBs are also considered as Boundary BLOBs if they are touching a zone masked with the “Edit search region / Function: Mask” (even if masked zones are inside the image / search region).

Example 1: Boundary BLOBs, touching outer search region.



Connection mode: ○ Online ● Offline

Count	4	
Pass	0	0.00%
Fail	4	100.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Detector name	Detector type
1	Detector 1	BLOB

Absolute threshold

Gray value range: 0 – 147

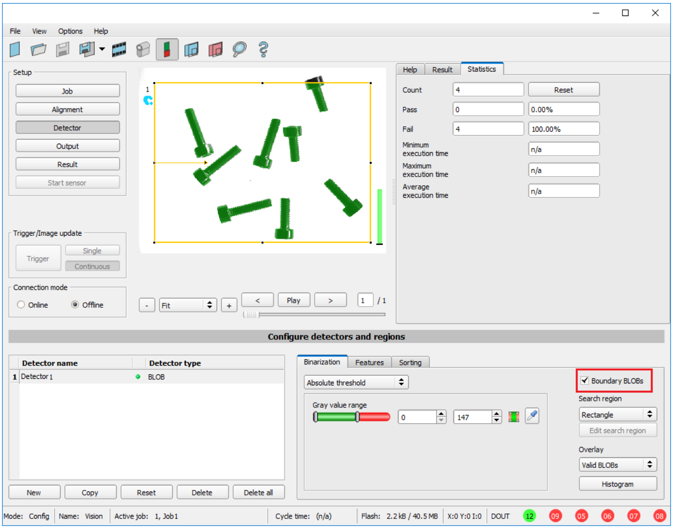
□ Boundary BLOBs

Search region: Rectangle

Overlay: Valid BLOBs

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 2.2 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT

Fig. 163: Boundary BLOB example 1/1: BLOB is touching outer yellow search region, it is not considered as valid BLOB as setting “Boundary BLOBs” is NOT active.



Count	4
Pass	0 / 0.00%
Fail	4 / 100.00%
Minimum execution time	n/a
Maximum execution time	n/a
Average execution time	n/a

	Detector name	Detector type
1	Detector 1	BLOB

Absolute threshold

Gray value range: 0 – 147

☑ Boundary BLOBs

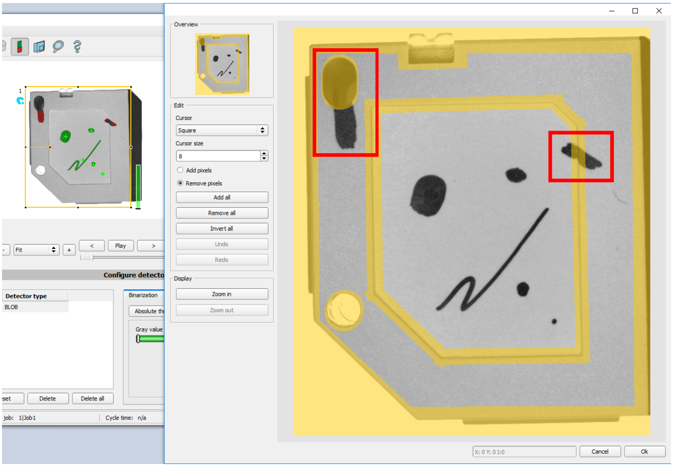
Search region: Rectangle

Overlay: Valid BLOBs

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 2.2 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 164: Boundary BLOB example 1/2: BLOB is touching outer yellow search region, but it is considered as valid BLOB, as setting “Boundary BLOBs” is ACTIVE now!

Example 2, Boundary BLOBs, touching inner “Mask” region.



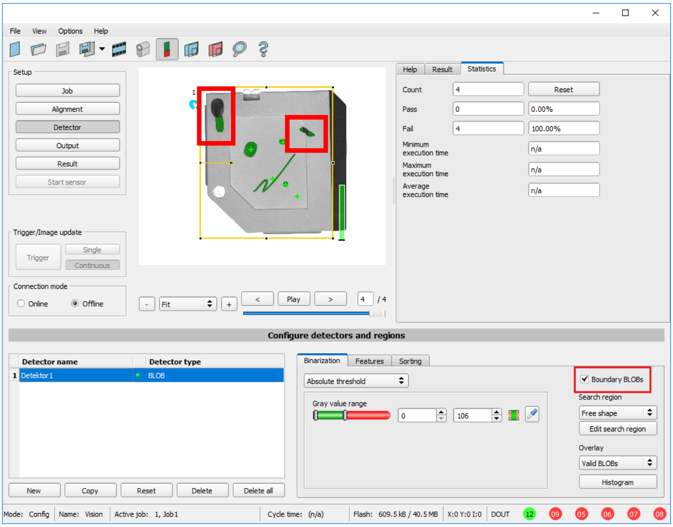
Cursor: Square

Cursor size: 8

○ Add pixels

● Remove pixels

Fig. 165: Boundary BLOB example 2/1: BLOBs are touching inner yellow “Mask” regions, they are not considered as valid BLOBs, as setting “Boundary BLOBs” is NOT active.



Count 4
Pass 0 0.00%
Fail 4 100.00%
Minimum execution time n/a
Maximum execution time n/a
Average execution time n/a

4 / 4

Configure detectors and regions

Detector name	Detector type
1 Detektor1	BLOB

Absolute threshold

Gray value range 0 106

☑ Boundary BLOBs

Search region Free shape

Overlay Valid BLOBs

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 609.5 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 166: Boundary BLOB example 2/2: BLOBs are touching inner yellow “Mask” regions, but they are considered as valid BLOBs yet, as setting “Boundary BLOBs” is ACTIVE now!

4.6.3.8.3.2 Detector BLOB, tab Binarization, Histogram

In this window the Histogram of the gray values inside the yellow ROI, and the chosen thresholds are shown. In the here shown example there are clear maxima for fore- and background. The binarization threshold is adjusted to approx. the center in between.

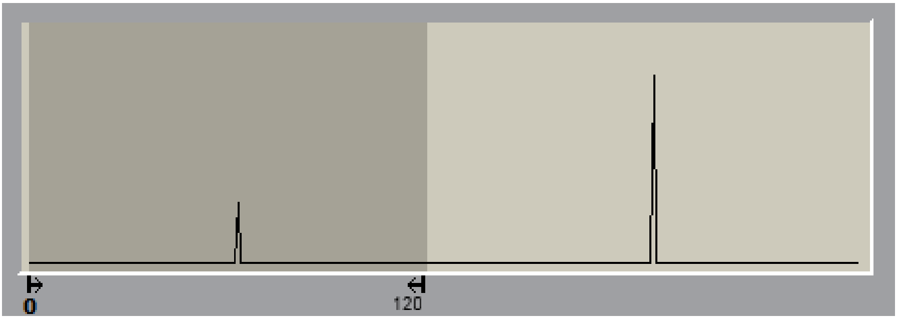
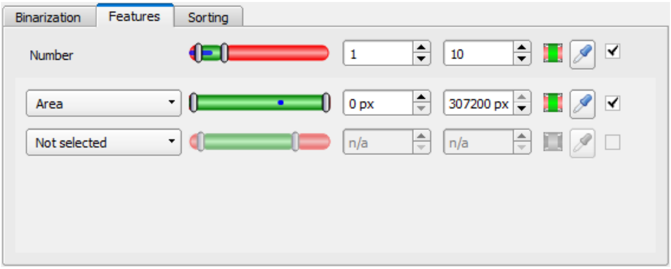


Fig. 167: Detector BLOB, tab Binarization, Histogram

4.6.3.8.4 Detector BLOB, tab “Features”

In this tab, the features / filter criteria to distinguish between valid and invalid BLOBs/objects can be defined. Only the valid BLOBs are processed further, e.g. for data output.

Example: If the feature “Area” is set to a range of 100 ... 150 (pixel), only BLOBs with an area within this range are considered as valid (green).



Number 1 10 ☑
Area 0 px 307200 px ☑
Not selected n/a n/a □

Fig. 168: Detector BLOB, tab Features

The following parameters can be configured in the “Features” tab:

Parameter	Function
Number	Additionally to the features used for filtering the BLOBs, the number of existing and valid BLOBs can be checked. For this purpose the lower and upper limit of accepted BLOBs (max. 10.000) is determined. • Detector result positive: Number of valid (filtered) BLOBs is within the range of “Number”. • Detector result negative: Number of valid (filtered) BLOBs is outside the range of “Number”. If the number of BLOBs is outside the defined limits, the detector result is negative, although valid BLOBs are marked in green. If the detector counts more than 10.000 BLOBs (maximum), the detector result is negative and no further calculations are performed. Please note: Defect detection via number = 0.
Invert button	With the “Invert button” (default: red/green/red) the logic of detection can be inverted. This way the relevant range can be included or excluded.
Pipette- button (Number)	By clicking this symbol, the lower and upper limit of “Number” are set to exactly the found number of BLOBs in the

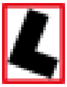
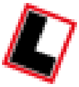
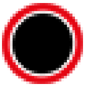
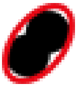
Parameter	Function
	image.
Pipette- button (Feature)	By clicking on the “Pipette button”, the cursor changes into a pipette symbol. By moving the cursor and clicking to any position (pixel) inside a valid (green) BLOB, the limits of the selected feature are adjusted automatically to +/-10% of the value of the selected BLOB. Example: With selected feature “Area” and clicking with Pipette active on any pixel inside the BLOB , the lower and upper limit of area is set to +/-10% of the found number of pixels of the selected BLOB.
Checkbox (Default: Active)	• Active: Feature is calculated, filtered (limits adjustable), and available for data output. • Inactive: Feature is calculated, but NOT filtered, but is available for data output.

List of features / first level: BLOB type / Geometric Model

The features of the first level (Area, Area incl. holes, Contour length, ...) are calculated directly from the BLOB data, i.e. the pixels belonging to the BLOB. For further features, a geometric model is first fitted to the data via a best-fit line. Then the features are based on this model and not directly on the pixels belonging to the BLOB.

Feature	Function
Area	Area of the BLOB, without holes, in pixels. Corresponds to the number of pixels belonging to the BLOB.
Area (incl. holes)	Area of the BLOB, including holes, in pixels. Corresponds to the number of pixels within the outer contour.
Contour length	Number of pixels of outer contour of the BLOB.
Compactness	Compactness of BLOB (Circle = 1, all other >1) The stronger the shape of the BLOB deviates from an ideal circle the larger the value of compactness will be. Range of slider: 1 -100 (clipped at 100, BLOBs with higher values are marked as invalid)
Center of gravity X	X- coordinate of center of gravity of BLOB If in the setup “Job” the calibration is activated, the

Feature	Function
	value can also be output in world coordinates, e.g. millimeters.
Center of gravity Y	Y- coordinate of center of gravity of BLOB If in the setup “Job” the calibration is activated, the value can also be output in world coordinates, e.g. millimeters.

BLOB type / Geometric Model	Function
Some features are calculated based on a given geometric model, e.g. eccentricity is based on an ellipse fit to the object	
 Rectangle, paraxial (R1)	Enclosing rectangle parallel to Y- axis and X- axis. Outliers are not eliminated.
 Rectangle, min. area (R2)	Enclosing rectangle with smallest area. Outliers are not eliminated.
 Circle, fit (C1)	Circle-fit, not enclosing, outlier correction (robust against outliers)
 Ellipse, equivalent (E1)	Equivalent ellipse, based on moments of area.

List of features / second level: BLOB type parameter

Feature	Relevant for	Function	Possibility of value output in world coordinates [mm] when calibration is activated
Center X	R1, R2, C1, E1	X- coordinate of the center of the fitted, geometric element (rectangle, ellipse)	✓

Feature	**Relevant for**	**Function**	**Possibility of value output in world coordinates [mm] when calibration is activated**
Center Y	R1, R2, C1, E1	Y- coordinate of the center of the fitted, geometric element (rectangle, ellipse)	✓
Width	R1, R2, E1	Width of geometric element. Width ≥ 0, width ≥ height. The orientation is choosen in a way that width is always bigger than height. (Exception: R1, Rectangle, paraxial: Width always in horizontal orientation = parallel to X-axis)	✓
Height	R1, R2, E1	Height of geometric element. Height ≥ 0, height ≤ width. The orientation is choosen in a way that width is always bigger than height. (Exception: R1, Rectangle, paraxial: Height always in vertical orientation = parallel to Y-axis)	✓
Angle (180)	R2, E1	Orientation of width (long axis) of object in degrees (range: -90 ... +90°, 0° = east, counterclockwise). See also: Feature Angle (Page 204)	

Feature	Relevant for	Function	Possibility of value output in world coordinates [mm] when calibration is activated
Angle (360)	R2, E1	Orientation of width of object in degrees (range: -180 ... +180°, 0° = east, counterclockwise). See also: Feature Angle (Page 204)	
Axial ratio	E1	Ratio long / short axis (a/b)	
Face up/down, area	E1	Face up/down discrimination, based on area, indicated by sign. See also: Face up / Face down, area or contour (Page 210)	
Radius	C1	Specifies the radius of the fitted circle.	✓
Deviation, in	C1	Indicates the largest deviation between the BLOB contour and the contour of the geometric element (deviation into the fitted circle). See also: Feature Deviation (Page 206)	✓
Deviation, out	C1	Indicates the largest deviation between the BLOB contour and the contour of the geometric element (deviation out of the fitted circle). See also: Feature Deviation (Page 206)	✓
Deviation, mean	C1	Indicates the mean of the	✓

Feature	Relevant for	Function	Possibility of value output in world coordinates [mm] when calibration is activated
		absolute "in" and "out" deviation values between the BLOB contour and the contour of the geometric element. See also: Feature Deviation (Page 206)	

Feature Angle

With the feature "Angle (180)" and "Angle (360)", the orientation of the object can be determined. The angle always indicates the orientation of the width axis (width is the longest side of an object). The angles are specified in [degrees °].

The "Angle (180)" feature has a rotational range of -90° to +90°, as shown in the following figure.

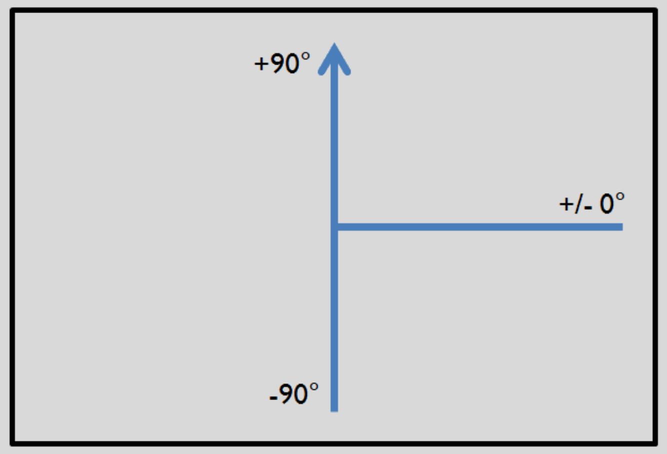


Fig. 169: Rotational direction of "Angle 180"

The "Angle (360)" feature depends on the selected geometric model (e.g. E1 Ellipse, R2 Rectangle min. area, etc). It has a rotational range of -180° to +180°, as shown in the following figure.

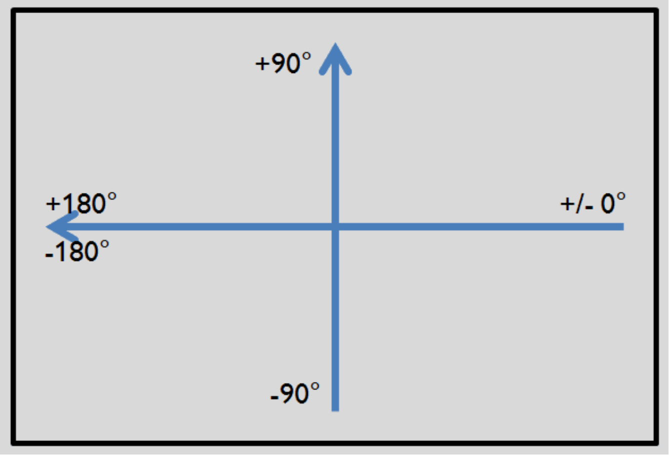


Fig. 170: Rotational direction of “Angle 360”

In contrast to the “Angle (180)” feature, in the “Angle (360)” feature the orientation of the width axis is set depending on a vector. This vector indicates the direction to the contour point with the longest distance to the center and the orientation (180°) of the vector point to the same side. Whether an object lies in half plane (-90° ... +90°) or in half plane (-180° ...-90°; 90°...180°) is determined by the half plane in which the vector lies. The following figures show two examples of the angle determination of the feature “Angle (360)”.

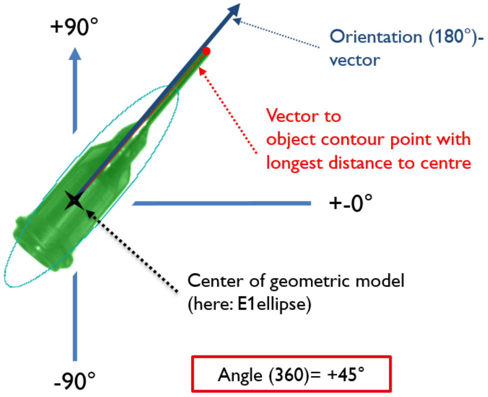

+90°
Orientation (180°)-vector
Vector to object contour point with longest distance to centre
+-0°
Center of geometric model (here: E1ellipse)
-90°
Angle (360)= +45°

Fig. 171: Example 1: Angle (360) = +45°

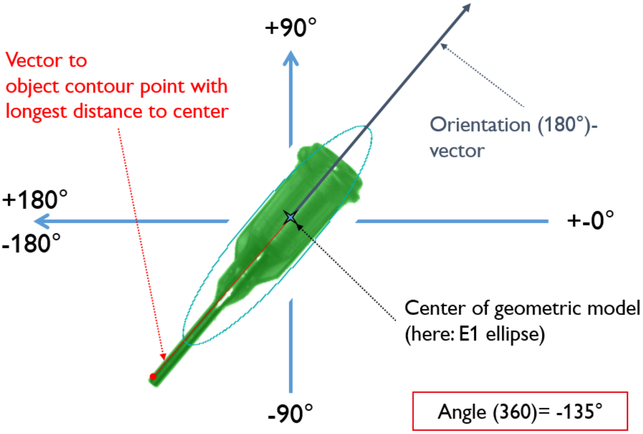


Fig. 172: Example 2: Angle (360) = -135°

Feature Deviation

The deviation feature calculates measures that quantitatively describe the deviation of the actual object from the fitted model. The features "Deviation, in", "Deviation, out" and "Deviation, mean" assess indentations and protruding elements of the BLOB/object contour. The deviation values always refer to the fitted circle. All indentations into the fitted circle are "Deviation, in". All elements which protrude out of the fitted circle are determined by the feature "Deviation, out". The orientation directions of the features are shown in the following figure.

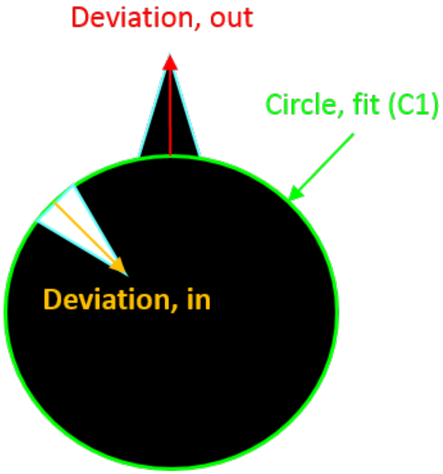


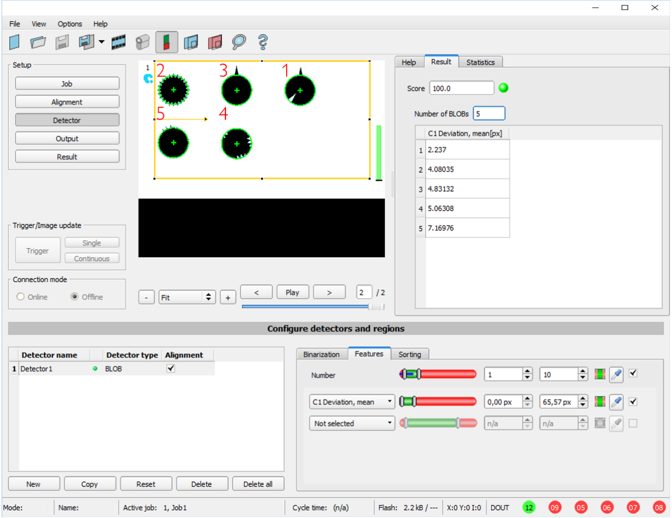
Fig. 173: Orientation direction “Deviation, in” and “Deviation, out”

In the “Result” tab of the VISOR® vision sensor software, the value of the largest “Deviation, in” and “Deviation, out” is displayed for each fitted circle (if active).

The “Deviation, mean” feature indicates the mean of the absolute deviation values to all positions, i.e. to all pixels, of the fitted circle.

Example: Deviation, mean

Jagged elements are checked by the feature “Deviation, mean”, see figure “Deviation, mean”.



Score 100.0

Number of BLOBs 5

	C1 Deviation, mean[px]
1	2.237
2	4.08035
3	4.83132
4	5.06308
5	7.16976

	Detector name	Detector type	Alignment
1	Detector1	BLOB	☑

Number	1	10	☑
C1 Deviation, mean	0,00 px	65,57 px	☑
Not selected	n/a	n/a	□

Mode: | Name: | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 2.2 kB / --- | X:0 Y:0 I:0 | DOUT

Fig. 174: Deviation, mean

The “Deviation, mean” feature calculates all deviations from the circle-fit (green) to the contour of the object/BLOB (cyan) per pixel of the fitted circle inwards and outwards. The following figure shows a zoomed-out section of the circle number “2” from the previous figure. The red arrows indicate the deviations per pixel of the fitted circle to the BLOB contour. The amounts of all determined values are averaged and form the result of the “Deviation, mean” feature.

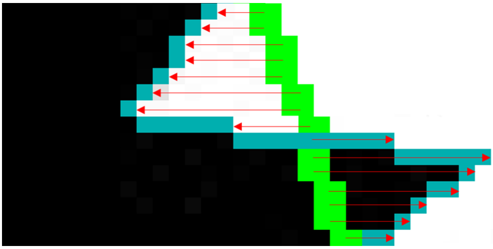


Fig. 175: Deviation, mean: Circle 2 zoomed

Example: Deviation, in and Deviation, out

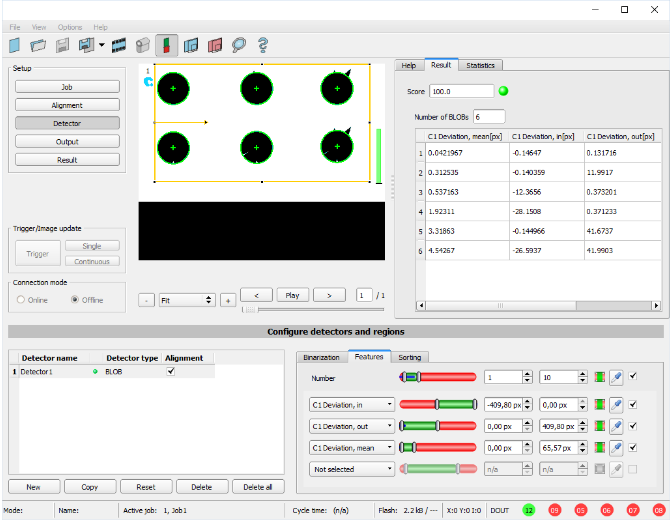
Six circles with different indentations and protruding elements are to be examined for the features “Deviation, in” and “Deviation, out”. In order to improve the presentation, “BLOB contour” is selected in the “Binarization” tab of the “BLOB” detector. Now the detector marks the contours of all circles in the search field in cyan.

In the “Features” tab, the features:

- “C1 Circle, fit” (first-level feature), “Deviation, in” (second-level feature)
- “C1 Circle, fit” (first-level feature), “Deviation, out” (second-level feature)
- “C1 Circle, fit” (first-level feature), “Deviation, mean” (second-level feature)

are selected.

Now the results of the features per circle can be read in the “Result” tab, see also the following figure. (Please note: The results can be assigned to the circles by moving the mouse over the circles in the field of view.)



Score 100.0

Number of BLOBs 6

	C1 Deviation, mean[px]	C1 Deviation, in[px]	C1 Deviation, out[px]
1	0.0421967	-0.14647	0.131716
2	0.312535	-0.140359	11.9917
3	0.537163	-12.3656	0.373201
4	1.92311	-28.1508	0.371233
5	3.31863	-0.144966	41.6737
6	4.54267	-26.5937	41.9903

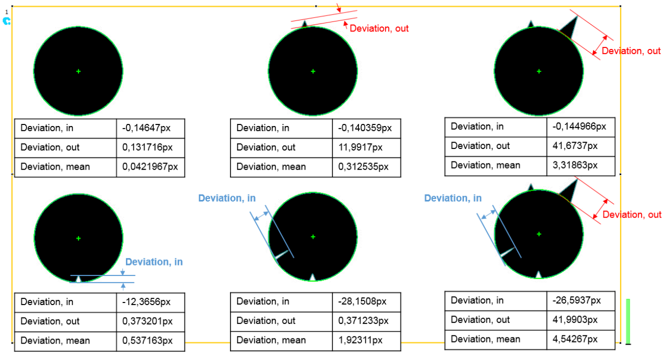
	Detector name	Detector type	Alignment
1	Detector 1	BLOB	☑

Number	1	10	☑
C1 Deviation, in	-409,80 px	0,00 px	☑
C1 Deviation, out	0,00 px	409,80 px	☑
C1 Deviation, mean	0,00 px	65,57 px	☑
Not selected	n/a	n/a	□

Mode: Name: Active job: 1, Job1 Cycle time: (n/a) Flash: 2.2 kB / --- X:0 Y:0 I:0 DOUT 12 09 05 06 07 08

Fig. 176: “Deviation, in”, “Deviation, out” and “Deviation, mean” results

The figure below serves to allocate and interpret the results from the screenshot above.



Deviation, in	-0,14647px
Deviation, out	0,131716px
Deviation, mean	0,0421967px

Deviation, in	-0,140359px
Deviation, out	11,9917px
Deviation, mean	0,312535px

Deviation, in	-0,144966px
Deviation, out	41,6737px
Deviation, mean	3,31863px

Deviation, in	-12,3656px
Deviation, out	0,373201px
Deviation, mean	0,537163px

Deviation, in	-28,1508px
Deviation, out	0,371233px
Deviation, mean	1,92311px

Deviation, in	-26,5937px
Deviation, out	41,9903px
Deviation, mean	4,54267px

Fig. 177: Interpretation of results: “Deviation, in”, “Deviation, out” and “Deviation, mean”

Feature Face up / Face down

“Face up/down, area” and “Face up/down, contour” assess the symmetry of the BLOB with respect to an axis determined by the center and the orientation of the BLOB. If a BLOB is fully symmetric with respect to this line the result value will be 0 otherwise it will deviate from 0. The sign of the value indicates whether the side to the left or right is “stronger”.

“Face up/down, area”and “Face up/down, contour” can be used e.g. for distinguishing between face up and face down position of an object as necessary in pick-and-place applications or with vibratory feeders.

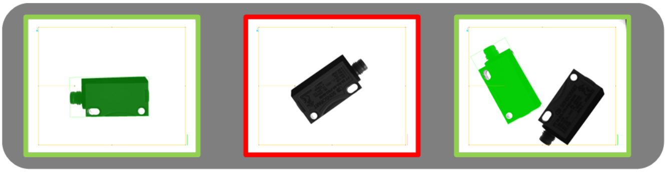


Fig. 178: Face up / Face down, area or contour

The left image displays the target object face down. The thresholds are chosen in a way that this position is considered OK. The image in the middle shows the same object face up and is considered not OK. The right image displays both objects in the image and only the one with face up is considered OK.

- “Face up/down, area” takes each pixel belonging to the BLOB into account for the calculation.
- “Face up/down, contour” only takes the pixels belonging to the BLOB’s contour into account.

This method can be used, if e.g. the object inside the contour varies or is subject to changes due to reflections or other environmental influences.

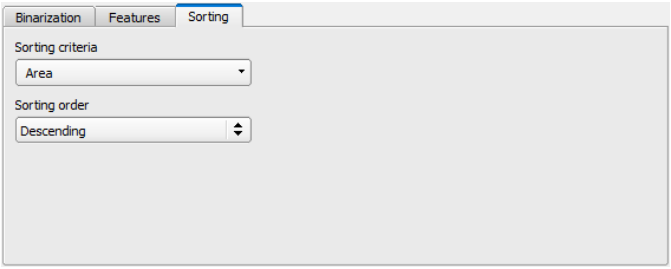
The axis used for the calculation is determined by the center and the rotation angle (360°) of the BLOB. Thus these values are dependent of the geometric model for the BLOB that has been chosen (e.g. smallest enclosing rectangle (rectangle 2) or equivalent ellipse (ellipse 1)).

The geometric model has to be chosen in a way that its orientation (360°) returns a stable and unambiguous value. Thus highly symmetric objects (e.g. perfect rectangles, circles, squares or point-symmetric objects) cannot be reliably evaluated with this method. For objects where the smallest enclosing rectangle (rectangle 2) returns an unambiguous orientation angle, e.g. “L”-shaped geometries or right-angled triangles, the ellipse model might return better results.

4.6.3.8.5 Detector BLOB, tab sorting

The features that have been defined in the tab features Detector BLOB, tab “Features” (Page 199) are calculated for each individual BLOB. For each BLOB the results of these calculations will be sent to the PLC or computer, if this feature is defined as a telegram Telegram, Data output (Page 296).The sequence of these results is defined in the tab “Sorting”.

If e.g. the feature “Center of gravity Y” is calculated and there are 5 BLOBs in the image, the telegram comprises the results of all 5 BLOBs. If sorting criterium “Area” and order “Descending” are selected, the result (here: Center of gravity Y) of the BLOB with the largest area will be transmitted first.



Sorting criteria: Area

Sorting order: Descending

Fig. 179: Detector BLOB, tab Sorting

Settings in tab Sorting

Parameter	Function
Sorting criteria	As a sorting criteria any feature explained in tab “Features” can be selected.
Order	Sorting order “Descending” or “Ascending”.

4.6.3.9 Detector Caliper

With this detector you can control the dimensional accuracy of an object.

Tab Color channel (Page 211)

Detector Caliper, tab Probe (Page 214)

Detector caliper, tab “Distance” (Page 215)

Detector caliper, tab “Optimization” (Page 216)

Caliper results / Histogram display (Page 218)

4.6.3.9.1 Tab Color channel

In the color channel tab, a color image (3 channel) can be converted to a gray value image (1 channel). In contrast to the gray value image of a monochrome VISOR® vision sensor, contrasts can be significantly increased. The highlighting of a color can be set individually for each detector. Thus, the flexibility compared to the use of optical color filters is significantly higher.

The image displayed is dependent on the selected detector.

- Color detectors: Display always colored.
- Object detectors: Monochrome image, display dependent upon selected color model and color channels.

The following parameters can be configured in the Color channel tab:

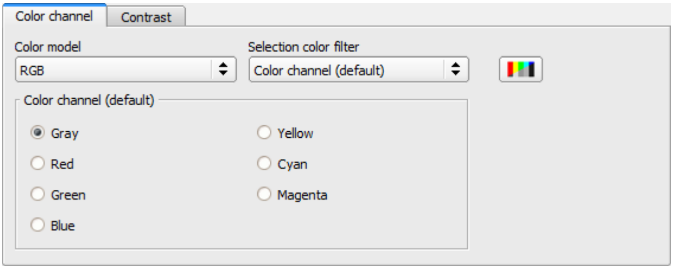
Parameter	Function
Color model	Color models: RGB, Color model RGB (Page 324) HSV, Color model HSV (Page 325) LAB, Color model LAB (Page 325)
Selection color filter	Depending on the color space, all or part of the following color filters are available: Color channel (default) Color distance Binarization
▥	Switching the image between color and monochrome.

4.6.3.9.1.1 Selection color filter

The following color filters are available:

Color channel (default)

The selected color channel is used as a gray value image.



Color channel | Contrast

Color model: RGB

Selection color filter: Color channel (default)

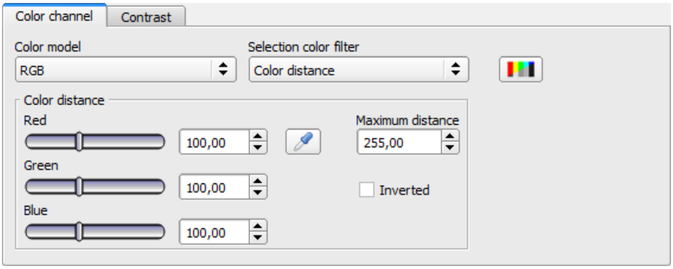
Color channel (default)

● Gray
○ Red
○ Green
○ Blue
○ Yellow
○ Cyan
○ Magenta

Fig. 180: Color filter, Color channel (default)

Color distance

A color is selected as reference color by specifying the color model values or by pipette. The gray value image indicates the distance of each pixel to this reference color. Typical application: Segmentation of characters for OCR.



Color model: RGB

Selection color filter: Color distance

Color distance

Red: 100,00

Green: 100,00

Blue: 100,00

Maximum distance: 255,00

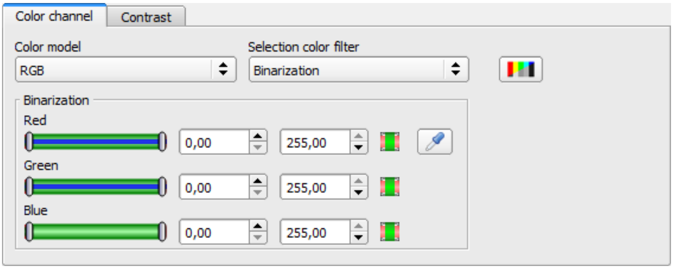
□ Inverted

Fig. 181: Color filter, Color distance

Parameter		Function
Red Green Blue	Lightness A B	Color channels: The color channel can be set via the slider or by entering a value (default 0).
Pipette button		With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.
Maximum distance		Distance of the current color versus the taught-in color. Colors that will exceed the maximum color distance will be black or white depending on the setting of “Inverted”.
Inverted		Inversion of the color distance image.

Binarization

A color range is selected. All pixels within this color range become white. Pixels with deviating color values become black.



Color model: RGB

Selection color filter: Binarization

Binarization

Red: 0,00 255,00

Green: 0,00 255,00

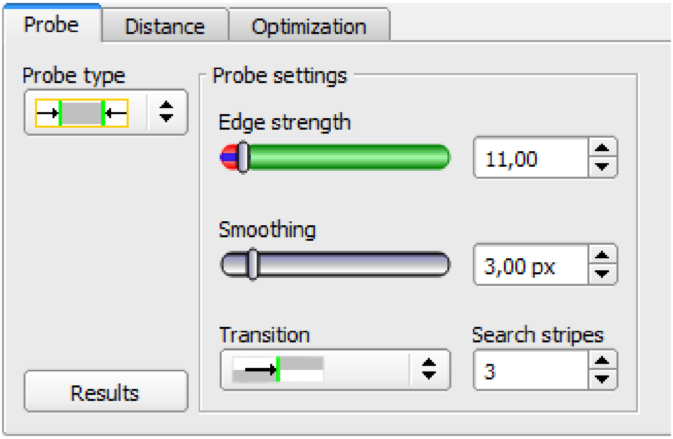
Blue: 0,00 255,00

Fig. 182: Color filter, Binarization

Parameter			Function
Red Green Blue	Hue Saturation Value	Lightness A B	Determination of the color range. The color ranges can be set via the slider or by entering a value.
Inverting button			The current setting is inverted when selecting the button.
Pipette button			With the selection of the pipette button and a subsequent click into the image, the selected color channel is determined automatically.

4.6.3.9.2 Detector Caliper, tab Probe

In this tab all parameters of the probe(s) can be set and the result / histogram display can be accessed.



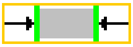
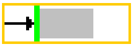
Edge strength 11,00

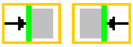
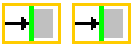
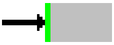
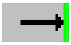
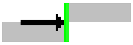
Smoothing 3,00 px

Transition

Search stripes 3

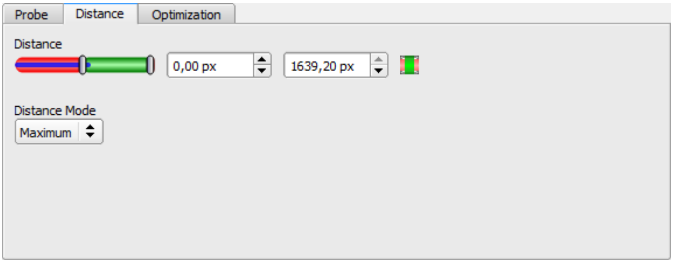
Fig. 183: Detector Caliper, tab Probe

Parameter	Function
Probe type	Selection of Probe type: • One probe, both sides  • One probe, one side  • Two probes, antiparallel (opposite direction)

Parameter	Function
	 - Two probes, same direction 
Edge strength	Edge strength / contrast above which an edge should be detected as an edge
Smoothing	Edges are smoothed in search direction. With higher values blurred or not to the search direction perpendicular edges are detected more reliably. Also tightly located bright-dark-bright or dark-bright- dark transitions can be eliminated. This way you can fade-out scratches or other disturbing edges. Via the Result button the effects for smoothing can be monitored in the histogram window.
Transition	Selection between: - Light → dark  - Dark → light  - Both directions (light-dark and dark-light transition) 
Search stripes	Number of parallel search stripes into which the width of the search zone is to be divided. Edge detection is processed in each search stripe over the whole width. The bigger the number of search stripes, the more probable the very first edge will be found. (Finer detection - longer execution time).
Results	Opens result and histogram display

4.6.3.9.3 Detector caliper, tab “Distance”

In this tab all parameters of the searched for distance can be set.



Distance: 0,00 px – 1639,20 px

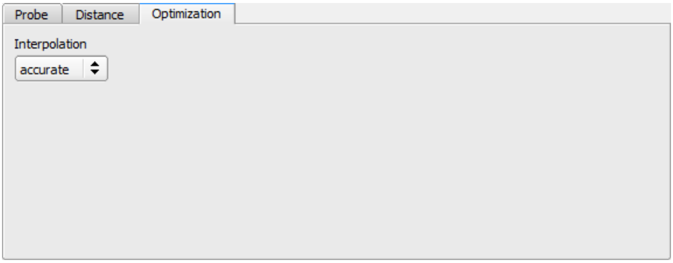
Distance Mode: Maximum

Fig. 184: Detector caliper, tab distance

Parameter	Function
Distance	Distance in pixels, with two limits for tolerance band Blue bar: current distance value
Distance mode	For each search stripe one touching point is calculated. If the number of search stripes >1 there a different possibilities how the final result is calculated. • Maximum: The touching point which represents the longest distance is selected. • Minimum: The touching point which represents the smallest distance is selected. • Mean (Average): All touching points are arithmetically averaged. If there are outliers these are also used for the calculation, and do influence the result. • Median: The values of the touching point are sorted ascending and the middle (central) value in the list is chosen. Outliers do not influence the result.

4.6.3.9.4 Detector caliper, tab “Optimization”

In the “Optimization” tab, further settings for optimizing the edge detection can be made. The following figure shows the “Optimization” tab.



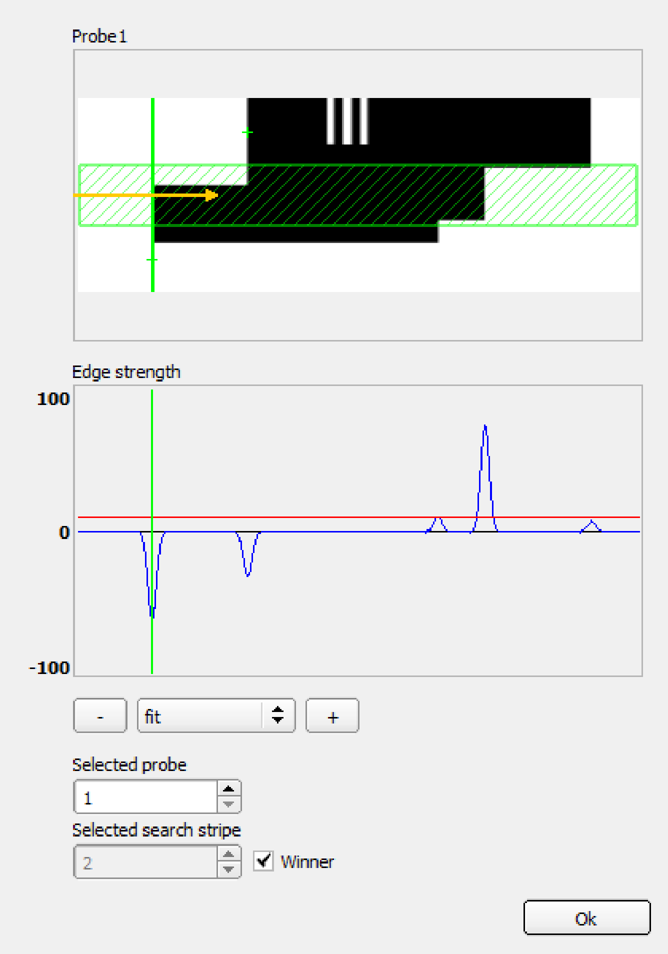
Interpolation: accurate

Fig. 185: Detector caliper, “Optimization” tab

The following parameters can be configured in the “Optimization” tab:

Parameter	**Function**
Interpolation	The calculation of the edge position can be performed either with sub-pixel accuracy (up to 1/10 pixels) or with simple accuracy.
• Accurate	Subpixel accuracy
• Fast	Simple accuracy: This setting partly provides calculations which are over 50% faster.

4.6.3.9.5 Caliper results / Histogram display



Selected probe
1

Selected search stripe
2 ☑ Winner

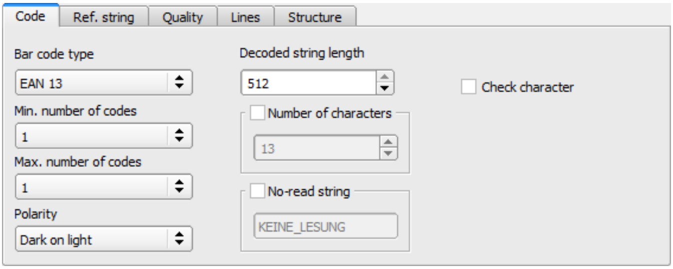
Fig. 186: Caliper results / Histogram display

Parameter	**Function**
Probe (x)	Image of probe (x) with: • Green line: detected overall result edge • Green crosses: detected edge transition per search ray

Parameter	Function
	• Light blue zone: display of "Selected search ray"
Edge strength	Histogram with: • Blue line: contrast gradient in image, depending on "Selected search ray" • Red line: required contrast for edge detection (Threshold) • Light blue line: detected edge transition, depending on "Selected search ray"
Fit, "+", "-"	Fit or zoom of "edge strength" histogram
Selected search stripe	Selection of search stripe to be displayed in "Probe (x)" image • Winner: winner search stripe (depending on settings in "Distance/Distance mode") • "1, 2, .." Number of search stripe

4.6.3.10 Detector Barcode

4.6.3.10.1 Detector Barcode, tab Code



Bar code type: EAN 13

Min. number of codes: 1

Max. number of codes: 1

Polarity: Dark on light

Decoded string length: 512

☐ Number of characters: 13

☐ No-read string: KEINE_LESUNG

☐ Check character

Fig. 187: Detector Barcode, tab Code

Settings in tab Code

Parameters	Functions
Bar code type	Select here the type of barcode to be read with the Code reader.
Decoded string length	Max. Length of a barcode. If contents of code are longer than this value, the rest will be cut off. If more than one code is read by this detector, this value has to be set for the longest code.

Parameters	Functions
Check character	This setting activates the processing for a check character in case it is part of the code. Barcodes with check characters are e.g. Code 39, Codabar, 25 Industrial or 25 Interleaved. If this setting is not activated, the check character will be given out with the normal result string.
Min. number of codes	Minimum number of codes to be read inside the search area.
Max. number of codes	Maximum number of codes to be read inside the search area. If this value is set higher than necessary, the reading time may increase slightly.
Number of characters	Number of expected characters in the barcode. Codes with a different number of characters are ignored. If the number of characters of the code is known, this check increases the detection. If codes with a certain number of characters are to be found under several codes, then the parameter "Max. number of codes" is to be set to a higher value than the number of searched codes.
No-read string	Specifies the text, which is given out over the interfaces in case of non successful reading.
Polarity	Specifies printing of code "black on white" or "white on black".

For newly generated detectors, all parameters are present as standard values, suitable for many applications.

Optimisation:

Execution speed:

- Search zone for position (yellow frame) only as large as necessary

Robust detection:

- Search zone for position (yellow frame) sufficiently large?
- Contrasts for model and image suitably set? (for model visible in sample)
- Are thresholds set correctly?

4.6.3.10.2 Detector Barcode, tab Reference string

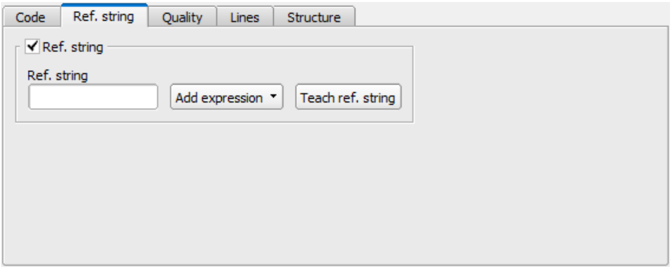


Fig. 188: Detector Barcode, tab Reference string

Settings in tab Reference String

Parameters	Functions
Compare string	Activates verification of contents of the result information. The verification is done by using of regular expressions.
Ref. string	This text or regular expression is taken for verification. Here can be entered characters or regular expressions. If codes with a certain reference string are to be found under several codes, then the parameter “Max. number of codes” in the “Code” tab is to be set to a higher value than the number of searched codes.
Add expression	Opens a list with examples for regular expressions.
Teach ref. string	Reads the code under the code reader and takes the contents of this code as a reference string. This text can be changed later.

For newly generated detectors, all parameters are preset as standard values, suitable for many applications.

Examples for reference strings specified by regular expressions:

Reference string	Hit	Example for hit
123	String containing 123	01234
\A123	String beginning with 123	1234
123\Z	String ending by 123	0123

Reference string	Hit	Example for hit
\A123\Z	String matching exactly 123	123
[123]	String containing one of the characters	33
[123]{2}	String containing sequence of the characters of length 2	23
[12]|[34]	String containing a character of one of both groups	4

Most important elements of regular expressions:

^ or \AMatches start of string

$ or \ZMatches end of string (a trailing newline is allowed)

.Matches any character except newline

[...]Matches any character listed in the brackets. If the first character is a '^', this matches any character except those in the list. You can use the '-' character as in '[A-Z0-9]' to select character ranges. Other characters lose their special meaning in brackets, except '\'.

*Allows 0 or more repetitions of preceding literal or group

+Allows 1 or more repetitions

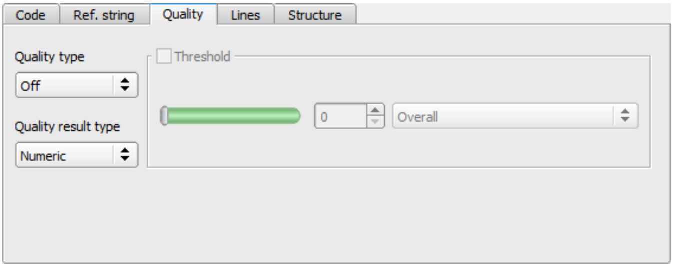
?Allows 0 or 1 repetitions

{n,m}Allows n to m repetitions

{n}Allows exactly 'n' repetitions

|Separates alternative search expressions

4.6.3.10.3 Detector Barcode, tab Quality



Quality type: Off

Quality result type: Numeric

□ Threshold

0

Overall

Fig. 189: Detector Barcode, tab Quality

Settings in tab Quality

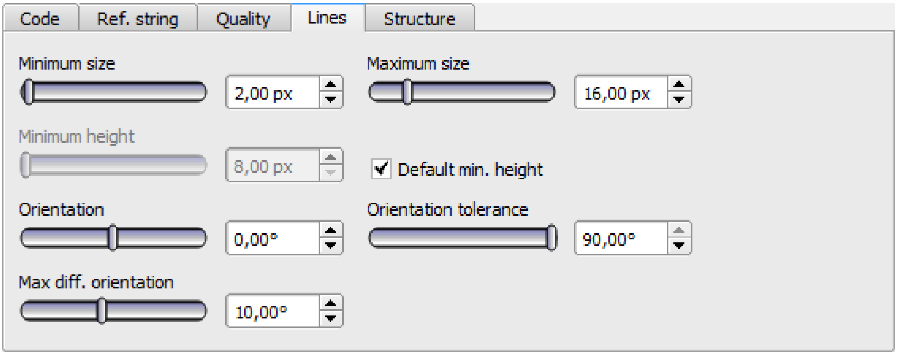
Parameters	Functions
Quality type	Evaluation of printing quality according to international standard ISO/IEC 15416. In order to achieve an evaluation according to the norm, there are defined minimum requirements for the size of the code inside the camera image (resolution) and mounting of camera and illumination. These requirements are specified inside the norm. For simple 1D Barcodes, the rating of printing quality is combined in a total of eight elements: Q1 Overall Q2 Not used Q3 Not used Q4 Decode Q5 Symbol Contrast Q6 Minimal Reflectance Q7 Minimal Edge contrast Q8 Modulation Q9 Defects Q10 Decodability Q11 Additional Requirements “Overall” is rating the total quality, the further elements give information about possible reasons for a reduced quality. Inside ISO/IEC15416 there is a list with common defects and their influence to the single grades. The single quality grades are defined as follows: “Overall” is the minimum value of all other grades. “Decode” has value 4 when the code was read and value 0 when the code was not read. “Symbol contrast” is the difference between minimum and maximum reflexion value of grayscale, better contrast gives better grading. “Minimal reflectance” is set to 4 if the lowest reflectance value in the scan reflectance profile is lower or equal to 0.5 of the maximal reflectance value. Otherwise a value of 0 is assigned. “Edge contrast” is the contrast between any two adjacent elements, either bar-to-space or space-to-bar. The “minimal edge contrast” grades the minimum of the edge contrast values measured in the reflectance profile. “Modulation” indicates how strong the amplitudes of the bar code elements are. Big amplitudes make the assignment of the elements to bars or spaces more certain, resulting in a high modulation grade. “Defects” is a grading of reflectance irregularities found within elements and quiet zones. “Decodability” grade reflects deviations of the element widths from the nominal widths defined for the corresponding symbology. “Additional requirements” are bar code symbology specific requirements: mostly regarding the required quiet zones, but sometimes it can be also related to

Parameters	Functions
	wide/narrow ratio, inter character gaps, guarding patterns or further symbology specific characteristics. For composite codes, the rating has 24 grades: OVERALL: Q1 Overall Q2 Overall Linear Q3 Overall Composite LINEAR: Q4 Decode Q5 Symbol Contrast Q6 Minimal Reflectance Q7 Minimal Edge contrast Q8 Modulation Q9 Defects Q10 Decodability Q11 Additional Requirements COMPOSITE: Q12 Decode Q13 Rap Overall COMPOSITE RAP: Q14 Contrast Q15 Minimal Reflectance Q16 Minimal Edge Contrast Q17 Modulation Q18 Defects Q19 Decodability Q20 Codeword Yield Q21 Unused Error Correction Q22 Modulation Q23 Decodability Q24 Defects The “overall” grade in the group OVERALL is the final symbol grade to be reported. It is just the lower from the other two in the group: “overall linear” and “overall composite”, which are the overall grades of the linear and the composite sub symbols, respectively. The other two groups, “LINEAR” and “COMPOSITE”, contain the corresponding individual grades for both sub symbols, and give information for possible causes for poor quality of the symbol. The grades in the “LINEAR” group correspond to those for the simple 1D bar code case, described above. The grades in the “COMPOSITE” group correspond to the grades for a PDF 417

Parameters	Functions
	data code symbol, where "rap overall" is called after the specific, so-called RAP, start/stop pattern of Composite symbols. Additionally, the sub group "COMPOSITE RAP" expands the individual grades for the reflectance profile of the RAP patterns. The RAP grades are consistent with the grades for the simple 1D bar code case explained above.
Quality result type	There are existing two possibilities, to display quality parameters. Both are according to the norm. The grades can be given in values from A to F or from 4 to 0. A and 4 are the best possible grades. This setting determines how the grades should be displayed. It affects the display on screen as well as the output over the interfaces. The assignment is the following: ABCDF 43210

For newly generated detectors, all parameters are preset as standard values, suitable for many applications.

4.6.3.10.4 Detector Barcode, tab Lines



Minimum size 2,00 px

Maximum size 16,00 px

Minimum height 8,00 px

☑ Default min. height

Orientation 0,00°

Orientation tolerance 90,00°

Max diff. orientation 10,00°

Fig. 190: Detector Barcode, tab Lines

Settings in tab Lines

Parameters	Functions
Minimum size	Minimal size of bar code elements, i.e. the minimal width of bars and spaces. For small bar codes the value should be reduced to 1.5. In the case of huge bar codes the value should be increased, which results in a shorter execution time.
Maximum size	Maximal size of bar code elements, i.e. the maximal width of bars and

Parameters	Functions
	spaces. This value should be adequate low such that two neighbouring bar codes are not fused into a single one. On this other hand the value should be sufficiently high in order to find the complete bar code region.
Minimum height	Minimal bar code height. In the case of a bar code with a height of less than 16 pixels the respective height should be set by the user. Note, that the minimal value is 8 pixels. If the bar code is very high, i.e. 70 pixels and more, manually adjusting to the respective height can lead to a speed-up of the subsequent finding and reading operation.
Orientation	With the parameter *Orientation* the range of angel for code reading can be restricted. Barcodes with rotated positions outside the specified angle range are not read. For example, the parameter can be used if a barcode can be located in different rotated positions in front of the code reader and not all rotated positions should be read. If codes with a certain rotated position are to be found under several codes, then the parameter *Max. number of codes* in the *Code* tab is to be set to a higher value than the number of searched codes. If the bar codes are expected to appear only in certain orientations in the processed images, one can reduce the orientation range adequately. This enables an early identification of false candidates and hence shorter execution times. This adjustment can be used for images with a lot of texture, which includes fragments tending to result in false bar code candidates.
Orientation tolerance	Orientation tolerance. See the explanation of 'orientation' parameter.
Max. diff orientation	A potential bar code region contains bars, and hence edges, with a similar orientation. This value denotes the maximal difference in this orientation between adjacent pixels and is given in degree. If a bar code is of bad quality with jagged edges this parameter should be set to bigger values. If the bar code is of good it can be set to smaller values, thus reducing the number of potential but false bar code candidates.

For newly generated detectors, all parameters are present as standard values, suitable for many applications.

Parameter Orientation

The following figure illustrates the orientation. Please note: The orientation indication refers to the image and not to the rotational position of the search range.

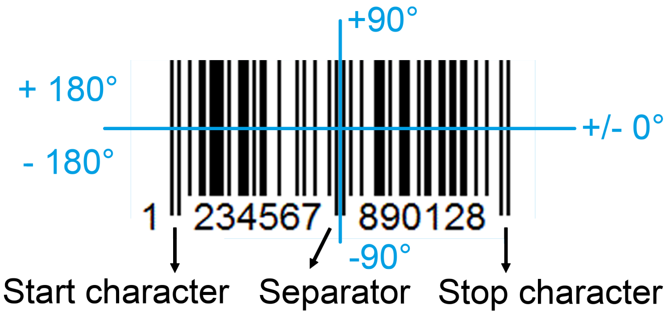


Fig. 191: Parameter Orientation

4.6.3.10.4.1 Optimization:

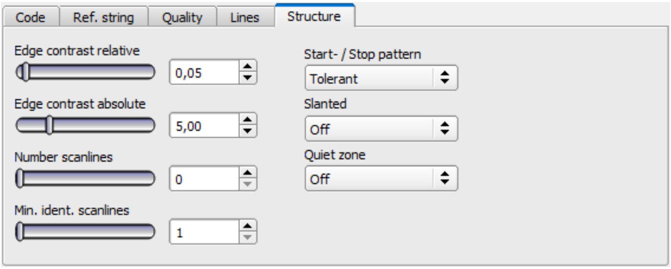
Execution speed:

- Search zone for position (yellow frame) only as large as necessary

Robust detection:

- Search zone for position (yellow frame) sufficiently large?
- Contrasts for model and image suitably set? (for model visible in sample)
- Are thresholds set correctly?
- Code size sufficient in the field of view?
- Width of barcode line sufficient?

4.6.3.10.5 Detector Barcode, tab Structure



Edge contrast relative 0,05

Edge contrast absolute 5,00

Number scanlines 0

Min. ident. scanlines 1

Start- / Stop pattern Tolerant

Slanted Off

Quiet zone Off

Fig. 192: Detector Barcode, tab Structure

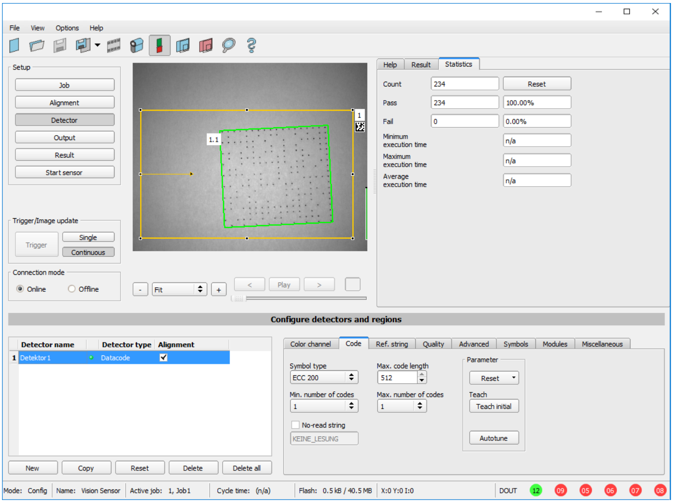
Settings in tab Structure

Parameters	Functions
Edge contrast relative	Edge contrast relative Edges inside barcode are found by setting of a threshold. Parameter ' Edge contrast relative´ defines how this threshold in respect to the dynamic range of the scan line pixels is calculated. In the case of disturbances in the bar code region or a high noise level, the value of ´Edge contrast relative´ should be increased. Typical values: [0.05 ... 0.2]; Default: 0.05
Edge contrast absolute	Edge contrast absolute prevents misdetections of edges. For images with high noise levels this value should be higher. In noise-free images with very weak contrast, this parameter might disturb the detection of real edges. So it might be necessary to reduce it or even completely disable it by setting it to 0.0. Typical values: [0.0 ... 10.0]; Default: 5.0
Number scanlines	Number of scanlines used during the scanning of a code. Reducing the number of scanlines improves speed.Images with higher quality need less scanlines than images of lower quality. For an average image, a value between 2 and 5 should be good. If a code can not be detected any more after reducing the number of scanlines, the number has to be increased again. Typical values: [0, 5, 10, 20 ...]; Default: 0
Min. ident. scanlines	Minimal number of identical scanlines for a decoding of a code symbol to be accepted. If this parameter is not set (has a value of 0) a bar code is considered decoded with the first scanline, which was successfully decoded. Increasing this parameter to 2 or more is useful to avoid wrong readings. Typical values: [0, 2, 3, ...]; Default: 0
Start- / Stop pattern	Set searching criteria for a start or stop pattern to 'tolerant' or 'accurate'. 'Tolerant' will increase the detection chances of a bar code especially in images with low contrast. 'Accurate' increases the robustness against false detections. List of values: 'Tolerant', 'Accurate'; Default: 'Tolerant'
Slanted	If 'slanted' = 'On' improves readability of codes if single lines are orientated different from the others like when the code is not on a plain surface. If 'slanted' = 'Off' default setting when all lines of the barcode are parallel in image. If 'slanted = 'Auto' the sensor tries first ‘On’ and then ‘Auto’, this setting can increase reading time. List of values: 'Off', 'Auto', 'On'; Default: 'Off'
Quiet zone	Enforces the detection of the quiet zones of a bar code.With 'Quiet zone' ='on' the Quiet zones must be at least as wide as specified by the corresponding bar code standard.

Parameters	Functions
	With 'Quiet zone' set to an integer value greater than or equal 1, the quiet zones must be at least as wide as 'Quiet zone' x X pixels. With 'Quiet zone' = 'tolerant' a limited number of edges are allowed in the quiet zone, but at most 1 per 4 module widths. The intent of this is to prevent detecting only part of a bar code, while still allowing to read bar codes with simple quiet zone violations. With 'Quiet zone' = 'off', the quiet zones detection is disabled. Detection of quiet zone prevents that simple bar code types are detected inside of a longer bar sequence. Usually, values between 2 and 4 achieve optimal results by effectively suppressing false bar codes, but still tolerating small disturbances, textures, label edges, etc. next to the symbol. Typical values: 'Off' 'On', 1, 2, 3, 4, 5; Default: 'Off'

4.6.3.11 Detector Datacode

4.6.3.11.1 Detektor Datacode, tab *Code*



Count: 234
Pass: 234 100.00%
Fail: 0 0.00%
Minimum execution time: n/a
Maximum execution time: n/a
Average execution time: n/a

Detector name	Detector type	Alignment
1 Detektor1	Datacode	☑

Symbol type: ECC 200
Max. code length: 512
Min. number of codes: 1
Max. number of codes: 1
□ No-read string: KEINE_LESUNG

Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 0.5 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 193: Detector Datacode, tab Code

Settings in tab Code

Parameters	Functions
Symbol type	Select here the type of code to be read with the Code reader.
Code length	Max. Length of a barcode. If the contents of code are longer, the rest will be cut off. If more than one code is read by this detector, this value has to be set for the longest code.
Min. number of codes	Minimum number of codes to be read inside the search area.
Max. number of codes	Maximum number of codes to be read inside the search area. If this value is set higher than necessary, the reading time may increase slightly.
Reset	Reset parameters are for setting the code reading parameters back to the default state before teaching. There can be selected “standard”, “enhanced” and “maximum”. “Standard” is setting the default parameters in a way that most of the codes can be read. If your code can not be read, please use setting “Enhanced”. If the code still cannot be read, use setting “Maximum”. Settings “Enhanced” and “Maximum” may increase the reading time. This reset function is only for resetting the detector parameters, not for resetting of other settings outside the detector (i.e. general settings like illumination, in-outputs, serial settings etc.). After resetting the parameters, there can be made an initial teach, again.
Initial teach / Additive teach	Teach: the region of interest is searched for codes. If a code was found the parameters are set for this code. After successful teaching, the code will be marked with a green frame. After teaching a code the code reader will search in “run”-mode only for this type of code. Once teaching was done at least one time successful, this button is named “Teach additive”. “Teach additive” is for extending the parameters either in order to read several different codes in one detector or in order to cover differences in printing quality.
Autotune	Automatic setting (pre-processing filter and image acquisition) for the optimization of code reading.
No-read string	Specifies the text, sent out over the interfaces in case of non successful reading.

For newly generated detectors, all parameters are preset as standard values, suitable for many applications.

Optimization

Execution speed:

- Search zone for position (yellow frame) only as large as necessary

Robust detection:

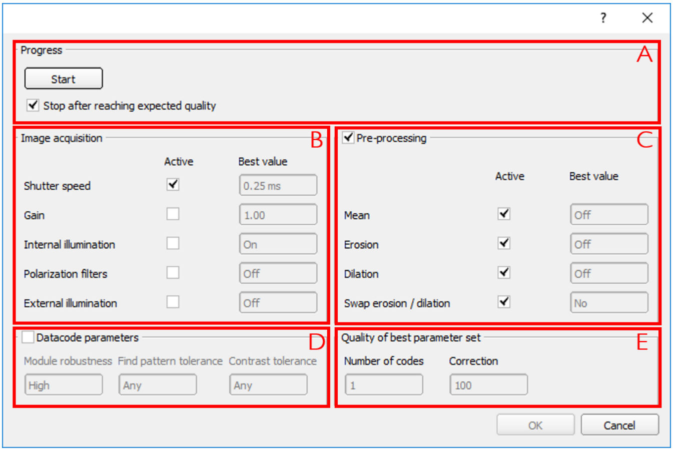
- Search zone for position (yellow frame) sufficiently large?
- Contrasts for model and image suitably set? (for model visible in sample
- Are thresholds set correctly?

4.6.3.11.1.1 Autotune

When selecting the Autotune function, the VISOR® vision sensor automatically detects all the settings to optimize code reading.

The function always starts with the parameters already set by the user. So, if parameters are set roughly before the start of the “Autotune” function, Autotune makes the fine adjustment to optimize the result.

After the optimization run of “Autotune”, “OK” or “Cancel” can be selected. In case of “OK”, the altered parameters are used. When “Cancel” is selected, the old parameters are restored to the values from before the “Autotune” execution.



Progress

Start

☑ Stop after reaching expected quality

Image acquisition

	Active	Best value
Shutter speed	☑	0.25 ms
Gain	□	1.00
Internal illumination	□	On
Polarization filters	□	Off
External illumination	□	Off

☑ Pre-processing

	Active	Best value
Mean	☑	Off
Erosion	☑	Off
Dilation	☑	Off
Swap erosion / dilation	☑	No

□ Datacode parameters

Module robustness	Find pattern tolerance	Contrast tolerance
High	Any	Any

Quality of best parameter set

Number of codes	Correction
1	100

OK Cancel

Fig. 194: “Autotune” Window

The “Autotune” function consists of the following elements:

(A) Progress	
Start	Start “Autotune” function. After pressing start, the progress is displayed.
Stop after reaching expected quality	The “Autotune” function stops the automatic settings when the minimum required quality is reached.

(B) Image acquisition	
Shutter Speed	**Active:** An activated Checkbox defines those parameters, that should be determined automatically by the VISOR® vision sensor. Not activated parameters remain unchanged. **Best Value:** The “Best Value” field shows the last setting, which was determined by the Autotune function.
Gain	
Internal illumination	
Polarization filters	
External illumination	

(C) Pre-processing	
Mean	**Active:** A checkmark in the “Active” field defines those parameters, that the VISOR® vision sensor should determines automatically. Not activated parameters remain unchanged. **Best Value:** The “Best Value” field shows the last setting, which was determined by the Autotune function.
Erosion	
Dilation	
Swap erosion/ dilation	

(D) Datacode parameters	
Module robustness	The best settings found by the Autotune function are displayed.
Find pattern tolerance	
Contrast tolerance	

(E) Quality of best parameter set	
Number of codes	Number of codes in the field of view tested by the Autotune function.
Correction	Decode error, which is achieved with selected parameters.

4.6.3.11.2 Detector Datacode, tab Ref. String

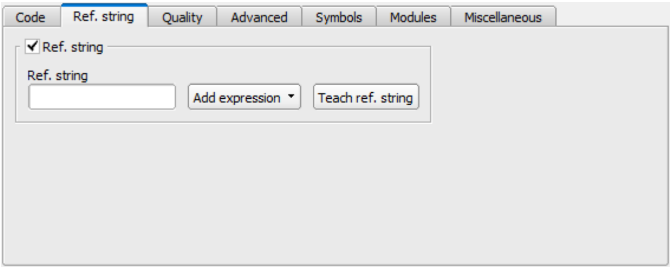


Fig. 195: Detector Datacode, tab Ref. String

Settings in tab Reference String

Parameters	Functions
Compare string	Activates verification of contents of the result information. The verification is done by using of regular expressions.
Ref. string	This text or regular expression is taken for verification. Here can be entered characters or regular expressions. If codes with a certain reference string are to be found under several codes, then the parameter “Max. number of codes” in the “Code” tab is to be set to a higher value than the number of searched codes.
Add expression	Opens a list with examples for regular expressions
Teach ref. string	Reads the code under the code reader and takes the contents of this code as a reference string. This text can be changed later.

For newly generated detectors, all parameters are preset as standard values, suitable for many applications.

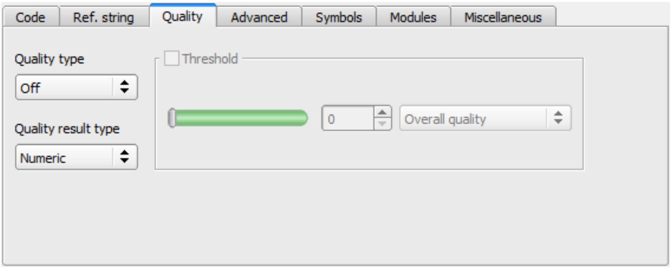
Reference string	Hit	Example for hit
123	String containing 123	01234
\A123	String beginning with 123	1234
123\Z	String ending by 123	0123
\A123\Z	String matching exactly 123	123

Reference string	Hit	Example for hit
[123]	String containing one of the characters	33
[123]{2}	String containing sequence of the characters of length 2	23
[12]|[34]	String containing a character of one of both groups	4

Most important elements of regular expressions:

^ or \A	Matches start of string
$ or \Z	Matches end of string (a trailing newline is allowed)
.	Matches any character except newline
[...]	Matches any character listed in the brackets. If the first character is a '^', this matches any character except those in the list. You can use the '-' character as in '[A-Z0-9]' to select character ranges. Other characters lose their special meaning in brackets, except '\'.
*	Allows 0 or more repetitions of preceding literal or group
+	Allows 1 or more repetitions
?	Allows 0 or 1 repetitions
{n,m}	Allows n to m repetitions
{n}	Allows exactly n repetitions
|	Separates alternative search expressions

4.6.3.11.3 Detector Datacode, tab quality



Quality type: Off

Quality result type: Numeric

□ Threshold: 0 Overall quality

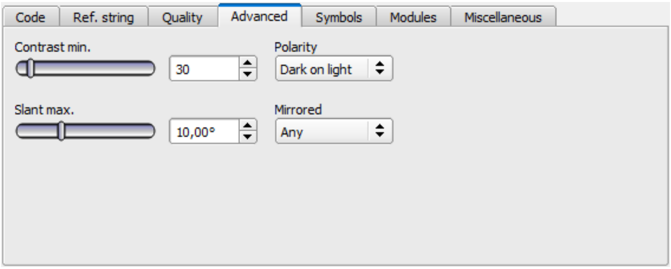
Fig. 196: Detector datacode, tab quality

Settings in tab quality

Parameters	Functions
Quality type	Quality parameters are additional information for rating the printing quality of the code. There are two different standards: AIM DPM-1-2006 and ISO/IEC 15415. Quality parameters are eight single parameters, the definition of the respective elements is as follows: Q1 Overall quality Q2 Contrast Q3 Modulation Q4 Fixed pattern damage Q5 Decode Q6 Axial non-uniformity Q7 Grid non-uniformity Q8 Unused error correction Q9 Mean light The overall quality is the minimum of all individual grades. The contrast is the range between the minimal and the maximal pixel intensity in the data code domain, and a strong contrast results in a good grading. The modulation indicates how strong the amplitudes of the data code modules are. Big amplitudes make the assignment of the modules to black or white more certain, resulting in a high modulation grade. The fixed pattern of both ECC200 and QR Code is of high importance for detecting and decoding the codes. Degradation or damage of the fixed pattern, or the respective quiet zones, is assessed with the fixed pattern damage quality. The decode quality always takes the grade 4, meaning that the code could be decoded. Naturally, codes which cannot be decoded cannot be assessed concerning print quality either. Originally, data codes have squared modules, i.e. the width and height of the modules are the same. Due to a potentially oblique view of the camera onto the data code or a defective fabrication of the data code itself, the width to height ratio can be distorted. This deterioration results in a degraded axial non-uniformity. If apart from an affine distortion the data code is subject to perspective or any other distortions too this degrades the grid non-uniformity. As data codes are redundant codes, errors in the modules or code words can be corrected. The amount of error correcting capacities which is not already used by the present data code symbol is expressed in the unused error correction quality. In a way, this grade reflects the reliability of the decoding process. Note, that even codes with an unused error correction grading of 0, which could possibly mean a false decoding result, can be decoded in a reliable way, because the implemented decoding functionality is more sophisticated and robust compared to the reference decode algorithm proposed by the standard. In order to achieve an evaluation according to the norm, there are defined minimum requirements for the size of the code inside the camera image (resolution)

Parameters	Functions
	and mounting of camera and illumination. These requirements are specified inside the norm. Quality parameters according to AIM DPM-1-2006 are a extension to ISO/IEC 15415 Standard, which define the requirements of the gray value conditions oft he image oft he data code, and so improves the reproducibility of the quality evaluation of different manufacturers. Quality parameters according to AIM consist of one value more than quality parameters according to ISO/IEC 15415. This value is called "Mean Light". "Mean light" is not a quality value of the code, it shows the quality of the image by calculating the average gray value of the bright data code modules. "Mean light" can vary from 0.0 to 1.0. A image has the required gray value conditions if the "mean light" value is between 70% and 86% (0.70 to 0.86).
Quality result type	There are existing two possibilities, to display quality parameters. Both are according to the norm. The grades can be given in values from A to F or from 4 to 0. A and 4 are the best possible grades. This setting determines how the grades should be displayed. It affects the display on screen as well as the output over the interfaces. The assignment is the following: A B C D F 4 3 2 1 0

4.6.3.11.4 Detector Datacode, tab Advanced



Contrast min. 30

Polarity: Dark on light

Slant max. 10,00°

Mirrored: Any

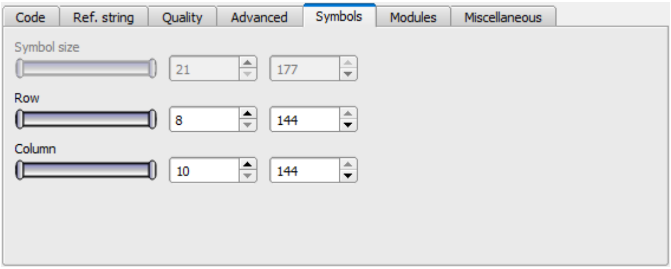
Fig. 197: Detector datacode, tab advanced

Settings in tab Advanced

Parameters	Function
Contrast min.	Minimum contrast in gray values between bright and dark parts of the code, range (1...100).
Polarity	Possible restrictions concerning the polarity of the modules, i.e., if they are prin-

Parameters	Function
	ted dark on a light background or vice versa.
Slant max.	Slant of the L-shaped finder pattern in radians. This is the difference between the angle of the 'L' and the right angle.
Mirrored	Describes whether the symbol is or may be mirrored (which is equivalent to swapping the rows and columns of the symbol). The function helps, if codes should be read through transparent parts like glass.

4.6.3.11.5 Detector Datacode, tab Symbols



Symbol size 21 177

Row 8 144

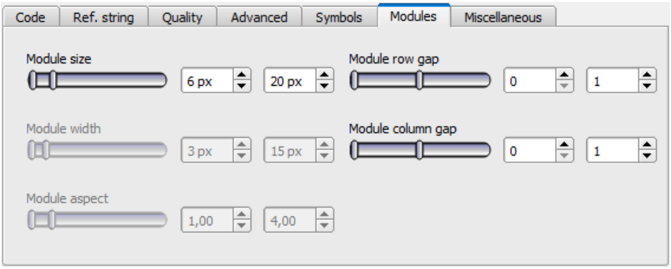
Column 10 144

Fig. 198: Detector Datacode, tab Symbols

Settings in tab Symbols

Parameters	Function
Symbol size	Only QR-Code: Size of symbol inside picture in pixel.
Row	Only ECC200 and PDF 417: Number of rows including finder pattern.
Column	Only ECC200 and PDF 417: Number of columns including finder pattern.

