RIFTEK
Sensors & Instruments

2D LASER SCANNERS

RF62x Series

User's manual

www.riftek.com
info@riftek.com

Certified according to ISO 9001:2015

Contents

1. Safety precautions ... 6
2. CE compliance ... 6
3. Laser safety ... 6
4. General information ... 6
5. Structure and operating principle ... 7
6. Configurations, operating modes, options ... 7
7. Basic technical data ... 8
   7.1. Specifications ... 8
      7.1.1. RF627Smart ... 9
      7.1.2. RF627BiSmart ... 10
      7.1.3. RF628 ... 10
      7.1.4. RF629 ... 11
      7.1.5. RF6292 ... 12
8. Example of item designation when ordering ... 12
9. Overall demands for mounting ... 13
10. Connection ... 13
   10.1. Button and indication ... 13
11. Ethernet interface and user software development ... 14
12. Network configuration and the first connection ... 14
   12.1. Network configuration ... 14
   12.2. First connection ... 15
13. Web interface ... 15
14. Web interface operating modes ... 18
15. Search for scanners on the network and connection ... 23
16. Results display area ... 23
   16.1. Controls ... 23
   16.2. Display modes ... 24
      16.2.1. Profile mode ... 24
      16.2.2. Video mode ... 25
17. Setting parameters ... 25
18. "Network" tab. Network parameters ... 26
19. "General" tab. General parameters ... 27
   19.1. CMOS sensor parameters ... 28
   19.2. Laser parameters ... 29
   19.3. Image quality settings ... 29
      19.3.1. Exposure time and laser power ... 29
      19.3.2. Multiple exposure mode ... 29
      19.3.3. Removing background light from extraneous light sources ... 30
   19.4. ROI mode settings ... 32
   19.5. Data stream control ... 33
20. "Processing" tab. Profile extraction settings ... 35
   20.1. "Pre processing" section. Profile extraction parameters ... 35
      20.1.1. "Peak selection mode" parameter ... 36
   20.2. "Post processing" section. Filtering ... 38
21. "Triggering" tab. Triggering modes ... 39
   21.1. Time cycle ... 40
   21.2. Synchronization diagram ... 41
   21.3. Selecting a source of synchronization events ... 42
   21.4. Synchronization by external trigger ... 42
      21.4.1. Setting the inputs ... 43
      21.4.2. Setting the encoder counter ... 43
      21.4.3. Examples ... 44
   21.5. Setting the outputs ... 45
22. "Triggering settings" tab. Synchronization of multiple scanners ... 46

22.1. Synchronous measurements 46
22.2. Asynchronous measurements 46
23. "Dump" tab. Accumulated profiles parameters 47
23.1. "Dump control" section. Building 3D models 48
23.2. "3D view" section. 3D model display parameters 48
23.3. "Download" section. Downloading profiles 49
23.4. Operations with profiles 49
23.4.1. Accumulation of profiles in internal memory of the scanner 49
23.4.2. Viewing accumulated profiles 49
23.4.3. Export of accumulated profiles 50
23.4.4. Importing a point cloud for viewing 51
23.4.5. Point cloud measurement tools 51
24. "System" tab 62
24.1. "Information" section 63
24.2. "Update" section 63
24.2.1. Updating and saving the firmware 63
24.2.2. Updating the calibration table 64
24.3. "Licenses" section 65
24.4. "Logs" section 66
25. "Smart" tab 67
25.1. Smart blocks and parameters 68
25.1.1. "Smart Blocks" tab 69
25.1.2. "Block Settings" tab 69
25.1.3. "Profile Approximation" tab 70
25.2. Creating a Smart function 71
25.2.1. Stage 1. Profile approximation 71
25.2.1.1. Splitting profile points into fragments 72
25.2.1.2. Splitting each fragment into a set of approximating line segments and arcs 73
25.2.1.2.1. Approximation by segments 73
25.2.1.2.2. Approximation by arcs 74
25.2.1.3. Filtering 76
25.2.1.4. Clarification of approximating line segments and arcs 77
25.2.2. Stage 2. Building a graph 77
25.2.2.1. Results display area 78
25.2.2.2. Example of building a graph 79
25.2.3. How it works 80
25.2.4. Saving and loading Smart functions 81
25.3. Smart blocks 82
25.3.1. Data types 82
25.3.2. Sections 83
25.3.2.1. "Feature detectors" section 84
25.3.2.2. "Math functions" section 91
25.3.2.3. "Welding" section 95
25.3.2.4. "Converters" section 102
25.3.2.5. "Control" section 104
25.3.2.6. "Base IO" section 106
25.3.2.7. "Industrial IO" section 108
25.3.2.8. "Robot IO" section 110
26. Maintenance 112
27. Troubleshooting 112
28. Annex 1. Recovery mode 113
29. Annex 2. Editing defective pixels 114
30. Annex 3. Web API 116
30.1. General device information 116
30.2. Reading and writing parameters 116
30.3. Saving and restoring settings. Rebooting the device 116
30.4. Getting information from the device log file 117

30.5. Authorization ... 117
30.6. Profile request ... 117
30.7. Smart ... 117
31. Annex 4. "Template detector" smart block and Template Editor ... 118
31.1. General information ... 118
31.2. Template structure and search principle ... 118
31.3. Template editor ... 119
31.3.1. Working with the template editor ... 119
31.3.1.1. Creating template set outputs ... 120
31.3.1.2. Adding a template to a set ... 121
31.3.1.3. Creating template elements ... 121
31.3.1.3.1. Creation of template elements by the user ... 121
31.3.1.3.2. Automatic creation of template elements ... 122
31.3.1.4. Creating constraints ... 123
31.3.1.4.1. Self constraints ... 123
31.3.1.4.2. Relative constraints ... 124
31.3.1.5. Adding constraints to a template ... 125
31.3.1.6. Creating and customizing template variants ... 127
31.3.1.7. Assigning template output values ... 128
31.3.1.8. Assigning a custom icon to a template ... 130
31.3.1.9. Saving a set of templates to the non-volatile memory of the scanner ... 130
31.3.2. Example of creating a custom template set ... 131
31.3.2.1. Step 1. Creating the template set file ... 131
31.3.2.2. Step 2. Adding template set outputs ... 131
31.3.2.3. Step 3. Adding a template ... 132
31.3.2.4. Step 4: Adding template variants (if necessary) ... 137
31.3.2.5. Step 5. Assigning template outputs ... 138
31.3.2.6. Step 6. Saving the template file to the non-volatile memory of the scanner ... 141
32. Annex 5. "C-script" smart block ... 142
32.1. General information ... 142
32.2. Supported data types ... 143
32.3. Supported methods ... 145
32.3.1. Basic methods ... 145
32.3.2. Special methods ... 145
32.3.2.1. System methods ... 145
32.3.2.2. Methods for working with input ports ... 145
32.3.2.3. Methods for working with output ports ... 146
32.3.2.4. Mathematical methods for speeding up calculations in 3D space ... 146
32.4. Examples of scripts ... 147
33. Annex 6. Calibration of the scanner relative to the robot ... 148
33.1. Data sampling ... 149
33.2. Calculation of the coordinate transformation matrix ... 153
33.3. Manual correction ... 154
33.4. Calibration accuracy verification ... 154
33.4.1. Verification by plane ... 154
33.4.2. Verification by 3D point ... 155
34. Annex 7. Overall and mounting dimensions ... 157
34.1. RF627Smart ... 157
34.2. RF627BiSmart ... 160
34.3. RF628Smart ... 161
34.4. RF629Smart ... 164
34.5. RF6292Smart ... 164
34.6. Scanners with additional options ... 165
34.6.1. Example of a scanner with replaceable protective windows, EW option ... 165
34.6.2. Example of a scanner with air cooling, AK-EW-AC option ... 167
34.6.3. Example of a scanner with water cooling, AK-EW-AC option ... 167
35. Annex 8. Connectors and cables ... 168

35.1. Pin assignment of connectors 168
35.2. Cables 169
36. Annex 9. ProfiTalk protocol 171
36.1. Specification 171
36.2. Scanner support 171
36.3. Message structure at the transport layer 171
36.4. Search for scanners on the network 172
36.5. Transmission of control commands 173
36.6. Commands 174
36.6.1. READ_PARAMETERS_DESCRIPTION 174
36.6.2. READ_PARAMETERS 175
36.6.3. WRITE_PARAMETERS 176
36.6.4. SAVE_CURRENT_PARAMETERS 176
36.6.5. SAVE_RECOVERY_PARAMETERS 177
36.6.6. LOAD_RECOVERY_PARAMETERS 177
36.6.7. REBOOT_DEVICE 177
36.6.8. READ_PROFILES_DUMP 178
36.7. Profile transfer format 178
36.8. Video frame transfer format 179
37. Warranty policy 180
38. Technical support 180
39. Revisions 180
40. Distributors 182

1. Safety precautions

- Use supply voltage and interfaces indicated in the scanner specifications.
- In connection/disconnection of cables, the scanner power must be switched off.
- Do not use scanners in locations close to powerful light sources.
- To obtain stable results, wait about 20 minutes after scanner activation to achieve uniform scanner warm-up.
- Scanners must be grounded.
- To avoid overheating and ensure proper operation, the scanner must be installed on a thermally conductive material that provides good heat dissipation.

2. CE compliance

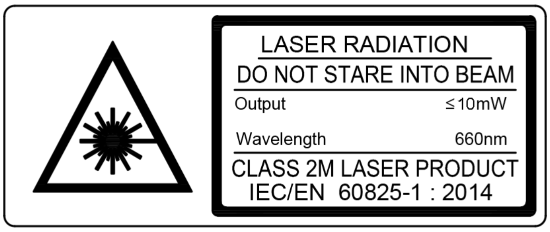
Laser scanners have been developed for use in industry and meet the requirements of the following Directives:

- EU directive 2014/30/EU. Electromagnetic compatibility (EMC).
- EU directive 2011/65/EU, "RoHS" category 9.

3. Laser safety

Scanners belong to 2M laser safety class according to IEC/EN 60825-1:2014.

Scanners make use of an c.w. 660 nm or 405 nm or 450 nm or 808 nm wavelength semiconductor laser. Maximum output power is 10 mW. The following warning label is placed on the scanner housing:



The following safety measures should be taken while operating the scanners:

- Do not target laser beam to humans.
- Do not disassemble the scanner.
- Avoid staring into the laser beam.

4. General information

Laser scanners are designed for non-contact measuring and checking of surface profile, position, displacement, dimensions, sorting and sensing of technological objects, 3D models construction. This User's Manual is for all RF62x laser scanners, namely:

- RF627Smart
- RF627BiSmart
- RF628Smart
- RF629Smart
- RF6292Smart

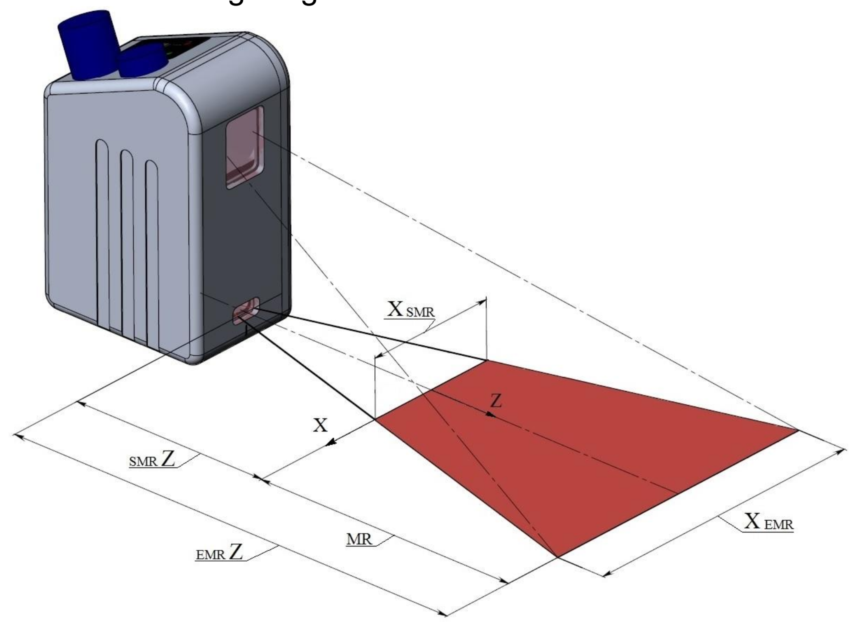
A common web interface is used to configure scanners. Laser scanners differ in technical characteristics (rate, resolution) and a set of supported functions.

5. Structure and operating principle

Operation of the scanners is based on the principle of optical triangulation (see Figure below).

Radiation of a semiconductor laser is formed by a lens in a line and projected to an object. Radiation scattered from the object is collected by the lens and directed to a two-dimensional CMOS image sensor or on two sensors, located symmetrically with respect to the laser (binocular scanners). The image of object outline thus formed is analyzed by a FPGA and signal processor, which calculates and transmits the distance to the object (Z-coordinate) for each point of the set along the laser line on the object (X-coordinate). Scanners are characterized by the following geometrical parameters:

- smrZ - the beginning of the range for the Z-coordinate,
- MR - the measuring range for the Z-coordinate,
- Xsmr - the measuring range for the X-coordinate at the beginning of Z,
- Xemr - the measuring range for the X-coordinate at the end of Z.



6. Configurations, operating modes, options

The following configurations are available:

- red laser scanners, 660 nm;
- blue laser scanners (BLUE version), 405 or 450 nm;
- infrared laser scanners (IR version), 808 nm;
- powerful red laser scanners, 637 nm (preliminary discussion required);
- powerful blue laser scanners, 450 nm (preliminary discussion required).
- powerful infrared laser scanners, 808 nm (preliminary discussion required).

We use different lasers due to a wide range of applications. For example, the use of blue lasers instead of red ones is optimal for the control of shiny materials, high-temperature objects and organic materials.

The use of scanners with lasers of different wavelengths in one measurement system makes it possible to avoid the scanners mutual influence and greatly simplifies the system construction. An example of system implementation: https://youtu.be/9evAIXqrPas.

Scanners can be equipped with a built-in heater for operation under low-temperature conditions. Scanners can be equipped with the air (water) cooling system and the air-knife system for windows.

Scanners support the ROI function, which makes it possible to increase the working frequency of the scanner in the limited working range.

The **Smart** scanners make it possible to measure geometric parameters of the object profile in real time directly in the scanner without connecting to a computer. Analysis, calculations, measurements, tolerance control are carried out according to the algorithm created by the user. To build an algorithm, a simple and intuitive tool is provided - a computation graph. The graph is formed from a library of ready-made blocks. Various combinations of blocks and connections between them allow the user to create an almost unlimited number of measuring functions, as well as to process profiles of any complexity. Measurement results can be transmitted via various protocols (Ethernet/IP, Modbus TCP, UDP, etc.), as well as to the logic outputs of the scanner in order to control the actuators and notify about product suitability. Examples of implementation: https://youtu.be/-KvKu5MQ6JM/

In addition, **Smart** scanners contain built-in protocols for interaction with various industrial robots and cobots, allowing the scanners to be integrated into automation systems without the need to purchase additional equipment.

Licensing of connected protocol packages for **Smart** scanners is described in par. 24.3. "Licenses section".

7. Basic technical data

Detailed specifications of the scanners are presented in the following paragraphs. This table compares the parameters characterizing the features of the models:

Parameter	**RF627Smart**	**RF627BiSmart Dual Camera Profiler**	**RF628Smart**	**RF629Smart**	**RF6292Smart**	**RF630** (available soon)	**RF631** (available soon)
Nominal sampling rate (full working range), profiles/s	not less than 520	not less than 520	not less than 4000	not less than 1000	not less than 4000	3400 (13600 for 1024 resolution)	1900 (7600 for 2300 resolution)
Maximum sampling rate (ROI mode), profiles/s	4200	4200	21500	21500	21500	14500 (58000 for 1024 resolution)	6500 (26000 for 2300 resolution)
Resolution (X axis), points	728 or 1456	for combined profile, up to 1456 or 2912	640 or 1280	1280 or 2560	1280 or 2560	2048	4608
Smart mode and industry protocols	YES	YES	YES	YES	YES	NO	NO
Xend/Z ratio	≤1	≤1	≤1	≤1	>2.5	>2	>2.5...4

7.1. Specifications

Laser	
660 nm or 405 nm or 450 nm or 808 nm Class 2M according to IEC/EN 60825-1:2014 or Class 3B on request	
Interface	
Basic	Ethernet / 1000 Mbps
Synchronization inputs	RS422, 3 channels
Laser on/off hardware input	1
Outputs	RS422, 1 channel
Power supply	9...30 V or 12...39 V for scanners with Blue laser
Power consumption, not more	RF627Smart - 6 W (without a built-in heater) RF627BiSmart - 11 W

	RF628Smart - 17 W RF629Smart and RF6292Smart - 17 W
Environmental resistance	
Enclosure rating	IP67
Vibration	20 g / 10...1000 Hz, 6 hours for each of XYZ axes
Shock	30 g / 6 ms
Operating ambient temperature	-20…+40°C, or -40…+40°C for scanners with built-in heater, or -40…+120°C for scanners with built-in heater and cooling system
Storage temperature	-20…+70°C
Relative humidity	5-95% (no condensation)
Housing/windows material	aluminum/glass

The housing of the scanner is made of anodized aluminum. The front panel of the housing has two windows: the output window and the window for receiving radiation reflected from the object under control. The housing has fastening holes for installing the scanner on the equipment. Some models are equipped with an adjustable support that makes it possible to implement three options for mounting the scanner. The housing has one or two connectors, **Reset** button and LED indicators.