4.6.3.11.6 Detector Datacode, tab Modules



Module size 6 px 20 px

Module row gap 0 1

Module width 3 px 15 px

Module column gap 0 1

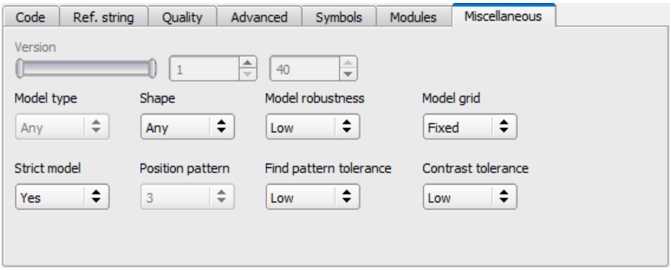
Module aspect 1,00 4,00

Fig. 199: Detector datacode, tab modules

Settings in tab Modules

Parameters	Function
Module size	Size of modules in pixels.
Module width	Only PDF 417: width of modules inside picture in pixels.
Module aspect	Only PDF 417: minimum aspect of modules (rows compared to columns).
Module row gap	Only ECC200 and QR-Code: allowed gap between rows, i.e. at dot peened codes which have no full size modules.
Module column gap	Only ECC200 and QR-Code: allowed gap between columns.

4.6.3.11.7 Detector Datacode, tab miscellaneous



Version: 1 – 40

Model type: Any

Shape: Any

Model robustness: Low

Model grid: Fixed

Strict model: Yes

Position pattern: 3

Find pattern tolerance: Low

Contrast tolerance: Low

Fig. 200: Detector datacode, tab miscellaneous

Settings in tab Miscellaneous

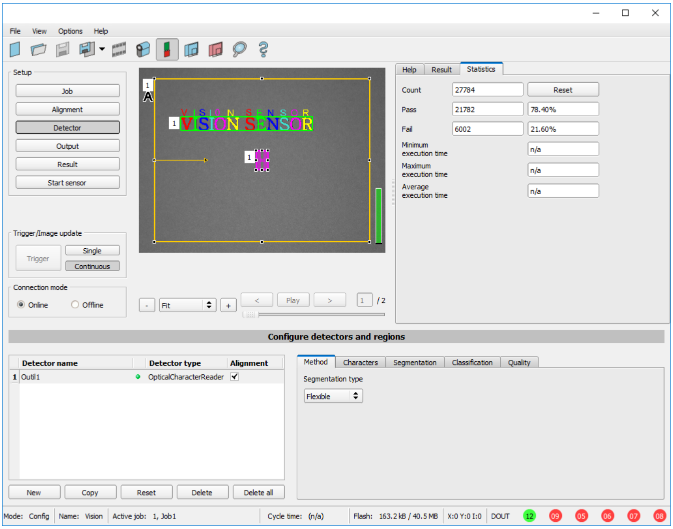
Parameters	Function
Version	Only QR-Code: Minimum symbol version to be read: [1. . . 40]
Model type	Only QR-Code: Type of the QR Code model specification: 1, 2, 0
Shape	Only ECC200 and QR-Code: Possible restrictions concerning the module shape (rectangle and/or square).
Model robustness	Robustness of the decoding of data codes with very small module sizes. Setting the parameter to 'high' increases the likelihood of being able to decode data codes with very small module sizes. Additionally, in that case the minimum mod-

Parameters	Function
	ule size should also be adapted accordingly, thus should be set to the expected minimum module size and width, respectively.
Model grid	Only ECC200: Describes whether the size of the modules may vary (in a specific range) or not. Dependent on the parameter different algorithms are used for the calculation of the module's center positions. If it is set to 'fixed', an equidistant grid is used. Allowing a variable module size ('variable'), the grid is aligned only to the alternating side of the finder pattern. With 'any' both approaches are tested one after the other. Please note that the value of 'module_grid' is ignored if 'finder_pattern_tolerance' is set to 'high'. In this case an equidistant grid is assumed.
Strict model	Specifies, if the code parameters have to be meet completely or not. If this parameter is set to "Yes", all codes outside the parameter range will be ignored.
Position pattern	Only QR-Code: Number of position detection patterns that have to be visible for reading a code (2 or 3).
Find pattern tolerance	Only ECC200: Tolerance of the search with respect to a disturbed or missing finder pattern. The finder pattern includes the L-shaped side as well as the opposite alternating side. In one case ('low'), it is assumed that the finder pattern is present to a high degree and shows almost no disturbances. In the other case ('high'), the finder pattern may be heavily disturbed or missing completely without influencing the recognition and the reading of the symbol. Note, however, that in this mode the run-time may significantly increase.
Contrast tolerance	Tolerance during Code search in regards to strong local contrast variations.

4.6.3.12 Detector OCR

4.6.3.12.1 Detector OCR, Procedure

To set up an OCR Detector please follow these steps. As some steps base on the results of the one which was processed before, for a correct processing the sequence of the steps must be as described.



Connection mode: ● Online ○ Offline

Count	27784	
Pass	21782	78.40%
Fail	6002	21.60%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Detector name	Detector type	Alignment
1	Outil1	OpticalCharacterReader	☑

Segmentation type: Flexible

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 163.2 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 201: Detector OCR

4.6.3.12.1.1 Basic sequence of setting parameters

- Segmentation by use of the tabs “Characters” and “Segmentation” as well as tab “Pre-Processing” in step “Job”.
- Classification by use of tab “Classification” by selection of a font and definition of a reference string.
- Removing of characters which not have been classified with sufficient quality in tab “Quality”.
- Using the OCR-Detector it is not sufficient to set the parameters with only one image. Stable reading results can only be achieved by using a large number of images. We recommend saving typically 20 to 30 images to cover all variations of the process, and optimising parameters in offline mode.

4.6.3.12.1.2 Segmentation

- Optimizing of segmentation by use of the tabs “Characters” and “Segmentation”. Goal is to get a stable segmentation for all single characters. The result of classification “reading result” is not important in this step, this will be optimized later.
- Segmentation can be improved by use of image pre-processing in tab “Job” – “Pre-Processing”, e.g. by use of “Gauss”, “Mean” or “Dilatation”/“Erosion” or a combination of them. To achieve a stable segmentation it is recommended to use smoothing filters like “Gauss” or “Mean”.

- Parameter “Groups of characters” may support segmentation by specification of the number of characters per group.
- Parameter “Max. deviation from base line” specifies, how much the vertical character position may be different from the base line of the font. Value is in percent of character height.
- Verify proper segmentation of all characters before going to step “Classification”. Classification has no influence to segmentation. Faulty segmented characters will be classified wrong.

4.6.3.12.1.3 Segmentation Examples:

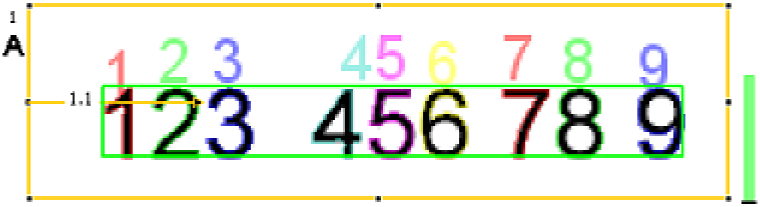


Fig. 202: Segmentation without any preset for parameter “Groups of characters”: All characters are found

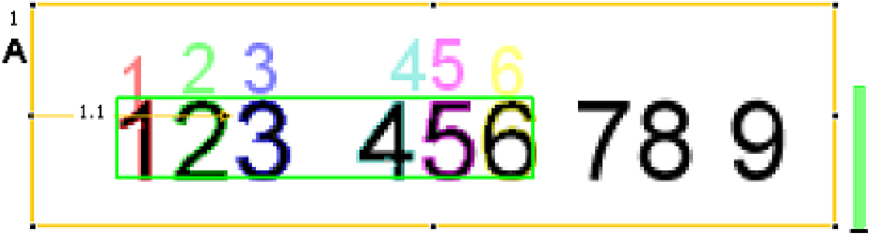


Fig. 203: Segmentation with value “3 3” for parameter “Groups of characters”: Only the both groups of 3 characters are found.

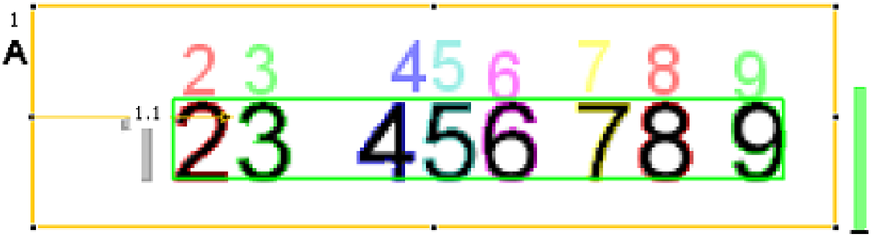


Fig. 204: Segmentation without preset for parameter “Groups of characters”: The segmentation for the first character “1” failed, as it´s contrast to background is much lower than all others.

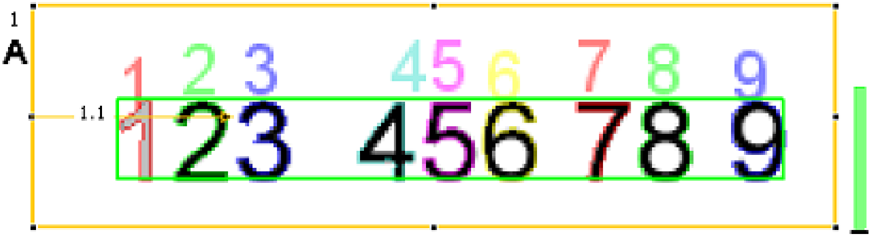


Fig. 205: Segmentation with value “3 3 2 1” for parameter “Groups of characters”: Also the “lower contrast character” get´s segmented.

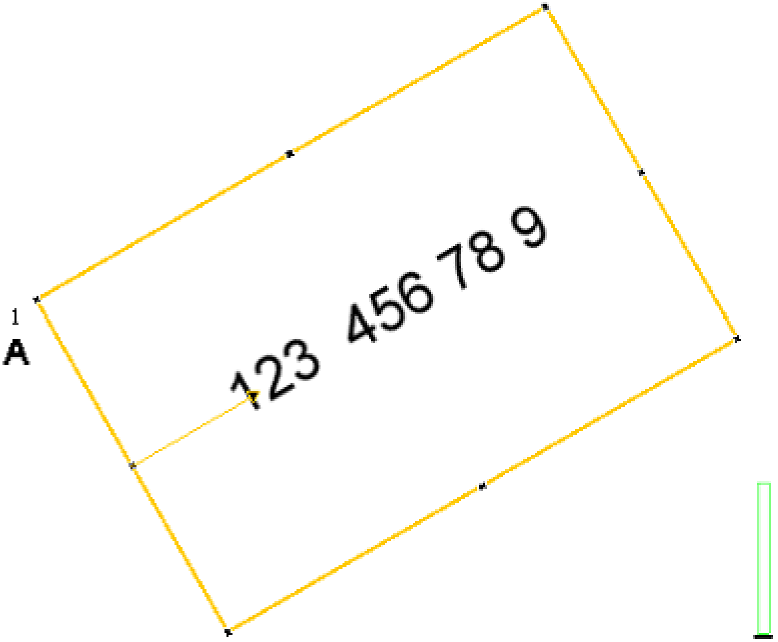


Fig. 206: Segmentation with parameter “Text orientation” = “Font horizontal in image”: No characters are segmented as there are no characters with horizontal orientation in the image.

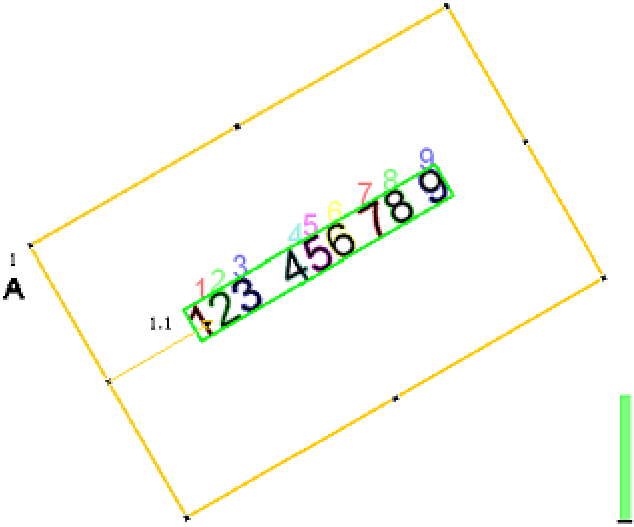


Fig. 207: Segmentation with parameter “Text orientation” = “Font horizontal in ROI”: Segmentation works as characters are horizontal relative to ROI (search area).

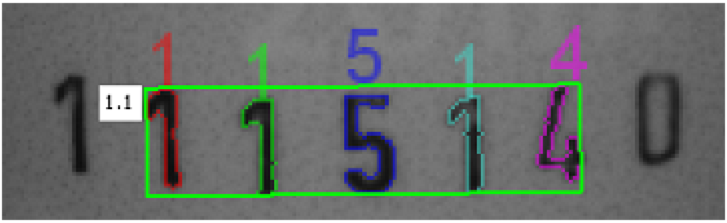


Fig. 208: Segmentation with value 15% for parameter: “Max deviation from base line”: Only the inner five characters are segmented.

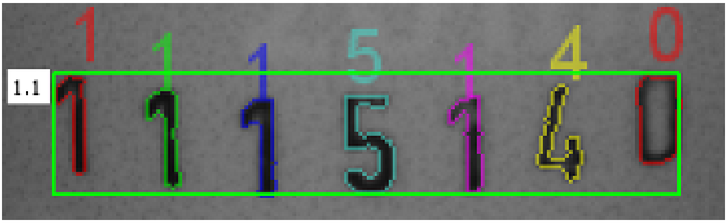


Fig. 209: Segmentation with value 25% for parameter “Max deviation from base line”: All characters are segmented.

4.6.3.12.1.4 Classification

- In this step a suitable character set “font” is selected. Each font is available with different character sets. Goal is to choose the font with the most stable results for the application.
- Naming of fonts by the example of group “Industrial”:
  - “Industrial_0-9”: all numbers
  - “Industrial_0-9+”: all numbers and special characters
  - “Industrial_A-Z+”: all capital letters and special characters
  - “Industrial_0-9A-Z”: all numbers and capital letters
  - “Industrial.omc”: all characters
- Reference string has two functions:

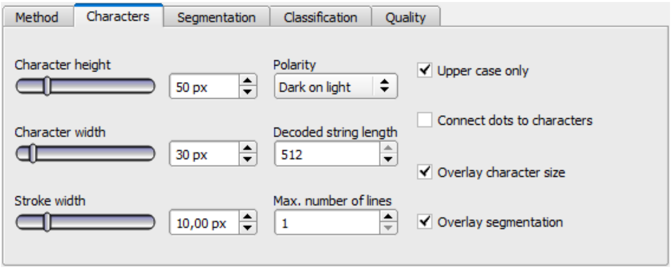
  1. Manipulation of classification (of the recognized characters):
     For each segmented character a rating value (confidence), in relation to each in the whole set of characters (font) available character is calculated.
     If reference string is not used, the character with the highest rating value (confidence) is the winner. By use of reference string the “N” best alternatives will be considered (No. of alternatives). Maximum number of allowed character changes which did not have the maximum rating value (confidence) is specified in: “No. of corrections”.
  2. Manipulation of detector result:
     A minimum quality for complete string is specified (Threshold). If quality is below the threshold, detector result will be “false”.

4.6.3.12.1.5 Quality

- If quality of one of the classified characters is below “Minimum confidence”, the detector result will be “false”.
- Low confidence shows, that a character was not classified reliably. High confidence value however, is not a guarantee for reliable classification!

4.6.3.12.2 Detector OCR, tab Character (flexible)

Basic settings for characters to read.



Character height: 50 px
Character width: 30 px
Stroke width: 10,00 px
Polarity: Dark on light
Decoded string length: 512
Max. number of lines: 1
☑ Upper case only
□ Connect dots to characters
☑ Overlay character size
☑ Overlay segmentation

Fig. 210: Detector OCR, tab Character

Parameters	**Functions**
Character height	Max. height of character in pixels.
Character width	Max. width of character in pixels.
Stroke width	Average width of lines of characters in pixels.
Polarity	Possibility to select between dark characters on bright background or vice versa.
Text orientation	“Font horizontal in Image”: text has to be horizontal in camera image. Rotated text will be not read or wrong read. “Font horizontal in ROI”: by rotation of ROI a rotation angle for reading of rotated text can be specified.
Max. number of lines	Max. number of lines to read.
Upper case only	Limitation to capital letters only.
Connect dots to characters	Connects single dots, e.g. of a dotted font or of a bad printed font for complete characters.
Overlay character size	Switch on and off overlay rectangle for size of letters.
Overlay segmentation	Switch on and off colored overlay for segmentation of characters.

For newly generated detectors, all parameters are preset as standard values, suitable for many applications.

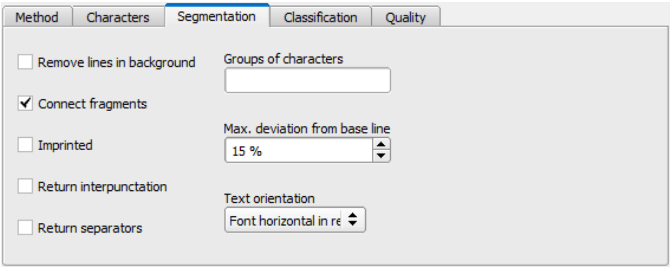
Optimisation:

Execution speed:

- Search zone for character (yellow frame) only as large as necessary

4.6.3.12.3 Detector OCR, tab segmentation

Definition of basic settings of characters to read.



□ Remove lines in background

☑ Connect fragments

□ Imprinted

□ Return interpunctation

□ Return separators

Groups of characters:

Max. deviation from base line: 15 %

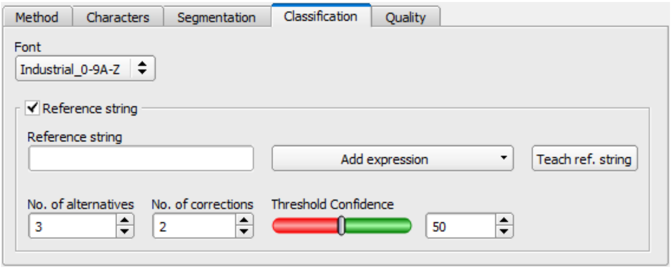
Text orientation: Font horizontal in re

Fig. 211: Detector OCR, tab segmentation

Parameters	**Functions**
Remove lines in background	This parameter can be used to remove disturbing lines in the background.
Connect fragments	Connects characters which may be divided e.g. by bad printing in two parts to one segment.
Imprinted	Enables reading of imprinted fonts e.g. if characters appear due to the illumination as white text with black outline (shadow) or vice versa.
Return Punctuation	Activates output of special characters such as full stops or comma.
Return separators	Activates output of special characters like dash.
Groups of characters	Enables possibility to specify the spacing of characters to read. E.G. if characters are always printed in two groups of four characters this can be specified by input of “4 4”. This function should be used, if in several reading attempts in one and the same image, a different string length is read.
Max. deviation from base line	Maximum allowed difference of horizontal position characters on a straight line between first and last character. This function may be used if characters are not printed on a horizontal line.

4.6.3.12.4 Detector OCR, tab classification

Definition of basic settings of characters to read.



Font: Industrial_0-9A-Z

☑ Reference string

Reference string:

No. of alternatives: 3

No. of corrections: 2

Threshold Confidence: 50

Fig. 212: Detector OCR, tab Classification

Parameters	**Functions**
Font	For available fonts see chapter:. Detector OCR, available fonts 0-9 => numbers only 0-9+ => numbers and special characters A-Z => only capital letters A-Z+ => capital letters and special characters No extension => all characters
Ref. String (Checkbox)	Activates verification of contents of the information read. Verification is done on base of regular expressions.
Ref. string	This text or regular expression is used for verification. Here can be entered definite characters, which are compared directly, or with regular expressions to verify the structure of the result read. Characters which look very similar as number or as letter like “8” and “B” can be corrected automatically by use of regular expressions. In the case of the 'Reference string' the detector algorithm purely uses this as a simple check string, after it has 'segmented' and 'classified' the characters, and its only to confirm that the decoded string is as the per the 'Reference string'. and it doesn't influence the classification in any way. In the case of the 'Reference string' that is made up of a 'regular expression', then the 'expression' will try to use known characters to 'best fit' the expression.ie Day 3 letter (MON / TUE / WED / etc) is the segmentation and decode gives M0N rather than MON then the camera software will automatically 'correct' the (number) 0 to become a (letter) O .
Add expression	Opens a list with regular expressions.

Parameters	Functions
Teach ref. string	Reads the code below the Code Reader and copies the contents into Ref. string. Text can be edited afterwards.
No. of alternatives	This command controls how many 'other' near characters are to be considered ie if we are physically looking at a number '8', the near characters could be 6,9,0,B,R,D,O,S and only the closest matching 'x' number of near alternatives will be considered.
No. of corrections	This command controls how many characters with in the string can be changed when using a regular expression in the reference string ie Day 3 letter (MON / TUE / WED / etc) is the segmentation and decode gives the letters W6O rather than WED then with a setting of '2' in this field the camera software will automatically 'correct' the (number) 6 and (letter) O to become a (letter) E and D - If the setting in the field was 1 then the detector would fail.
Threshold	Threshold for good-bad decision: if number of corrections is higher than this threshold, the text will be marked as “not read” (detector result false).

Most important elements of regular expressions

Reference string	Hi	Example for hit
123	String containing 123	01234
\A123	String beginning with 123	1234
123\Z	String ending by 123	0123
\A123\Z	String matching exactly 123	123
[123]	String containing one of the characters	33
[123]{2}	String containing sequence of the characters of length 2	23
[12]|[34]	String containing a character of one of both groups	4

Most important elements of regular expressions:

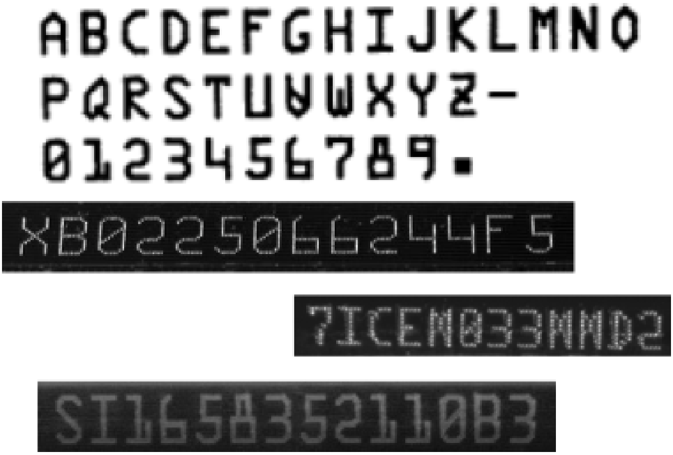
^ or \A	Matches start of string
$ or \Z	Matches end of string (a trailing newline is allowed)
.	Matches any character except newline
[...]	Matches any character listed in the brackets. If the first character is a '^', this matches any character except those in the list. You can use the '-' character as in '

	[A-Z0-9]' to select character ranges. Other characters lose their special meaning in brackets, except '\'.
*	Allows 0 or more repetitions of preceding literal or group
+	Allows 1 or more repetitions
?	Allows 0 or 1 repetitions
{n,m}	Allows n to m repetitions
{n}	Allows exactly n repetitions
|	Separates alternative search expressions

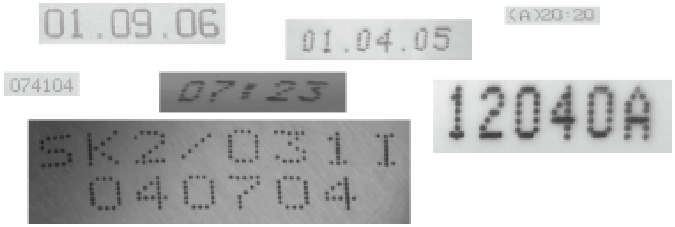
4.6.3.12.4.1 Detector OCR, available fonts

Overview of fonts:

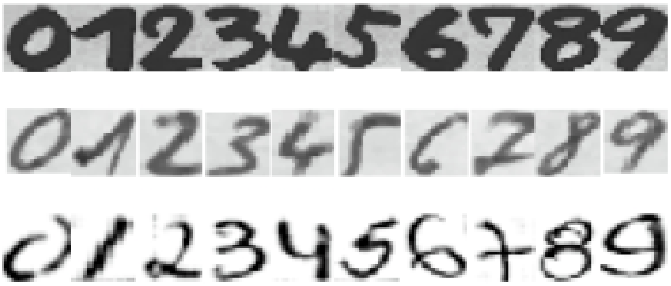
Semi



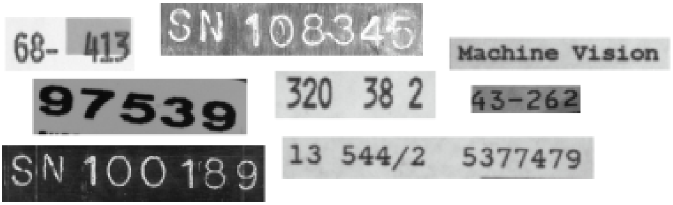
Dot print



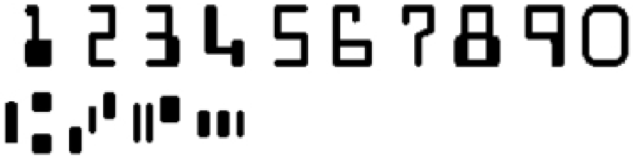
Handwritten



Industrial



MICR



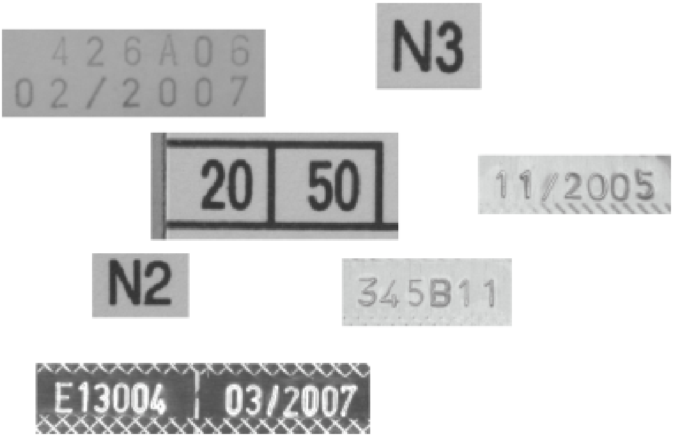
OCRA

0123456789
ABCDEFGHIJKLM
NOPQRSTUVWXYZ
abcdefghijklm
nopqrtsuvwxyz
-?!/\=+<>.#$%&()@*

OCRB

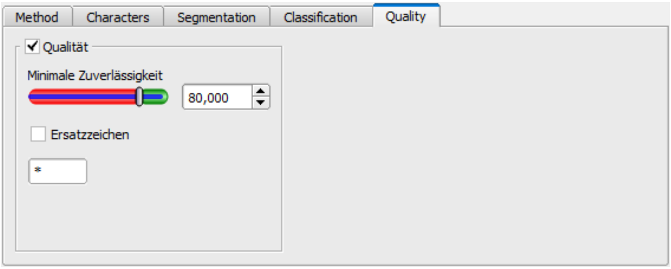
0123456789
ABCDEFGHIJKLM
NOPQRSTUVWXYZ
abcdefghijklm
nopqrtsuvwxyz
-?!/\=+<>.#$%&()@*

Pharma



4.6.3.12.5 Detector OCR, tab quality

Definition of basic settings of characters to read.



☑ Qualität

Minimale Zuverlässigkeit: 80,000

□ Ersatzzeichen

*

Fig. 213: Detector OCR, tab quality

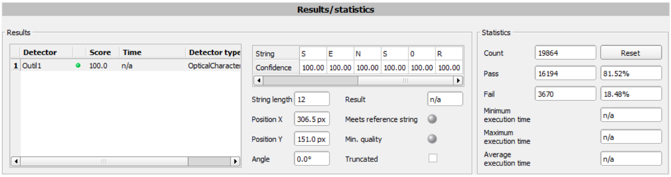
Parameters	Functions
Quality	Quality of each character gets a value of 0 – 100 %. As higher the value, as higher is the confidence to the result. Small values are a sign for a bad reading quality.
Minimum confidence	If minimum confidence was not reached the character is considered to be not read and will be replaced by the replacement character.
Replacement character	Output character for the case that minimum confidence was not reached.

4.6.3.12.6 Result OCR

This function executes the job defined on the PC and the Result statistics window is displayed with Detector list and Evaluation results. Execution times are not updated in this mode, as they are not available from the sensor.

Detailed inspection results from the detector marked in the selection list are displayed in run mode.

In the image window the search- and feature areas and the result bar graphs are displayed – if set up.



	Detector		Score	Time	Detector type
1	Outil1	●	100.0	n/a	OpticalCharacte

String	S	E	N	S	0	R
Confidence	100.00	100.00	100.00	100.00	100.00	100.00

String length: 12

Position X: 306.5 px

Position Y: 151.0 px

Angle: 0.0°

Result: n/a

Meets reference string ●

Min. quality ●

Truncated □

Count	19864	
Pass	16194	81.52%
Fail	3670	18.48%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

Fig. 214: Detector OCR, Result display

The parameters displayed vary according to the type of detector selected:

Parameters	**Functions**
String	Characters read
Confidence	Value from 0-100%, shows how reliably a character has been read
String length	Length of string
Position X	Position X in pixels
Position Y	Position Y in pixels
Angle	Angle compared to horizontal line
Compare result	Is an indication for the quality of a result. If no characters had to be replaced according the reference string, this value is at 100%. The value decreases with rising number of corrections
Meets reference string	Indicates if string meets the reference string.
Compare result	Indicates if minimum quality was reached.
Truncated	Indicates if a part of the string was truncated.

4.6.3.13 Detector Wafer

The wafer detector is a very powerful, easy to use vision tool to detect position and possible chips on wafer or cells during production. It can extremely accurately measure the size and position of the wafer and so can also be used as an aid to pick and place robotic systems.

Note: The tabs “Binarization, Rectangle fit and Miscellaneous” are accessible in expert mode only. Activation via menu bar “Options/Expert mode”.

s. also Document: VISOR® SolarUserManual.pdf in Startmenü/SensoPart /VISOR® vision sensor/

/Documentation/...

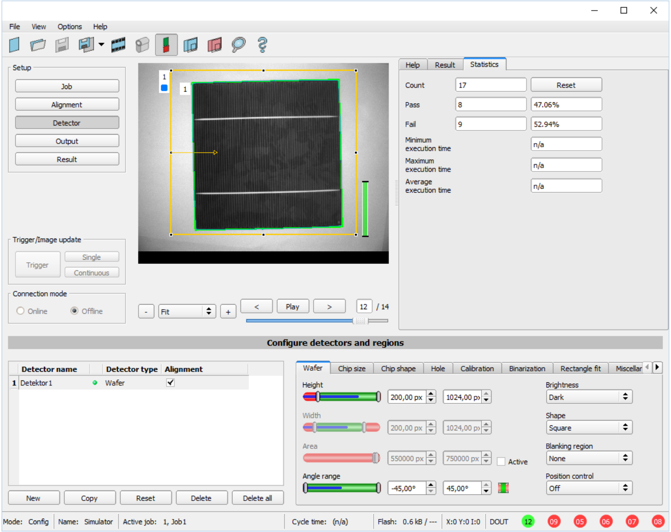
The Wafer detector is designed for:

- Powerful and reliable detection of cracks and chips at straight or wavy outlines
- Flexible adjustment of all measured results e.g. tolerances for wafer size, position, orientation as well as depth and area of defects
- Easy optimization of sensor settings regarding evaluation speed and accuracy (sub pixel method)
- Free cutting method e.g. to get a 5" out of a 6" wafer

- Image pre-processing tools available e.g. distortion correction
- Auto-identification of different types of wafer and cells,

4.6.3.13.1 Detector Wafer, tab Wafer

This detector is designed to check cracks at the edges of the wafer contour and measure geometric parameters such as width, height, position, rotation angle, etc



Count	17	
Pass	8	47.06%
Fail	9	52.94%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

	Detector name		Detector type	Alignment
1	Detektor1	●	Wafer	☑

Height: 200,00 px – 1024,00 px

Width: 200,00 px – 1024,00 px

Area: 550000 px – 750000 px □ Active

Angle range: -45,00° – 45,00°

Brightness: Dark

Shape: Square

Blanking region: None

Position control: Off

Mode: Config | Name: Simulator | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 0.6 kB / --- | X:0 Y:0 I:0 | DOUT

Fig. 215: Detector Wafer, tab Wafer

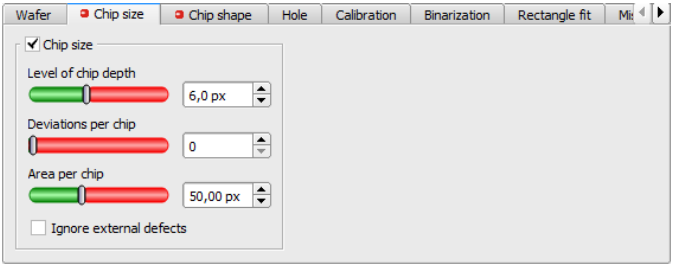
Settings at tab Wafer

Parameter	Functions
Height	Range to accept the height of a wafer.
Width	Range to accept the width of a wafer.
Area	Range to accept the area of a wafer.
Angle range	Range to accept the current value of rotation.
Brightness	Select the brightness of your object comparing to the background.
Shape	Select the type of wafer shape.

Parameter	Functions
Blanking region	This option offers to define up to 12 free programmable rectangle areas. The image information inside these regions of interest will be not inspected.
Position control	To check the position of the center of gravity you can define a rectangle or elliptical area where it has to be in.

For newly generated detectors, all parameters are preset as standard values, suitable for many applications.

4.6.3.13.2 Detector Wafer, Chip size



☑ Chip size

Level of chip depth: 6,0 px

Deviations per chip: 0

Area per chip: 50,00 px

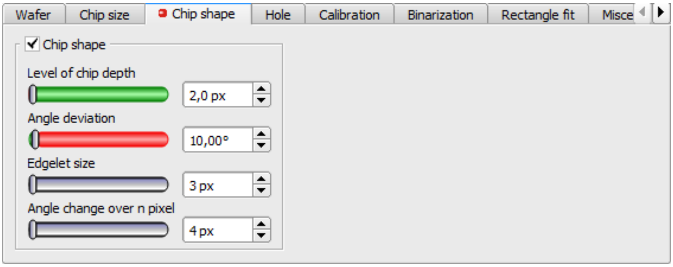
☐ Ignore external defects

Fig. 216: Detector Wafer, Chip size

Settings at tab for Chip size

Parameter	Functions
Chip size	Activate chip size.
Level of chip depth	Out of all detected contour points the algorithm is generating a best-fit of a rectangle box. In the next step all distances between the box and each contour point will be calculated. The threshold defines an incorrect distance.
Deviations per chip	The threshold for the number of faulty distances defines a GOOD/BAD-object.
Area per chip	Definition of a defective area.
Ignore external defects (Expert mode)	This option offers to define 8 free programmable rectangle areas. The image information inside these regions of interest will be not inspected.

4.6.3.13.3 Detector Wafer, Chip shape



☑ Chip shape

Level of chip depth: 2,0 px

Angle deviation: 10,00°

Edgelet size: 3 px

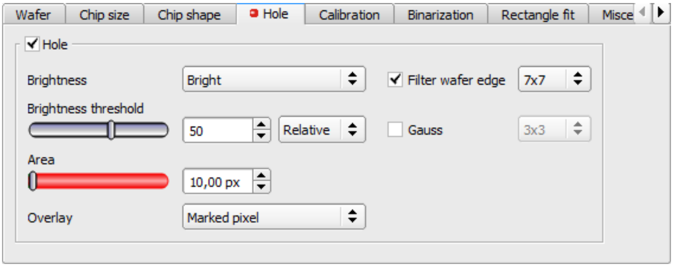
Angle change over n pixel: 4 px

Fig. 217: Detector Wafer, Chip shape

Settings at tab Chip shape

Parameter	Functions
Chip shape	Activate chip shape.
Level of chip depth	If exists a outside defect and includes these marked defect region also a chip defect, all chip defects which have a specified distance below the threshold setting will be still detected.
Angle deviation	A contour point will be detected as a failure if the local angular deviation is higher than the predefined threshold.
Edgelet size (Expert mode)	Based on two neighbouring contour points for each detected contour point a secant (length, position, and angle) will be calculated. Parameter: distance to adjacent edges.
Angle change over n pixel (Expert mode)	Based on ‘n’ neighbouring contour points for each detected contour point the maximum difference out of all n specific rotations (angle of each secant) will be calculated - angular deviation around each contour point.

4.6.3.13.4 Detector wafer, tab hole



☑ Hole

Brightness: Bright

☑ Filter wafer edge: 7x7

Brightness threshold: 50 Relative

□ Gauss: 3x3

Area: 10,00 px

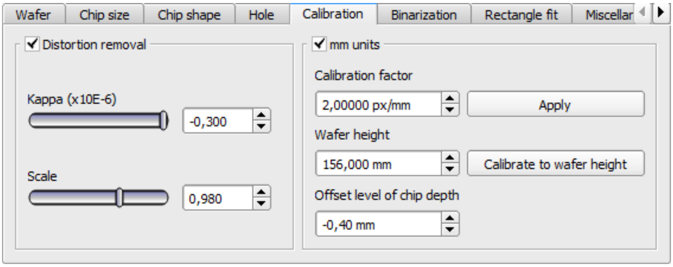
Overlay: Marked pixel

Fig. 218: Detector wafer, tab hole

Settings at tab hole

Parameter	**Functions**
Hole	Activate hole.
Brightness	Select bright or dark object intensity in relation to the brightness of solar wafer.
Brightness threshold/absolute	Define the intensity threshold to detect an faulty object as a fixed gray value
Brightness threshold/relative	Define the intensity threshold to detect a faulty object as an offset in addition to the averaged brightness of each solar wafer.
Area	Smallest area size of a wafer hole (value in pixel by pixel or mm by mm).
Overlay	(De-) activate drawing / marking of all detected objects.
Filter wafer edge (Expert mode)	Extension of dark zones, elimination of light pixels in dark zones, elimination of artefacts, division of bright objects.
Gauss (Expert mode)	Reduction of disturbance, suppression of disturbing details and artefacts, smooth edges.

4.6.3.13.5 Detector Wafer, tab Calibration



☑ Distortion removal

Kappa (x10E-6): -0,300

Scale: 0,980

☑ mm units

Calibration factor: 2,00000 px/mm

Wafer height: 156,000 mm

Offset level of chip depth: -0,40 mm

Fig. 219: Detector Wafer, tab Calibration

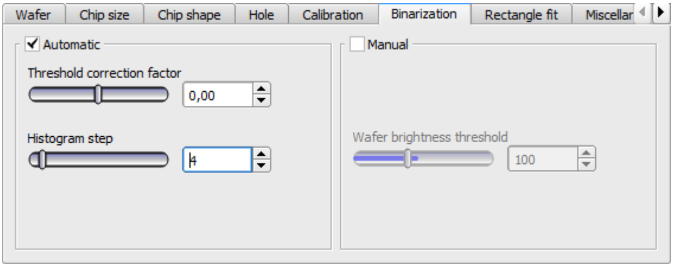
Settings at tab Calibration

Parameter	**Functions**
Distortion removal	Activate distortion removal.

Parameter	Functions
• Kappa (x10E-6)	Constant factor of term to correct the radial distortion.
• Scale	Multiplicative factor to scale the correction.
mm units	Activate mm units.
• Calibration factor	Pixel pro mm; Calibration factor to convert image values into world data.
• Apply	By pressing “Apply” the dimension in other tabs will be automatically calculated and updated.
• Wafer height	Program set automatically the calibration factor dependant on measured wafer height in pixel.
• Calibrate to wafer	Calibration factor is calculated from value “Wafer height”.
• Offset level of chip depth	Correction factor for the measured value of chip depth. The factor is added to the actually measured value.

4.6.3.13.6 Detector Wafer, tab binarization

Note: The tabs “Binarization, Rectangle fit and Miscellaneous” are accessible in expert mode only. Activation via menu bar “Options/Expert mode”.



☑ Automatic

Threshold correction factor: 0,00

Histogram step: 4

□ Manual

Wafer brightness threshold: 100

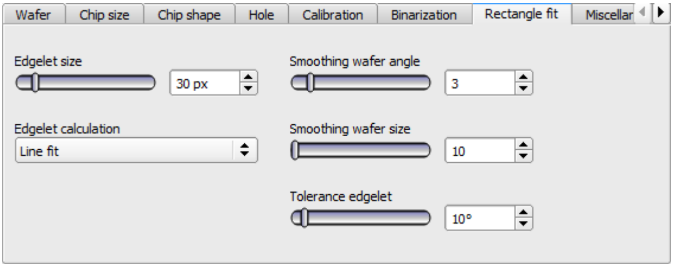
Fig. 220: Detector Wafer, tab binarization

Settings at tab Binarization (Expert mode)

Parameter	Functions
Automatic	Activate automatic binarization.
Threshold correction factor	For each image the intensity threshold will be automatically calculated by evaluating the current brightness of the object and the background. This dynamic threshold can be manually corrected so that the final value will be move closer or further away to the intensity of the background.
Histogram step	Step range of gray values at the histogram.
Manual	Activate manual binarization.
Wafer brightness threshold	Fixed threshold of contrast.

4.6.3.13.7 Detector wafer, tab rectangle fit

Note: The tabs “Binarization, Rectangle fit and Miscellaneous” are accessible in expert mode only. Activation via menu bar “Options/Expert mode”.



Edgelet size: 30 px

Edgelet calculation: Line fit

Smoothing wafer angle: 3

Smoothing wafer size: 10

Tolerance edgelet: 10°

Fig. 221: Detector wafer, tab rectangle fit

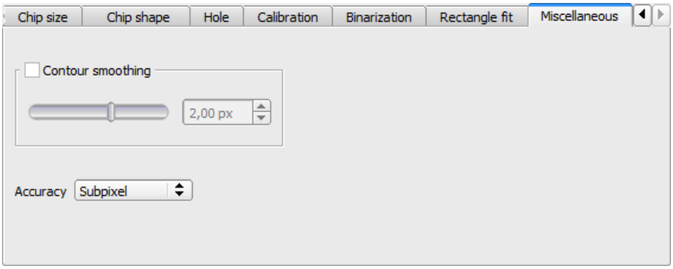
Settings at tab Rectangle fit (Expert mode)

Parameter	Functions
Edgelet size	Step size or number of contour points to calculate a local line along the contour.
Edgelet calculation	Two options are available: Line fit or Secant.
Smoothing wafer angle	Range of detected angles around the maximum of the Gaussian distribution which will be used to calculate the finale angle.
Smoothing wafer size	Range of detected sizes around the maximum of the Gaussian distribution which will be used to calculate the finale wafer dimensions.

Parameter	Functions
Tolerance edgelet	Range of detected angles for each local line which will be used to calculate the wafer orientation in relation to the mean value.

4.6.3.13.8 Detector wafer, tab miscellaneous

Note: The tabs “Binarization, Rectangle fit and Miscellaneous” are accessible in expert mode only. Activation via menu bar “Options/Expert mode”.



□ Contour smoothing 2,00 px

Accuracy Subpixel

Fig. 222: Detector wafer, tab miscellaneous

Settings at tab for miscellaneous (Expert mode)

Parameter	Functions
Contour smoothing (positive or negative)	This option is activating two functions of morphological image processing. If the parameter is set below 0 the OPENING operator will increase the contour failure. At the other side, a value bigger than 0 is CLOSING the wafer cracks.
Accuracy	De-(activate) sub pixel algorithm.

4.6.3.13.9 Settings of thresholds to distinguish False- from Good- parts.

Extract from VISOR® SolarUserManual1WIP 05-14 V.pdf

The VISOR® vision sensors are able to provide you high quality test results, this quality should be kept in mind when you adjust the threshold ranges of your criteria. A typical set-up of the sensor criteria provides the following tasks: all good wafers PASS the tests and all bad wafers are tested as bad wafers and therefore sorted out. To reach this aim, some test wafers (good and bad ones) should be tested under several operating modes with some start criteria, and then the criteria should be re-adjusted until they provide your production needs.

To provide a high reliability of your tests you may tighten the criteria, this means the number of wafer damages identified by the sensor increases and you lower the risk of production downtime caused by broken wafers. If the quality criteria are too high, you get a possibility of false rejected wafers.

To provide a high yield you may lower your criteria. This implies the possibility of getting a PASS result of a bad wafer with all the bad impacts on your production.

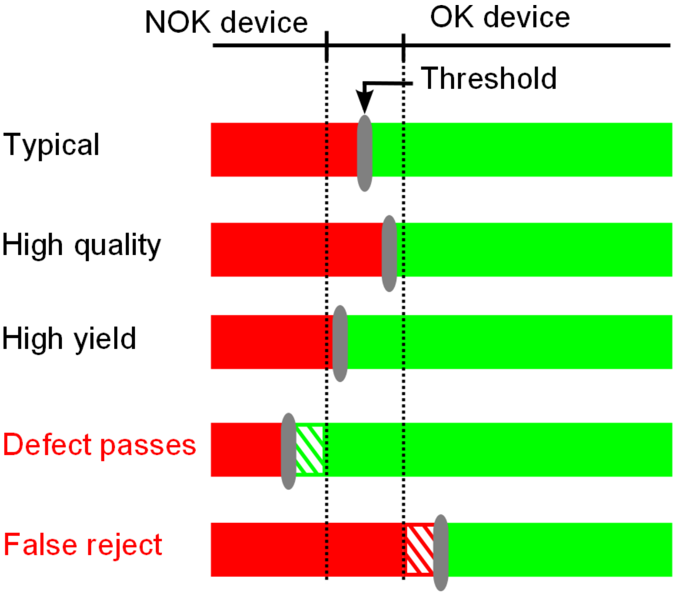


Fig. 223: Settings of threshold

4.6.3.14 Detector Busbar

s. also Document: VISOR® SolarUserManual.pdf in Startmenü/SensoPart /VISOR® vision sensor/

/Documentation/...

Detector busbar, tab busbar (Page 263)

Detector busbar, tab binarization (Page 264)

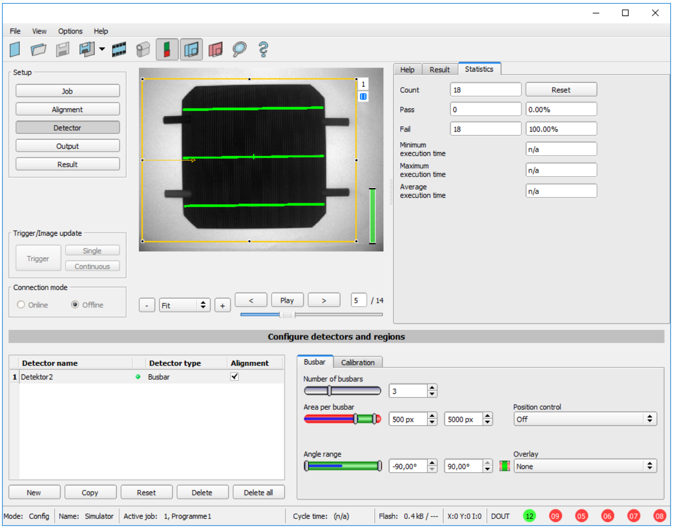
Detector busbar, tab calibration (Page 265)

Detector busbar, tab rectangle fit (Page 265)

Result Busbar (Page 312)

4.6.3.14.1 Detector busbar, tab busbar

Detector to locate and check busbars.



Count: 18

Pass: 0 — 0.00%

Fail: 18 — 100.00%

Minimum execution time: n/a

Maximum execution time: n/a

Average execution time: n/a

	Detector name	Detector type	Alignment
1	Detektor2	Busbar	✓

Number of busbars: 3

Area per busbar: 500 px — 5000 px

Position control: Off

Angle range: -90,00° — 90,00°

Overlay: None

Mode: Config | Name: Simulator | Active job: 1, Programme1 | Cycle time: (n/a) | Flash: 0.4 kB / --- | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 224: Detector busbar, tab busbar

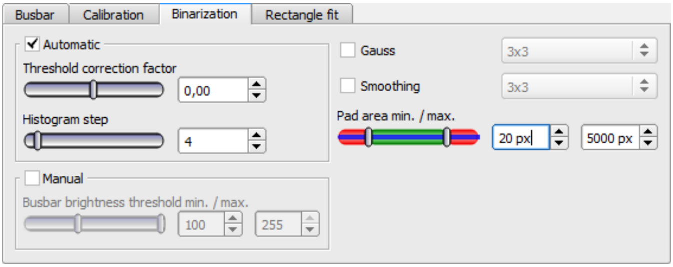
Settings in tab busbar

Parameter	Functions
Number of busbars	Setting for expected number of busbars.
Area per busbar	Range to accept the area of a busbar. If area of busbar is outside specified range of this parameter, computing of all other values is aborted. Area of busbars is calculated from the total number of selected pixels.
Angle range	Range to accept the current value of rotation.
Position control	To check the position of the center of gravity you can define a rectangle or elliptical area where it has to be in.
Overlay	Activate graphical overlays for busbar pixels.

For newly generated detectors, all parameters are preset as standard values, suitable for many applications.

4.6.3.14.2 Detector busbar, tab binarization

Note: The tabs “Binarization and Rectangle fit” are accessible in expert mode only. Activation via menu bar “Options/Expert mode”.



☑ Automatic

Threshold correction factor 0,00

Histogram step 4

□ Manual

Busbar brightness threshold min. / max. 100 255

□ Gauss 3x3

□ Smoothing 3x3

Pad area min. / max. 20 px 5000 px

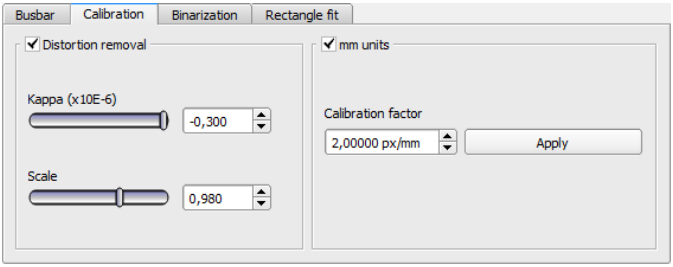
Fig. 225: Detector busbar, tab binarization

Settings in tab binarization

Parameter	**Functions**
Automatic	Activate automatic binarization.
Threshold correction factor	For each image the intensity threshold will be automatically calculated by evaluating the current brightness of the object and the background. These dynamic threshold can be manually corrected so that the final value will be move closer or further away to the intensity of the background.
Histogram step	Step range of gray values at the histogram.
Manual	Activate manual binarization.
Busbar brightness threshold min. max	Min. max. graylevel attributed to busbar.
Smoothing	This option is activating two functions of morphological image processing. If the parameter is setted below 0 the OPENING operator will increase the contour failure. At the other side, a value bigger than 0 is CLOSING the wafer cracks.
Gauss	Reduction of disturbance, suppression of disturbing details and artefacts, smooths edges.
Pad area min. max	Minimum and maximum value for detection of a single pad.

For newly generated detectors, all parameters are preset as standard values, suitable for many applications.

4.6.3.14.3 Detector busbar, tab calibration



☑ Distortion removal

Kappa (x10E-6) -0,300

Scale 0,980

☑ mm units

Calibration factor 2,00000 px/mm

Fig. 226: Detector busbar, tab calibration

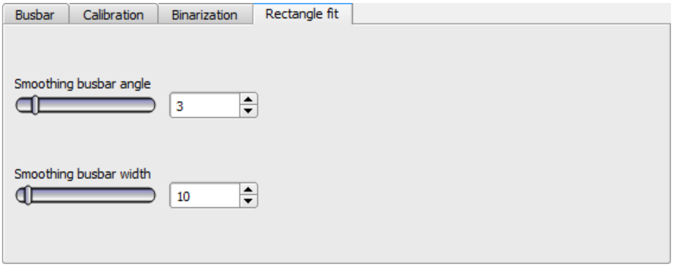
Settings in tab calibration

Parameter	Functions
Distortion removal	Activate distortion removal.
Kappa (x10E-6)	Constant factor of term to correct the radial distortion.
Scale	Multiplicative factor to scale the correction.
mm units	Activate mm units.
Calibration factor	Pixel per mm; Calibration factor to convert image values into world data.
Apply	By pressing “Apply” the dimension in other tabs will be automatically calculated and updated.

For newly generated detectors, all parameters are preset as standard values, suitable for many applications.

4.6.3.14.4 Detector busbar, tab rectangle fit

Note: The tabs “Binarization and Rectangle fit” are accessible in expert mode only. Activation via menu bar “Options/Expert mode”.



Smoothing busbar angle: 3

Smoothing busbar width: 10

Fig. 227: Detector busbar, tab rectangle fit

Settings in tab Rectangle fit (expert mode only)

Parameter	Functions
Smoothing busbar angle	Range of detected angles around the maximum of the gaussian distribution which will be used to calculate the finale angle.
Smoothing busbar width	Range of detected sizes around the maximum of the gaussian distribution which will be used to calculate the finale busbar dimensions.

For newly generated detectors, all parameters are preset as standard values, suitable for many applications.

4.6.3.15 Detector Color value

Output of average color values RGB / HSV / LAB over one of the interfaces.

Tab Color channel (Page 272)

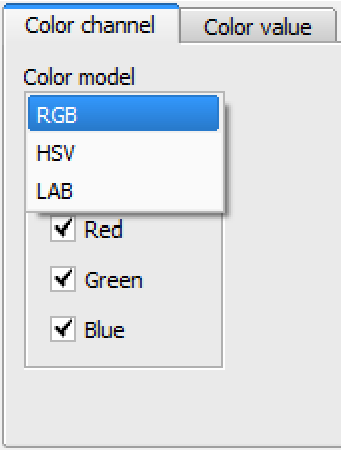
Tab Color value (Page 267)

4.6.3.15.1 Tab Color channel

Selection of Color models (Page 323) or color channel on which the detector should work.

The display of the image depends on the image chip and the selected detector. An image, taken with a color chip contains more information by the color component than a monochrome image. This feature can be used with the color channel selection. By selection of single color channels specific zones can be intensified or weakend.

- Monochrome chip: Display always black/ white
- Color chip + Color detector: Display always colored
- Color chip + Object detector: Monochrome image, display depending on selected color model and color channel



Color model

RGB
HSV
LAB

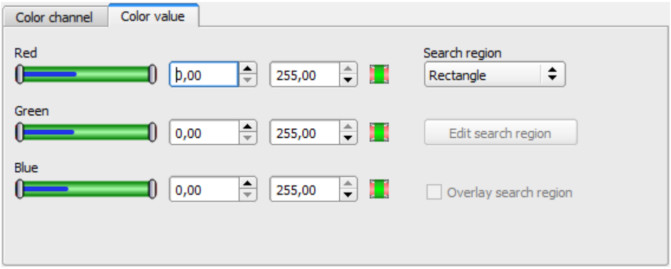
☑ Red
☑ Green
☑ Blue

Fig. 228: Color channel

Parameter	Function
Color model	Color model: RGB, Color model RGB (Page 324), HSV, Color model HSV (Page 325), LAB, Color model LAB (Page 325)
Color channel	One ore more channels can be selected.

4.6.3.15.2 Tab Color value

Output of average color values RGB / HSV / LAB over one of the interfaces.



Red 0,00 255,00

Green 0,00 255,00

Blue 0,00 255,00

Search region: Rectangle

□ Overlay search region

Fig. 229: Color value

Parameter (Color channel dependent from setting of color model)detector	**Function**
Red (Hue / Lightness)1	Threshold for selected channel min. / max.
Green (Saturation / A)	Threshold for selected channel min. / max.
Blue (Value/ B)	Threshold for selected channel min. / max.
Search region	Sets search region as rectangle, as circle or as free shape. If free shape was selected, “Edit search region” gets active.
Edit search region	By edit ROI there can be masked out parts of the search area. The parts which are not relevant for this examination can be painted out like using an erasor. Masks can also be inverted, means that parts which are interesting can be marked.
Overlay search region	Activate overlays for free shape search regions.

Predestinated applications

- Output of calculated color parameters via one of the data interfaces for further processing.

For newly generated detectors, all parameters are present as standard values, suitable for many applications.

4.6.3.16 Detector Color area

Determines percentage of area covered by a color or a range of colors. Depending from area there can be created a good / bad decision.

Tab Color channel (Page 272)

Detector Color area, Color select (Page 269)

Detector color area, tab thresholds (Page 271)

4.6.3.16.1 Tab Color channel

Selection of Color models (Page 323) or color channel on which the detector should work.

The display of the image depends on the image chip and the selected detector. An image, taken with a color chip contains more information by the color component than a monochrome image. This feature can be used with the color channel selection. By selection of single color channels specific zones can be intensified or weakend.

- Monochrome chip: Display always black/ white
- Color chip + Color detector: Display always colored
- Color chip + Object detector: Monochrome image, display depending on selected color model and color channel

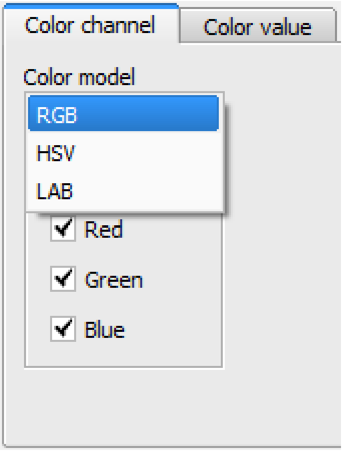
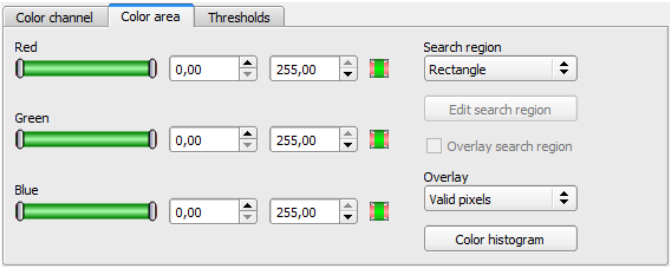


Fig. 230: Color channel

Parameter	Function
Color model	Color model: RGB, Color model RGB (Page 324), HSV, Color model HSV (Page 325), LAB, Color model LAB (Page 325)
Color channel	One ore more channels can be selected.

4.6.3.16.2 Detector Color area, Color select

Determines percentage of area covered by a color or a range of colors. Depending from area there can be created a good / bad decision.



Red: 0,00 – 255,00

Green: 0,00 – 255,00

Blue: 0,00 – 255,00

Search region: Rectangle

□ Overlay search region

Overlay: Valid pixels

Fig. 231: Color area

Parameter (Color channel dependent from setting of color model)detector	**Function**
Red (Hue / Lightness)1	Threshold for selected channel min. / max.
Green (Saturation / A)	Threshold for selected channel min. / max.
Blue (Value/ B)	Threshold for selected channel min. / max.
Search region	Sets search region as rectangle, as circle or as free shape. If free shape was selected, “Edit search region” gets active.
Edit search region	By edit ROI there can be masked out parts of the search area. The parts which are not relevant for this examination can be painted out like using an erasor. Masks can also be inverted, means that parts which are interesting can be marked.
Overlay search region	Activate overlays for free shape search regions.
Overlay	Color marking of pixels inside or outside of specified color range. This is a help during setup to vizualise detector results and to set thresholds more accurate.
Color histogram	Offers possibility to enter the thresholds inside a color histogram.

Predestinated applications:

- Colored object with certain size and variable position in the ROI

For newly generated detectors, all parameters are preset as standard values, suitable for many applications.

4.6.3.16.2.1 Color histogram

Depending from selected color model there are displayed histograms for RGB, HSV or LAB. The histogram shows the distribution of colors in region of interest. By the buttons there can be switched on and off single channels. Limits for color detection can by set by moving small markings below the histogram. The selected range of colors is shown by colored areas. Crossing the limits results in invertion of the selection. If a color can be detected reliable by using only one channel, the other channels have to be set to max./min. limits to avoid disturbing influence to detection.

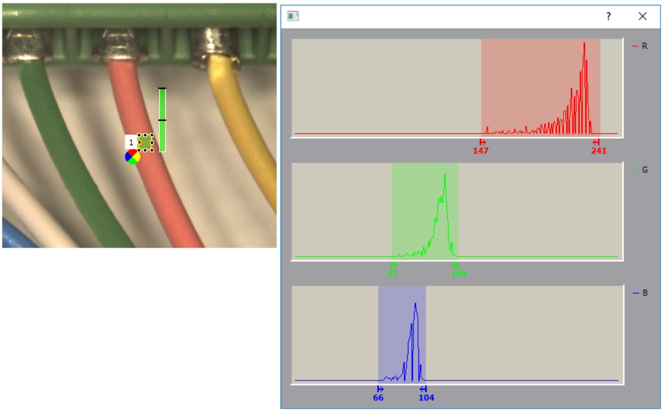
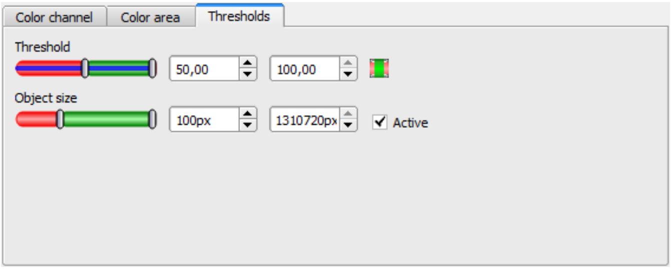


Fig. 232: Color histogram

4.6.3.16.3 Detector color area, tab thresholds

Determines percentage of area covered by a color or a range of colors. Setting of thresholds.



Threshold 50,00 100,00

Object size 100px 1310720px ☑ Active

Fig. 233: color area, tab thresholds

Parameter	Function
Threshold	Threshold for percentage of the area min. / max.
Object size	Min. / Max. object size (connected area)

For newly generated detectors, all parameters are preset as standard values, suitable for many applications.

4.6.3.17 Detector Color list

Compares a color with a list of known colors. Result: number or name of the color clossest to a color in the list. This enables sorting of parts by color.

Tab Color channel (Page 272)

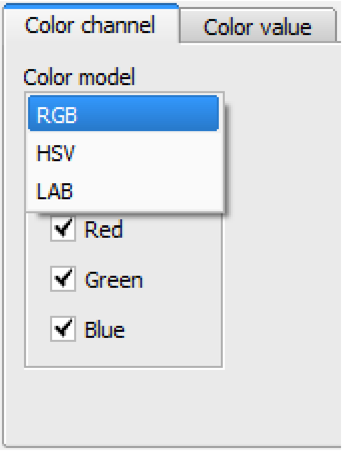
Detector Color list, Color select (Page 273)

4.6.3.17.1 Tab Color channel

Selection of Color models (Page 323) or color channel on which the detector should work.

The display of the image depends on the image chip and the selected detector. An image, taken with a color chip contains more information by the color component than a monochrome image. This feature can be used with the color channel selection. By selection of single color channels specific zones can be intensified or weakend.

- Monochrome chip: Display always black/ white
- Color chip + Color detector: Display always colored
- Color chip + Object detector: Monochrome image, display depending on selected color model and color channel



☑ Red

☑ Green

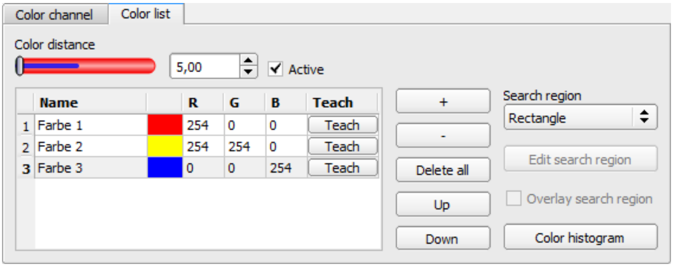
☑ Blue

Fig. 234: Color channel

Parameter	Function
Color model	Color model: RGB, Color model RGB (Page 324), HSV, Color model HSV (Page 325), LAB, Color model LAB (Page 325)
Color channel	One ore more channels can be selected.

4.6.3.17.2 Detector Color list, Color select

Compares a color with a list of known colors. Result: number or name of the color clossest to a color in the list. This enables sorting of parts by color.



Color distance 5,00 ☑ Active

	Name		R	G	B	Teach
1	Farbe 1	🟥	254	0	0	Teach
2	Farbe 2	🟨	254	254	0	Teach
3	Farbe 3	🟦	0	0	254	Teach

Search region: Rectangle

☐ Overlay search region

Fig. 235: Color list

Parameter	Function
Color distance	Distance of current color against taught color. The metric of the color distance depends on the the Color models (Page 323) used, only the selected color channels are considered. *1)
Name	Name of color, can be changed by doubleclick, e.g. red, green, blue...der Farbe, kann per Doppelklick auf den Namen geändert werden, z.B. Rot, Gelb, Blau
Sample color	Ouput of taught color as colored area and in numbers (RGB / HSV / LAB)
Teach	Teach color in active line, if more than one color has to be taught in one and the same image, a small ROI has to be moved to every color.
+	Add new line at end of list.
-	Delete active line.
Delete all	Delete complete list.
Up	Move marked line one line up.
Down	Move marked line one line down.
Search region	Sets search region as rectangle, as circle or as free shape. If free shape was selected, “Edit search region” gets active.
Edit search region	By edit ROI there can be masked out parts of the search area. The parts which are not relevant for this examination can be painted out like using an erasor. Masks can also be inverted, means that parts which are interesting can be marked.
Overlay search region	Activate overlays for free shape search regions.
Overlay	Color marking of pixels inside or outside of specified color range. This is a help during setup to vizualise detector results and to set thresholds more accurate.
Color histogram	Offers possibility to enter the thresholds inside a color histogram.

1*) In the RGB- and LAB- color model the color distance is the euklidean distance.

In the color model LAB the distribution of colors is nearly homogenious over the entire model, that means that color distances of the same value lead to the very equal cognition of color difference over the entire model. That is why we can state that a distance of a value of >= 5 leads to a cognition of a different color in this color model.

Predestinated applicaitons:

- Sorting of colored object via the list index
- Simple control of homogenious colored areas (average of color value over ROI, teach, adjust small color distance (tolerance band) .. that´s it)

For newly generated detectors, all parameters are preset as standard values, suitable for many applications.

4.6.3.17.2.1 Color histogram

Depending from selected color model there are displayed histograms for RGB, HSV or LAB. The histogram shows the distribution of colors in region of interest. By the buttons there can be switched on and off single channels. Limits for color detection can by set by moving small markings below the histogram. The selected range of colors is shown by colored areas. Crossing the limits results in invertion of the selection. If a color can be detected reliable by using only one channel, the other channels have to be set to max./min. limits to avoid disturbing influence to detection.

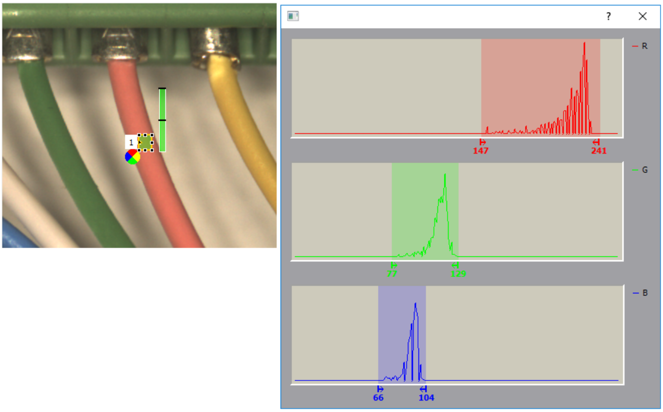


Fig. 236: Color histogram

4.6.4 Output of inspection results

Here you define the assignment and logical connection of the digital signal outputs as well as the interfaces and output data of your VISOR®.

I/O mapping (Page 276)

Output signals (Digital outputs / Logic) (Page 282)

Interfaces (Page 284)

Timing (Page 289)

Telegram, Data output (Page 296)

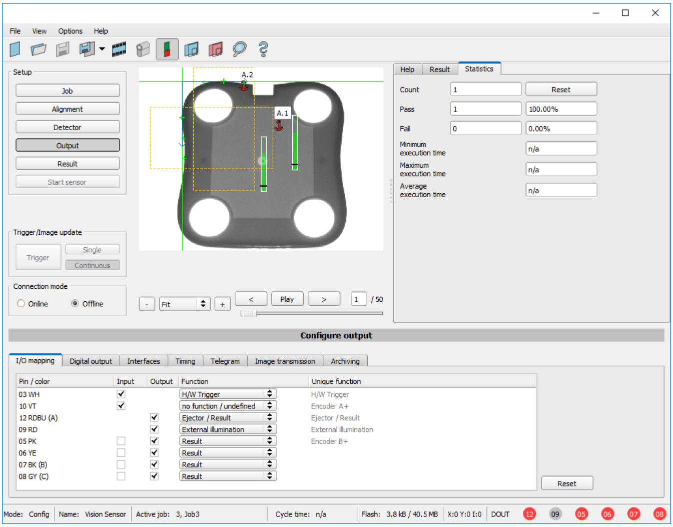
Parameters for image transmission (Page 300)

Parameters Archiving (Page 302)

4.6.4.1 I/O mapping

Here the following settings can be made:

1. Definition, if I/O is used as an input or output (Pin 05 - 08, can be used as input or output)
2. Assignment of functionality to inputs and outputs. In the list-box there can be seen and selected all available functions for this input or output. Some functions can be assigned only to one special input or output (e.g. HW/Trigger).



Count	1	
Pass	1	100.00%
Fail	0	0.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

Pin / color	Input	Output	Function	Unique function
03 WH	☑		H/W Trigger	H/W Trigger
10 VT	☑		no function / undefined	Encoder A+
12 RDBU (A)		☑	Ejector / Result	Ejector / Result
09 RD		☑	External illumination	External illumination
05 PK	□	☑	Result	Encoder B+
06 YE	□	☑	Result	
07 BK (B)	□	☑	Result	
08 GY (C)	□	☑	Result	

Mode: Config | Name: Vision Sensor | Active job: 3, Job3 | Cycle time: n/a | Flash: 3.8 kB / 40.5 MB | X:0 Y:0 I:0

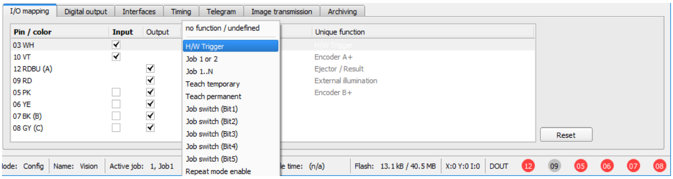
Fig. 237: Output, I/O Mapping

4.6.4.1.1 Functions of inputs

Function	Description
H/W Trigger	Hardware Trigger (only on pin 03 WH available)
Encoder A+	Input for encoder, Track A+ (only on pin 10 VT available)

Function	Description
Encoder B+	Input for encoder, Track B+ (only on pin 05 PK available)
Enable Trigger	Enable or disable trigger signals (input needs a minimum signal length of 2ms before raising trigger signal).
Job 1 or 2	Job change between Job 1 and Job 2, depending on status of this input. Low = Job 1, High = Job 2.
Job 1 … N	Job change by pulses on one input
Teach temporary / permanent	Teaching of all detectors. Rising edge on this input and trigger start teaching. Temporary: storage in RAM, void after reset. Permanent: storage in flash, still valid after reset.
Job switch (BitX), binary coded	Job change by binary bit pattern. Up to 5 inputs can be used to select up to 32 jobs. Bit1 = LSB
Repeat mode enable	Images are captured and evaluated as long as: this input is on high level and none of the following stop criteria is fulfillled: - "Overall job result" = positive (access via Output/Digital output) - "Max. cycle time" is not elapsed (if active) If "Repeat mode enable" is used, this implicitly causes function "Trigger enable" at the same time. That means only if a high signal is at this input, triggers are accepted and executed. see below: Input, Repeat Mode Enable, with Trigger (Page 282)
Multishot trigger (only if Multishot active)	Default setting if Multishot is active, instead of above mentioned H/W Trigger
No function, undefined	no function, not used

Functions which are used already are displayed in gray, because they cannot be used any more. All inputs need a minimum signal length of 2ms.



Pin / color	Input	Output	Unique function
03 WH	☑		H/W Trigger
10 VT	☑		Encoder A+
12 RDBU (A)		☑	Ejector / Result
09 RD		☑	External illumination
05 PK	□	☑	Encoder B+
06 YE	□	☑	
07 BK (B)	□	☑	
08 GY (C)	□	☑	

lode: Config | Name: Vision | Active job: 1, Job1 | le time: (n/a) | Flash: 13.1 kB / 40.5 MB | X:0 Y:0 I:0

Fig. 238: Output, Inputs

4.6.4.1.1.1 Encoder Connection

If both tracks A+ and B+ are used increasing or decreasing counting can be done / forward or backward movement of e.g. conveyor can be recognized. The encoder inputs can work with a frequency of max. 18kHz.

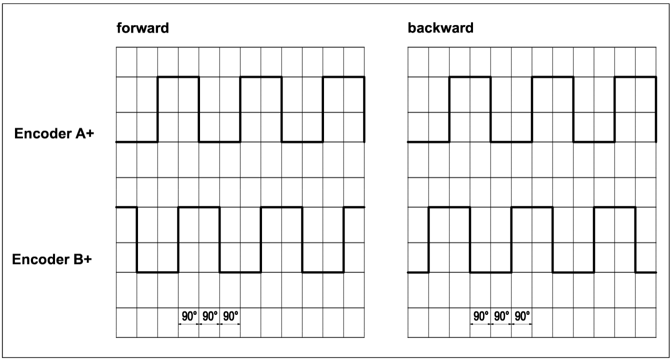
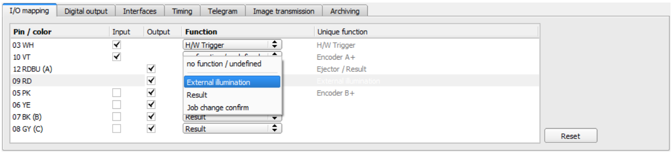


Fig. 239: Encoder connection A+ / B+

4.6.4.1.2 Functions of outputs

Function	Description
Ejector	Dedicated ejector output, maximum load 100mA (all other outputs 50 mA), only on pin 12 RDBU available. (corresponds LED “A”)
Result	Result output, every result output can be covered with a detector result or a logical expression.
Acknowledge job change	Can be used to get a confirmation after successful job change via digital I/O (“Job 1..n” or “Job Pin ‘X’, binary coded”). Rising edge indicates successful job change; high level is reset after 20ms. If job switch was not successful, signals remain low.
External illumination	If this setting is selected (via pin 09 RD available only), a external illumination can be connected / triggered
No function, undefined	no function, not used



Pin / color	Input	Output	Function	Unique function
03 WH	☑		H/W Trigger	H/W Trigger
10 VT	☑		no function / undefined	Encoder A+
12 RDBU (A)		☑		Ejector / Result
09 RD		☑		External illumination
05 PK	□	☑		Encoder B+
06 YE	□	☑		
07 BK (B)	□	☑	Result	
08 GY (C)	□	☑	Result	

Fig. 240: Outputs

There are 2 predefined outputs:

- Ready: indicates, that Sensor is ready to receive a trigger.
- Valid: indicated, that data on outputs are valid.

4.6.4.2 Functions of the programmable, digital inputs:

During operation with process control, the following cases can be carried out via the inputs:

- inactive
- enable/disable
- load Job (binary coded)
- load Job 1 ... n
- teach temporarily
- teach permanently

Description of different cases with a signal diagram.

All following signal diagrams are based on the setting “PNP”.

4.6.4.2.1 Input: “Trigger enable”

For enable (high) or disable (low) of trigger input.