7.1.1. RF627Smart

Sampling rate, accuracy, resolution	
Nominal sampling rate (full working range), not less	520 profiles/s
Maximum sampling rate (ROI mode)	4200 profiles/s
Linearity (measurement error), Z axis	±0.01% of the range (standard mode)[1)]
Resolution, X axis	728 or 1456 points (programmable value)

Range	MR, mm	smrZ, mm	emrZ, mm	Xsmr, mm	Xemr, mm
25/10-8/11	10	25	35	8	11
65/25-20/22	25	65	90	20	22
75/50-30/41	50	75	125	30	41
70/100-48/82	100	70	170	48	82
70/150-58/122	150	70	220	58	122
95/150-53/106	150	95	245	53	106
82/200-60/150	200	82	282	60	150
90/250-65/180	250	90	340	65	180
180/250-170/278	250	180	430	170	278
190/300-160/300	300	190	490	160	300
220/300-203/330	300	220	520	203	330
260/400-210/400	400	260	660	210	400
325/500-268/500	500	325	825	268	500
400/600-320/600	600	400	1000	320	600
475/700-374/700	700	475	1175	374	700
545/800-425/800	800	545	1345	425	800
615/900-480/900	900	615	1515	480	900
690/1000-535/1000	1000	690	1690	535	1000
620/1165-430/1010	1165	620	1785	430	1010

Special sensors for scanning internal holes and threads:

Range	MR, mm	smrZ, mm	emrZ, mm	Xsmr, mm	Xemr, mm	Body diameter, mm
2/10-8/11	10	2	12	8	11	45
1/25-20/22	25	1	26	20	22	56

1) - linearity for height measurement inside scanner FOV.

Overall dimensions and weight of the scanners are given in Annex 7.
Detailed CAD documentation (2D and 3D) is available here:
https://riftek.com/upload/iblock/0ba/2D_CAD.rar
https://riftek.com/upload/iblock/c80/RF627_3D.zip

7.1.2. RF627BiSmart

Sampling rate, accuracy, resolution	
Nominal sampling rate (full working range), not less	520 profiles/s
Maximum sampling rate (ROI mode)	4200 profiles/s
Linearity (measurement error), Z axis	±0.01% of the range 1)
Resolution for combined profile, X axis	1456 or 2912 points (programmable value)

Range	MR, mm	smrZ, mm	emrZ, mm	Xsmr, mm	Xemr, mm
27/10-8/11	10	27	37	8	11
65/25-20/22	25	65	90	20	22
75/50-30/41	50	75	125	30	41
70/100-48/82	100	70	170	48	82
70/150-58/122	150	70	220	58	122
95/150-53/106	150	95	245	53	106
82/200-60/150	200	82	282	60	150

1) - linearity for height measurement inside scanner FOV.

Overall dimensions and weight of the scanners are given in Annex 7.
Detailed CAD documentation (2D and 3D) is available here:
https://riftek.com/upload/iblock/0ba/2D_CAD.rar
https://riftek.com/upload/iblock/c80/RF627_3D.zip

7.1.3. RF628

Sampling rate, accuracy, resolution	
Nominal sampling rate (full working range), not less	4000 profiles/s
Maximum sampling rate (ROI mode)	21500 profiles/s
Linearity (measurement error), Z axis	±0.01% of the range1)
Resolution, X axis	640 or 1280 points (programmable value)

Range	MR, mm	smrZ, mm	emrZ, mm	Xsmr, mm	Xemr, mm
65/10-11/12	10	65	75	11	12
75/25-20/22	25	75	100	20	22
90/50-32/44	50	90	140	32	44
125/75-42/58	75	125	200	42	58
150/100-50/74	100	150	250	50	74
150/150-64/112	150	150	300	64	112
210/300-148/276	300	210	510	148	276

285/400-198/376	400	285	685	198	376
370/500-250/466	500	370	870	250	466
450/600-300/556	600	400	1000	320	600
530/700-350/650	700	530	1230	350	650
610/800-400/744	800	610	1410	400	744
685/900-450/836	900	685	1585	450	836
765/1000-500/930	1000	765	1765	500	930

[1)] - linearity for height measurement inside scanner FOV.

Overall dimensions and weight of the scanners are given in Annex 7.
Detailed CAD documentation (2D and 3D) is available here:
https://riftek.com/upload/iblock/0ba/2D_CAD.rar
https://riftek.com/upload/iblock/c80/RF627_3D.zip

7.1.4. RF629

Sampling rate, accuracy, resolution	
Nominal sampling rate (full working range), not less	1000 profiles/s
Maximum sampling rate (ROI mode)	21500 profiles/s
Linearity (measurement error), Z axis	±0.01% of the range[1)]
Resolution, X axis	1280 or 2560 points (programmable value)

Range	MR, mm	smrZ, mm	emrZ, mm	Xsmr, mm	Xemr, mm
60/25-22/26	25	60	85	22	26
60/50-36/50	50	60	110	36	50
65/100-56/100	100	65	165	56	100
90/150-70/140	150	90	240	70	140
110/200-84/178	200	110	310	84	178
95/250-100/250	250	95	345	100	250
110/300-120/300	300	190	490	120	300
145/400-158/400	400	145	545	158	400
180/500-198/500	500	180	680	198	500
230/600-236/600	600	230	830	236	600
265/700-274/700	700	265	965	274	700
310/800-314/800	800	310	1110	314	800
345/900-352/900	900	345	1245	352	900
375/1000-392/1000	1000	375	1375	392	1000

[1)] - linearity for height measurement inside scanner FOV.

Overall dimensions of the scanners are given in Annex 7.
Detailed CAD documentation (2D and 3D) is available here:
https://riftek.com/upload/iblock/0ba/2D_CAD.rar
https://riftek.com/upload/iblock/c80/RF627_3D.zip

7.1.5. RF6292

Sampling rate, accuracy, resolution	
Nominal sampling rate (full working range), not less	4000 profiles/s
Maximum sampling rate (ROI mode)	21500
Linearity (measurement error), Z axis	±0.01% of the range[1)]
Resolution, X axis	1280 or 2560 points (programmable value)

Range	MR, mm	smrZ, mm	emrZ, mm	Xsmr, mm	Xemr, mm
70/5-24/24	5	75	80	24	24
80/15-40/44	15	80	95	40	44
95/25-70/81	25	95	120	70	81
135/35-90/105	35	135	170	90	105
170/45-110/130	45	170	215	110	130
170/75-146/194	75	170	245	146	194
220/90-200/256	90	220	310	200	256
355/120-302/376	120	355	575	302	376
455/170-400/500	170	455	625	400	500
550/225-500/634	225	550	775	500	634

[1)] - linearity for height measurement inside scanner FOV.

Overall dimensions of the scanners are given in Annex 7.
Detailed CAD documentation (2D and 3D) is available here:
https://riftek.com/upload/iblock/0ba/2D_CAD.rar
https://riftek.com/upload/iblock/c80/RF627_3D.zip

8. Example of item designation when ordering

RF62X.(WAVE)-smrZ/MR-Xsmr/Xemr-M(R)-H-AK-EW-AC-IndIO

Symbol	Description
X	7Smart – RF627Smart scanner; 7BiSmart – RF627BiSmart scanner4; 8Smart, 9Smart, 92Smart – RF628Smart, RF629Smart and RF6292Smart scanners respectively.
(WAVE)	Laser wavelength. 660 nm – no symbol, 405 nm or 450 nm – BLUE, 808 nm – IR.
smrZ	Beginning of the measuring range for Z, mm.
MR	Measuring range for Z, mm.
Xsmr	Measuring range for X-coordinate at the beginning of the measuring range for Z-coordinate, mm.
Xemr	Measuring range for X-coordinate at the end of the measuring range for Z-coordinate, mm.
M	Cable length, m .
R	Option, robot-cable.
H	Built-in heater.
AK	Air knife for windows.
EW	Removable protective windows.
AC (WC)	AC – air cooling system, WC – water cooling system. NOTE. The WC option includes the AC option by default.
IndO	Smart scanner version depending on supported protocols. **Basic** – basic version, by default contains smart blocks of profile primitives extraction, smart blocks of their mathematical and statistical processing, smart blocks of data transmission/reception in the form of TCP and UDP packets.

	Ind – Industrial version, contains smart blocks of "Base" package, as well as smart blocks of data transmission/reception via industrial protocols (EthernetIP, ModbusTCP) and smart blocks of interaction with industrial robots

Example: RF627BLUE-70/50-30/42-5-Ind – Scanner with a blue laser, smrZ – 70 mm, MR – 50 mm, Xsmr – 30 mm, Xemr – 42 mm, cable length – 5 m, Industrial version.

9. Overall demands for mounting

The scanner should be positioned so that the object under control has to be placed within the working range of the scanner. In addition, no foreign objects should be allowed to stay on the path of the incident and reflected laser radiation.

Where the objects to be controlled have intricate shapes and textures, the incidence of mirror component of the reflected radiation to the receiving window should be minimized.

> **ATTENTION!**
> The scanner must be grounded. Static electricity may cause the failure of electronic components.

10. Connection

Depending on the modification, scanners are supplied with:

- two cables (1 - cable for connecting the scanner to the Ethernet network; 2 - power cable with synchronization and output lines) or
- one universal cable.

> **ATTENTION!**
> Below is a description of the cables that come with standard configuration scanners. Documentation on the cables is always included in the delivery package.

Pin assignment of connectors and cables is given in Annex 8.

10.1. Button and indication

To reboot the scanner, press the **Reset** button for 5 seconds. If you press the **Reset** button for 1 second, a broadcast packet containing a response to the “GET_HELLO” command will be sent in accordance with the service protocol.

Indication:

Red LED indication	
Flashes	Scanner software is loading from Flash memory
Lights up constantly	Scanner is ready to operate
Shows SOS signal (three short - three long - three short)	Scanner is operating in Recovery mode
Green LED indication	
Flashes for 0.5 sec with a period of about 3 sec	Network connection is not available
Flashes quickly (individual flashes are not visible to the eye)	Network connection is functioning normally, the speed is 1000 Mbps
Flashes quickly (individual flashes are visible to the eye)	Network connection is functioning normally, the speed is 100 Mbps
Flashes twice, then pauses (with red LED flashing)	Connection speed is slower than required for data transfer by the scanner

11. Ethernet interface and user software development

Profiles are transmitted via the UDP protocol and the proprietary ProfiTalk protocol.

The results of smart function calculations can be transmitted either together with the profile or separately via the ProfiTalk protocol (in development).

Scanner settings can be changed in four ways:

1. Through the embedded web interface (see a description below).

2. Through software developed by the customer using the ProfiTalk protocol described in Annex 9.

3. Through software developed by the customer using the provided SDK (Software Development Kit). The SDK includes the detailed description of all functions of the library and the examples of programs in different languages (C, C++, C#, Python), and also the examples of using the libraries in different environments (MATLAB, LabVIEW). The SDK is compatible with any operating systems of the Windows, Linux and MacOS families.

- SDK source code, as well as the necessary information for downloading, installing and configuring the development environment: https://github.com/RIFTEK-LLC/RF62X-SDK
- Developer guide: https://github.com/RIFTEK-LLC/RF62X-SDK/blob/master/Docs/RF62X-SDK.en.pdf
- Latest library releases: https://github.com/RIFTEK-LLC/RF62X-SDK/releases
- Demo videos of compiling and running the SDK: https://cloud.riftek.com/index.php/s/q55Zq8i8kccAERj

4. Through Web API using GET and PUT HTTP requests (see Annex 3 of this Manual).

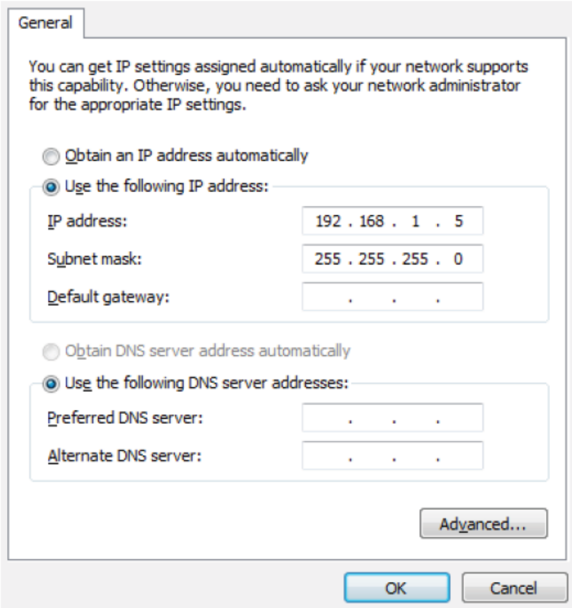
12. Network configuration and the first connection

12.1. Network configuration

All scanners are shipped with the following network configuration unless otherwise specified in the order:

- Autonegotiation of connection speed (100/1000 Mbps)
- IP address of the scanner: 192.168.1.30
- Subnet mask: 255.255.255.0
- Gateway: 192.168.1.1
- Host IP address (device that receives profiles): 192.168.1.2
- Host port that receives data: 50001
- HTTP connection port (for connecting a browser): 80
- Service port of the scanner: 50011

Since the laser scanner is configured to work in the 192.168.1.* address space, configure the network card of your PC, for example, as follows:



The network settings of the scanner can be changed using the service software (SDK), the service protocol, or via the web page of the scanner.

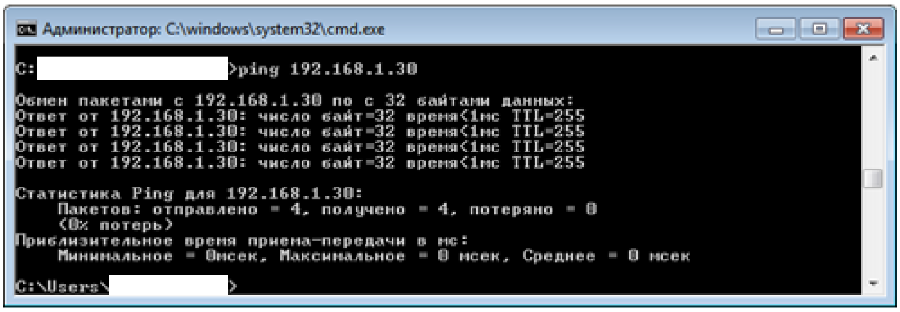
NOTE. Ethernet Jumbo frames are not supported.

12.2. First connection

- Perform the network configuration in accordance with the previous paragraph.
- Connect the scanner to the PC or to the network switch.
- Connect the power supply (9...30V) to the scanner (cable #2, a red wire is "plus" of the power supply, a brown wire is "minus").

Within 8 seconds after powering on, the FPGA firmware is loaded and the Ethernet interface is initialized (the red LED blinks).

Next, it is recommended to check the connection using the console command "ping 192.168.1.30 (or the current IP address of the scanner)". If all the settings are correct, the scanner will respond to the command. A typical result is shown below:

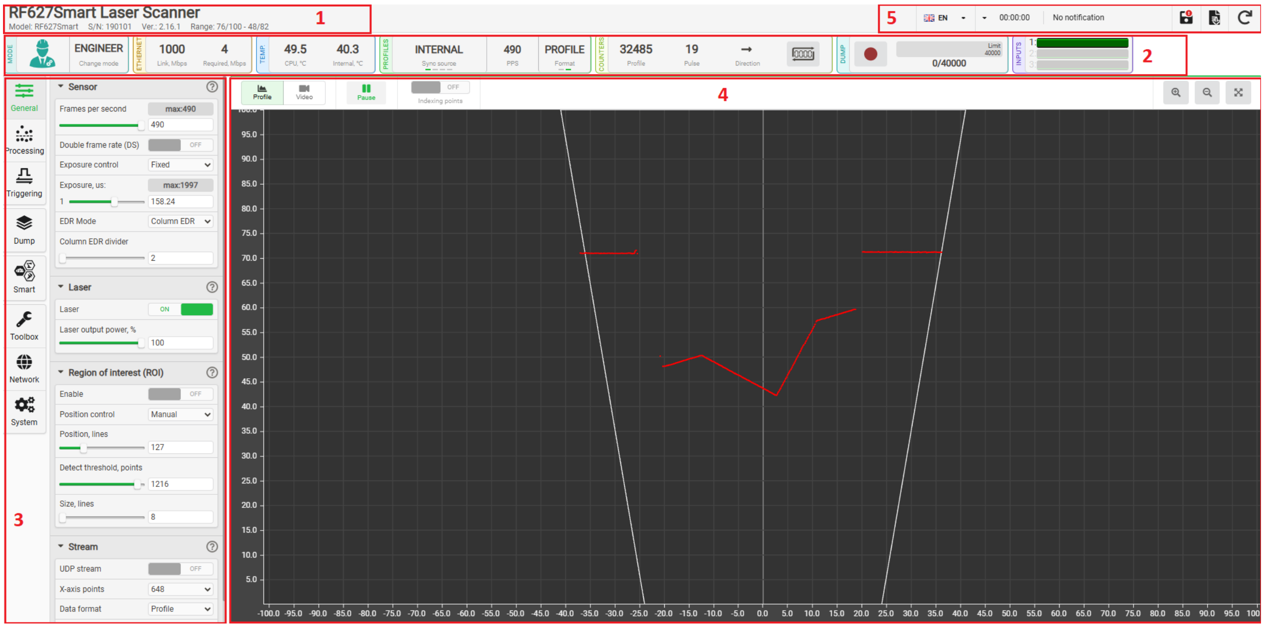


The scanner is ready to operate.

To turn off the scanner, turn off the power supply.