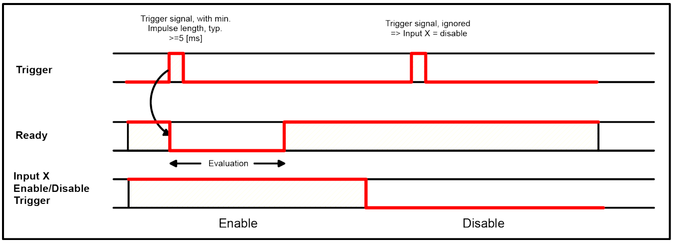

Trigger signal, with min. Impulse length, typ. >=5 [ms]
Trigger signal, ignored => Input X = disable
Trigger
Ready
Evaluation
Input X Enable/Disable Trigger
Enable
Disable

Fig. 241: Input timing, Trigger enable

4.6.4.2.2 Input: Job change binary or by function Job 1 or 2:

Job change binary over up to 5 inputs (Job 1- max. 31):

Possible only if Ready = high. As soon as the binary input signal change Ready is set to low.

Ready remains low until switch-over to the new job is done. If the option “Job change confirm” is used, this signal occurs after the job change, and hereafter “Ready” is set high again. During Job Change via binary inputs there must not be sent any trigger signal. The change of the logic levels of the according inputs must happen at the same time (during maximum 10ms all inputs must have a stable logic level)

Job change by function: Job 1 or 2:

Possible only if Ready = high. At the level change of the according input Ready is set low. Ready remains low till the job change is done. If the option “Job change confirm” is used, this signal occurs after the job change, and hereafter “Ready” is set high again. During Job Change over binary inputs there must not be sent any trigger signal. A high level causes evaluation according to job 2; a low level produces evaluation according to job 1.

Differences between binary switching and Job 1 or 2:

By usage of binary job switch the desired job number must be represented binary coded via the selected inputs. Therefore in this mode to switch between 2 jobs minimum 2 inputs are necessary.

In case of Job change Job 1 or 2 a high level cause’s evaluation according to job 2, a low level produces evaluation according to job 1. In this way with only one input two the switching between two jobs can be done.

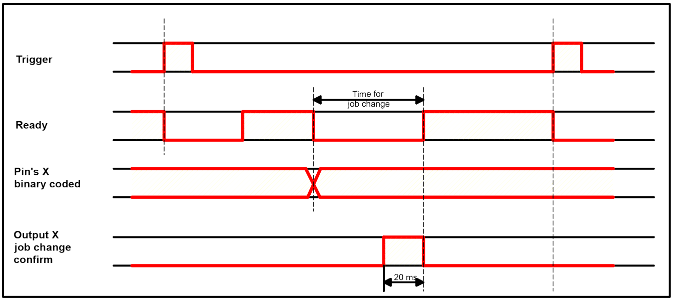


Fig. 242: Input timing, Job change via Binary / Job 1 or 2

4.6.4.2.3 Input: Job 1 ... n

For switching between jobs via impulses. With the first impulse Ready is set to low. Impulses are counted until the first delay of >= 50ms and then switches to the appropriate job. Ready remains low until switch-over to the new job occurs. If the option “Job change confirm” is used, this signal occurs after the job change, and hereafter “Ready” is set high again. During Job Change over binary inputs there must not be sent any trigger signal. Pulse length for job change should be 5 ms pulse and 5 ms delay.

If possible job change should be made by binary coded signals like in fig. 2, this is the faster way.

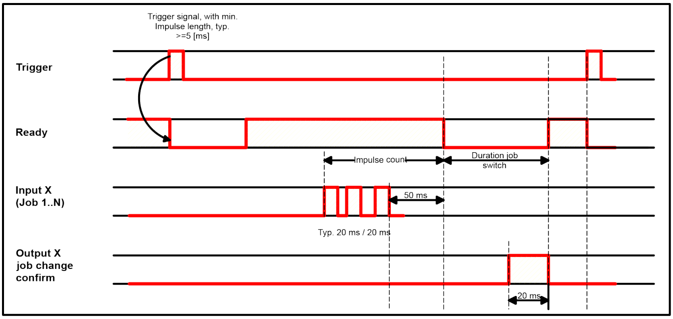


Fig. 243: Input timing, Job 1 ... n

Attention!

At Job switch please take care of the following:

- All Jobs must have the same setting for job change
- All Jobs must be in triggered mode
- Ready signal must be high when trigger sequence starts

4.6.4.2.4 Input: Teach temp. / perm.

For re-teaching samples of all detectors of the current job. A rising edge initiates teaching, during which a high level must exist at least until the next trigger, so that an image of an inspection part can be recorded in the correct position. Ready is set to low and remains low until teaching has been completed. Storage is either temporary (only in RAM), or permanent (in flash) according to the setting.

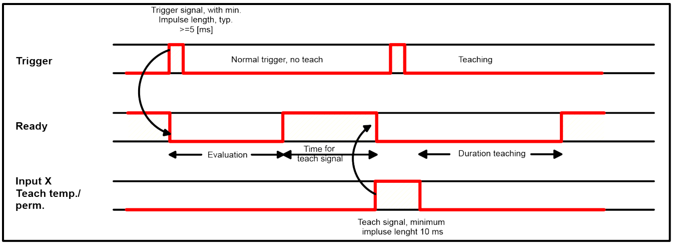

Trigger signal, with min. Impulse length, typ. >=5 [ms]
Trigger
Normal trigger, no teach
Teaching
Ready
Evaluation
Time for teach signal
Duration teaching
Input X Teach temp./ perm.
Teach signal, minimum impluse lenght 10 ms

Fig. 244: Input timing, Teach

Attention!

The functions Job 1 or 2, Job 1 ... n or teach temp. /perm. can only be used in trigger mode

4.6.4.2.5 Input, Repeat Mode Enable, with Trigger

Images are captured and evaluated as long as,
this input is on high level and none of the following stop criteria is fulfillled:

- “Overall job result” = positive (access via Output/Digital output)
- “Max. cycle time” is not elapsed (if active)

If “Repeat mode enable” is used, this implicitly causes function “Trigger enable” at the same time. That means only if a high signal is at this input, triggers are accepted and executed

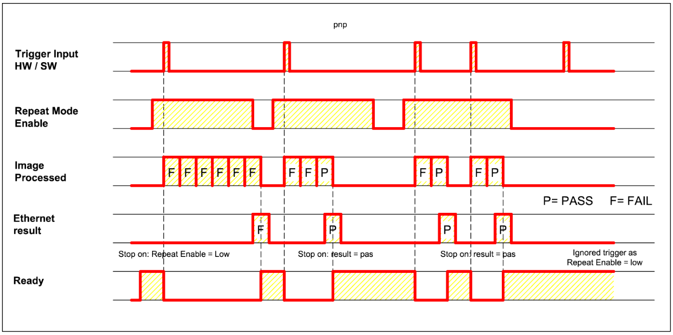


Fig. 245: Input, Repeat Mode Enable, with Trigger

4.6.4.2.6 Input, Repeat Mode Enable, in Freerun

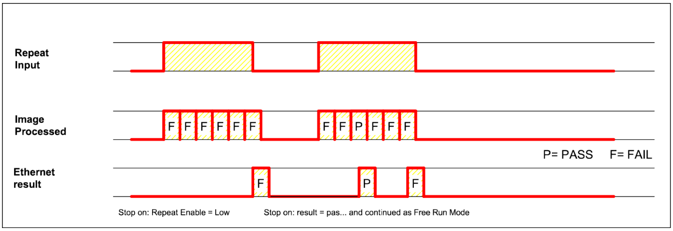
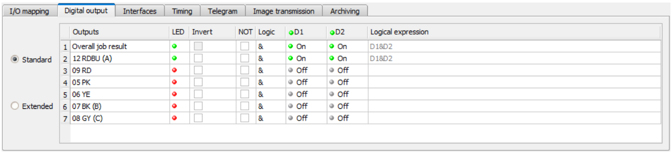


Fig. 246: Input, Repeat Mode Enable, with Trigger

4.6.4.3 Output signals (Digital outputs / Logic)

In this tab, you define the switching behaviour and logical connection of the digital outputs. Number of outputs depends from settings in tab IO mapping. Additionally an IO-extension can be connected over the serial interface.



● Standard

○ Extended

	Outputs	LED	Invert	NOT	Logic	●D1	●D2	Logical expression
1	Overall job result	●	□	□	&	● On	● On	D1&D2
2	12 RDBU (A)	●	□	□	&	● On	● On	D1&D2
3	09 RD	●	□	□	&	● Off	● Off	
4	05 PK	●	□	□	&	● Off	● Off	
5	06 YE	●	□	□	&	● Off	● Off	
6	07 BK (B)	●	□	□	&	● Off	● Off	
7	08 GY (C)	●	□	□	&	● Off	● Off	

Fig. 247: Output, tab digital output

Description of different cases with a signal diagram.

For each pin (output) there are the following possibilities:

Parameter	Function
Overall job result	No physical output, effects recorder, statistics and archiving functions
Invert	Invert total result for this pin (output)
Mode	Standard: combine several detectors by logical expressions like AND (&) / OR (|) / NOT (!) to one logical expression. Advanced: Free edit of logical expression.
NOT	Select: operator NOT (!)
Logic	Select: operator AND (&) / OR (|)
D1 - D...	All active detectors are shown in this list depending from number of detectors. These can be assigned to the listed output. Each detector can be set to on, off and invert.
Logical Expression	Here is shown either the logical expression that was build automatically by using of standard mode or the logical expression can be entered free by using the advanced mode.

Defining logical connection:

Define the logical connection between the inspection results of the individual detectors and the status of the selected output. You have two input possibilities:

4.6.4.3.1 Logical connection – Standard mode

In standard mode, connection of detector inspection results with the selected output must be carried out using the option buttons operator and the checkboxes in the detector selection list. The result is displayed in the logical formulas window (cannot be edited).

Connecting results:

1. Select the logical operator to be used for connecting the detectors in the selection list, from the operator window.

2. Activate the detector in the selection list which is to contribute to the result (tick in the Active column).

By activation the “Inverted” column, you can individually invert the respective detector result.

The entry in the “Result” column alters accordingly.

Examples:

The detector results can only be connected by one logical operation, e.g.:

- (D1&D2&D3) or
- !((!D1)|D2|D3) etc.

Please note: If a detector is assigned to an image acquisition (see “Repeat mode”, chapter Tab Cycle time (Page 117)), its result in the remaining image acquisitions does not affect the logic result.

4.6.4.3.2 Logical connection – Formula mode

In formula mode, connection of detector inspection results with the selected output is defined by the direct input of a logical formula. The operators AND, OR and NOT and round brackets are available for this purpose.

Please use the following characters for the logical operators when editing the formula:

- “&” for AND
- “|” for OR (“AltCtrl” key and “<>” key)
- “!” for NOT

Examples:

Logical expressions of any complexity can be created, e.g.:

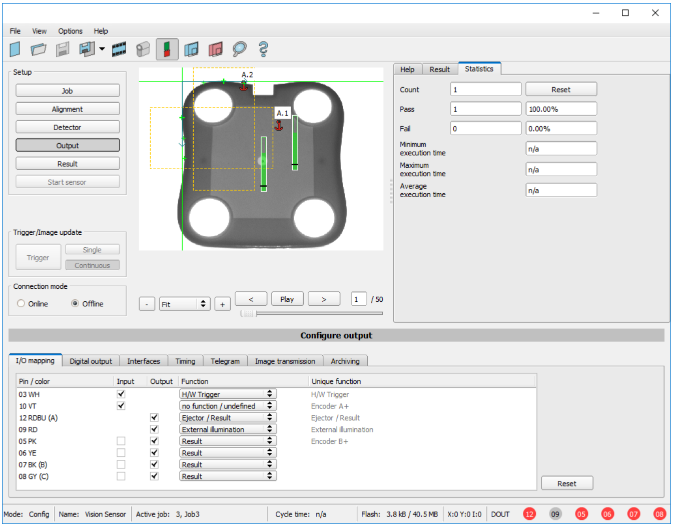
- (D1&D2)|(D3&D4)
- !((D1|D2)&(D3|D4))
- (D1|D2)&(D3|D4)&(D5|D6)

etc.

Please note: If a detector is assigned to an image acquisition (see “Repeat mode”, chapter Tab Cycle time (Page 117)), its result in the remaining image acquisitions is set to logic “0”. The logic result must be adjusted accordingly.

4.6.4.4 Interfaces

In this tab you select and activate the digital inputs/outputs used and the interfaces for data output:



Count 1

Pass 1 100.00%

Fail 0 0.00%

Minimum execution time n/a

Maximum execution time n/a

Average execution time n/a

Connection mode: ○ Online ● Offline

1 / 50

Pin / color	Input	Output	Function	Unique function
03 WH	☑		H/W Trigger	H/W Trigger
10 VT	☑		no function / undefined	Encoder A+
12 RDBU (A)		☑	Ejector / Result	Ejector / Result
09 RD		☑	External illumination	External illumination
05 PK	□	☑	Result	Encoder B+
06 YE	□	☑	Result	
07 BK (B)	□	☑	Result	
08 GY (C)	□	☑	Result	

Mode: Config | Name: Vision Sensor | Active job: 3, Job3 | Cycle time: n/a | Flash: 3.8 kB / 40.5 MB | X:0 Y:0 I:0

Fig. 248: Output, tab Interfaces

Parameter	**Function**
Internal I/O	Selection of I/O-type: PNP or NPN
RS 422 (baud rate)	RS422 for data output with choice of data transmission rate
Ext. (digital I/O)	External inputs and outputs (with I/O and encoder extension module)
Ethernet	Ethernet TCP/IP for data output. Sensor is a socket server. There are used two ports which can be defined by the user. Default is port 2006 (IN) for commands to sensor and port 2005 (OUT) for data transfer. SensoPart offer utilities for explanation of Ethernet communication. They are installed together with this software in utilities directory.
Ethernet/IP	Field bus Ethernet/IP for data output. VISOR® vision sensor, EtherNet/IP, Introduction (Page 410)
PROFINET	Field bus PROFINET for data output, PLC communication. The VISOR® vision sensor starts the PROFINET-Stack as soon as a job with PROFINET is selected. Due to this the cycle time is slightly extended. Switching into a job without PROFINET does not stop the PROFINET- stack. To stop the stack the device must be turned off. Note:

Parameter	Function
	The sensor starts the PROFINET stack as soon as a job with PROFINET is selected. This causes a small slow down of the execution speed. Switching to another job without PROFINET does not stop the stack. Only a new start / reset starts the sensor without execution of the stack. VISOR® vision sensor, PROFINET, Introduction (Page 382)
SensoWeb	Activates the webserver on the VISOR® vision sensor. Similar like in the local installed module “SensoView” now via “SensoWeb” images and result data can be displayed via webbrowser. Following browsers are supported: Microsoft Internet Explorer®ab IE10, Google Chrome® and Mozilla Firefox®. To start SensoWeb: • Activate SensoWeb, at Output/Interfaces/SensoWeb • “Start sensor” (press button in SensoConfig) Open Browser • Type the IP address of the sensor (see SensoFind) into the address field of the browser. Format: “http://Your Sensor IP”, e.g.: “http://192.168.100.100” (default). See also: VISOR® – SensoWeb (Page 286)

For further informations see User manual, chapter “Communication”

Information

The outputs and interfaces can be separately activated or deactivated in the Active column.

Logical outputs:

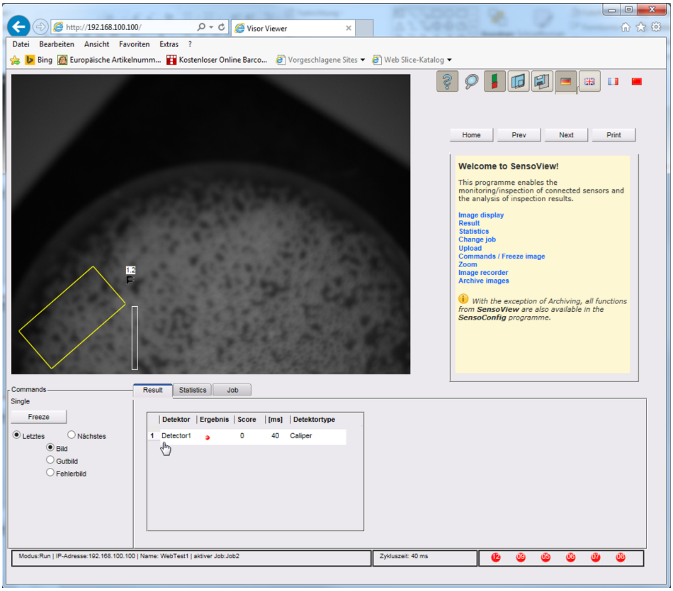
By using the RS422, Ethernet and EtherNet/IP interface additional pure logic outputs can be defined, which just exist logically and can be communicated via one of these interfaces only.

Logical outputs can be assigned to an e.g. detector result or to a logic expression (formula).

4.6.4.4.1 VISOR® – SensoWeb

With this software a connected sensor can be monitored, and results analysed.

From here no new settings on the sensor can be done, it’s a pure display tool to visualize images and results via a web browser.



Welcome to SensoView!

This programme enables the monitoring/inspection of connected sensors and the analysis of inspection results.

Image display
Result
Statistics
Change job
Upload
Commands / Freeze image
Zoom
Image recorder
Archive images

*With the exception of Archiving, all functions from **SensoView** are also available in the **SensoConfig** programme.*

● Letztes ○ Nächstes
● Bild
○ Gutbild
○ Fehlerbild

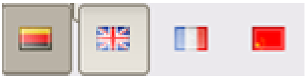
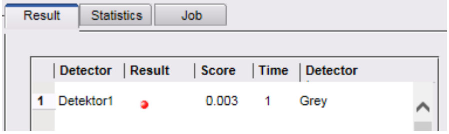
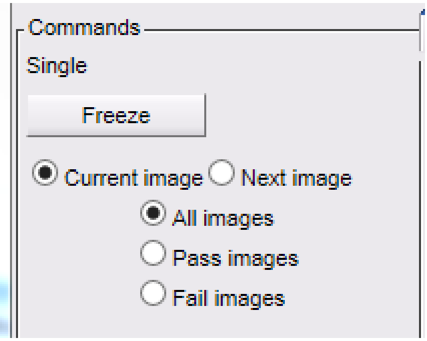
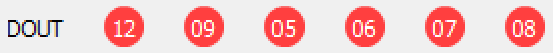
	Detektor	Ergebnis	Score	[ms]	Detektortype
1	Detector1	●	0	40	Caliper

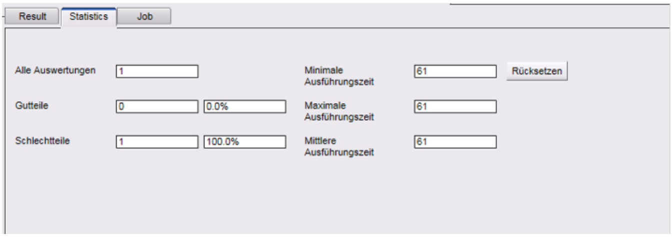
Modus:Run | IP-Adresse:192.168.100.100 | Name: WebTest1 | aktiver Job:Job2

Fig. 249: SensoWeb in the Browser / Results

Functions

?	Switch off help window.
🔍	Zoom of image. A click into the images brings back the original, smaller view.
🟩🟥	On / off of result bargraph.
❐	On / off of overlays.
💾	Store current image as a file.

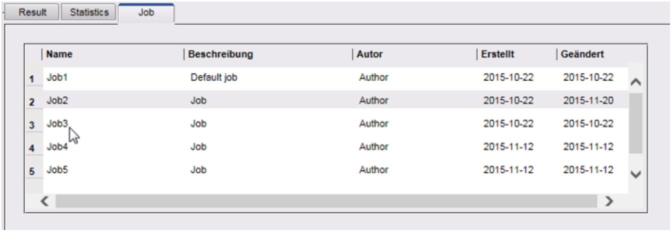
	Switches between languages
 Result Statistics Job | Detector | Result | Score | Time | Detector 1 Detektor1 ● 0.003 1 Grey	Switches between Result, Statistics and the list of Jobs available on the sensor.
 Commands Single Freeze ● Current image ○ Next image ● All images ○ Pass images ○ Fail images	Commands for image control: Possibility to“Freeze” an image. Only the image view is frozen, image capturing and execution is continued.
 DOUT 12 09 05 06 07 08	Status of outputs



Result Statistics Job

Alle Auswertungen	1		Minimale Ausführungszeit	61	Rücksetzen
Gutteile	0	0.0%	Maximale Ausführungszeit	61	
Schlechtteile	1	100.0%	Mittlere Ausführungszeit	61	

Fig. 250: SensoWeb / Statistics



	Name	Beschreibung	Autor	Erstellt	Geändert
1	Job1	Default job	Author	2015-10-22	2015-10-22
2	Job2	Job	Author	2015-10-22	2015-11-20
3	Job3	Job	Author	2015-10-22	2015-10-22
4	Job4	Job	Author	2015-11-12	2015-11-12
5	Job5	Job	Author	2015-11-12	2015-11-12

Fig. 251: SensoWeb / Job

To start SensoWeb:

- Activate SensoWeb, at Output/Interfaces/SensoWeb
- “Start sensor” (press button in SensoConfig)
- Open Browser
- Type the IP address of the sensor (see SensoFind) into the address field of the browser.

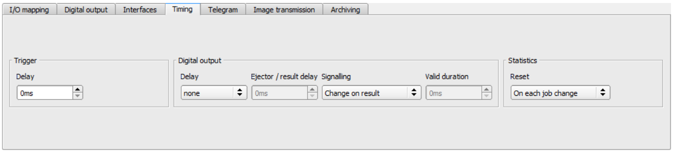
Format: “http://Your Sensor IP”, e.g.: “http://192.168.100.100” (default).

Note:

- The following web browsers are supported: Microsoft Internet Explorer®from IE10, Google Chrome® and Morzilla Firefox®.
- With http://192.168.100.100/zoom.html (IP address of the sensor) a zoomed view is directly accessible.
- Per one VISOR® vision sensor only one browser connection is allowed.

4.6.4.5 Timing

In this tab, you determine the time response of the selected signal output. If encoder was selected the delays are entered in encoder steps. Depending on the settings in the I/O configuration all following time delays are done in ms or in encoder steps.



Trigger
Delay: 0ms

Digital output
Delay: none
Ejector / result delay: 0ms
Signalling: Change on result
Valid duration: 0ms

Statistics
Reset: On each job change

Fig. 252: Output, tab Timing

Parameters	Functions
Trigger delay	Time between trigger and start of image capturing (in ms or encoder pulses). Max. time / no. of steps, is 3000 ms / encoder pulses.

Parameters	Functions
	In case of use of: • H/W Trigger (digital input): this delay is effective. • Trigger (via Ethernet, PROFINET): this delay is not effective (image is captured on trigger directly)
Digital outputs	All outputs can be delayed or only the ejector output.
Ejector / result delay	Time between trigger and appearance of result level at the signal outputs (in ms or encoder pulses). Between trigger and ejector maximum 20 parts are allowed (buffer size). Max. time / no. of steps, is 3000 ms / encoder pulses. In case of use of: • H/W Trigger (digital input): this delay is effective and starts with the trigger. • Trigger (via Ethernet, PROFINET): this delay is effective, but starts only after image is processed (not with the trigger!)
Reset signal	Determines, how to reset outputs.
Duration of result	Duration of result signal in ms

Attention:

At Job Change and change from Run- to Config Mode outputs will get the following states: Buffer of delayed outputs will be deleted.

Digital outputs:

Will be reset to default at change from "Run" to "Config". Defaults are set by flag "Invert" in output tab. "Invert" inverts the default setting and also the result.

Reset of digital outputs:

The reset of the result outputs can happen depending on different settings 7 events. This are:

- "Change on result" (default).
  The output changes its level according to the logical result when the next logical result is generated and valid. Typical use at controlling switch points e.g. in sorting applications.
- "Change on trigger"
  The output is set to "inactive" (in operating mode PNP = low) with the next trigger. Typical use at operation with a PLC.
- "Valid duration"
  The output changes back to inactive after the "Valid" duration time setting here in ms. typical use with e.g. pneumatic ejectors.

S. SensoConfig/Output/Timing/Signalling

READY AND VALID

- If Ready = high: Ready for next image / evaluation.
- If Valid = high: Results are valid at the outputs.

PNP or NPN operating mode.

All the described examples are in the operation mode “PNP”. If the setting “NPN” is used, the examples are valid, but with inverted signal levels.

S. SensoConfig/Output/Interfaces/Internal I/O

4.6.4.5.1 The following cases for output timing are available:

4.6.4.5.1.1 Normal trigger, no delays:

Sequence: (Signalling here: Change in result)

- Rising edge at Trigger input (Pin03 WH)
- Consequence of Trigger = high: Ready = low, and Valid = low
- After the VISOR® has evaluated the image and the results are valid the defined outputs change to the according logical states. Ready and Valid are set to high again (ready for next task, outputs valid).

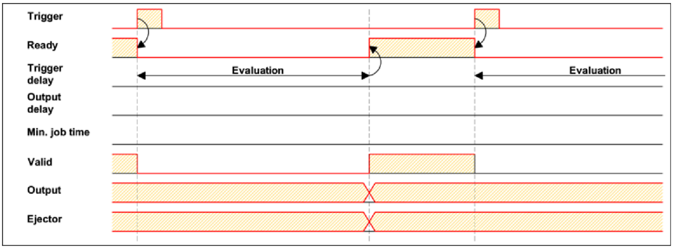


Fig. 253: Output timing, standard sequence at normal trigger

4.6.4.5.1.2 Trigger delay active

(Trigger delay concerns hardware trigger only)

This setting is used to delay the image capturing / start of evaluation against the real physical trigger, which was e.g. caused by a light barrier or by the PLC. With this function the fine tuning of the trigger point in time can be done without any change in mechanics or PLC programming.

Sequence:

Image is taken after the trigger delay time is elapsed. The cycle time is trigger delay time + evaluation time.

s. SensoConfig/Output/Timing/Trigger/Delay

- Rising edge at Trigger input (Pin03 WH)
- Consequence of Trigger = high: Ready = low, Valid = low, all defined result outputs = low (Signalling = Change on trigger)
- Before the image for evaluation is taken, the adjusted Trigger delay time elapses.
- Now the evaluation starts. As soon as the results are valid the outputs change to the according logical levels. Ready and Valid are set to high again (ready for next task, outputs valid).

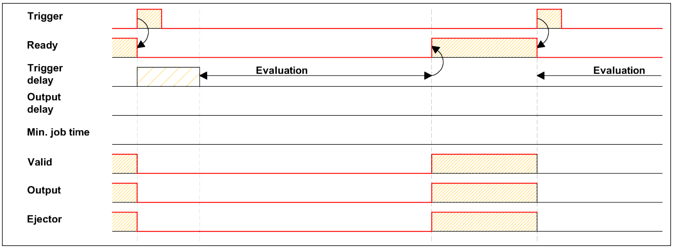


Fig. 254: Output timing, and Trigger delay

4.6.4.5.1.3 Trigger delay + Result delay (here: Ejector only):

(Trigger delay concerns hardware trigger only)

The result delay (if for all outputs or ejector only) is used to fine tune the ejector point in time, independent from evaluation time, as especially the evaluation time can have slight variations.

Sequence:

Image is taken after the trigger delay time is elapsed. Furthermore the Result delay is active, but in this example just for the ejector output (pin 12 RDBU)

For all defined result outputs, except the ejector output the cycle time is: Trigger delay time + evaluation time.

For the ejector output the cycle time is: Result delay only! (Counted from trigger, only make sense if longer than summation of above mentioned times!) S. SensoConfig/Output/Timing/Output/Delay.

- Rising edge at Trigger input (Pin03 WH)
- Consequence of Trigger = high: Ready = low, Valid = low, all defined result outputs = low. Except Ejector, as for this a fix result delay is defined.
- Before the image for evaluation is taken, the adjusted Trigger delay time elapses.

- Now the evaluation starts. As soon as the results are valid the outputs change to the according logical levels. Ready and Valid are set to high again (ready for next task, outputs valid).
- In this operation mode the Ejector output only is set after the Result delay is elapsed. In this example the Ejector output is also used with Result duration, therefore it´s reset after the Result duration time is elapsed.

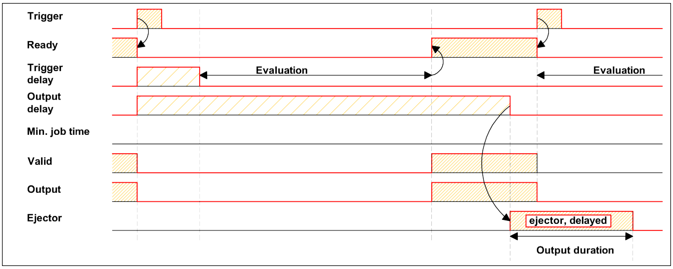


Fig. 255: Output timing, Result delay, ejector

4.6.4.5.1.4 Trigger delay + Result delay (here: all outputs):

(Trigger delay concerns hardware trigger only)

The result delay (if for all outputs or for ejector only) is used to fine tune the ejector point in time, independent from the evaluation time, as the evaluation time of the “job” can have slight variations.

Sequence:

Image is taken after the trigger delay time is elapsed. Furthermore the Result delay is active, in this example to ALL outputs.

For all defined outputs, the cycle time is: Result delay only! (Counted from trigger, only make sense if longer than summation of Trigger delay + Evaluation time) s. SensoConfig/Output/Timing/Output/Delay.

- Rising edge at Trigger input (Pin03 WH)
- Consequence of Trigger = high: Ready = low, Valid = low.
- Before the image for evaluation is taken, the adjusted Trigger delay time elapses.
- Now the evaluation starts. As soon as the results are valid, only the Ready signal is now directly set to high again (ready for next evaluation). Now the result delay time must elapse. After this has happened all defined outputs change to the according logical levels. Now also the Valid signal is reset to high level (Valid = high: results / outputs valid. Signalling = Change on result).

In this operation mode the Ready signal only is reset to high level after Trigger delay + Evaluation time is elapsed. (Ready = high: Ready for next evaluation). This make sense as the VISOR®

independent from the later setting of the other outputs, is now already available for the next evaluation task.

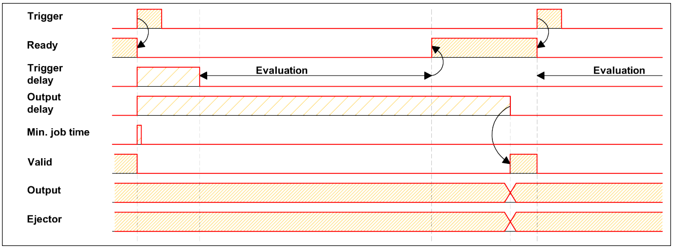


Fig. 256: Figure 142; Output timing, Result delay for all outputs.

4.6.4.5.1.5 Result duration active. (Here e.g. all outputs):

This timing setting is used to achieve a pulse at an output of defined length, for e.g. control of a pneumatic ejector in case of a bad part.

All defined result outputs are reset to low level (inactive in PNP operation) after the Result duration in ms is elapsed.

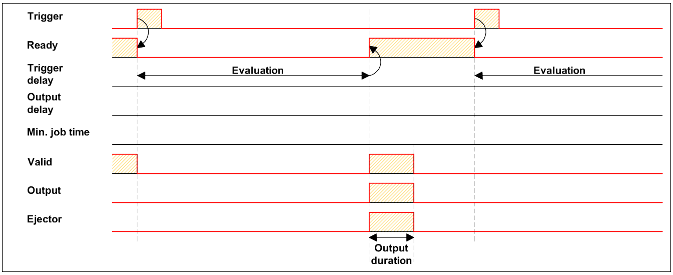


Fig. 257: Output timing, Result duration

4.6.4.5.1.6 Cycle time (Min, Max) active:

(Here: Signalling: Change on Trigger)

Parameter control for the minimum and maximum time for a job. Minimum job time blocks trigger signals which are coming in before the minimum job time was reached. (If during the Min Cycle time a further trigger is coming in it is ignored)

Maximum job time interrupts a job after a defined time. Job result after a timeout is “not o.k.” Maximum job time should be selected higher than the time demand for one execution.

The Cycle time measures the time from Trigger till the setting of the outputs. If the cycle time should be limited, e.g. because of a machine cycle must not be exceeded, it should be set to an appropriate value. The result of all till this point of time not completely processed detectors is set to false. By selecting the Max. Cycle time please consider that this may not be 100% exact, as depending on the currently processed detector it’s possible that there will elapse a few more milliseconds the function can be stopped. It's recommended to check this possible exceeding of the Cycle time in real operation and to decrease the value for the setting according to this offset.

Sequence:

All outputs and the signal “Valid” (Outputs valid) are set directly after evaluation. But the signal “Ready” (Ready for next evaluation) is set not until the Min Cycle time is elapsed. Therefore only from this point in time the next trigger will be accepted.

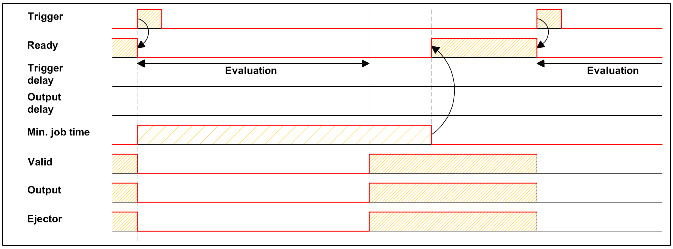


Fig. 258: Output timing, Cycle time

4.6.4.5.1.7 Multiple Result delay for Ejector

This mode of operation is used, if between trigger / evaluation for part A and it´s ejection is so much time / distance, that the VISOR® already has to check n (up to 20 parts possible) further parts which also has to be ejected later.

(Only available in mode: SensoConfig/Output/Timing/Delay: “Ejector only / Ejector- / result delay”

Here: Signalling = Result duration (alternatively also “Change on result”)

This function is limited on 20 parts between trigger and ejector.

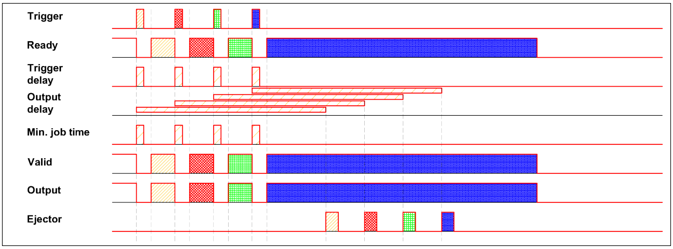


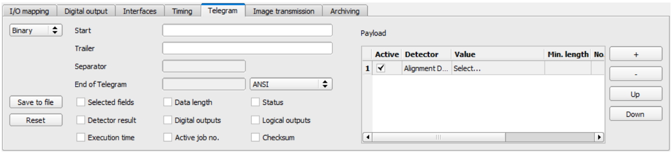
Fig. 259: Output timing, Multiple Result delay, ejector

Examples:

In operation with an I/O Box preferably use the timing functions of the I/O Box.

4.6.4.6 Telegram, Data output

Configuration of data output via serial interfaces RS 422 and Ethernet as well as for archiving in .csv. files. Here all settings can be done, which result data of the VISOR® vision sensor should be transferred via the before selected interface.



Binary

Start

Trailer

Separator

End of Telegram ANSI

□ Selected fields □ Data length □ Status

□ Detector result □ Digital outputs □ Logical outputs

□ Execution time □ Active job no. □ Checksum

Payload

	Active	Detector	Value	Min. length	No.
1	☑	Alignment D...	Select...		

Fig. 260: Output, tab Telegram

Parameters	**Functions**
Binary / ASCII	Output data in Binary- (Hex) or ASCII- format.
Save to file	Exportation of file format with current results as .csv. Detailed file format of the free defined output string as .csv file with: Byte position (start position in string), Data type, Field name, Detector name, Value, Length (in Byte), Detector number and Detector type.
Reset	Reset of all parameters in this tab

Standard contents of protocol

Often required standard contents can be added to the output string by simply filling them in, or activation via the checkbox.

Start	Characters which are inserted at the beginning of the payload data sting (Binary or ASCII)
Trailer	Characters which are inserted at the end of the payload data sting (Binary or ASCII)
Separator	Characters which are inserted behind each payload value (ASCII only)
End of telegram	Characters which are sent at the end of a response to a PC or PLC (Reaction to a command, not with payload data, in ASCII mode only, output selectable in ANSI or Hexa Decimal)
Selected fields	Shows which of the following checkboxes are activated.

.... further standard content, like e.g. “Selected fields, Data length”	to data string: “Payload” Sequence: Selected fields, Data length, Status, Detector result, Digital outputs, Logical outputs, Execution time, Active job no., Checksum

Detector-specific individual results

First create a new entry by activating the “+” button.

Function of buttons

- “+”: Insert new entry
- “-”: Delete marked entry
- “Up”, “Down”: Displace marked entry

You can add detector-specific individual results to the data telegram in the required flexible order via the selection list: (adding new values via button “+”)

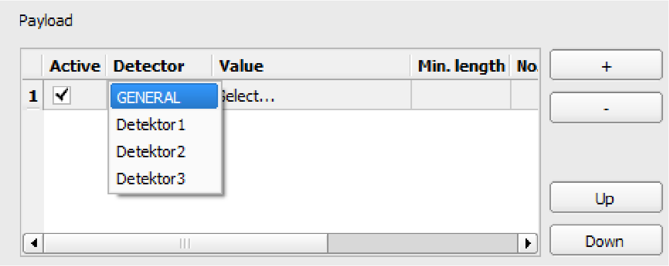


Fig. 261: Output, Detector specific payload

Column	Function
Active	Activates/deactivates the marked output value
Detector	Detector name (select from drop-down list)
Value	Available detector results (select from drop-down menu)
Min. length	Define the minimum length of the Value box; if the actual length is smaller than that specified, the box is filled with spaces (ASCII) or zeros (binary)
No. of results	BLOB only! Number of results of a BLOB detector which found several objects. Example: feature “area” was selected and 10 BLOBs have been found, here up to 10 of these area values can be transmitted. All available output data see: Serial Communication ASCII (Page 458), Serial communication BINARY (Page 518), chapter: Data Output in

Column	Function
	ASCII/Binary

4.6.4.6.1 Possibilities of data output of VISOR® (see also chapter: Communication)

4.6.4.6.1.1 (Ethernet-) port 2005 / RS422

Numerical data, which has been defined under Output/Telegram, now can be transferred in ASCII- or Binary- format.

Ethernet: The sensor here is the (socket-) “server” and serves the Data via a “server-socket” interface. This is basically a “programming interface”. To read or process the Data a “socket client” (PC, PLC, ...) must establish a (socket-) connection (active) to the sensor.

4.6.4.6.1.2 PC-Archiving (SensoView)

Here images and numeric result data (in .csv. format) can be stored by “SensoView” into a folder on the PC.

The configuration (folder, …) of this archiving function is done via “SensoView” (Menu: File/Result archiving, this is a pure PC- function).

4.6.4.6.1.3 Sensor- archiving (ftp, smb)

With this function images and numeric result data (in .csv format) can be stored actively by the sensor via ftp/smb. This kind of archiving is configured under “Job/Archiving”, in this case:

a. With “ftp” used: the senor is a “ftp client” and “writes” the data to a “ftp server” folder on a drive which is available in the network. With Job/Start the sensor connects to the ftp-Server.
b. With “smb” used: the sensor “writes” the data direct in a folder in a network. With Job/Start the sensor connects/mounts with this folder.

4.6.4.6.1.4 Ram disk (in the sensor)

In the sensor the last image as well as the numeric data of the last evaluation, which has been configured under Output/Telegram, are stored (in a .csv file) in a Ram disc- folder under. “/tmp/results/”.

This function is activated under “Job/Image transmission”. To access this data an ftp- connection must be established actively to the sensor. Therefore an ftp client is necessary.

Attention

- The format of the .csv files is always the same (ftp, smb, ram-disk, SensoView).
- The data are stored readable (by default separated by comma) into the .csv file.
- Only payload data which has been defined under (Output/Telegram) are transferred.

4.6.4.6.2 Communication settings

Communication	**TCP / IP**	**RS422 / RS232**	**EtherNet/IP**	**PROFINET**
Telegram format	ASCII / Binary	ASCII / Binary	Binary	Binary

Protocol settings

Parameters	**Functions**
Binary / ASCII	Output data in Binary- (Hex) or ASCII- format.
Save to file	Exportation of file format with current results as .csv
Reset	Reset of all parameters in this tab

Basics for establishing of a connection:

VISOR® is always tcp/ip (socket-) server.

VISOR® vision sensor opens always two (socket-) communication ports (default: 2005 + 2006).

- 2005 = Data port for sending of numerical results.
- 2006 = Command port for receiving of commands.

At a time only one (socket-) client (PC or PLC) can be connected to a port.

Recommendations:

Existing socket connections have only to be reconnected, if an error occurred (on ports 2005 + 2006)

(e.g.: PLC or client in stop mode or error mode, etc.). During normal operation there is no need to reconnect existing connections.

Ethernet data handling: Especially if several VISOR® vision sensor are used Ethernet should be preferred.

Please see also installed help:

...:\Program files\SensoPart \VISOR® vision sensors\Utilities\Ethernet

Commands to sensor in ASCII

4.6.4.7 Parameters for image transmission

Image transmission and/or the image recorder and the Ram disc can be activated in the Image transmission tab.

Set image sharpness with the focus setting screw on the back of the VISOR®.

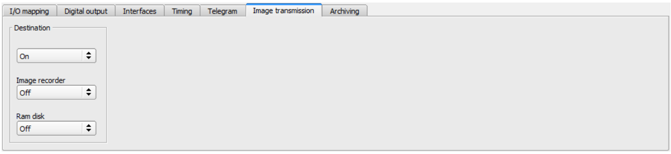
The symbol “exclamation mark” inside life picture means, that image display / transfer on PC is slower than image processing on VISOR®. Not all images are transferred and displayed on the PC. This may cause lost images during archiving. If this symbol occurs often, PC-programs running in background should be closed in order to improve PC performance.

Parameters	Functions and setting possibilities
SensoView	Transmission of images to SensoView can be switch on and off (Off increases the speed of VISOR®). - Off: no images are transmitted to SensoView - On: images are transmitted. Pre-processing filters do not effect the images. (But, if activated, Arrangement filters do effect the transmitted images!) - On (with Pre-processing): Images are transmitted, all activated Pre-processing and Arrangement filter do effect the image.
Image recorder	Storage of max. 10 images in the sensor's internal ring buffer. Setting possibilities via pop-up menu: Off, Any, Pass, Fail.
Ram disk	Storage of last image in ram memory, this image can be taken by a FTP-client. Ram disk Settings: Off, Any, Pass, Fail. The image is stored under name “image.bmp” in folder /tmp/results/ . Parameters for FTP-client: user “user”, password “user” Example Windows Console: Microsoft Windows XP [Version 5.1.2600] (C) Copyright 1985-2001 Microsoft Corp. C:\>ftp 192.168.100.100 Verbindung mit 192.168.100.100 wurde hergestellt. 220 Welcome to VISOR® ftp-server! Benutzer (192.168.100.100:(none)): user 331 Please specify the password. Kennwort: user 230 Login successful. ftp> cd /tmp/results 250 Directory successfully changed. ftp> get image.bmp 200 PORT command successful. Consider using PASV. 150 Opening BINARY mode data connection for image.bmp (354358 bytes). 226 File send OK. FTP: 64d Bytes empfangen in 0,23Sekunden 1514,35KB/s ftp> Image is now in drive C of executing PC.

Parameters	Functions and setting possibilities
	If activated, results can be also received in the same way via the file "results.csv" (all defined data in "Output/Telegram", with divider ";".

Different types of archiving images

Access	Description	Max. number of images	Image filter	Drawings
Image recorder in VISOR® (Ram)	Images stored in run-mode on VISOR® can be transferred by SensoConfig or SensoView to a PC.	10	like predefined in settings "Filter"	no
SensoView archiving / SensoConfig save image	Images transferred to SensoView can be stored on hard disc of PC.	unlimited (Limit is size of hard disc in PC)	like predefined in settings "Filter"	selectable yes / no
Saving of filmstrips in SensoConfig	Current images from filmstrip can be saved as filmstrip (*.flm) or as bitmap (*.bmp) on hard disc of PC.	50	without filtering	no
Last image in VISOR® (Ram Disk)	Last image is stored in ram disk of VISOR® and can be taken by FTP from directory /tamp/results.	1	without filtering	no
Archiving of images via FTP or SMB	Archiving of images via FTP or SMB	unlimited (Limit is size of hard disc in PC)	selectable with / without filtering	no
Get Image Request	Last image from VISOR® by using GetImage command in a program of a PLC or PC.	unlimited (Limit is size of hard disc in PC)	like predefined in settings "Filter"	no



Destination

On

Image recorder

Off

Ram disk

Off

Fig. 262: Tab Output / Image transmission

4.6.4.8 Parameters Archiving

In tab Archiving the archiving of data can be defined.

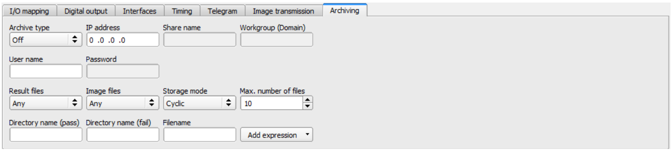
Parameters	**Functions**
Archive type	Off: No archiving, FTP: Archiving to FTP server, SMB: Archiving to a drive via SMB-service (Server Message Block) Attention: if archiving server is in different sub network set gateway first with SensoFind.
IP Address	IP-Address of target server
Sharing name	Sharing name, specified in dialog “Advanced Sharing” in PC
Workgroup (Domainname)	Option !, Workgoup / Domainname of server / client
User name	User name for FTP / SMB connection.
Password	Password for FTP / SMB connection.
Directory name (pass)	Directory for archiving of data of good parts (pass) (for C:\TESTPASS just enter TESTPASS)
Directory name (fail)	Directory for archiving of data of bad parts (fail) (for C:\TESTFAIL just enter TESTFAIL)
Filename	Filename for images and protocol file, this name is extended automatically by the image number (e.g. TESTFILE).
Add expression	A dynamic part (information such as date and time) is added to the filename. See table below
Image files	Activates archiving of images. Please note: • Images are stored without preprocessing settings, but with the settings for the arrangement (e.g. rotated or mirrored)

Parameters	Functions
	- FTP and SMB save images without overlays. To store images with overlays, please use SensoView.
Result files	If protocol file is active, there will be generated automatically a .csv file for each inspection (trigger). Contents of the file are specified in “Output / Telegram”. Files will have increasing numbers.
Image contents	Possibility to select, whether images should be stored including the selected software filter or “raw” as taken from the camera.
Storage mode	Limit: after reaching maximum number of files transmission is stopped. Unlimited: files are stored, until target drive is full. Cyclic: after reaching maximum number of files the older files are replaced by the newer ones.
Max. number of files	Maximum number of file sets (image+protocol) which are allowed to be stored in the target directory.

The following table shows the expressions that can be added to the filename.

Expression	Description	Example
TIME	HHhMMmSSsSSSms	09h05m11s034ms
HOUR	hh	09
MIN	mm	05
SEC	ss	11
MSEC	sss	034
DATE	YYYY-MM-DD	2011-09-21
YEAR	YYYY	2011
2YEAR	YY	11 (for 2011)
MONTH	MM	09
DAY	DD	21
STRINGID	“Data” entry from extended trigger request “TRX”	Part 34

Expression	Description	Example
COUNTER	Taken from statistics.	3824
XXCOUNTER	Counter taken from statistics with a defined number of digits. XX indicates the number of displayed digits and can accept values from 01 to 10. Please note: If number of counter digits is too small, leading 0 will be added. If number of counter digits is too large most significant digits will be discarded.	06COUNTER → 003824
RESULT	Overall result of job	Pass or Fail
SENSORNAME	As specified in SensoFind	
JOBNAME	As specified in SensoConfig	



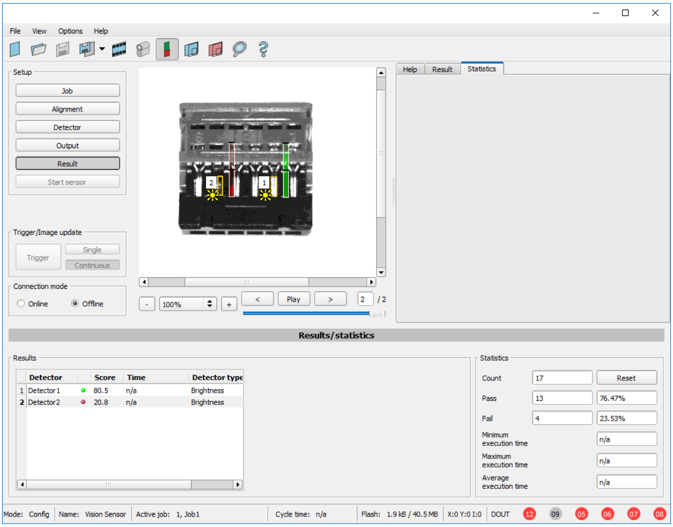
Archive type: Off
IP address: 0 .0 .0 .0
Result files: Any
Image files: Any
Storage mode: Cyclic
Max. number of files: 10

Fig. 263: Tab Output / Archiving

4.6.5 Result

With this function the defined job is processed in the PC, and the “Results/statistics” window with the detector list and the evaluation results is displayed. The cycle times are not displayed in this mode as they are not available from the sensor.

In “Run” mode the results of the detector marked in the detector list are displayed. In the image window – if adjusted – the image, the search- and feature-frames, and the result- graphs are displayed.



Connection mode: ○ Online ● Offline

100% — 2 / 2

Results

	Detector		Score	Time	Detector type
1	Detector1	🟢	80.5	n/a	Brightness
2	Detector2	🔴	20.8	n/a	Brightness

Statistics

Count	17	Reset
Pass	13	76.47%
Fail	4	23.53%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

Mode: Config | Name: Vision Sensor | Active job: 1, Job1 | Cycle time: n/a | Flash: 1.9 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 264: Result

Param. results displayed	Function	Detector type
Detector result	Boolean detector result	All detectors
Score value 1 ... n	Score (0..100%)	All detectors
Execution time	Execution time of individual detector in [msec].	All detectors
Distance	Calculated distance, [1/1000] *1)	Caliper
Position X 1 ... n	Position found X (x-coordinate). [1/1000] *1)	Pattern matching Contour Edge detector Caliper Datacode Barcode

Param. results displayed	Function	Detector type
		OCR
Position Y 1 ... n	Position found Y (y-coordinate). [1/1000] *1)	Pattern matching Contour Edge detector Caliper Datacode Barcode OCR
DeltaPos X	Delta position X between object taught and object found [1/1000] *1)	Pattern matching Contour Edge detector
DeltaPos Y	Delta position X between object taught and object found [1/1000] *1)	Pattern matching Contour Edge detector
Angle	Orientation of object found (0°..360°) [1/1000] *1)	Pattern matching Contour Edge detector Datacode Barcode OCR Wafer Busbar
Delta Angle	Angle between object taught and object found (0°..360°) [1/1000] *1)	Pattern matching Contour Edge detector
Scaling	Only with contour (0.5..2) [1/1000] *1)	Contour

Param. results displayed	**Function**	**Detector type**
R(ed)	Value for color parameter, signed integer [1/1000] *1)	Color value Color list
G(reen)	Value for color parameter, signed integer [1/1000] *1)	Color value Color list
B(lue)	Value for color parameter, signed integer [1/1000] *1)	Color value Color list
H(ue)	Value for color parameter, signed integer [1/1000] *1)	Color value Color list
S(aturation)	Value for color parameter, signed integer [1/1000] *1)	Color value Color list
V(alue)	Value for color parameter, signed integer [1/1000] *1)	Color value Color list
L(uminanz)	Value for color parameter, signed integer [1/1000] *1)	Color value Color list
A	Value for color parameter, signed integer [1/1000] *1)	Color value Color list
B	Value for color parameter, signed integer [1/1000] *1)	Color value Color list
Result index	Index in list, signed integer [1/1000] *1)	Color list
Color distance	Distance between taught and current color, signed integer [1/1000] *1)	Color list
Area	Area of the BLOB, without holes, in pixels, signed integer [1/1000] *1)	BLOB

Param. results displayed	Function	Detector type
Area (incl. holes)	Area of the BLOB, including holes, in pixels, signed integer [1/1000] *1)	BLOB
Contour length	Number of pixels of outer contour, signed integer [1/1000] *1)	BLOB
Compactness	Compactness of BLOB (Circle = 1, all other >1) The stronger the shape of the BLOB deviates from circle the larger the value of compactness will be. Signed integer [1/1000] *1)	BLOB
Center of gravity X	X- coordinate of center of gravity of BLOB, signed integer [1/1000] *1)	BLOB
Center of gravity Y	Y- coordinate of center of gravity of BLOB, signed integer [1/1000] *1)	BLOB
Center X	X- coordinate of fitted, geometric element (rectangle, ellipse), signed integer [1/1000] *1)	BLOB Wafer Busbar
Center Y	Y- coordinate of fitted, geometric element (rectangle, ellipse), signed integer [1/1000] *1)	BLOB Wafer Busbar
Width	Width of geometric element. Width >= 0, width >= height, negative value indicates failure, signed integer [1/1000] *1)	BLOB Wafer
Height	Height of geometric element. Heigth >= 0, height <= width, negative value indicates failure, signed integer [1/1000] *1)	BLOB Wafer
Angle (360)	Orientation of width of object in degree (range: -180 ... +180°, 0° = east, counterclockwise), signed integer [1/1000] *1)	BLOB Wafer Busbar
Eccentricity	Eccentricity numerical (range 0,0 ... 1,0), signed integer [1/1000] *1)	BLOB
Face up/-down, area	Face up/down discrimination, based on area, indicated by sign, signed integer [1/1000] *1)	BLOB
String	Contents of Code, depending from code string length may change, if a fix string length is needed, parameters minimum string length (detector specific data output) and maximum string length (detector parameters) have to be used.	Datacode Barcode OCR

Param. results displayed	**Function**	**Detector type**
String length	Length of Code in Bytes	Datacode Barcode OCR
Truncated	Code truncated	Datacode Barcode OCR
Compare result	Result of string comparison	Datacode Barcode OCR
Quality parameter	Output of quality parameters according to selection	Datacode Barcode
Contrast	Contrast of the code (0-100%)	Barcode
Correction	Number of modules corrected by error corrections	Barcode
Module height	Height of modules in pixels	Datacode
Module width	Width of modules in pixels	Datacode
Confidence	Output of the confidence values of the individual characters	OCR
Result	Degree of similarity between the read string and the reference string from 0 to 100%	OCR
Min. Quality	Minimum required quality was achieved	OCR
Length	Length of busbar	Busbar
Width	Width of busbar	Busbar

*1) All detector-specific data with decimal places are transmitted as whole numbers (multiplied by 1000) and must therefore be divided by 1000 after receipt of data.

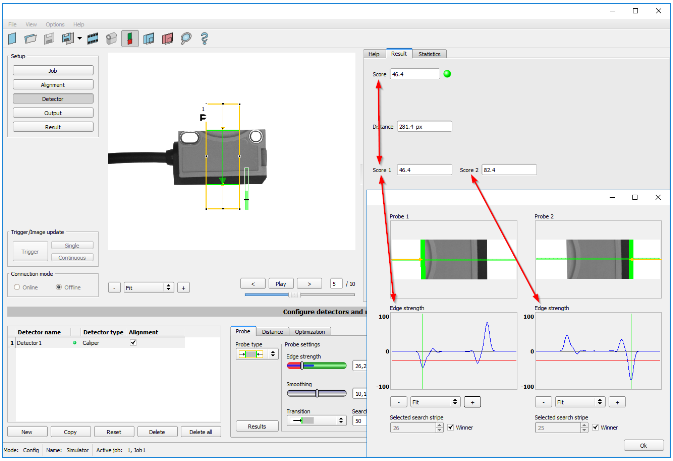
The displayed parameters vary depending on the selected detector type. To see the results of another detector mark it in the detector list. In module SensoView numeric results, statistics and images with or without the selected frames can be archived.

4.6.5.1 1) Score value with result of caliper detector.

in case of Caliper- detector the result value “Score”, “Score 1” and “Score 2” have the following meaning:

Score 1 / Score 2: value of Edge strength in gray values, normalised to 100 (hight of maximum in histogram).

Score: smaller value of both: Score 1 or Score 2.



Score: 46.4

Distance: 281.4 px

Score 1: 46.4

Score 2: 82.4

Fig. 265: Score value caliper detector

4.6.5.2 Result Wafer

This function executes the job defined on the PC and the Result statistics window is displayed with Detector list and Evaluation results. Execution times are not updated in this mode, as they are not available from the sensor.

Detailed inspection results from the detector marked in the selection list are displayed in run mode.

The image, Search and parameter zones (Page 322) and result graphs appear – when set –in the display window.

The parameters displayed vary according to the type of detector selected:

General Outputs

Param. results displayed	Significance

Result	Part / parameter detected (detected = green, not detected = red)
Score	Degree of concordance of pattern found with pattern taught
Execution time	Cycle time for evaluation in ms
Detector Type	Name of active detector for result display

Outputs in Tab Wafer

Param. results displayed	**Significance**
Center X, Center Y	Coordinates of center
Angle	Orientation (absolute angle)
Hight	Hight of wafer
Width	Width of wafer
Area	Area of wafer serves as a stop criterion

Outputs in Tab Summery

Param. results displayed	**Significance**
Contour points found	Number of contour points
Chip size: Deviations	Number of erroneous contour points
Chip size: Area exceeded	Number of adhered chips exceeding area limit
Hole	Number of holes (max. 10)

Outputs in Tab Chip Overview

Param. results displayed	**Significance**
Area	Area of chip
Depth	Maximum depth of chip perpendicular to contour
Angle deviation	Maximum angle deviation
Width	Width of chip along the contour

Param. results displayed	Significance
Angle	Relative angle to wafer orientation
Center X, Center Y	Coordinates of chip center

Outputs in Tab Hole

Param. results displayed	Significance
Area	Area of hole
Center X, Center Y	Coordinates of hole center
Width	Width of hole (horizontal)
Height	Height of hole (vertical)

Outputs in Tab Chip contour

Param. results displayed	Significance
Position X, Position Y	Coordinates of chip center
Depth	Maximum depth of chip perpendicular to contour
Root cause	Possible reason for contour point failure Deviation of contour; chip size, chip shape

To call up inspection results for another detector, mark it in the selection list.

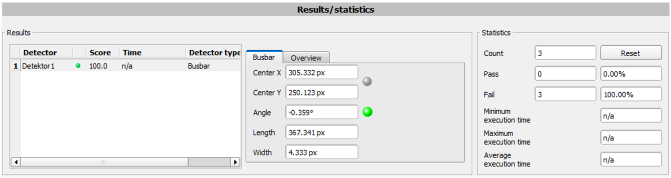
You can archive inspection results and statistical evaluations including selected graphics in the SensoView programme.

4.6.5.3 Result Busbar

This function executes the job defined on the PC and the Result statistics window is displayed with Detector list and Evaluation results. Execution times are not updated in this mode, as they are not available from the sensor.

Detailed inspection results from the detector marked in the selection list are displayed in run mode. The image, search and parameter zones and result graphs appear – when set –in the display window.

The parameters displayed vary according to the type of detector selected:



Results/statistics

Results

	Detector		Score	Time	Detector type
1	Detektor 1	●	100.0	n/a	Busbar

Center X 305.332 px

Center Y 250.123 px

Angle -0.359°

Length 367.341 px

Width 4.333 px

Statistics

Count 3

Pass 0 0.00%

Fail 3 100.00%

Minimum execution time n/a

Maximum execution time n/a

Average execution time n/a

Fig. 266: Result Busbar

General Outputs

Param. results displayed	Significance
Result	Part / parameter detected (detected = green, not detected = red)
Score	Detector result pass/fail (0/100)
Execution time	Cycle time for evaluation in ms
Detector Type	Name of active detector for result display

Outputs in Tab 'Busbar'

Param. results displayed	Significance
Center X, Center Y	Coordinates of center of all busbars
Angle	Orientation (average angle of busbars)
Length	Average length of busbars
Width	Average width of busbars

Outputs in Tab 'Overview'

Param. results displayed	Significance
Center X, Center Y	Coordinates of each busbar center
Area	Area of each busbar
Pads	No of pads of each busbar

Outputs in Tab 'Binarization'

Param. results displayed	Significance
Busbar brightness threshold min. max.	Limit values for binarization of the busbars

To call up inspection results for another detector, mark it in the selection list.

You can archive inspection results and statistical evaluations including selected graphics in the SensoView program.

4.6.6 Start sensor

This function sets the sensor to run mode and executes the job.

Starting execution of a job:

Click on the “Start Sensor” button.

The active (= marked in the selection list) job is transmitted to the sensor, stored in the sensor's non-volatile memory and started (run mode).

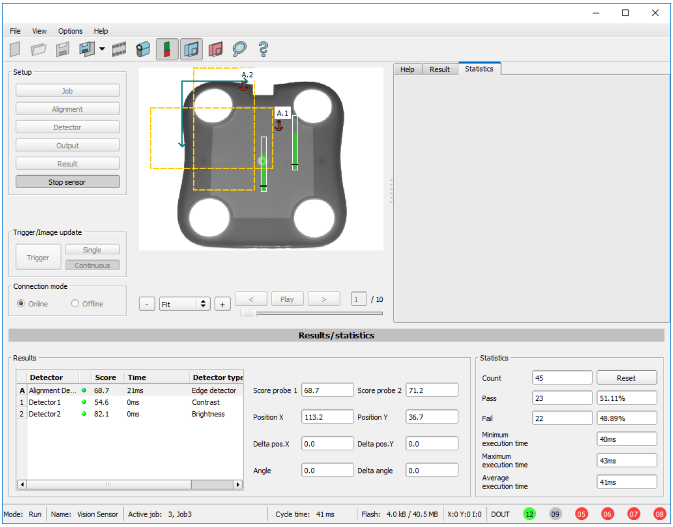
The parameters found are shown in the display window; the inspection results from the first detector or the detector selected in the selection list are shown in the configuration window along with statistical parameters.

Changing detector display:

To display the inspection results for another detector, mark it in the selection list or click on its graphic representation in the display window.

Quitting job execution:

Click on the “Stop Sensor” button. You are now back in configuration mode and can edit your job.



Connection mode: ● Online ○ Offline

Results

	Detector		Score	Time	Detector type
A	Alignment De...	●	68.7	21ms	Edge detector
1	Detector1	●	54.6	0ms	Contrast
2	Detector2	●	82.1	0ms	Brightness

Score probe 1: 68.7 Score probe 2: 71.2

Position X: 113.2 Position Y: 36.7

Delta pos.X: 0.0 Delta pos.Y: 0.0

Angle: 0.0 Delta angle: 0.0

Statistics

Count	45	Reset
Pass	23	51.11%
Fail	22	48.89%
Minimum execution time		40ms
Maximum execution time		43ms
Average execution time		41ms

Mode: Run | Name: Vision Sensor | Active job: 3, Job3 | Cycle time: 41 ms | Flash: 4.0 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 267: Start sensor

4.6.7 Further topics of SensoConfig

4.6.7.1 Trigger settings

Select the required trigger mode in the job settings in the “General” tab:

Parameters	Functions
Triggered	Operation with external trigger, or trigger button in the interface
Free run	Operation with automatically running self-trigger; the sensor supplies images with the maximum possible frequency

Select the form in which the images are to be supplied by the sensor using the option buttons in the zone Trigger/Collect image:

Parameters	Functions
Single image	Recording of a single image, image recording occurs once when: 1. Trigger mode = triggered: First external trigger signal or with the trigger button on the SensoConfig interface 2. Trigger mode = free run: First click on the “Single image” button
Continuous	Continuous supply of images, image recording occurs continuously when: 1. Trigger mode = triggered: Each external trigger or with each click on the trigger button on the SensoConfig interface 2. Trigger mode = free run: Continuously through internal self-triggering with maximum frequency

When exposure time, amplification, illumination or resolution parameters are modified in the Job settings, a new image is automatically requested from the sensor.

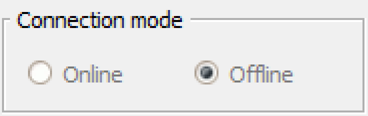
To obtain a continuously updated live image even without trigger, carry out the following (if necessary temporary) settings:

- Set to free run under “Job/General”
- Set to continuous under “Trigger / Collect image”

4.6.7.2 Connection mode: Switching between Online and Offline mode

Two operating modes are available for sensor configuration and test run, which you can select in the *Connection mode* window.

- Online mode: Configuration with connected sensor.
- Offline mode: Simulation of a sensor with the help of images stored in film strips.



Connection mode

○ Online ● Offline

Fig. 268: Connection mode

When the sensor is connected, both modes are available; it is possible to switch between the two. If no sensor is available, it is only possible to work in Offline mode, i.e. with sensor simulation.

4.6.7.3 Simulation of jobs (offline mode)

You can create and test your configuration without a sensor being connected using stored film strips (= series of images). Simulation can be worthwhile to prepare a configuration or to improve a configuration carried out online.

Information:

- Several films are available in SensoConfig when delivered.
- Further methods for image acquisition: Image recorder (Page 328).

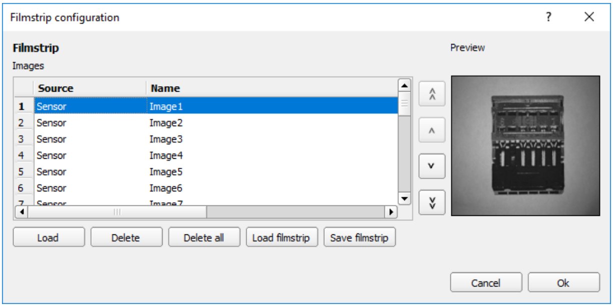
4.6.7.4 Creating filmstrips

In configuration mode, images from the sensor are continuously loaded into the PC's RAM. After switching from online to offline mode, max. 30 images are available and can be stored as a series of images in a filmstrip file. Alternatively or in addition to the images stored on the sensor, you can load series of archived images or individual images on your PC or an external storage medium and combine them into new films.

When you mark an image in the list, it is displayed in small format in the preview window on the right.

4.6.7.4.1 Storing images from the sensor as filmstrips:

1. First connect the PC to the sensor and fill the memory with images in free run and collect image / continuous (Mode of connection = online).
2. Select option button “offline” in the window mode of connection.
3. Select configure filmstrips in the File menu or click on the icon filmstrips in the toolbar. The images loaded from the sensor appear in the selection list that appears below:



Filmstrip configuration

Filmstrip

Images

	Source	Name
1	Sensor	Image1
2	Sensor	Image2
3	Sensor	Image3
4	Sensor	Image4
5	Sensor	Image5
6	Sensor	Image6
7	Sensor	Image7

Fig. 269: Filmstrip

The images now can be examined; re-sorted or individual images can be deleted or added. The maximum number of images in a filmstrip is 30.

4. Click on Button “Save filmstrip” under the selection list.

All images in the list will be saved in a filmstrip file (extension .flm) in the order shown and are now available for future simulation.

4.6.7.4.2 Loading filmstrips and individual images from PC:

1. Select option button “Offline” in the window Mode of connection.
2. Select configure filmstrip in the File menu or click on the icon filmstrip in the tool bar.
3. Select a film file from the selection list and click on “Load filmstrip” button or load individual images from your PC or an external storage medium with the “Load image” button.

The loaded images are added to the selection list.

The type and memory location of the file is shown in the column source: filmstrips stored on the PC (Film), individual image stored on the PC (File), image in sensor memory (Sensor). After switching from online to offline mode all entries are Sensor.

4.6.7.4.3 Editing filmstrips:

You can create new films from the individual images in the selection list regardless of their source. The following functions are available for this purpose:

Button	Function
"<", "<<", ">", ">>"	Change order of images: The marked image is moved up/down one place or is moved to the end of the list.
Load image	Load further images from an external storage medium
Delete, Delete all	Delete image from the list/Delete all images from the list. (The images on the data carrier are not deleted here.)
Abort>	Quit the list without any modification
Import	Load all images into the film memory on the PC in the order shown. These are now available for display and analysis in offline mode.
Load / Save film strip	Load filmstrip from data carrier or save there

4.6.7.4.4 Displays in image window

4.6.7.4.4.1 Controlling image reproduction

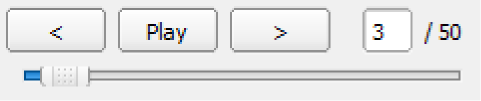


Fig. 270: Image reproduction

You can control the selection and reproduction of stored images using the "<" (back), Start / Stop and ">" (next) buttons as well as the slide bar underneath the display window. The image counter indicates the number of the current image as well as the number of images in the active filmstrip.

4.6.7.4.4.2 Image section and enlargement:

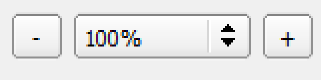


Fig. 271: Zoom

You can select the required image section using the buttons or drop-down menu under the display window.

4.6.7.4.4.3 Graphical display of results

You can active or deactivate the following graphics in the View menu:

- Bar graph result: Displays the inspection result as a bar graph.
- Drawings: Displays search, parameter and position frames detectors and alignment detectors.
- Focussing aid: Displays image sharpness (see also Job settings).
- Enlarged display: Insertion of a separate enlarged display window, which can be adapted to the required scale using the adjustment handles at the corners of the frame.

The module SensoView offers a limited selection of these functions.

4.6.7.5 Image recorder

An image recorder is available in the SensoConfig and SensoView programmes. When the recorder is activated, either all images or just error images are continuously loaded into the internal memory. This covers 10 images, the oldest images are in turn replaced (FIFO buffer). The recorded images can then be called-up and displayed with a PC, or stored on a PC or on an external storage medium, and are then available for analysis or simulation purposes in offline mode.

In the SensoView program, you may be required to enter a password (if activated) to call up recorder images (User user group, see user administration).

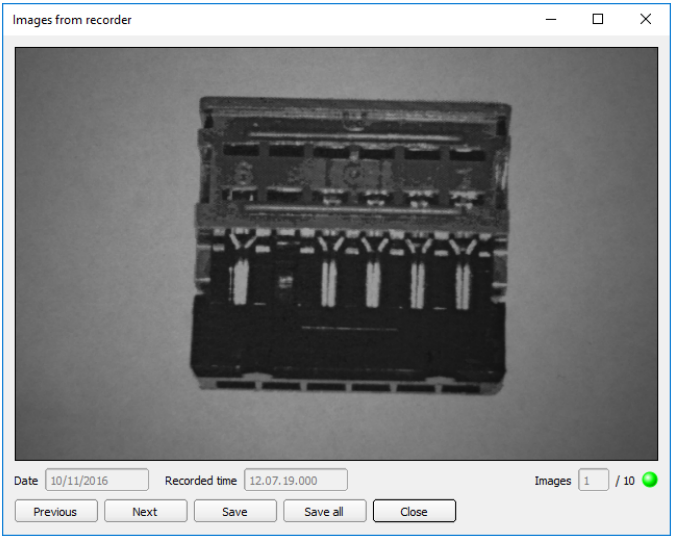
Activating recorder:

Activate the recording function in the setup “output” under the “Image transmission” tab. You can select whether all images (“Any”), only “Pass” images or only “Fail” images should be recorded in the pop-up list of recorder parameters.

Selecting and recording images:

Select “Get images from sensor” from the File menu or click on the button “Rec.images” (only in SensoView).

A display window appears in which you can load images stored in the sensor's RAM on to the PC and then examine and save them:



Date 10/11/2016 Recorded time 12.07.19.000 Images 1 / 10

Fig. 272: Image recorder

Parameter	Function
Back	Displays the previous image
Next	Displays the next image
Save	Saves the image displayed on the PC or an external storage medium
Save all	Saves all images

Information:

- The running number of the selected image and the total number of images recorded on the sensor (max. 10) are displayed in the counter under the display window.
- During storage, the images are deposited in bitmap format (extension .bmp) with a resolution of 640 x 480 pixels (VGA).
- The inspection results associated with the images (OK or error) and the date are stored in the file name (format YYMMDD_running no._Pass/Fail.bmp, e.g. 090225_123456_ Pass.bmp).
- If you want to record detailed inspection results with the images, use the function Archive in SensoView.
- If you only want to record a single image with or without overlay, you can use the function save current image in the file menu, instead of using the recorder.

- Images will get a time stamp when loading them from VISOR® vision sensor.
- Loading images from the sensor on to the PC deletes data on the sensor. If the recorder window is closed without images having been saved, they will also be deleted from the PC.
- Images are lost from the buffer in the event of a loss of power.

4.6.7.6 Displays in image window

4.6.7.6.1 Controlling image reproduction

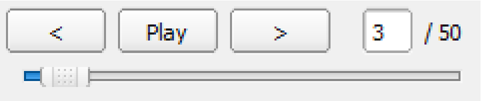


Fig. 273: Image reproduction

You can control the selection and reproduction of stored images using the “<” (back), Start / Stop and “>” (next) buttons as well as the slide bar underneath the display window. The image counter indicates the number of the current image as well as the number of images in the active filmstrip.

4.6.7.6.2 Image section and enlargement:

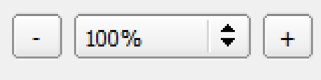


Fig. 274: Zoom

You can select the required image section using the buttons or drop-down menu under the display window.

4.6.7.6.3 Graphical display of results

You can active or deactivate the following graphics in the View menu:

- Bar graph result: Displays the inspection result as a bar graph.
- Drawings: Displays search, parameter and position frames detectors and alignment detectors.
- Focussing aid: Displays image sharpness (see also Job settings).
- Enlarged display: Insertion of a separate enlarged display window, which can be adapted to the required scale using the adjustment handles at the corners of the frame.

The module SensoView offers a limited selection of these functions.

4.6.7.7 Search and parameter zones

You can define search and parameter zones in the configuration steps alignment and detectors. These are identified in the image window by different colored frames.

Drawings in the screen (yellow, red frames etc.) can be activated or deactivated for any detector or category in the menu item “View/all drawings”. With “View/drawings of current detector only”, all drawings on the screen can be deactivated with the exception of the detector currently being processed.

4.6.7.7.1 Definition of search and parameter zones

When a new detector is created, a yellow frame is displayed, which defines the detector's search zone. The standard shape of the search zone is a rectangle; with contrast and gray level detectors, a circle can also be selected. The defined parameters (red frame) are found (green frame) provided its center is within the search zone (yellow frame).

With pattern matching and contour detection detectors, there is also a parameter zone within the search zone which is represented by a red or green frame:

- Red frame = teach parameters
- Green frame = parameters found

If position control / check is defined, a blue frame appears also (either a rectangle, circle or ellipse).

If an alignment detector is defined, it's frame is shown in dotted yellow lines.

At the according upper left corner of each frame the number of the detector is shown.

4.6.7.7.2 Adapting search and parameter zones

The zones initially displayed in standard size and position can be selected / marked in the image or in the detector list and altered in size and position. Eight adjustment handles on the frame enable you to adapt the shape and size of the frame. Its position can be displaced by clicking anywhere inside the frame. The arrow at the side of the frame pointing to the center can be used to change the rotational position of the frame.

The taught sample is represented in original size in the General or Parameters tab in the bottom, right-hand corner of the screen. Only the frame of the currently active detector, selected in the image or detector list, is shown with thick lines and adjustment handles, all other frames which are not selected are shown with thin or dotted lines (alignment detector).