13. Web interface

The web interface is intended to test the operation and configure the parameters of RF62X scanners. To access the web interface, enter the IP address of the scanner into the address bar of the web browser:

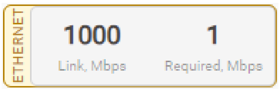
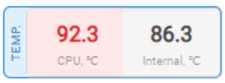
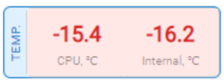
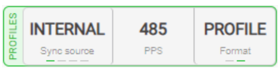


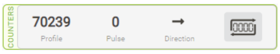
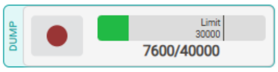
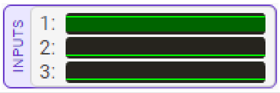
The web page is divided into five areas:

1. Scanner name, scanner model, serial number, firmware version and measuring ranges.
2. Scanner status indicators.
3. Parameterization tabs.
4. Visualization area.
5. Control buttons and notifications.

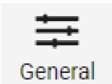
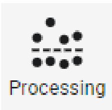
Area 1 contains the scanner name, scanner model, serial number, firmware version and measuring ranges. The scanner name can be changed by the user.

Area 2 contains the following indicators:

Group	Icon	Description
Mode		Web interface mode. Three modes are available: Engineer, Adjuster, Operator.
Ethernet		If the connection is established, the **Link** field and the connection speed value will be displayed. The **Required** field displays the recommended connection speed required for correct operation.
		This message appears when there are delays in the network during data transfer.
		If the connection with the scanner is lost (for example, when the scanner is restarted or the connection is broken), the web page will be displayed, but the connection status will be changed to **Disconnected**.
Temp		The processor temperature (**CPU, °C**) and the temperature inside the scanner body (**Internal, °C**). The processor temperature (**CPU**) and the internal temperature of the scanner (**Internal**) in °C. This information is used to assess the operating conditions of the scanner. Do not allow the temperature to rise to 90°C or more. The indication turns on when the temperature rises above 90°C, or if the temperature is below -15°C:  
Profiles		Displays the following parameters: the synchronization source (Icon), the current number of profiles per second (PPS) and the current format of the profile data (Format) sent by the scanner via UDP.

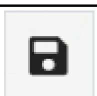
Group	Icon	Description
		Synchronization sources: • **Internal** - Synchronization by the internal generator of the scanner. • **External** - Synchronization by the external trigger. • **Soft** - Synchronization by the software request.
Counters		Displays the value of the profile counter (**Profile**), the value of the pulse counter of the encoder (**Pulse**), the direction of the encoder (**Direction**). On the right side is a button to reset the counters to zero.
Dump		The internal memory level for recording profiles and the recording control button (**Record**). Recording is only possible for calibrated profiles (**Data format > Profile**); otherwise, the recording start button will be disabled. The scale displays the set limit for the number of profiles available for recording.
Inputs		The status of the scanner inputs. Waveforms of digital signals at the inputs. Waveforms are only displayed for enabled inputs.

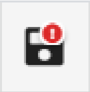
Area 3 provides access to the scanner settings and includes the following tabs:

Tab	Icon	Description
General		General scanner settings (CMOS sensor parameters, ROI parameters, laser control, data stream control).
Profile		Profile extraction settings.
Triggering		Settings of input channels of the scanner (triggering modes) and output channels for synchronizing the operation of several scanners.
Dump		Settings of the profiles accumulation in the internal memory of the scanner.
Smart		Access to the functions of mathematical processing of profiles, smart blocks of measurement of various geometrical and statistical quantities, the calculation graph.
Network		Network settings of the scanner.
System		Scanner system settings, including general information about the scanner, support for compatibility modes, firmware update, and the device's operation log (log file).

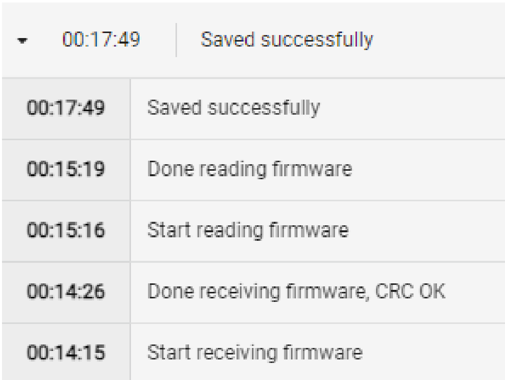
Area 4 is intended to quickly display the results. The controls for this area are described in par. 16.1.

Area 5 is located in the upper right corner and contains elements for selecting the language of the WEB interface, notifications from the scanner and control buttons.

Button	Name	Description
	**Language**	Responsible for selecting the display language for web interface elements. The following languages are supported: English, German, Spanish, French, Chinese, Korean, Russian.
	**Save configuration**	Save settings to the flash memory of the scanner.

Button	Name	Description
		The button with a red icon means that the settings are changed but not saved.
	**Load defaults**	Restore the factory settings. After restoring the factory settings, the scanner will reboot automatically.
	**Restart device**	Restart the scanner.

The notification area contains a drop-down list of important messages and events from the scanner:



14. Web interface operating modes

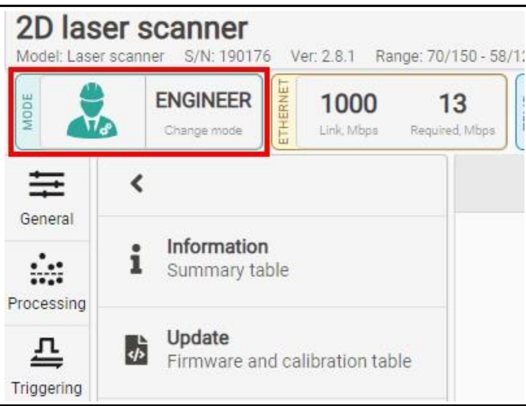
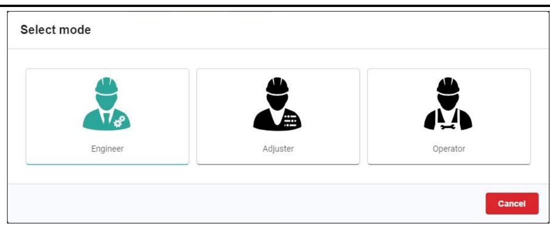
This chapter only applies to Smart scanners.

Smart scanners have three modes of the web interface:

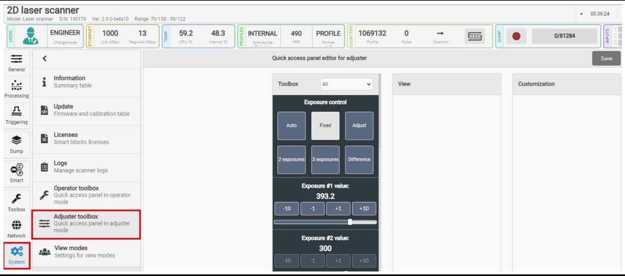
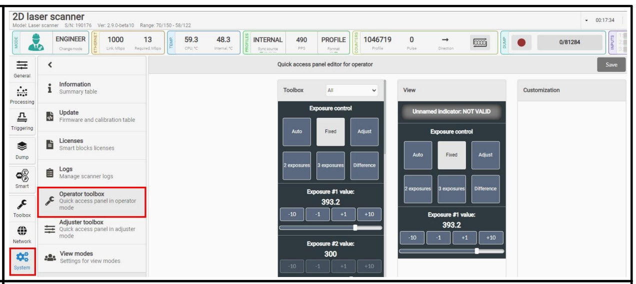
1) **Engineer**. Password access (if enabled). The Engineer has access to all scanner settings and can set up controls for the Adjuster and Operator.
2) **Adjuster**. Password access (if enabled). The Adjuster has access only to controls configured by the Engineer.
3) **Operator**. Password-free access. The Operator has access only to controls configured by the Engineer.

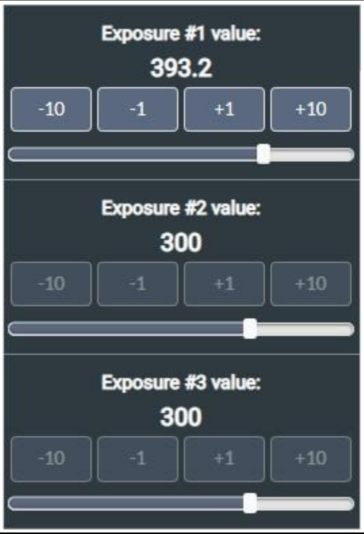
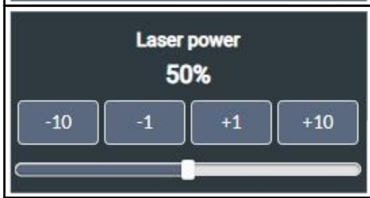
The procedure for setting the modes is described in Annex 7.

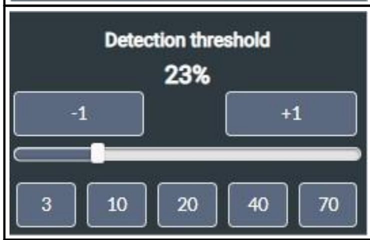
The Adjuster and Operator modes must be configured taking into account the specifics of the tasks solved using the scanner (similar to the HMI panels - human-machine interface). Switching between modes is done using the **Mode** panel:

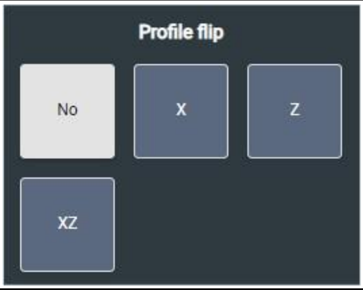
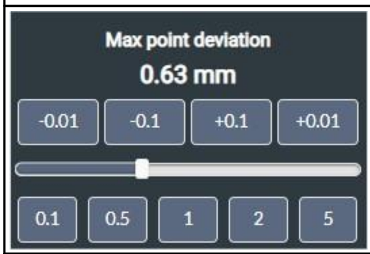
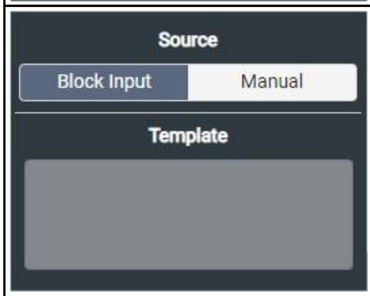
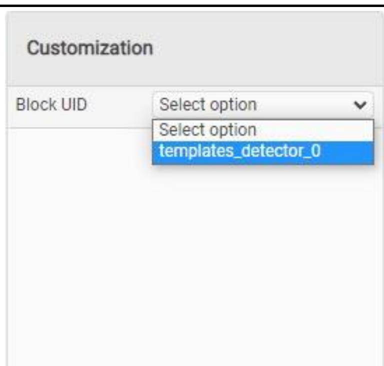
	
"Mode" panel	Mode selection

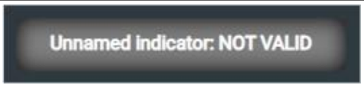
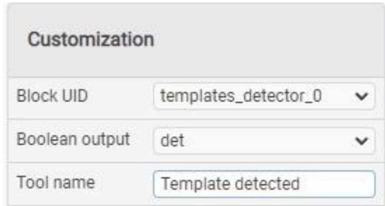
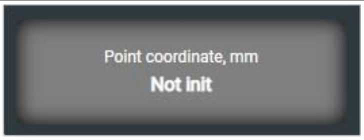
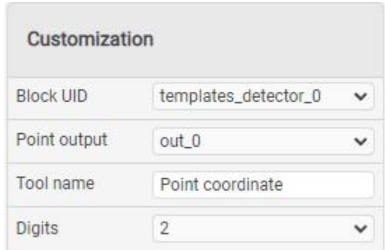
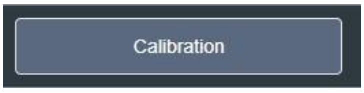
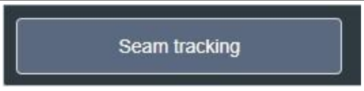
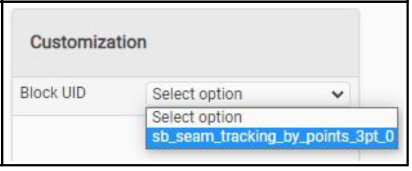
The Adjuster mode and the Operator mode are configured in the Engineer mode in the **Adjuster toolbox** and **Operator toolbox** sections, respectively:

	
"Adjuster toolbox" section	"Operator toolbox" section

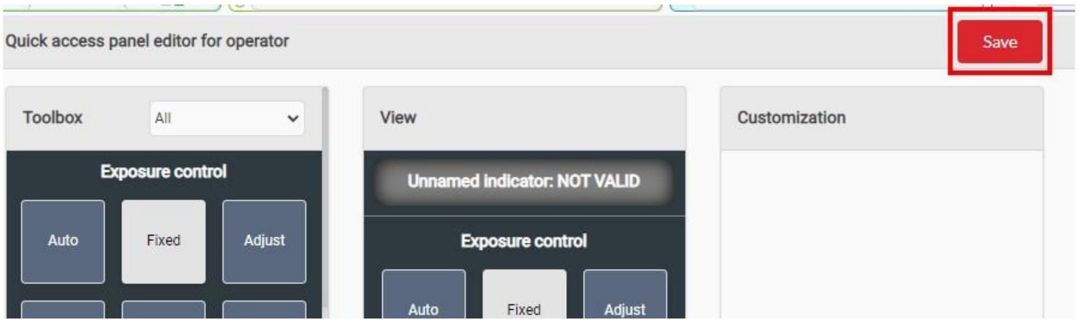
View	Description	Available settings
Image acquisition control		
	Selecting the exposure mode.	-
	Setting the exposure value (separately for each exposure).	-
	Setting the parameter value for the extended dynamic range (EDR) mode.	-
Laser control		
	Turning on/off the laser.	
	Setting the laser output power.	
ROI control		
	Turning on/off the ROI mode.	

View	Description	Available settings
	Selecting the ROI positioning mode.	
	Setting the number of points in the profile to detect the ROI position.	
	Setting the ROI size in the lines of the CMOS sensor.	
Profile preprocessing and postprocessing		
	Setting the signal cutoff threshold by amplitude.	
	Setting the peak selection mode for profile detection.	
	Setting the threshold for detecting profile points.	
	Setting the allowable peak width in pixels.	
	Setting the width of the median filter.	

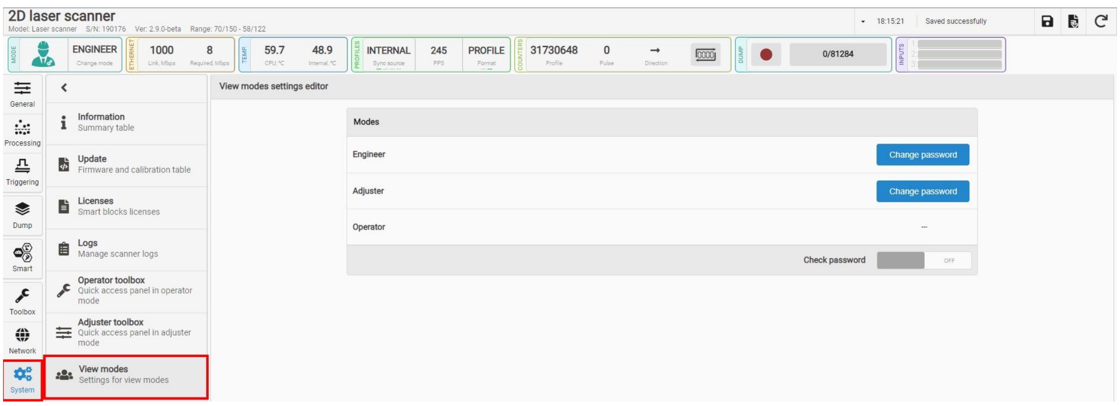
View	Description	Available settings
	Setting the width of the bilateral filter.	
	Selecting the profile display mode.	
Synchronization, physical inputs and outputs		
	Selecting the measurement synchronization source.	
	Timing diagrams of signals at the inputs.	
Profile approximation and smart blocks		
	Setting the maximum deviation of a point when splitting into lines.	
	Setting the maximum number of lines in the profile fragment.	
	Setting the parameters of the "templates detector" smart block. The block from the current graph is set as parameters.	

<table>
<tr><th>View</th><th>Description</th><th>Available settings</th></tr>
<tr><td></td><td>Status indicator. Shows the status of the “bool” type output (logical output) for the smart block specified in the parameters. The signature on the indicator can be changed.</td><td></td></tr>
<tr><td></td><td>Element for displaying point coordinates in 2D or 3D coordinate system. The signature on the indicator can be changed.</td><td></td></tr>
<tr><td></td><td>This button opens the calibration window of the "cst calibration" smart block.</td><td></td></tr>
<tr><td></td><td>This button opens the tracking window for the "3-pt tracking (by points)" and "3-pt tracking (by velocity) smart blocks.</td><td></td></tr>
</table>

After making changes, you need to save them by clicking the **Save** button:



Changing the password for access to the "Engineer" and "Adjuster" modes is done on the **View modes** tab:



If password access is enabled (the **Check password** switch is set to "ON"), then every time you change the interface mode:

- "Operator" > "Adjuster" - the password set by the Adjuster will be requested;
- "Operator" > "Engineer" - the password set by the Engineer will be requested;

- "Adjuster" > "Engineer" - the password set by the Engineer will be requested;
- in other cases, no password is requested.

15. Search for scanners on the network and connection

Enter the IP address of the scanner into the address bar of the web browser and press the **Enter** key. When the scanner is detected on the network, the browser will display its web page.