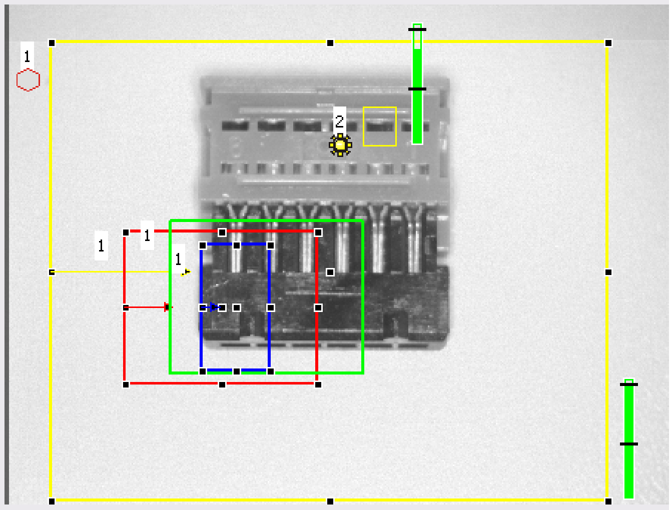


Fig. 275: Search- and feature frames

Information:

- For optimum detection, parameters must be distinct and not contain any variable parts, e.g. shadows.
- Significant contours, edges and contrast distinctions are of advantage.
- To reduce evaluation time, the search zone selected should not be unnecessarily large.

Result bar

On the right next to the search zone, the degree of concordance of the parameter searched for and found is displayed as a fixed result bar with a set threshold value:

- Green bar = The searched for parameter has been found and the pre-set threshold value of minimum concordance has been achieved.
- Red bar = The object could not be found with the required degree of concordance. The graphics displayed can be selected in the View menu.

4.6.7.8 Color models

For description of colors there are available color models.

VISOR® Color is able to work on different color models.

The following color models can be selected:

Color model RGB (Page 324)

Color model HSV (Page 325)

Color model LAB (Page 325)

4.6.7.8.1 Color model RGB

RGB color model is an additive color model, which describes colors by adding the components of the base color red, green and blue.

The RGB- color space is described as a linear color space, as a cube with the three axis Red, Green and Blue.

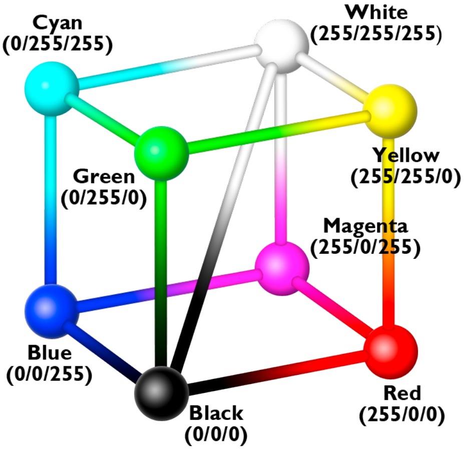


Fig. 276: Color model RGB

red, green, blue, 0-255

RGB color model is used from image capturing chip and from display to define the colors. But image capturing chip and display have different sensivities on each channel.

Because of this there has to be a compensation, means RGB is never the same as RGB.

Linear RGB

RGB values are calculated as linear RGB values, as the sensor chip delivers linear RGB values. Advantage of the linear RGB value is the linear relation between physical impact and RGB value.

Example: Doubling the shutter time leads to doubling of RGB values, if all other illumination conditions remain stable.

4.6.7.8.2 Color model HSV

HSV color model is the most similar to describe what the human eye sees.

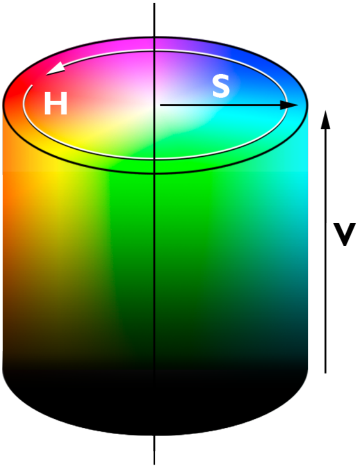


Fig. 277: Color model HSV

- H (hue) stands for the angle on the color circle (e. g. 0° = red, 120° = green, 240° = blue)
- S (saturation) in percent (0 % = light gray, 50 % = low saturated color, 100 % = maximum saturated color)
- V (value) in percent (0 % = dark, 100 % = full brightness)

4.6.7.8.3 Color model LAB

LAB or L*a*b*-color model is built from a three dimensional coordinate system:

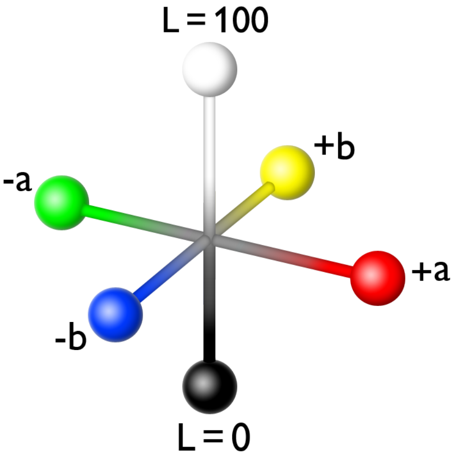


Fig. 278: Color model LAB

- a*-axis describes the red and green components of a color, negative values stand for green- and positive values stand for red. Range of values from -150 to +100.
- b*-axis describes the blue and yellow components of a color, negative values stand for blue positive values stand for yellow. Range of values from -100 to +150.
- L*-axis describes the lightness of the color with values from 0 to 100.

One of the most important properties of the L*a*b color model is the independency from the technology used for capturing and displaying the images.

LAB values are calculated from linear RGB values. This is based in a D65 illuminant and a 2° observer.

4.6.7.9 Application Examples

In Menu “File” “Examples” predefined examples can be loaded.

A filmstrip is loaded together with a job-file.

4.7 VISOR® – Operating- and configuration software – SensoView, all functions

This program enables the monitoring of the image from the camera and the inspection results.

Image display (Page 327)

Result (Page 333)

Statistics (Page 332)

Changing active job (Page 334)

Upload (Page 336)

Commands / Freeze image (Page 328)

Image recorder (Page 328)

Archiving test results and images (Page 330)

From this software ONLY monitoring and job change (loading of already defined jobs) can be done. It can be password protected so that you can only view (worker level), or view and load predefined jobs (Supervisor level).

4.7.1 Image display

The graphical display of an image and the inspection results in the display window depend on the setting of the parameter in tab “Image transmission” in job settings (“Parameters for image transmission” in SensoConfig) program:

- Image transmission active: The current image along with the frames for the defined search, parameter and position zones and parameters found are displayed.
- Image transmission inactive: Only the frames for the defined search, parameter and position zones and parameters found are displayed (current image is not displayed).

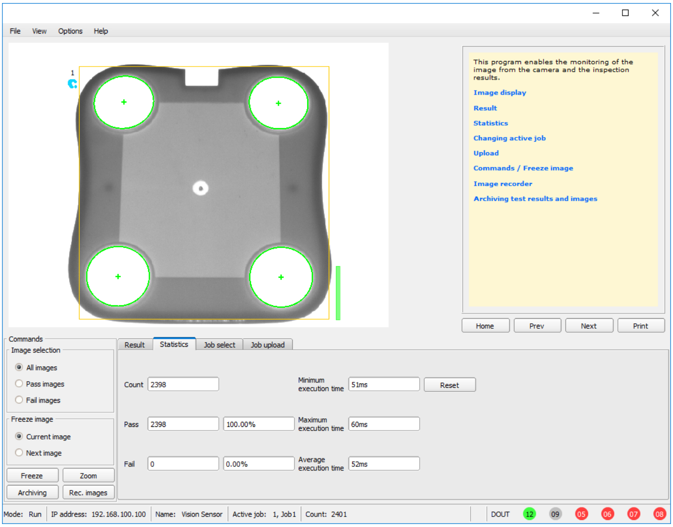
The degree of concordance between the parameter searched for and the parameter found appears to the right next to the search zone of the respective detector, in the form of a vertical result bar with a set threshold value:

- Green bar: The parameter searched for has been found and the pre-set threshold value for concordance has been reached.
- Red bar: The object could not be found with the required degree of concordance.

An exclamation mark in the top right hand corner of the live picture means, that image processing on PC is slower than image processing on VISOR® i.e. Not all images are transferred to PC.

This may cause lost images in images archiving. If this symbol occurs often, PC-programs running in background should be closed in order to improve PC performance.

You can configure the graphics of the inspection results in the View menu.



This program enables the monitoring of the image from the camera and the inspection results.

Image display

Result

Statistics

Changing active job

Upload

Commands / Freeze image

Image recorder

Archiving test results and images

Image selection

● All images

○ Pass images

○ Fail images

Freeze image

● Current image

○ Next image

Count 2398

Pass 2398 100.00%

Fail 0 0.00%

Minimum execution time 51ms

Maximum execution time 60ms

Average execution time 52ms

Mode: Run | IP address: 192.168.100.100 | Name: Vision Sensor | Active job: 1, Job1 | Count: 2401

Fig. 279: SensoView

Except the archiving all functions of SensoView are available also in the module SensoConfig.

4.7.2 Commands / Freeze image

With the “Freeze image” button, you can request single images according to the type required (current image, next image, next failed image) and freeze them in the display window.

The required single image is displayed and the image counter stops at the corresponding image number.

Press “Continue” to end the frozen image state.

4.7.2.1 Zoom

With the button “Zoom” the image is opened in a new window with enlarged display.

4.7.3 Image recorder

An image recorder is available in the SensoConfig and SensoView programmes. When the recorder is activated, either all images or just error images are continuously loaded into the internal memory. This covers 10 images, the oldest images are in turn replaced (FIFO buffer). The recorded images can then be called-up and displayed with a PC, or stored on a PC or on an external storage medium, and are then available for analysis or simulation purposes in offline mode.

In the SensoView program, you may be required to enter a password (if activated) to call up recorder images (User user group, see user administration).

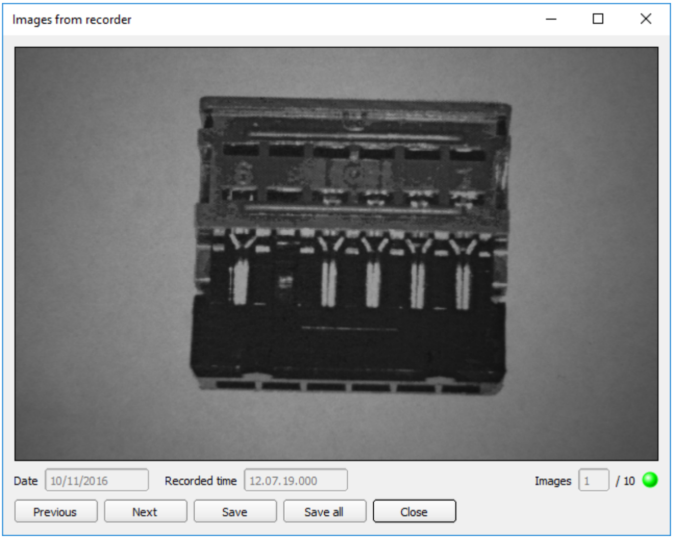
Activating recorder:

Activate the recording function in the setup “output” under the “Image transmission” tab. You can select whether all images (“Any”), only “Pass” images or only “Fail” images should be recorded in the pop-up list of recorder parameters.

Selecting and recording images:

Select “Get images from sensor” from the File menu or click on the button “Rec.images” (only in SensoView).

A display window appears in which you can load images stored in the sensor's RAM on to the PC and then examine and save them:



Date 10/11/2016 Recorded time 12.07.19.000 Images 1 / 10 ●

Fig. 280: Image recorder

Parameter	**Function**
Back	Displays the previous image
Next	Displays the next image
Save	Saves the image displayed on the PC or an external storage medium
Save all	Saves all images

Information:

- The running number of the selected image and the total number of images recorded on the sensor (max. 10) are displayed in the counter under the display window.
- During storage, the images are deposited in bitmap format (extension .bmp) with a resolution of 640 x 480 pixels (VGA).
- The inspection results associated with the images (OK or error) and the date are stored in the file name (format YYMMDD_running no._Pass/Fail.bmp, e.g. 090225_123456_ Pass.bmp).
- If you want to record detailed inspection results with the images, use the function Archive in SensoView.
- If you only want to record a single image with or without overlay, you can use the function save current image in the file menu, instead of using the recorder.
- Images will get a time stamp when loading them from VISOR® vision sensor.
- Loading images from the sensor on to the PC deletes data on the sensor. If the recorder window is closed without images having been saved, they will also be deleted from the PC.
- Images are lost from the buffer in the event of a loss of power.

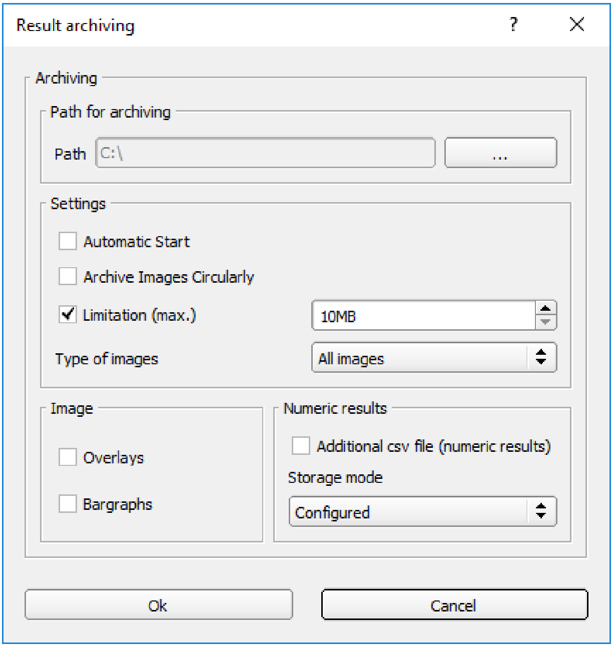
4.7.4 Archiving test results and images

You can archive images with and without graphics, and inspection results on to your PC or an external storage medium for analysis or simulation purposes (see Offline mode).

Access to this function may require password entry (user group, see user administration).

Configuring archiving:

1. Select Configure archiving ... from the File menu.
   A dialogue box appears with the following options:



Result archiving

Archiving

Path for archiving

Path C:\

Settings

☐ Automatic Start

☐ Archive Images Circularly

☑ Limitation (max.) 10MB

Type of images All images

Image

☐ Overlays

☐ Bargraphs

Numeric results

☐ Additional csv file (numeric results)

Storage mode Configured

Fig. 281: Archiving configuration

Parameter	Function
Path for archiving	Directory in which archived file(s) are stored.
Settings, Automatic Start	Starts archiving automatically after start of SensoView.
Settings, Archive image circularly	Activates cyclic overwriting of oldest images if limitation of storage is reached.
Settings, Limitation (max.)	In this drop-down menu it is possible to specify which images (all images or only good or bad images) are to be stored.

Parameter	Function
Type of images	Specifies, whether all, good or bad pictures have to be stored.
Graphics, Bar graph result	Choice of graphics to be archived in the image.
Numerical results	If "record with" is activated, numerical result data such as coordinate values etc. are archived in an additional .csv file. Setting "Legacy" / "Configured" determines the format of storage (.csv). With "Legacy" *1) the content is predefined, with "Configured" the content can be defined in "Output/Telegram". *1) The storage mode "Legacy" is obsolete and only provided for reasons of backward compatibility. It will be omitted with one of the next versions.

2. Select the required options and confirm your choice with OK.

Start/end archiving:

Click on the button "Archive images" in the "Commands" filed to start or end the archiving function with the above mentioned settings. The name of the image file currently to be stored appears in the status bar. Archiving is carried out for as long as the button "Archive images" is pressed.

4.7.5 Statistics

Statistical data from the inspection process is displayed in the Statistics tab in run mode. The statistical data displayed is identical for all types of detectors:

Parameter	Function
All evaluations	Total number of inspections
Good parts	Number of inspections with result "OK"
Bad parts	Number of inspections with result "Error"
Min./max./mean execution time	Min./max./mean execution time for evaluation in ms

All statistic values can be reset to zero with the "Reset" button.

You can archive inspection results and statistical evaluations including selected graphics in the SensoView program.

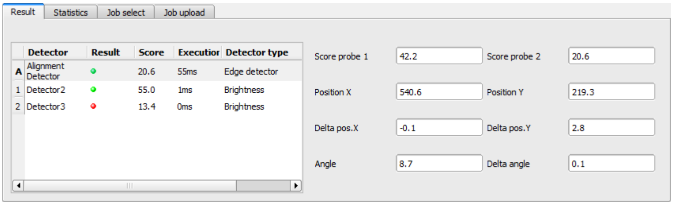
4.7.6 Result

This function executes the job defined on the PC and the Result statistics window is displayed with Detector list and Evaluation results. Execution times are not updated in this mode, as they are not available from the sensor.

Detailed inspection results from the detector marked in the selection list are displayed in run mode.

The image, search and parameter zones and result graphs appear – when set – in the display window.

The parameters displayed vary according to the type of detector selected:



	Detector	Result	Score	Executior	Detector type
A	Alignment Detector	🟢	20.6	55ms	Edge detector
1	Detector2	🟢	55.0	1ms	Brightness
2	Detector3	🔴	13.4	0ms	Brightness

Score probe 1: 42.2
Score probe 2: 20.6
Position X: 540.6
Position Y: 219.3
Delta pos.X: -0.1
Delta pos.Y: 2.8
Angle: 8.7
Delta angle: 0.1

Fig. 282: SensoView, Result

Param. results displayed	Detector type	Function
Result	all	Part / parameter detected (detected = green, not detected = red)
Score 1 .. n	all	Degree of concordance of pattern found with pattern taught
Distance	Caliper	Calculated distance
Execution time	all	Cycle time for an evaluation in ms
Position X 1 .. n, Position Y 1 .. n	Pattern match., Contour. Caliper	Coordinates of parameter found (center point)
Delta X, Delta Y	Pattern match., Contour	Deviation of coordinates found in contrast to taught position / through alignment
Position check	Pattern match., Contour	Position found within the defined position frame
Angle	Pattern match., Contour	Orientation (absolute angle) of parameter found
Delta angle	Pattern match., Con-	Angle deviation between parameter taught and

Param. results displayed	Detector type	Function
	tour	parameter found
Scale	Contour	Scale of contour found in contrast to taught contour.
Result index	Color list	Number in list
Color distance	Color list	Distance of measured color to taught color
Red (Color model RGB)	Color list, Color value	Mean value red
Green (Color model RGB)	Color list, Color value	Mean value green
Blue (Color model RGB)	Color list, Color value	Mean value blue
Hue (Color model HSV)	Color list, Color value	Hue value of color
Saturation (Color model HSV)	Color list, Color value	Saturation of color
Brightness (Color model HSV)	Color list, Color value	Brightness of color
Lightness (Color model LAB)	Color list, Color value	Lightness of color
A (Color model LAB)	Color list, Color value	A- value of color
B (Color model LAB)	Color list, Color value	B- value of color

To show inspection results for another detector, mark it in the selection list.

You can archive inspection results and statistics including selected graphics in SensoView.

4.7.7 Changing active job

In the Job tab, the jobs available on the sensor are displayed in the selection list. Here you can switch between different jobs stored on the sensor. The green arrow (➤) marks the active job

The use of functions which stop an active sensor may require password entry (User group user, see user administration).

Password levels

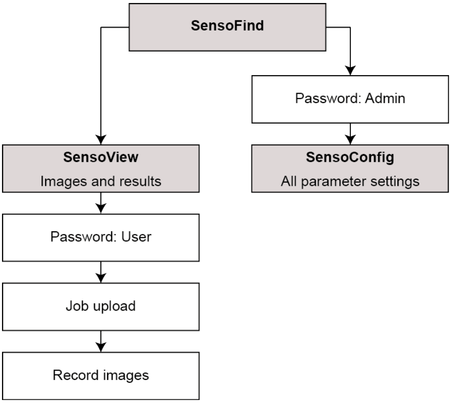
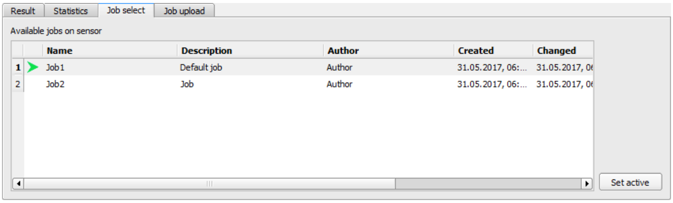


Fig. 283: Password levels



Available jobs on sensor

		Name	Description	Author	Created	Changed
1	➤	Job1	Default job	Author	31.05.2017, 06:...	31.05.2017, 06
2		Job2	Job	Author	31.05.2017, 06:...	31.05.2017, 06

Fig. 284: SensoView, Job select

Select a job from the list and activate it with the “Activated” button. The previous job is deactivated; the selected job is now active.

Attention:

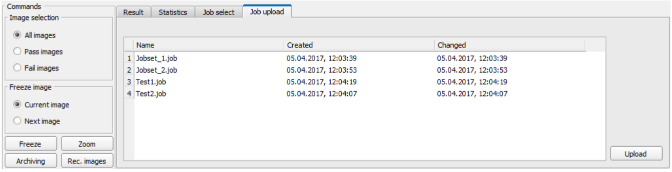
At Job Change and change from Run- to Config Mode outputs will get the following states:

- Buffer of delayed outputs will be deleted.
- Digital outputs: will be reset to default at change from “Run” to “Config”. Defaults are set by flag “Invert” in output tab. “Invert” inverts the default setting and also the result.
- Ready and Valid: Ready and Valid show at Job change and at change of operation mode from Run to Config, that the VISOR® is not ready and that results are not valid (Low level).

4.7.8 Upload

You can load new jobs or entire job sets from the PC to the sensor memory in the Upload tab. The available jobs and job sets are displayed in the selection list.

Jobs and job sets can be created in the SensoConfig program and stored there under File / Save Job / Save Jobset as ...



	Name	Created	Changed
1	Jobset_1.job	05.04.2017, 12:03:39	05.04.2017, 12:03:39
2	Jobset_2.job	05.04.2017, 12:03:53	05.04.2017, 12:03:53
3	Test1.job	05.04.2017, 12:04:19	05.04.2017, 12:04:19
4	Test2.job	05.04.2017, 12:04:07	05.04.2017, 12:04:07

Fig. 285: SensoView, Job set upload

Information:

- A job set consists of one or several jobs which are simultaneously stored in the sensor or on the hard disk.
- Use of functions which can stop the active sensor may require password entry (User user group, see user administration).
- Select a job or job set from the list and load it on to the sensor with the “Upload” button.
- This action deletes all jobs previously stored on the sensor!

5 Communication

5.1 Possibilities of image- / data transfer and archiving

The VISOR® is able to communicate and exchange data via different communication channels with a PLC, I/O extension or a PC. It´s possible to send data on request or cyclical from the VISOR® to a PLC/PC. But the PLC/PC can also actively communicate with the VISOR®, for e.g. only on demand / request to get result- or settings- data or to do a job switch.

The physically available communication interfaces are:

- Ethernet
- RS422

Via Ethernet also the fieldbus interface Ethernet/IP is supported. Via RS422 and the according interface converter the fieldbus Profibus is supported.

A complete overview about all available telegrams you find in chapter Serial Communication ASCII (Page 458).

In the following pages the function and the according settings how to use the different possibilities to communicate with a VISOR® is illustrated in a few examples.

The following examples show how to work on the PC end with a Serial- and Ethernet- software-tool. Here the tool “Hercules” is used. This tool and the settings made here are examples for your PC- or PLC application, and all settings necessary you can see in these examples. If you also like to use the tool Hercules SETUP utility - produced by www.HW-group.com, you can download as freeware.

5.1.1 Ethernet, Port 2005 / 2006

Numerical data, which has been defined under Output/Telegram, now can be transferred in ASCII- or Binary- format.

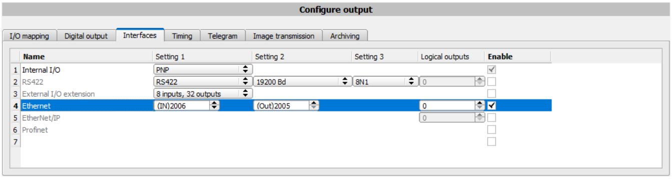
The sensor here is the (socket-) “server” and serves the Data via a “server-socket” interface. This is basically a “programming interface”. To read or process the Data a “socket client” (PC, PLC, ...) must establish a (socket-) connection (active) to the sensor.

Handling, settings

5.1.1.1 Ethernet example 1: Pure data output from VISOR® to PC/ PLC

Step 1:

After the job with all necessary detectors, if so alignment is set up, here the Ethernet interface gets activated and if necessary it’s parameter are set also.



Configure output

	Name	Setting 1	Setting 2	Setting 3	Logical outputs	Enable
1	Internal I/O	PNP				☑
2	RS422	RS422	19200 Bd	8N1	0	□
3	External I/O extension	8 inputs, 32 outputs				□
4	Ethernet	(IN)2006	(Out)2005		0	☑
5	EtherNet/IP				0	□
6	Profinet					□
7						□

Fig. 286: Data output, Ethernet

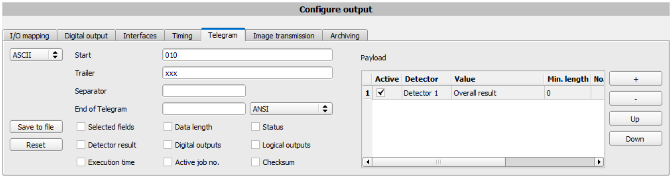
In the example the Ethernet interface in the parameter field at the bottom in tab “interfaces” is activated by marking the checkbox. The default settings for input port (IN) = 2006 and output port (OUT) = 2005 remain as they are in this example. Of course here any other settings can be chosen to do a setup which fit to your network environment. If necessary please contact your network administrator.

Step 2:

In tab “Telegram” the payload which should be transferred via Ethernet port 2005 are set up.

In this example it is:

- Start: “010”
- Overall result of detector 1
- Trailer: “xxx”
- As format “ASCII” is defined, that makes traceability easier. The function with other payload data or in binary format works analogue to this example and to the here made settings.



Configure output

ASCII

Start: 010

Trailer: xxx

Separator:

End of Telegram: ANSI

□ Selected fields □ Data length □ Status

□ Detector result □ Digital outputs □ Logical outputs

□ Execution time □ Active job no. □ Checksum

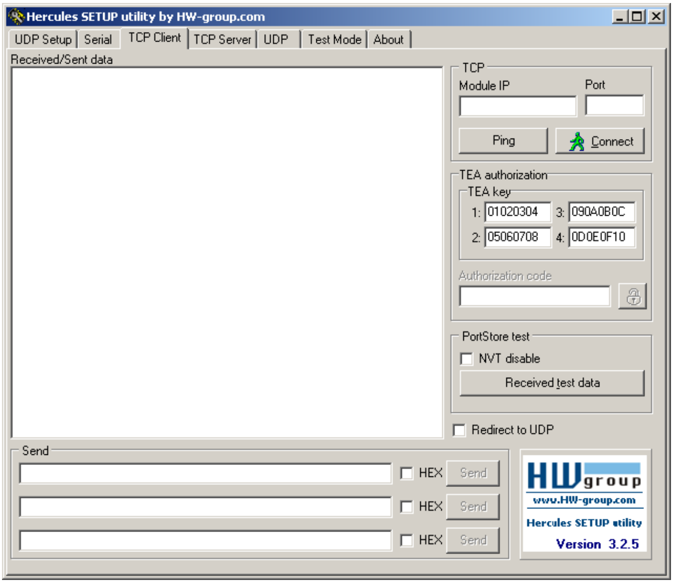
Payload

	Active	Detector	Value	Min. length	No.
1	☑	Detector 1	Overall result	0	

Fig. 287: Data output, configuration of output data

Step 3:

After starting the Ethernet tool “Hercules” the tab “TCP-Client“ must be selected to communicate via Ethernet with the socket- server VISOR®.



Hercules SETUP utility by HW-group.com

UDP Setup | Serial | TCP Client | TCP Server | UDP | Test Mode | About

TEA authorization

TEA key

1: 01020304

2: 05060708

3: 090A0B0C

4: 0D0E0F10

PortStore test

□ NVT disable

□ Redirect to UDP

□ HEX

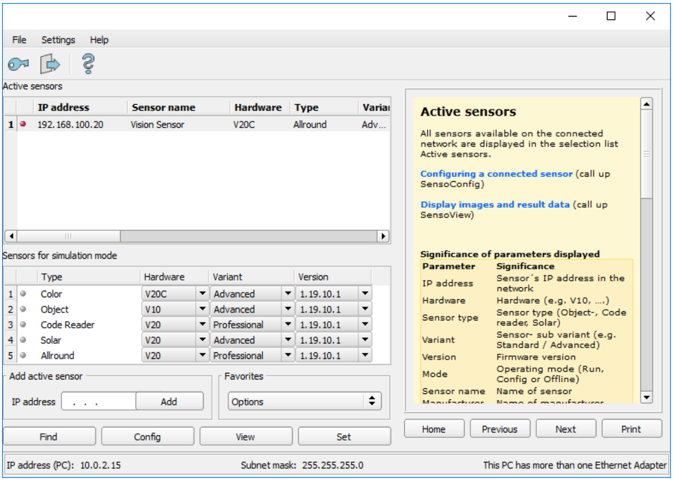
□ HEX

□ HEX

Fig. 288: Data output, Ethernet tool / 1

Here the IP address of the des VISOR® and the correct port number must be set up to receive data.

The IP address of the VISOR® you find in SensoFind. Please look at the first line in the window “Active Sensors” = 192.168.60.199



Active sensors

	IP address	Sensor name	Hardware	Type	Variar
1	192.168.100.20	Vision Sensor	V20C	Allround	Adv...

Sensors for simulation mode

	Type	Hardware	Variant	Version
1	Color	V20C	Advanced	1.19.10.1
2	Object	V10	Advanced	1.19.10.1
3	Code Reader	V20	Professional	1.19.10.1
4	Solar	V20	Advanced	1.19.10.1
5	Allround	V20	Professional	1.19.10.1

Active sensors

All sensors available on the connected network are displayed in the selection list Active sensors.

Configuring a connected sensor (call up SensoConfig)

Display images and result data (call up SensoView)

Significance of parameters displayed

Parameter	Significance
IP address	Sensor´s IP address in the network
Hardware	Hardware (e.g. V10, ...)
Sensor type	Sensor type (Object-, Code reader, Solar)
Variant	Sensor- sub variant (e.g. Standard / Advanced)
Version	Firmware version
Mode	Operating mode (Run, Config or Offline)
Sensor name	Name of sensor
Manufacturer	Name of manufacturer

IP address (PC): 10.0.2.15 Subnet mask: 255.255.255.0 This PC has more than one Ethernet Adapter

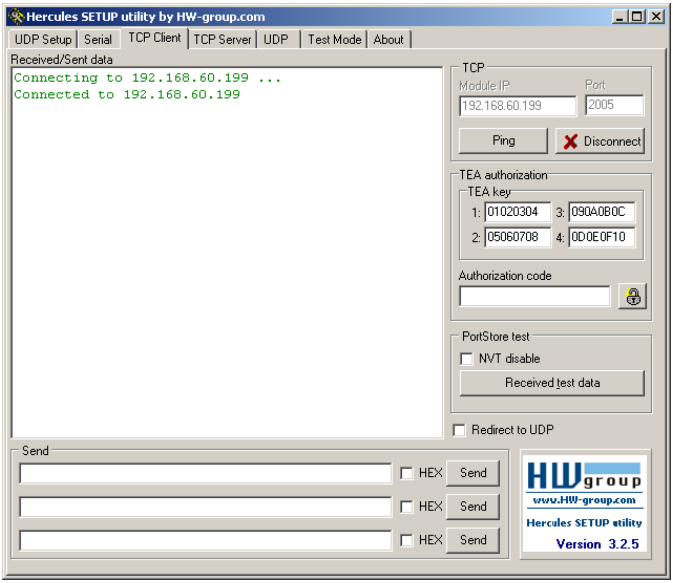
Fig. 289: SensoFind, IP address ...

The port number for the output port was taken over from Step 1 with port 2005.

Step 4:

Therefore the following settings are made in Hercules: Module IP = 192.168.60.199, Port = 2005.

The rest of all settings remain on default. With a click to the button “Connect” the connection to the VISOR® is established and shown in the main window in green letters.



Hercules SETUP utility by HW-group.com

Received/Sent data

```
Connecting to 192.168.60.199 ...
Connected to 192.168.60.199
```

TCP

Module IP: 192.168.60.199

Port: 2005

TEA authorization

TEA key

1: 01020304

2: 05060708

3: 090A0B0C

4: 0D0E0F10

Authorization code

PortStore test

□ NVT disable

□ Redirect to UDP

Send

□ HEX

□ HEX

□ HEX

Fig. 290: Figure 168 Data output, Ethernet Tool / 2

Step 5:

The VISOR® now needs to be started form the PC application with “Start sensor”. (Later in autonomous operation the VISOR® directly starts after power on, and sends data, if configured this way).

In the example Trigger mode is “Continuous”, that means evaluation is done continuously and data is sent continuously too. All this data is visible in the main window of Hercules.

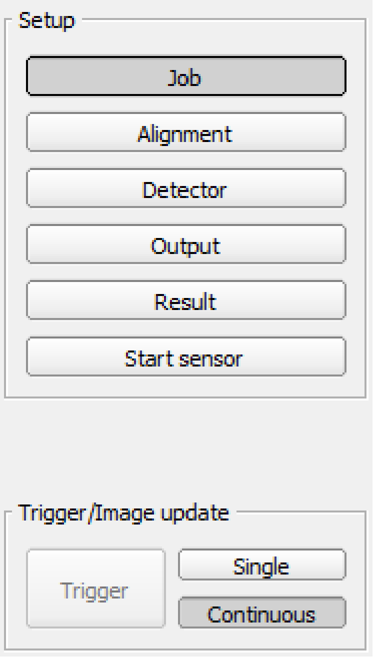
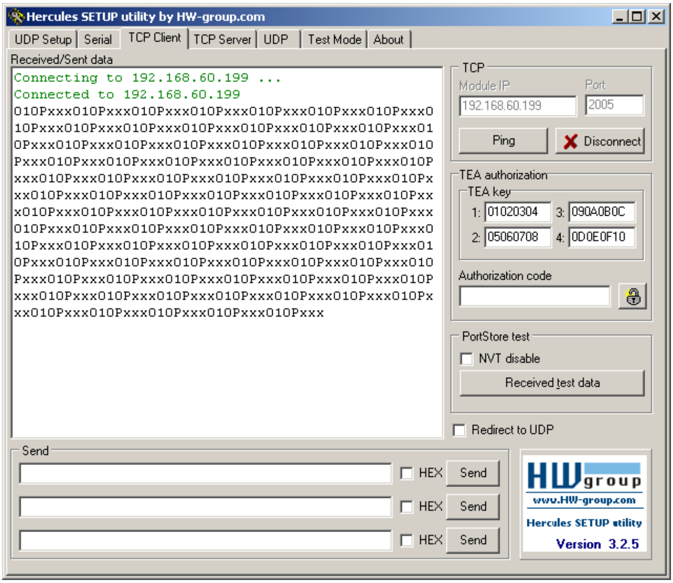


Fig. 291: Data output, Ethernet, Start sensor



Hercules SETUP utility by HW-group.com
UDP Setup | Serial | TCP Client | TCP Server | UDP | Test Mode | About
Received/Sent data

```
Connecting to 192.168.60.199 ...
Connected to 192.168.60.199
010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx0
10Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx01
0Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010
Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010P
xxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Px
xx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxx
x010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx
010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx0
10Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx01
0Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010
Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010P
xxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx010Px
xx010Pxxx010Pxxx010Pxxx010Pxxx010Pxxx
```

TCP
Module IP: 192.168.60.199
Port: 2005

TEA authorization
TEA key
1: 01020304
3: 090A0B0C
2: 05060708
4: 0D0E0F10
Authorization code:

PortStore test
□ NVT disable

□ Redirect to UDP

Send
□ HEX
□ HEX
□ HEX

Fig. 292: Data output, Ethernet, Tool / 3

Then here visible data are displayed (as set up in “Output”):

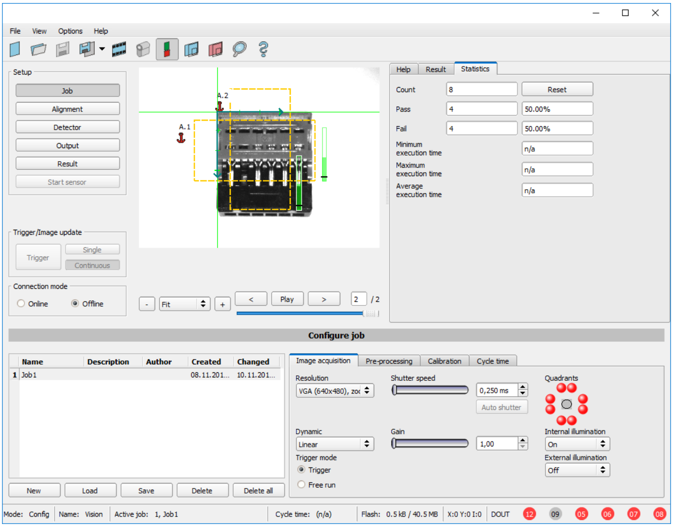
- Start: “010”
- Overall result of detector 1 (“P” for positive, as result of detector Brightness is = “Pass”)
- Trailer: “xxx”

5.1.1.2 Ethernet example 2: commands (requests) from PC / PLC to VISOR®

With response / data output from VISOR®

Step 1

For better traceability in this example the triggered mode is used. That can be done as follows: Adjust Job/Image acquisition/Trigger mode = Trigger. All other settings remain the same like in example 1.



Count: 8
Pass: 4 — 50.00%
Fail: 4 — 50.00%
Minimum execution time: n/a
Maximum execution time: n/a
Average execution time: n/a

Connection mode: ○ Online ● Offline

Fit — 2 / 2

	Name	Description	Author	Created	Changed
1	Job1			08.11.201...	10.11.201...

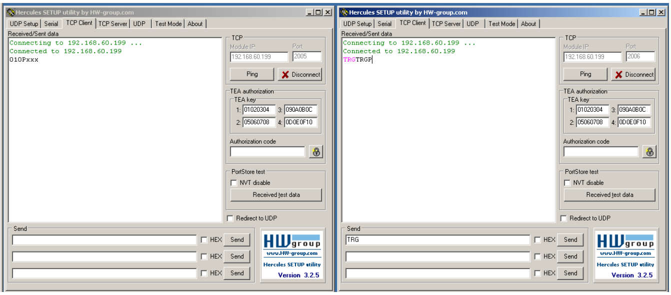
Resolution: VGA (640x480), zo(
Shutter speed: 0,250 ms
Dynamic: Linear
Gain: 1,00
Trigger mode: ● Trigger ○ Free run
Internal illumination: On
External illumination: Off

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 0.5 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 293: Data output, Ethernet, Trigger

Step 2

To send commands / requests to the VISOR®, a second instance of Hercules is started. This time with Port 2006 as input port of the VISOR®, where it can receive commands. All telegrams (commands and response strings) to and from the VISOR® you find in chapter Serial Communication ASCII ...



Received/Sent data

```
Connecting to 192.168.60.199 ...
Connected to 192.168.60.199
010Pxxx
```

TCP — Module IP: 192.168.60.199 Port: 2005
TEA key 1: 01020304 2: 05060708 3: 090A0B0C 4: 0D0E0F10
Authorization code:
□ NVT disable
□ Redirect to UDP

Received/Sent data

```
Connecting to 192.168.60.199 ...
Connected to 192.168.60.199
TRGTRGP
```

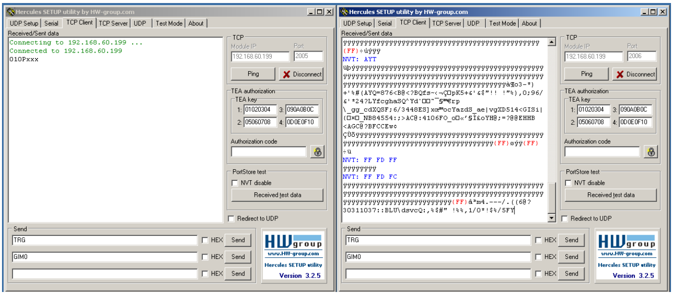
TCP — Module IP: 192.168.60.199 Port: 2006
TEA key 1: 01020304 2: 05060708 3: 090A0B0C 4: 0D0E0F10
Authorization code:
□ NVT disable
□ Redirect to UDP
Send: TRG □ HEX

Fig. 294: Data output, Ethernet Tool / 4

In the window to the right the command “TRG” (for Trigger, command see below, first line) was sent to the VISOR®, by a click to the according button “Send”. This command is shown as soon as it´s sent in the main window in red letters.

The VISOR® responds via port 2006 as a acknowledge to the command with “TRG”, and in this case with “P” for a positive result for detector 1, both in black letters, also in the right Hercules window.

In the left window the VISOR® sends via the output port 2005 the Output defined values “010Pxxx”, like in example Ethernet 1. (Right window)



Hercules SETUP utility by HW-group.com (left window, TCP Client) — Module IP: 192.168.60.199, Port: 2005

```
Connecting to 192.168.60.199 ...
Connected to 192.168.60.199
010Pxxx
```

TEA key 1: 01020304, 2: 05060708, 3: 090A0B0C, 4: 0D0E0F10

Send: TRG; GIM0

Hercules SETUP utility by HW-group.com (right window, TCP Client) — Module IP: 192.168.60.199, Port: 2006

```
ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
{FF}÷úýÿÿ
NVT: AYT
üþýÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿà¶o3-*)
+'%#(AYQ=876<B@<?BQfs~‹¬Ç□pK5+&'&$"!! !"%),0;96/
&'*24?LYfcghaSQ^Yd^□□~¯§™€rp
\_gg_cdXQSF;6/3448ES]xœ™ocYazdS_ae|vgXD514<GISi|
{□¤□_NB84554:;>AC@:4106FO_o□«'§Ī£oYH@;=?@@EHHB
<AGC@?BFCCEw¢
ÇÙõÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ{FF}øýÿ{FF}
÷ü
NVT: FF FD FF
ÿÿÿÿÿÿÿÿ
NVT: FF FD FC
ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ{FF}â³m4.---/.((6@?
30311037::BLU\dsvcQ:,%$#" !%%,1/0*!$%/5FY
```

TEA key 1: 01020304, 2: 05060708, 3: 090A0B0C, 4: 0D0E0F10

Send: TRG; GIM0

Fig. 295: Data output, Ethernet Tool / 5

In the example the command GIM0 (GetIMage0) was sent to the VISOR®. It responds with the binary image data which are shown in the right window. That means, the data output of the manually under “Output” defined payload data happened via port 2005. But the response to the request “GIM0” was transferred via port 2006. This rule is valid for all payload- or response data.

Attention: to use the command GIMx the image recorder must be switched on.

5.1.1.2.1 Ethernet example 2.1 command job switch from PC/PLC to VISOR®

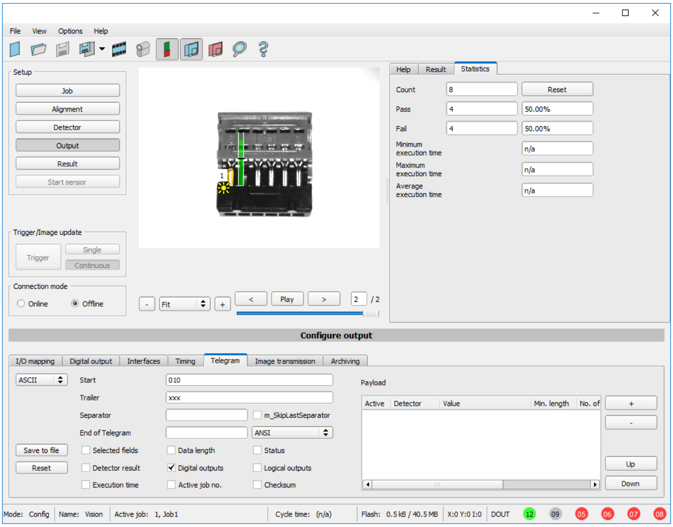
With response / data output from VISOR®

Step 1

For better traceability in this example the triggered mode and ASCII format is used. That can be done as follows: Adjust Job/Image acquisition/Trigger mode = Trigger. All other settings remain the same like in example 1.

For this example Job 1 was set up with the below visible data output:

- Start: “010”
- Trailer: “xxx”



Statistics:

Count	8	Reset
Pass	4	50.00%
Fail	4	50.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

Connection mode: ○ Online ● Offline

2 / 2

Telegram: ASCII

Start: 010

Trailer: xxx

Separator: □ m_SkipLastSeparator

End of Telegram: ANSI

□ Selected fields □ Data length □ Status

□ Detector result ☑ Digital outputs □ Logical outputs

□ Execution time □ Active job no. □ Checksum

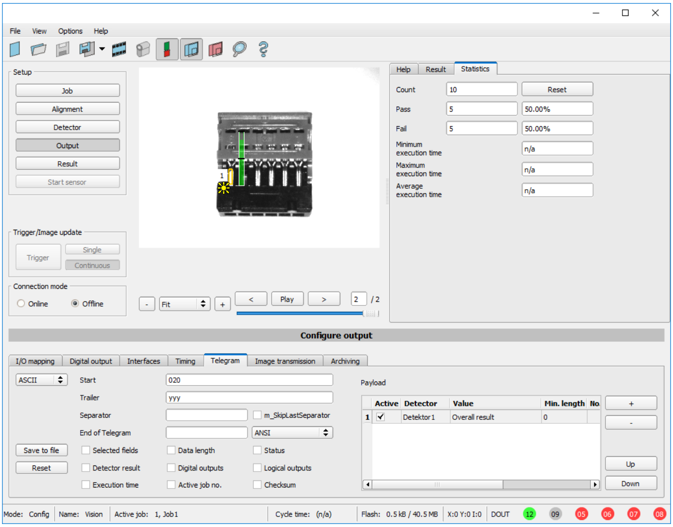
Active	Detector	Value	Min. length	No. of

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 0.5 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 296: Data output, Ethernet, Job switch Job 1

Job 2 was set up with detector 1 and data output:

- Start: “020”
- Overall result of detector 1
- Trailer: “yyy”



Connection mode: ○ Online ● Offline

Count	10	
Pass	5	50.00%
Fail	5	50.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

ASCII

Start: 020

Trailer: yyy

Separator: □ m_SkipLastSeparator

End of Telegram: ANSI

□ Selected fields □ Data length □ Status

□ Detector result □ Digital outputs □ Logical outputs

□ Execution time □ Active job no. □ Checksum

	Active	Detector	Value	Min. length	No
1	☑	Detektor1	Overall result	0	

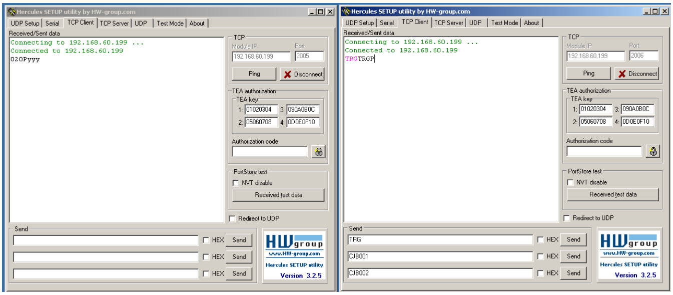
Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 0.5 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 297: Data output, Ethernet, Job switch, Job 2

Step 2

Here the application Hercules is started two times again. First with port 2005 (to receive results like defined under “Output”) and port 2006 (commands and response), as the input port of the VISOR® to receive commands.

All telegrams (commands and response strings) to and from VISOR® you find in chapter Serial Communication ASCII .



```
Connecting to 192.168.60.199 ...
Connected to 192.168.60.199
020Pyyy
```

Module IP: 192.168.60.199 Port: 2005

TEA key 1: 01020304 2: 05060708 3: 090A0B0C 4: 0D0E0F10

□ NVT disable □ Redirect to UDP

```
Connecting to 192.168.60.199 ...
Connected to 192.168.60.199
TRGTRGP
```

Module IP: 192.168.60.199 Port: 2006

TEA key 1: 01020304 2: 05060708 3: 090A0B0C 4: 0D0E0F10

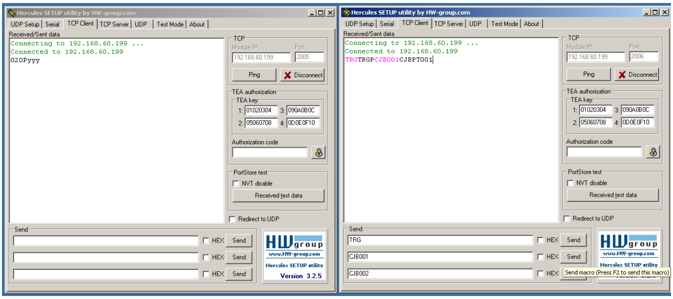
□ NVT disable □ Redirect to UDP

Send: TRG □ HEX; CJB001 □ HEX; CJB002 □ HEX

Fig. 298: Data output, Ethernet, Job switch, tool / 1

In the window to the right (port 2006) the command TRG (Trigger, s. below, first line “Send”) was sent. This is displayed in the main window in red letters “TRG”. The VISOR® responds with the acknowledge “TRGP” (repetition of the command “TRG” and “P” for positive)

In the window to the left (port 2005) the VISOR®, where currently Job 2 is active, sends the according result string which was defined under “Output” in Job 2 with “020Pyyy”.



Left window (port 2005):

```
Connecting to 192.168.60.199 ...
Connected to 192.168.60.199
020Pyyy
```

Module IP: 192.168.60.199 Port: 2005
TEA key 1: 01020304 3: 090A0B0C 2: 05060708 4: 0D0E0F10

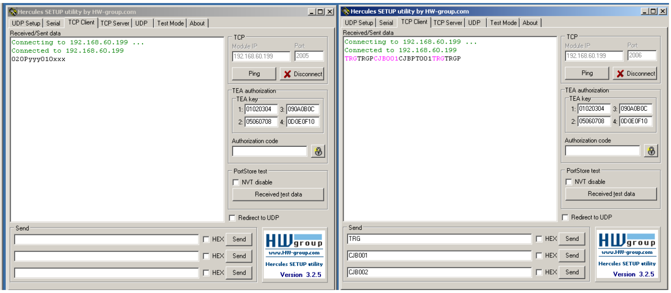
Right window (port 2006):

```
Connecting to 192.168.60.199 ...
Connected to 192.168.60.199
TRGTRGPCJB001CJBPT001
```

Module IP: 192.168.60.199 Port: 2006
TEA key 1: 01020304 3: 090A0B0C 2: 05060708 4: 0D0E0F10
Send: TRG / CJB001 / CJB002

Fig. 299: Data output, Ethernet, Job switch, tool / 2

Now in the right window (port2006) the command CJB001 (ChangeJoB 001, 001 = Job Nr. 1, s. below, second line “Send”) was sent. This is displayed in the main window in red letters “CJB001”. The VISOR® responds with the acknowledge “CJBPT001” (repetition of command “CJB”, “P” for positive, “T” = Triggered, “001” Job number to which was switched).



Left window (port 2005):

```
Connecting to 192.168.60.199 ...
Connected to 192.168.60.199
020Pyyy010xxx
```

Module IP: 192.168.60.199 Port: 2005
TEA key 1: 01020304 3: 090A0B0C 2: 05060708 4: 0D0E0F10

Right window (port 2006):

```
Connecting to 192.168.60.199 ...
Connected to 192.168.60.199
TRGTRGPCJB001CJBPT001TRGTRGP
```

Module IP: 192.168.60.199 Port: 2006
TEA key 1: 01020304 3: 090A0B0C 2: 05060708 4: 0D0E0F10
Send: TRG / CJB001 / CJB002

Fig. 300: Data output, Ethernet, Job switch, tool / 3

After the next Trigger command TRG (s. below third line “Send”) the command “TRG” is displayed again in the main window in red letters. The VISOR® responds with “TRGP” (repetition of command “TRG” and “P” for positive)

In the window left (port2005) the VISOR®, after switching to Job 1!, now the according result sting which was defined under Output in Job 1 with “010xxx”!

Function of the both Ethernet- ports for in- and output:

*A: Port 2005, only one direction: Sensor >> PC, all payload data, defined in “Output”

*B: Port 2006, both directions: Sensor <> PC, commands / requests to the VISOR®, with acknowledge, + all response data to the request (no payload data !).

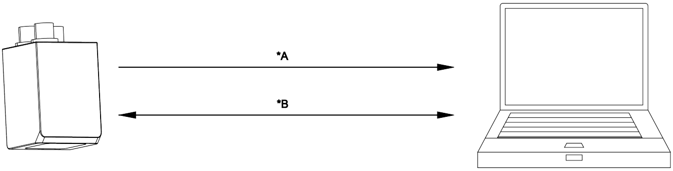


Fig. 301: Ethernet- ports

5.1.2 RS422

Numerical data that has been defined under Output/Telegram, now can be transferred in ASCII- or Binary- format.

Ethernet: The sensor here is the (socket-) “server” and serves the Data via a “server-socket” interface . This is basically a “program interface”. To read or process the Data a “socket client” (PC, PLC, ...) must establish a (socket-) connection (active) to the sensor.

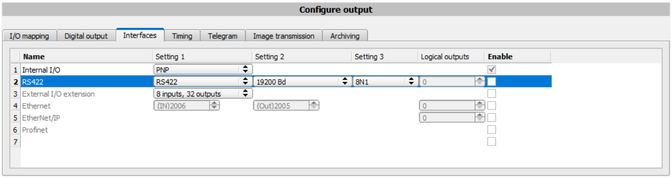
Handling, settings

5.1.2.1 RS422 example 1: Data output from VISOR® to PC / PLC, and commands (requests) to the VISOR®

With response / Data output from VISOR®

Step 1:

After the job with all necessary detectors, if so alignment is set up, here the RS422 interface gets activated and if necessary it’s parameter are set also.



	Name	Setting 1	Setting 2	Setting 3	Logical outputs	Enable
1	Internal I/O	PNP				☑
2	RS422	RS422	19200 Bd	8N1	0	□
3	External I/O extension	8 inputs, 32 outputs				□
4	Ethernet	(IN)2006	(Out)2005		0	□
5	EtherNet/IP				0	□
6	Profinet					□
7						□

Fig. 302: Data output RS422

In the example the RS422 interface in the parameter area at the bottom in tab “Interfaces” gets activated by marking the checkbox.

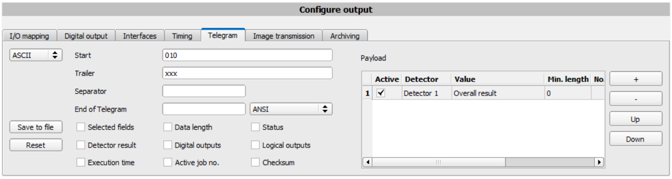
The default settings for Baud rate = 19200 and Logical outputs = 0 remain as they are. Here of course any other settings can be done which must have its corresponding setting at the other side (at the PC or PLC, whatever used).

Step 2:

In tab “Output” the payload data which shall be transferred via RS422 are defined.

In this example this is:

- Start: “010”
- Overall result of detector 1
- Trailer: “xxx”
- As format “ASCII” is defined, that makes traceability easier. The function with other payload data or in binary format works analogue to this example and to the here made settings.



ASCII

Start: 010

Trailer: xxx

Separator:

End of Telegram: ANSI

□ Selected fields □ Data length □ Status

□ Detector result □ Digital outputs □ Logical outputs

□ Execution time □ Active job no. □ Checksum

Payload

	Active	Detector	Value	Min. length	No.
1	✓	Detector 1	Overall result	0	

Fig. 303: Data output RS422, configuration of output data

Step 3:

The VISOR® now needs to be started form the PC application with “Start sensor”. (Later in autonomous operation the VISOR® directly starts after power on, and sends data, if configured this way).

In the example Trigger mode is continuous, that means evaluation is done continuously and data is sent continuously too. All this data is visible in the main window of Hercules.

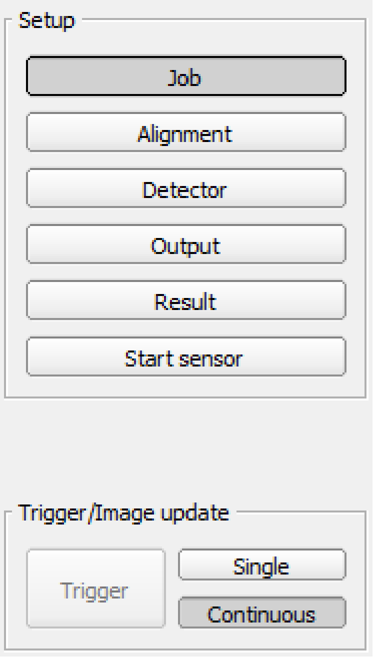
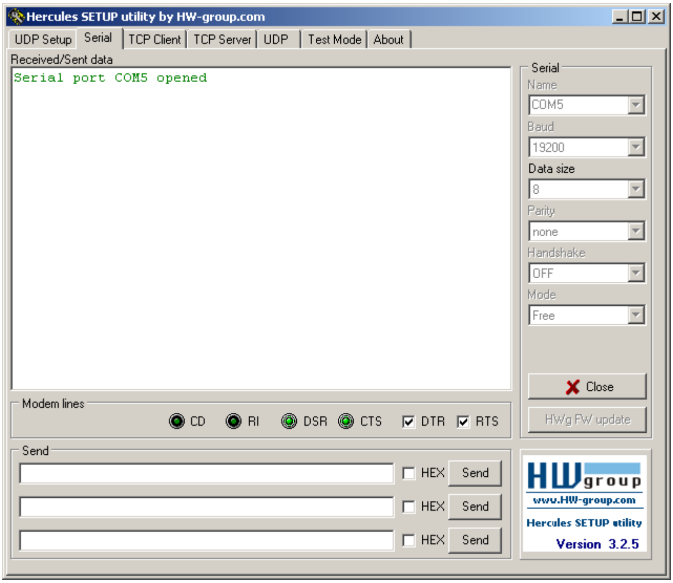


Fig. 304: Start sensor

Step 4:

After start of Serial- tool Hercules, tab “Serial” must be selected to communicate via RS422 with the socket server VISOR®.



```
Serial port COM5 opened
```

Serial
Name: COM5
Baud: 19200
Data size: 8
Parity: none
Handshake: OFF
Mode: Free

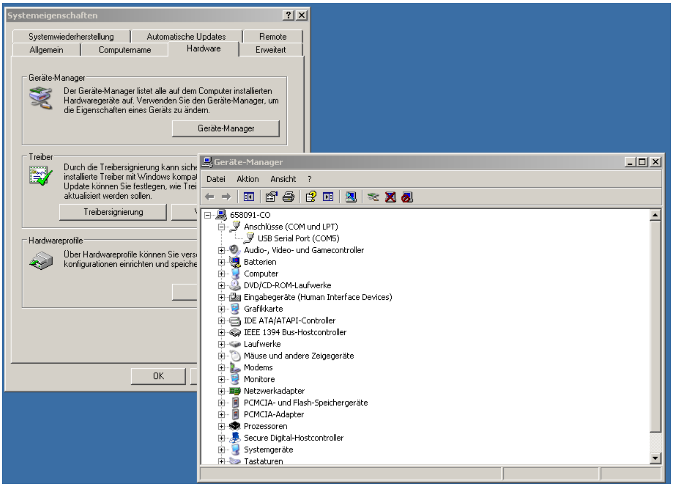
☑ DTR ☑ RTS

Fig. 305: Data output, RS422 tool / 1

Now the corresponding settings for baud rate like in VISOR® must be done. Also the correct serial port COMx must be set up her to receive data.

The baud rate you see in tab Output/Interfaces. The number of the serial COM port (COM x of the PC) you find out in Windows at: Start/Control Panel/Performance and Maintenance/System/Hardware/Device Manager, at Universal Serial Bus Controllers. (Here COM5).

The rest of the settings at the right are the default values of Hercules. “DTR” and “RTS” must be activated. With a click to the button “Connect” the connection to the VISOR® is established and shown in the main window in green letters.



Der Geräte-Manager listet alle auf dem Computer installierten Hardwaregeräte auf. Verwenden Sie den Geräte-Manager, um die Eigenschaften eines Geräts zu ändern.

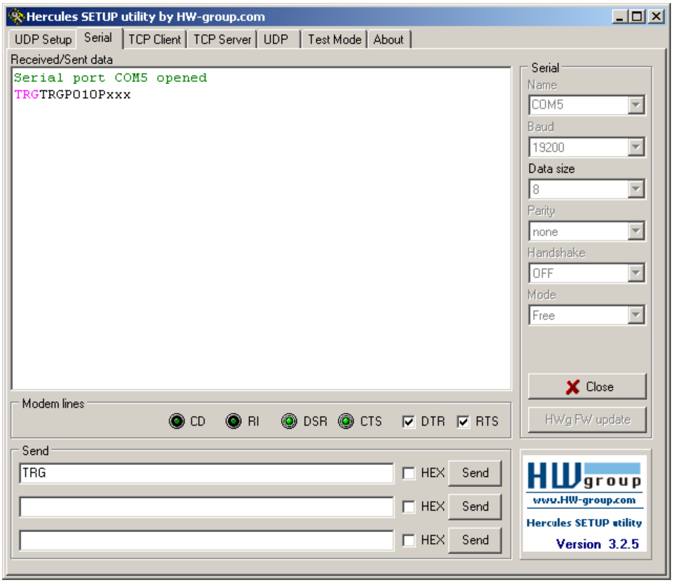
Durch die Treibersignierung kann siche
installierte Treiber mit Windows kompa
Update können Sie festlegen, wie Trei
aktualisiert werden sollen.

Über Hardwareprofile können Sie vers
konfigurationen einrichten und speiche

Fig. 306: Data output, RS422 COMx

Step 5:

With a click to button “Send” the command “TRG” is sent to the VISOR®. It responds with “TRG”, followed by “P” for positive and the payload data “010Pxxx”.



```
Serial port COM5 opened
TRGTRGP010Pxxx
```

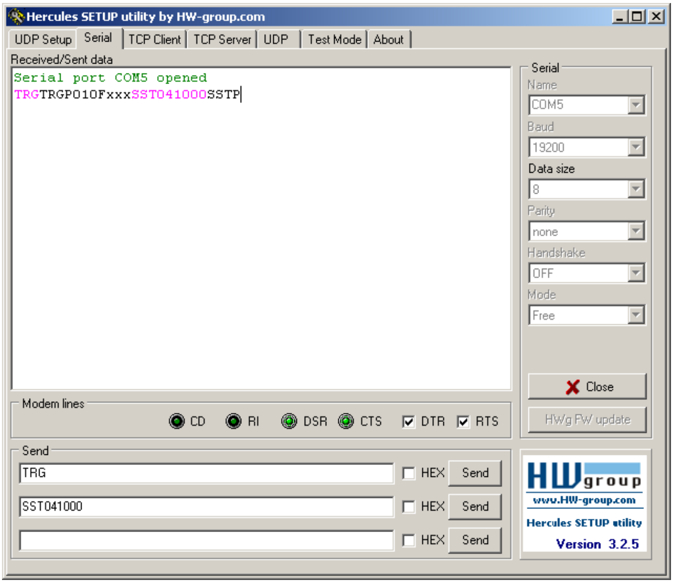
Name: COM5
Baud: 19200
Data size: 8
Parity: none
Handshake: OFF
Mode: Free

Send: TRG

Fig. 307: Data output, RS422, tool / 2

Step 6:

In the following example the command “SST041000” (SetShutterTemporary, 04 = number of letters of shutter value, 1000 = shutter value in microseconds) is sent and the VISOR® responds with SSTP (SetShutterTemporary, P = positive). All available telegrams you find in chapter Serial Communication ASCII . and are used in analogue way.



```
Serial port COM5 opened
TRGTRGP010FxxxSST041000SSTP
```

Serial: Name COM5, Baud 19200, Data size 8, Parity none, Handshake OFF, Mode Free

Send: TRG; SST041000

Fig. 308: Data output, RS422, tool / 3

5.1.2.1.1 RS422 example 1.1: command Job switch from PC / PLC to VISOR®

With response / data outputs from VISOR®

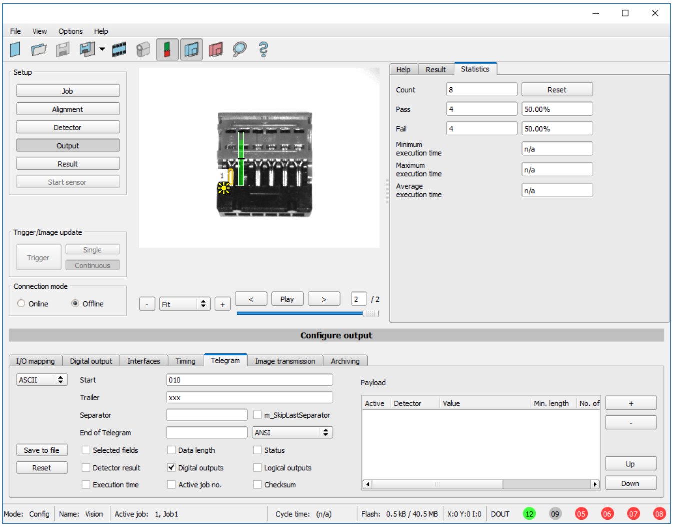
Step 1

Here the same setting for Job and Output are used as in “Ethernet Example 2.1”.

For better traceability in this example the triggered mode and ASCII format is used. That can be done as follows: Adjust Job/Image acquisition/Trigger mode = Trigger. All other settings remain the same like in example 1. In Output/Interfaces here the interface RS422 was activated.

For this example Job 1 was set up with the below visible data output:

- Start: “010”
- Trailer: “xxx”



Count: 8

Pass: 4 50.00%

Fail: 4 50.00%

Minimum execution time: n/a

Maximum execution time: n/a

Average execution time: n/a

Connection mode: ○ Online ● Offline

2 / 2

ASCII

Start: 010

Trailer: xxx

Separator: □ m_SkipLastSeparator

End of Telegram: ANSI

□ Selected fields □ Data length □ Status

□ Detector result ☑ Digital outputs □ Logical outputs

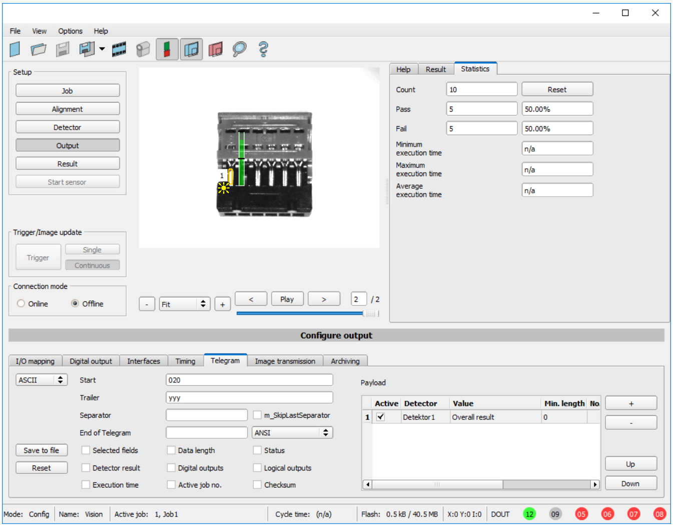
□ Execution time □ Active job no. □ Checksum

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 0.5 kB / 40.5 MB | X:0 Y:0 I:0

Fig. 309: Data output, RS422, Job switch, Job 1

Job 2 was set up with detector 1 and data output:

- Start: “020”
- Overall result of detector 1
- Trailer: “yyy”



Count: 10

Pass	5	50.00%
Fail	5	50.00%
Minimum execution time		n/a
Maximum execution time		n/a
Average execution time		n/a

Connection mode: ○ Online ● Offline

2 / 2

ASCII

Start: 020

Trailer: yyy

Separator: □ m_SkipLastSeparator

End of Telegram: ANSI

□ Selected fields □ Data length □ Status

□ Detector result □ Digital outputs □ Logical outputs

□ Execution time □ Active job no. □ Checksum

Payload

	Active	Detector	Value	Min. length	No.
1	☑	Detektor1	Overall result	0	

Mode: Config | Name: Vision | Active job: 1, Job1 | Cycle time: (n/a) | Flash: 0.5 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 310: Data output, RS422, Job switch, Job 2

Step 2

After start of Serial- tool Hercules, tab “Serial” must be selected to communicate via RS422 with the socket server VISOR®.

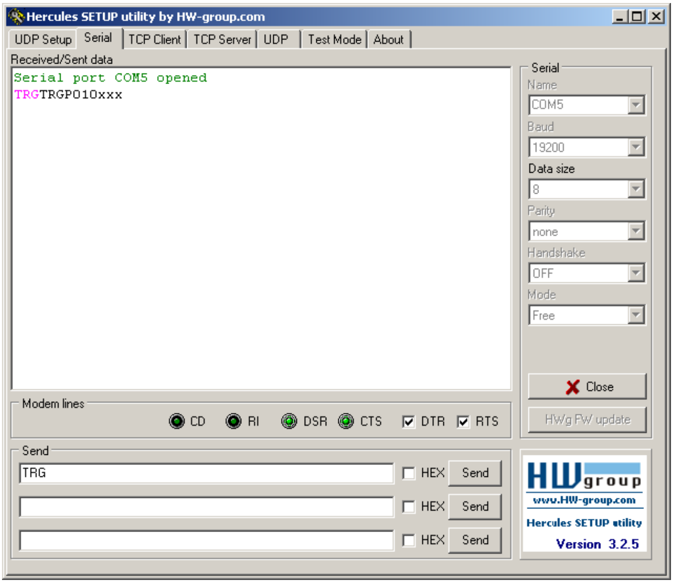
Now the corresponding settings for baud rate like in VISOR® must be done. Also the correct serial port COMx must be set up here to receive data.

The baud rate you see in tab Output/Interfaces. The number of the serial COM port (COM x of the PC) you find out in Windows at: Start/Control Panel/Performance and Maintenance/System/Hardware/Device Manager, at Universal Serial Bus Controllers. (Here COM5).

The rest of the settings at the right are the default values of Hercules. “DTR” and “RTS” must be activated. With a click to the button “Connect” the connection to the VISOR® is established and shown in the main window in green letters.

Step 3

With the command “TRG” (Trigger, s. below, line 1, “Send”) an image acquisition and an evaluation was initiated. The VISOR® immediately responds with “TRGP” (“P” for positive). Also, as in this moment Job 1 is active, the result data string “010xxx” is sent.


Received/Sent data

```
Serial port COM5 opened
TRGTRGP010xxx
```

Serial: Name COM5, Baud 19200, Data size 8, Parity none, Handshake OFF, Mode Free

Modem lines: CD, RI, DSR, CTS, ☑ DTR, ☑ RTS

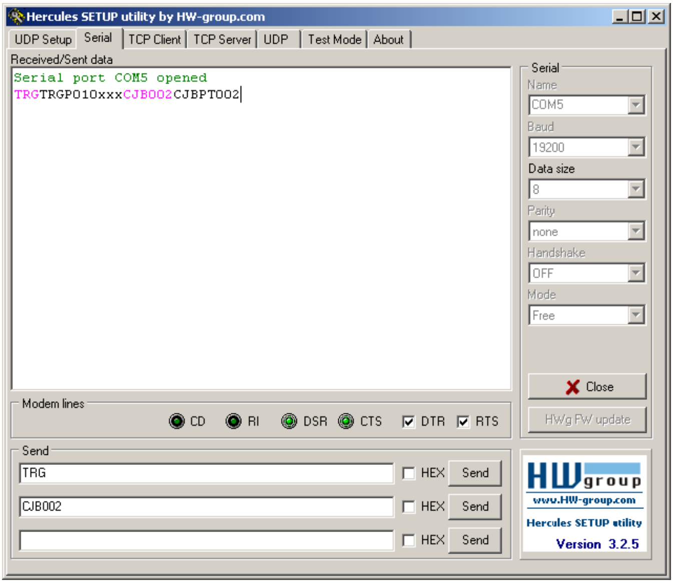
Send: TRG

Fig. 311: Data output, RS422, Job switch tool / 1

Step 4

With the command “CJB002” (ChangeJoB, Job Nr. 002, s. below line2, “Send”) the VISOR® now switches to Job 2.

The response: “CJBPT002” (repetition of command “CJB”, “P” for positive, “T” = Triggered, 002 Job number switched to) is sent and displayed in main window.



```
Serial port COM5 opened
TRGTRGP010xxxCJB002CJBPT002
```

Serial: Name COM5; Baud 19200; Data size 8; Parity none; Handshake OFF; Mode Free

Modem lines: ☑ DTR ☑ RTS

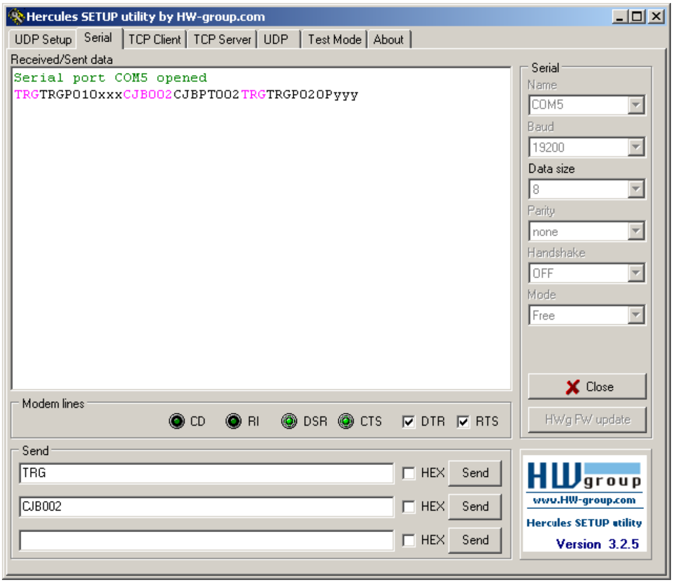
Send:

- TRG □ HEX Send
- CJB002 □ HEX Send
- □ HEX Send

Fig. 312: Data output, RS422, Job switch tool / 2

Step 5

After the next Trigger command TRG (s. below line 1, “Send”) the command “TRG” the next evaluation is performed and the response “TRGP” (repetition of command “TRG” and “P” for positive) is sent. Also, as now Job 2 is active, the result string “020Pyyy” like in Job 2 defined is transmitted.



Hercules SETUP utility by HW-group.com

Received/Sent data

```
Serial port COM5 opened
TRGTRGP01OxxxCJB002CJBPT002TRGTRGP02OPyyy
```

Serial
Name: COM5
Baud: 19200
Data size: 8
Parity: none
Handshake: OFF
Mode: Free

Modem lines: ● CD ● RI ● DSR ● CTS ☑ DTR ☑ RTS

Send
TRG □ HEX Send
CJB002 □ HEX Send
□ HEX Send

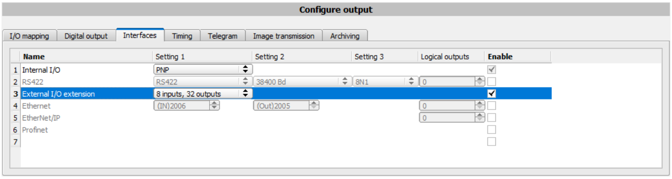
Fig. 313: Data output, RS422, Job switch tool / 3

5.1.2.2 Settings to connect the "I/O-Box" for I/O- extension or ejector control to the VISOR®

To operate the I/O-Box with the VISOR® the following settings in Output/ Interfaces/External I/O extension must be done.

Setting 1: 8Inputs_32Outputs

Enable: Mark checkbox in column "Enable"



Configure output

	Name	Setting 1	Setting 2	Setting 3	Logical outputs	Enable
1	Internal I/O	PNP				☑
2	RS422	RS422	38400 Bd	8N1	0	□
3	External I/O extension	8 inputs, 32 outputs				☑
4	Ethernet	(IN)2006	(Out)2005		0	□
5	EtherNet/IP				0	□
6	Profinet					□
7						□

Fig. 314: Data output, connection of I/O Box

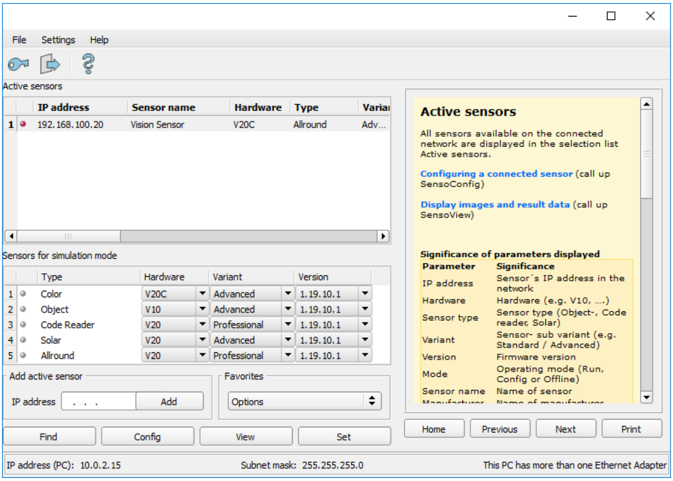
5.1.3 PC- Archiving (SensoView)

Via SensoView images and numerical data (in .csv format) can be stored into a folder on the PC.

The setup (folder ...) is done via SensoView in menu "File/Archiving". This function is available on PC only.

Step 1:

Start SensoView from SensoFind, Click to button "View"



Active sensors

		IP address	Sensor name	Hardware	Type	Variai
1	●	192.168.100.20	Vision Sensor	V20C	Allround	Adv...

Sensors for simulation mode

		Type	Hardware	Variant	Version
1	●	Color	V20C	Advanced	1.19.10.1
2	●	Object	V10	Advanced	1.19.10.1
3	●	Code Reader	V20	Professional	1.19.10.1
4	●	Solar	V20	Advanced	1.19.10.1
5	●	Allround	V20	Professional	1.19.10.1

Add active sensor

IP address: . . .

Favorites

IP address (PC): 10.0.2.15 Subnet mask: 255.255.255.0 This PC has more than one Ethernet Adapter

Active sensors

All sensors available on the connected network are displayed in the selection list Active sensors.

Configuring a connected sensor (call up SensoConfig)

Display images and result data (call up SensoView)

Significance of parameters displayed

Parameter	Significance
IP address	Sensor´s IP address in the network
Hardware	Hardware (e.g. V10, ...)
Sensor type	Sensor type (Object-, Code reader, Solar)
Variant	Sensor- sub variant (e.g. Standard / Advanced)
Version	Firmware version
Mode	Operating mode (Run, Config or Offline)
Sensor name	Name of sensor
Manufacturer	Name of manufacturer

Fig. 315: SensoFind

SensoView is started.

The conditions for a correct image display are the settings:

- Free run (set in Job/Image acquisition) or
- At least one trigger happened
- Image transmission active (set in Job/Image transmission)

Step 2

Select in menu: File/Archiving

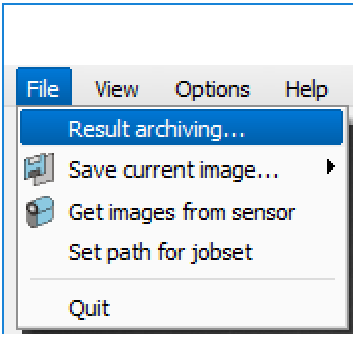


Fig. 316: SensoView, Archiving

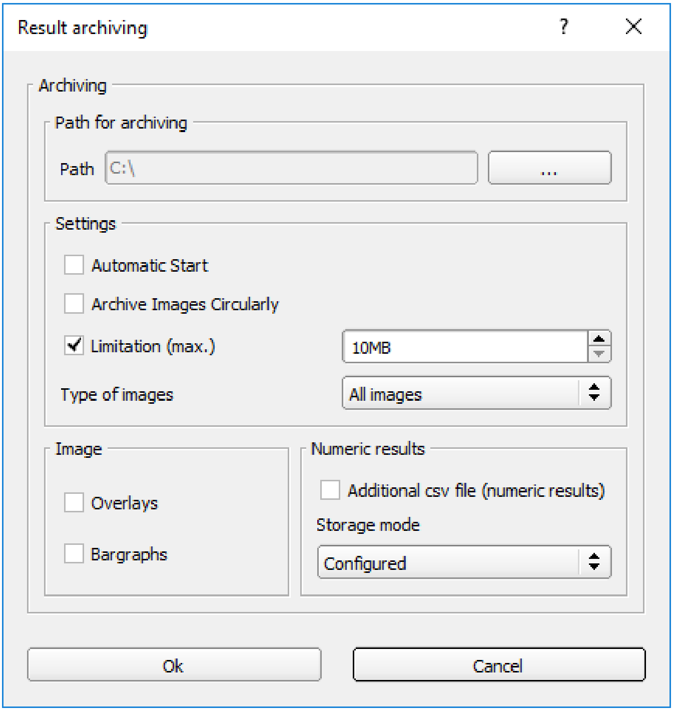
Now the following dialog box occurs to set up parameter for archiving.

Parameter	Function
Path for archiving	Directory in which archived file(s) are stored.
Settings, Automatic Start	Starts archiving automatically after start of SensoView.
Settings, Archive image circularly	Activates cyclic overwriting of oldest images if limitation of storage is reached.
Settings, Limitation (max.)	In this drop-down menu it is possible to specify which images (all images or only good or bad images) are to be stored.
Type of images	Specifies, whether all, good or bad pictures have to be stored.
Graphics, Bar graph result	Choice of graphics to be archived in the image.
Numerical results	If “record with” is activated, numerical result data such as coordinate values etc. are archived in an additional .csv file. Setting FA46 / VISOR® determines the format of storage, at FA46 contents of .csv file is predefined, at VISOR® contents can be defined in “Output / Telegram”

Select the required options and confirm your choice with OK.

5.1.3.1 Start/end archiving:

Click on the button “Archive images” in the “Commands” filed to start or end the archiving function with the above mentioned settings. The name of the image file currently to be stored appears in the status bar. Archiving is carried out for as long as the button “Archive images” is pressed.


Result archiving

Archiving

Path for archiving

Path C:\

Settings

□ Automatic Start

□ Archive Images Circularly

☑ Limitation (max.) 10MB

Type of images: All images

Image

□ Overlays

□ Bargraphs

Numeric results

□ Additional csv file (numeric results)

Storage mode: Configured

Fig. 317: SensoView, Archiving configuration

5.1.4 Archiving via ftp or smb

With this function images and numeric result data (in .csv format) can be stored actively by the sensor via ftp/smb. This kind of archiving is configured under “Job/Archiving”, in this case:

a. **With “ftp” used**: the sensor is a “ftp client” and “writes” the data to a “ftp server” folder on a drive which is available in the network. With Job/Start the sensor connects to the ftp-Server.

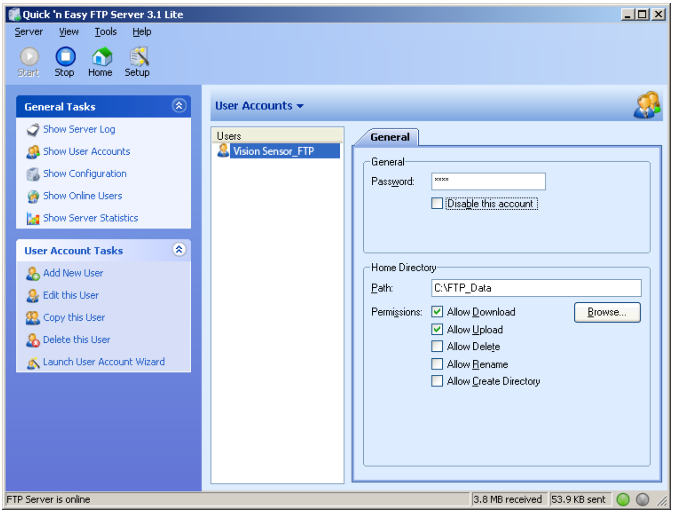
b. **With “smb” used**: the sensor “writes” the data direct in a folder in a network. With Job/Start the sensor connects/mounts with this folder.

With this kind of data archiving in normal operation case no PC application like SensoFind or SensoConfig is running, just a accordingly configured ftp- or smb- server.

5.1.4.1 Example: Archiving via ftp

In this example with the ftp- server freeware “Quick´n Easy FTP Server” a ftp communication was established and image- and result data are stored on the hard disc of the PC.

In the ftp server with the account wizard a user account with the name “VISOR®_FTP” was created. A password and a path for data storage have been specified, and upload and download are activated.



Vision Sensor_FTP

General

Password: ****

□ Disable this account

Home Directory

Path: C:\FTP_Data

Permissions: ☑ Allow Download
☑ Allow Upload
□ Allow Delete
□ Allow Rename
□ Allow Create Directory

FTP Server is online | 3.8 MB received | 53.9 KB sent

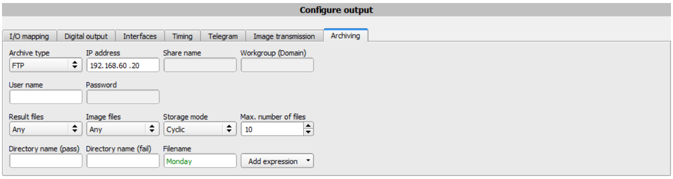
Fig. 318: FTP Server

In SensoConfig now at: Output/Archiving the according settings for the ftp server on the VISOR® must be done. This are:

- Archive type = FTP
- IP address = IP of the PC where the ftp server is running (IP address of PC connected you find in status line in SensoFind in the corner left, below)
- User name = Name of the user account in the ftp server
- Password = in the ftp account used Password (option)

With this the for ftp communication according settings are done.

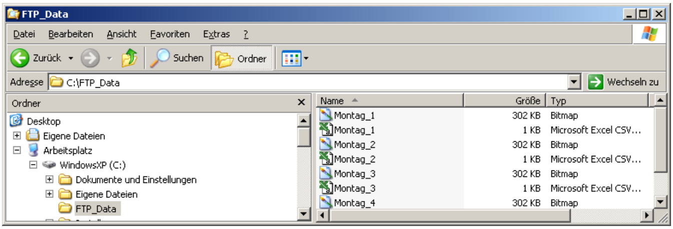
Also other settings like: Filename, Max. number of files, Storage mode … can be made here



Archive type: FTP
IP address: 192.168.60.20
Share name:
Workgroup (Domain):
User name:
Password:
Result files: Any
Image files: Any
Storage mode: Cyclic
Max. number of files: 10
Directory name (pass):
Directory name (fail):
Filename: Monday

Fig. 319: FTP Server, settings in SensoConfig

As soon as this settings are done and transferred to the VISOR® (with “Start Sensor”), the image and result data are transferred and stored into the specified folder on the PC, without any of the applications SensoFind, SensoConfig or SensoView active.



Adresse: C:\FTP_Data

Name	Größe	Typ
Montag_1	302 KB	Bitmap
Montag_1	1 KB	Microsoft Excel CSV…
Montag_2	302 KB	Bitmap
Montag_2	1 KB	Microsoft Excel CSV…
Montag_3	302 KB	Bitmap
Montag_3	1 KB	Microsoft Excel CSV…
Montag_4	302 KB	Bitmap

Fig. 320: Transferring files with FTP.

The function via smb works analogue via a smb server, which must be set up in the according kind.

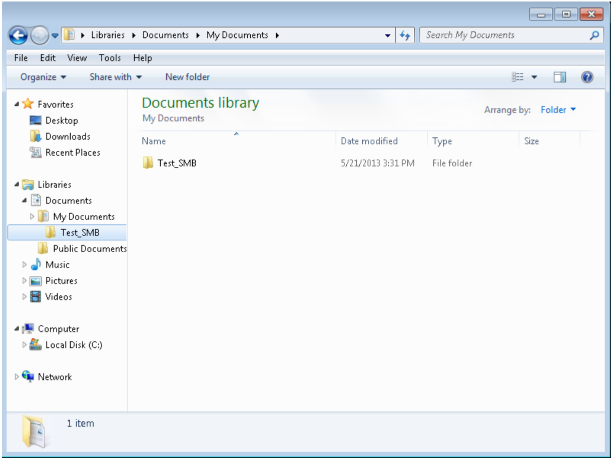
5.1.4.2 Example: Archiving via SMB

To archive data and / or images via SMB (Server message block), at the end of the PC a folder must be shared.

The following example shows the settings for archiving data via SMB exemplarily.

5.1.4.2.1 Settings for SMB on PC: Create folder and share it

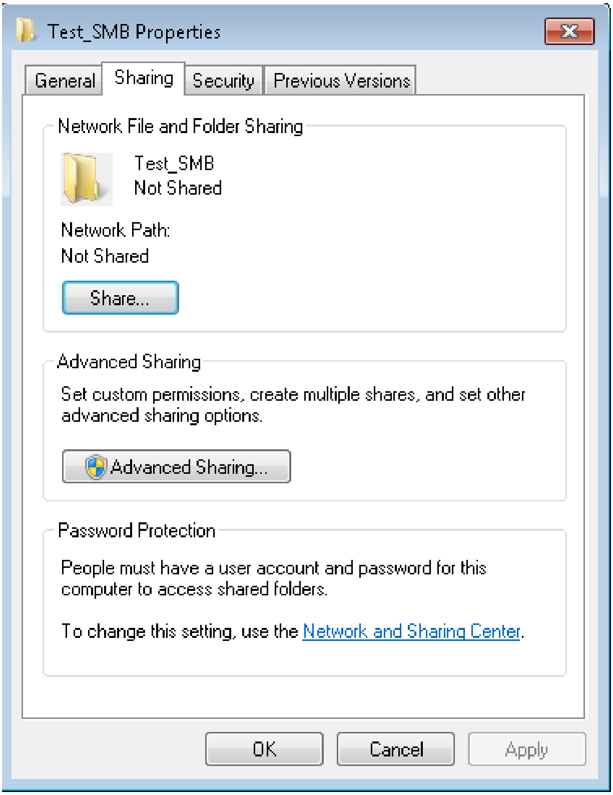
1. Via right- click to the folder (here “Test_SMB”), select “properties”.



Name	Date modified	Type	Size
Test_SMB	5/21/2013 3:31 PM	File folder	

Fig. 321: Create folder to write data and / or images.

2. In the following dialog “Test_SMB Properties” select tab “Sharing” and open “Advanced Sharing”.



Network File and Folder Sharing

Test_SMB
Not Shared

Network Path:
Not Shared

Advanced Sharing

Set custom permissions, create multiple shares, and set other advanced sharing options.

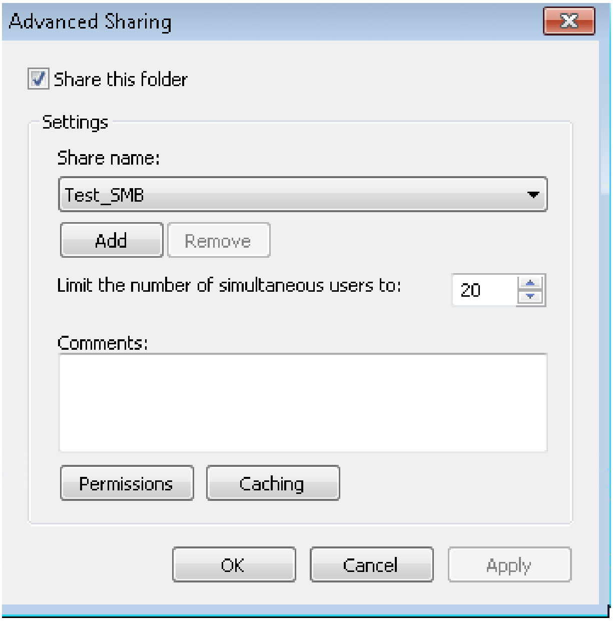
Password Protection

People must have a user account and password for this computer to access shared folders.

To change this setting, use the Network and Sharing Center.

Fig. 322: Sharing of folder > Advanced sharing.

3. In the dialog “Advanced Sharing” activate “Share this folder”. As “Share name” the name of the folder “Test_SMB” is suggested. Here any other name can be set. In this example the suggested folder name is used.
   Important: This “Share name” must be set later in the VISOR®- SMB- Interface!



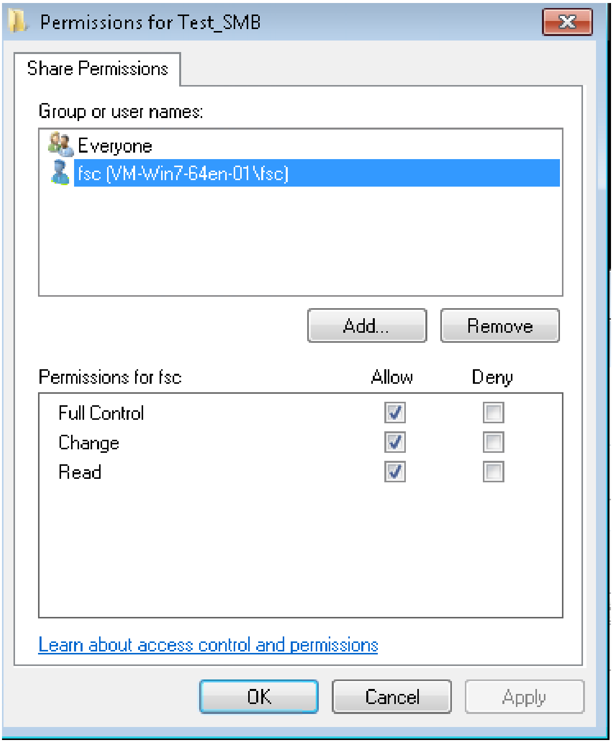
☑ Share this folder

Share name: Test_SMB

Limit the number of simultaneous users to: 20

Fig. 323: Set Share name.

4. With a click to “Permissions” the following dialog appears.



Permissions for Test_SMB

Group or user names:

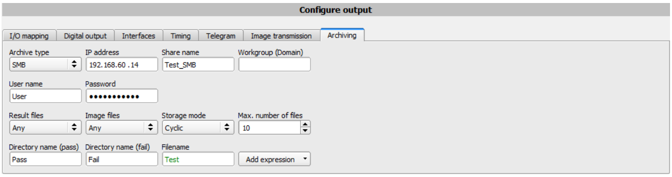
- Everyone
- fsc (VM-Win7-64en-01\fsc)

Permissions for fsc	Allow	Deny
Full Control	☑	□
Change	☑	□
Read	☑	□

Fig. 324: Set permissions

5. In the window “Permissions for Test_SMB”, select a user (here “fsc”), (for which user name and password is known). User name and password are necessary later to be set in the VISOR®- SMB- Interface.
6. Activate “Full control”, and close the dialog with “Apply” and “OK”.
7. Now close the dialog “Advanced Sharing” and “Test_SMB Properties” with “Apply” and “OK” also.
8. The access for the here selected user to the selected folder on the PC now is set, and now the corresponding settings in the VISOR® - Interface “SensoConfig” can be made.

5.1.4.2.2 Settings SMB VISOR®



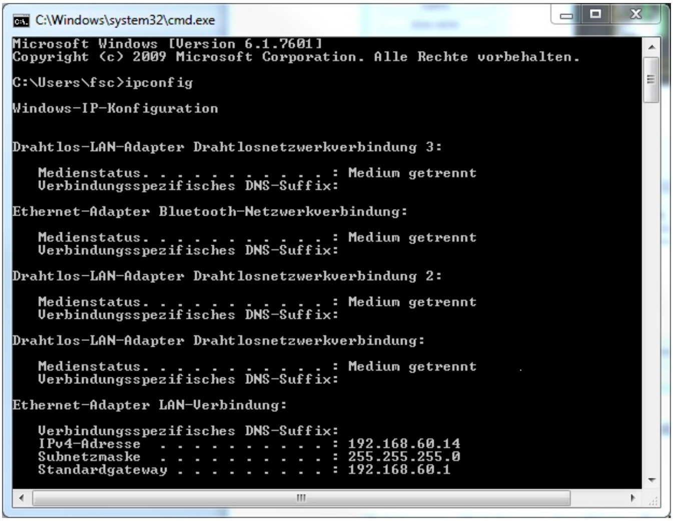
Archive type: SMB
IP address: 192.168.60 .14
Share name: Test_SMB
Workgroup (Domain):
User name: User
Password: ●●●●●●●●●●●
Result files: Any
Image files: Any
Storage mode: Cyclic
Max. number of files: 10
Directory name (pass): Pass
Directory name (fail): Fail
Filename: Test

Fig. 325: Settings in VISOR®- SMB- Interface

After starting SensoConfig, select select Job/Archiving/Archive type: “SMB”.

Do the follwing settings

- IP addresse: IP addresse of PC (this can be found with command “ipconfig” via Start/run-/cmd, s. following screenshot). In this example: 192.168.60.14



```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Alle Rechte vorbehalten.

C:\Users\fsc>ipconfig

Windows-IP-Konfiguration


Drahtlos-LAN-Adapter Drahtlosnetzwerkverbindung 3:

   Medienstatus. . . . . . . . . . . : Medium getrennt
   Verbindungsspezifisches DNS-Suffix:

Ethernet-Adapter Bluetooth-Netzwerkverbindung:

   Medienstatus. . . . . . . . . . . : Medium getrennt
   Verbindungsspezifisches DNS-Suffix:

Drahtlos-LAN-Adapter Drahtlosnetzwerkverbindung 2:

   Medienstatus. . . . . . . . . . . : Medium getrennt
   Verbindungsspezifisches DNS-Suffix:

Drahtlos-LAN-Adapter Drahtlosnetzwerkverbindung:

   Medienstatus. . . . . . . . . . . : Medium getrennt
   Verbindungsspezifisches DNS-Suffix:

Ethernet-Adapter LAN-Verbindung:

   Verbindungsspezifisches DNS-Suffix:
   IPv4-Adresse  . . . . . . . . . . : 192.168.60.14
   Subnetzmaske  . . . . . . . . . . : 255.255.255.0
   Standardgateway . . . . . . . . . : 192.168.60.1
```

Fig. 326: IP- Adresse des PC via Start/Ausführen/cmd/ipconfig

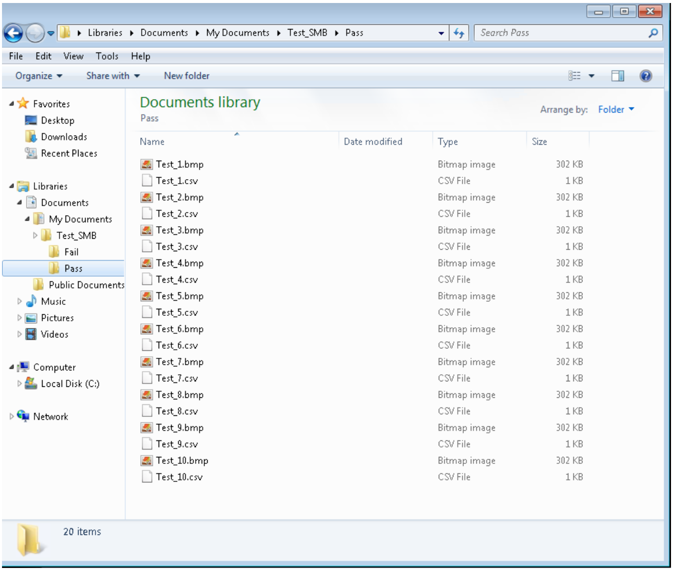
- Share name: Here enter Share name like set in PC- dialog “Advanced Sharing”, Fig.3.
- Workgroup: Option! Name of workgroup.
- User name and Password: Depending on the selection made in dialog “Test_SMB Permissions”:

1. User group “Everyone”: User name and Password remain empty
2. Enter corresponding User name and (here in example User name: “fsc”)

- Directory name (Pass), Directory name (Fail): Chose a name for the folders in which in case of Pass- or Fail- parts the data and images should be archived. (These folders are crated below the shared folder (here: “Test_SMB”).
- Filename: Enter any filename.
- Result files: If protocol file is active, there will be generated automatically a .csv file for each inspection (trigger). Contents of the file is like specified in “Output / Telegram”. Filename with incremented counter.
- Image contents: Possibility to select, whether images should be stored including the selected software filter or “raw” as taken from the camera.
- Storage mode: Limit: after reaching maximum number of files transmission is stopped. Unlimited: files are stored, until target drive is full. Cyclic: after reaching maximum number of files the older files are replaced by the newer ones.
- Max. number of files: Maximum number of filesets (image+ data) which are allowed to be stored in the target directory.

5.1.4.2.3 Archiving via SMB, output data

After starting of the sensor the images and data (as .csv- file), which has been defined under: SensoConfig/Output/Telegram are stored in the corresponding subfolder of the shared folder.



Name	Date modified	Type	Size
Test_1.bmp		Bitmap image	302 KB
Test_1.csv		CSV File	1 KB
Test_2.bmp		Bitmap image	302 KB
Test_2.csv		CSV File	1 KB
Test_3.bmp		Bitmap image	302 KB
Test_3.csv		CSV File	1 KB
Test_4.bmp		Bitmap image	302 KB
Test_4.csv		CSV File	1 KB
Test_5.bmp		Bitmap image	302 KB
Test_5.csv		CSV File	1 KB
Test_6.bmp		Bitmap image	302 KB
Test_6.csv		CSV File	1 KB
Test_7.bmp		Bitmap image	302 KB
Test_7.csv		CSV File	1 KB
Test_8.bmp		Bitmap image	302 KB
Test_8.csv		CSV File	1 KB
Test_9.bmp		Bitmap image	302 KB
Test_9.csv		CSV File	1 KB
Test_10.bmp		Bitmap image	302 KB
Test_10.csv		CSV File	1 KB

Fig. 327: Successful processed data and image archiving via SMB.

5.1.5 Ram disk (on the sensor)

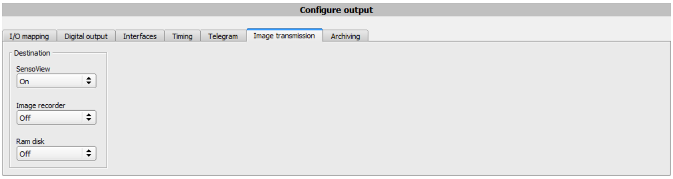
If Ram disk is active, always the according last image and the numeric result data, which have been specified in: “Output/Telegram” (in format .csv) are stored on the sensor in the ram disk folder /tmp/results/.

This function is activated in “Job/Image transmission”.

To access these data an ftp client connection must be established to the sensor.

If:

- SensoConfig/Job/Image transmission/Ram Disk is activated in the VISOR® always the last image (any, pass, failed parts) are stored. File: image.bmp in folder /tmp/results/
- SensoConfig/Output/Telegram data has been specified this are also stored in format .csv, on the VISOR® in folder “/tmp/results”.



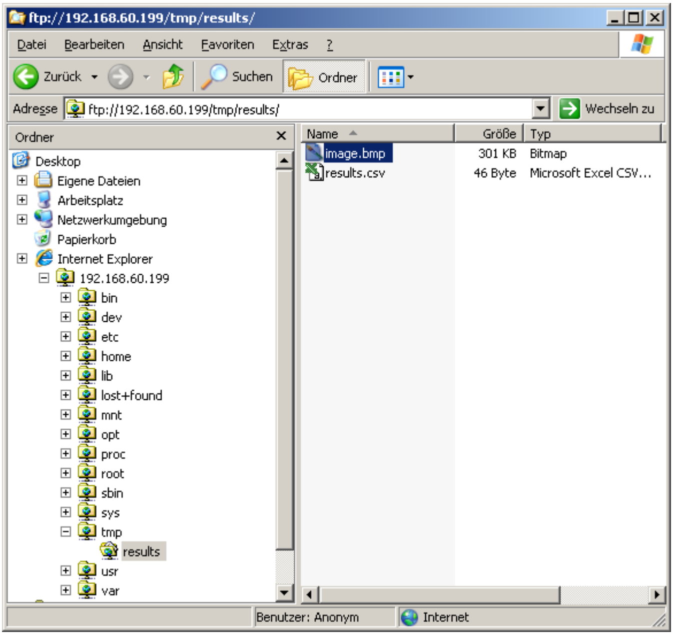
SensoView: On

Image recorder: Off

Ram disk: Off

Fig. 328: Ram Disk

To access this data an ftp client connection like follows e.g. with Windows Explorer is established via: ftp://”IPAdr_VISOR®”/tmp/results.



Name	Größe	Typ
image.bmp	301 KB	Bitmap
results.csv	46 Byte	Microsoft Excel CSV...

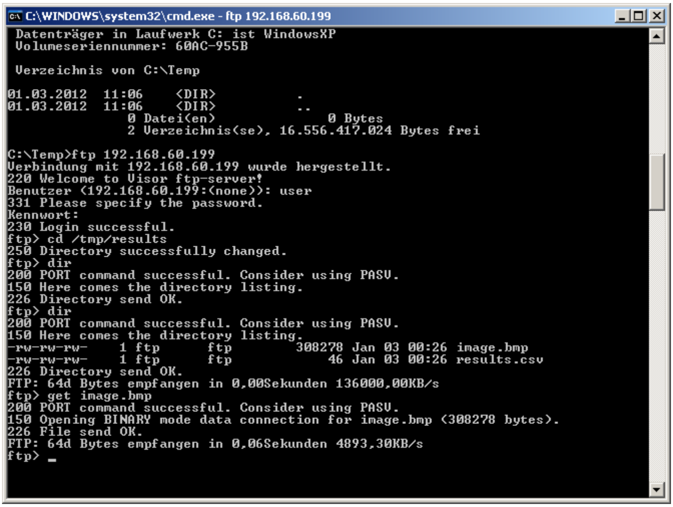
Fig. 329: Ram Disk Sensor via Explorer

A further possibility to access the data on the sensor e.g. is:

Use Windows command “cmd” in Start/Run to open a DOS- window. Process the following commands.

The password in factory setting is “user”.

- First change to the folder on the PC where the data should be stored.
- With ftp “IP_Adr_VISOR®” a connection to VISOR® is established.
- User name: user
- Password: user
- Go to folder: /tmp/results on theVISOR®.
- There are the both files: image.bmp and results.csv (if in Output/Telegram a data string was defined), as image and result data of the latest evaluation.
- With command “get image.bmp”, or. “get results.csv” the files are copied to the selected folder on the PC



```
 Datenträger in Laufwerk C: ist WindowsXP
 Volumeseriennummer: 60AC-955B

 Verzeichnis von C:\Temp

01.03.2012  11:06    <DIR>          .
01.03.2012  11:06    <DIR>          ..
               0 Datei(en)              0 Bytes
               2 Verzeichnis(se), 16.556.417.024 Bytes frei

C:\Temp>ftp 192.168.60.199
Verbindung mit 192.168.60.199 wurde hergestellt.
220 Welcome to Visor ftp-server!
Benutzer (192.168.60.199:(none)): user
331 Please specify the password.
Kennwort:
230 Login successful.
ftp> cd /tmp/results
250 Directory successfully changed.
ftp> dir
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
226 Directory send OK.
ftp> dir
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-rw-rw-    1 ftp      ftp        308278 Jan 03 00:26 image.bmp
-rw-rw-rw-    1 ftp      ftp            46 Jan 03 00:26 results.csv
226 Directory send OK.
FTP: 64d Bytes empfangen in 0,00Sekunden 136000,00KB/s
ftp> get image.bmp
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for image.bmp (308278 bytes).
226 File send OK.
FTP: 64d Bytes empfangen in 0,06Sekunden 4893,30KB/s
ftp> _
```

Fig. 330: Ram Disk via DOS

Attention:

- The format of all .csv files (ftp, smb, ram-disk, SensoView) is always the same.
- The data is readable (by default divided with semicolon comma) stored into the .csv file.
- Only (payload) data, which have been defined under Output/Telegram are transmitted.

5.2 Backup

5.2.1 Backup creation

To save all setting of the sensor, which have been made to check one or some parts, please store all these settings with the command “Save job as ...” or “Save job set ...” in SensoConfig/File. With the commands “Load job ...” or “Load job set ...” these settings can be restored to the sensor later.

5.2.2 Exchange VISOR®

Before exchanging a sensor store all necessarily settings (as described in chapter Backup creation). By exchanging one VISOR® vision sensor against another please consider that the sensors are not calibrated optically or mechanically. That means the new sensor must be: installed mechanically and electrically like described in chapter Installation. And also must be optically focused and set up correctly to work in the network.

After this the in advance stored parameter settings can be restored from the PC to the sensor.

5.3 Job switch

5.3.1 Job switch via digital inputs

To switch between several jobs, which are already stored on the sensor, via digital inputs the following options are available:

Also see chapter: I/O mapping (Page 276), timing diagrams and comments

5.3.1.1 Job 1 or Job 2

To switch between Job1 and Job2 any input can be defined in SensoConfig/Output/I/O mapping with the function “Job 1 or 2”. After the according logical level is connected to this input Job 1 or Job 2 is processed Low = Job1, High = Job 2). Also see chapter: I/O mapping (Page 276) / Function of inputs.

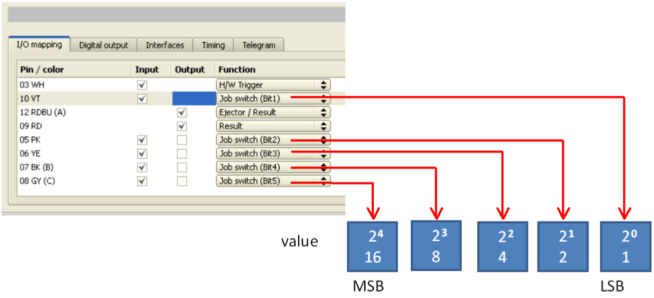
5.3.1.2 Job 1… 31 via binary bit pattern

To switch between up to 31 jobs by binary input pattern via the up to 5 digital inputs, all needed inputs in SensoConfig/Output/I/O mapping are set to the according function “Job switch (Bitx)”.

The in the following graphics shown binary input pattern then switch directly to the according job number. Also see chapter: I/O mapping (Page 276) / Function of inputs.

Attention:

- Job switch starts / happens immediately after the input pattern has changed.
- The display of the active job changes with the first following trigger.
- The mapping of the I/O´s is not fixed. It´s depending on the settings in SensoConfig/Output/I/O mapping.
- The change of the logical level of all related inputs must happen at the same time.



Pin / color	Input	Output	Function
03 WH	☑		H/W Trigger
10 VT	☑		Job switch (Bit1)
12 RDBU (A)		☑	Ejector / Result
09 RD		☑	Result
05 PK	☑	□	Job switch (Bit2)
06 YE	☑	□	Job switch (Bit3)
07 BK (B)	☑	□	Job switch (Bit4)
08 GY (C)	☑	□	Job switch (Bit5)

e.g. Job No.	Output 5	Output 4	Output 3	Output 2	Output 1
1	*false*	*false*	*false*	*false*	*true*
2	*false*	*false*	*false*	*true*	*false*
6	*false*	*false*	*true*	*true*	*false*
13	*false*	*true*	*true*	*false*	*true*
31	*true*	*true*	*true*	*true*	*true*
No change	*false*	*false*	*false*	*false*	*false*

Fig. 331: Job- switch binary

5.3.1.3 Job 1..n via pulses

To switch between job´s with function “Job 1..n” any input can be set up with this function in SensoConfig/Output/I/O mapping. Only possible if Ready = High. After the last impulse (+50ms) Ready is set to low. Impulses are counted until the first delay of >= 50ms and then switches to the appropriate job. Ready remains low until switch-over to the new job occurs. If the option “Job change confirm” is used, this signal occurs after the job change, and hereafter Ready is set high again. During Job Change over binary inputs there must not be sent any trigger signal. Pulse length for job change should be 5 ms pulse and 5 ms delay. Also see chapter: I/O mapping (Page 276) / Function of inputs.

If possible job change should be made by binary coded signals like in chapter Job 1… 31 via binary bit pattern, this is the faster way.

5.3.2 Job switch via Ethernet

See chapter: Ethernet example 2.1 command job switch from PC/PLC to VISOR®

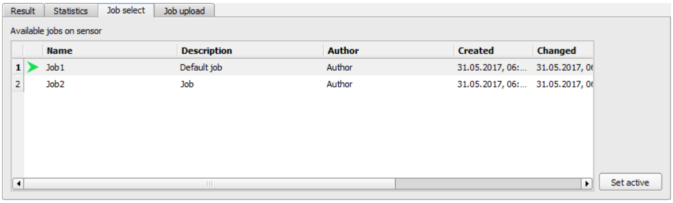
5.3.3 Job switch via Serial

See chapter: RS422 example 1.1: command Job switch from PC / PLC to VISOR®

5.3.4 Job switch via SensoView

In the application SensoView a job switch can be made, or completely new job set’s can be uploaded.

In tab “SensoView/Job” on in the sensor stored jobs are displayed. If there is more than one job in the sensor memory, one of them can be marked in the job list, and activated with button “Set active”. Also see chapter: Changing active job (Page 334).



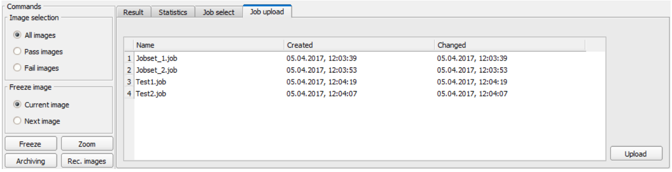
		Name	Description	Author	Created	Changed
1	➤	Job1	Default job	Author	31.05.2017, 06:...	31.05.2017, 0(
2		Job2	Job	Author	31.05.2017, 06:...	31.05.2017, 0(

Fig. 332: SensoView, Job switch

In tab “SensoView/Job upload” all on PC available job set are displayed. This can be marked in the job list and uploaded to the sensor via the button “Upload”.

Attention:

By uploading a new job set all jobs in the sensor memory are deleted.



	Name	Created	Changed
1	Jobset_1.job	05.04.2017, 12:03:39	05.04.2017, 12:03:39
2	Jobset_2.job	05.04.2017, 12:03:53	05.04.2017, 12:03:53
3	Test1.job	05.04.2017, 12:04:19	05.04.2017, 12:04:19
4	Test2.job	05.04.2017, 12:04:07	05.04.2017, 12:04:07

Fig. 333: SensoView, Job upload

5.4 Operation with PLC

5.4.1 Profibus plug adapter (RS422)

Via the Profibus plug adapter the communication between sensor and PLC can be established. This is realized with the RS422 / Profibus converter described in document: “Anybus Profibus operating instruction” in: Startmenue/SensoPart /VISOR® vision sensor/Tools/Anybus Profibus/...

5.4.2 Example Siemens S7

The connection to a Siemens S7 PLC and it´s parameter settings is described in document: “Siemens S7 operating instruction” in: Startmenue/SensoPart /VISOR® vision sensor/Tools/SPS/PLC/...

5.4.3 Example Beckhoff CX 1020

The connection to a Beckhoff CX 1020 and it´s parameter settings is described in document: “Beckhoff operating instruction” in: Startmenue/SensoPart /VISOR® vision sensor/Tools/SPS/PLC/...

5.5 Network connection

5.5.1 Installation of VISOR® into a network / gateway

In SensoFind/Active sensors, all VISOR® sensors, which are installed in the same network segment as the PC which runs SensoFind are displayed as list. To update this list press the button “Find”, to see sensors which e.g. have been powered after SensoFind was started.

For sensors, which are installed in the network, but are located in a different network segment via a gateway, please enter their IP address in the field “Add active sensor” and press button “Add”.

The according sensor now appears in the list “Active sensors” and can be accessed now.

5.5.2 Proceeding/Troubleshooting - Direct Connection

Creating a functioning Ethernet connection between VISOR® vision sensor and PC.

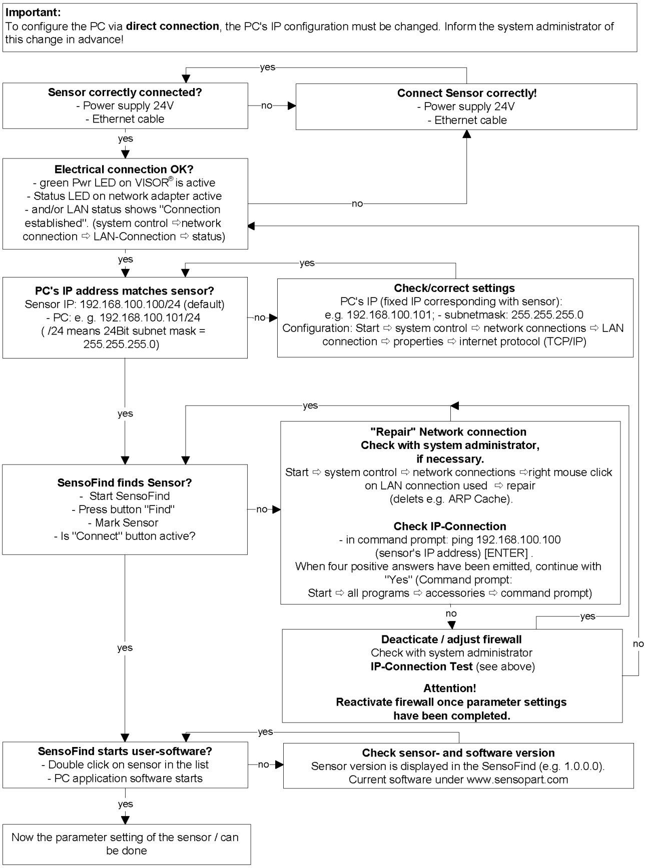
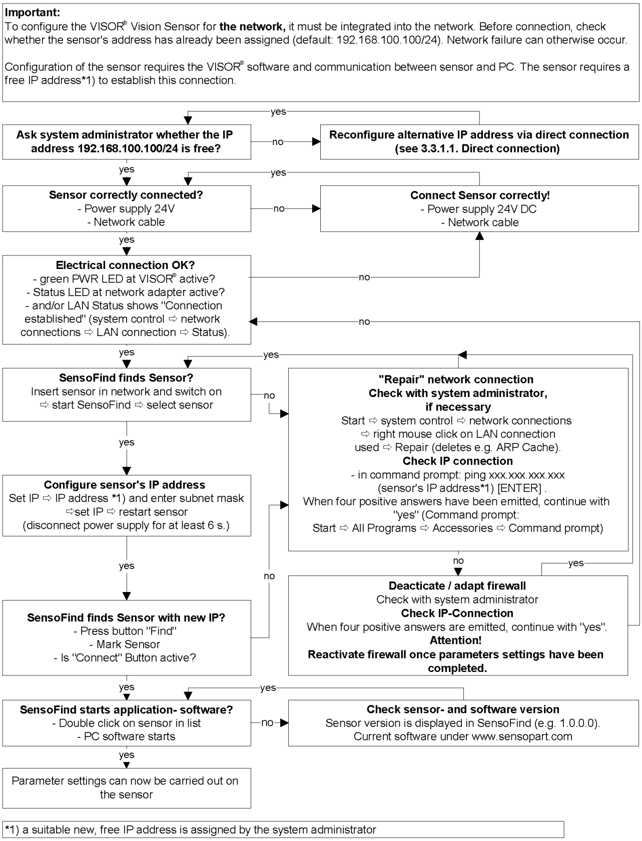

Important:
To configure the PC via direct connection, the PC's IP configuration must be changed. Inform the system administrator of this change in advance!
Sensor correctly connected?
- Power supply 24V
- Ethernet cable
no
yes
Connect Sensor correctly!
- Power supply 24V
- Ethernet cable
yes
Electrical connection OK?
- green Pwr LED on VISOR® is active
- Status LED on network adapter active
- and/or LAN status shows "Connection established". (system control ⇨network connection ⇨ LAN-Connection ⇨ status)
no
yes
yes
PC's IP address matches sensor?
Sensor IP: 192.168.100.100/24 (default)
- PC: e. g. 192.168.100.101/24
(/24 means 24Bit subnet mask = 255.255.255.0)
no
Check/correct settings
PC's IP (fixed IP corresponding with sensor):
e.g. 192.168.100.101; - subnetmask: 255.255.255.0
Configuration: Start ⇨ system control ⇨ network connections ⇨ LAN connection ⇨ properties ⇨ internet protocol (TCP/IP)
yes
yes
"Repair" Network connection
Check with system administrator, if necessary.
Start ⇨ system control ⇨ network connections ⇨right mouse click on LAN connection used ⇨ repair (delets e.g. ARP Cache).
Check IP-Connection
- in command prompt: ping 192.168.100.100 (sensor's IP address) [ENTER] .
When four positive answers have been emitted, continue with "Yes" (Command prompt: Start ⇨ all programs ⇨ accessories ⇨ command prompt)
SensoFind finds Sensor?
- Start SensoFind
- Press button "Find"
- Mark Sensor
- Is "Connect" button active?
no
no
yes
Deacticate / adjust firewall
Check with system administrator
IP-Connection Test (see above)
Attention!
Reactivate firewall once parameter settings have been completed.
no
yes
yes
SensoFind starts user-software?
- Double click on sensor in the list
- PC application software starts
no
Check sensor- and software version
Sensor version is displayed in the SensoFind (e.g. 1.0.0.0).
Current software under www.sensopart.com
yes
Now the parameter setting of the sensor / can be done

Fig. 334: Direct connection sensor / PC, proceeding / troubleshooting

5.5.3 Proceeding/Troubleshooting - Network Connection

Establishing an operational Ethernet connection between VISOR® vision sensor and PC



Important:
To configure the VISOR® Vision Sensor for **the network,** it must be integrated into the network. Before connection, check whether the sensor's address has already been assigned (default: 192.168.100.100/24). Network failure can otherwise occur.

Configuration of the sensor requires the VISOR® software and communication between sensor and PC. The sensor requires a free IP address*1) to establish this connection.

Ask system administrator whether the IP address 192.168.100.100/24 is free?

Reconfigure alternative IP address via direct connection (see 3.3.1.1. Direct connection)

Sensor correctly connected?
- Power supply 24V
- Network cable

Connect Sensor correctly!
- Power supply 24V DC
- Network cable

Electrical connection OK?
- green PWR LED at VISOR® active?
- Status LED at network adapter active?
- and/or LAN Status shows "Connection established" (system control ⇨ network connections ⇨ LAN connection ⇨ Status).

SensoFind finds Sensor?
Insert sensor in network and switch on ⇨ start SensoFind ⇨ select sensor

"Repair" network connection
Check with system administrator,
if necessary
Start ⇨ system control ⇨ network connections ⇨ right mouse click on LAN connection used ⇨ Repair (deletes e.g. ARP Cache).
Check IP connection
- in command prompt: ping xxx.xxx.xxx.xxx (sensor's IP address*1) [ENTER] .
When four positive answers have been emitted, continue with "yes" (Command prompt:
Start ⇨ All Programs ⇨ Accessories ⇨ Command prompt)

Configure sensor's IP address
Set IP ⇨ IP address *1) and enter subnet mask ⇨set IP ⇨ restart sensor
(disconnect power supply for at least 6 s.)

Deacticate / adapt firewall
Check with system administrator
Check IP-Connection
When four positive answers are emitted, continue with "yes".
Attention!
Reactivate firewall once parameters settings have been completed.

SensoFind finds Sensor with new IP?
- Press button "Find"
- Mark Sensor
- Is "Connect" Button active?

SensoFind starts application- software?
- Double click on sensor in list
- PC software starts

Check sensor- and software version
Sensor version is displayed in SensoFind (e.g. 1.0.0.0).
Current software under www.sensopart.com

Parameter settings can now be carried out on the sensor

*1) a suitable new, free IP address is assigned by the system administrator

Fig. 335: Connection via network sensor / PC, proceeding / troubleshooting

5.5.4 Used Ethernet- Ports

If the VISOR® should be installed into a network, the following ports must be enabled, if so by the network administrator. This is necessary only in case that this specific ports have been locked e.g. in a company network by a firewall installed on a PC.

To communicate between a PC fur configuration and the VISOR® the following ports are used:

- Port 80, TCP (SensoWeb)
- Port 2000, TCP
- Port 2001, UDP Broadcast (to find sensors via SensoFind)
- Port 2002, TCP
- Port 2003, TCP
- Port 2004, TCP

To communicate between PLC (PLC- PC also) and the VISOR® the following ports are used.

- Port 2005, TCP (Implicit results, that means, user configured result data)
- Port 2006, TCP (Explicit requests, e.g. trigger or job switch)

If the ports 2005 or 2006 are changed in SensoConfig, the according ports also must be enabled in the firewall by the administrator.

If the interface EtherNet/IP is used the following two ports must be enabled too.

- Port 2222, UDP (EtherNet/IP)
- Port 44818, TCP (EtherNet/IP)

5.5.5 Access to VISOR® via network

Exemplary values for IP etc.

Access to VISOR® 1 from PC 1, if in same subnet.

- Via SensoFind (/find)

Access to VISOR® 2 from PC 1, if in different subnet.

Only if:

- Gateway is set correct in sensor 2 (here to 192.168.30.1) - and
- in SensoFind via Add- IP- the sensor IP of sensor 2 is set correct

Now the VISOR® 2 appears in the list “Active Sensors” in SensoFind!

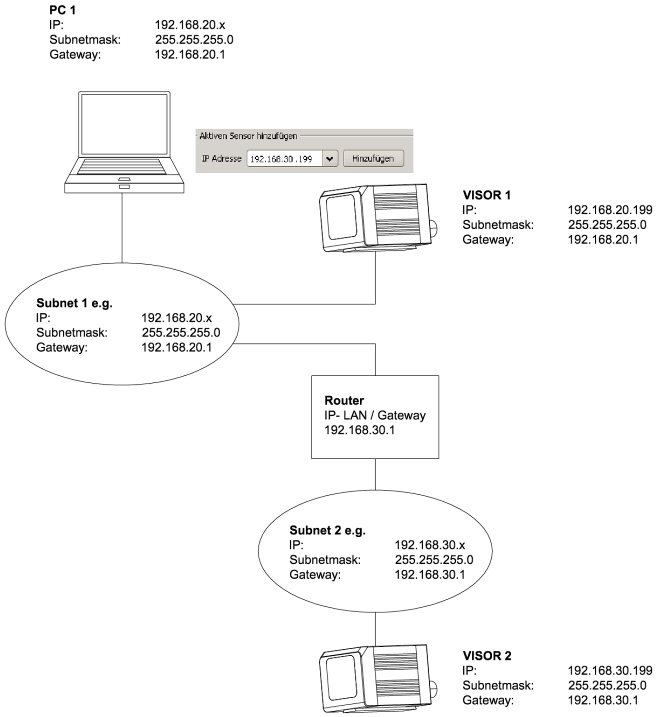


Fig. 336: Access to VISOR® via network, same or different subnet

5.5.6 Access to VISOR® via Internet / World Wide Web

Exemplary values for IP etc.

Access from PC 1 (company network 1), via Word Wide Web, into company network 2 to VISOR® 1.

- Add the IP- WAN of router 2 (company network 2) in PC1 (company network 1) in SensoFind under “Add active sensor” (here in example: 62.75.148.101).

In router 2 the ports which should be used by the sensor must be defined. (see also chapter: Used Ethernet- Ports (Page 379)).

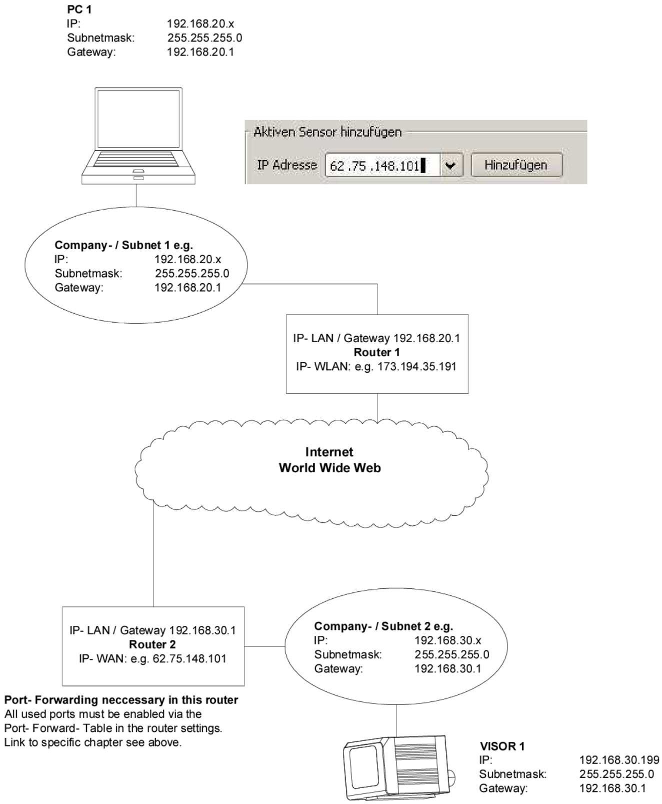

PC 1
IP: 192.168.20.x
Subnetmask: 255.255.255.0
Gateway: 192.168.20.1
Aktiven Sensor hinzufügen
IP Adresse 62 .75 .148.101
Hinzufügen
Company- / Subnet 1 e.g.
IP: 192.168.20.x
Subnetmask: 255.255.255.0
Gateway: 192.168.20.1
IP- LAN / Gateway 192.168.20.1
Router 1
IP- WLAN: e.g. 173.194.35.191
Internet
World Wide Web
IP- LAN / Gateway 192.168.30.1
Router 2
IP- WAN: e.g. 62.75.148.101
Company- / Subnet 2 e.g.
IP: 192.168.30.x
Subnetmask: 255.255.255.0
Gateway: 192.168.30.1
Port- Forwarding neccessary in this router
All used ports must be enabled via the
Port- Forward- Table in the router settings.
Link to specific chapter see above.
VISOR 1
IP: 192.168.30.199
Subnetmask: 255.255.255.0
Gateway: 192.168.30.1

Fig. 337: Access to VISOR® via Internet / World Wide Web

5.6 VISOR® vision sensor, PROFINET, Introduction

This chapter explains the use of SensoPart VISOR® vision sensor with PROFINET. The PROFINET interface is implemented starting with version 1.12.x.x.

For data communication between VISOR® vision sensor and PLC via PROFINET the following topics are explained: electrical connection, settings in VISOR® vision sensor and PLC (as example for Siemens S7), available telegrams formats and the telegram timing.

5.6.1 Electrical connection VISOR® in the PROFINET network

The VISOR® vision sensor is connected via a Ethernet TCP/IP connection and a PROFINET switch to the network, and so to the PROFINET environment.

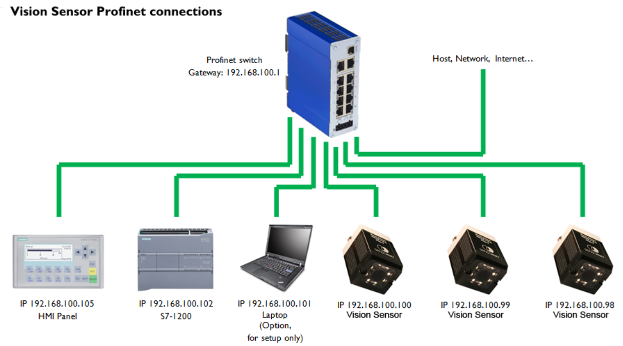
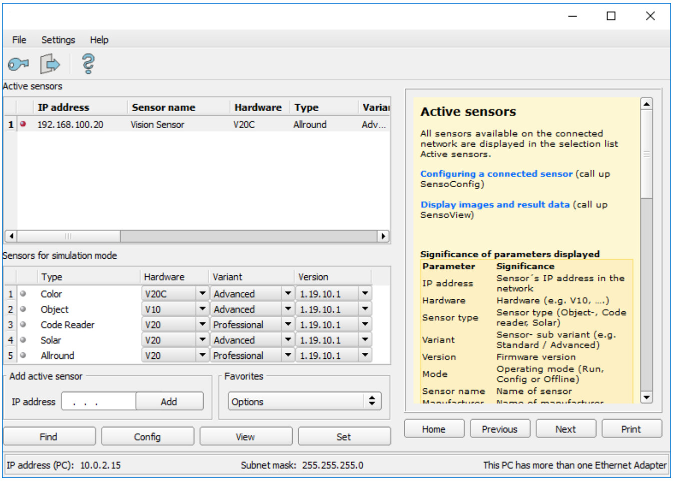


Fig. 338: Connection of VISOR® via PROFINET switch

5.6.2 Configuration of VISOR® via SensoPart SensoConfig for the use with PROFINET

In this example the configuration of VISOR® V20 CR Advanced is described. For all other types the configuration works analogue.

5.6.2.1 Settings in SensoFind



Active sensors

	IP address	Sensor name	Hardware	Type	Variai
1	192.168.100.20	Vision Sensor	V20C	Allround	Adv...

Sensors for simulation mode

	Type	Hardware	Variant	Version
1	Color	V20C	Advanced	1.19.10.1
2	Object	V10	Advanced	1.19.10.1
3	Code Reader	V20	Professional	1.19.10.1
4	Solar	V20	Advanced	1.19.10.1
5	Allround	V20	Professional	1.19.10.1

Add active sensor

IP address . . .

Favorites

Active sensors

All sensors available on the connected network are displayed in the selection list Active sensors.

Configuring a connected sensor (call up SensoConfig)

Display images and result data (call up SensoView)

Significance of parameters displayed

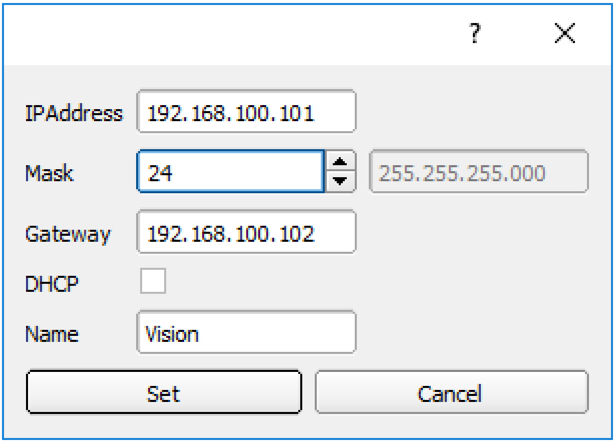
Parameter	Significance
IP address	Sensor´s IP address in the network
Hardware	Hardware (e.g. V10, ….)
Sensor type	Sensor type (Object-, Code reader, Solar)
Variant	Sensor- sub variant (e.g. Standard / Advanced)
Version	Firmware version
Mode	Operating mode (Run, Config or Offline)
Sensor name	Name of sensor
Manufacturer	Name of manufacturer

IP address (PC): 10.0.2.15 Subnet mask: 255.255.255.0 This PC has more than one Ethernet Adapter

Fig. 339: VISOR® is displayed and selected in SensoFind

At the start of SensoFind or by click to the button “Find” the sensor is listed in window “Active sensors”. By click to the button “Set” the following dialog starts.

5.6.2.2 Setting of IP and name



IPAddress	192.168.100.101	
Mask	24	255.255.255.000
Gateway	192.168.100.102	
DHCP	□	
Name	Vision	

Fig. 340: Setting of IP and name

Here the IP address of the VISOR® and it’s name is set.

If here a name is set which shall be used permanently, the identical name must be used in the PLC as well.

Caution: these settings are active not before a reboot of the sensor.

Close this dialog with “Set”

Important conditions:

Independent from which possibility is used to do the settings, for a properly working PROFINET communication it’s neccessary:

- The VISOR® name must be identical in PLC and sensor
- The IP address of VISOR® and PLC must correspond (same address range)

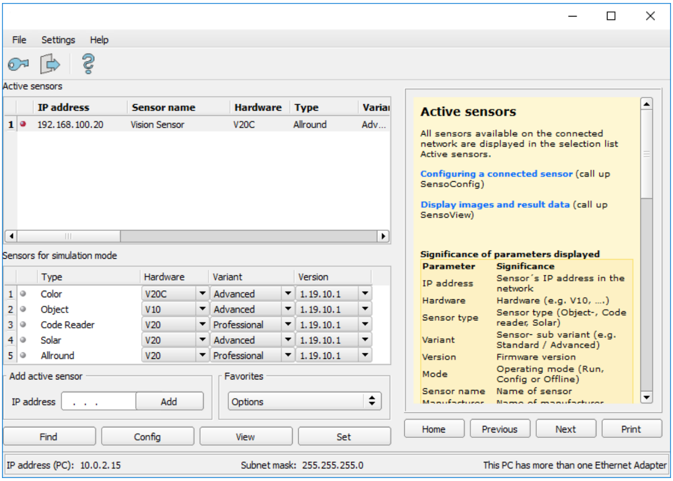
IP address and name of the VISOR® can be set in different ways:

- Either via VISOR® software SensoFind, or
- Via PLC interface, here Siemens TIA.

The name must be DNS compatible. That means:

- Hostnames may only consist of the characters 'a'-'z', '0'-'9', '-' and '.'. (lower case only)
- The Character '.' may just occure as divider between labels in domain names.
- The character '-' may not occure as first or last character

Setting a name via SensoFind please take care to meet the above mentioned DNS conventions, as they are not checked. Via the input in the TIA PLC interface the names are converted automatically. See chapter: Set the name with TIA interface (Page 392)



Active sensors

		IP address	Sensor name	Hardware	Type	Variai
1	●	192.168.100.20	Vision Sensor	V20C	Allround	Adv...

Sensors for simulation mode

		Type	Hardware	Variant	Version
1	●	Color	V20C	Advanced	1.19.10.1
2	●	Object	V10	Advanced	1.19.10.1
3	●	Code Reader	V20	Professional	1.19.10.1
4	●	Solar	V20	Advanced	1.19.10.1
5	●	Allround	V20	Professional	1.19.10.1

Add active sensor

IP address: . . .

Favorites

IP address (PC): 10.0.2.15 Subnet mask: 255.255.255.0 This PC has more than one Ethernet Adapter

Active sensors

All sensors available on the connected network are displayed in the selection list Active sensors.

Configuring a connected sensor (call up SensoConfig)

Display images and result data (call up SensoView)

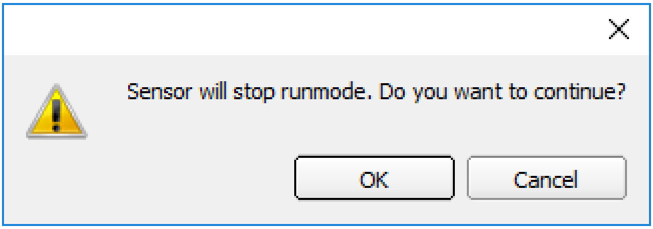
Significance of parameters displayed

Parameter	Significance
IP address	Sensor´s IP address in the network
Hardware	Hardware (e.g. V10, ...)
Sensor type	Sensor type (Object-, Code reader, Solar)
Variant	Sensor- sub variant (e.g. Standard / Advanced)
Version	Firmware version
Mode	Operating mode (Run, Config or Offline)
Sensor name	Name of sensor
Manufacturer	Name of manufacturer

Fig. 341: IP and name has been updated

5.6.2.3 Open SensoConfig

With click to “Config” in SensoFind, and to “OK” in the following dialog SensoConfig starts.

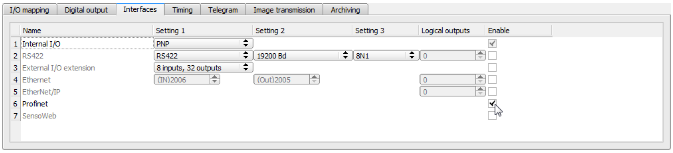


Sensor will stop runmode. Do you want to continue?

Fig. 342: Open SensoConfig

5.6.2.4 Select Interface “PROFINET”

In Output/Interface/PROFINET via the checkbox the PROFINET interface is selected. By this command the PROFINET stack gets started.



	Name	Setting 1	Setting 2	Setting 3	Logical outputs	Enable
1	Internal I/O	PNP				☑
2	RS422	RS422	19200 Bd	8N1	0	□
3	External I/O extension	8 inputs, 32 outputs				□
4	Ethernet	(IN)2006	(Out)2005		0	□
5	EtherNet/IP				0	□
6	Profinet					☑
7	SensoWeb					□

Fig. 343: Activation of PROFINET in SensoConfig

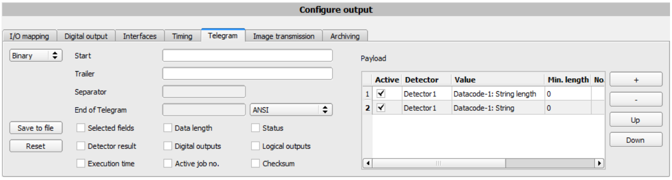
5.6.2.5 Definition of the telegram

In the tab “Telegram” the data which should be transferred can be defined completely free. For the use with PROFINET this must be done with format “Binary”.

5.6.2.5.1 Definition of the output data

The output data itself are configured identically as the data output via Ethernet TCP/IP or RS422 in: SensoConfig/Output/Telegram.

The description you find in the VISOR® user manual in chapter Telegram, Data output (Page 296) under: SensoConfig/Help/Manual.



Configure output

Binary

Start:

Trailer:

Separator:

End of Telegram: ANSI

□ Selected fields □ Data length □ Status
□ Detector result □ Digital outputs □ Logical outputs
□ Execution time □ Active job no. □ Checksum

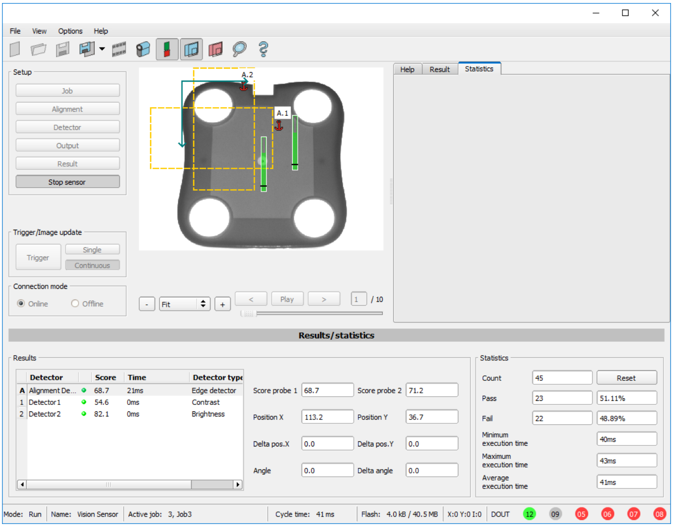
Payload

	Active	Detector	Value	Min. length	No.
1	☑	Detector1	Datacode-1: String length	0	
2	☑	Detector1	Datacode-1: String	0	

Fig. 344: Data output, protocol: Binary

5.6.2.6 Start sensor, data output

With “Start sensor” the configuration data are transferred to the VISOR®. The sensor gets started and now the output data are transferred as defined.



Results

	Detector		Score	Time	Detector type
A	Alignment De...	●	68.7	21ms	Edge detector
1	Detector1	●	54.6	0ms	Contrast
2	Detector2	●	82.1	0ms	Brightness

Score probe 1	68.7	Score probe 2	71.2
Position X	113.2	Position Y	36.7
Delta pos.X	0.0	Delta pos.Y	0.0
Angle	0.0	Delta angle	0.0

Statistics

Count	45	
Pass	23	51.11%
Fail	22	48.89%
Minimum execution time		40ms
Maximum execution time		43ms
Average execution time		41ms

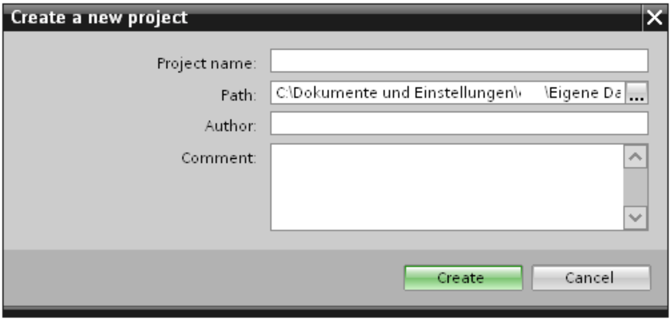
Mode: Run | Name: Vision Sensor | Active job: 3, Job3 | Cycle time: 41 ms | Flash: 4.0 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT 12 09 05 06 07 08

Fig. 345: Start sensor

5.6.3 PROFINET configuration of PLC, example Siemens S7-1200 TIA

5.6.3.1 Create a new project

New project with: Project/Create new project



Create a new project

Project name:

Path: C:\Dokumente und Einstellungen\ \Eigene Da

Author:

Comment:

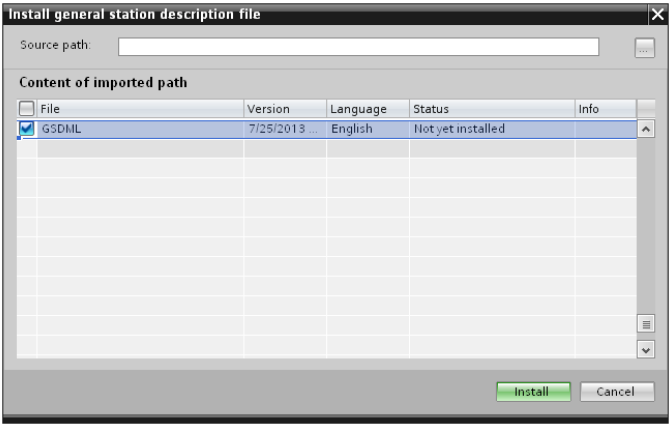
Fig. 346: Create new project

5.6.3.2 Select GSD file

First a PROFINET PLC must be added to the project.

To use the PROFINET functions of the VISOR®, the GSD file for the VISOR® must be installed in it´s latest version. This is done at: Options/Install general station description file.

The GSD file is available in the installation path of VISOR®: ...\SensoPart \VISOR® vision sensor\Tools\PROFINET, and as download at www.sensopart.com.



Install general station description file

Source path:

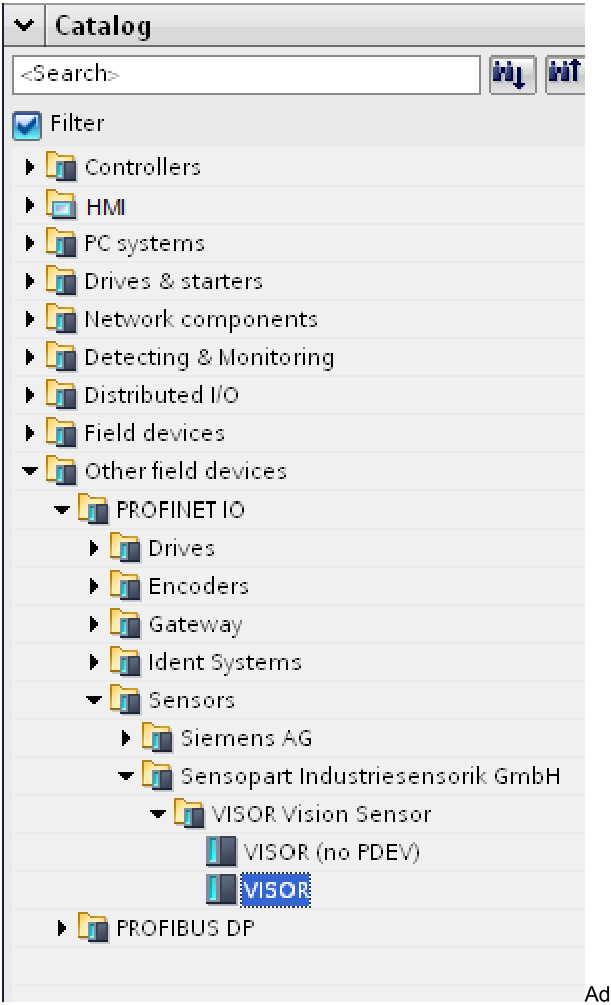
Content of imported path

□	File	Version	Language	Status	Info
☑	GSDML	7/25/2013 ...	English	Not yet installed	

Fig. 347: Select and install GSD file

5.6.3.3 Adding VISOR® to Project

The VISOR® modules are added in the hardware catalog: Other field devices/PROFINET IO/Sensors/SensoPart Industriesensorik GmbH.



Catalog

<Search>

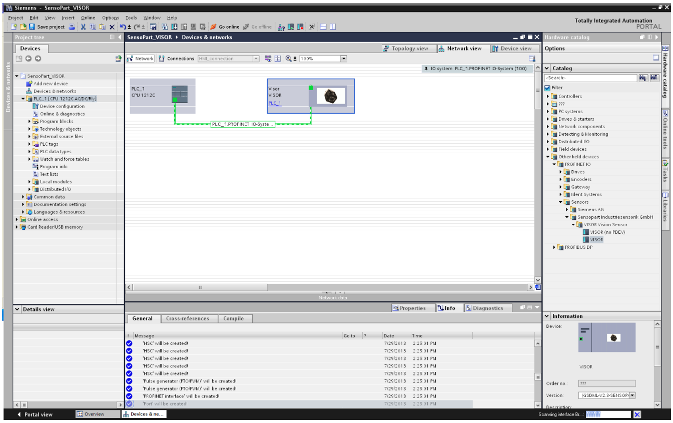
☑ Filter

- Controllers
- HMI
- PC systems
- Drives & starters
- Network components
- Detecting & Monitoring
- Distributed I/O
- Field devices
- Other field devices
  - PROFINET IO
    - Drives
    - Encoders
    - Gateway
    - Ident Systems
    - Sensors
      - Siemens AG
      - Sensopart Industriesensorik GmbH
        - VISOR Vision Sensor
          - VISOR (no PDEV)
          - VISOR
  - PROFIBUS DP

Fig. 348: Add VISOR® to the project

5.6.3.4 Connect VISOR® to PLC

With drag and drop a VISOR® module can be put into the Network view. Now connect the VISOR® via PROFINET to the PLC (Tab. Network view).



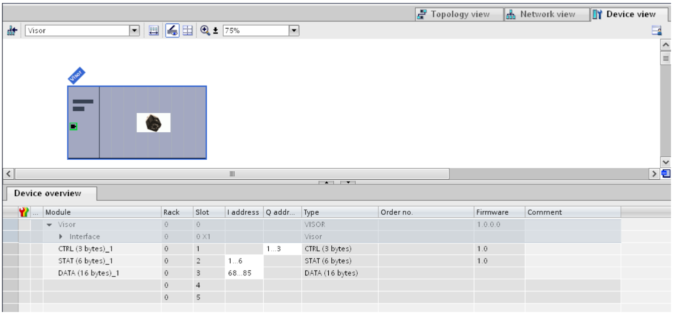
!	Message	Go to	?	Date	Time
✓	'HSC' will be created!			7/29/2013	2:25:01 PM
✓	'HSC' will be created!			7/29/2013	2:25:01 PM
✓	'HSC' will be created!			7/29/2013	2:25:01 PM
✓	'HSC' will be created!			7/29/2013	2:25:01 PM
✓	'HSC' will be created!			7/29/2013	2:25:01 PM
✓	'Pulse generator (PTO/PWM)' will be created!			7/29/2013	2:25:01 PM
✓	'Pulse generator (PTO/PWM)' will be created!			7/29/2013	2:25:01 PM
✓	'PROFINET interface' will be created!			7/29/2013	2:25:01 PM
✓	'Port' will be created!			7/29/2013	2:25:01 PM

Fig. 349: Connect VISOR® to PLC

5.6.3.5 Definition of I/O data

In the tab “Device view” as default the modules CTRL (Control) and STAT (Status) are active. As an option the module DATA (Data module) can be added with a certain size of payload.

In the example: 2 Byte + 16 Byte payload (1Byte: Image ID; 1Byte: Result data overrun (see Module 3: “Data” (From VISOR® to PLC) (Page 402)), + 16 Byte payload data): If the data is longer than the defined range the payload is truncated (in this case: Result data overrun = 1), if it´s shorter the rest of the 16 byte are filled with 00h.