If all the settings are correct and the entered IP address is the IP address of the scanner, the **Ethernet** field will display **Link** and the current connection speed. The scanner is ready to operate.

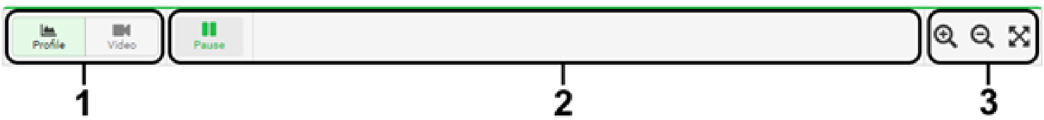
16. Results display area

In this area you can view:

- a calibrated profile (the profile in Cartesian coordinates of the scanner), or
- an uncalibrated profile extracted from the image, or
- a video stream from the CMOS sensor of the scanner with the overlay of the uncalibrated profile extracted from the image.

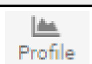
16.1. Controls

The controls are located at the top of the results display area:



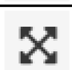
1 – display mode buttons;
2 – additional display options;
3 – zoom buttons.

Area 1 contains buttons that are intended to select the data display mode. Possible modes:

Display mode	Icon	Description
Profile		Displaying the profile on a 2D grid.
Video		Viewing the video stream from the CMOS sensor of the scanner.

The content of the controls in **Area 2** depends on the selected display mode and is described in section 16.

Area 3 contains the following buttons:

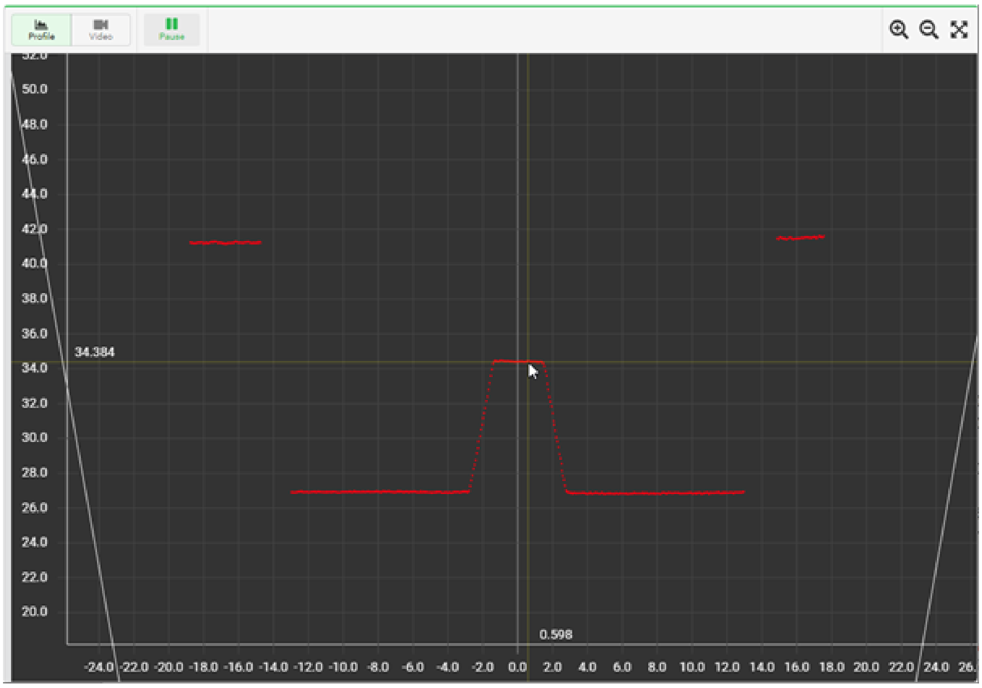
Icon	Description
	Zoom in.
	Zoom out.
	Reset zoom. NOTE: Returning the image to its original scale is also possible by double-clicking the left mouse button in the display area.

NOTE. Zooming in / out can also be done with the mouse wheel.

16.2. Display modes

16.2.1. Profile mode

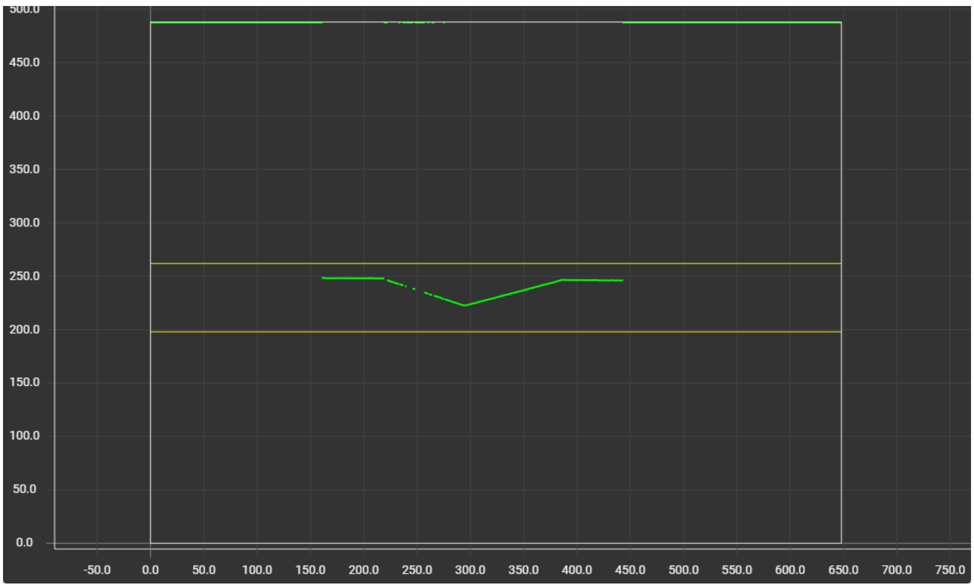
The **Profile** mode is intended to view a two-dimensional profile on the grid. The vertical axis corresponds to the Z coordinate of the scanner, the horizontal axis corresponds to the X coordinate of the scanner.



The profile is displayed in red, the measuring range of the scanner is displayed in white, the region of interest (ROI) is displayed in yellow (if ROI mode is enabled). When you hover the mouse over the selected area of the grid, a cursor appears indicating the position in the scanner coordinates. Moving an image is done with the mouse while holding down the right key.

Viewing the current profile in real time can be controlled by the **Pause** ⏸ / **Play** ▶ button, which is located in the area of additional display options.

In **Raw** mode (**General** tab > **Stream** section > **Data format**), an uncalibrated profile is displayed on the grid. In this case, the coordinate grid has a pixel dimension.



16.2.2. Video mode

The **Video** mode provides viewing of the video stream from the CMOS sensor of the scanner with overlapping of the detected profile on the image (in **Raw** mode).

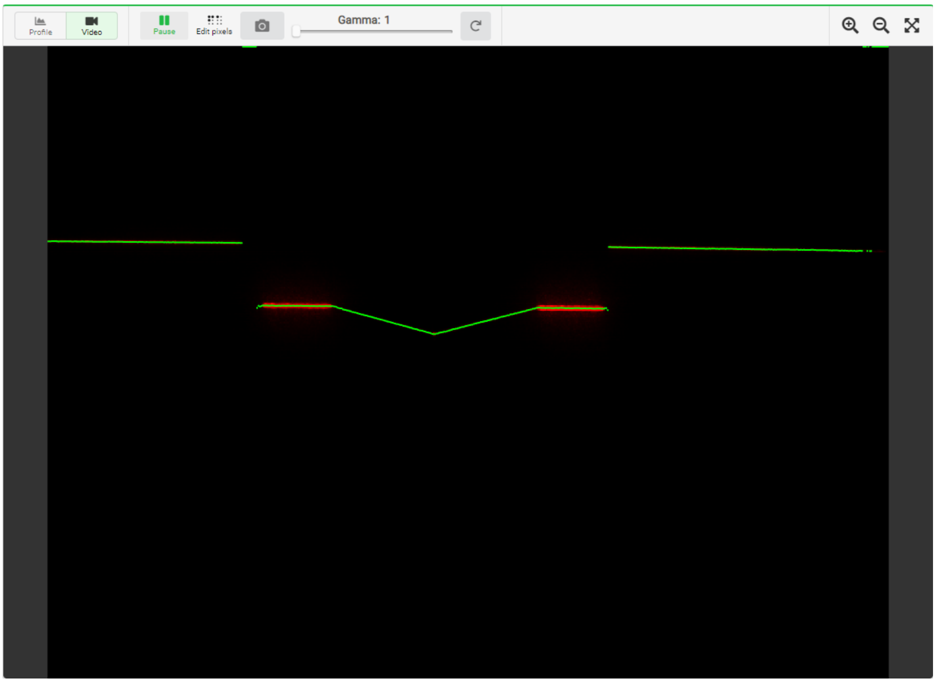
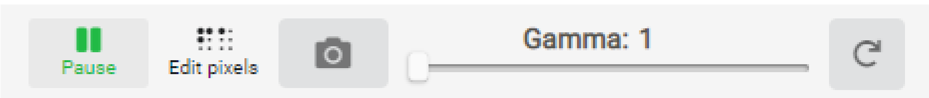


Image transfer speed is determined by computer performance (average value is about 15 frames/s).

The image is displayed on two screens. On a small screen, the yellow rectangle shows the position of the viewing area.

Green color indicates the points of the uncalibrated profile selected by the scanner from the image. When viewing the image in the calibrated profile mode, only the video signal is displayed.

The area of additional display options contains a button for starting / pausing the video stream, a button for editing defective pixels (**Edit pixels**), a button for saving a screenshot (i.e. saving the full image from the scanner image sensor regardless of the display scale), a slider for adjusting the gamma correction of the image, and a button for resetting the gamma correction to its original value.



Gamma correction is applied only to the displayed frame in the web interface and is intended to improve the visual visibility of low intensity areas.

The procedure for editing defective pixels is described in Annex 2.

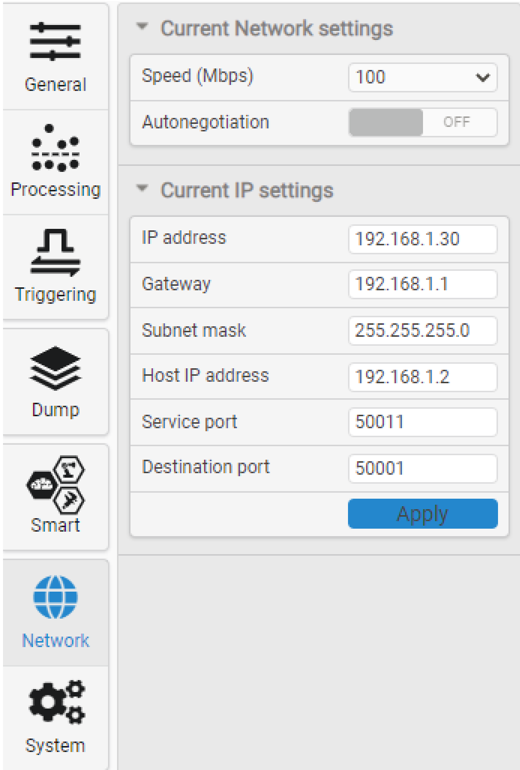
17. Setting parameters

To configure the scanner settings, go to the required tab and make changes.

All the settings, except network settings, are applied immediately. In order for the network settings to take effect, it is necessary to click the **Apply** button. All changes are made in RAM and will be lost when you restart the scanner. If you want to save parameters, write them to the nonvolatile memory of the scanner before restarting. Control buttons are located in the upper right corner of the window (see par. 13).

18. "Network" tab. Network parameters

To configure the network parameters of the scanner, go to the **Network** tab.



Current Network settings:

Parameter	Factory value	Description
Speed (Mbps)	-	Connection speed. Available modes: • 10 Mbps; • 100 Mbps; • 1000 Mbps.
Autonegotiation	ON	Automatic negotiation of network connection speed.

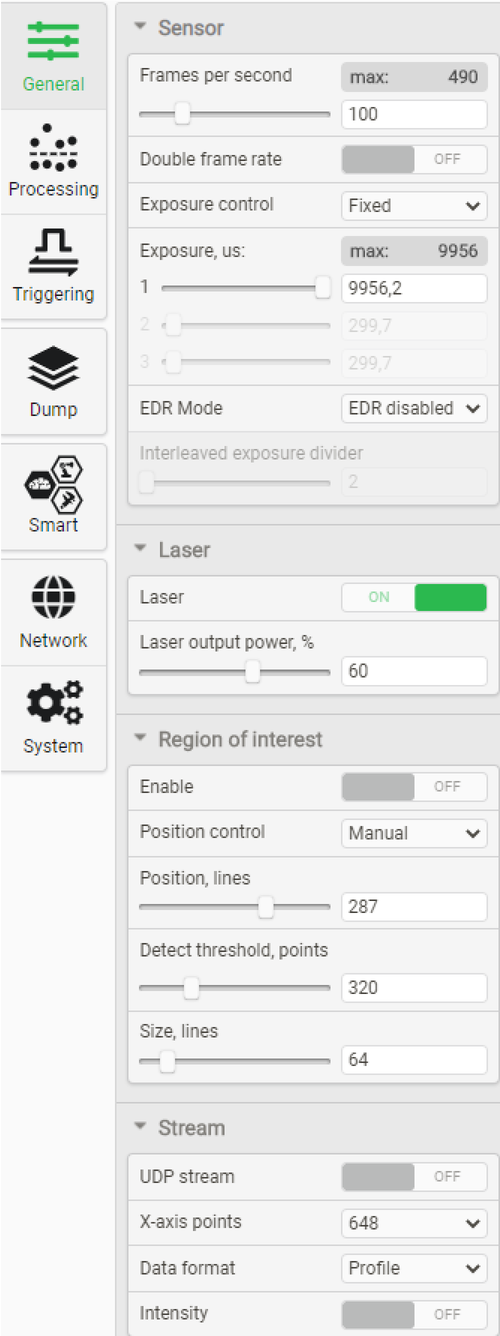
Current IP settings:

Parameter	Factory value	Description
IP address	192.168.1.30	IP address of the scanner.
Gateway	192.168.1.1	Gateway address.
Subnet mask	255.255.255.0	Network mask.
Host IP address	192.168.1.2	IP address of the PC (or other network device) receiving profiles.
Service port	50011	Scanner port number for the service protocol.
Destination port	50001	Port number of the PC (or other network device) receiving profiles, to which the scanner must send UDP packets with profiles.

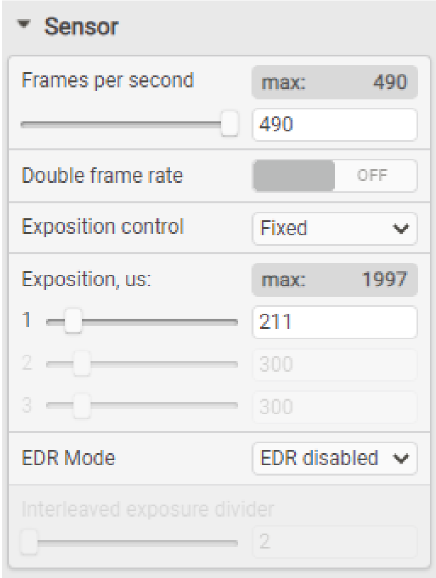
In order for the changes to take effect, it is necessary to click the **Apply** button.

19. "General" tab. General parameters

The menu may differ depending on the scanner series.



19.1. CMOS sensor parameters

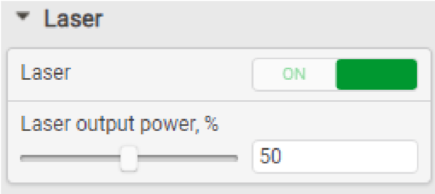


Parameters:

Parameter	Factory value	Description
Frames per second	520 - RF627Smart 520 - RF627BiSmart at least 4000 - RF628Smart at least 1000 - RF629Smart at least 4000 - RF6292Smart	The current number of profiles (frames) per second that the scanner processes and transmits.
Exposition control	Fixed	Exposure control mode. Possible options: • **Auto** – Automatic exposure adjustment. • **Fixed** – The exposure time is set by the user. • **Adjust** – The exposure time is automatically selected by the device when the "user_sensor_exposure Adjust" parameter is set to "TRUE". After completing the selection, the value of this parameter will be automatically changed to "FALSE". • **2 exposures** – Combining a profile from 2 frames with different exposure. • **3 exposures** – Combining a profile from 3 frames with different exposure. • **Difference** – (for Smart only) Removing background light (such as glare from the sun and other intense light sources). In this mode, the profile frequency is reduced by 2 times (PPS value) relative to the frame rate of the CMOS sensor (“Frames per second” parameter).
Exposition, us	3000 - RF627Smart 300 - RF628Smart 1500 - RF629Smart	The exposure time of the CMOS sensor (signal accumulation time) in microseconds, step - 1 μs. The minimum value is 3 μs, the maximum possible value depends on the frame rate, the ROI , and is limited to 1/FPS. Exposures numbered 2 and 3 (located under the **Exposition, us** parameter) are available only in the **2 exposures** and **3 exposures** modes, respectively (see the **Exposition control** parameter). **Note:** The laser automatically turns on during the exposure time only.

To configure the required parameter, use the slider, or enter the required value in the field and press **Enter** (valid for standard operation mode, as well as for DS and ROI modes). The maximum possible value of the parameter is shown next to the field.

19.2. Laser parameters



Parameters:

Parameter	Factory value	Description
Laser	ON	Turning on/off the laser.
Laser output power, %	10	Laser output power level. Range of values: 0...100%.