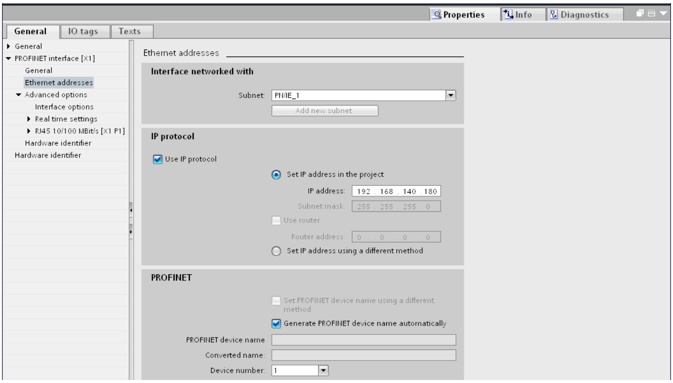
Device overview

Module	Rack	Slot	I address	Q addr...	Type	Order no.	Firmware	Comment
Visor	0	0			VISOR		1.0.0.0	
Interface	0	0 X1			Visor			
CTRL (3 bytes)_1	0	1		1...3	CTRL (3 bytes)		1.0	
STAT (6 bytes)_1	0	2	1...6		STAT (6 bytes)		1.0	
DATA (16 bytes)_1	0	3	68...85		DATA (16 bytes)			
	0	4						
	0	5						

Fig. 350: Define I/O data

5.6.3.6 Set IP address of VISOR® in the project (Option 1)

The IP address of the VISOR® can be set via the project. Select option “Set IP address in the project” and set IP address. Address from the field “IP address” is written into the VISOR®. The IP address of the PLC and of the VISOR® must not be the same, but must correspond, what means they have to be in the same address range.



Ethernet addresses

Interface networked with

Subnet: PN/IE_1

IP protocol

☑ Use IP protocol

● Set IP address in the project

IP address: 192 . 168 . 140 . 180

Subnet mask: 255 . 255 . 255 . 0

□ Use router

Router address: 0 . 0 . 0 . 0

○ Set IP address using a different method

PROFINET

□ Set PROFINET device name using a different method

☑ Generate PROFINET device name automatically

PROFINET device name

Converted name:

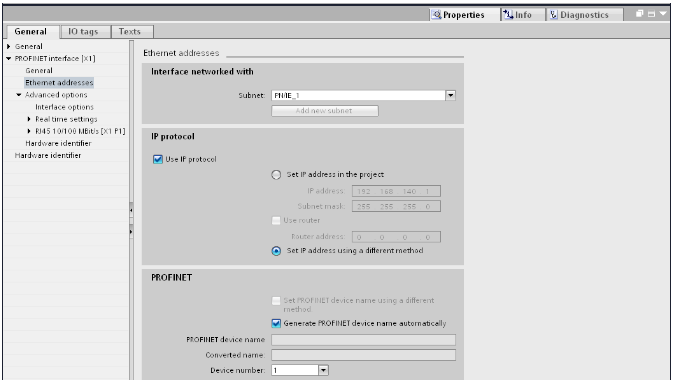
Device number: 1

Fig. 351: Set IP address in project

The VISOR® can be used without a started project also, and so can be configured via SensoFind. If the IP address of the VISOR® does not correspond to the one in the TIA project, the PLC is setting a IP address. In this case the IP address of the VISOR® is overwritten with 0.0.0.0. That means that the IP address is set correctly, but the IP configuration is deleted (this is important for a restart without a connected PLC).

5.6.3.7 Set IP Address with SensoFind (Option 2)

The IP address of the VISOR® can be set also via SensoFind. Select option “Set IP Address using a different method” in the PLC / TIA Interface, and set IP address via SensoFind (See chapter:Setting of IP and name (Page 384)).



Ethernet addresses

Interface networked with

Subnet: PN/IE_1

IP protocol

☑ Use IP protocol

○ Set IP address in the project

IP address: 192 . 168 . 140 . 1

Subnet mask: 255 . 255 . 255 . 0

□ Use router

Router address: 0 . 0 . 0 . 0

● Set IP address using a different method

PROFINET

□ Set PROFINET device name using a different method.

☑ Generate PROFINET device name automatically

PROFINET device name

Converted name:

Device number: 1

Fig. 352: Set IP address via SensoFind, settings therefor in the PLC/TIA interface

5.6.3.8 Set the name with TIA interface

To set the name of VISOR® from TIA interface there are two options.

5.6.3.8.1 Generate name automatically

The PROFINET name of the VISOR® can be generated automatically from the PLC. Option: “Generate PROFINET device name automatically” takes the name from the project. This name originally comes from the GSD- file.

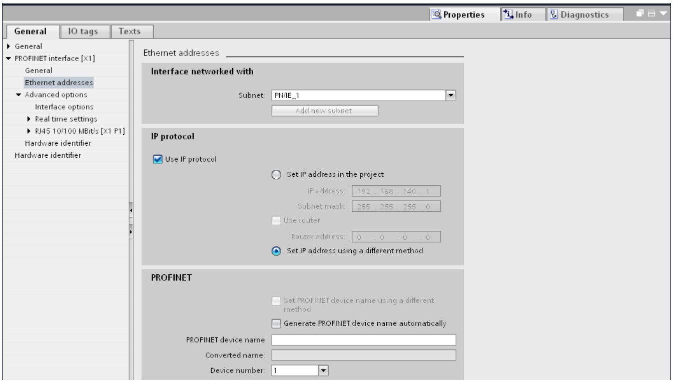
5.6.3.8.2 Set name manually

If the option “Set PROFINET device name using a different method” is selected any name can be set.

Information: In the field “Converted name” a different name may be shown than the one edited, as with PROFINET not all characters can be used a conversion may be necessary and is done automatically (names must be DNS compatible, see also chapter: 3.2).

If a name for the VISOR® is set via this option, in each case it must be written to the sensor with the “Assign PROFINET device name”-Tool (as described in chapter 4.9).

The PROFINET name in the project and in the VISOR® must be the same.



Ethernet addresses

Interface networked with

Subnet: PN/IE_1

Add new subnet

IP protocol

☑ Use IP protocol

○ Set IP address in the project

IP address: 192 . 168 . 140 . 1

Subnet mask: 255 . 255 . 255 . 0

□ Use router

Router address: 0 . 0 . 0 . 0

● Set IP address using a different method

PROFINET

□ Set PROFINET device name using a different method

□ Generate PROFINET device name automatically

PROFINET device name:

Converted name:

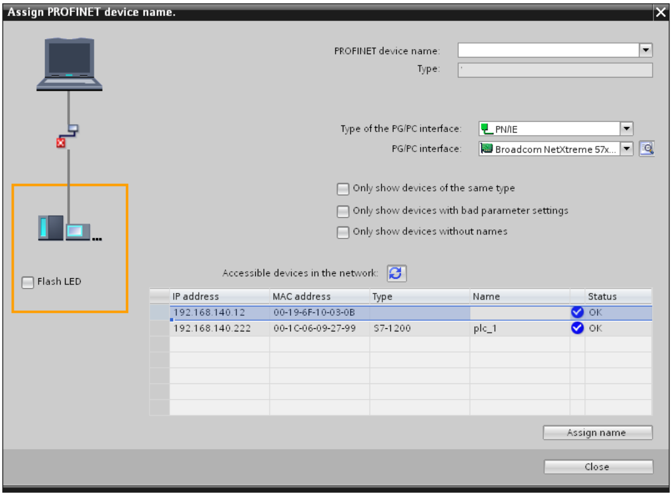
Device number: 1

Fig. 353: Set name in project

5.6.3.9 Write name into VISOR®

In case that the PROFINET name in the VISOR® has to be updated, it´s necessary to write the name into the sensor to establish a communication.

This is done with the tool: Online/Assign PROFINET device name. Select the device in the list (VISOR®) and with “Assign name” the name is written into the sensor.



PROFINET device name:

Type:

Type of the PG/PC interface: PN/IE

PG/PC interface: Broadcom NetXtreme 57x...

□ Only show devices of the same type

□ Only show devices with bad parameter settings

□ Only show devices without names

□ Flash LED

Accessible devices in the network:

IP address	MAC address	Type	Name	Status
192.168.140.12	00-19-6F-10-03-0B			OK
192.168.140.222	00-1C-06-09-27-99	S7-1200	plc_1	OK

Fig. 354: Write name into VISOR®

5.6.3.10 Translate project and write to PLC

To finish the configuration and save changes of the project: 1. translate and 2. transfer / write to the PLC

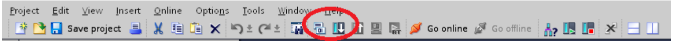
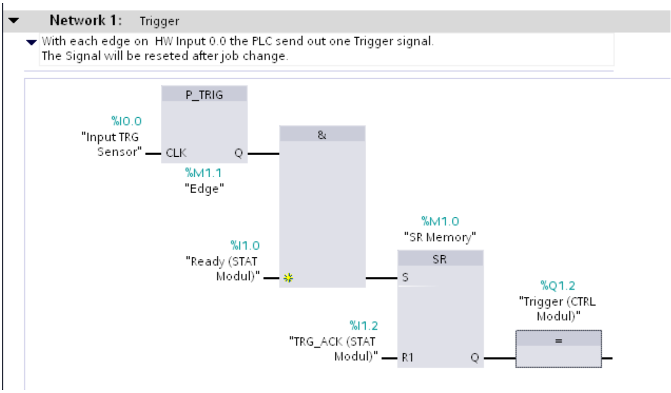


Fig. 355: Translate project and write to PLC

5.6.4 PLC Examples, PROFINET

The following PLC example programs show some basic functions.

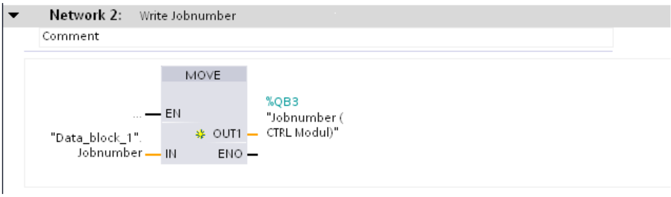
5.6.4.1 PLC Example 1: Trigger when VISOR® Ready


Network 1: Trigger

With each edge on HW Input 0.0 the PLC send out one Trigger signal.
The Signal will be reseted after job change.

Fig. 356: Trigger when VISOR® Ready, (without error handling)

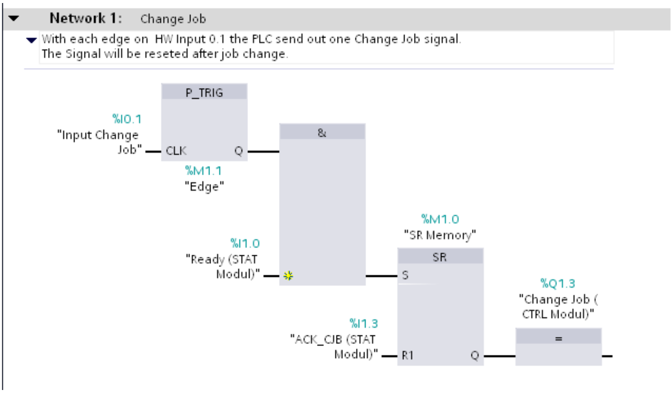
5.6.4.2 SPS Example 2: Send Job number to VISOR®


Network 2: Write Jobnumber

Comment

Fig. 357: Send Job number

5.6.4.2.1 PLC Example 2.1: Job change when VISOR® Ready

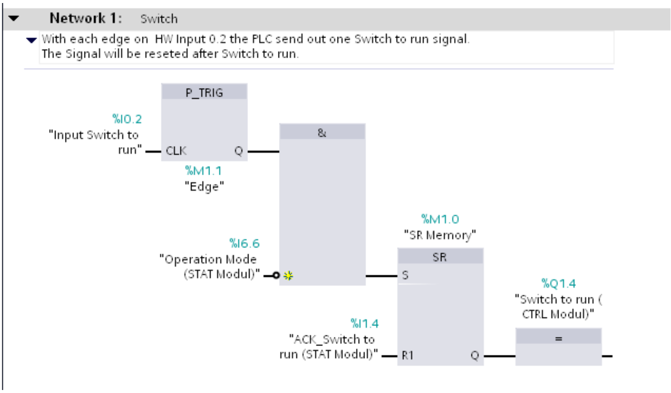


Network 1: Change Job

With each edge on HW Input 0.1 the PLC send out one Change Job signal.
The Signal will be reseted after job change.

Fig. 358: Job change when VISOR® Ready , (without error handling)

5.6.4.3 PLC Example 3: Switch to Run when VISOR® in configuration mode



Network 1: Switch

With each edge on HW Input 0.2 the PLC send out one Switch to run signal.
The Signal will be reseted after Switch to run.

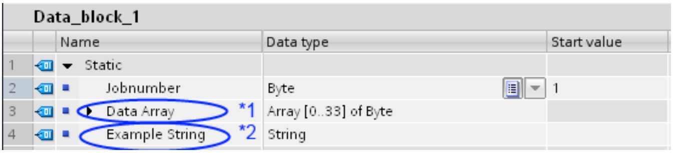
P_TRIG
%I0.2
"Input Switch to run"
CLK
Q
%M1.1
"Edge"
&
%I6.6
"Operation Mode (STAT Modul)"
%M1.0
"SR Memory"
SR
S
%I1.4
"ACK_Switch to run (STAT Modul)"
R1
Q
%Q1.4
"Switch to run (CTRL Modul)"
=

Fig. 359: Switch to Run when VISOR® in configuration mode (without error handling)

5.6.4.4 PLC Example 4, Data transfer PLC data module, Set variables

Variable “Data Array” (Type: Array of Byte) Length(34 Byte) = Payload(32) + 2Byte (Header)

(Module “Data” with 32 Byte: Payload + 1 Byte: Image ID + 1 Byte: Result data overrun = 34 Byte)



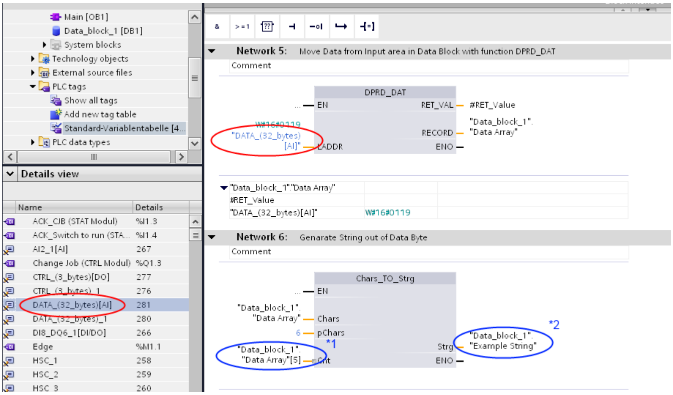
Data_block_1

	Name	Data type	Start value
1	▼ Static		
2	Jobnumber	Byte	1
3	▶ Data Array *1	Array [0..33] of Byte	
4	Example String *2	String	

Fig. 360: Data modules for data transfer

5.6.4.4.1 PLC Example 4.1 , Data transfer

Data transfer on PLC from input memory into data module with function DPRD_DAT. Access to diagnose address via “PLC-Tags”. Conversion of data of the read codes into a string of variable lenght.



Details view

Name	Details
ACK_CJB (STAT Modul)	%I1.3
ACK_Switch to run (STA...	%I1.4
AI2_1[AI]	267
Change Job (CTRL Modul)	%Q1.3
CTRL_(3_bytes)[DO]	277
CTRL_(3_bytes)_1	276
DATA_(32_bytes)[AI]	281
DATA_(32_bytes)_1	280
DI8_DQ6_1[DI/DO]	266
Edge	%M1.1
HSC_1	258
HSC_2	259
HSC_3	260

Network 5: Move Data from Input area in Data Block with function DPRD_DAT

Comment

```
W#16#0119
"DATA_(32_bytes)[AI]" — LADDR      RET_VAL — #RET_Value
                                   RECORD — "Data_block_1"."Data Array"
```

▼ "Data_block_1"."Data Array"	
#RET_Value	
"DATA_(32_bytes)[AI]"	W#16#0119

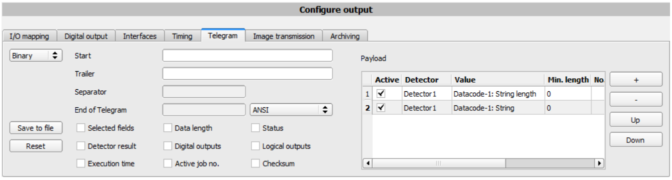
Network 6: Genarate String out of Data Byte

Comment

```
"Data_block_1"."Data Array" — Chars
6 — pChars                           Strg — "Data_block_1"."Example String"
"Data_block_1"."Data Array"[5] — Cnt
```

Fig. 361: Data transfer

5.6.4.4.2 PLC Example 4.2, VISOR® telegram settings



Binary

Start:

Trailer:

Separator:

End of Telegram: ANSI

□ Selected fields □ Data length □ Status

□ Detector result □ Digital outputs □ Logical outputs

□ Execution time □ Active job no. □ Checksum

Payload

	Active	Detector	Value	Min. length	No.
1	☑	Detector 1	Datacode-1: String length	0	
2	☑	Detector 1	Datacode-1: String	0	

Fig. 362: Settings in example telegram in VISOR®

5.6.5 PROFINET- telegram description VISOR®

5.6.5.1 Module1: "Control" (From PLC to VISOR®)

Name in PLC "CTRL (3 bytes)"

Byte-Position in Module	Size in Bytes	Member name	Data-Type	Bit number	Meaning
0	3	Reset error	1 Bit	0	Reset Error clears 4bit Errorcode in Module: "Status". Rising edge (low ==> high) clears error code.
		HW-Trigger Disable	1 Bit	1	This bit is set to disable triggering via the hardware trigger. Valid for triggered and free-run mode. Low (0): Hardware trigger or free run enabled. High (1): Hardware trigger or free run disabled If the HW-Pin "Trigger enable" is used, both (Digital input "Hardware- Trigger" and "HW Trigger Disable Bit") have to be set on "Enable" to accept triggers.
		Trigger	1 Bit	2	Rising edge (low ==> high) Trigger is executed immediately. If Trigger was not executed, Trigger Ack-Bit stays low and Bitfield "Error" has error code "1: Failure trigger request". S. also Timing diagram, chapter: Case: Trigger not possible (not ready) (Page 405).
		Change job	1 Bit	3	Rising edge (low ==> high) indicates, to switch to the job with the number in byte "Jobnumber" in Control Module. This request can be executed delayed. After successful Jobchange, the byte "Jobnumber" in Status Module equals to that in Control Module. If Jobchange could not be executed due to error (e.g. wrong Jobnumber), Bitfield "Error" has error code "2: Failure change job" (and Ready stays low!). S. also Timing diagram, chapter: Case: Jobchange not possible (e.g. wrong job number) (Page 407).
		Switch to run	1 Bit	4	Rising edge (low ==> high) "Switch to Run" is executed. Success or failure of Switch to Run request is signalized with bitfield "Error" (error

					code “3: Failure Switch to run request”) and Bit “Operation Mode”. S. also Timing diagram, chapter: Case: Switch to run not possible (Page 407).
		Reserve	1 Bit	5	
		Reserve	1 Bit	6	
		Reserve	1 Bit	7	
1		Reserve	1 Byte		
2		Job number	U8		Number of job to be changed to, on rising edge of Change-job bit. Binary value 1-255 for “Jobnumber Change” 0 indicates no change, even if Change job bit toggles

Example 1.1: Module 1 ”Control”: Trigger bit set

Must change from 0 to 1, and remain till Trigger ack. is received

Byte 0								Byte 1								Byte 2							
Bit 2:Trigger bit = 1 (rest not relevant in this case)								Reserve								Job number							
0.-7	0.-6	0.-5	0.-4	0.-3	0.-2	0.-1	0.-0	1.-7	1.-6	1.-5	1.-4	1.-3	1.-2	1.-1	1.-0	2.-7	2.-6	2.-5	2.-4	2.-3	2.-2	2.-1	2.-0
x	x	x	x	x	1	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Example 1.2: Module 1 “Control”: Change Job

Must change from 0 to 1, and remain till Change job ack. is received

Byte 0								Byte 1								Byte 2							
Bit 3: Change job = 1, (rest not relevant in this case)								Reserve								Job number: Binary value e.g. = 10101010 (=170dez)							
0.-	0.-	0.-	0.-	0.-	0.-	0.-	0.-	1.-	1.-	1.-	1.-	1.-	1.-	1.-	1.-	2.-	2.-	2.-	2.-	2.-	2.-	2.-	2.-

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
x	x	x	x	1	x	x	x	x	x	x	x	x	x	x	x	1	0	1	0	1	0	1	0

5.6.5.2 Module 2: “Status” (From VISOR® to PLC)

Name in PLC “STAT (6 bytes)”

Byte-Position in Module	Size in Bytes	Member name	Data-Type	Bit number	Meaning
0	3	Ready	1 Bit	0	VISOR® is ready to receive trigger. Ready = 1. Caution: The Ready Bit is reserved to indicate that the VISOR® is ready for the next evaluation cycle. It is not suitable to indicate that a evaluation cycle is finished or the results are valid! (Rising edge of Ready is not equivalent with result valid! The Ready Bit is a replication of the digital Ready- signal and it follows this as fast as possible, but due to the cycle nature of the PROFINET protocol this is not possible hundred per cent.)
		Reserve	1 Bit	1	
		Trigger acknowledge	1 Bit	2	Acknowledge for successful trigger request (via Trigger Bit in Control Module). Acknowledge is cleared as a response of clearing the Trigger bit. If trigger was not executed, Trigger Ack-Bit stays low.
		Change job acknowledge	1 Bit	3	Acknowledge for completion of Change job request (via Change Job Bit in Control Module) - independent of success. Acknowledge is cleared as soon as Change job Request bit is cleared. Success or failure of Change job request is signalized with bitfield “Error” (error code “2: Failure change job”) and byte “Jobnumber” in Status Module . This Ack-Bit can be delayed due to delayed execution of Job Change.

		Switch to run acknowledge	1 Bit	4	Acknowledge for completion of Switch-to-run request (via Switch to run request Bit in Control Module). Acknowledge is cleared as soon as request bit is cleared. Success or failure of Switch to run request is signalized with bitfield “Error” (error code “3: Failure Switch to run request”) and Bit “Operation Mode”. Acknowledge is given after SensoConfig has been disconnected and job has been reloaded from flash, or a failure is detected.
		Reserve	1 Bit	5	
		Reserve	1 Bit	6	
		Reserve	1 Bit	7	
1		Reserve	1 Byte		
2		Digital results (same as in Ethernet Payload, without length)	1 Bit	0	12 RDBU
			1 Bit	1	09 RD
			1 Bit	2	05 PK
			1 Bit	3	06 YE
			1 Bit	4	07 BK
			1 Bit	5	08 GY
					This byte is filled with results of hardware digital output pins. Bit positions are fix (see column “Significance”, same as Ethernet Payload without length information). Value of digital output bit is defined in “Output” - tab “Digital output”, column “Logical Expression” in SensoConfig. If not selected as result output pin or not having a valid logical expression, value of output bit is 0.
		Reserve	1 Bit	6	
		Reserve	1 Bit	7	
3		Job number	U8		Number of current job: Jobnumber: 1-255
4		Image ID	U8		Image ID (0-255) is incremented with each

					job execution, independent from trigger source.
5		Error	4 Bit	0	4 bit error code. Used to indicate failures on requests or system error via Control Module. Error is cleared by “Reset error”, or over-written with next error. 0: No error 1: Failure trigger request (sensor not ready) 2: Failure change job 3: Failure switch to run 5: Failure PROFINET not active in job 15: System error
		Trigegr mode	1 Bit	4	1 = Free run 0 = Triggered
		Reserve	1 Bit	5	
		Operation mode	1 Bit	6	1 = Run 0 = Config
		Reserve	1 Bit	7	

Example 2.1: Module 2 “Status”: Trigger acknowledge is set

- Trigger ack. is set to 1 (Trigger received)
- Ready is set to 0 (Busy)

Byte 0								Byte 1								Byte 2							
Bit 0: Ready = 0 Bit 2: Trigger ack. = 1								Reserve								Digital results							
0.-7	0.-6	0.-5	0.-4	0.-3	0.-2	0.-1	0.-0	1.-7	1.-6	1.-5	1.-4	1.-3	1.-2	1.-1	1.-0	2.-7	2.-6	2.-5	2.-4	2.-3	2.-2	2.-1	2.-0
0	x	1	x	x	x	x	x	x	x	x	X	x	x	x	x	x	x	x	x	x	x	x	x

Byte 3								Byte 4								Byte 5							
Job number								Image ID								Error 4 bit, Trigger mode etc.							
3.-7	3.-6	3.-5	3.-4	3.-3	3.-2	3.-1	3.-0	4.-7	4.-6	4.-5	4.-4	4.-3	4.-2	4.-1	4.-0	5.-7	5.-6	5.-5	5.-4	5.-3	5.-2	5.-1	5.-0

x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Example 2.2: Module 2 “Status”: Change job acknowledge is set

- Change job ack. is set to 1 (Change job received)
- Ready is set to 0 (Busy)

Byte 0								Byte 1								Byte 2							
Bit 0: Ready = 0 Bit 2: Trigger ack. = 1								Reserve								Digital results							
0.-7	0.-6	0.-5	0.-4	0.-3	0.-2	0.-1	0.-0	1.-7	1.-6	1.-5	1.-4	1.-3	1.-2	1.-1	1.-0	2.-7	2.-6	2.-5	2.-4	2.-3	2.-2	2.-1	2.-0
0	x	x	1	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Byte 3								Byte 4								Byte 5							
Job number								Image ID								Error 4 bit, Trigger mode etc.							
3.-7	3.-6	3.-5	3.-4	3.-3	3.-2	3.-1	3.-0	4.-7	4.-6	4.-5	4.-4	4.-3	4.-2	4.-1	4.-0	5.-7	5.-6	5.-5	5.-4	5.-3	5.-2	5.-1	5.-0
1	0	1	0	1	0	1	0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

5.6.5.3 Module 3: “Data” (From VISOR® to PLC)

Name in PLC DATA (2 + 8/16/… bytes)

Byte-Position in Module	Size in Bytes	Member name	Data-Type	Bit number	Meaning
0	1	Image ID	U8		Image ID (0-255) is incremented with each job execution, independent from trigger source.
1	1	Result data overrun	1 Bit	0	Result data has been truncated. 1: Data overrun = truncated 0: No overrun

		Reserve	7 Bit	1-7	Reserve
2	One block of 8, 16, 32, 64, 128 or 256 Bytes	Result data	Byte-array		Data as defined in SensoConfig in “Output/Telegram/Payload”. In case of PROFINET in tab “Telegram” = “Binary” must be selected.

Example 3.1: Module 3 “Data”

- E.g.: No overrun
- Data Byte 2 … n as defined in SensoConfig “Output/Telegram/Payload”

Byte 0								Byte 1								Byte 2 ... n							
Image ID								Result data overrun Reserve								Result data: as defined in SensoConfig “Output/Telegram/Payload” in binary format.							
0.7	0.6	0.5	0.4	0.3	0.2	0.1	0.0	1.7	1.6	1.5	1.4	1.3	1.2	1.1	1.0	2.7	2.6	2.5	2.4	2.3	2.2	2.1	2.0
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

5.6.5.4 Module 4: “Request” (From PLC to VISOR®)

Name in PLC “REQU (4 + 8/16/… bytes)”

Byte-Position in Module	Size in Bytes	Member name	Data-Type	Bit number	Meaning
0	1	Key	1 Byte		Request key (Request counter)
1	1	Reserve	1 Byte		Reserve
2	1	Reserve	1 Byte		Reserve
3	1	Reserve	1 Byte		Reserve
4	One block of 8,	Request	Byte-		Same data as for TCP requests, s.

	16, 32, 64, 128 or 256 Bytes	data	array		addendum: .... Serial communication BINARY (Page 518)

5.6.5.5 Module 5: “Response” (From VISOR® to PLC)

Name in PLC “RESP (4 + 8/16/… bytes)”

Byte-Position in Module	Size in Bytes	Member name	Data-Type	Bit number	Meaning
0	1	Key	U8		Response key which is mirrored from request
1	1	Result data overrun	1 Bit	0	Response data has been truncated
		Reserve	7 Bit	1-7	Reserve
2	1	Reserve	1 Byte		Reserve
3	1	Reserve	1 Byte		Reserve
4	One block of 8, 16, 32, 64, 128 or 256 Bytes	Result data	Byte-array		Same data as for TCP responses s. addendum: … Serial communication BINARY (Page 518)

5.6.5.6 Start- / End- criteria per each PROFINET command

Command (Modul “Control”)	**Start- condition (Modul “Status”)**	**Confirmation of acceptance (Modul “Status”)**	**Confirmation of execution (Modul “Status”)**
Trigger	Ready = High	Trigger Ack = High	Image ID changed
Change Job	/	Job Change Ack = High	Job Nr. changed
Switch to run	Operation Mode = Low	Switch to run Ack = High	Operation Mode = High

5.6.6 Timing diagrams to the VISOR® PROFINET communication with a PLC

5.6.6.1 Case: Trigger ok

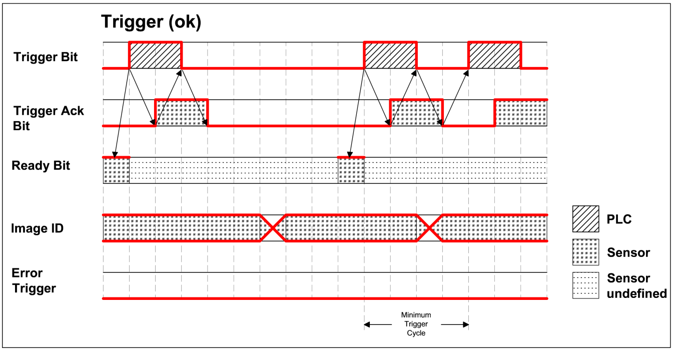


Fig. 363: Timing Trigger ok

5.6.6.2 Case: Trigger not possible (not ready)

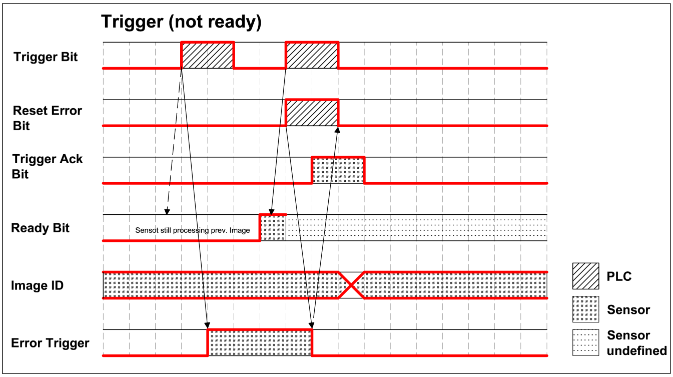


Fig. 364: Timing Trigger not ready

5.6.6.3 Case: Jobchange ok

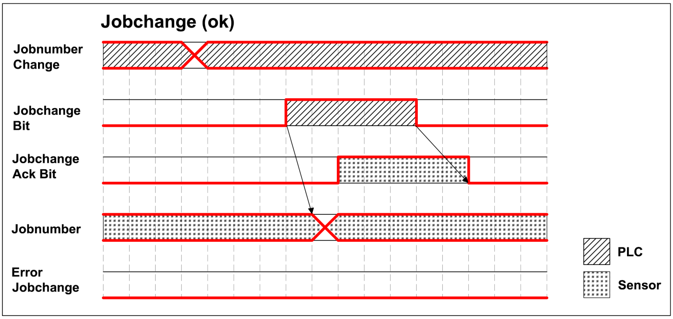


Fig. 365: Timing Jobchange ok

5.6.6.4 Case: Jobchange delayed

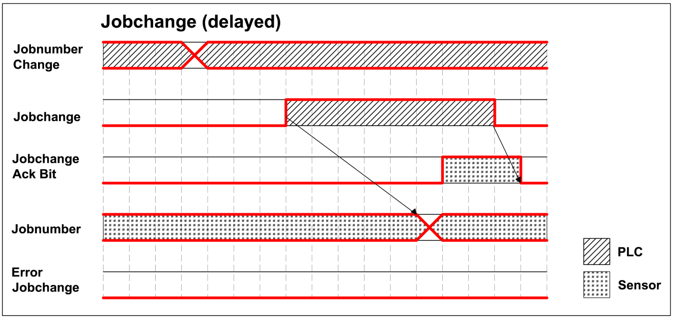


Fig. 366: Timing Jobchange delayed

5.6.6.5 Case: Jobchange not possible (e.g. wrong job number)

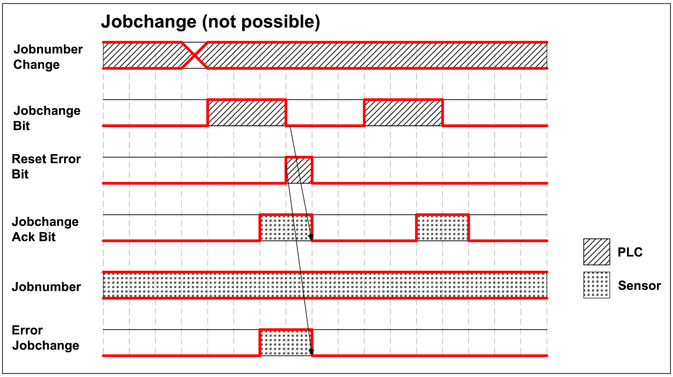


Fig. 367: Timing Jobchange not possible

5.6.6.6 Case: Switch to run ok

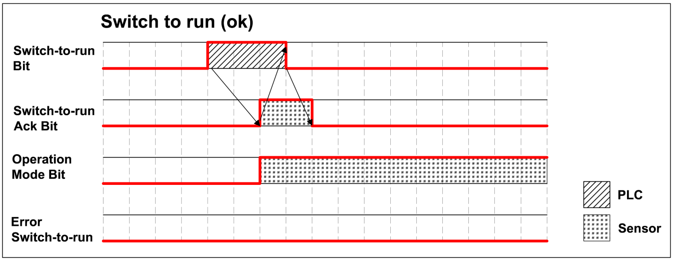


Fig. 368: Timing Switch to run ok

5.6.6.7 Case: Switch to run not possible

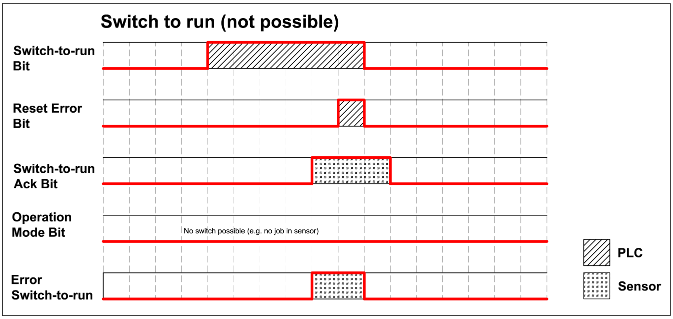


Fig. 369: Switch to run not possible

5.6.6.8 Strong recommendations for PLC programmer

1. Follow the sequence for requests.
2. Wait for completion of an action before sending the next one. Completion of action is given by change in image ID for trigger request and reception of acknowledge bit for other requests.
   Note that completion of action cannot be safely detected by low-high transition of READY because long exchange rates between PLC and VISOR®, e.g. 32ms, may result in READY not getting low.
3. READY should be high before sending trigger request.

5.6.6.9 Request sequences

5.6.6.9.1 Accepting / Discarding of Requests of Control Module

1. Request is accepted with rising Ack bit.
2. Request is discarded with error bit.
3. Request is discarded without error and Ack, if sensor is processing previous request and has not given Ack to that request. (Not obeying recommended “Handshake”).

5.6.6.9.2 Trigger Request Sequence

1. Check Ready Bit high in Status module.
2. Set Trigger Request Bit high in Control Module.
3. Check Trigger Ack Bit high and Error Bitfield in status Module.
   a. if Trigger Ack Bit high (Trigger successful), set Trigger Request Bit low (continue with step 4).
   b. if Trigger Ack Bit low and Error Bitfield has Errorcode “1: Failure trigger request”, then set Trigger Request Bit low and set Reset Error Bit high (continue with step 6).
4. (Case Trigger successful) check Trigger Ack Bit low.
5. (Case Trigger successful) then check ImageID Byte incremented (value 255 change to 0).
6. (Case Trigger not successful) Check Error Bitfield going 0, then set Reset Error Bit low.

5.6.6.9.3 Change Job Request Sequence

1. Set Byte Jobnumber in Control module to desired value.
2. Check Ready Bit in Status module (in case of previous jobchange failure, ignore Ready).

3. Set ChangeJob Request Bit high in Control Module.
4. Wait and Check for ChangeJob Ack Bit high.
5. Check Error Bitfield in status Module.
   a. if Error Bitfield has not Errorcode “2: Failure change job”, then set ChangeJob Request Bit low (continue with step 6).
   b. if Error Bitfield has Errorcode “2: Failure change job”, then set ChangeJob Request Bit low and set Reset Error Bit high. (continue with step 8)
6. (Case ChangeJob successful) Check ChangeJob Ack Bit low.
7. (Case ChangeJob successful) then check Jobnumber Byte in Status module. If jobnumber is correct.
8. (Case ChangeJob not successful) Check Error Bitfield going 0, then set Reset Error Bit low.
   Check the correct jobnumber and repeat the request with Step 3 (Ready bit stays low).

5.6.6.9.4 Switch-To-Run Request Sequence

1. Check Ready Bit high and Operation Mode Bit low (Config mode) in Status module.
2. Set Switch-to-Run Request Bit high in Control Module.
3. Wait and Check Switch-to-Run Ack Bit high.
4. Check Error Bitfield in Status Module.
   a. if Error Bitfield has not Errorcode “3: Failure switch to run request”, then set Switch-to-Run Request Bit low (continue with step 5).
   b. if Error Bitfield has Errorcode “3: Failure switch to run request”, then set Switch-to-Run Request Bit low and set Reset Error Bit high (continue with step 6).
5. (Case Switch-to-Run successful) Check Switch-to-Run Ack Bit low and Operation Mode Bit high (Run mode).
6. (Case Switch-to-Run not successful) Check Switch-to-Run Ack Bit low and Error Bitfield going 0, then set Reset Error Bit low.

5.6.6.9.5 Sequence for requests via request/response module:

1. Request ID and request data is set.
2. Request key is incremented.
3. PLC waits for until request key is mirrored in response key.
4. PLC reads results including error included in results. See TCP payload.

5.6.6.9.6 Error Reset (depicted in UseCase "Jobchange not possible")

1. Reset by "Reset Error Bit"
2. Error bits are overwritten by new error bits.

5.7 VISOR® vision sensor, EtherNet/IP, Introduction

This chapter explains the use of the Vision Sensor with EtherNet/IP.

For data communication betweenVISOR® vision sensor and PLC via EtherNet/IP the following topics are explained: electrical connection, settings in VISOR® vision sensor and PLC (as example for Rockwell RSLogix), available telegrams formats and the telegram timing.

5.7.1 Electrical connection of the VISOR® vision sensor in the EtherNet/IP network

The VISOR® vision sensor is connected via an Ethernet TCP/IP and a EtherNet/IP switch to the network.

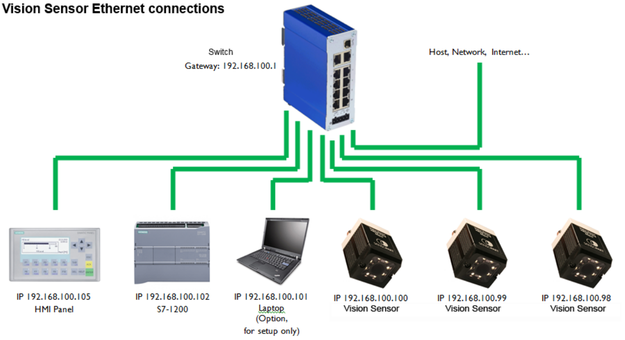
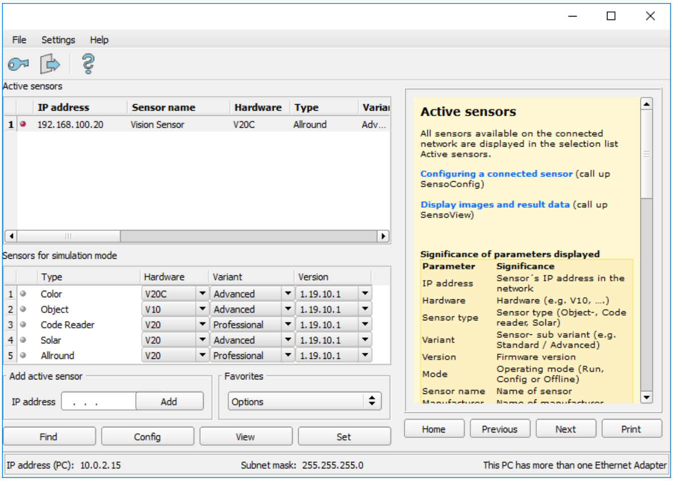


Fig. 370: Connection of Vision Sensor via EtherNet/IP switch

5.7.2 Configuration of VISOR® vision sensor for the use with EtherNet/IP

In this example the configuration of the VISOR® vision sensor is described.

5.7.2.1 Settings in SensoFind



Active sensors

		IP address	Sensor name	Hardware	Type	Variai
1	●	192.168.100.20	Vision Sensor	V20C	Allround	Adv...

Sensors for simulation mode

		Type	Hardware	Variant	Version
1	●	Color	V20C	Advanced	1.19.10.1
2	●	Object	V10	Advanced	1.19.10.1
3	●	Code Reader	V20	Professional	1.19.10.1
4	●	Solar	V20	Advanced	1.19.10.1
5	●	Allround	V20	Professional	1.19.10.1

Add active sensor

IP address . . .

Favorites

IP address (PC): 10.0.2.15 Subnet mask: 255.255.255.0 This PC has more than one Ethernet Adapter

Active sensors

All sensors available on the connected network are displayed in the selection list Active sensors.

Configuring a connected sensor (call up SensoConfig)

Display images and result data (call up SensoView)

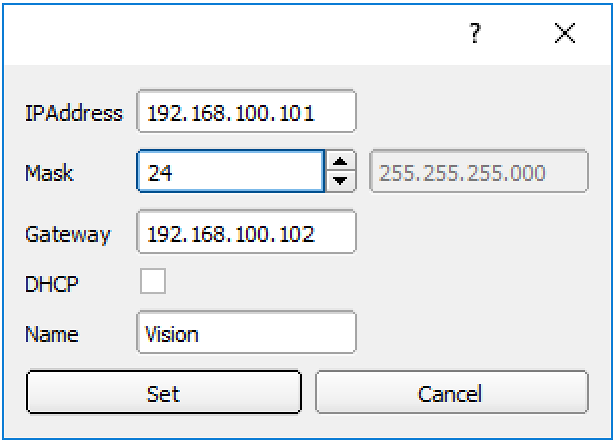
Significance of parameters displayed

Parameter	Significance
IP address	Sensor´s IP address in the network
Hardware	Hardware (e.g. V10, ...)
Sensor type	Sensor type (Object-, Code reader, Solar)
Variant	Sensor- sub variant (e.g. Standard / Advanced)
Version	Firmware version
Mode	Operating mode (Run, Config or Offline)
Sensor name	Name of sensor
Manufacturer	Name of manufacturer

Fig. 371: VISOR® vision sensor is displayed and selected in SensoFind.

When SensoFind launches or by clicking the “Find” button, all active sensors are listed in the upper window called “Active sensors”. You can change the IP address, subnet mask and other parameters on the VISOR® vision sensor by clicking the “Set” button. This displays the following dialog box.

5.7.2.2 Setting of IP and name



IPAddress 192.168.100.101

Mask 24 255.255.255.000

Gateway 192.168.100.102

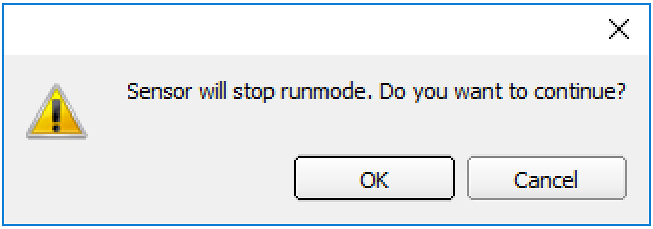
DHCP □

Name Vision

Fig. 372: Setting of IP and name

5.7.2.3 Open SensoConfig

With click to “Config” in SensoFind, and to “OK” in the following dialog SensoConfig starts. With the desired VISOR® vision sensor is selected in SensoFind, click “Config.” When the following dialog box is displayed, click “OK” to stop the VISOR® vision sensor and begin configuring it.

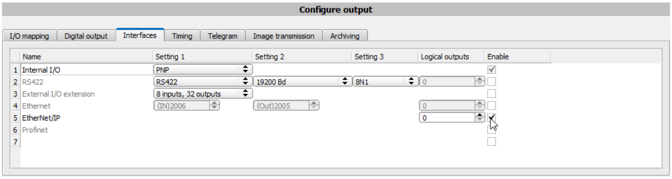


Sensor will stop runmode. Do you want to continue?

Fig. 373: Open SensoConfig

5.7.2.4 Select Interface “EtherNet/IP”

In the setup menue click “Output”. On the ”Interface” tab, check the box to select EtherNet/IP.



Configure output

	Name	Setting 1	Setting 2	Setting 3	Logical outputs	Enable
1	Internal I/O	PNP				☑
2	RS422	RS422	19200 Bd	8N1	0	□
3	External I/O extension	8 inputs, 32 outputs				□
4	Ethernet	(IN)2006	(Out)2005		0	□
5	EtherNet/IP				0	☑
6	Profinet					□
7						□

Fig. 374: Activation of EtherNet/IP in SensoConfig

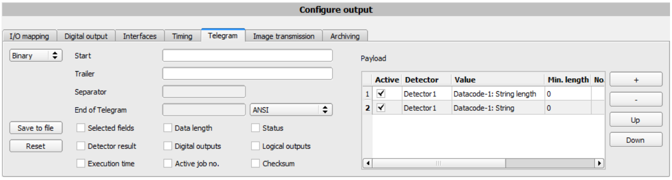
5.7.2.5 Definition of the telegram

In the tab “Telegram” the data which should be transferred can be defined completely free. For the use with EtherNet/IP this must be done with format “Binary”.

5.7.2.5.1 Definition of the output data

The output data itself are configured identically as the data output via Ethernet TCP/IP or RS422 in: SensoConfig/Output/Telegram.

The description you find in the Vision Sensor User manual in chapter Telegram, Data output (Page 296) under: SensoConfig/Help/Manual.



Binary

Start:

Trailer:

Separator:

End of Telegram: ANSI

☐ Selected fields ☐ Data length ☐ Status

☐ Detector result ☐ Digital outputs ☐ Logical outputs

☐ Execution time ☐ Active job no. ☐ Checksum

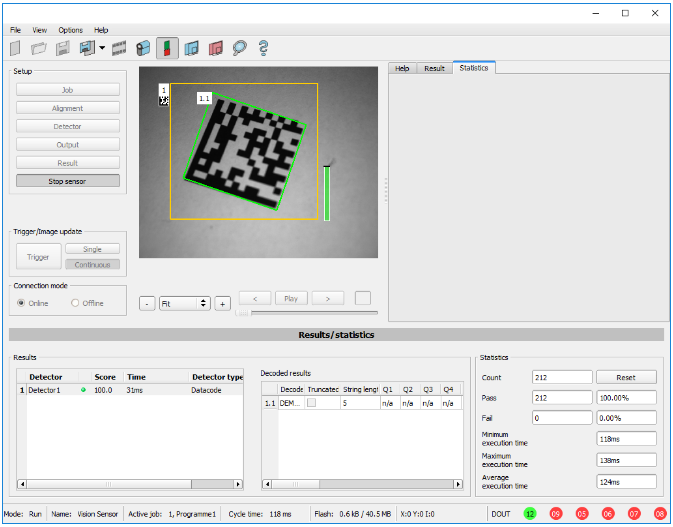
Payload

	Active	Detector	Value	Min. length	No.
1	☑	Detector1	Datacode-1: String length	0	
2	☑	Detector1	Datacode-1: String	0	

Fig. 375: Data output, protocol: Binary

5.7.2.6 Start sensor, data output

With “Start sensor” the configuration data are transferred to the VISOR® vision sensor. The sensor get´s started and now the output data are transferred as defined.



Connection mode: ● Online ○ Offline

Results

	Detector		Score	Time	Detector type
1	Detector1	●	100.0	31ms	Datacode

Decoded results

	Decode	Truncated	String lengt	Q1	Q2	Q3	Q4
1.1	DEM...	☐	5	n/a	n/a	n/a	n/a

Statistics

Count	212	
Pass	212	100.00%
Fail	0	0.00%
Minimum execution time		118ms
Maximum execution time		138ms
Average execution time		124ms

Mode: Run | Name: Vision Sensor | Active job: 1, Programme1 | Cycle time: 118 ms | Flash: 0.6 kB / 40.5 MB | X:0 Y:0 I:0

Fig. 376: Start sensor

5.7.3 EtherNet/IP protocol

EtherNet/IP has a predefined protocol, consisting oft two assemblies.

- Assembly request (PLC to Sensor, 344 bytes long) and
- Assembly response (Sensor to PLC, 444 bytes long)

5.7.3.1 Assembly request

Request key

Position 0 (Byte 0 and 1) are the request key. Every change in the request key indicates to the sensor that there are new data inside the assembly request available. Changing the request key triggers a command like trigger, job change…

Command configuration

The command configuration starts on position 2 and has a size of 2 bytes (byte 2 and 3).
It is possible with selected code, to choose between: Trigger, Change job and Set reference string.

- **Trigger:** To make a trigger (to take a new picture), the code is: 0x01
- **Change job:** to send the command to change a job, the code is: 0x02

Commands which need further arguments like "change job" need to get the arguments on the correct byte positions : the job number is an integer value to be placed on "pnValueInt" (byte 264), the length of this information is 1 byte long, following Parameter "unNumint" (byte 6) has to be set to "1".

Examples

Trigger

Request structure	Key	ID
Storage	unKey	unId
Byte position	0	2
Request pattern	Count	0x01

Change Job

Request structure	Key	ID	NumInt	Job number
Storage	unKey	unId	NumInt	ValueInt[0]
Byte position	0	2	6	264
Request pattern	Count	0x02	0x01	Job no.

Set reference string permanent

Request structure	Key	ID	Length of string	NumInt	Ref. String	Detector number	Parameter number	Parameter type
Storage	unKey	unId	NumChar	NumInt	Char	Int[0]	Int[1]	Int[2]

Byte position	0	2	4	6	8	264	268	272
Request pattern	Count	0x05	0x01	0x03	0x43	0x01	0x65	0x0A
Explanation			Example 1 character	Constant value	Example string f. “C”	Example for detector 1	Command set ref. string	Example param. type string

Set reference string temporary

Request structure	**Key**	**ID**	**Length of string**	**NumInt**	**Ref. String**	**Detector number**	**Parameter number**	**Parameter type**
Storage	unKey	unId	NumChar	NumInt	Char	Int[0]	Int[1]	Int[2]
Byte position	0	2	4	6	8	264	268	272
Request pattern	Count	0x06	0x02	0x03	0x41 0x42	0x01	0x65	0x0A
Explanation			Example 2 character	Constant value	Example string f. “AB”	Example for detector 1	Command set ref. string	Example param. type string

5.7.3.1.1 Sensor Ready information / signaling and handshake

Over hardware IO the VISOR® vision sensor offers a “Ready” signal. Sending a Trigger is allowed only if “Ready” signal is high.

When hardware ready signal is not connected to the PLC it is very easy to find out the ready status just over EtherNet/IP.

After first connection of PLC to VISOR® the VISOR® must be in a “ready”-state, otherwise there would have been no connection.

Following chart shows the hardware ready signal in relation to the commands over EtherNet/IP at the example of a typical trigger sequence:

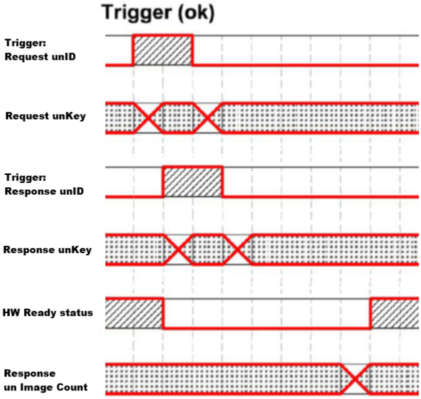
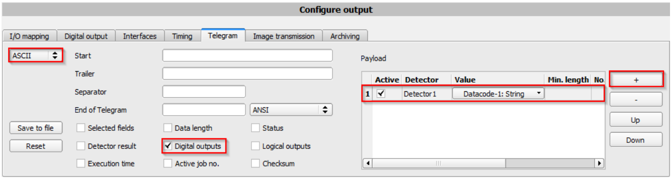


Fig. 377: EtherNet/IP, Sensor ready

5.7.3.2 Assembly response

User defined data output to be configured in the result telegram specification:



ASCII

Start

Trailer

Separator

End of Telegram: ANSI

- ☐ Selected fields
- ☐ Data length
- ☐ Status
- ☐ Detector result
- ☑ Digital outputs
- ☐ Logical outputs
- ☐ Execution time
- ☐ Active job no.
- ☐ Checksum

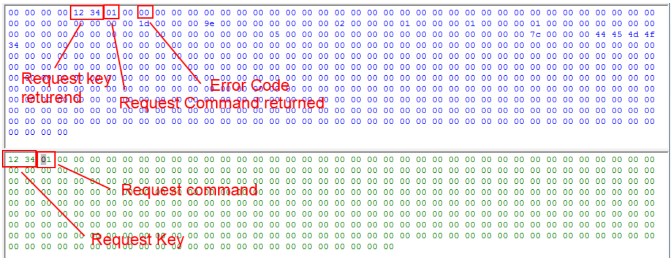
	Active	Detector	Value	Min. length	No.
1	✓	Detector1	Datacode-1: String		

Fig. 378: User defined data output

Depending from kind of output data they can be found in the assembly response at

- Boolean: byte 92 (pucBool)
- String: byte 116 (pcString)
- Integer: byte 244 (pnInt)

Example Trigger Handshake

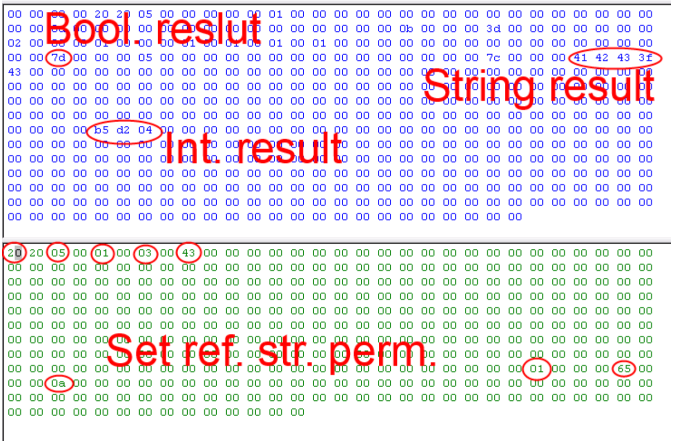


```
00 00 00 00 12 34 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 1d 00 00 00 9e 00 00 00 00 00 00 00 02 00 00 00 01 00 00 00 01 00 00 00 01 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 05 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 7c 00 00 00 44 45 4d 4f
34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00
```

```
12 34 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Fig. 379: EtherNet/IP, Trigger handshake

Response and request bytes



```
00 00 00 00 20 20 05 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0b 00 00 00 3d 00 00 00 00 00 00 00
02 00 00 00 01 00 00 00 01 00 01 00 01 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 7d 00 00 00 05 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 7c 00 00 00 41 42 43 3f
43 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 b5 d2 04 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

```
20 20 05 00 01 00 03 00 43 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 65 00
00 00 0a 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Fig. 380: EtherNet/IP, Response and request bytes

A complete documentation of the assemblies can be found in the end of this chapter.

5.7.4 EDS file

SensoPart provides an EDS file for easy implementation into controllers which support EDS files. Concerning installation and use of EDS files please use the documentation of the controller.

Example: Installation of EDS file in RSLogix:

1.) Use dialog for installation of EDS files:

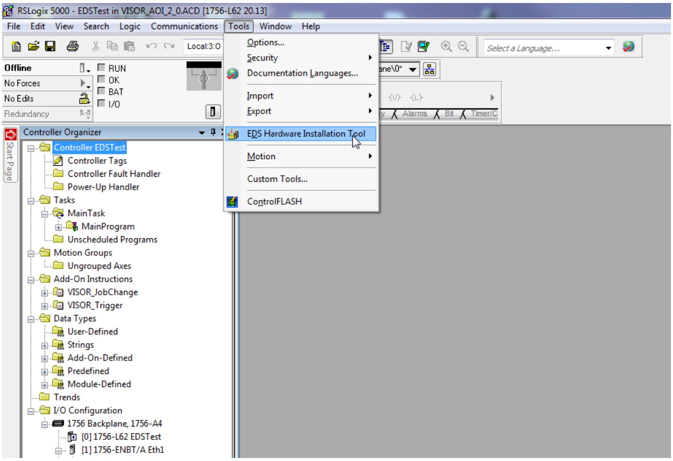
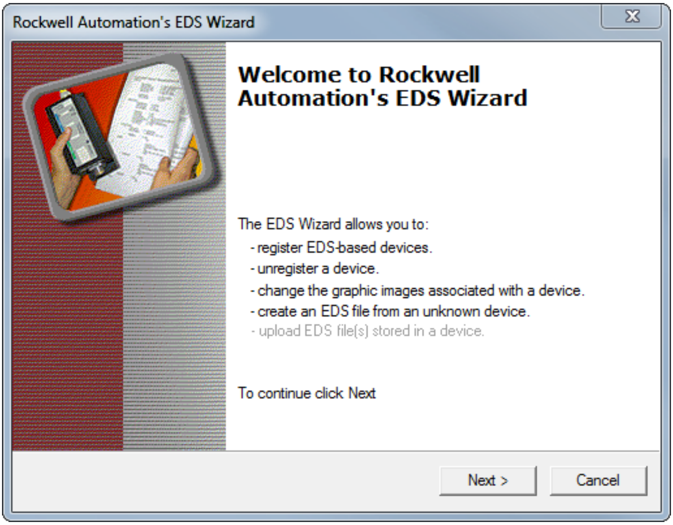


Fig. 381: Installation of EDS files

2.) Follow the instructions of the Wizard:


Rockwell Automation's EDS Wizard

Welcome to Rockwell Automation's EDS Wizard

The EDS Wizard allows you to:

- register EDS-based devices.
- unregister a device.
- change the graphic images associated with a device.
- create an EDS file from an unknown device.
- upload EDS file(s) stored in a device.

To continue click Next

Fig. 382: Wizard, EDS file installation

5.7.5 Implementation of VISOR® vision sensor into RSLogix

Establish a network-connection between RSLogix and each sensor by adding a Generic Ethernet Module in the Ethernet I/O network for each sensor.

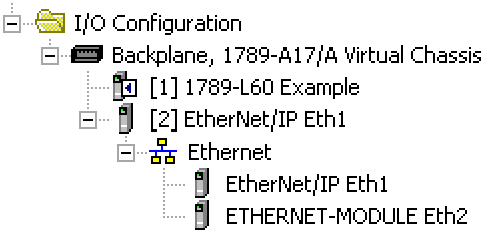


Fig. 383: EtherNet/IP, Ethernet Module

You will also need to set up the suitable network adapter which is mounted in side the PLC.

The Ethernet Card will need to setup as a module on the Ethernet I/O network within the same subnet as the camera(s) you will be communicating with.

In this example the IP adress of RSLogix is 192.168.100.84, this can be configured by click with right mouse button on “EtherNet/IP Eth1” ➔ “New Module”:

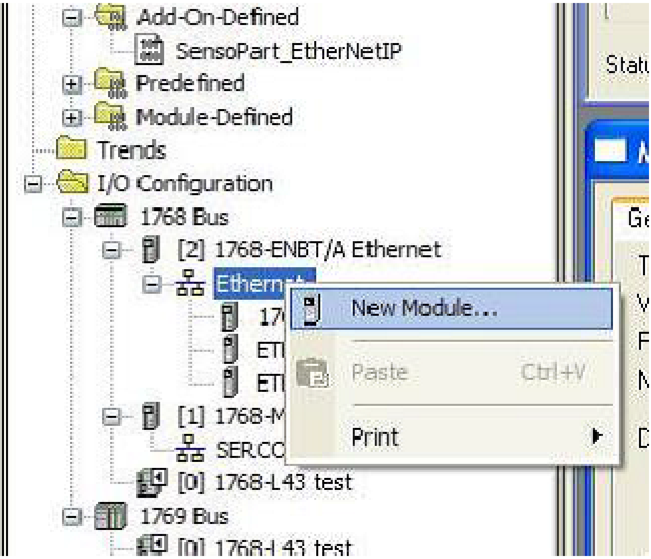
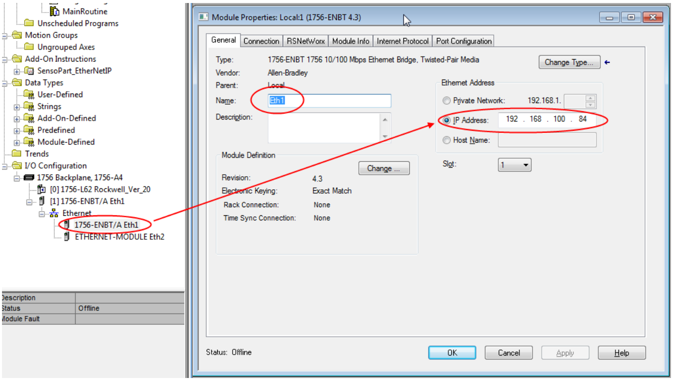


Fig. 384: EtherNet/IP, New EtherNet/IP Module



Description	
Status	Offline
Module Fault	

Module Properties: Local:1 (1756-ENBT 4.3)

Type: 1756-ENBT 1756 10/100 Mbps Ethernet Bridge, Twisted-Pair Media
Vendor: Allen-Bradley
Parent: Local
Name: Eth1
Description:

Module Definition
Revision: 4.3
Electronic Keying: Exact Match
Rack Connection: None
Time Sync Connection: None

Ethernet Address
○ Private Network: 192.168.1.
● IP Address: 192 . 168 . 100 . 84
○ Host Name:

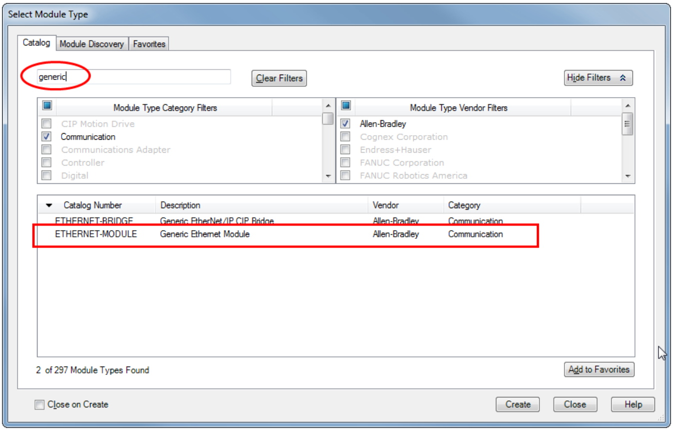
Slot: 1

Status: Offline

Fig. 385: Ethernet/IP, adress

5.7.5.1 Over Generic Profile

Each sensor is added as a “Generic Ethernet Module” as shown in the following two screenshots: enter IP adress of sensor (as set before with SensoFind software) and the number of input and output bytes like shown in screenshot:



Select Module Type

generic

Module Type Category Filters	Module Type Vendor Filters
□ CIP Motion Drive	☑ Allen-Bradley
☑ Communication	□ Cognex Corporation
□ Communications Adapter	□ Endress+Hauser
□ Controller	□ FANUC Corporation
□ Digital	□ FANUC Robotics America

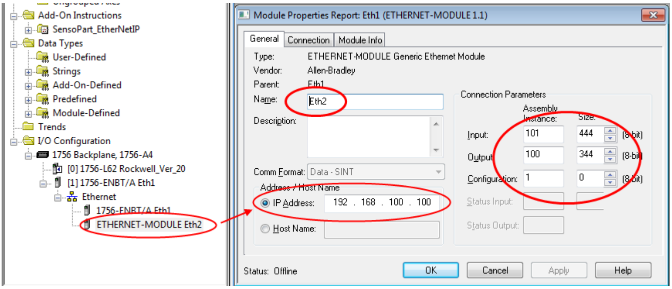
Catalog Number	Description	Vendor	Category
ETHERNET-BRIDGE	Generic EtherNet/IP CIP Bridge	Allen-Bradley	Communication
ETHERNET-MODULE	Generic Ethernet Module	Allen-Bradley	Communication

2 of 297 Module Types Found

□ Close on Create

Fig. 386: EtherNet/IP, select Generic Module

Add one Ethernet module for each sensor



Module Properties Report: Eth1 (ETHERNET-MODULE 1.1)

Type: ETHERNET-MODULE Generic Ethernet Module
Vendor: Allen-Bradley
Parent: Eth1
Name: Eth2
Description:
Comm Format: Data - SINT
Address / Host Name
● IP Address: 192 . 168 . 100 . 100
○ Host Name:

Connection Parameters

	Assembly Instance:	Size:	
Input:	101	444	(8-bit)
Output:	100	344	(8-bit)
Configuration:	1	0	(8-bit)
Status Input:			
Status Output:			

Status: Offline

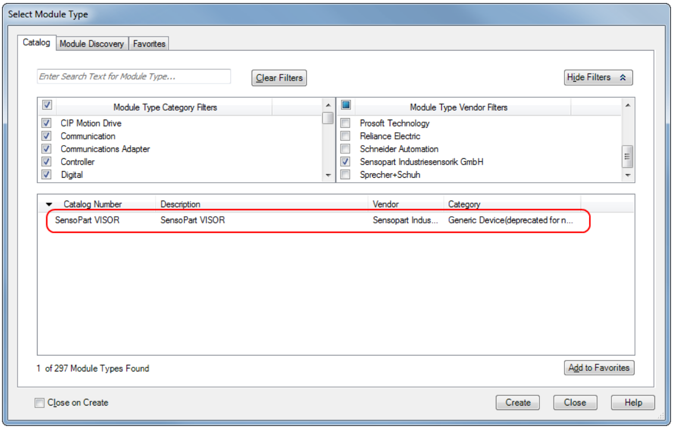
Fig. 387: EtherNet/IP, number of input and output byte

Duplicate this step with different name and IP-address for each sensor, rest of settings the same.

5.7.5.2 Over EDS-File

If an EDS file has been installed before “SensoPart VISOR®” can be selected directly inside the list of available modules.

Assembly size and Assembly instance is set automatically in this case. Only IP address of VISOR® has to be entered.



☑ Module Type Category Filters
☑ CIP Motion Drive
☑ Communication
☑ Communications Adapter
☑ Controller
☑ Digital

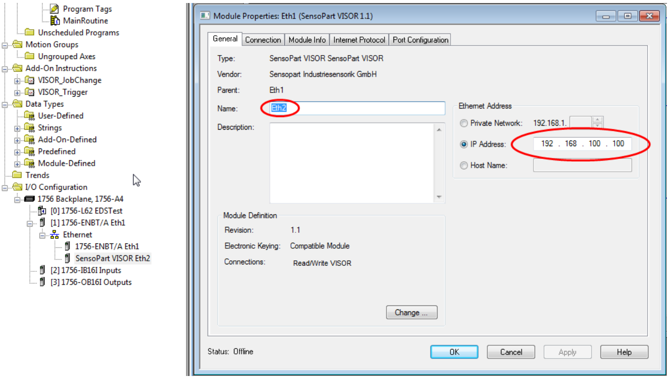
■ Module Type Vendor Filters
□ Prosoft Technology
□ Reliance Electric
□ Schneider Automation
☑ Sensopart Industriesensorik GmbH
□ Sprecher+Schuh

Catalog Number	Description	Vendor	Category
SensoPart VISOR	SensoPart VISOR	Sensopart Indus...	Generic Device(deprecated for n...

1 of 297 Module Types Found

□ Close on Create

Fig. 388: EtherNet/IP, select Generic Module



Type: SensoPart VISOR SensoPart VISOR
Vendor: Sensopart Industriesensorik GmbH
Parent: Eth1
Name: Eth2
Description:

Ethernet Address
○ Private Network: 192.168.1.
● IP Address: 192 . 168 . 100 . 100
○ Host Name:

Module Definition
Revision: 1.1
Electronic Keying: Compatible Module
Connections: Read/Write VISOR

Status: Offline

Fig. 389: EtherNet/IP, set IP address, EDS- file

Setup of RSLogix

Open a new project in RSLogix or take one of your existing projects where you want to implement the SensoPart AOI. Right-click “Add-On Instructions” in “Controller Organizer”:

Choose “Import Add-On Instruction…” and select file.

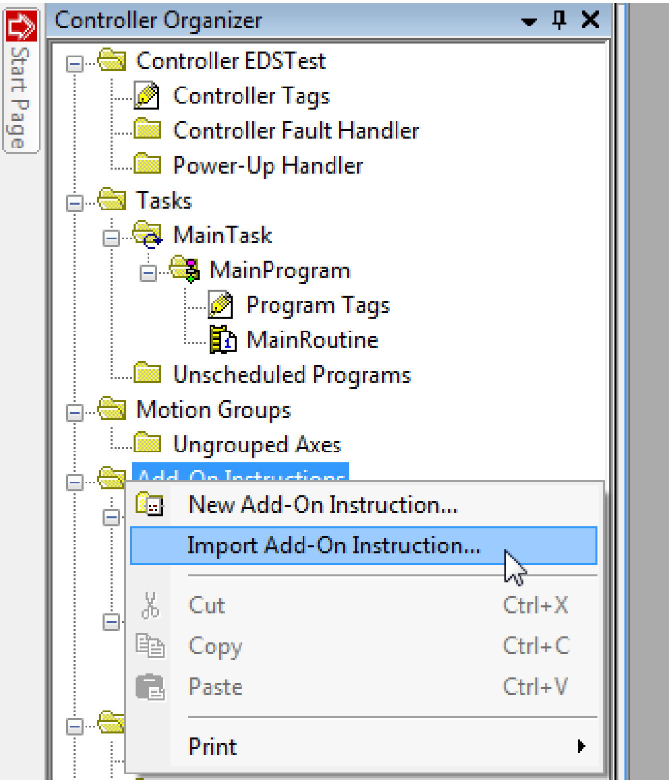
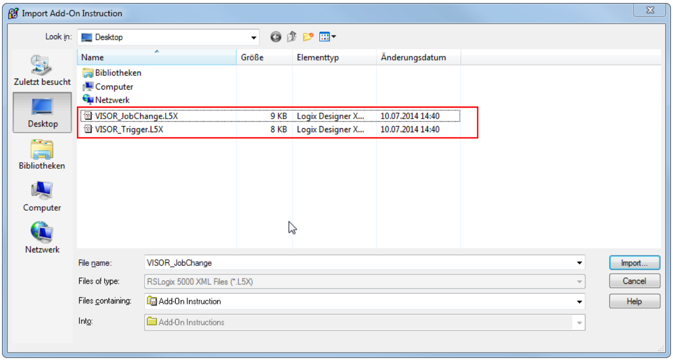


Fig. 390: EtherNet/IP, Import Add-On Instruction

This step has to be done only once, even when using more than one sensor.



Look in: Desktop

Name	Größe	Elementtyp	Änderungsdatum
Bibliotheken			
Computer			
Netzwerk			
VISOR_JobChange.L5X	9 KB	Logix Designer X...	10.07.2014 14:40
VISOR_Trigger.L5X	8 KB	Logix Designer X...	10.07.2014 14:40

File name: VISOR_JobChange

Files of type: RSLogix 5000 XML Files (*.L5X)

Files containing: Add-On Instruction

Into: Add-On Instructions

Fig. 391: EtherNet/IP, select AOI

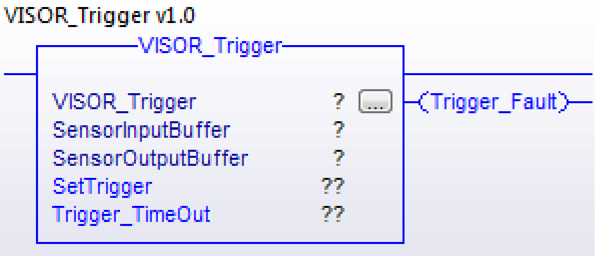
There are existing two AOIs:

- VISOR®_Trigger: send a trigger to the VISOR®
- VISOR®_JobChange: send a command to VISOR® to change the active job

Import one or both AOIs depending from the requirements of your application.

Alternative you can load example project “VISOR®_AOI_2_0.ACD” which is a complete project with all settings as an example.

5.7.5.3 Parameters and functions of the AOIs

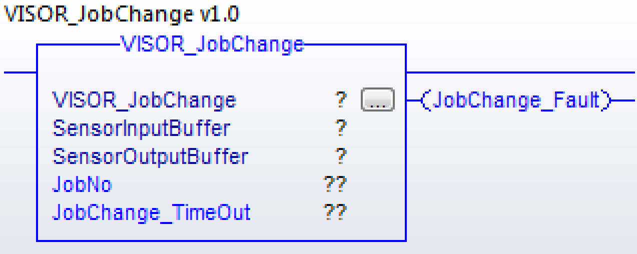


VISOR_Trigger v1.0

VISOR_Trigger

VISOR_Trigger	?
SensorInputBuffer	?
SensorOutputBuffer	?
SetTrigger	??
Trigger_TimeOut	??

Fig. 392: EtherNet/IP, Trigger AOI



VISOR_JobChange v1.0

VISOR_JobChange

VISOR_JobChange	?
SensorInputBuffer	?
SensorOutputBuffer	?
JobNo	??
JobChange_TimeOut	??

Fig. 393: EtherNet/IP, JobChange AOI

Parameters

SENSOR_Input_ Buffer	Array (444 SINT)	to be linked to Generic Ethernet Module or EDS file
SENSOR_Output_ Buffer	Array (344 SINT)	to be linked to Generic Ethernet Module or EDS file

SetTrigger	BOOL	rising edge on this boolean tag will start image capture
JobNo	DINT	number target job for jobchange, change of value will perform jobchange
Trigger_TimeOut	DINT	max. time in ms until confirmation of trigger expected
JobChange_ TimeOut	DINT	max. time in ms until confirmation of jobchange expected
Trigger_Fault	BOOL	confirmation of trigger was not received inside Trigger_TimeOut time, no response for trigger, check possible reasons and clear fault by sending new trigger
JobChange_Fault	BOOL	confirmation of jobchange was not received inside JobChange_TimeOut, e.g. when trying to change to a job which is not existing, clear fault by sending valid job number

5.7.6 How to use the AOI inside your project

A detailed application example see also document: EtherNetIP-OperatingManual.pdf in Start/SensoPart /VISOR® vision sensor/Tools/...

Next step is to add the function block into your program and then configure the input data in the Add On Instruction.

Choose register “Add on” in RSLogix and click on the “VISOR® Trigger” or “VISOR® JobChange” button. Then place the function block into your program and configure.

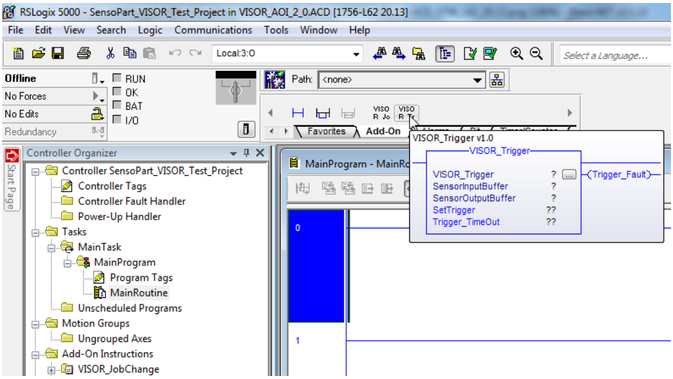


Fig. 394: EtherNet/IP, Trigger AOI

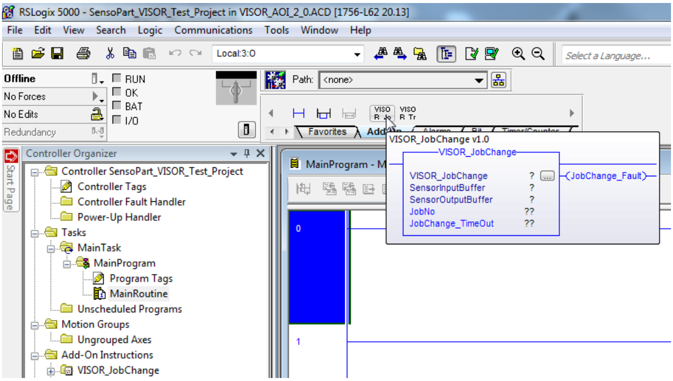


Fig. 395: EtherNet/IP, JobChange AOI

There must be at least one SENSOR_Input_Buffer and one SENSOR_Output_Buffer. You can use instance created with the configuration of the Generic Ethernet Module or EDS file, or create a new tag that is mapped or aliased to it.

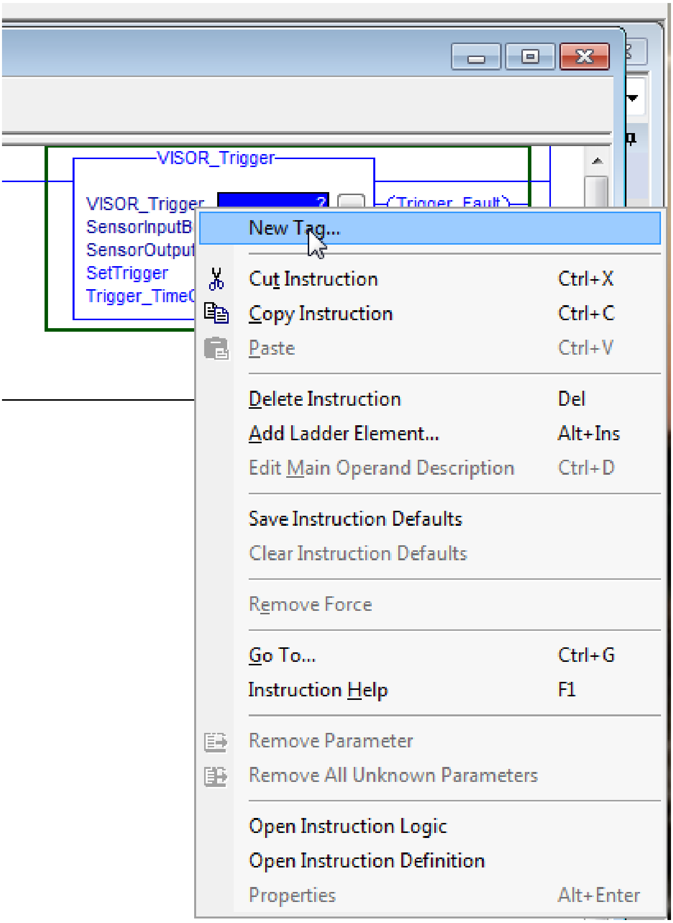
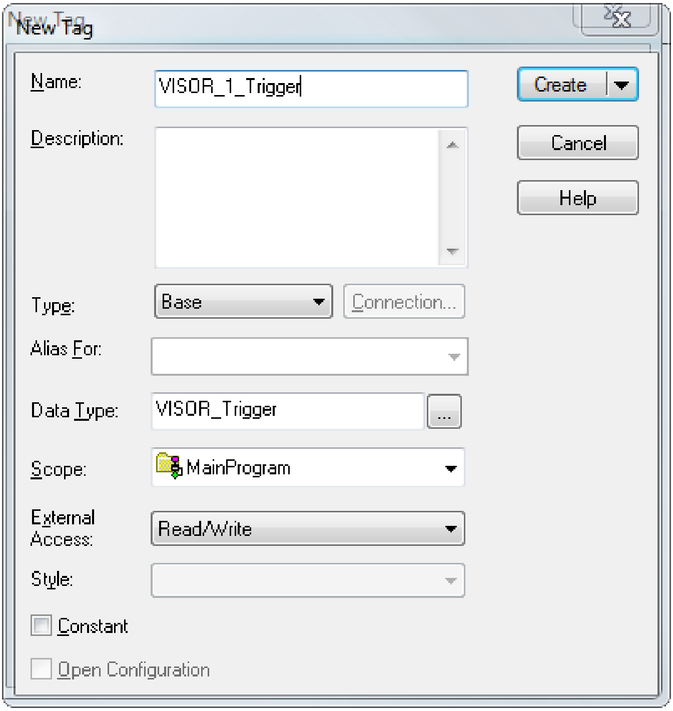


Fig. 396: EtherNet/IP, New Tag



New Tag

Name: VISOR_1_Trigger

Description:

Type: Base

Alias For:

Data Type: VISOR_Trigger

Scope: MainProgram

External Access: Read/Write

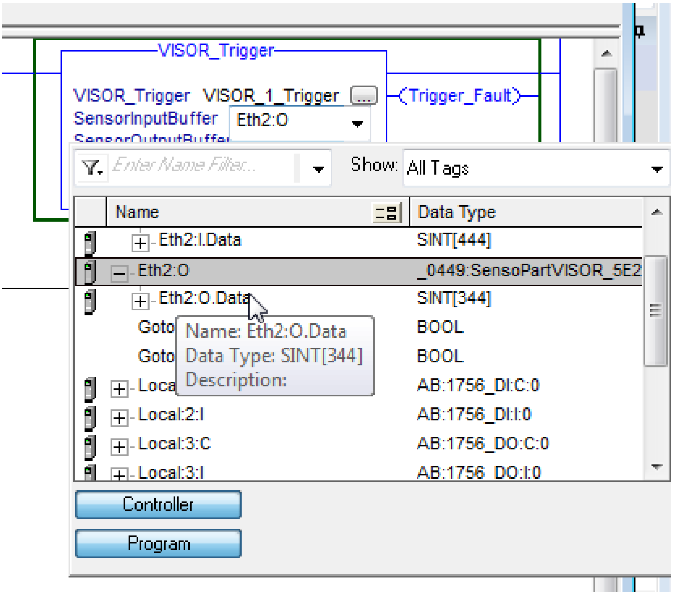
Style:

□ Constant

□ Open Configuration

Fig. 397: EtherNet/IP, New Tag “VISOR®_1_Trigger”, first instance of Trigger AOI

Sensor input and output buffers will already be there by creating the instance of the generic ethernet module or importing the EDS file



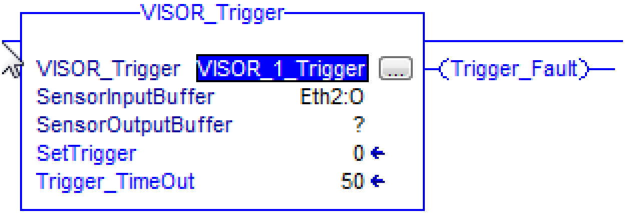
VISOR_Trigger VISOR_1_Trigger
SensorInputBuffer Eth2:O
SensorOutputBuffer

Name	Data Type
Eth2:I.Data	SINT[444]
Eth2:O	_0449:SensoPartVISOR_5E2
Eth2:O.Data	SINT[344]
Goto	BOOL
Goto	BOOL
Loca	AB:1756_DI:C:0
Local:2:I	AB:1756_DI:I:0
Local:3:C	AB:1756_DO:C:0
Local:3:I	AB:1756_DO:I:0

Name: Eth2:O.Data
Data Type: SINT[344]
Description:

Fig. 398: EtherNet/IP, Buffers

Values for SetTrigger and TimeOut can be entered directly into the function block or can be accessed externally by calling the tag name.



VISOR_Trigger VISOR_1_Trigger
SensorInputBuffer Eth2:O
SensorOutputBuffer ?
SetTrigger 0
Trigger_TimeOut 50

Fig. 399: EtherNet/IP, VISOR® Trigger

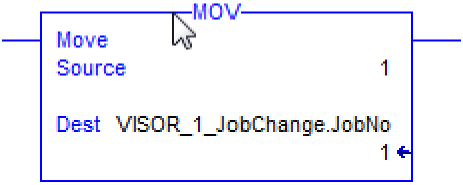


Fig. 400: EtherNet/IP, MOV

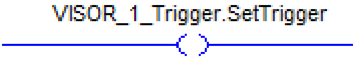
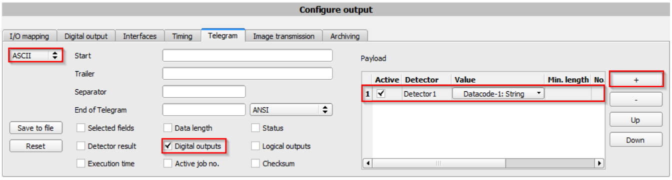


Fig. 401: EtherNet/IP, SetTrigger

5.7.7 Result data: assembly response

User defined data output to be configured in the result telegram specification:


Configure output

I/O mapping | Digital output | Interfaces | Timing | Telegram | Image transmission | Archiving

ASCII

Start:
Trailer:
Separator:
End of Telegram: ANSI

□ Selected fields
□ Detector result
□ Execution time
□ Data length
☑ Digital outputs
□ Active job no.
□ Status
□ Logical outputs
□ Checksum

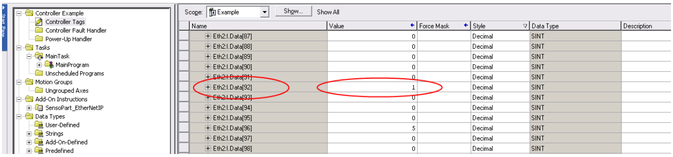
Payload

	Active	Detector	Value	Min. length	No
1	☑	Detector1	Datacode-1: String		

Fig. 402: EtherNet/IP, Result data specification

Depending on the kind of output data they can be found in the assembly response at

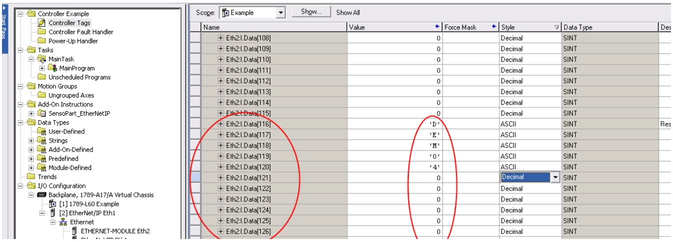
- Boolean: byte 92 (pucBool)



Name	Value	Force Mask	Style	Data Type	Description
Eth2:I.Data[87]	0		Decimal	SINT	
Eth2:I.Data[88]	0		Decimal	SINT	
Eth2:I.Data[89]	0		Decimal	SINT	
Eth2:I.Data[90]	0		Decimal	SINT	
Eth2:I.Data[91]	0		Decimal	SINT	
Eth2:I.Data[92]	1		Decimal	SINT	
Eth2:I.Data[93]	0		Decimal	SINT	
Eth2:I.Data[94]	0		Decimal	SINT	
Eth2:I.Data[95]	0		Decimal	SINT	
Eth2:I.Data[96]	5		Decimal	SINT	
Eth2:I.Data[97]	0		Decimal	SINT	
Eth2:I.Data[98]	0		Decimal	SINT	

Fig. 403: EtherNet/IP, Output data, Bool

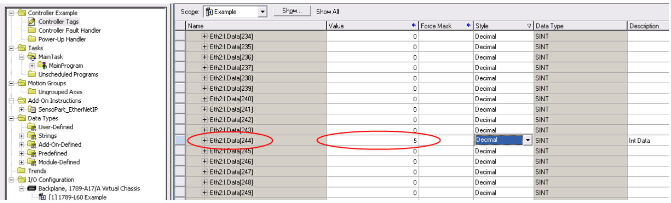
- String: byte 116 (pcString)



Name	Value	Force Mask	Style	Data Type	Des
Eth2:I.Data[108]	0		Decimal	SINT	
Eth2:I.Data[109]	0		Decimal	SINT	
Eth2:I.Data[110]	0		Decimal	SINT	
Eth2:I.Data[111]	0		Decimal	SINT	
Eth2:I.Data[112]	0		Decimal	SINT	
Eth2:I.Data[113]	0		Decimal	SINT	
Eth2:I.Data[114]	0		Decimal	SINT	
Eth2:I.Data[115]	0		Decimal	SINT	
Eth2:I.Data[116]	'D'		ASCII	SINT	Res
Eth2:I.Data[117]	'E'		ASCII	SINT	
Eth2:I.Data[118]	'M'		ASCII	SINT	
Eth2:I.Data[119]	'0'		ASCII	SINT	
Eth2:I.Data[120]	'4'		ASCII	SINT	
Eth2:I.Data[121]	0		Decimal	SINT	
Eth2:I.Data[122]	0		Decimal	SINT	
Eth2:I.Data[123]	0		Decimal	SINT	
Eth2:I.Data[124]	0		Decimal	SINT	
Eth2:I.Data[125]	0		Decimal	SINT	
Eth2:I.Data[126]	0		Decimal	SINT	

Fig. 404: EtherNet/IP, Output data, String

- Integer: byte 244 (pnInt)



Name	Value	Force Mask	Style	Data Type	Description
Eth2:I.Data[234]	0		Decimal	SINT	
Eth2:I.Data[235]	0		Decimal	SINT	
Eth2:I.Data[236]	0		Decimal	SINT	
Eth2:I.Data[237]	0		Decimal	SINT	
Eth2:I.Data[238]	0		Decimal	SINT	
Eth2:I.Data[239]	0		Decimal	SINT	
Eth2:I.Data[240]	0		Decimal	SINT	
Eth2:I.Data[241]	0		Decimal	SINT	
Eth2:I.Data[242]	0		Decimal	SINT	
Eth2:I.Data[243]	0		Decimal	SINT	
Eth2:I.Data[244]	5		Decimal	SINT	Int Data
Eth2:I.Data[245]	0		Decimal	SINT	
Eth2:I.Data[246]	0		Decimal	SINT	
Eth2:I.Data[247]	0		Decimal	SINT	
Eth2:I.Data[248]	0		Decimal	SINT	
Eth2:I.Data[249]	0		Decimal	SINT	

Fig. 405: EtherNet/IP, Output data, Int

To see boolean results of Q1 to Q3 you have to activate the transmission in SensoConfig-Software:

=> Output => Telegram => Digital Outputs

If this setting is correct, you get them on Q1 = Eth2:I.Data[60].01, Q2 = Eth2:I.Data[60].02, Q3 = Eth2:I.Data[60].03

5.7.8 EtherNet/IP Appendix

5.7.8.1 Assembly Request

Communication settings

Description:	Request posted from PLC to VISOR® vision sensor
Class:	Class 1
nAssemblyInstance:	100
nType:	AssemblyConsuming

nLength (bytes):	344
szAssemblyName:	AssemblyRequest

Assembly request

VISOR® vision sensor receives a data frame of 344 bytes.

To release commands to the sensor, proceed as follow:

Each byte corresponds to values which are sent from the PLC to the sensor. The position defines the byte to use and the size defines the length of this command.

Position	Size (bytes)	Member	Data type	Description
0	2	unKey	U16	request key, eg. a request counter
2	2	unId	U16	request ID, eg. for requests “trigger”, “change job”
4	2	unNumChar	U16	no. of valid char parameters
6	2	unNumInt	U16	no. of valid int parameters
8	256	pcValueChar[RQST_NUM_CHAR]	I8	char parameters for request, member may only hold one string
264	80	16 int parameters for request	I16	int parameters for request

The request key:

The position 0 (Byte 0) with a size of 2 bytes, corresponds to the request key. It valid the modification of parameters sent. For that, you need to increment the request key bytes with a value of your choice to release a command.

Example:

I want to make a trigger on the VISOR® vision sensor. The default code of the request key is 0x0 0x0. After Trigger configuration (description follows), I increment the request key to engage the trigger. The request key code is now: 0x0 0x2.

Position 0:

Byte 1	Byte 0
Always 0	0000 0010

Command configuration:

The request key has a size of 2 bytes (at position 0 and 1), the command configuration will start on position 2 and has a size of 2 bytes. It is possible to choose a command called **Request ID**between: Trigger, Change job, statistics reset, auto shutter, permanent or temporary teach.

Position 2:

Byte 3	Byte 2
Always 0	0000 0001

Change job: to send the change job command, the code is: 0x0 0x2 in position 2. You have to set the LSB of position 6 to "1". (Standard version: job 1 or job 2; Advanced versions: 255 jobs available). For that, write the job number 4 bytes to position / byte 264 . To validate your request, you have to increment the request key. After that you need to make a trigger to change the job (don't forget to set the LSB of position 6 to "0").

Position 2:

Byte 3	Byte 2
Always 0	0000 0010

Position 6:

Byte 7	Byte 6
Always 0	0000 0001

Position 264:

Byte 265	Byte 264
Always 0	0000 0010

Byte 3	Byte 2
Always 0	0000 0100

Auto Shutter: For function auto shutter you have to write the code: 0x0 0x7 on position 2.

Position 2:

Byte 3	Byte 2
Always 0	0000 0111

Permanent teach: The permanent teach allows to teach a new reference pattern / contour etc. with same tools and same settings. These teach is permanent, it means the new reference pattern / contour etc. is stored permanentely in the sensor memory, even if the sensor is reset. The code is: 0x0 0x8 on position 2. To activate this command, you have to launch a new trigger to catch a new picture and you have to increment the request key.

Position 2:

Byte 3	Byte 2
Always 0	0000 1000

Temporary teach: The temporary teach allows a new reference pattern / contour etc. with same tools and same setting. These teach is temporary, it means the reference pattern / contour etc. is not available after a reset of the sensor. The code is: 0x0 0x9 on position 2. To active this command, you have to launch two trigger to catch a new picture and you have to increment the request key.

Position 2:

Byte 3	Byte 2
Always on 0	0000 1001

Summary of available commands:

Commands	Position	Size	Code
Trigger	2	2	0x0 0x1
Change job	2	2	0x0 0x2
Job number	264	4	Job number
Statistics reset	2	2	0x0 0x4
Auto shutter	2	2	0x0 0x7
Permanent teach	2	2	0x0 0x8
Temporary teach	2	2	0x0 0x9

Example: I want to make a trigger, I write the code: 0x0 0x1 on position 2, I modify the request key on position 0: 0x0 0x2 => 0x0 0x4. The VISOR® vision sensor take a new picture.

Attention: Don’t forget to increment the request key to valid the commands.

5.7.8.2 Assembly Response

Communication settings

Description:	Response returned from VISOR® vision sensor to PLC
Class:	Class 1
nAssemblyInstance:	101
nType:	AssemblyProducing
nLength (bytes):	444
szAssemblyName:	AssemblyResponse

Assembly response

Assembly responses are data sent by the sensor after made some commands by the PLC or by the software.

For the commands by PLC, please see Ethernet / IP request file.

To set commands by the software with the SensoConfig, proceed as follow:

After PLC configuration and SensoConfig configuration, the size of the frame assembly response is of 444 Bytes. Each of them corresponds to some values describe as follow:

<table>
<tr><th>Position</th><th>Size (bytes)</th><th>Member</th><th>Data type</th><th>Description</th></tr>
<tr><td>0</td><td>4</td><td>unFault</td><td>U32</td><td>member is standard in Rockwell RSLogix</td></tr>
<tr><td>4</td><td>2</td><td>unKey</td><td>U16</td><td>Request key is returned in response</td></tr>
<tr><td>6</td><td>2</td><td>unId</td><td>U16</td><td>Request ID is returned in response. (Trigger, Change job, Statistics reset…)</td></tr>
<tr><td>8</td><td>2</td><td>unError</td><td>U16</td><td>Error code of response</td></tr>
<tr><td>10</td><td>4</td><td>unNumChar</td><td>U32</td><td>Responses values for requests like job change, teach …
<table>
<tr><th></th><th>Byte 13</th><th>Byte 12</th><th>Byte 11</th><th>Byte 10</th></tr>
<tr><td>Trigger</td><td>Always</td><td>Always</td><td>Always</td><td>0000</td></tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td></td><td><table><tr><td></td><td>0</td><td>0</td><td>0</td><td>0001</td></tr><tr><td>Change job</td><td>Always 0</td><td>Always 0</td><td>Always 0</td><td>0000 0010</td></tr><tr><td>Permanent teach</td><td>Always 0</td><td>Always 0</td><td>Always 0</td><td>0000 1000</td></tr></table></td></tr>
<tr><td>14</td><td>2</td><td></td><td></td><td>RESERVED</td></tr>
<tr><td>16</td><td>16</td><td>pcValueChar [RPNS_NUM_ CHAR]</td><td>18</td><td>char parameters for response, member may only hold one string</td></tr>
<tr><td>32</td><td>16</td><td>pnValueInt[RPNS_ NUM_INT]</td><td>U32</td><td>int parameters for response</td></tr>
<tr><td>48</td><td>4</td><td>unImageCount</td><td>U32</td><td>Number of images taken by the VISOR® vision sensor.</td></tr>
<tr><td>52</td><td>4</td><td>unExecutionTime</td><td>U32</td><td>Average execution time of last processed image. (To active this data, select in SensoConfig: Execution time)</td></tr>
<tr><td>56</td><td>4</td><td>pucStatus[RPNS_ IMPL_NUM_ BYTE_STATUS]</td><td>U32</td><td>Status : VISOR® vision sensor mode (To active this data, select in SensoConfig: Status)
Free run: The sensor takes a new picture when the processing is finished.
Trigger: The sensor wait an external signal to take a new picture.

Example Byte 56, bit “0” and “1”:<table><tr><td></td><td>Byte 59</td><td>Byte 58</td><td>Byte 57</td><td>Byte 56</td></tr><tr><td>Freerun</td><td>Always 0</td><td>Always 0</td><td>0000 000X</td><td>0000 0X01</td></tr><tr><td>Trigger mode</td><td>Always 0</td><td>Always 0</td><td>0000 000X</td><td>0000 0X10</td></tr></table>Additional data for 1.7.10.1 version or more
Configuration : The sensor is connected to a PC for configuration

Example Byte 56, bit “2”:</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td></td><td>
<table>
<tr><th></th><th>Byte 59</th><th>Byte 58</th><th>Byte 57</th><th>Byte 56</th></tr>
<tr><td>Configuration</td><td>Always 0</td><td>Always 0</td><td>0000 000X</td><td>0 0000 00XX</td></tr>
<tr><td>Run</td><td>Always 0</td><td>Always 0</td><td>0000 000X</td><td>0000 01XX</td></tr>
</table>
Run : The job is downloaded in the VISOR® vision sensor memory. The sensor works stand alone.
Byte “57”, bit “0” shows the sensor ready status
<table>
<tr><th></th><th>Byte 59</th><th>Byte 58</th><th>Byte 57</th><th>Byte 56</th></tr>
<tr><td>Sensor ready</td><td>Always 0</td><td>Always 0</td><td>0000 0001</td><td>0000 0XXX</td></tr>
<tr><td>Sensor not ready</td><td>Always 0</td><td>Always 0</td><td>0000 0000</td><td>0000 0XXX</td></tr>
</table>
</td></tr>
<tr><td>60</td><td>2</td><td>unActiveJob</td><td>U16</td><td>Active job : Value of job number</td></tr>
<tr><td>62</td><td>2</td><td></td><td></td><td>RESERVED</td></tr>
<tr><td>64</td><td>2</td><td>unNumDigital</td><td>U16</td><td>Number of active digital outputs (assigned to one tool) (To active this data, select in SensoConfig: Digital outputs)
According to: Byte 1 and 2, of “Digital outputs”, in “Serial communication / Data output Binary”</td></tr>
<tr><td>66</td><td>2</td><td>unNumLogic</td><td>U16</td><td>Number of active logical outputs (assigned to one tool) (To active this data, select in SensoConfig: Logical outputs)</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td></td><td>According to: Byte 1 and 2, of “Logical outputs”, in “Serial communication / Data output Binary”</td></tr>
<tr><td>68</td><td>2</td><td>unNumDetector</td><td>U16</td><td>Number of selected tools (It is a default value)
According to: Byte 2 and 3, of “Detector result”, in “Serial communication / Data output Binary”</td></tr>
<tr><td>70</td><td>2</td><td>unNumBool</td><td>U16</td><td>no. of valid boolean parameters</td></tr>
<tr><td>72</td><td>2</td><td>unNumString</td><td>U16</td><td>no. of strings included in pcValueChar</td></tr>
<tr><td>74</td><td>2</td><td>unNumInt</td><td>U16</td><td>Number of received payload (To active this data, select a data in SensoConfig: Payload)</td></tr>
<tr><td>76</td><td>2</td><td>pucDigital[RPNS_IMPL_NUM_BYTE_DIGITAL]</td><td>U8</td><td>Digital outputs results: result according to the order of the outputs. LSB => first output. MSB => Last output.
Example: 4 active outputs (12, 09, 05, 06).
Status of outputs :
12 = OK; 09 = NOK; 05 = OK; 06 = OK. The code will be :
<table>
<tr><th></th><th>Byte 79</th><th>Byte 78</th><th>Byte 77</th><th>Byte 76</th></tr>
<tr><td>Result</td><td>0000 0000</td><td>0000 0000</td><td>0000 0000</td><td>0000 1101</td></tr>
</table>
(To active this data, select in SensoConfig : Digital outputs)
According to: Byte 3 ... n, of “Digitaloutputs”, in “Serial communication / Data output Binary”</td></tr>
<tr><td>80</td><td>8</td><td>pucLogic[RPNS_IMPL_NUM_BYTE_LOGIC]</td><td>U8</td><td>Logical outputs results: result according to the order of the outputs. LSB => first output. MSB => Last output.
Example: 4 active outputs (12, 09, 05, 06).
Status of outputs :
12 = OK; 09 = NOK; 05 = OK; 06 = OK. The code will be : 1011
<table>
<tr><th></th><th>Byte 83..87</th><th>Byte 82</th><th>Byte 81</th><th>Byte 80</th></tr>
<tr><td>Result</td><td>0000 0000</td><td>0000 0000</td><td>0000 0000</td><td>0000 1011</td></tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td></td><td>(To active this data, select in SensoConfig: Logical outputs)
According to: Byte 3 ... n, of “Logical outputs”, in “Serial communication / Data output Binary”</td></tr>
<tr><td>88</td><td>1</td><td>pucDetector [RPNS_IMPL_ NUM_BYTE_ DETECTOR]</td><td>U8</td><td>Global result (Only available on SensoConfig and SensoView):
Only coded on the third LSB bits.
Bit0 = Global result status (0 : OK ; 1 : NOK)
Bit1 = Status of the case « Detector result » in Optional field during the data configuration.
Bit2 = Indicate if one of tools is NOK even if result global is OK => 0
Example 1: We select Detector result case; Tool1 OK; Tool 2 OK; Global result on tool1 and on Tool 2 => OK, the bit2 will be on 1.
<table>
<tr><td></td><td>Byte 88</td></tr>
<tr><td>Result</td><td>0000 0111</td></tr>
</table>
Example 2: We select Detector result case; Tool1 OK; Tool2 NOK; Global result on tool1 => OK, the bit2 will be on 0.

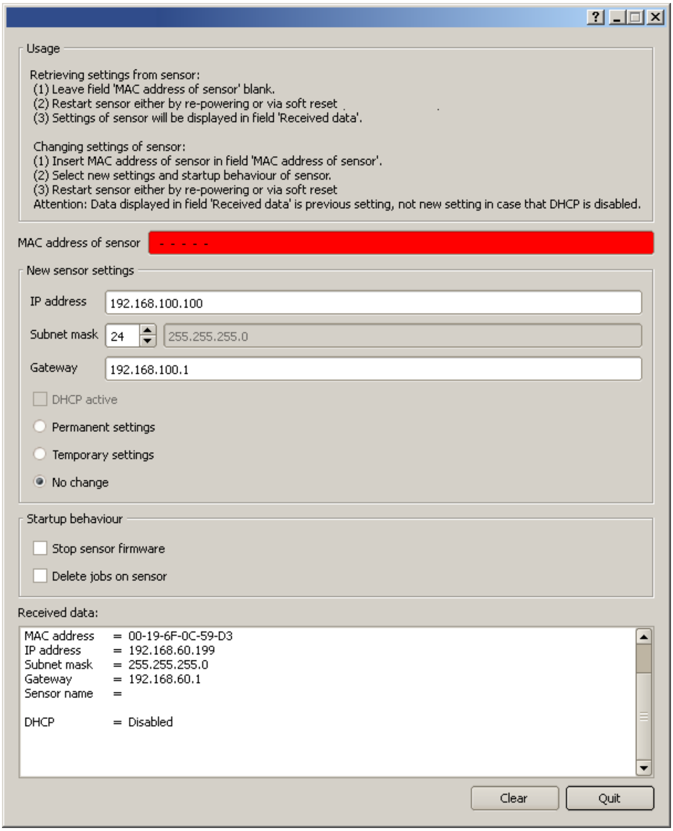
According to: Byte 1, of “Detector result”, in “Serial communication / Data output Binary”
<table>
<tr><td></td><td>Byte 88</td></tr>
<tr><td>Result</td><td>0000 0011</td></tr>
</table>
Other bits always on 0.
(To active this data, select in SensoConfig: Detector results)</td></tr>
<tr><td>89</td><td>3</td><td>pucDetector [RPNS_IMPL_ NUM_BYTE_ DETECTOR]</td><td>U8</td><td>Detector result: Each bit corresponds to a tool. Only on 1Byte: Bit1 = tool1; bit2 = tool2; bit3 = tool3... until 8 bits.
Other bytes, always on 0.
Future Applications, coded on 3 bytes.
(To active this data, select in SensoConfig: Detector results)</td></tr>
<tr><td>92</td><td>4</td><td>pucBool[RPNS_ IMPL_NUM_ BYTE_BOOL]</td><td>U8</td><td>boolean results (bitwise) as configured in HMI (listbox)</td></tr>
<tr><td>96</td><td>16</td><td>punStringLength</td><td>U16</td><td>lengths of strings included in pcValueChar</td></tr>
</table>

		[RPNS_IMPL_NUM_STRING]		
112	2	pucStringTruncated[RPNS_IMPL_NUM_BYTE_STRING_TRUNCATED]	U8	indicates for each string whether it has been truncated (bitwise)
114	2			RESERVED
116	128	pcString[RPNS_IMPL_NUM_BYTE_STRING]	I8	char result as configured in HMI (listbox), member may hold multiple strings
244	200	pnInt[RPNS_IMPL_NUM_INT]	U32	Results of payload configured on SensoConfig in tab « frame ». All data on payload are describe as follow :

5.8 SensoRescue

The utility “SensoRescue” is used to reset VISOR® sensors, which no longer can be found by SensoFind, to a default status to be able to be accessed via SensoFind and SensoConfig again.

- Start SensoRescue (leave empty field “Mac address of Sensor”).
- Reset VISOR®, Power off/on or SensoFind/File/Sensor soft reset (VISOR® must be connected via Ethernet and be located in the same network as the PC).
- In the field below “Received Data” now all settings of the VISOR® are displayed.



Usage

Retrieving settings from sensor:
(1) Leave field 'MAC address of sensor' blank.
(2) Restart sensor either by re-powering or via soft reset
(3) Settings of sensor will be displayed in field 'Received data'.

Changing settings of sensor:
(1) Insert MAC address of sensor in field 'MAC address of sensor'.
(2) Select new settings and startup behaviour of sensor.
(3) Restart sensor either by re-powering or via soft reset
Attention: Data displayed in field 'Received data' is previous setting, not new setting in case that DHCP is disabled.

MAC address of sensor: - - - - -

New sensor settings

IP address: 192.168.100.100

Subnet mask: 24 — 255.255.255.0

Gateway: 192.168.100.1

☐ DHCP active

○ Permanent settings

○ Temporary settings

● No change

Startup behaviour

☐ Stop sensor firmware

☐ Delete jobs on sensor

Received data:

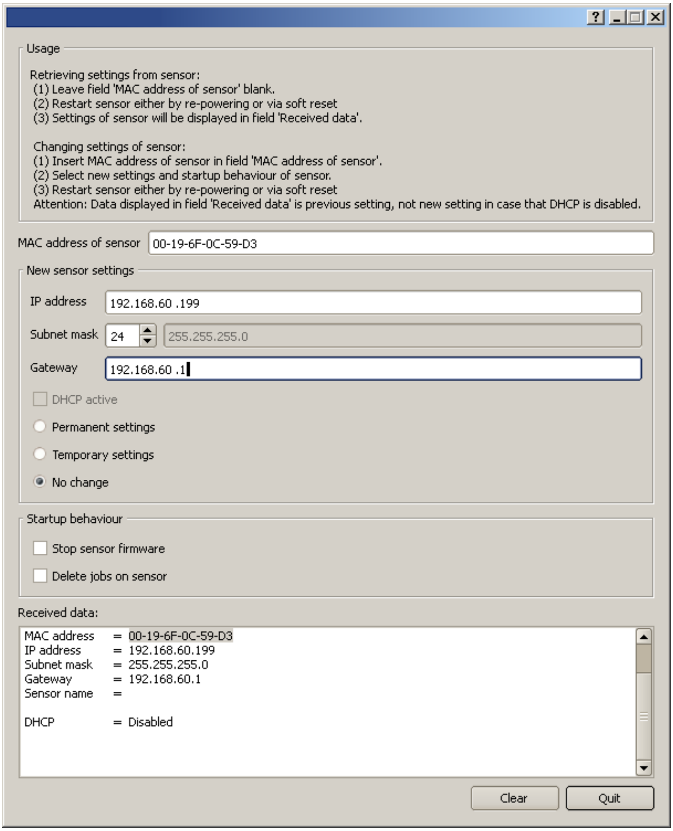
```
MAC address   = 00-19-6F-0C-59-D3
IP address    = 192.168.60.199
Subnet mask   = 255.255.255.0
Gateway       = 192.168.60.1
Sensor name   =

DHCP          = Disabled
```

Fig. 406: SensoRescue /1

- Now the below shown Mac address can be entered into the field “Mac address of Sensor”.
- Into the lines below, all the network settings like, IP address, Subnet Mask etc., which the VISOR® should have after the next Restart (Power off/on), can be entered.
- Restart VISOR®.

Attention:

The after the next restart displayed data are the old ones as they are not refreshed by sensor restart.



Usage

Retrieving settings from sensor:
(1) Leave field 'MAC address of sensor' blank.
(2) Restart sensor either by re-powering or via soft reset
(3) Settings of sensor will be displayed in field 'Received data'.

Changing settings of sensor:
(1) Insert MAC address of sensor in field 'MAC address of sensor'.
(2) Select new settings and startup behaviour of sensor.
(3) Restart sensor either by re-powering or via soft reset
Attention: Data displayed in field 'Received data' is previous setting, not new setting in case that DHCP is disabled.

MAC address of sensor: 00-19-6F-0C-59-D3

New sensor settings

IP address: 192.168.60 .199

Subnet mask: 24 255.255.255.0

Gateway: 192.168.60 .1

□ DHCP active

○ Permanent settings

○ Temporary settings

● No change

Startup behaviour

□ Stop sensor firmware

□ Delete jobs on sensor

Received data:

```
MAC address    = 00-19-6F-0C-59-D3
IP address     = 192.168.60.199
Subnet mask    = 255.255.255.0
Gateway        = 192.168.60.1
Sensor name    =