19.3. Image quality settings

The intensity of the reflected light entering the scanner depends on the properties of the surface of the object under control. In turn, the value of electric signal generated by the CMOS image sensor of the scanner depends on the time of accumulation of radiation (exposure time). Therefore, in order to obtain optimal signal, it is necessary to set optimal exposure time.

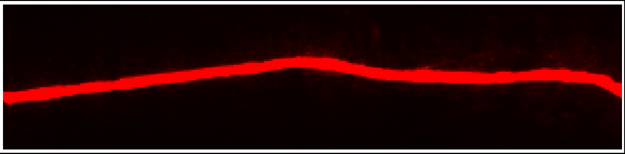
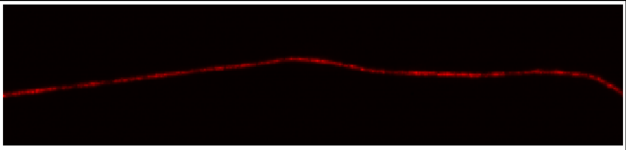
Since the exposure time cannot exceed the frame duration, it is necessary to set the required frame rate (**FPS** parameter) before setting the exposure time.

19.3.1. Exposure time and laser power

Exposure time and laser output power are set manually based on visual analysis of the quality of the image obtained from the image sensor, and on analysis of the quality of the resulting profile (see par. 16).

To set the exposure time, use the slider, or enter the required value into the field and press the **Enter** key. For convenience, you can select **Data format > Raw** (the **Stream** section of the **General** tab, see par. 19.5). In this case, the **Video** tab simultaneously displays a video signal and an extracted profile in the coordinate system of the CMOS sensor (uncalibrated data).

To enable the autoexposure mode, click **Autoexposure**. The scanner will automatically set the optimal exposure time.

Exposure time is too high	Exposure time is optimal
	

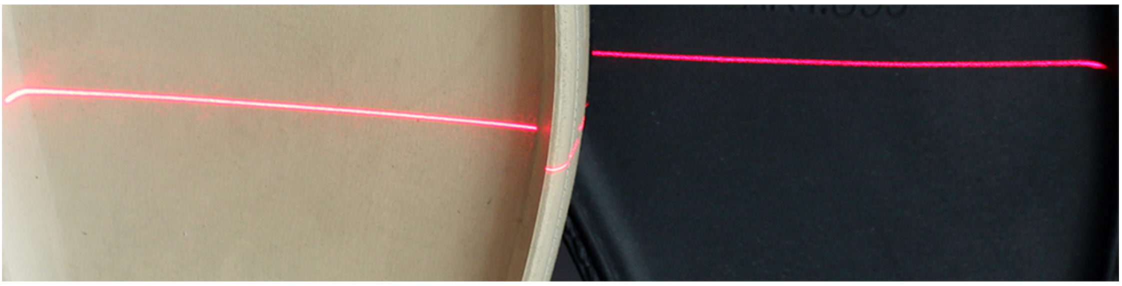
19.3.2. Multiple exposure mode

Multiple exposure mode (**Exposition control > 2 exposures, 3 exposures**) is intended to expand the dynamic range of the scanner. This mode is used when the objects (or the surfaces of one object) located in the field of view of the scanner have different reflective abilities.

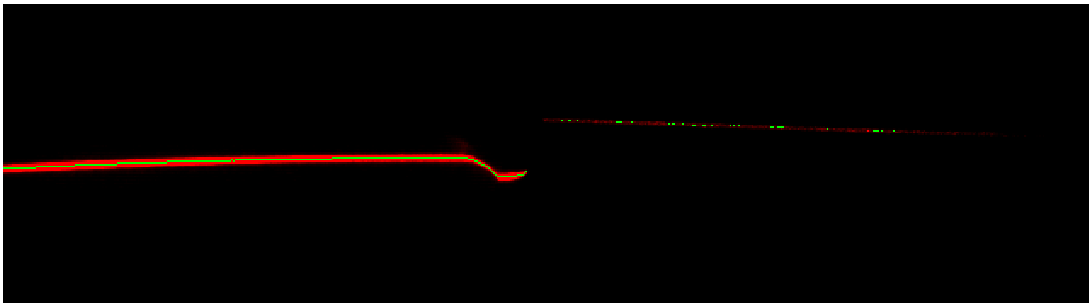
In the multiple exposure mode, the final profile is formed as a result of combining several (2 or 3) profiles obtained with different exposure times.

NOTE. In this mode, the frequency of profile output decreases in proportion to the number of exposures.

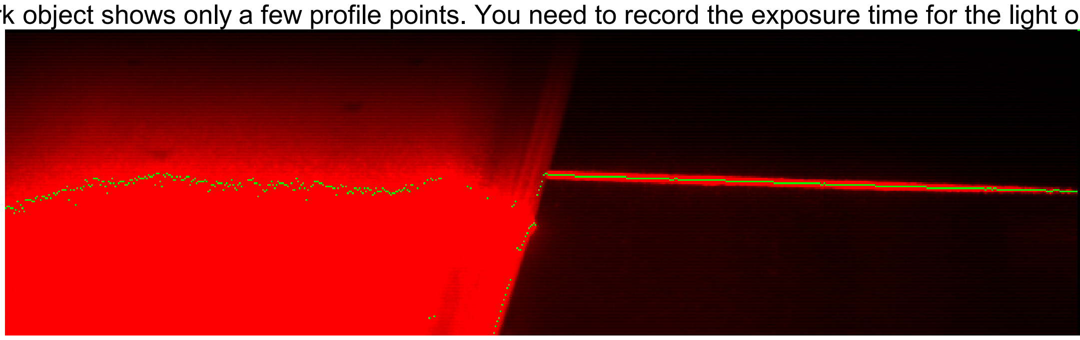
Example:



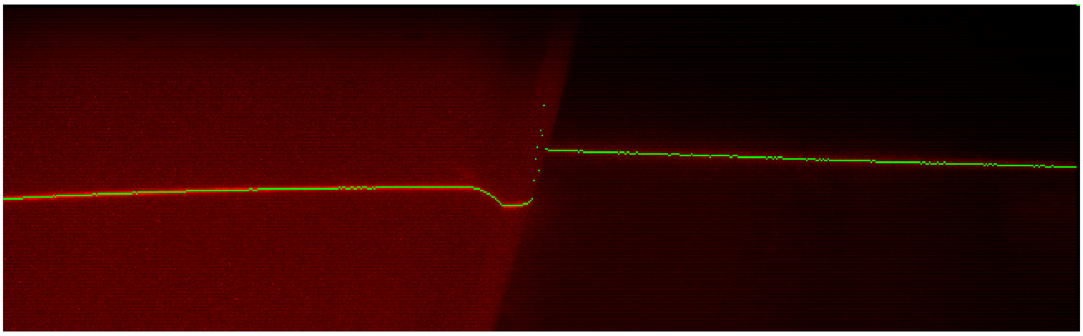
The scanner sees two objects: a light object and a dark object.



With a lower exposure time, the profile of a light object looks well-defined;
a dark object shows only a few profile points. You need to record the exposure time for the light object.



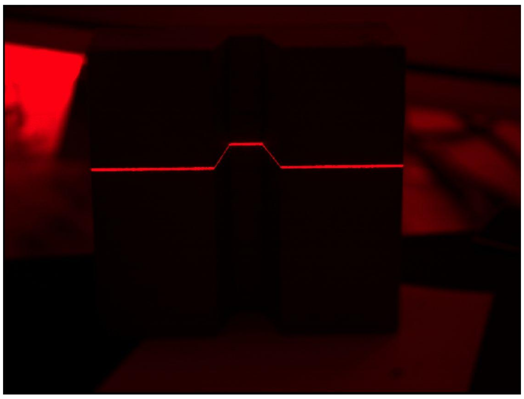
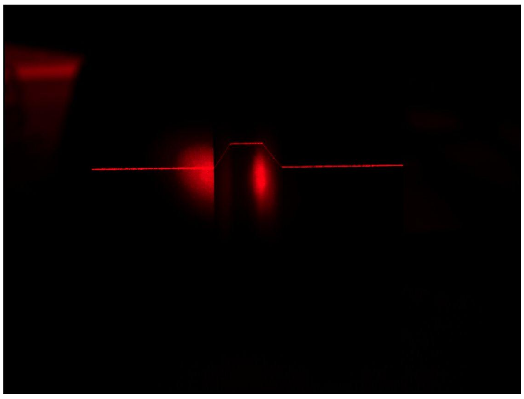
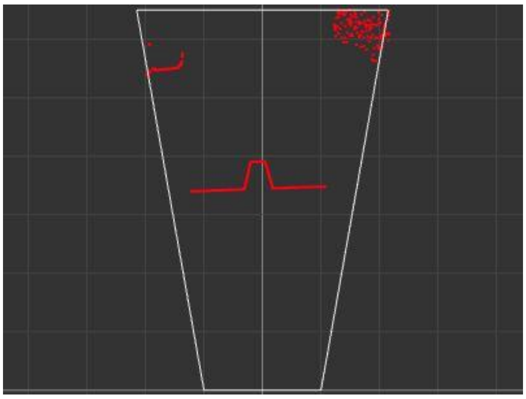
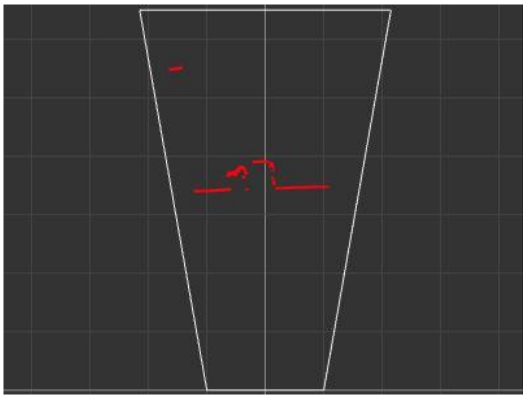
When increasing the exposure time, the profile of the dark object is well-defined, but the profile of the light one is not. You need to record the exposure time for the dark object.



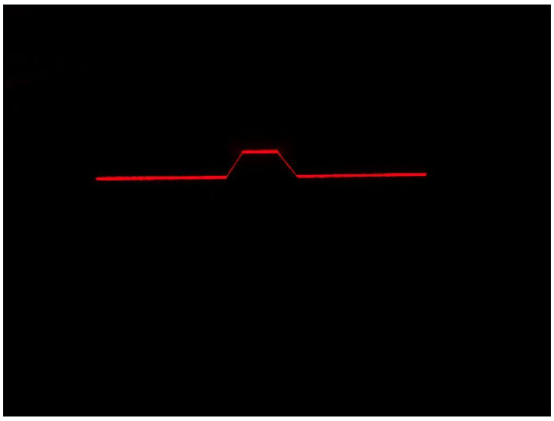
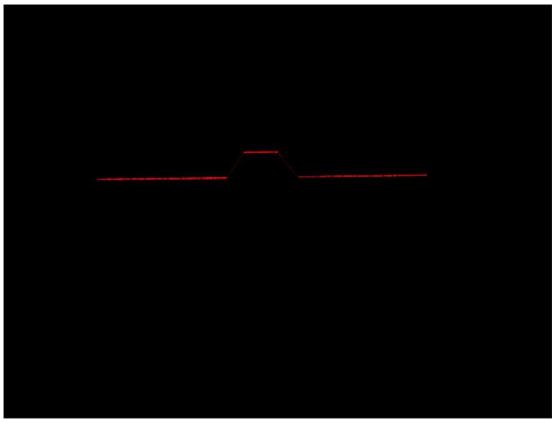
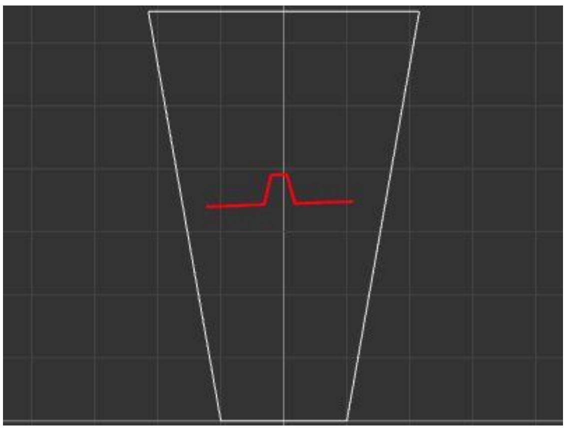
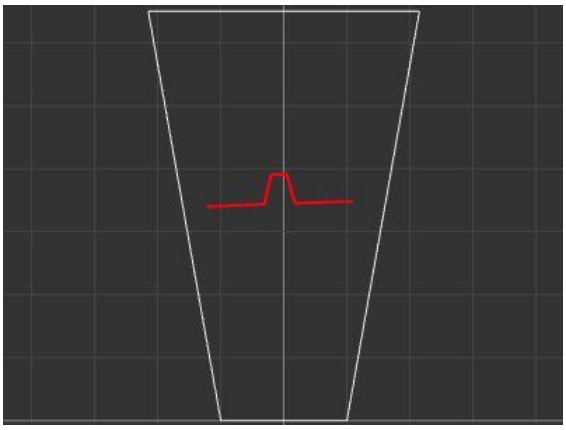
Select the multiple exposure mode and specify the recorded values of the exposure time. As a result, you get a high-quality image and a profile of a complex object.

19.3.3. Removing background light from extraneous light sources

This mode can only be used for Smart scanners. The mode is recommended when intense radiation reflected from extraneous sources (sun, lighting, etc.) enters the field of view of the scanner.

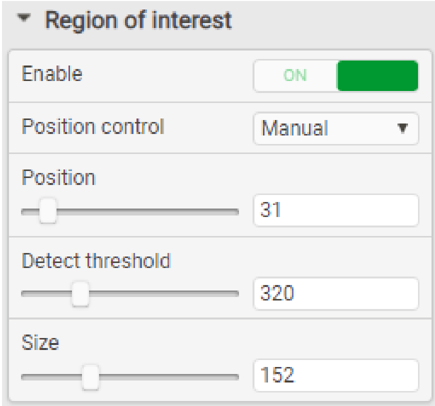
Examples of images and profiles with light and glare on the scanned surface.

Images and profiles with the "Difference" mode enabled.

This mode provides almost complete suppression of a stationary or not rapidly changing background relative to the frame rate (glare, reflections, superimposition of light spots on the scanned surface, etc.). In this mode, the profile rate (PPS) will be S of the frame rate.

19.4. ROI mode settings



The **ROI (region of interest)** parameters control the size and position of the CMOS sensor active area. By default, the active area covers the entire area of the sensor. Decreasing the active area size allows to increase the scanner speed due to decreasing of the image reading time. Resizing is possible in Z direction only and is performed in the coordinate system of the CMOS sensor.

Dependence of the operating frequency of the scanner on the size of the region of interest (typical values) for RF627Smart scanners:

ROI size, (% of MR) / (lines)	**Frequency**
100% / 536	524
75% / 400	683
51% / 272	955
24% / 128	1733
12% / 64	2715

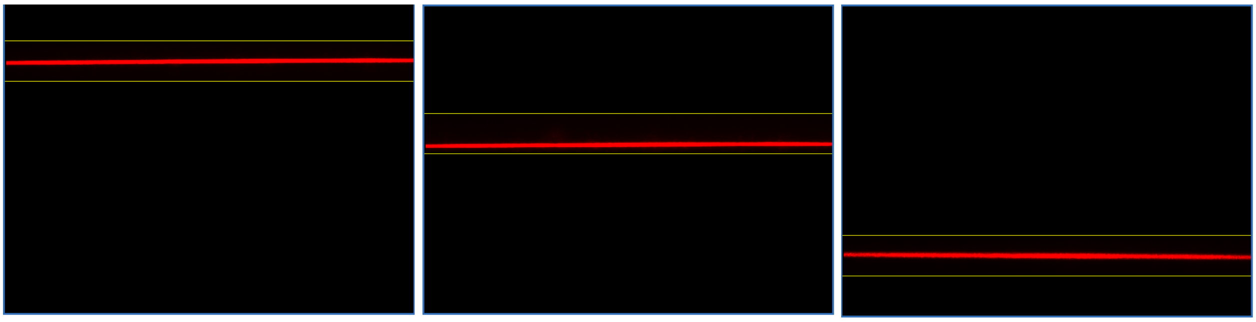
For scanners of other series, the proportions "ROI size - frequency" are close to those indicated above.

Parameters:

Parameter	Factory value	Description
Enable	OFF	Enable/disable ROI mode: • **ON** - enabled; • **OFF** - disabled. When ROI mode is enabled, the CMOS sensor processes a part of the active area set by the **Position** and **Size** parameters. The frequency of profiles increases inversely with the size of the region of interest (**Size**).
Position control	Fixed	ROI position control mode: • **Fixed** - Manual mode. The position of the region of interest is fixed and is determined by the **Position** parameter. The size of the region of interest is determined by the **Size** parameter. • **Auto** - Automatic position control keeping the profile in the center. When a profile is lost, the scanner switches to the operating mode without the region of interest (operates in the entire working range, the frame rate is reduced to standard). When a profile is detected, the scanner automatically switches to the region of interest with an increase in the frame rate. • **Auto-scan** - Automatic position control keeping the profile in the center. When a profile is lost, the scanner switches to the mode of scanning the working range by the region of interest (the

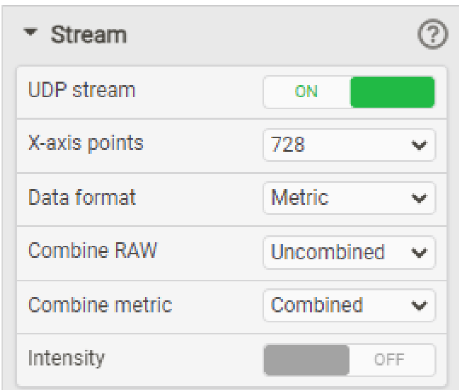
Parameter	Factory value	Description
		frame rate does not decrease). When a profile is detected, the scanner automatically switches to holding the profile in the region of interest.
Position	300	The position of the upper boundary of the region of interest in **FIXED** mode. This parameter is specified in lines. Valid values: from 0 to (488 - Size).
Detect threshold	324	This parameter is active in **AUTO** mode. It sets the number of points in the profile, which indicates that the profile is located within the region of interest. If the number of points in the region of interest is less than the specified value, the scanner automatically starts searching for the profile on the entire field of the CMOS sensor (the region of interest expands to the entire CMOS sensor with a corresponding change in speed). When the specified number of profile points is detected, the scanner automatically returns to the specified ROI size. Valid number of points: from 1 to 648. The size of the region of interest is determined by the **Size** parameter, the **Position** parameter is changed automatically.
Size	64	The size of the region of interest. This parameter is specified in lines. Valid values: from 24 to 480.