DHCP           = Disabled
```

Fig. 407: SensoRescue / 2

6 Image settings and accessories

6.1 Good images

To achieve good images follow these steps:

- Align the sensor to the desired field of view. Take care for stable mounting.
- For high contrast images adjust angles and illumination as described in chapter The most important types of illumination are: Bright field, Dark field and Diffuse illumination.
- Adjust a sharp image with the focus screw at the backside of the sensor housing.
- Adjust the brightness of the image with the parameter “Shutter speed” in SensoConfig/Job/Image acquisition. (Do not use parameter “Gain”, not until you are not able to achieve desired brightness via “Shutter speed”).

6.2 Environmental light, shrouding, IR- version

Mechanical shrouding

In most cases it´s much simpler and highly cost effective to protect the scene against disturbing light or sun beams, which e.g. shine temporary at a certain time of day or season from windows or roof lights, by mechanical shrouding like metal plates, than to create illumination conditions, e.g. by additional illumination which is strong enough not to be disturbed in any situation.

Version with Infrared illumination

A further elegant way to get independent from the environmental light is to use the according VISOR® version with Infrared illumination. Here the scene get´s illuminated with the built in powerful IR- illumination. The receiver is equipped with the according filter. That means the sensor works in a narrow range of this specific wavelength, and for that as far as possible with its own light only.

Another advantage of the infrared light is, that the light flashes are not visible and do not disturb any human workers which are near the plant.

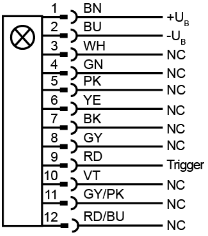
6.3 External illumination

For the VISOR® a large range of accessories is available, which also covers a big range of external illuminations, which can be used additionally or instead of the internal illumination.

Further information on vision accessory: http://www.sensopart.com/de/download.

The both types LF45 xxx and LFR115 xxx can be connected directly to the sensor.

Anschluss / Wiring / Raccordement



Anschluss Ringlicht mit Sensor / Connection ring light with sensor / Raccordement lumière externe avec le capteur

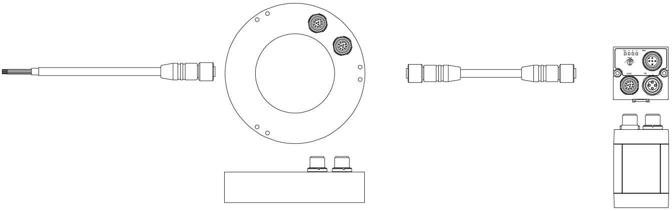
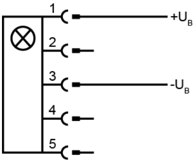


Fig. 408: Connection of external illumination LF45 xxx and LFR45 xxxAll other listed types are connected to the VISOR® as follows.

Anschluss / Wiring / Raccordement



Anschluss Ringlicht - Anschlussadapter - Sensor / Connection ring light - connection adapter - sensor / Raccordement lumière ronde - adapteur de raccordement - capteur

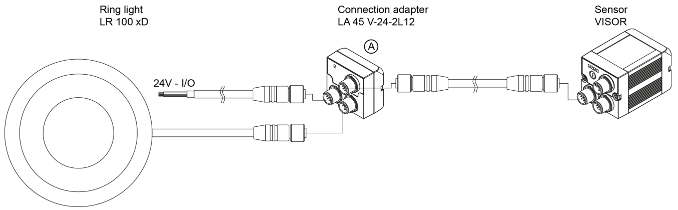


Fig. 409: Connection of external illumination, all types except LF45 xxx and LFR115 xxx.

6.4 The most important types of illumination are: Bright field, Dark field and Diffuse illumination.

6.4.1 Bright field illumination

Bright field internal / Bright field external

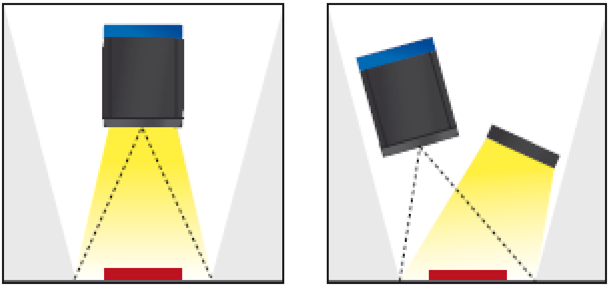


Fig. 410: Bright field illumination

With bright field lighting, the lighting, sensor and object are arranged so that the object's surface reflects the light directly into the sensor. The smooth surface of the object appears as a bright area and each indentation, bump or defect, such as e.g. scratches, are a dark edge.

Attention: With bright field lighting, the angle of alignment between the lighting, object and sensor and the object's surface is critical as direct reflection by the object's surface only works when the angle and surface characteristics (shiny, mat, oily ….) are constant!

With Bright field / With Dark field

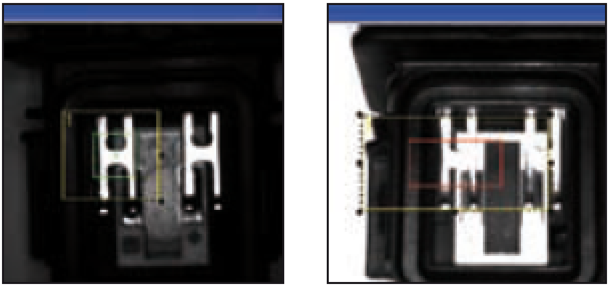


Fig. 411: Example Bright field illumination

By the direct reflection of the highly reflective (shiny) metal part, even before a white background, this is possible to be distinguished and recognized with Bright field illumination! With Dark field illumination it´s not possible to distinguish between shiny metal part and white background!

6.4.2 Dark field illumination

Dark field internal / Dark field external

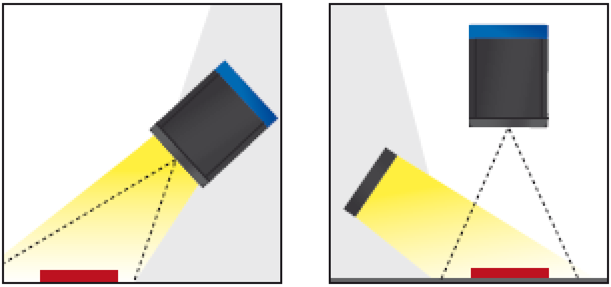


Fig. 412: Dark field illumination

With dark field illumination, the lighting, sensor and object are arranged so that the smooth surface of the object does not reflect the light directly into the sensor. Object edges (indentations and bumps) appear as bright areas, smooth object surfaces however are dark. This type of illumination functions with wide angle ranges and depends little on the object’s surface.

With Bright field / With Dark field

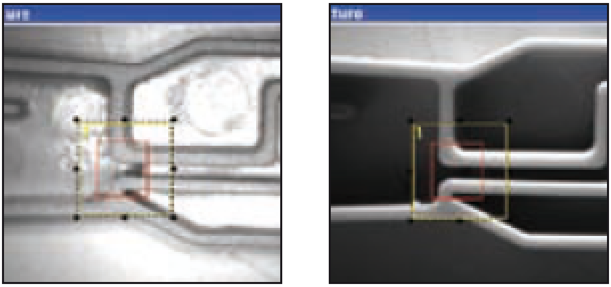


Fig. 413: Example, Dark field

Edges are clearly accentuated with Dark field illumination.

6.4.3 Diffuse illumination (external only)

Diffuse external

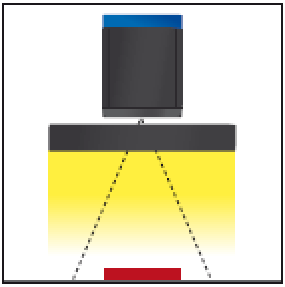


Fig. 414: Diffuse illumination

Diffuse lighting is used everywhere where highly-reflective, curved and above all irregularly-shaped object surfaces are concerned (e.g. aluminium foil on blister packs etc.). Such objects cannot be illuminated with spot-shaped lighting, but only with diffuse lighting (i.e. even lighting from all directions). Diffuse lighting is also known as “cloudy day” illumination, i.e. uniform light from behind the cover of clouds rather than from direct sunlight.

Spot illumination / Diffuse illumination

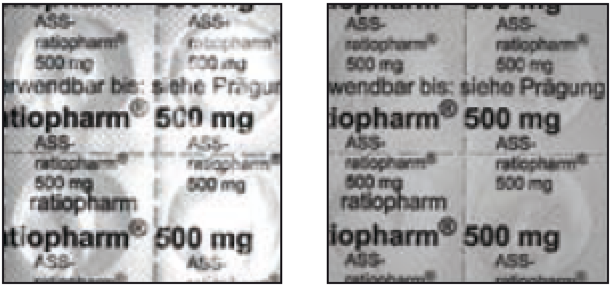


Fig. 415: Diffuse illumination

That means; clear homogeneous image with diffuse illumination! With any spot illumination the reflections of the aluminium foil from one part to another are always different.

6.5 IO-Box as IO-Extension (RS422)

Via the IO-Box the digital in- and outputs can be extended (8 inputs, 32 outputs), or an encoder controlled ejector can be realized. The connection and parameter setting of the I/O-Box is described in document: "IO-Mounting and operating instructions" in:
Startmenue/SensoPart /VISOR® vision sensor/Documentation/...

7 Technical Data

Electrical data	
Operating voltage	U_B 24 V DC , -25% / +10%
Residual ripple	< 5 Vss
Current consumption (no I/O)	≤ 200 mA
All inputs	PNP / NPN High > U_B - 1 V, Low < 3 V
Input resistance	> 20 kOhm
Encoder input	High > 4 V, max. 18kHz
Outputs	PNP / NPN
Maximum output current (per output)	50 mA, Ejector (Pin 12 / RDBU) 100 mA
Short-circuit protection (all outputs)	yes
Inductive load	typ.: Relays 17K / 2H, pneumatic valve 1.4K / 190mH
Protection against inverse polarity	yes
Interfaces VISOR®-XX-Standard Interfaces VISOR®-XX-Advanced	Ethernet (LAN) Ethernet (LAN), RS422/RS232
Readiness delay	Typ. 13 s after power on

Optical data	
Number of pixels , chip size, pixel size	VISOR®- V10...: 736 (H) x 480 (V), 1/3", 6,0 um square VISOR®- V20...: 1280 (H) x 1024 (V), 1/1.8", 5,5 um square
Technology	CMOS (mono / color)
Integrated scan illumination	8 LEDs (except C-Mount)
Integrated lens, focal length	6, 12 or 25 mm, adjustable focus

Optical data				
	V10	V10	V10	V20
Lens (adjustable to infinity)	6	12	25	12
Min. scan distance	6	30	140	30
Min. field of view X x Y	5 x 4	8 x 6	18 x 14	16 x 13

Mechanical data	
Length x width x height	65 x 45 x 45 mm (without plug)
Weight	approx.160 g
Vibration / shock	EN 60947-5-2
Ambient operating temperature	0° C .... 50° C (80% humidity, non-condensing)
Storage temperature	-20° C ... 60° C (80% humidity, non-condensing)
Protective system	IP 65/67
Plug connection	24V DC and I/O M12 12-pin, LAN M12 4-pin, Data M12 5-pin
Housing material	aluminium, plastic

Function and characteristics	
Object detection	
Number of jobs / detectors	VISOR®-XX-Standard: 8 / 32 VISOR®-XX-Advanced: max. 255 / max. 255
Evaluation modes	• alignment • contour match with/without position detection • pattern match with/without position detection • area test gray level • area test contrast • area test brightness • direction info, or coordinates for position detection VISOR®-XX-Advanced: • Caliper, distances between edges

Function and characteristics	
	• BLOB, object evaluation and counting
Typical cycle time	typ. 20 ms pattern matching typ. 30 ms contour typ. 2 ms area test
Code Reader	
Number of jobs / detectors	VISOR®-XX-Standard: 8 / 2 VISOR®-XX-Advanced / Professional: max. 255 / max. 255
Evaluation modes	• DataMatrix Code acc. ECC200 in any rotational position, square and Rectangular. • QR-Code, Model 1 and Model 2, Version 1 . . . 40 • Barcode Interleaved 2 of 5, Code 39, EAN13-Gruppe (EAN8, EAN13, UPC-A, UPC-E), EAN128 (Codes A, B, C), PDF417 • OCR Optical character recognition (Professional) • position and size of field of view freely adjustable • logic operation of single configuration (AND, OR = sorting) • verify
Typical cycle time	40 ms one evaluation Coder reading, 10 ms per character OCR
Solar	
Number of jobs / detectors	VISOR®-XX-Standard: 8 / 32 VISOR®-XX-Advanced: max. 255 / max. 255
Evaluation modes	• Wafer position and dimensions • Chip and hole detection • Output of all inspection parameters • area test gray level • area test contrast • area test brightness VISOR®-XX-Advanced: • additionally bus bar localization • Caliper, distances between edges • alignment
Typical cycle time	100 ms for one evaluation
Color	
Number of jobs / detectors	VISOR®-XX-Standard: 8 / 32

Function and characteristics	
	VISOR®-XX-Advanced: max. 255 / max. 255
Evaluation modes	• alignment (Advanced) • contour match with/without position detection • pattern match with/without position detection • area test gray level • area test contrast • area test brightness • direction info, or coordinates for position detection • color value • color area • color list
Typical cycle time	typ. 30 ms pattern match typ. 60 ms contour typ. 2 ms brightness typ. 2 ms contrast typ. 2 ms gray threshold typ. 2 ms color value typ. 30 ms color area typ. 2 ms color list
Allround	
Number of jobs / detectors	VISOR® V20-ALL ... : max. 255 / max. 255
Evaluation modes / Typical cycle time	• alignment • contour match with/without position detection • pattern match with/without position detection • area test gray level • area test contrast • area test brightness • direction info, or coordinates for position detection • Caliper, distances between edges • BLOB, object evaluation and counting • color value • color area • color list • DataMatrix Code acc. ECC200 in any rotational position, square and Rectangular. • QR-Code, Model 1 and Model 2, Version 1 . . . 40 • Barcode Interleaved 2 of 5, Code 39, EAN13-Gruppe (EAN8, EAN13, UPC-A, UPC-E), EAN128 (Codes A, B, C), PDF417 • OCR Optical character recognition (Professional) • position and size of field of view freely adjustable • logic operation of single configuration (AND, OR = sort-

Function and characteristics	
	ing) • verify

8 Addendum

8.1 Telegram, tab Data output

This topic describes the telegrams available for the VISOR® vision sensor. These telegrams can be sent to VISOR® vision sensor via different interfaces.

- Serial interface RS422/RS232
- EtherNet TCP/IP
- PROFINET (request/response modul)

The telegrams are available in ASCII and Binary format. The format can be selected in the modul “SensoConfig”, in tab “data output” of the setup “Output”.

The following settings are possible:

Communication	**TCP / IP**	**RS422 / RS232**	**EtherNet/IP**	**PROFINET**
Telegram format	ASCII / Binary	ASCII / Binary	Binary	Binary

8.1.1 Overview VISOR® vision sensor telegram:

8.1.1.1 VISOR® in general

- **Reset statistic** (ASCII /Binary)
  With the telegram “Reset statistic” the internal statistic counter of the VISOR® vision sensor can be reset.

8.1.1.2 VISOR® control

- **Trigger** (ASCII / Binary)
  With the telegram “Trigger” an image acquisition will be started. Some commands need an additional image acquisition. The result data of the image are transferred via the “Out” port.
- **Extended trigger** (ASCII/ Binary)
  This telegram is an enhancement to the “trigger” telegram. Besides the result data there is also the option to assign an ID or to receive information about the operating mode (run-/config). Unlike the “trigger” telegram the result data of “Extended trigger” telegram are also transferred via the “In” port.
- **Change Job** (ASCII/ Binary)
  With the telegram “Change job” a job change on the VISOR® vision sensor is initiated.

- **Change job permanent** (ASCII/ Binary)
  With the telegram "Change job permanent" a permanent job change on the VISOR® vision sensor is initiated. The job runs again after restarting.

8.1.1.3 VISOR® job settings

- **Set shutter speed** (ASCII/Binary)
  With the telegram "Set shutter speed" the shutter speed of the job can be changed. The telegram can be used for e.g. brightness compensation.
- **Get shutter speed** (ASCII/ Binary)
  With the telegram "Get shutter speed" the set shutter speed of the job can be read.
- **Set gain** (ASCII/Binary)
  With the telegram "Set gain" the gain of the job can be changed. The telegram can be used for e.g. brightness compensation.
- **Get gain** (ASCII/Binary)
  With the telegram "Get gain" the set gain of the job can be read.
- **Set parameter** (ASCII/Binary)
  With the telegram "Set parameter" the detector parameters can be adjusted, e.g. reference strings, detector thresholds.
- **Get parameter** (ASCII/Binary)
  With the telegram "Get parameter" the set parameters of the detector can be read.
- **Set ROI** (ASCII/Binary)
  With the telegram "Set ROI" the position of the selected detector can be changed.
- **Get ROI** (ASCII/Binary)
  With the telegram "Get ROI" the position of the selected detector can be read.
- **Get job list** (ASCII/Binary)
  With the telegram "Get job list" a list of all available jobs on the VISOR® vision sensor will be displayed.
- **Get detector list** (ASCII/Binary)
  With the telegram "Get detector list " a list of all detectors in the current job will be displayed.
- **Teach detector** (ASCII/Binary)
  With the telegram "Teach detector" the selected detector is re-taught (only for pattern matching and contour detection).

8.1.1.4 VISOR® calibration

- **Add point** (ASCII/Binary)
  With the telegram "Add point" a point is added to the point list.
- **Clear point list** (ASCII/Binary)
  With the telegram "Clear point list" the point list will be initialized.

- **Calibration point list** (ASCII/Binary)
  With the telegram “Calibration point list” the calibration is carried out using the point list in the current job.
- **Calibration calibration plate (only VISOR® Allround Professional)** (ASCII/Binary)
  With the telegram “Calibration calibration plate” the calibration is carried out using the calibration plate.
- **Validate calibration** (ASCII/ Binary)
  With the telegram “Validate calibration” the calibration is validated using the point list.
- **Copy calibration**(ASCII/Binary)
  With the telegram “Copy calibration” the calibration of the current job will be copied to the selected destination.

8.1.1.5 VISOR® visualization

- **Get image** (ASCII/Binary)
  With the telegram “Get image” the image from VISOR® vision sensor will be received.

8.1.1.6 VISOR® service (only available on port 1998 and in ASCII format)

- **Get sensor identity** (ASCII)
  With the telegram “Get sensor identity” the current firmware status as well as the hardware type can be queried.
- **Update firmware** (ASCII)
  With the telegram “Update firmware” the firmware updates are started. Previously the firmware file must be loaded onto the VISOR® vision sensor.
- **Set jobset** (ASCII)
  With the telegram “Set jobset” the jobset of the VISOR® vision sensor can be changed. Previously the jobset file must be loaded onto the VISOR® vision sensor.
- **Get jobset** (ASCII)
  With the telegram “Get jobset” the jobset of the VISOR® vision sensor can be read.

8.1.1.7 Data output

(ASCII/Binary)

This section provides information about the data output, especially, which format the individual results receive.

8.1.2 Serial Communication ASCII

Reset statistic (ASCII)

Reset statistic (ASCII) request string to sensor		
Byte No.	Content	Significance
1	R	Reset statistic
2	S	
3	T	
Example:	RST	
Reset statistic (ASCII) response string from sensor		
Byte No.	Content	Significance
1	R	Reset statistic
2	S	
3	T	
4	P F	P Pass F Fail
Example:	RSTP	
Additional information:		
Accepted in run mode:		Yes
Accepted in config mode:		No
Accepted when Ready is Low:		Yes
Status of Ready signal during processing		Not altered
End of telegram		Max. 4 byte (option)

Trigger (ASCII)

Trigger (ASCII) request string to sensor		
Byte no.	ASCII contents	Significance
1	T	Trigger, (simple trigger without index, in-port)
2	R	
3	G	
Example:	TRG	
Trigger (ASCII) response string from sensor		
Byte no.	Contents	Significance
1	T	Trigger, (response to trigger without index, via port 2006. If defined: result date without index via port 2005)
2	R	
3	G	
4	P F	P: Pass F: Fail
Example:	TRGP	
Additional information:		
Accepted in run mode:		Yes
Accepted in configuration mode:		Yes
Accepted when ready low:		No
Status of ready signal during processing:		Low
End of telegram:		Max. 4 byte (option)

Extended Trigger (ASCII)

Extended Trigger (ASCII) request string to sensor		
Byte no.	ASCII contents	Significance
1	T	Extended Trigger, (trigger with index, for correlation of trigger to the corresponding result data, via port 2006)
2	R	
3	X	
4 - 5	X	Length of following data (n)
6...n	X	Data
Example:	TRX06MyPart	
Extended Trigger (ASCII) response string from sensor		
Byte no.	ASCII contents	Significance
1	T	Extended Trigger, (response to trigger with index and result data, via port 2006, for correlation of trigger to the corresponding result. Result data without index via port 2005 also)
2	R	
3	X	
4	P F	Pass Fail
5 - 6	X	Length of following data (n)
7... n	x	Data of request command

n+1	C
	R
		C = Config
		R = Run

n+2 ... n+9	X	Length of following result data (m)
n+9... m	X	Result data
Additional information:		
Accepted in run mode:		Yes
Accepted in configuration mode:		Yes

Accepted when Ready Low:	No
Status of Ready signal during processing:	Low
End of telegram	Max. 4 byte (option)

Change Job (ASCII)

Change Job (ASCII) request string to sensor		
Byte no.	ASCII contents	Significance
1	C	Change job
2	J	
3	B	
4 - 6	X	Job number
Example	CJB005	
Change Job (ASCII) response string from sensor		
Byte no.	Contents	Significance
1	C	Change job
2	J	
3	B	
4	P F	Pass Fail
5	T F	Triggered Free-run
6 - 8	X	Job number
Example 1: Example 2:	CJBPT005 CJBFF005	
Additional information:		
Accepted in run mode:		Yes
Accepted in configuration mode:		No
Accepted when Ready Low:		Yes
Status of Ready signal during processing:		Low
End of telegram		Max. 4 byte (option)

Change job permanent (ASCII)

Change Job permanent (ASCII) request string to sensor		
Byte no.	ASCII contents	Significance
1	C	Change job permanent
2	J	
3	P	
4 - 6	X	Job number
Example	CJP005	
Change Job permanent (ASCII) response string from sensor		
Byte no.	Contents	Significance
1	C	Change job permanent
2	J	
3	P	
4	P F	Pass Fail
5	T F	Triggered Free-run
6 - 8	X	Job number
Example 1: Example 2:	CJPPT005 CJPFF005	
Additional information:		
Accepted in run mode:		Yes
Accepted in configuration mode:		No
Accepted when Ready Low:		Yes
Status of Ready signal during processing:		Low
End of telegram		Max. 4 byte (option)

Set shutter speed (ASCII)

Set shutter speed (ASCII) request string to sensor		
Byte No.	Contents	Significance
1	S	Set shutter speed
2	S	
3	P T	Permanent Temporary
4 - 5	X	Number of chars of shutter value, e.g. 04
6 - 9	X	New shutter value in microseconds, e.g. 8000 = 8 ms
Example:	SSP048000	
Set shutter speed(ASCII) response string from sensor		
Byte No.	Contents	Significance
1	S	Set shutter speed
2	S	
3	P T	Permanent Temporary Permanent change affect all parameters, including those that have only been changed temporarily.
4	P F	P Pass F Fail
Example:	SSPP	
Additional information:		
Accepted in run mode:		Yes
Accepted in configuration mode:		No
Accepted when Ready Low:		Yes
Status of Ready signal dur-		Low

ing processing:	
End of telegram:	Max. 4 byte (option)

Get shutter speed (ASCII)

Get Shutter (ASCII) request string to sensor (since version 1.6.5.3)		
Byte No.	Contents	Significance
1	G	Get shutter speed (from active job)
2	S	
3	H	
Example	GSH	
Get Shutter (ASCII) response string from sensor		
Byte No.	Contents	Significance
1	G	Get shutter speed
2	S	
3	H	
4	P F	P Pass F Fail
5	X	Shutter value length
6 ... n	X	Shutter value
Example run mode: Example configuration mode:	GSHP41200 GSHF0	
Additional information:		
Accepted in run mode:		Yes
Accepted in configuration mode:		No
Accepted when Ready Low:		Yes
Status of Ready signal during processing:		Not altered
End of telegram:		Max. 4 byte (option)

Set Gain (ASCII)

Set Gain (ASCII), request string to sensor		
Byte No.	Content	Significance
1	S	Set Gain
2	G	
3	A	
4	1 0	1 - Permanent 0 - Temporary Permanent change affect all parameters, including those that have only been changed temporarily.
5 - 9	X	New gain value (in value * 1000), e.g. 2.0 = 02000
Example	SGA102000	
Set Gain (ASCII) response string from sensor		
Byte No.	Contents	Significance
1	S	Set Gain
2	G	
3	A	
4	P F	P Pass F Fail
5 - 9	X	Current gain
Example	SGAP02000	
Additional information:		
Accepted in run mode:		Yes
Accepted in configuration mode:		No
Accepted when Ready Low:		Yes
Status of Ready signal dur-		Low

ing processing:	
End of telegram	Max. 4 byte (option)

Get gain (ASCII)

Get gain (ASCII) request string to sensor		
Byte No.	Content	Significance
1	G	Get Gain
2	G	
3	A	
Example	GGA	
Get gain (ASCII) response string from sensor		
Byte No.	Content	Significance
1	G	Get Gain
2	G	
3	A	
4	P F	P: Pass F: Fail
5 - 9	X	Current gain (in value * 1000), e.g.. 1,0 = 01000
Example	GGAP01000	
Additional information:		
Accepted in Run mode		Yes
Accepted in Config mode		No
Accepted when Ready is Low		Yes
Status of Ready signal during processing		Not altered
End of telegram		Max. 4 byte (option)

Set parameter (ASCII)

Set parameter (ASCII) request string to sensor		
Byte No.	Contents	Significance
1	S	Set parameter
2	P	
3	P T	P Permanent T Temporary Permanent change affect all parameters, including those that have only been changed temporarily.
4 - 6	X	Detector No.
7 - 9	X	Command: Set reference string / value *1), see below !
10 - 14	X	Length of reference string / value in Bytes (n)
15 ... n	X	Reference string / value
Example	SPP0010010044196	
Set parameter (ASCII) response string from sensor		
Byte No.	Contents ASCII	Significance
1	S	Set parameter
2	P	
3	P T	P Permanent T Temporary
4	P F	P Pass F Fail

5	S	SI08 - Signed Integer 08 UI08 - Unsigned Integer 08 SI16 - Signed Integer 16 UI16 - Unsigned Integer 16 SI32 - Signed Integer 32 UI32 - Unsigned Integer 32 SI40 - Signed Integer 40 UI40 - Unsigned Integer 40 FLOT - Float DOBL - Double STRG - String BOOL - Boolean SP08 - Special Signed 8 UDEF - Undefined IARR - Integer Array ZERO - Default Zero Parameter
6	T	
7	R	
8	G	
Example	SPPPSTRG	
Additional information:		
Accepted in run mode:		Yes
Accepted in configuration mode:		No
Accepted when Ready Low:		Yes
Status of Ready signal during processing:		Low
End of telegram		Max. 4 byte (option)

*1) Byte No. 7: Command: set reference string / value:

Detector	**Function**	**Command**
Alignment Pattern matching	Threshold Min Threshold Max ResultOffsetPos_X ResultOffsetPos_Y	001 002 031 032

Detector	Function	Command
	ResultOffsetAngle	033
Alignment Contour	Threshold Min Threshold Max ResultOffsetPos_X ResultOffsetPos_Y ResultOffsetAngle	001 002 031 032 033
Alignment Edge	Threshold Min Threshold Max Transition_Horizontal Transition_Vertical	001 002 101 102
Pattern matching	Threshold Min Threshold Max ResultOffsetPos_X ResultOffsetPos_Y ResultOffsetAngle	001 002 031 032 033
Contour	Threshold Min Threshold Max ResultOffsetPos_X ResultOffsetPos_Y ResultOffsetAngle	001 002 031 032 033
Gray Level	Threshold Min Threshold Max GrayMin GrayMax GrayInvert	001 002 101 102 103
Contrast	Threshold Min Threshold Max	001 002
Brightness	Threshold Min Threshold Max	001 002
Barcode	Reference String	101
Datacode	Reference String	101
OCR	Reference String	101
Color Value	ColorMinChannel1 ColorMaxChannel1 ColorInvertChannel1 ColorMinChannel2	101 102 103 104

Detector	Function	Command
	ColorMaxChannel2 ColorInvertChannel2 ColorMinChannel3 ColorMaxChannel3 ColorInvertChannel3	105 106 107 108 109
ColorArea	ColorMinChannel1 ColorMaxChannel1 ColorInvertChannel1 ColorMinChannel2 ColorMaxChannel2 ColorInvertChannel2 ColorMinChannel3 ColorMaxChannel3 ColorInvertChannel3	101 102 103 104 105 106 107 108 109
BLOB	GrayAbsoluteMin GrayAbsoluteMax GrayAbsoluteInvert	101 102 103

Get parameter (ASCII)

Get parameter (ASCII) request string to sensor		
Byte No.	Contents	Significance
1	G	Get parameter
2	P	
3	A	
4 - 6	X	Detector No. e.g. 001
7 - 9	X	Command: Set reference string / value *1), see below !
Example	GPA001001	
Get parameter (ASCII) response string from sensor		
Byte No.	Contents	Significance
1	G	Get parameter
2	P	
3	A	
4	P F	P Pass F Fail

5	S	SI08 - Signed Integer 08 UI08 - Unsigned Integer 08 SI16 - Signed Integer 16 UI16 - Unsigned Integer 16 SI32 - Signed Integer 32 UI32 - Unsigned Integer 32 SI40 - Signed Integer 40 UI40 - Unsigned Integer 40 FLOT - Float DOBL - Double STRG - String BOOL - Boolean SP08 - Special Signed 8 UDEF - Undefined IARR - Integer Array ZERO - Default Zero Parameter
6	T	
7	R	
8	G	
9 - 13	X	Length of Reference strings / value (n) e.g. 00005
14...n	X	Reference string / value
Example	GPAPSTRG00005Test1	
Additional information:		
Accepted in run mode:		Yes
Accepted in configuration mode:		No
Accepted when Ready Low:		Yes
Status of Ready signal during processing:		Not altered
End of telegram		Max. 4 byte (option)

*1) Byte No. 7: Command: Get reference string / value:

Detector	**Function**	**Command**
Alignment Pattern matching	Threshold Min Threshold Max ResultOffsetPos_X	001 002

Detector	Function	Command
	ResultOffsetPos_Y ResultOffsetAngle	031 032 033
Alignment Contour	Threshold Min Threshold Max ResultOffsetPos_X ResultOffsetPos_Y ResultOffsetAngle	001 002 031 032 033
Alignment Edge	Threshold Min Threshold Max Transition_Horizontal Transition_Vertical	001 002 101 102
Pattern matching	Threshold Min Threshold Max ResultOffsetPos_X ResultOffsetPos_Y ResultOffsetAngle	001 002 031 032 033
Contour	Threshold Min Threshold Max ResultOffsetPos_X ResultOffsetPos_Y ResultOffsetAngle	001 002 031 032 033
Gray Level	Threshold Min Threshold Max GrayMin GrayMax GrayInvert	001 002 101 102 103
Contrast	Threshold Min Threshold Max	001 002
Brightness	Threshold Min Threshold Max	001 002
Barcode	Reference String	101
Datacode	Reference String	101
OCR	Reference String	101
Color Value	ColorMinChannel1 ColorMaxChannel1	101 102

Detector	Function	Command
	ColorInvertChannel1	103
	ColorMinChannel2	104
	ColorMaxChannel2	105
	ColorInvertChannel2	106
	ColorMinChannel3	107
	ColorMaxChannel3	108
	ColorInvertChannel3	109
ColorArea	ColorMinChannel1	101
	ColorMaxChannel1	102
	ColorInvertChannel1	103
	ColorMinChannel2	104
	ColorMaxChannel2	105
	ColorInvertChannel2	106
	ColorMinChannel3	107
	ColorMaxChannel3	108
	ColorInvertChannel3	109
BLOB	GrayAbsoluteMin	101
	GrayAbsoluteMax	102
	GrayAbsoluteInvert	103

Set ROI (ASCII)

Set ROI (ASCII) request string to sensor (not available for RS232/RS422)		
Byte No.	Contents	Significance
1	S	Set ROI
2	R	
3	P T	Permanent Temporary Permanent change affect all parameters, including those that have only been changed temporarily.
4 - 11	X	ROI Info length in bytes from Byte 4 to end e.g. 00000055
12 - 14	X	Detector No. e.g. 001
15 - 16	X	ROI Index = 00 for yellow ROI = 01 for red ROI = 02 for position control
17 - 18	X	ROI shape 01=circle 02=rectangle 03=ellipse 04=free shape
19 - 26	X	Centre X (in pixels * 1000), e.g. 160 pixels = 00160000
27 - 34	X	Centre Y (in pixels * 1000), e.g. 120 pixels = 00120000
35 - 42	X	Half width / X-radius (in pixels * 1000), e.g. 80 Pixel = 00080000
43 - 50	X	Half height / Y-radius (in pixels * 1000), e.g. 40 Pixel = 00040000
51 - 58	X	Angle (not at circle / ellipse) (in ° * 1000),

		e.g. 180° = 00180000
Example:	SRP00000055001000200160000001200000008000000040000000180000 Length 55, Detector=1, yellow ROI, rectangle, center X=160, center Y=120, half width= 80, half height=40	
Set ROI (ASCII) response string from sensor		
Byte No.	Contents	Significance
1	S	Set ROI
2	R	
3	P T	Permanent Temporary
4	P F	P Pass F Fail
Example:	SRPP	
Additional information:		
Accepted in run mode:		Yes
Accepted in configuration mode:		No
Accepted when Ready Low:		Yes
Status of Ready signal during processing:		Low
End of telegram		Max. 4 byte (option)
Parameter		The parameter are given in alignment frames and not in image frames.

Get ROI (ASCII)

Get ROI (ASCII) request string to sensor (not available for RS232/RS422)		
Byte No.	Contents	Significance
1	G	Get ROI
2	R	
3	I	
4 - 6	X	Detector No. e.g. 001
7 - 8	X	ROI Index = 00 for yellow ROI = 01 for red ROI = 02 for position control
Example:	GRI00100	
Get ROI (ASCII) response String from sensor		
Byte No.	Contents	Significance
1	G	Get ROI
2	R	
3	I	
4	P F	P Pass F Fail
5 - 12	X	ROI Info length in bytes, from Byte 5 to end of string
13 - 15	X	Detector No. e.g. 001
16 - 17	X	ROI Index = 00 for yellow ROI = 01 for red ROI = 02 for position control
18 - 19	X	ROI shape 01=circle

		02=rectangle 03=ellipse 04=free shape
20 - 27	X	Centre X (value in pixels * 1000)
28 - 35	X	Centre Y (value in pixels * 1000)
36 - 43	X	Half width /X-radius (value in pixels * 1000)
44 - 51	X	Half height /Y-radius (value in pixels * 1000)
52 - 59	X	Angle (not at circle / ellipse) (valuein ° * 1000)
Example	GRIP0000005500100020016000000120000000080000000040000000090000 (Length= 55, detector 1, search region, rectangle, Centre X=160, Centre Y=120, half width =80, half height =40, angle = 90)	
Additional information:		
Accepted in run mode:		Yes
Accepted in configuration mode:		No
Accepted when Ready Low:		Yes
Status of Ready signal during processing:		Low
End of telegram		Max. 4 byte (option)

Get joblist (ASCII)

Get joblist (ASCII) request string to sensor		
Byte No.	Content	Significance
1	G	Get Job List
2	J	
3	L	
Example	GJL	
Get joblist (ASCII) response string from sensor		
Byte No.	Content	Significance
1	G	Get Job List
2	J	
3	L	
4	P F	P: Pass F: Fail
5 - 7	X	Response version
8 - 10	X	Number of jobs
11 - 13	X	Active job number
	Please note: The following byte sequence is repeated for each job from 1 to “Number of jobs”. The byte numbers shift accordingly.	
14 - 16	X	Number of characters for the job name. This can be used to specify a unique name for job n.
17 ... n	X	From this position, the name for job n follows in the specified length.
n+1 ... n + 3	X	Number of following bytes. A description for the job n can be specified.
n + 4 ... m	X	From this position, the description for job n follows in the specified length.

m + 1 ... m + 3	X	Number of following bytes. This can be used to specify a unique name for the author of job n.
m + 4 ... k	X	From this position, the name for the author of job n follows in the specified length.
k + 1 ... k + 8	X	Date of creation of job n.
k + 9 ... k + 16	X	Date of last change of job n.
Example	GJLP001001001007testjob010DefaultJob004Test2014112720141128	
Additional information:		
Accepted in Run mode		Yes
Accepted in Config mode		No
Accepted when Ready is Low		Yes
Status of Ready signal during processing		Not altered
End of telegram		Max. 4 byte (option)

Get detector list (ASCII)

Get detector list (ASCII) request string to sensor		
Byte No.	Content	Significance
1	G	Get detector dist
2	D	
3	L	
Example:	GDL	
Get detector list (ASCII) response string from sensor		
Byte No.	Content	Significance
1	G	Get detector list
2	D	
3	L	
4	P F	P: Pass F: Fail
5 - 7	X	Job number of the current job
8 - 10	X	Number of detectors in the current job
	Please note: The following byte sequence is repeated for each detector in the job. The byte numbers shift accordingly.	
11 - 13	X	Number of subsequent bytes. This allows a unique name for the detector n to be specified.
14 ... n	X	From this position, the name for detector n follows, in the given length.
n + 1 ... n+ 5	X	001 - Pattern matching 004 - Contour 005 - Gray 006 - Contrast 007 - Brightness 010 - Wafer 011 - OCR 013 - Datacode 014 - Barcode

		017 - Busbar 018 - Color value 019 - Color area 020 - Color list 021 - Caliper 022 - BLOB
Example	GDLP001001012testdetector00005	
Additional information:		
Accepted in Run mode		Yes
Accepted in Config mode		No
Accepted when Ready is Low		Yes
Status of Ready signal during processing		Not altered
End of telegram		Max. 4 byte (option)

Teach detector (ASCII)

Teach detector (ASCII) request string to sensor		
Byte No.	Content	Significance
1	T	Teach detector
2	E	
3	D	
4 - 6	X	0 = Alignment >= 1 Detectors
7	X	0 = Temporary 1 = Permanent Permanent change affect all parameters, including those that have only been changed temporarily.
8	X	Trigger 0 = no Trigger 1 = Trigger
Example:	TED00111	
Teach detector (ASCII) response String from sensor		
Byte No.	Content	Significance
1	T	Teach detector
2	E	
3	D	

4	P
	F
		P = Pass
		F = Fail

Example:	TEDP	
Additional information:		
Accepted in run mode:	Yes	
Accepted in configuration mode:	No	
Accepted when Ready	Yes	

Low:	
Status of Ready signal during processing:	Low
End of telegram	Max. 4 byte (option)

Calibration Add Point (ASCII)

Calibration Add Point (ASCII) request string to sensor		
Byte No.	Content	Significance
1	C	Calibration Add Point
2	A	
3	P	
4 - 8	0	Constant
9 - 16	X	World X (in mm *1000)
17 - 24	X	World Y (in mm *1000)
Example	CAP000000010000000200000 (Welt-X=100mm; Welt-Y=200mm)	
Calibration Add Point (ASCII) response string from sensor		
Byte No.	Content	Significance
1	C	Calibration Add Point
2	A	
3	P	
4	P F	P: Pass F: Fail
5 - 9	X	Current number of points in the list
10 - 17	X	Image X
18 - 25	X	Image Y
Example	CAPP000010064000000512000 (Index 1; Bild-X=640; Bild-Y=512)	
Additional information:		
Accepted in Run mode		Yes
Accepted in Config mode		No

Accepted when Ready is Low	Yes
Status of Ready signal during processing	Not altered
Supported Interfaces	UserApp
End of telegram	Max. 4 byte (option)
Necessary settings in requesting job	In “Out-put/Telegram/Payload” as first and second value the X- and Y- value of the finding position must be set.

Calibration: Clear point list (ASCII)

Calibration: Clear point list (ASCII) request string to sensor		
Byte No.	Content	Significance
1	C	Calibration Clear Data
2	C	
3	D	
Example	CCD	
Calibration: Clear point list (ASCII) response string from sensor		
Byte No.	Content	Significance
1	C	Calibration Clear Data
2	C	
3	D	
4	P F	P: Pass F: Fail
Example	CCDP	
Additional information:		
Accepted in Run mode		Yes
Accepted in Config mode		No
Accepted when Ready is Low		Yes
Status of Ready signal during processing		Not altered
Supported Interfaces		UserApp
End of telegram		Max. 4 byte (option)

Calibration point list (ASCII)

Calibration point list (ASCII) request string to sensor		
Byte No.	Content	Significance
1	C	Calibration point list
2	C	
3	L	
4	X	Permanency 0 = Temporary 1 = Permanent
Example	CCL1	
Calibration point list (ASCII) response string from sensor		
Byte No.	Content	Significance
1	C	Calibration point list
2	C	
3	L	
4	P F	P: Pass F: Fail
5 - 9	X	Current highest point pair index
10 - 17	X	RMSE (Root Mean Square Error)
18 - 25	X	Mean
26 - 33	X	Max
34 - 41	X	Min
Example	CCLP0001012345678123456781234567812345678	
Additional information:		
Accepted in Run mode		Yes

Accepted in Config mode	No
Accepted when Ready is Low	Yes
Status of Ready signal during processing	Not altered
Supported Interfaces	UserApp
End of telegram	Max. 4 byte (option)

Calibration calibration plate (ASCII)

Calibration calibration plate (ASCII) request string to sensor		
Byte No.	Content	Significance
1	C	Calibration calibrate plate
2	C	
3	P	
4	1	Request version
5	X	0 = Temporary 1 = Permanent Permanent change affect all parameters, including those that have only been changed temporarily.
6	X	0 - No fiducials are used. The origin of the world system is identical to the origin of the calibration plate. 1 - No fiducials are used. World system is identical to the image system. 2 - Use world system, fiducials job 3 - Use world system, fiducials request CAW. For more information on the origin of the World System, see: Calibration methods, location of the world system (Page 115)
7	X	0 - Calibration 1 - Validation
Examle	CCP1110	
Calibration calibration plate (ASCII) response string from sensor		
Byte No.	Content	Significance
1	C	Calibration Calibrate by Plate
2	C	
3	P	

4	P F	P: Pass F: Fail
5 - 7	X	Error code
8 - 12	X	Current number of detected calibration points.
13 - 20	X	RMSE (Root Mean Square Error)
21 - 28	X	Mean
29 - 36	X	Max
37 - 44	X	Min
45 - 52	X	Delta X (in mm *1000)
53 - 60	X	Delta Y (in mm *1000)
61 - 68	0	Reserved
69 - 76	0	Reserved
77 - 84	0	Reserved
85 - 92	X	Delta Gamma (in degrees *1000)
93 - 99	X	Fiducial, Deviation Mean
100 - 107	X	Fiducial, Deviation Max
108- 115	X	Fiducial, Deviation Min
Example	CCPP00000012000010010000200200003003000040040 00050050000600600007007000080080000900900001001	
Additional information:		
Accepted in Run mode		Yes
Accepted in Config mode		No
Accepted when Ready is Low		Yes
Status of Ready signal during processing		Not altered
End of telegram		Max. 4 byte (option)

Error code	Description
000	Success
001	Error
005	Invalid request
006	Input parameters with invalid size or invalid value
018	Calibration plate data not available
030	Calibration not enabled
033	Calibrate/Validate error
034	Invalid number of points
036	Invalid fiducial

Calibration Validate (ASCII)

Calibration validate (ASCII) request string to sensor		
Byte No.	Content	Significance
1	C	Calibration validate
2	V	
3	L	
Example	CVL	
Calibration validate (ASCII) response string from sensor		
Byte No.	Content	Significance
1	C	Calibration validate
2	V	
3	L	
4	P F	P: Pass F: Fail
5 - 9	X	Current highest point pair index
10 - 17	X	RMSE (Root Mean Square Error)
18 - 25	X	Mean
26 - 33	X	Max
34 - 41	X	Min
Example	CVLP0001012345678123456781234567812345678	
Additional information:		
Accepted in Run mode		Yes
Accepted in Config mode		No
Accepted when Ready is Low		Yes

Status of Ready signal during processing	Not altered
Supported Interfaces	UserApp
End of telegram	Max. 4 byte (option)

Copy Calibration (ASCII)

Copy Calibration (ASCII) request string to sensor		
Byte No.	Content	Significance
1	C	Calibration Copy Calibration
2	C	
3	C	
4	1	Request version
5	1	Constant
6 - 8	X	Destination 0 : To all Jobs from Jobset >0 : To specifed Job only
9	X	0: Always copy when the calibration is active. 1: Only copy if the calibration method is the same.
Example	CCC110021	
Copy Calibration (ASCII) response string from sensor		
Byte No.	Content	Significance
1	C	Calibration Copy Calibration
2	C	
3	C	
4	P F	P: Pass F: Fail
5 - 7	X	Error code
8 - 10	X	Job Number at which error occurred. 00: Success > 0 : Job number at which error occurred.
Example	CCCP000000	
Additional information:		

Accepted in Run mode	Yes
Accepted in Config mode	No
Accepted when Ready is Low	Yes
Status of Ready signal during processing	Not altered
End of telegram	Max. 4 byte (option)

Error code	**Description**
000	Success
001	Error
005	Invalid request
006	Input parameters with invalid size or invalid value
018	Calibration plate data not available
030	Calibration not enabled.
031	Calibration Copy error
032	Mismatched input conditions for destination Job

Get image (ASCII)

Get image (ASCII) Request string to sensor (not available for RS232/RS422)		
Byte No.	Contents	Significance
1	G	Get image
2	I	
3	M	
4	X	0 – Last Image 1 – Last Failed Image 2 – Last Good Image
Example:	GIM1	
Get image (ASCII) Response String from Sensor		
Byte No.	Contents	Significance
1	G	Get image
2	I	
3	M	
4	P F	P Pass F Fail
5	X	Error code
6	X	Image type 0 - Grayscale 3 – COLOR_BAYER_BG At conversion of the image from Bayer into RGB, the appropriate image type must be considered. Pre-processing filters of the category “arrangement” have an influence on the Bayer type. Bayer Pattern begins with blue - green.
7	X	Image result 1 - good image 0 - failed image
8 - 11	X	No of rows e.g. 0480 / 0200

12 - 15	X	No of columns e.g. 0640 / 0320
16 - 19	X	End of message string
20...n	X	Binary image data (rows * columns)
Example:	GIMP0004800640...	
Additional information:		
Accepted in run mode:		Yes
Accepted in configuration mode:		No
Accepted when Ready Low:		Yes
Status of Ready signal during processing:		Low
End of telegram		Max. 4 byte (option)

Error code	**Description**
0	Success
8	Recorder Off
9	No Matching Image of requested type

Get sensor identity V1 (ASCII)

Get sensor identity V1 (ASCII) request string to sensor (since version 1.19.)		
Byte No.	Content	Significance
1	G	Get sensor identity
2	S	
3	I	
4	1	Request version
Example:	GSI1	
Get sensor identity V1(ASCII) response string from sensor		
Byte No.	Content	Significance
1	G	Get sensor identity
2	S	
3	I	
4	P F	P: Pass F: Fail
5 - 7	X	Error code
8 - 10	X	Length of the following data
11 ... n	X	Version of the firmware as well as information about the hardware. Areas are clearly separated by a separator.
Example	GSIP0000221.19.3.2;V20-OB-AX-W12	
Additional information:		
Accepted in Run mode:	Yes	
Accepted in Config mode:	No	
Accepted when Ready is Low:	Yes	
End of telegram	Max. 4 byte (option)	

Error code	Description
000	Success
001	Error

Update firmware V1 (ASCII)

Update firmware V1 (ASCII) request string to sensor		
Byte No.	Content	Significance
1	U	Update firmware
2	F	
3	W	
4	1	Request version
Example:	UFW1	
Update firmware V1 (ASCII) response string from sensor		
Byte No.	Content	Significance
1	U	Update firmware
2	F	
3	W	
4	P F	P: Pass F: Fail
5 - 7	X	Error code
Example:	UFWP000	
Additional information:		
Accepted in run mode:		Yes
Accepted in configuration mode:		No
Accepted when ready low:		Yes
End of telegram:		Max. 4 byte (option)

Error code	**Description**
000	Success

Error code	Description
007	File does not exist
008	More than one vis file present
009	Sensor type does not match for VIS file
016	Firmware version mismatch

After sending the command, a valid firmware file is checked in the /tmp/ directory on the VISOR® vision sensor. The name must correspond to the typical name allocation (for example, as after the download from the SensoPart home page). The end is reached as soon as the camera signal is ready (pin 4 GN) again . Alternatively, the telegram “GSI1” can be used to check whether a valid response is sent.

Please note: The voltage supply must be ensured during the firmware update. An update may take up to 10 minutes.

Set jobset V1 (ASCII)

Set jobset V1 (ASCII) request string to sensor		
Byte No.	Content	Significance
1	S	Set jobset
2	J	
3	S	
4	1	Request version
5 - 7	X	Length of the subsequent file name. Maximum length 250 characters.
8 ... n	X	Optional file name. If no file name is specified, the default name "Jobset.bjs" is set.
Example	SJS1012myjobset.bjs	
Set jobset V1 (ASCII) response string from sensor		
Byte No.	Content	Significance
1	S	Set jobset
2	J	
3	S	
4	P F	P: Pass F: Fail
5 - 7	X	Error code
8 - 10	X	Active job number in the loaded jobset
Example	SJSP000001	
Additional information:		
Accepted in Run mode:	Yes	
Accepted in Config mode:	No	
Accepted when Ready is Low:	No	

Status of Ready signal during processing:	Low
End of telegram :	Max. 4 byte (option)

Error code	**Description**
000	Success
001	Error
007	File does not exist
010	Invalid name or length
011	Invalid data length
012	Not allowed due to jobset mismatch
013	Failed to start new Job from jobset

The jobset with the specified name is searched in the /tmp/ directory on the VISOR® vision sensor. If the file exists, this job record is activated. The file is then removed.

Get jobset V1 (ASCII)

Get jobset V1 (ASCII) request string to sensor		
Byte No.	Content	Significance
1	G	Get jobset
2	J	
3	S	
4	1	Request version
5 - 7	X	Length of the subsequent file name. Maximum length 250 characters.
8 ... n	X	Optional file name. If no file name is specified, the default name "Jobset.bjs" is set.
Example	GJS1012myjobset.bjs	
Get jobset V1 (ASCII) response string from sensor		
Byte No.	Content	Significance
1	G	Get jobset
2	J	
3	S	
4	P F	P: Pass F: Fail
5 - 7	X	Error code
Example	GJSP000	
Additional information:		
Accepted in Run mode		Yes
Accepted in Config mode		No
Accepted when Ready is Low		Yes
End of telegram		Max. 4 byte (option)

Error code	Description
000	Success
001	Error
007	File does not exist.
010	Invalid name or length
011	Invalid data length

The get jobset file is now available for download in the /tmp/ directory under the specified name.

Data output in ASCII

Output data (ASCII), dynamically composed from user settings in the software. For detailed informations to the file format see also:Telegram, Data output (Page 296)

Basic String Construction:

<START> (((<OPTIONAL FIELDS> <SEPARATOR> <PAYLOAD>))) <CHKSUM> <TRAILER>

Output data (ASCII):

<OPTIONAL FIELDS>			
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**
Header	1 - max. 8	User defined, max. 8 characters	Start string (Header)
Selected Fields	16	1 Byte per field	by this field output of all active checkboxes "byte-wise" can be activated - Output order is from left to right and from top to down. - For each checkbox there is one byte beginning with LSB = low significant bit. - Checkbox "Selected fields" is not part of the output! P = logical output set F = logical output not set 0 = logical output not active
Separator	1 - 5	User defined, max. 5 characters (per separator)	Separator from: "after first optional field", or "after first detector spec. date"
Data length	n	One byte per figure of decimal number	Length of telegram in bytes

<OPTIONAL FIELDS>			
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**
		e.g. 102 "1"; "0"; "2"	
Status	3	"110" triggered mode or "101" free-run mode	
Detector result	n	Byte 1 = AND conjunction of all detectors Byte 2 = Boolean result of alignment Byte 3 = global result of the active job Following Bytes: number of detectors Following Bytes: Detector results, "P" = Pass, "F" = Fail, last byte is first detector Following Bytes: Detector results, "P" = Pass, "F" = Fail, last byte is first detector Length: 4 Byte + 1 Byte per each used detector	
Digital outputs	n	First Bytes: number of active outputs Following Bytes: digital outputs	P = logical output set F = logical output not set 0 = logical output not active
Logical outputs	n	First Bytes: number of active logical outputs Following Bytes: logical outputs	Example: 18 logical outputs are configured, but only output1,2 and 9 are linked to functions (are active): 3PP000000P 2 bytes number of active outputs, all results bit-coded ... In this example there

<OPTIONAL FIELDS>			
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**
			are needed 2 bytes because of output 9.... P = logical output set F = logical output not set 0 = logical output not active
Total exec. time	n		Current (job) cycle time in [ms]
Active job no.	1-3		Active job no. (1..255)

<PAYLOAD>				
Detector specific				**For detector:**
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**	

Detector result	1	P = Pass
		F = Fail
			Boolean detector result	All detectors

Score value 1 ... n	n		Score (0..100%)	All detectors
Execution time	n		Execution time of individual detector in [msec].	All detectors
Distance	n		Calculated distance, [1/1000] *1)	Caliper
Position X 1 ... n	n	e.g.: X = 180 (pix) = (in ASCII) “180000” = 6 Byte	Position found X (x-coordinate). [1/1000] *1)	Pattern matching Contour Edge detector Caliper Datacode Barcode

<PAYLOAD>				
Detector specific				**For detector:**
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**	
				OCR
Position Y 1 ... n	n		Position found Y (y-coordinate). [1/1000] *1)	Pattern matching Contour Edge detector Caliper Datacode Barcode OCR
DeltaPos X	n		Delta position X between object taught and object found [1/1000] *1)	Pattern matching Contour Edge detector
DeltaPos Y	n		Delta position X between object taught and object found [1/1000] *1)	Pattern matching Contour Edge detector
Angle	n		Orientation of object found (0°..360°) [1/1000] *1)	Pattern matching Contour Edge detector Datacode Barcode OCR Wafer Busbar
Delta Angle	n		Angle between object taught and object found (0°..360°) [1/1000] *1)	Pattern matching Contour

<PAYLOAD>				
Detector specific				**For detector:**
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**	
				Edge detector
Scaling	n		Only with contour (0.5..2) [1/1000] *1)	Contour

R(ed)
G(reen)
B(lue)
	n		Value for color parameter, signed integer [1/1000] *1)	Color value
				Color list

H(ue)
S(aturation)
V(alue)
	n		Value for color parameter, signed integer [1/1000] *1)	Color value
				Color list

L(uminanz) A B	n		Value for color parameter, signed integer [1/1000] *1)	Color value Color list
Result index	n		Index in list, signed integer [1/1000] *1)	Color list
Color distance	n		Distance between taught and current color, signed integer [1/1000] *1)	Color list
Area	n		Area of the BLOB, without holes, in pixels, signed integer [1/1000] *1)	BLOB
Area (incl. holes)	n		Area of the BLOB, including holes, in pixels, signed integer [1/1000] *1)	BLOB
Contour length	n		Number of pixels of outer contour, signed integer [1/1000] *1)	BLOB
Compactness	n		Compactness of BLOB (Circle = 1, all other >1) The stronger the shape of the BLOB deviates from circle the larger the value of compactness will be. Signed integer [1/1000] *1)	BLOB
Center of gravity X	n		X- coordinate of center of gravity of BLOB, signed integer [1/1000] *1)	BLOB

<PAYLOAD>				
Detector specific				**For detector:**
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**	
Center of gravity Y	n		Y- coordinate of center of gravity of BLOB, signed integer [1/1000] *1)	BLOB
Center X	n		X- coordinate of fitted, geometric element (rectangle, ellipse), signed integer [1/1000] *1)	BLOB Wafer Busbar
Center Y	n		Y- coordinate of fitted, geometric element (rectangle, ellipse), signed integer [1/1000] *1)	BLOB Wafer Busbar
Width	n		Width of geometric element. Width >= 0, width >= height, negative value indicates failure, signed integer [1/1000] *1)	BLOB Wafer
Height	n		Height of geometric element. Heigth >= 0, height <= width, negative value indicates failure, signed integer [1/1000] *1)	BLOB Wafer
Angle (360)	n		Orientation of width of object in degree (range: -180 ... +180°, 0° = east, counterclockwise), signed integer [1/1000] *1)	BLOB Wafer Busbar
Eccentricity	n		Eccentricity numerical (range 0,0 ... 1,0), signed integer [1/1000] *1)	BLOB
Face up/-down, area	n		Face up/down discrimination, based on area, indicated by sign, signed integer [1/1000] *1)	BLOB
String	n	Maximum length 127!!	Contents of Code, depending from code string length may change, if a fix string length is needed, parameters minimum string length (detector specific data output) and maximum string length (detector parameters) have to be used.	Datacode Barcode OCR
String length	n		Length of Code in Bytes	Datacode Barcode OCR

<PAYLOAD>				
Detector specific				**For detector:**
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**	

Truncated	1	F = Code complete,
		P = Code truncated
			Code truncated	Datacode
				Barcode
				OCR

Compare result	1		Result of string comparison	Datacode Barcode OCR
Quality parameter	n		Output of quality parameters according to selection	Datacode Barcode
Contrast	n		Contrast of the code (0-100%)	Barcode
Correction	n		Number of modules corrected by error corrections	Barcode
Module height	n		Height of modules in pixels	Datacode
Module width	n		Width of modules in pixels	Datacode
Confidence	5 ... n		Output of the confidence values of the individual characters	OCR
Result	n		Degree of similarity between the read string and the reference string from 0 to 100%	OCR
Min. Quality	1		Minimum required quality was achieved	OCR
Length	n		Length of busbar	Busbar
Width	n		Width of busbar	Busbar

<CHKSUM>			
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**
Checksum	1		XOR checksum of all bytes in telegram

<TRAILER>			
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**
Trailer	1 - max. 8	User defined, max. 8 characters	End of string (Trailer)

[*1)] All detector-specific data with decimal places are transmitted as integers (multiplied by 1000) and must therefore be divided by 1000 after data reception.

8.1.3 Serial communication BINARY

Reset statistic (BINARY)

Reset statistic (Binary) request string to sensor			
Byte No.	Data Type	Content	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x05	
5	Unsigned Char	0x04	Reset statistic
Reset statistic (Binary) response string from sensor			
Byte No.	Data Type	Content	Significance
1	Unsined Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x07	
5	Unsigned Char	0x04	Reset statistic
6 - 7	Unsigned Short	0xXX	Error code (s. table below)
Additional information:			
Accepted in run mode:			Yes
Accepted in configuration mode:			No
Accepted when ready low:			Yes
Status of ready signal during processing:			Low

Error code	**Description**
0	Success
1	Error

Trigger (BINARY)

Trigger (Binary) request string to sensor			
Byte no.	Data type	Contents	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x05	
5	Unsigned Char	0x01	Trigger, (simple trigger without index, via port 2006)
Trigger (Binary) response string from sensor			
Byte no.	Data type	Contents	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x07	
5	Unsigned Char	0x01	Trigger, (response to trigger without index, via port 2006. If defined: result data without index via port 2005)
6 - 7	Unsigned Short	0xXX	Error code(s. table below)
Additional information:			
Accepted in run mode:			Yes
Accepted in configuration mode:			Yes
Accepted when Ready Low:			No
Status of Ready signal during processing:			Low

Error code	Description
0	Success
1	Error

Extended trigger (BINARY)

Extended trigger (Binary) request string to sensor			
Byte no.	Data type	Contents	Significance
1	Unsigned Int	0xXX	Length of telegram
2		0xXX	
3		0xXX	
4		0xXX	
5	Unsigned Char	0x13	Extended Trigger command, (trigger with index for correlation of trigger to the correponding result data, via port 2006)
6	Unsigned Char	0xXX	Length of following data (n)
7... n	Unsigned Char	0xXX	Data
Extended Trigger (Binary) Response string from sensor			
Byte no.	Data type	Contents	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x07	
5	Unsigned Char	0x013	Extended Trigger command, (response to trigger with index and result data, via port 2006, for correlation of trigger to corresponding result, Result data without index, via port 2005 also)
6 - 7	Unsigned Short	0x00	Error code (s. table below)
8	Unsigned Char	0xXX	Length of following data (n)
9 ... n	Unsigned Char	0xXX	Data of request command

n + 1	Unsigned Char	0xXX	Operating mode 0 = Config mode 1 = Run mode
n + 2 ... n + 5	Unsigned Int	0xXX	Length of following result data (m)
n + 6 ... m	Unsigned Int	0xXX	Result data
Additional information:			
Accepted in run mode:			Yes
Accepted in configuration mode:			Yes
Accepted when Ready Low:			No
Status of Ready signal during processing:			Low

Error code	**Description**
0	Success
1	Error

Change job (BINARY)

Change job (Binary) request string to sensor			
Byte no.	Data type	Contents	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x06	
5	Unsigned Char	0x02	Change job
6	Unsigned Char	0xXX	Job no, XX = 1- n
Change job (Binary) response string from sensor			
Byte no.	Data type	Contents	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x09	
5	Unsigned Char	0x02	Change job
6 - 7	Unsigned Short	0xXX	Error code (s. table below)
8	Unsigned Char	0xXX	Trigger mode 0 = triggered 1 = free-run
9	Unsigned Char	0xXX	Job no, XX = 1- n
Additional information:			
Accepted in run mode:		Yes	
Accepted in configuration mode:		No	
Accepted when ready low:		Yes	
Status of ready signal during processing:		Low	

Error code	Description
0	Success
1	Error

Change job permanent (BINARY)

Change job permanent (Binary) request string to sensor			
Byte no.	Data type	Contents	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x06	
5	Unsigned Char	0x02	Change job permanent
6	Unsigned Char	0xXX	Job no, XX = 1- n
Change job permanent(Binary) response string from sensor			
Byte no.	Data type	Contents	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x09	
5	Unsigned Char	0x02	Change job permanent
6 - 7	Unsigned Short	0xXX	Error code (s. table below)
8	Unsigned Char	0xXX	Trigger mode 0 = triggered 1 = free-run
9	Unsigned Char	0xXX	Job no, XX = 1- n
Additional information:			
Accepted in run mode:		Yes	
Accepted in configuration mode:		No	
Accepted when ready low:		Yes	
Status of ready signal during processing:		Low	

Error code	Description
0	Success
1	Error

Set shutter speed (BINARY)

Set shutter speed (Binary) request string to sensor			
Byte No.	Data Type	Contents	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x09	
5	Unsigned Char	0x0E 0x0F	Set shutter speed temporary Set shutter speedpermanent Permanent change affect all parameters, including those that have only been changed temporarily.
6 - 9	Unsigned Int	0xXX	Shutter value (in microseconds)
Set shutter speed (Binary) response string from sensor			
Byte No.	Data Type	Contents	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x07	
5	Unsigned Char	0x0E 0x0F	Set shutter speed temporary Set shutter speed permanent
6 - 7	Unsigned Short	0xXX	Error code (s. table below)
Additional information:			
Accepted in run mode:			Yes
Accepted in configuration mode:			No
Accepted when Ready Low:			Yes

Status of Ready signal during processing:	Low

Error code	Description
0	Success
1	Error

Get shutter speed (BINARY)

Get shutter speed (Binary) request string to sensor (since version 1.6.5.3)			
Byte No.	Data type	Contents	Significance
1	Unsigned int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x05	
5	Unsigned Char	0x17	Get shutter speed
Get shutter speed (Binary) response string from sensor			
1	Unsigned int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x0B	
5	Unsigned Char	0x17	Get shutter speed
6 - 7	Unsigned short	0xXX	Error code (s. table below)
8 - 11	Unsigned int	0xXX	Shutter value
Additional information:			
Accepted in run mode:			Yes
Accepted in configuration mode:			No
Accepted when Ready Low:			Yes
Status of Ready signal during processing:			Not altered

Error code	**Description**
0	Success

Error code	Description
1	Error

Set gain (BINARY)

Set gain (Binary) request string to sensor			
Byte No.	DataType	Content	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x0A	
5	Unsigned Char	0x1B	Set gain
6	Unsigned Char	0xXX	1 = Permanent 0 = Temporary Permanent change affect all parameters, including those that have only been changed temporarily.
7 - 10	Unsigned Int	0xXX	Gain value
Set gain (Binary) response string from sensor			
Byte No.	DataType	Content	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x0B	
5	Unsigned Char	0x1B	Set gain
6 - 7	Unsigned Short	0xXX	Error code (s. table below)
8 - 11	Unsigned Int	0xXX	Current gain value (value *1000)
Additional information:			
Accepted in run mode:			Yes
Accepted in configuration mode:			No
Accepted when ready low:			Yes

Status of ready signal during processing:	Not altered

Error code	Description
0	Success
1	Error

Get gain (BINARY)

Get gain (Binary) request string to sensor			
Byte No.	Data Type	Content	Significance
1	Unsigned Int.	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x05	
5	Unsigned Char	0x1C	Get gain
Get gain (Binary) response string from sensor			
Byte No.	Data Type	Content	Significance
1	Unsigned Int.	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x0B	
5	Unsigned Char	0x1C	Get gain
6	Unsigned Short	0xXX	Error ode (s. table below)
7		0xXX	
8 - 11	Unsigned Int.	0xXX	Current gain (in value * 1000), e.g. 1,0 = 01000
Additional information:			
Accepted in run mode:		Yes	
Accepted in configuration mode:		No	
Accepted when Ready Low:		Yes	
Status of Ready signal during processing:		Not altered	

Error code	Description
0	Success
1	Error

Set parameter (BINARY)

Set parameter (Binary) request string to sensor			
Byte no.	Data type	Contents	Significance
1	Unsigned Int	0xXX	Length of telegram = 9 Bytes + length of string (n)
2		0xXX	
3		0xXX	
4		0xXX	
5	Unsigned Char	0x05 0x06	Set parameter permanent Set parameter temporary Permanent change affect all parameters, including those that have only been changed temporarily.
6	Unsigned Char	0xXX	Detector no., XX = 1- n
7	Unsigned Char	0xXX	Command: Set reference string 7 value *1), see below !
8 - 9	Unsigned Short	0xXX	Length new reference string / value (n)
10..n	Unsigned Char	0xXX	Reference string / value
Set parameter (Binary) Response string from Sensor (may be 4-5 seconds delayed)			
Byte no.	Data type	Contents	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x08	
5	Unsigned Char	0x05 0x06	ID set reference string permanent ID set reference string temporary
6 - 7	Unsigned Short	0xXX	Error code (s. table below)

8	Unsigned Char	0x0A	Parameter type string

Additional information:	
Accepted in run mode:	Yes
Accepted in configuration mode:	No
Accepted when Ready Low:	Yes
Status of Ready signal during processing:	Low

Error code	Description
0	Success
1	Error

*1) Byte No. 7: Command: set reference string / value:

Detector	Function	Command	Length of following data
Alignment Pattern matching	Threshold Min Threshold Max ResultOffsetPos_X ResultOffsetPos_Y ResultOffsetAngle	1 2 31 32 33	4 4 5 5 5
Alignment Contour	Threshold Min Threshold Max ResultOffsetPos_X ResultOffsetPos_Y ResultOffsetAngle	1 2 31 32 33	4 4 5 5 5
Alignment Edge	Threshold Min Threshold Max Transition_Horizontal Transition_Vertical	1 2 101 102	4 4 4 4
Pattern matching	Threshold Min Threshold Max ResultOffsetPos_X ResultOffsetPos_Y ResultOffsetAngle	1 2 31 32 33	4 4 5 5 5

Detector	Function	Command	Length of following data
Contour	Threshold Min Threshold Max ResultOffsetPos_X ResultOffsetPos_Y ResultOffsetAngle	1 2 31 32 33	4 4 5 5 5
Gray Level	Threshold Min Threshold Max GrayMin GrayMax	1 2 101 102	4 4 4 4
Contrast	Threshold Min Threshold Max	1 2	4 4
Barcode	Reference String	101	n
Datacode	Reference String	101	n
OCR	Reference String	101	n
ColorValue	ColorMinChannel1 ColorMaxChannel1 ColorInvertChannel1 ColorMinChannel2 ColorMaxChannel2 ColorInvertChannel2 ColorMinChannel3 ColorMaxChannel3 ColorInvertChannel3	101 102 103 104 105 106 107 108 109	4 4 4 4 4 4 4 4 4
ColorArea	ColorMinChannel1 ColorMaxChannel1 ColorInvertChannel1 ColorMinChannel2 ColorMaxChannel2 ColorInvertChannel2 ColorMinChannel3 ColorMaxChannel3 ColorInvertChannel3	101 102 103 104 105 106 107 108 109	4 4 4 4 4 4 4 4 4
BLOB	GrayAbsoluteMin GrayAbsoluteMax GrayAbsoluteInvert	101 102 103	4 4 1

Get parameter (BINARY)

Get parameter (Binary) request string to sensor			
Byte no.	Data type	Contents	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x07	
5	Unsigned Char	0x0A	Get parameter
6	Unsigned Char	0xXX	Detector no., XX = 1- n
7	Unsigned Char	0xXX	Command: Set reference string / value *1), see below !
Get Parameter (Binary) response string from Sensor (may be 4-5 Seconds delayed)			
Byte no.	Data type	Contents	Significance
1	Unsigned Int	0xXX	Length of telegram = 10 Bytes + Length of string (n)
2		0xXX	
3		0xXX	
4		0xXX	
5	Unsigned Char	0x0A	Get parameter
6 - 7	Unsigned Short	0xXX	Error code (s. table below)
8	Unsigned Char	0xXX	Parameter type string
9 - 10	Unsigned Short	0xXX	Length of parameter (n)
11... n + n	Unsigned Char	0xXX	Reference string / value
Additional information:			
Accepted in run mode:			Yes
Accepted in configuration mode:			No

Accepted when Ready Low:	Yes
Status of Ready signal during processing:	Not altered

Error code	**Description**
0	Success
1	Error

*1) Byte No. 7: Command: Get reference string / value:

Detector	**Function**	**Command**	**Length of following data**
Alignment Pattern matching	Threshold Min Threshold Max ResultOffsetPos_X ResultOffsetPos_Y ResultOffsetAngle	1 2 31 32 33	4 4 5 5 5
Alignment Contour	Threshold Min Threshold Max ResultOffsetPos_X ResultOffsetPos_Y ResultOffsetAngle	1 2 31 32 33	4 4 5 5 5
Alignment Edge	Threshold Min Threshold Max Transition_Horizontal Transition_Vertical	1 2 101 102	4 4 4 4
Pattern matching	Threshold Min Threshold Max ResultOffsetPos_X ResultOffsetPos_Y ResultOffsetAngle	1 2 31 32 33	4 4 5 5 5
Contour	Threshold Min Threshold Max ResultOffsetPos_X ResultOffsetPos_Y ResultOffsetAngle	1 2 31 32 33	4 4 5 5 5
Gray Level	Threshold Min Threshold Max GrayMin	1 2 101	4 4 4

Detector	Function	Command	Length of following data
	GrayMax	102	4
Contrast	Threshold Min Threshold Max	1 2	4 4
Barcode	Reference String	101	n
Datacode	Reference String	101	n
OCR	Reference String	101	n
ColorValue	ColorMinChannel1 ColorMaxChannel1 ColorInvertChannel1 ColorMinChannel2 ColorMaxChannel2 ColorInvertChannel2 ColorMinChannel3 ColorMaxChannel3 ColorInvertChannel3	101 102 103 104 105 106 107 108 109	4 4 4 4 4 4 4 4 4
ColorArea	ColorMinChannel1 ColorMaxChannel1 ColorInvertChannel1 ColorMinChannel2 ColorMaxChannel2 ColorInvertChannel2 ColorMinChannel3 ColorMaxChannel3 ColorInvertChannel3	101 102 103 104 105 106 107 108 109	4 4 4 4 4 4 4 4 4
BLOB	GrayAbsoluteMin GrayAbsoluteMax GrayAbsoluteInvert	101 102 103	4 4 1

Set ROI (BINARY)

Set ROI (Binary) request string to sensor (not available with RS232/RS422)			
Byte No.	Data Type	Contents	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x20	
5	Unsigned Char	0x10 0x11	Set ROI temporary Set ROI permanent Permanent change affect all parameters, including those that have only been changed temporarily.
6 - 9	Unsigned Int	0xXX	ROI Info Length in Bytes from Byte 6 to end
10	Unsigned Char	0xXX	Detector No.
11	Unsigned Char	0x00	ROI Index 00 = yellow ROI 01 = Teach 02 = Position control
12	Unsigned Char	0xXX	ROI shape 01=circle 02=rectangle 03=ellipse 04=free shape
13 -16	Unsigned Int	0xXX	ROI Parameter: center X (in Pixels * 1000)
17 - 20	Unsigned Int	0xXX	ROI Parameter: center Y (in Pixels * 1000)
21 - 24	Unsigned Int	0xXX	ROI Parameter: width / radius X (in Pixels* 1000)
	Only ellipse / rectangle:		
25 - 28	Unsigned Int	0xXX	ROI Parameter: width / radius Y (in Pixels* 1000)
29 - 32	Unsigned Int	0xXX	ROI Parameter: Angle in ° (in ° * 1000)

Set ROI (Binary) Response String from Sensor			
Byte No.	Data Type	Contents	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x07	
5	Unsigned Char	0x10 0x11	Set ROI temporary Set ROI permanent
6 - 7	Unsigned Short	0xXX	Error code (s. table below)
Additional information:			
Accepted in run mode:			Yes
Accepted in configuration mode:			No
Accepted when ready low:			Yes
Status of ready signal during processing:			Low
Parameter:			The parameter are given in alignment frames and not in image frames.

Error code	Description
0	Success
1	Error

Get ROI (BINARY)

Get ROI (Binary) request string to sensor (not available with RS232/RS422)			
Byte No.	Data Type	Content	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x07	
5	Unsigned Char	0x12	Get ROI
6	Unsigned Char	0xXX	Detector No.
7	Unsigned Char	0xXX	ROI Index 00 = yellow ROI 01 = Teach 02 = Position control
Get ROI (Binary) Response String from Sensor			
Byte No.	Data Type	Contents	Significance
1	Unsigned Int	0xXX	Length of telegram
2		0xXX	
3		0xXX	
4		0xXX	
5	Unsigned Char	0x12	Get ROI
6 - 7	Unsigned Short	0xXX	Error code (s. table below)
8 - 11	Unsigned Int	0xXX	ROI Info Length in Bytes from Byte 8 to end
12	Unsigned Char	0xXX	Detector No.
13	Unsigned Char	0x00	ROI Index 00 = yellow ROI 01 = Teach 02 = Position control
14	Unsigned Char	0xXX	ROI shape 01=circle

			02=rectangle 03=ellipse 04=free shape
15 - 18	Unsigned Int	0xXX	ROI Parameter: center X (in Pixels * 1000)
19 - 22	Unsigned Int	0xXX	ROI Parameter: center Y (in Pixels * 1000)
23 -26	Unsigned Int	0xXX	ROI Parameter: width / radius X (in Pixels* 1000)
	Only ellipse / rectangle:		
27 - 30	Unsigned Int	0xXX	ROI Parameter: width / radius Y (in Pixels* 1000)
31 - 34	Unsigned Int	0xXX	ROI Parameter: Angle in ° (in ° * 1000)
Additional information:			
Accepted in run mode:			Yes
Accepted in configuration mode:			No
Accepted when Ready Low:			Yes
Status of Ready signal during processing:			Low

Error code	**Description**
0	Success
1	Error

Get job list (BINARY)

Get job list (Binary) request string to sensor			
Byte No.	Data Type	Content	Significance
1	Unsigned Int.	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x05	
5	Unsigned Char	0x14	Get job list
Get job list (Binary) response string from sensor			
Byte No.	Data Type	Content	Significance
1	Unsigned Int.	0xXX	Length of telegram
2		0xXX	
3		0xXX	
4		0xXX	
5	Unsigned Char	0x14	Get job list
6	Unsigned Short	0xXX	Error code (s. table below)
8	Unsigned Char	0x01	Constant
9	Unsigned Char	0xXX	Number of jobs
10	Unsigned Char	0xXX	Active job number
	Please note: The following byte sequence is repeated for each job from 1 to "Number of jobs". The byte numbers shift accordingly.		
11	Unsigned Char	0xXX	Number of characters for the job name. This can be used to specify a unique name for job n.
11 ... n	Char	0xXX	From this position, the name for job n follows in the specified length.

n + 1 ... n + 3	Unsigned Char	0xXX	Number of following bytes. A description for the job n can be specified.
n + 4 ... m	Char	0xXX	From this position, the description for job n follows in the specified length.
m + 1 ... m+ 3	Unsigned Char	0xXX	Number of following bytes. This can be used to specify a unique name for the author of job n.
m + 4 ... k	Char	0xXX	From this position, the name for the author of job n follows in the specified length.
k + 1 ... k + 8	Unsigned Int.	0xXX	Date of creation of job n.
k + 9 ... k + 16	Unsigned Int.	0xXX	Date of last change of job n.
Additional information:			
Accepted in run mode:		Yes	
Accepted in configuration mode:		No	
Accepted when ready low:		Yes	
Status of ready signal during processing:		Not altered	

Error code	**Description**
0	Success
1	Error

Get detector list (BINARY)

Get detector list (Binary) request string to sensor			
Byte No.	Data Type	Content	Significance
1	Unsigned Int.	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x05	
5	Unsigned Char	0x15	Get Detector List
Get detector list (Binary) response string from sensor			
Byte No.	Data Type	Content	Significance
1	Unsigned Int.	0xXX	Length of telegram
2		0xXX	
3		0xXX	
4		0xXX	
5	Unsigned Char	0x18	Get Detector List
6	Unsigned Short	0xXX	Error code
8	Unsigned Char	0xXX	Job number of the current job
9	Unsigned Char	0xXX	Number of detectors in the current job
	Please note: The following byte sequence is repeated for each detector in the job. The byte numbers shift accordingly.		
10	Unsigned Char	0xXX	Number of subsequent bytes. This allows a unique name for the detector n to be specified.
11 ... n	Char	0xXX	From this position, the name for detector n follows, in the given length.
n + 1 ... n + 3	Unsigned Char	0xXX	001 - Pattern matching 004 - Contour 005 - Gray 006 - Contrast

			007 - Brightness 010 - Wafer 011 - OCR 013 - Datacode 014 - Barcode 017 - Busbar 018 - Color value 019 - Color area 020 - Color list 021 - Caliper 022 - BLOB
Additional information:			
Accepted in run mode:		Yes	
Accepted in configuration mode:		No	
Accepted when Ready Low:		Yes	
Status of Ready signal during processing:		Not altered	

Error code	**Description**
0	Success
1	Error

Teach detector (BINARY)

Teach detector (Binary) request string to sensor			
Byte No.	Data Type	Content	Significance
1	Unsigned Int.	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x08	
5	Unsigned Char	0x18	Teach detector
6	Unsigned Char	0xXX	0 = Alignment >= 1 Detectors
7	Unsigned Char	0xXX	0 = Temporary 1 = Permanent
8	Unsigned Char	0xXX	0 = No Trigger 1 = Trigger
Teach detector (Binary) response string from sensor			
Byte No.	Data Type	Content	Significance
1	Unsigned Int.	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x07	
5	Unsigned Char	0x18	Teach detector
6 - 7	Unsigned Short	0xXX	Error code (s. table below)
Additional information:			
Accepted in run mode:			Yes
Accepted in configuration mode:			No
Accepted when ready low:			Yes
Status Ready signal during processing:			No altered

Error code	Description
0	Success
1	Error

Calibration: Add point (BINARY)

Calibration: Add point (Binary) request string to sensor			
Byte No.	DataType	Content	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x0F	
5	Unsigned Char	0x1D	Calibration Add Point (CAP)
6 - 7	Unsigned Short	0x00	Constant
8 - 11	Unsigned Int	0xXX	World X (in mm *1000)
12 - 15	Unsigned Int	0xXX	World Y(in mm *1000)
Calibration: Add point (Binary) response string from sensor			
Byte No.	Data Type	Content	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x11	
5	Unsigned Char	0x1D	Calibration Add Point (CAP)
6 - 7	Unsigned Short	0xXX	Error code (s. table below)
8 - 9	Unsigned Short	0xXX	Current number of points in the list
10 -13	Unsigned Int	0xXX	Image X (in px * 1000)
14 - 17	Unsigned Int	0xXX	Image Y (in px * 1000)
Additional information:			
Accepted in run mode:			Yes

Accepted in configuration mode:	No
Accepted when ready low:	Yes
Status of ready signal during processing:	Not altered
Supported interfaces:	UserApp, PROFINET
Necessary settings in requesting job:	In “Output/Telegram/Payload” as first and second value the X- and Y- value of the finding position must be set.

Error code	**Description**
0	Success
1	Error

Calibration clear point list (BINARY)

Calibration clear point list (Binary) request string to sensor			
Byte No.	DataType	Content	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x05	
5	Unsigned Char	0x1F	Calibration: Clear point list
Calibration clear point list (Binary) response string from sensor			
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x07	
5	Unsigned Char	0x1F	Calibration: Clear point list
6 - 7	Unsigned Short	0xXX	Error code (s. table below)
Additional information:			
Accepted in run mode:			Yes
Accepted in configuration mode:			No
Accepted when Ready Low:			Yes
Status of Ready signal during processing:			Not altered
Supported interfaces			UserApp, PROFINET

Error code	Description
0	Success
1	Error

Calibration point list (BINARY)

Calibration point list (Binary) request string to sensor			
Byte No.	DataType	Content	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x06	
5	Unsigned Char	0x1E	Calibration point list
6	Unsigned Char	0xXX	Permanency 0 = Temporary 1 = Permanent
Calibration point list (Binary) response string from sensor			
Byte No.	DataType	Content	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x19	
5	Unsigned Char	0x1E	Calibration point list
6 - 7	Unsigned Short	0xXX	Error code
8 - 9	Unsigned Short	0xXX	Current highest point pair index
10 - 13	Unsigned Int	0xXX	RMSE (Root Mean Square Error)
14 - 17	Unsigned Int	0xXX	Mean
18 - 21	Unsigned Int	0xXX	Max
22 - 25	Unsigned Int	0xXX	Min
Additional information:			

Accepted in run mode:	Yes
Accepted in configuration mode:	No
Accepted when Ready Low:	Yes
Status of Ready signal during processing:	Not altered

Error code	Description
0	Success
1	Error

Calibration calibration plate (BINARY)

Calibration calibration plate (Binary) request string to sensor			
Byte No.	Data Type	Content	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x09	
5	Unsigned Char	0x24	Calibration calibration plate
6	Unsigned Char	0x01	Request version
7	Unsigned Char	0xXX	0 = Temporary 1 = Permanent
8	Unsigned Char	0xXX	0 - No fiducials are used. The origin of the world system is identical to the origin of the calibration plate. 1 - No fiducials are used. World system is identical to the image system. 2 - Use world system, fiducials job 3 - Use world system, fiducials request CAW. For more information on the origin of the World System, see: Calibration methods, location of the world system (Page 115)
9	Unsigned Char	0xXX	0 - Calibration 1 - Validation
Calibration calibration plate (Binary) response string from sensor			
Byte No.	Data Type	Content	Significance

1	Unsined Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x3D	
5	Unsigned Char	0x24	Calibration calibration plate
6 - 7	Unsigned Short	0xXX	Error code (s. table below)
8 - 9	Unsigned Short	0xXX	Current number of detected calibration points.
10 - 13	Unsigned Int	0xXX	RMSE (Root Mean Square Error)
14 - 17	Unsigned Int	0xXX	Mean
18 - 21	Unsigned Int	0xXX	Max
22 - 25	Unsigned Int	0xXX	Min
26 - 29	Unsigned Int	0xXX	Delta X (in mm *1000)
30 - 33	Unsigned Int	0xXX	Delta Y (in mm *1000)
34 - 37	Unsigned Int	0	Reserved
38 - 41	Unsigned Int	0	Reserved
42 - 45	Unsigned Int	0	Reserved
46 - 49	Unsigned Int	0xXX	Delta Gamma (in degrees *1000)
50 - 53	Unsigned Int	0xXX	Fiducial, Deviation Mean
54 - 57	Unsigned Int	0xXX	Fiducial, Deviation Max
58 - 61	Unsigned Int	0xXX	Fiducial, Deviation Min

Additional information:	
Accepted in run mode:	Yes
Accepted in configuration mode:	No
Accepted when Ready Low:	Yes
Status of Ready signal during processing:	Not altered

Error code	**Description**
00	Success
01	Error
05	Invalid request
06	Input parameters with invalid size or invalid value
18	Calibration plate data not available
30	Calibration not enabled
33	Calibrate/Validate error
34	Invalid number of points
36	Invalid fiducial

Calibration validate (BINARY)

Calibration validate (Binary) request string to sensor			
Byte No.	DataType	Content	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x05	
5	Unsigned Char	0x20	Calibration validate
Calibration validate (Binary) response string from sensor			
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x19	
5	Unsigned Char	0x20	Calibration validate
6 - 7	Unsigned Short	0xXX	Error code (s. table below)
8 - 9	Unsigned Short	0xXX	Current highest point pair index
10 - 13	Unsigned Int	0xXX	RMSE (Root Mean Square Error)
14 - 17	Unsigned Int	0xXX	Mean
18 - 21	Unsigned Int	0xXX	Max
22 - 25	Unsigned Int	0xXX	Min
Additional information:			
Accepted in run mode:			Yes
Accepted in configuration mode:			No
Accepted when Ready Low:			Yes

Status of Ready signal during processing:	Not altered

Error code	Description
0	Success
1	Error
3	Insufficient parameter data

Copy calibration (BINARY)

Copy calibration (Binary) request string to sensor			
Byte No.	DataType	Content	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x09	
5	Unsigned Char	0x25	Copy calibration
6	Unsigned Char	0x01	Request version
7	Unsigned Char	0x01	Constant
8	Unsigned Char	0xXX	Destination 0 : To all Jobs from Jobset >0 : To specifed Job only
9	Unsigned Char	0xXX	0: Always copy when the calibration is active. 1: Only copy if the calibration method is the same.
Copy calibration (Binary) response string from sensor			
Byte No.	DataType	Content	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x08	
5	Unsigned Char	0x25	Copy calibration
6 - 7	Unsigned Short	0xXX	Error code (s. table below)
8	Unsigned Char	0xXX	00: Success > 0 : Job number at which error occurred.
Additional information:			
Accepted in run mode:			Yes

Accepted in configuration mode:	No
Accepted when ready low:	Yes
Status of ready signal during processing:	Not altered

Error code	**Description**
0	Success
1	Error
3	Insufficient parameter data
5	Invalid request
6	Input parameters with invalid size or invalid value
18	Calibration plate data not available
30	Calibration not enabled.
31	Calibration Copy error
32	Mismatched input conditions for destination Job

Get image (BINARY)

Get image (Binary) request string to sensor (not available with RS232/RS422)			
Byte No.	Data type	Contents	Significance
1	Unsigned Int	0x00	Length of telegram
2		0x00	
3		0x00	
4		0x06	
5	Unsigned Char	0x03	Get image
6	Unsigned Char	0xXX	0 – Last Image 1 – Last Failed Image 2 – Last Good Image
Get image (Binary) response string from sensor			
Byte No.	Data type	Contents	Significance
1	Unsigned Int	0xXX	Length of telegram e.g. 00 04 B0 0D (Dez. 307213)
2		0xXX	
3		0xXX	
4		0xXX	
5	Unsigned Char	0x03	Response ID Get image
6 - 7	Unsigned short	0xXX	Error code (s. table below)
8	Unsigned Char	0xXX	Image type 0 - Grayscale 3 – COLOR_BAYER_BG At conversion of the image from Bayer into RGB, the appropriate image type must be considered.
9	Unsigned Char	0xXX	Image result 01 - Good image 00 - Failed image
10 - 11	Unsigned	0xXX	No. of rows

	short		e.g. 01 E0 = 480
12 - 13	Unsigned short	0xXX	No. of columns e.g. 02 80 = 640
14 ... n	Unsigned Char	0xXX	Binary image data (rows * columns)
Additional information:			
Accepted in run mode:			Yes
Accepted in configuration mode:			No
Accepted when ready low:			Yes
Status of ready signal during processing:			Low

Error code	Description
00	Success
01	Error
08	Recorder Off
09	No Matching Image of requested type

Data output in BINARY

Output data (BINARY), dynamically composed from user settings in the software. For detailed informations to the file format see also:Telegram, Data output (Page 296)

Basic String Construction:

<START> (((<OPTIONAL FIELDS> <SEPARATOR> <PAYLOAD>))) <CHKSUM> <TRAILER>

Output data (BINARY):

<OPTIONAL FIELDS>			
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**
Header	1 - max. 8	User defined, max. 8 characters	Start string (Header)
Selected Fields	16	1 Byte per field	by this field output of all active checkboxes “byte-wise” can be activated - Output order is from left to right and from top to down. - For each checkbox there is one byte beginning with LSB = low significant bit. - Checkbox “Selected fields” is not part of the output! P = logical output set F = logical output not set 0 = logical output not active
Data length	n	One byte per figure of decimal number e.g. 102 “1”; “0”; “2”	Length of telegram in bytes
Status	3	“110” triggered mode or “101” free-run mode	
Detector result	n	Byte 1 = AND conjunction of all	

<OPTIONAL FIELDS>			
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**
		detectors Byte 2 = Boolean result of alignment Byte 3 = global result of the active job Following Bytes: number of detectors Following Bytes: Detector results, “P” = Pass, “F” = Fail, last byte is first detector Following Bytes: Detector results, “P” = Pass, “F” = Fail, last byte is first detector Length: 4 Byte + 1 Byte per each used detector	
Digital outputs	n	First Bytes: number of active outputs Following Bytes: digital outputs	P = logical output set F = logical output not set 0 = logical output not active
Logical outputs	n	First Bytes: number of active logical outputs Following Bytes: logical outputs	Example: 18 logical outputs are configured, but only output1,2 and 9 are linked to functions (are active): 3PP000000P 2 bytes number of active outputs, all results bit-coded ... In this example there are needed 2 bytes because of output 9.... P = logical output set F = logical output not set

<OPTIONAL FIELDS>			
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**
			0 = logical output not active
Total exec. time	n		Current (job) cycle time in [ms]
Active job no.	1-3		Active job no. (1..255)

<PAYLOAD>				
Detector specific				**For detector:**
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**	

Detector result	1	P = Pass
		F = Fail
			Boolean detector result	All detectors

Score value 1 ... n	1-3		Score (0..100%)	All detectors
Execution time	n		Execution time of individual detector in [msec].	All detectors
Distance	n		Calculated distance, [1/1000] *1)	Caliper
Position X 1 ... n	n	e.g.: X = 180 (pix) = (in ASCII) “180000” = 6 Byte	Position found X (x-coordinate). [1/1000] *1)	Pattern matching Contour Edge detector Caliper Datacode Barcode OCR
Position Y 1 ... n	n		Position found Y (y-coordinate). [1/1000] *1)	Pattern matching Contour Edge

<PAYLOAD>				
Detector specific				**For detector:**
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**	
				detector Caliper Datacode Barcode OCR
DeltaPos X	n		Delta position X between object taught and object found [1/1000] *1)	Pattern matching Contour Edge detector
DeltaPos Y	n		Delta position X between object taught and object found [1/1000] *1)	Pattern matching Contour Edge detector
Angle	n		Orientation of object found (0°..360°) [1/1000] *1)	Pattern matching Contour Edge detector Datacode Barcode OCR Wafer Busbar
Delta Angle	n		Angle between object taught and object found (0°..360°) [1/1000] *1)	Pattern matching Contour Edge detector
Scaling	n		Only with contour (0.5..2) [1/1000] *1)	Contour
R(ed)	n		Value for color parameter, signed integer [1/1000] *1)	Color

<PAYLOAD>				
Detector specific				**For detector:**
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**	

G(reen)
B(lue)
				value
				Color list

H(ue)
S(aturation)
V(alue)
	n		Value for color parameter, signed integer [1/1000] *1)	Color value
				Color list

L(uminanz) A B	n		Value for color parameter, signed integer [1/1000] *1)	Color value Color list
Result index	n		Index in list, signed integer [1/1000] *1)	Color list
Color distance	n		Distance between taught and current color, signed integer [1/1000] *1)	Color list
Area	n		Area of the BLOB, without holes, in pixels, signed integer [1/1000] *1)	BLOB
Area (incl. holes)	n		Area of the BLOB, including holes, in pixels, signed integer [1/1000] *1)	BLOB
Contour length	n		Number of pixels of outer contour, signed integer [1/1000] *1)	BLOB
Compactness	n		Compactness of BLOB (Circle = 1, all other >1) The stronger the shape of the BLOB deviates from circle the larger the value of compactness will be. Signed integer [1/1000] *1)	BLOB
Center of gravity X	n		X- coordinate of center of gravity of BLOB, signed integer [1/1000] *1)	BLOB
Center of gravity Y	n		Y- coordinate of center of gravity of BLOB, signed integer [1/1000] *1)	BLOB
Center X	n		X- coordinate of fitted, geometric element (rectangle, ellipse), signed integer [1/1000]	BLOB Wafer

<PAYLOAD>				
Detector specific				**For detector:**
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**	
			*1)	Busbar
Center Y	n		Y- coordinate of fitted, geometric element (rectangle, ellipse), signed integer [1/1000] *1)	BLOB Wafer Busbar
Width	n		Width of geometric element. Width >= 0, width >= height, negative value indicates failure, signed integer [1/1000] *1)	BLOB Wafer
Height	n		Height of geometric element. Heigth >= 0, height <= width, negative value indicates failure, signed integer [1/1000] *1)	BLOB Wafer
Angle (360)	n		Orientation of width of object in degree (range: -180 ... +180°, 0° = east, counterclockwise), signed integer [1/1000] *1)	BLOB Wafer Busbar
Eccentricity	n		Eccentricity numerical (range 0,0 ... 1,0), signed integer [1/1000] *1)	BLOB
Face up/-down, area	n		Face up/down discrimination, based on area, indicated by sign, signed integer [1/1000] *1)	BLOB
String	1...n	Maximum length 127!!	Contents of Code, depending from code string length may change, if a fix string length is needed, parameters minimum string length (detector specific data output) and maximum string length (detector parameters) have to be used.	Datacode Barcode OCR
String length	n		Length of Code in Bytes	Datacode Barcode OCR

Truncated	1	F = Code complete,
		P = Code truncated
			Code truncated	Datacode
				Barcode
				OCR

<PAYLOAD>				
Detector specific				**For detector:**
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**	
Compare result	1		Result of string comparison	Datacode Barcode OCR
Quality parameter	1 ... n		Output of quality parameters according to selection	Datacode Barcode
Contrast	4		Contrast of the code (0-100%)	Barcode
Correction	4		Number of modules corrected by error corrections	Barcode
Module height	4		Height of modules in pixels	Datacode
Module width	4		Width of modules in pixels	Datacode
Confidence	5 ... n		Output of the confidence values of the individual characters	OCR
Result	4		Degree of similarity between the read string and the reference string from 0 to 100%	OCR
Min. Quality	1		Minimum required quality was achieved	OCR
Length	4		Length of busbar	Busbar
Width	4		Width of busbar	Busbar

<CHKSUM>			
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**
Checksum	1		XOR checksum of all bytes in telegram

<TRAILER>			
Name	**Number of bytes**	**ASCII contents / example**	**Significance /Comments**
Trailer	1 - max. 8	User defined, max. 8 characters	End of string (Trailer)

[*1)] All detector-specific data with decimal places are transmitted as whole numbers (multiplied by 1000) and must therefore be divided by 1000 after receipt of data.

Values are transferred in format “Big-endian”. (there are two different architectures for handling memory storage. They are called Big Endian and Little Endian and refer to the order in which the bytes are stored in memory, in the case of the VISOR® architecture the data is stored Big End In first)

Example: “Score” Value (Binary protocol)

In SensoConfig/SensoView “Score” = 35 is displayed.
Over Ethernet there will be received the following four bytes: 000,000,139,115
Formula for recalculating: (HiWordByte*256 + HiLowByte) *65536 + HiByte*256 + LoByte = Value

Because Big-endian (from Sensor) is sent calculation goes as following:
000 = HiWordByte, 000 = HiLowByte, 139 = HiByte, 115 = LoByte
(0*256 + 0) * 65536 + (139 * 256) + 115 = 35699 / 1000 = 35,699 (real score value)
Angles or other negative values are transferred in two's complement.

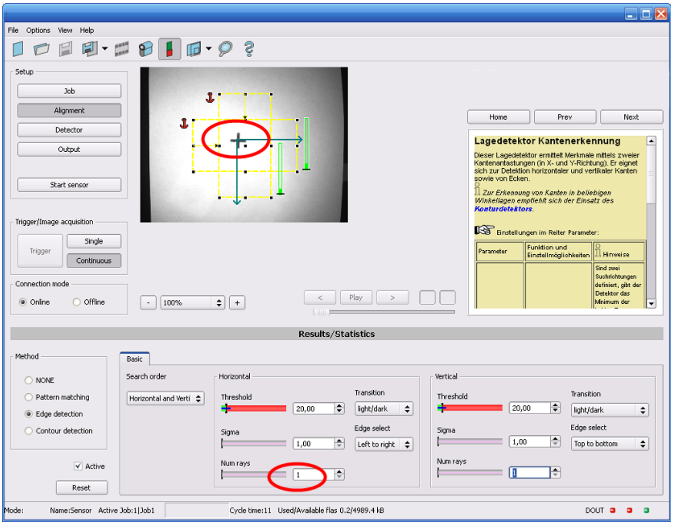
8.2 Further explanations to Edge detector (alignment)

Function of “Number search rays”

“Number search rays” parameter which defines in how many parallel sub- search regions the search area is divided. The edge detector searches in each sub region for the first edge separately.

Increasing the value of “Number search rays”, increases the chance to find the very first edge in the search area.

By increasing “Number search rays” it may happen, that the threshold value fluctuates strongly, e.g. if just the half of the search area is covered by the edge. The reason therefore is, that the first, not the strongest, edge is detected, which is above the threshold limit in search direction.



Connection mode: ● Online ○ Offline

Lagedetektor Kantenerkennung

Dieser Lagedetektor ermittelt Merkmale mittels zweier Kantenantastungen (in X- und Y-Richtung). Er eignet sich zur Detektion horizontaler und vertikaler Kanten sowie von Ecken.

Zur Erkennung von Kanten in beliebigen Winkellagen empfiehlt sich der Einsatz des ***Konturdetektors****.*

☞ Einstellungen im Reiter Parameter:

Parameter	Funktion und Einstellmöglichkeiten	Hinweise
		Sind zwei Suchrichtungen definiert, gibt der Detektor das Minimum der

Method: ○ NONE ○ Pattern matching ● Edge detection ○ Contour detection

☑ Active

Reset

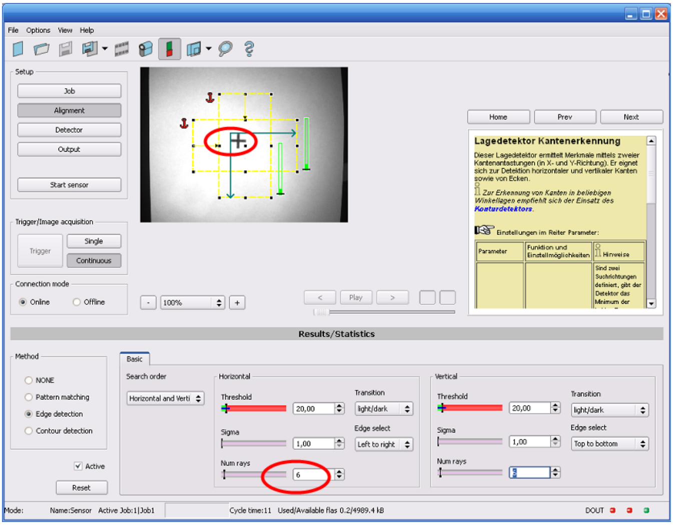
Search order: Horizontal and Verti

Horizontal: Threshold 20,00; Transition light/dark; Sigma 1,00; Edge select Left to right; Num rays 1

Vertical: Threshold 20,00; Transition light/dark; Sigma 1,00; Edge select Top to bottom; Num rays 1

Mode: Name:Sensor Active Job:1|Job1 Cycle time:11 Used/Available flas 0.2/4989.4 kB DOUT

Fig. 416: Edge detection with “Number search rays” = 1. The dominating edge, perpendicular to the search direction is found.



Connection mode: ● Online ○ Offline

Lagedetektor Kantenerkennung

Dieser Lagedetektor ermittelt Merkmale mittels zweier Kantenantastungen (in X- und Y-Richtung). Er eignet sich zur Detektion horizontaler und vertikaler Kanten sowie von Ecken.

Zur Erkennung von Kanten in beliebigen Winkellagen empfiehlt sich der Einsatz des ***Konturdetektors****.*

☞ Einstellungen im Reiter Parameter:

Parameter	Funktion und Einstellmöglichkeiten	Hinweise
		Sind zwei Suchrichtungen definiert, gibt der Detektor das Minimum der

Method: ○ NONE ○ Pattern matching ● Edge detection ○ Contour detection

☑ Active

Reset

Search order: Horizontal and Verti

Horizontal: Threshold 20,00; Transition light/dark; Sigma 1,00; Edge select Left to right; Num rays 6

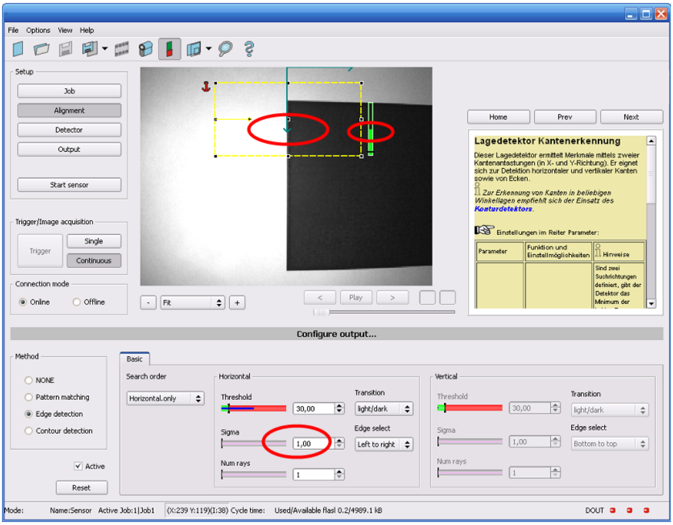
Vertical: Threshold 20,00; Transition light/dark; Sigma 1,00; Edge select Top to bottom; Num rays 1

Mode: Name:Sensor Active Job:1|Job1 Cycle time:11 Used/Available flas 0.2/4989.4 kB DOUT

Fig. 417: Edge detection with “Number search rays” >> 1. The first edge perpendicular to the search direction is found.

Function of Sigma (smoothing) to sharp or blurred edges

The edge strength represents the assumption of edge steps over a certain area in search direction, which is quantified in “Sigma” (smoothing). With sharp edges the edge strength is not increased with increasing sigma. But with blurred edges the edge strength is increased by increasing sigma value.



Connection mode: ● Online ○ Offline; Fit

Lagedetektor Kantenerkennung

Dieser Lagedetektor ermittelt Merkmale mittels zweier Kantenantastungen (in X- und Y-Richtung). Er eignet sich zur Detektion horizontaler und vertikaler Kanten sowie von Ecken.

*Zur Erkennung von Kanten in beliebigen Winkellagen empfiehlt sich der Einsatz des **Konturdetektors**.*

Einstellungen im Reiter Parameter:

Parameter	Funktion und Einstellmöglichkeiten	Hinweise
		Sind zwei Suchrichtungen definiert, gibt der Detektor das Minimum der

Method: ○ NONE ○ Pattern matching ● Edge detection ○ Contour detection

☑ Active

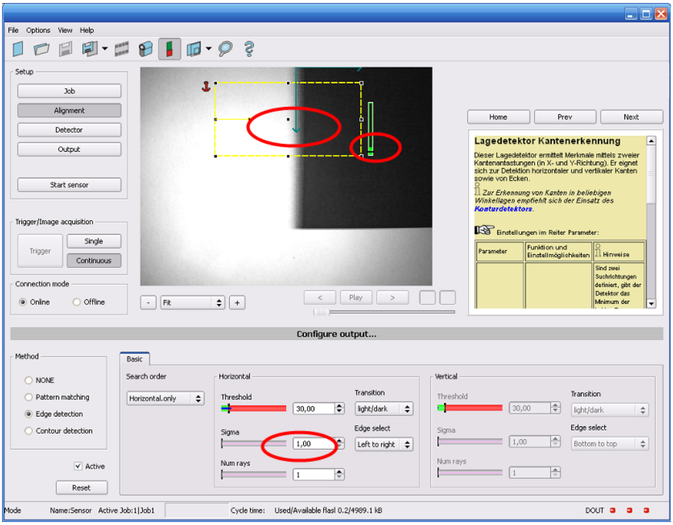
Search order: Horizontal.only

Horizontal: Threshold 30,00; Transition light/dark; Sigma 1,00; Edge select Left to right; Num rays 1

Vertical: Threshold 30,00; Transition light/dark; Sigma 1,00; Edge select Bottom to top; Num rays 1

Mode: Name:Sensor Active Job:1|Job1 (X:239 Y:119)(I:38) Cycle time: Used/Available flasl 0.2/4989.1 kB

Fig. 418: Edge detection of sharp edge. High edge strength with low sigma value (smoothing).



Connection mode: ● Online ○ Offline

Lagedetektor Kantenerkennung

Dieser Lagedetektor ermittelt Merkmale mittels zweier Kantenantastungen (in X- und Y-Richtung). Er eignet sich zur Detektion horizontaler und vertikaler Kanten sowie von Ecken.

*Zur Erkennung von Kanten in beliebigen Winkellagen empfiehlt sich der Einsatz des **Konturdetektors**.*

Einstellungen im Reiter Parameter:

Parameter	Funktion und Einstellmöglichkeiten	Hinweise
		Sind zwei Suchrichtungen definiert, gibt der Detektor das Minimum der

Method: ○ NONE ○ Pattern matching ● Edge detection ○ Contour detection

☑ Active

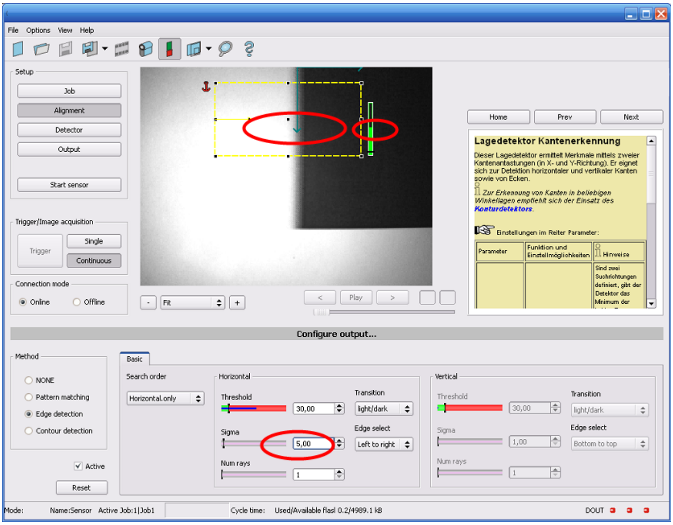
Search order: Horizontal only

Horizontal: Threshold 30,00; Transition light/dark; Sigma 1,00; Edge select Left to right; Num rays 1

Vertical: Threshold 30,00; Transition light/dark; Sigma 1,00; Edge select Bottom to top; Num rays 1

Mode Name:Sensor Active Job:1|Job1 Cycle time: Used/Available flasl 0.2/4989.1 kB DOUT

Fig. 419: Edge detection of blurred edge. Low edge strength with low sigma value.



Connection mode: ● Online ○ Offline

Lagedetektor Kantenerkennung

Dieser Lagedetektor ermittelt Merkmale mittels zweier Kantenantastungen (in X- und Y-Richtung). Er eignet sich zur Detektion horizontaler und vertikaler Kanten sowie von Ecken.

*Zur Erkennung von Kanten in beliebigen Winkellagen empfiehlt sich der Einsatz des **Konturdetektors**.*

Einstellungen im Reiter Parameter:

Parameter	Funktion und Einstellmöglichkeiten	Hinweise
		Sind zwei Suchrichtungen definiert, gibt der Detektor das Minimum der

Method: ○ NONE ○ Pattern matching ● Edge detection ○ Contour detection

☑ Active

Search order: Horizontal only

Horizontal: Threshold 30,00; Transition light/dark; Sigma 5,00; Edge select Left to right; Num rays 1

Vertical: Threshold 30,00; Transition light/dark; Sigma 1,00; Edge select Bottom to top; Num rays 1

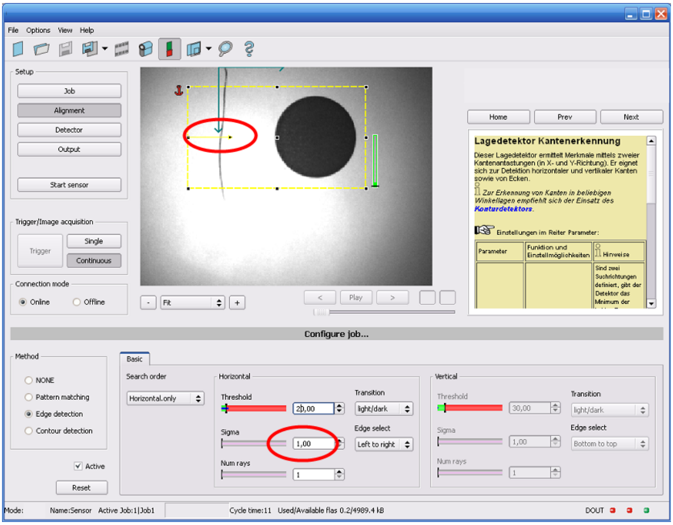
Mode: Name:Sensor Active Job:1|Job1 Cycle time: Used/Available flasl 0.2/4989.1 kB DOUT

Fig. 420: Edge detection of blurred edge. High edge strength with high sigma value.

Function of Sigma (Smoothing) to residual edges

Like mentioned above, the edge strength represents the assumption of edge steps over a certain area in search direction, which is quantified in “Sigma” (smoothing).

If in this area edges are found with different polarity (dark- bright: positive polarity, bright-dark: negative polarity) it´s edges steps can neutralize each other. This can be used to eliminate residual edges, by choosing a sigma value which is high enough.



Lagedetektor Kantenerkennung

Dieser Lagedetektor ermittelt Merkmale mittels zweier Kantenantastungen (in X- und Y-Richtung). Er eignet sich zur Detektion horizontaler und vertikaler Kanten sowie von Ecken.

*Zur Erkennung von Kanten in beliebigen Winkellagen empfiehlt sich der Einsatz des **Konturdetektors**.*

Einstellungen im Reiter Parameter:

Parameter	Funktion und Einstellmöglichkeiten	Hinweise
		Sind zwei Suchrichtungen definiert, gibt der Detektor das Minimum der

Method: ○ NONE ○ Pattern matching ● Edge detection ○ Contour detection

☑ Active

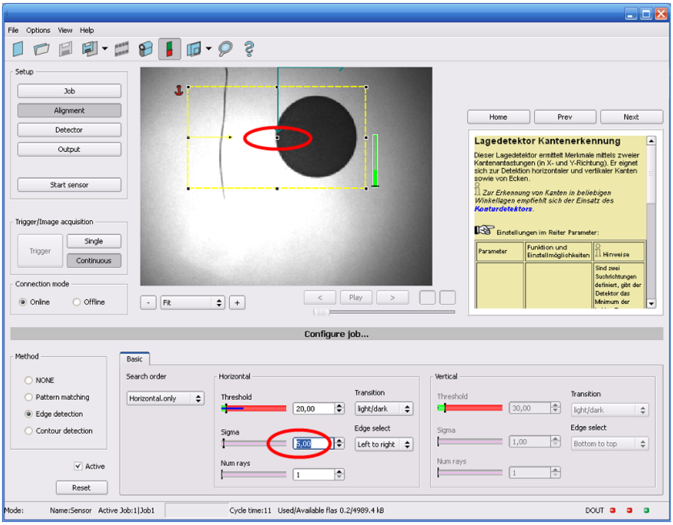
Search order: Horizontal only

Horizontal: Threshold 20,00; Transition light/dark; Sigma 1,00; Edge select Left to right; Num rays 1

Vertical: Threshold 30,00; Transition light/dark; Sigma 1,00; Edge select Bottom to top; Num rays 1

Mode: Name:Sensor Active Job:1|Job1 Cycle time:11 Used/Available flas 0.2/4989.4 kB DOUT

Fig. 421: Edge detection with sigma value = 1. Residual edge is not eliminated.



Connection mode: ● Online ○ Offline

Fit

Lagedetektor Kantenerkennung

Dieser Lagedetektor ermittelt Merkmale mittels zweier Kantenantastungen (in X- und Y-Richtung). Er eignet sich zur Detektion horizontaler und vertikaler Kanten sowie von Ecken.

*Zur Erkennung von Kanten in beliebigen Winkellagen empfiehlt sich der Einsatz des **Konturdetektors**.*

Einstellungen im Reiter Parameter:

Parameter	Funktion und Einstellmöglichkeiten	Hinweise
		Sind zwei Suchrichtungen definiert, gibt der Detektor das Minimum der

Method: ○ NONE ○ Pattern matching ● Edge detection ○ Contour detection

☑ Active

Basic

Search order: Horizontal.only

Horizontal: Threshold 20,00; Transition light/dark; Sigma 5,00; Edge select Left to right; Num rays 1

Vertical: Threshold 30,00; Transition light/dark; Sigma 1,00; Edge select Bottom to top; Num rays 1

Mode: Name:Sensor Active Job:1|Job1 Cycle time:11 Used/Available flas 0.2/4989.4 kB DOUT

Fig. 422: Edge detection with sigma value >> 1. Residual edge is eliminated.

8.3 Care and maintainance

8.3.1 Cleaning

The VISOR® vision sensor is to be cleaned with a clean, dry cloth.

Dirt on the front panel is to be cleaned with a soft cloth and a small amount of plastic cleaner if necessary.

Attention

- Never use aggressive detergents such as solvents or benzine.
- Never use sharp objects. Do not scratch!

Kontaktadressen / Contact adresses / Contacts

Deutschland

SensoPart Industriesensorik GmbH
Nägelseestr. 16
79288 Gottenheim
Deutschland

Tel.: +49 (0) 7665 94769-0
Fax: +49 (0) 7665 94769-765
info@sensopart.de
www.sensopart.com

France

SensoPart France SARL
11, rue Albert Einstein
Espace Mercure
F-77420 Champs sur Marne

Tel.: +33 1 64 73 00 61
Fax: +33 1 64 73 10 87
info@sensopart.fr
www.sensopart.com

United Kingdom

SensoPart UK Ltd.
Pera Business Park,
Nottingham Road,
Melton Mowbray
Leicestershire, LE13 0PB
United Kingdom

Tel: +44 1664 561539
uk@sensopart.com
www.sensopart.com

USA

SensoPart Inc.
28400 Cedar Park Blvd
Perrysburg OH 43551
USA

Tel.: +1 866 282 7610
Fax: +1 419 931 7697
usa@sensopart.com
www.sensopart.com

China

SensoPart (Shanghai)
Industrial Automation Co. Ltd.
202, No. 35, Lane 1555, West Jinshajiang Road
Jiading District
Shanghai, 201803
China

Tel.: +86 21 6901 7660
+86 21 6901 7668
Fax +86 21 6951 6058
china@sensopart.cn
www.sensopart.com