Example: Automatic displacement of the region of interest with keeping the profile within the set boundaries (yellow lines).



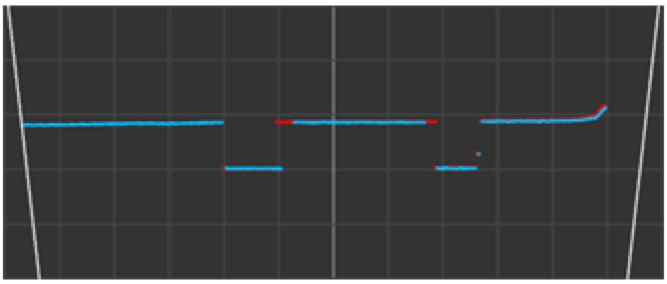
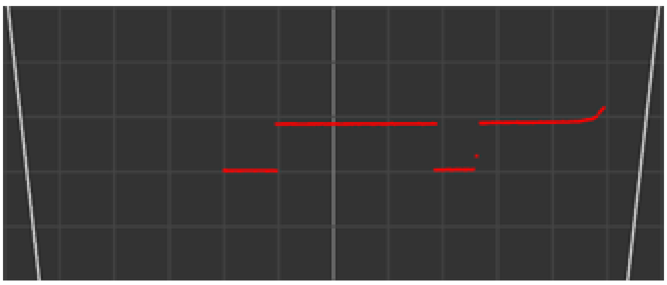
19.5. Data stream control

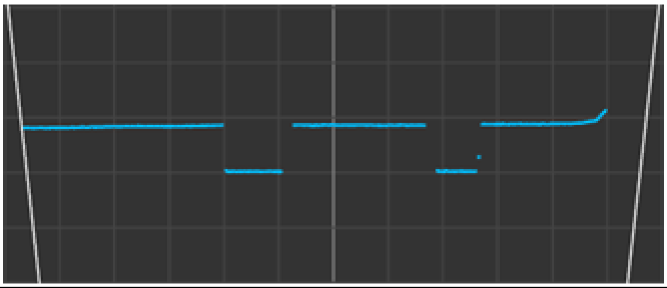
The **Stream** parameters control the data stream of the scanner, the resolution along the X coordinate, the current format of the scanner data, as well as the presence of the brightness values in the profile packet.



Parameters:

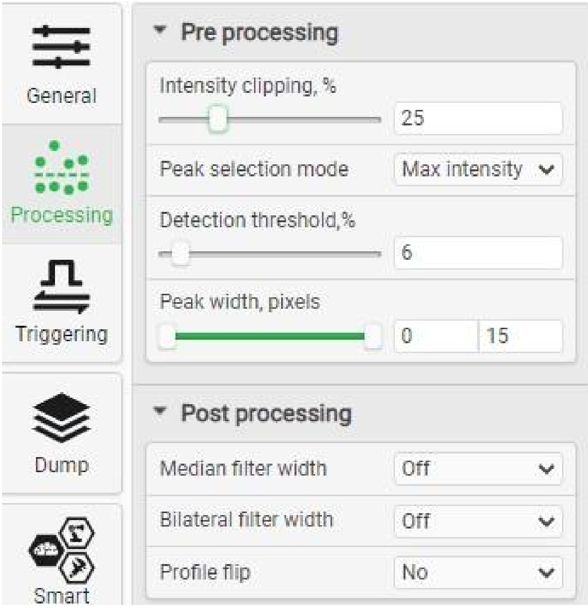
Parameter	Factory value	Description
UDP stream	ON	Enable/disable the UDP data stream.
X-axis points	RF627Smart - 1456 RF627BiSmart - 1456	The number of points along the X coordinate: • RF627Smart - 728 or 1456.

Parameter	Factory value	Description
	RF628Smart - 1280 RF629Smart, RF6292Smart - 1280	• RF627BiSmart - 728 or 1456. • RF628 - 640 or 1280. • RF629 and RF6292 - 1280 or 2560.
Data format	Calibrated profile	Data transfer formats: • **Calibrated profile** - transfer of calibrated data (profile in Cartesian coordinates of the measuring area). • **RAW profile** - transfer of uncalibrated data (profile in the coordinate system of the CMOS sensor). Obtaining a profile in this format allows you to visually match the profile and the image formed by the CMOS sensor. This format is used for debugging.
Intensity	OFF	Include the point intensity values in the profile packet: • **ON** - intensity values are included in the profile packet; • **OFF** - intensity values are not included in the profile packet. The data format description is given in the Developer Guide.
Combine RAW for dual camera scanners only	Uncombined	Method for combining profiles from two channels when using RAW format: • **Sensor 1** - only the profile received from the first channel is transmitted; • **Sensor 2** - only the profile received from the second channel is transmitted; • **Uncombined** - an uncombined profile from both channels is transmitted.
Combine metric for dual camera scanners only	Combined	Method for combining profiles from two channels when using the Metric format: • **Combined** - a combined profile from both channels is transmitted:  • **Sensor 1** - only the profile received from the first channel is transmitted:  • **Sensor 2** - only the profile received from the second channel is transmitted:

Parameter	Factory value	Description
		

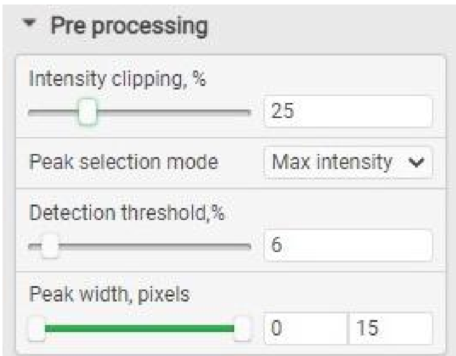
20. "Processing" tab. Profile extraction settings

The **Processing** tab contains parameters that control the procedure for extracting a profile from an image (**Pre processing** section) and filtering the points of the selected profile (**Post processing** section).



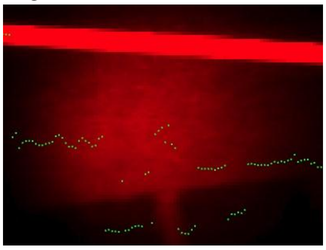
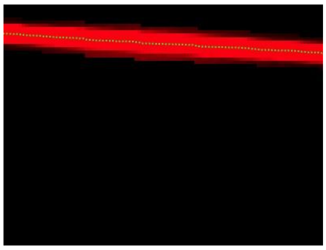
20.1. "Pre processing" section. Profile extraction parameters

The parameters of the **Pre processing** section define characteristics of the profile extraction algorithm.



Parameters:

Parameter	Factory value	Description
Intensity clipping, %	1	Signal clipping threshold. The frame is analyzed with a vertical window of 5 points. If the window has the intensity value greater than the threshold, the value of the central pixel of the window remains unchanged. If the value is less than the

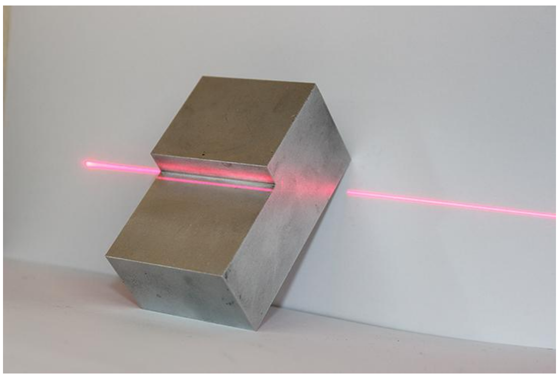
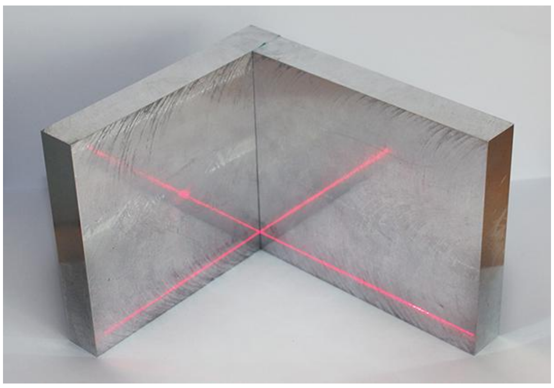
Parameter	Factory value	Description
		threshold, it is replaced with 0. Adjusting the parameter value makes it possible to reduce the influence of stray light of medium intensity (especially in modes when the "Peak selection mode" parameter is not set to "Max intensity"). Range of values: 0...100.   "Intensity clipping" = 4% "Intensity clipping" = 70%
Peak selection mode (see par. 20.1.1.)	Max intensity	The algorithm for determining the peak brightness in the image column to obtain the profile point. It is used to suppress false images resulting from multiple reflections on complex profiles. Modes: • Max intensity – Selecting the peak with the greatest brightness. • First – Selecting the first peak in the column above. • Last – Selecting the last peak in the column above. • #2…#4 – Selecting the peak in the column above with the corresponding number.
Detection threshold, %	10	This parameter determines the profile detection level. Increasing this parameter makes it possible to reduce the effect of image noise caused by, for example, ambient light. Range of values: 0...100%. If the value is 100%, the image is not processed.
Peak width, pixels	0...15	Peak brightness width in pixels. Range of values: 0...15.

20.1.1. "Peak selection mode" parameter

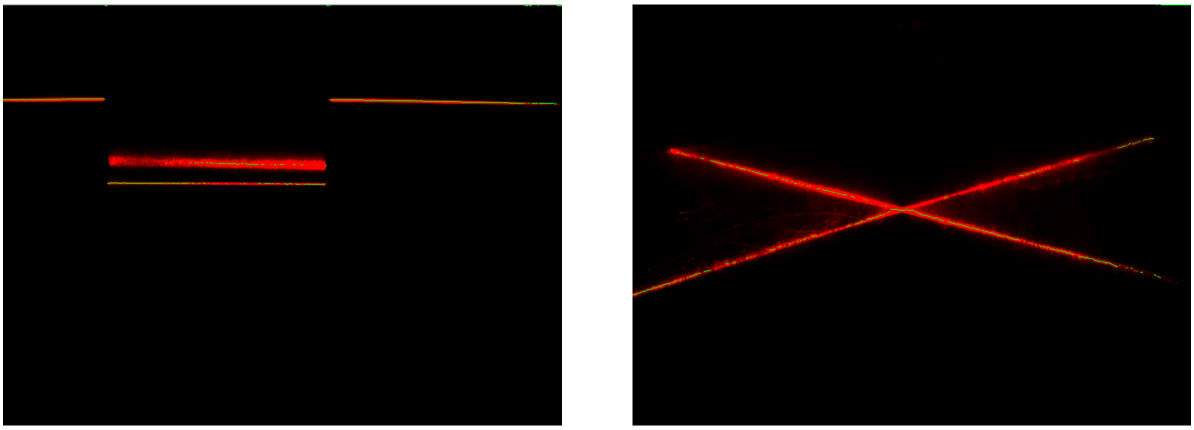
The **Peak selection mode** parameter defines the algorithm for detecting the brightness peak in the column of the CMOS sensor to obtain the profile point. Changing this parameter helps to correctly extract the profile in the case of laser beam re-reflections from the object surface or in the case of brightening from external sources of optical radiation.

The intensity of the re-reflected beam or brightening from external light sources can sometimes exceed the intensity of the laser line. In this case, you can use the modes with an indication of a more specific detection point.

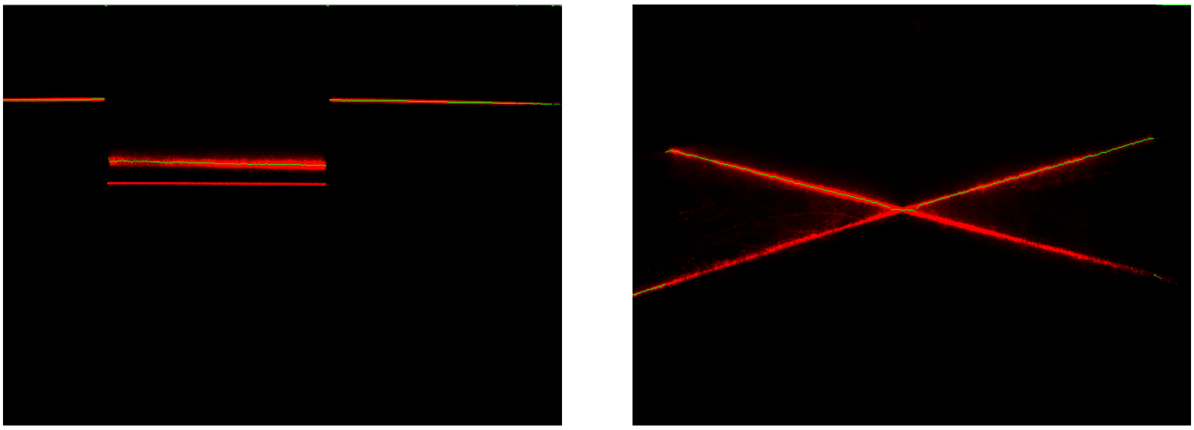
Example:

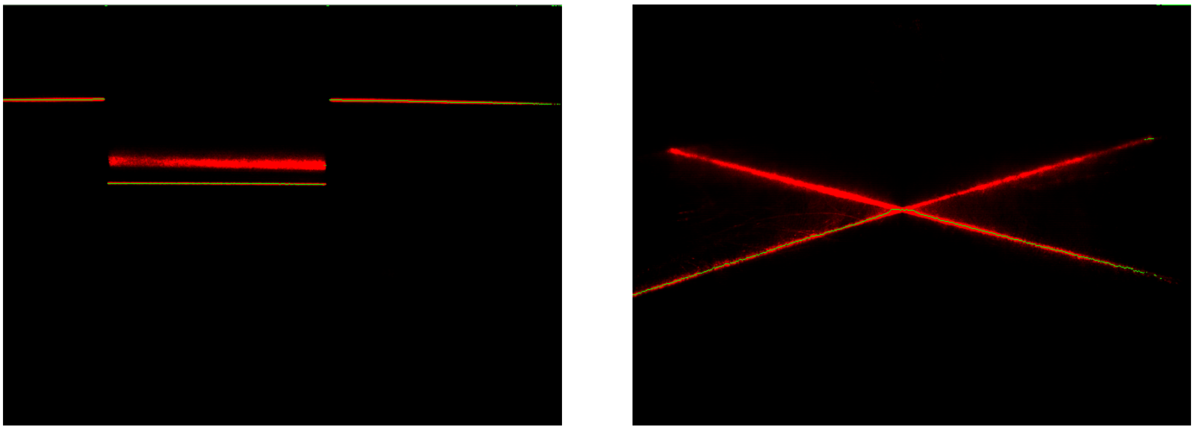
Re-reflections of a laser beam on the object having a complex profile.



The **Max Intensity** value determines the selection of the profile point based on the maximum brightness of the image in the CMOS sensor column. The brightness of the re-reflected signal may be greater than the brightness of the original signal. The scanner incorrectly selects the profile, placing it both on the initial laser line and on the re-reflexion.



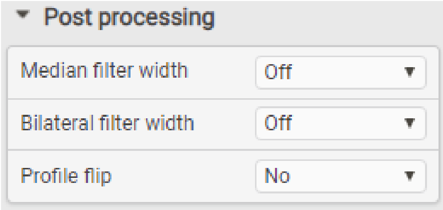
The **First** value determines the selection of the first peak in the CMOS sensor column.
The scanner selects a profile by a re-reflected signal.



The **Last** value determines the selection of the last peak in the CMOS sensor column.
The scanner selects a profile by a real signal.

20.2. "Post processing" section. Filtering

The parameters of the **Post processing** section define the operations performed directly on the profile points.

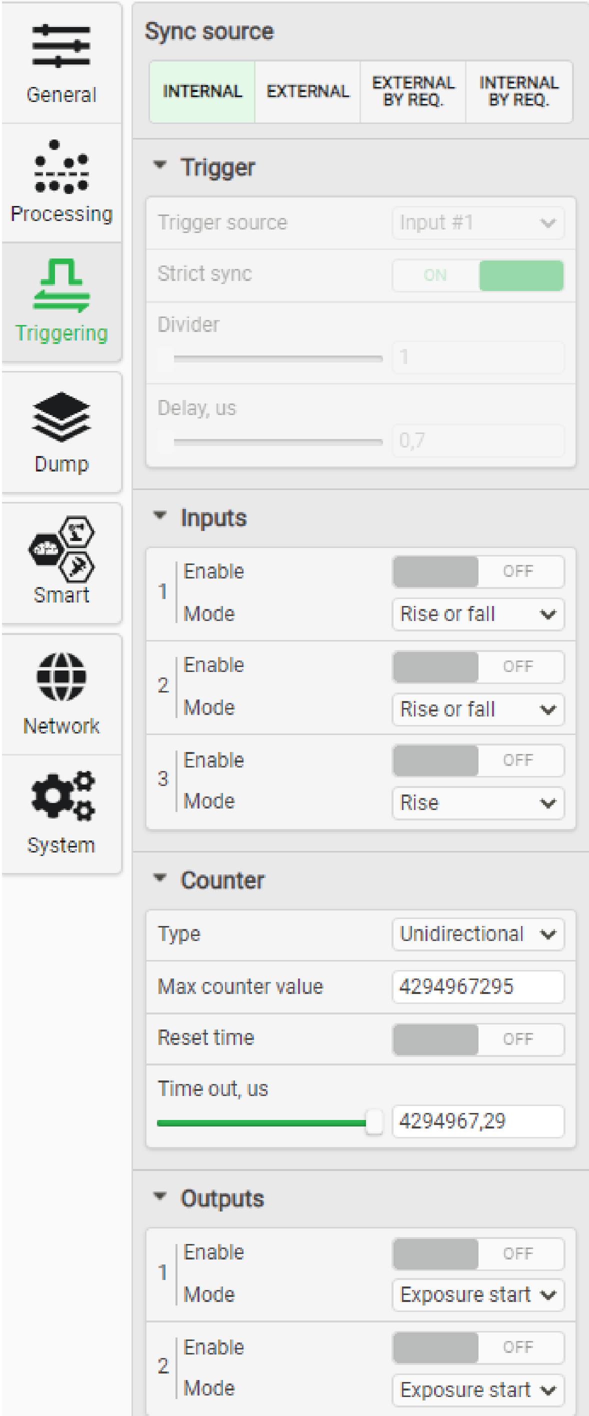


Parameters:

Parameter	Factory value	Description
Median filter width	OFF	The size (number of points) of the sliding window of the median filter. Valid values: OFF, 3, 5, 7, 9, 11, 13, 15.
Bilateral filter width	OFF	The size (number of points) of the sliding window of the bilateral smoothing filter. Valid values: OFF, 3, 5, 7, 9, 11, 13, 15. For more information about bilateral filtering, refer to: https://people.csail.mit.edu/sparis/bf_course/course_notes.pdf
Profile flip	NO	Flip a profile in direction of selected axes. Possible options: NO - no flip; X - flip along the X axis of the scanner; Z - flip along the Z axis of the scanner; XZ - flip along both axes.

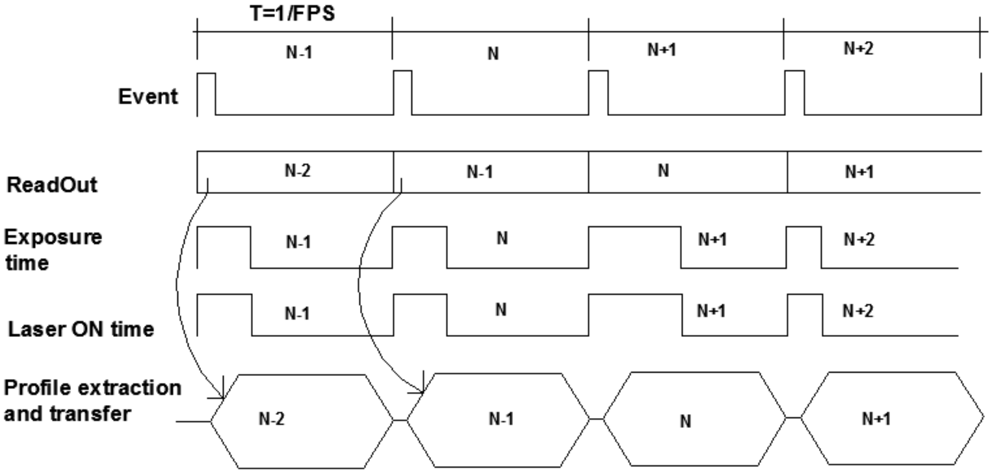
21. "Triggering" tab. Triggering modes

The **Triggering** tab is intended to configure the measurement (synchronization) triggering modes, as well as the scanner output channels.



21.1. Time cycle

Image capture, processing (profile extraction) and result transfer are performed in a pipeline mode. The pipeline mode is illustrated by the following diagram:



Description:

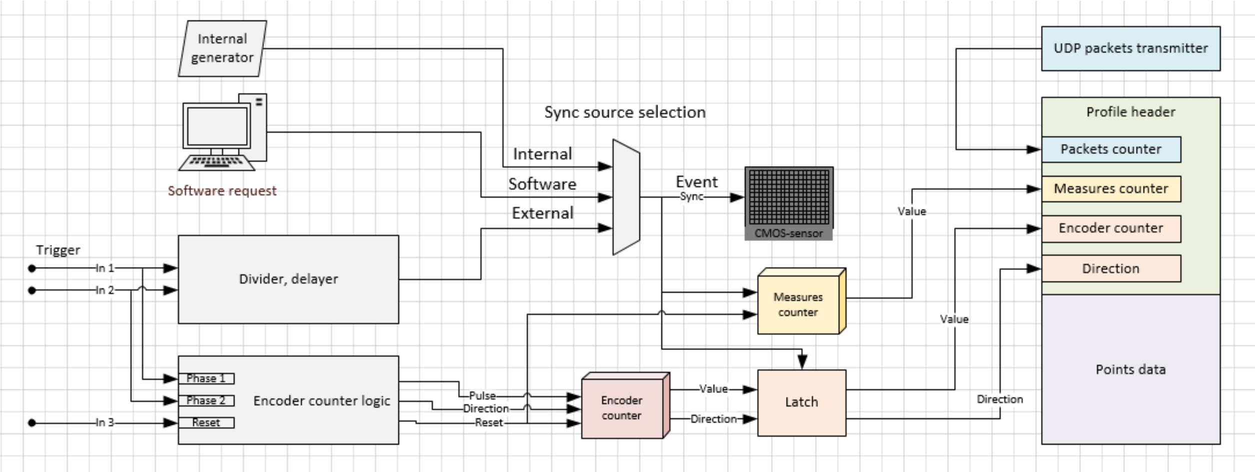
T	Frame (profile) period.
FPS	Frame (profile) rate.
N-1, N...	Frame (profile) numbers.
Event	Event that triggers the measurement cycle of obtaining a single frame (profile).
Exposure time	Exposure time of the image sensor.
Laser ON time	Time during which the laser is turned on.
Profile extraction and transfer	Time required to extract the profile and start its transfer.

To facilitate understanding of the synchronization of measurements, the concept of "synchronization event" is used. The synchronization event indicates the occurrence of a condition (internal or external signals at the inputs, or combinations thereof) under which the scanner starts the next cycle of exposure, calculation, data transfer.

The measuring cycle (start of measurement to take one profile) always begins with an event. Upon the occurrence of the event, the electronic shutter is opened and the laser is turned on, i.e. the CMOS sensor is exposed. After that, the frame is read and the profile is calculated, after which the profile is transmitted as a UDP packet. Simultaneously with the frame reading, the next frame is exposed (if the synchronization event has occurred).

21.2. Synchronization diagram

Block diagram of the synchronization module:



The source of synchronization events is selected by the **Sync Source selection** selector (multiplexer).

For external synchronization modes (**External**), the divider and sync delay (**Divider**, **delayer**) are available, as well as a special counter called the **Encoder counter**, which provides uni- or bi-directional pulse counting at inputs #1 and #2. In addition, the encoder counter can count the pulses of the internal high-speed generator (10 MHz) if the inputs are set to operate according to level, and not according to the rise or fall. The encoder counter value is latched at the moment of the synchronization event and transmitted along with the profile.

The synchronization scheme also includes the measurements counter (**Measures counter**), which counts the performed measurements.

All counters (except **Packets counter**) can be reset by an external or internal signal (for example: by input #3, by timer, by program request, etc.).

Notes:

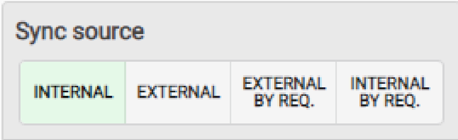
1. The maximum processed frequency at inputs #1, #2 and #3 is 10 MHz. If the event arriving rate is higher than the FPS, the measurement is started at the closest synchronization event after the end of the current cycle. The minimum allowable pulse duration is 40 ns. When using the input divider (**Divider**), the frequency of the events, triggering the measurement, equals to (input frequency) / (divider value).

2. The data packet with the profile coordinates, transmitted by the scanner (see the Developer Guide), contains information about the contents of several cyclic counters:

- System time counter for the beginning of each measurement.
- Input pulse counter (**Encoder counter**). This counter is incremented by the input signal (or input signals). The counter can work in reverse. The indication of the direction is transmitted in the data packet.
- Measurements counter (**Measures counter**). This counter is incremented by the synchronization event.
- Packets counter. This counter is incremented when sending a UDP packet with a profile.

21.3. Selecting a source of synchronization events

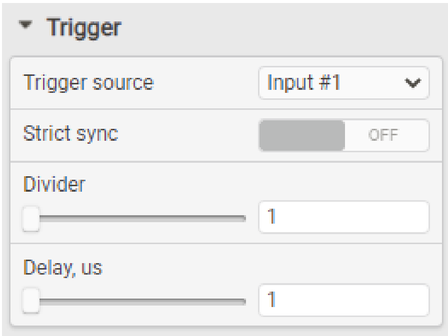
To select the source of synchronization events, use the **Sync source** section of the **Triggering** tab:



Sync source	Description
INTERNAL	Default source. Synchronization of profiles by the internal scanner generator. The events that trigger the measurement cycle follow at a frequency equal to the set FPS.
EXTERNAL	Synchronization of profiles by external trigger. A detailed description is given below.
EXTERNAL BY REQUEST	Waiting for a request (by service protocol) for profiles from third-party software. An external trigger is used for synchronization. If there is no request, the measuring cycle is not started.
INTERNAL BY REQUEST	Waiting for a request (by service protocol) for profiles from third-party software. An internal generator is used for synchronization. If there is no request, the measuring cycle is not started.

21.4. Synchronization by external trigger

The **Trigger** section is used to configure an external synchronization signal. This section is available only when the **External** source is selected. Input #1 and/or Input #2 are used to send a signal.



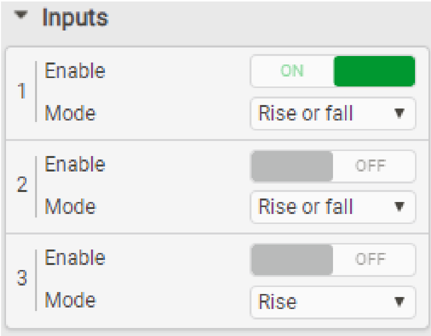
Parameters:

Parameter	Factory value	Description
Trigger source	Input #1	Selecting an input for an external sync signal or a combination of inputs. Available modes: • Input #1 – Synchronization by signal from Input #1. • Input #2 – Synchronization by signal from Input #2. • Input #1 OR #2 – Synchronization by any of the signals from both inputs. • Input #1 AND #2 – Synchronization by coincidence of signals on both inputs.
Strict sync	ON	Forced binding of the beginning of exposure to the synchronization signal. This mode is designed to eliminate the stroboscopic effect on the synchronization inputs.
Divider	1	Input pulse divider. The measuring cycle starts with an external synchronization signal, taking into account the **Divider** parameter. If the **Divider** is "1", it applies for each signal at the input. If the **Divider** is "2", it applies for every second signal at the input, etc. Setting the **Divider** parameter allows, for example, to match the

Parameter	Factory value	Description
		frequency of the input signals and the permissible frequency of the scanner.
Delay, us	OFF	Delay from the start of the synchronization signal to the synchronization event (the start of the measurement cycle).

21.4.1. Setting the inputs

The **Inputs** section contains the parameters of the inputs.



Parameters of Inputs #1 and #2:

Parameter	Factory value	Description
Enable	OFF	Enable / disable the input.
Mode	Rise or Fall	Input signal processing mode: • Rise or fall – Synchronization by rise or fall. • Rise – Synchronization by rise. • Fall – Synchronization by fall. • High level — Synchronization by high level. • Low level — Synchronization by low level.

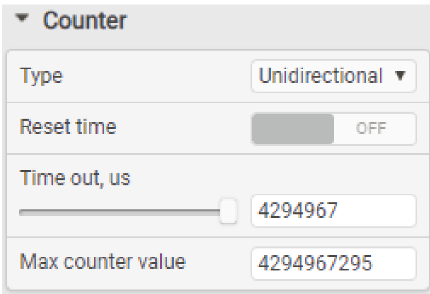
Input #3 of the scanner is designed to connect the reset signal of the measurement counters and the encoder.

Parameters:

Parameter	Factory value	Description
Enable	OFF	Enable / disable the input.
Mode	Rise	Reset signal processing mode: • Rise — Reset by front. • Fall — Reset by fall.

21.4.2. Setting the encoder counter

The **Counter** section contains the settings for the encoder counter.



Parameters:

Parameter	Factory value	Description
Type	Unidirectional	Counter type: • Unidirectional – Unidirectional counter (non-reversible).

Parameter	Factory value	Description
		• Bidirectional – Bidirectional counter (reversible).
Reset time	OFF	Resetting the counter after a specified time in the absence of synchronization events.
Time out, us	4294967 (maximum value)	Time interval for reset in the absence of synchronization events.
Max counter value	4294967295 (maximum value)	The maximum value of the counter. The counter will be reset when exceeding this value.

21.4.3. Examples

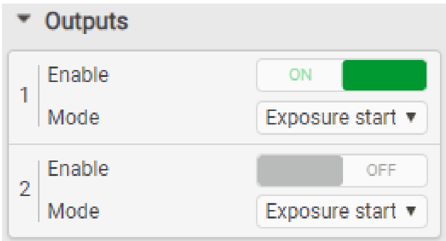
Examples of trigger settings:

#	Event source	How it works	Options	How to install
1	Internal generator.	Profiles are transmitted continuously at the set frame rate (FPS). Each measurement starts with an internal generator.		• Set the required frame rate. • Select the **Internal** source.
2	Software request.	Each measurement starts with receiving the software request.		• Select the **Software** source. • See the Developer Guide.
3	External trigger. Triggering a single measurement.	Each measurement starts with receiving the trigger signal at input #1, taking into account the set parameters.	• Triggering the measurement on the rise of the input pulse. • Triggering the measurement on the fall of the pulse. • Triggering the measurement on the rise and fall of the pulse. • Delay. • Divider.	• Connect the source to input #1 and enable the output. • In the **Inputs** section, select the required mode. • If necessary, set the **Delay** parameter. • If necessary, set the **Divider** parameter.
4	Encoder, one phase.	Same as #3.	Same as #3.	Same as #3.
5	Encoder, one phase and "0" mark.	Same as #3. The measurement counter is reset on phase Z.	Same as #3.	• Same as #3. • Connect phase Z to input #3. • Enable input #3 and select the mode.
6	Encoder, two phases.	Each measurement starts with receiving the quadrature encoder signals (multiplication by 4) at inputs #1 and #2, taking into account the set division ratio. The direction of movement is controlled (or is not controlled), the direction indication is transmitted (or is not transmitted) in the data packet.	• Divider. • Reversible / non-reversible counting.	• Connect phase A to input #1, enable the input. • Connect phase B to input #2, enable the input. • Select the operating mode (**Mode**) for both inputs: Rise or Fall. • If necessary, set the **Divider** parameter. • Select the **Counter type**: Bidirectional or Unidirectional.
7	Encoder, two phases and "0" mark.	Same as #6. The measurement counter is reset on phase Z.	Same as #6.	• Same as #6. • Connect phase Z to input #3. • Enable input #3 and select the operating mode (**Mode**).
8	Step/Dir signal (Step/Direction).	Each measurement starts with receiving the Step signal at input #1,	• Triggering the measurement on	• Connect the Step signal to input #1.

#	Event source	How it works	Options	How to install
		taking into account the set division ratio. The direction indication (Dir) is transmitted in the data packet.	the rise of the pulse. • Triggering the measurement on the fall of the pulse. • Divider.	• Connect the Dir signal to input #2 • If necessary, set the **Divider** parameter.
9	External trigger. Starting a series of measurements with the internal generator.	The series of measurements with the set frame rate (FPS) starts with receiving a signal at input # 1. The series of measurements stops when the signal level changes.	• Triggering the series of measurements at a high level of the pulse. • Triggering the series of measurements at a low level of the pulse.	• Connect the source to input #1 • Set the required value: **High Level** or **Low Level**.

21.5. Setting the outputs

The **Outputs** section contains the parameters of the outputs.



Parameters:

Parameter	Factory value	Description
Enable	OFF	Enable / disable the output.
Mode	Exposure start	Output signal generation mode: • Exposure start – Formation of the output pulse with a duration of 1 µs upon the event that triggers the measurement cycle. • Exposure time – Formation of the output signal that coincides with the **Exposure time** signal on the timing diagram. • In1 repeater – Duplication of the signal from input #1 to the output. • In2 repeater – Duplication of the signal from input #2 to the output. • In3 repeater – Duplication of the signal from input #3 to the output. **Note:** The delay of the output signal in relation to the duplicated signals is about 50 ns.

22. "Triggering settings" tab. Synchronization of multiple scanners

Where measurements are made by several scanners, it is often necessary to ensure **synchronous** measurements, in order, for example, to combine profiles obtained from different parts of the moving object into a single profile.

When installing scanners in a line or around an object or opposite each other, it becomes necessary to ensure **asynchronous** measurements in order to eliminate the mutual influence of laser beams on each other.

To synchronize the operation of multiple scanners, the OUT output of one of the scanners is used. The Rise of the scanner output signal always corresponds to the moment of switching on the laser of the scanner (the beginning of the integration time), the signal Fall corresponds to the moment of switching off the laser (the end of the integration time).

22.1. Synchronous measurements

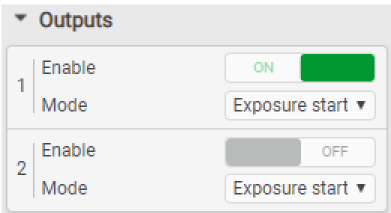
There are two options to connect the scanners for synchronous measurements.

Option 1.

All scanners in the system are configured to operate in one of eight modes, #2...#9 (mode #1 is not used). The event source is connected simultaneously (in parallel) to all scanners.

Option 2.

- One of the scanners (hereinafter - Master) is configured to operate in the required mode, #1...#9.
- The OUT Master output is initialized.

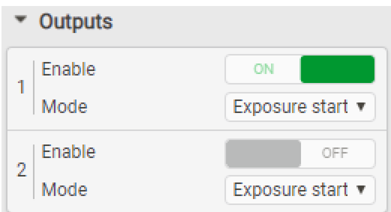


- The other scanners (Slave) are switched to mode #3, the operating mode is "Rise".
- The Master output is connected to Input #1 of all Slave scanners.

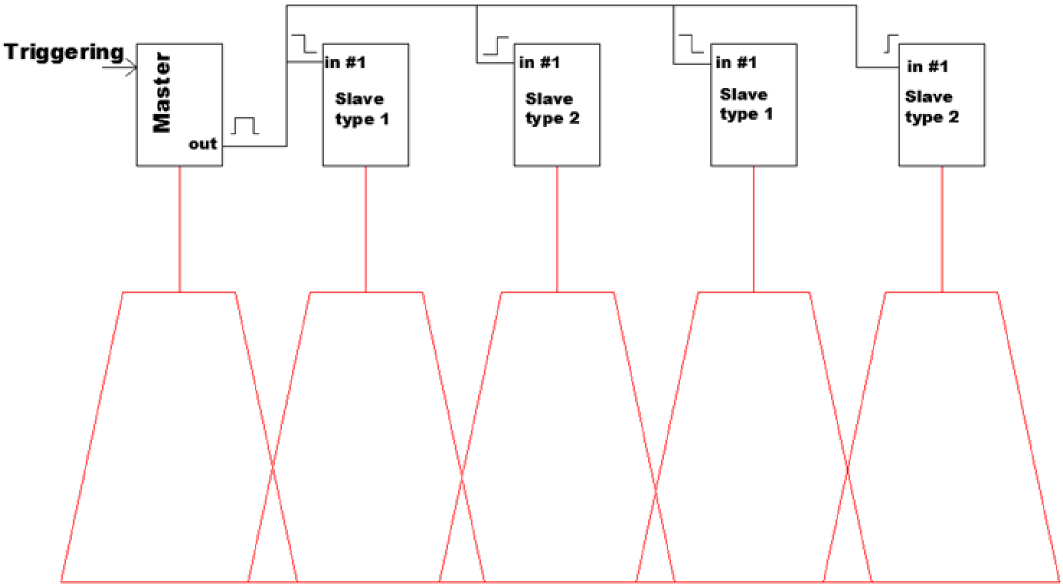
22.2. Asynchronous measurements

To perform asynchronous measurements, the scanners are connected as follows:

- One of the scanners (hereinafter - Master) is configured in the required mode, #1...#9.
- The other scanners (Slave type 1 and Slave type 2) are switched to mode #1.
- For nearby scanners (Slave type 1 and Slave type 2), the following parameters are set: **Mode** - Fall and **Mode** - Rise.
- The OUT Master output is initialized.



- The Master output is connected to Input #1 of all Slave scanners.

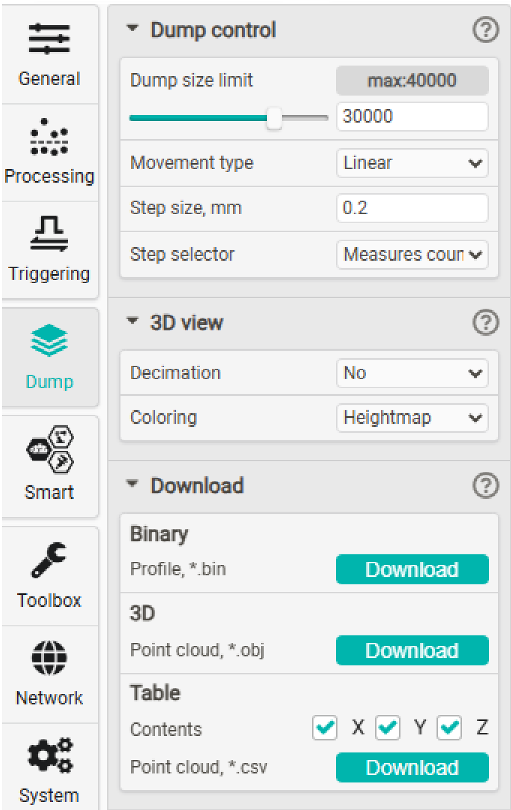


As a result, the lasers of the scanners of the "Master + Slave type 2" group and the "Slave type 1" group will alternately turn on.

NOTE: The total accumulation time of the Slave type 1 and Slave Type 2 scanners must not exceed the measuring cycle time = 1/FPS.

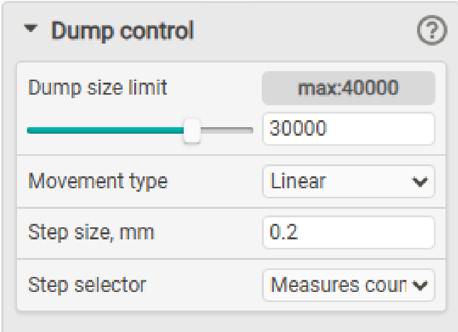
23. "Dump" tab. Accumulated profiles parameters

The **Dump** section contains parameters for working with accumulated profiles.



23.1. "Dump control" section. Building 3D models

The parameters of the **Dump control** section determine the parameters for building 3D models.

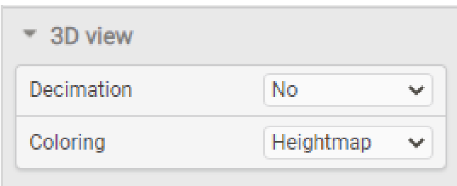


Parameters:

Parameter	Factory value	Description
Dump size limit	40000	Determines the maximum number of profiles allowed to be recorded to the internal memory. When the set limit is reached, the recording process automatically stops.
Movement type	Linear	Type of mechanical movement system used for obtaining the point cloud: • **Linear** – Linear movement system. The scanner (object) moves along a straight path; • **Radial** – Angular movement system. The scanner doesn't move. The scanned object rotates around its own axis. The axis of rotation of the object coincides with the **Xemr** line of the scanner range. This mode is used to receive the point clouds of rotation bodies.
Step size	0	Step size between measurements (in millimeters for the **Linear** system, and in degrees for the **Radial** system).
Step selector	System Time	Selector, which is used to build the point cloud. The step value is multiplied by the value of the parameter selected by the selector. • **System Time** – Time stamp in the profile. • **Step counter** – Encoder counter. • **Measurement counter** – Internal measurement counter.

23.2. "3D view" section. 3D model display parameters

The parameters of the **3D view** section determine the peculiarities of displaying the 3D model contained in the dump.



Parameters:

Parameter	Factory value	Description
Decimation	No	Decimation of profiles for displaying. Used to reduce the load on the computer's GPU. If it is necessary to display the entire set of profiles from the dump (80000), the number of displayed points reaches 103680000, which significantly slows down the interface. To eliminate this problem, it is recommended to decimate the dump when rendering a 3D model.

Parameter	Factory value	Description
		IMPORTANT: This parameter does not affect dump export.
Coloring	Heightmap	Profile points coloring mode. **Heightmap** - The color of a point is determined by its height. **Intensity** - Grayscale. The brightness of a point is determined by the intensity of the radiation reflected from the surface (to use this mode, it is necessary to enable the transmission of intensity: **General > Stream > Intensity = ON**).

23.3. "Download" section. Downloading profiles

This section is intended for saving profiles in various formats. For more details, go to par. 23.4.3 "Export of accumulated profiles".

23.4. Operations with profiles

23.4.1. Accumulation of profiles in internal memory of the scanner

To start recording profiles to the scanner's memory, click the button  on the **Dump** indicator. After that, each profile received by the scanner will be saved in its internal memory. The maximum number of profiles for recording is 80000. Only calibrated profiles can be recorded (**Data format > Profile**). For the **Raw** format (uncalibrated profile) the start button will be inactive. During recording, you cannot change the data format and the **Stream** section will not be available.

NOTE. The accumulation of profiles is carried out in accordance with the selected Triggering mode (see par. 21).

23.4.2. Viewing accumulated profiles

To view the accumulated profiles, open the **Dump** tab by clicking the corresponding button on the left panel:

To view the accumulated profiles, it is necessary to select the **Dump** source in the data source area.

In **Profile** mode, the selected profile from those accumulated in the internal memory will be displayed.

In **3D** mode, the accumulated profiles in the form of a three-dimensional point cloud will be displayed on the three-dimensional scene. You must first configure the display settings in the **Dump control** section (see par. 23.1):

- Select the type of displacement system when receiving a point cloud (**Movement type**).
- Specify the step between measurements (linear in mm for the **Linear** type, and angular in degrees for the **Radial** type).
- Choose the selector, which is used to build a point cloud (the **Measurement** and **Step** counters, or the **System time** profile time stamp). The step value is multiplied by the value of the parameter selected by the selector.

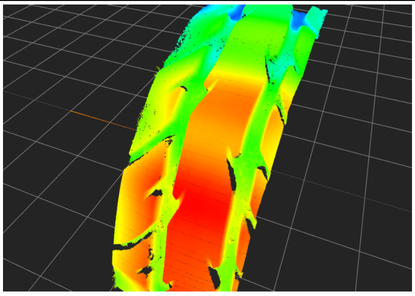
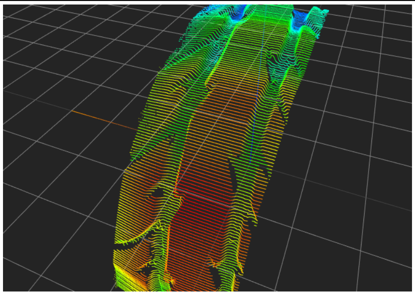
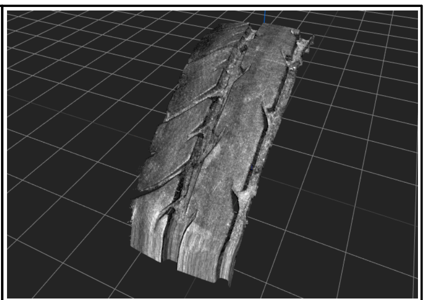
After configuring the display parameters, it is necessary to click the refresh button . After that, the data will be downloaded from the scanner and a point cloud will appear.

After changing any parameters in the **Dump control** section, you must click the refresh button  to redraw a point cloud with new parameters.

NOTE. To view a three-dimensional point cloud, the PC must have an appropriate video card. To view a point cloud on weak computers, adjust the decimation of the point cloud. To do this, select the appropriate coefficient in the **Decimation** drop-down list.

To view a point cloud with coloring by intensity, you must select the **Intensity** mode for the **Coloring** parameter.

NOTE. Coloring by intensity is possible only if, during recording, the intensity values were included in the profile packet (see par. 19.5). Otherwise, the intensity of all points will be zero (black color).

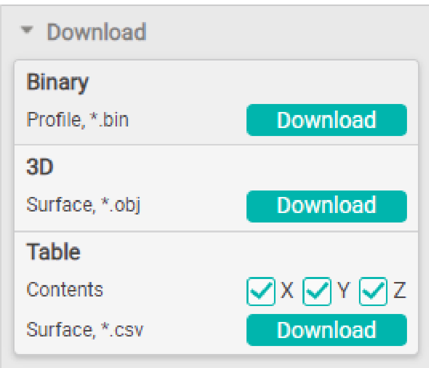
		
Point cloud without decimation and with coloring by height	Point cloud (decimation coefficient is set to 5)	Point cloud with coloring by intensity

Use the left mouse button to rotate the camera in the 3D scene, and the right mouse button to move the scene in the horizontal plane. Zooming is done with the mouse wheel.

23.4.3. Export of accumulated profiles

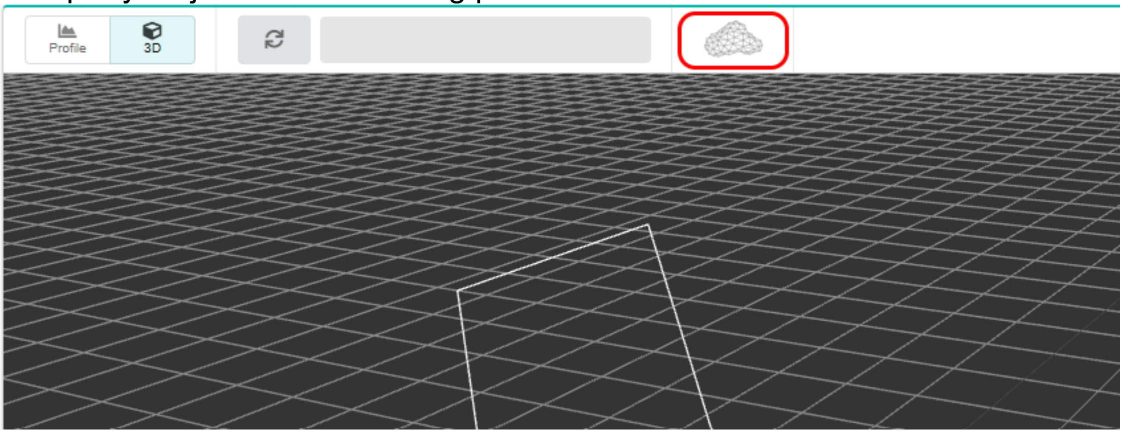
Export of accumulated profiles is possible in three formats:

- **Binary** - Export of individual profiles in a special format. A description of this format is given in the Developer Guide. To view the accumulated profiles in *.bin format, use the **RFProfileView** software. Download link: https://riftek.com/upload/medialibrary/558/RFProfileView.zip
- **3D** - Export of a point cloud in **obj** format. This format is a commonly available format for describing 3D geometry and can be opened by almost any software for working with 3D objects. For example, the **MeshLab** software. Download link: http://www.meshlab.net/#download.
- **Table** - Export of profiles to **csv** table. When exporting to this format, it is possible to select the data composition. The export results can be imported into spreadsheet editors (MS Excel, WPS Spreadsheet, Libbre Office Calc, etc.).

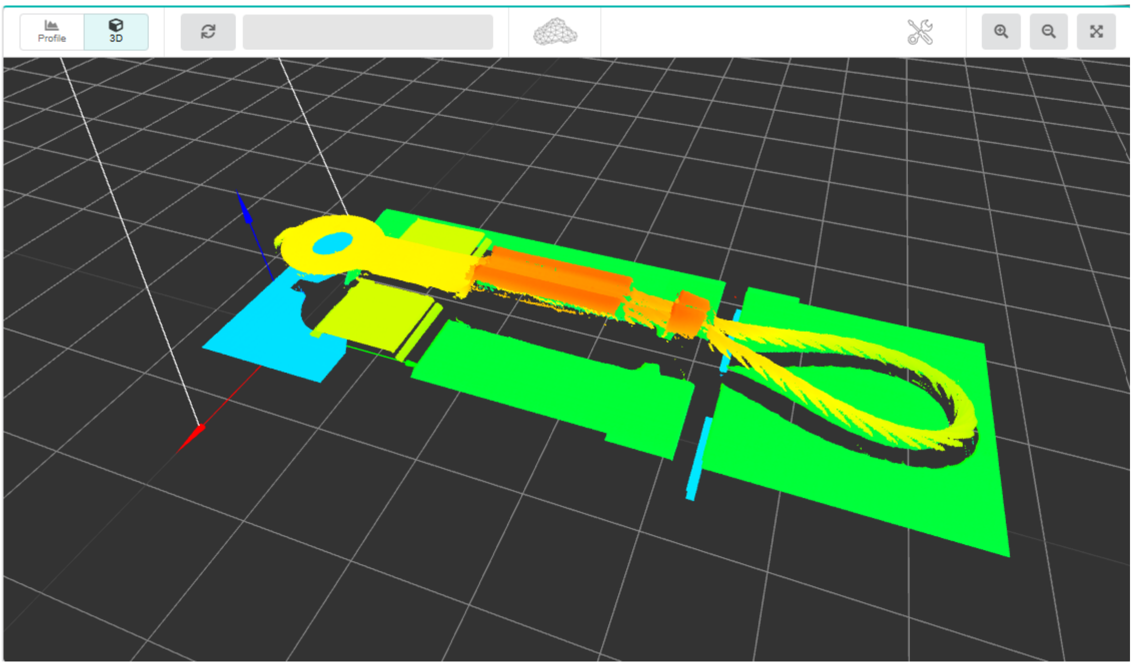


23.4.4. Importing a point cloud for viewing

In 3D viewing mode, you can import a point cloud using the corresponding button. The .obj format is supported. You can load both point clouds exported from a scanner and third-party .obj models containing point data.



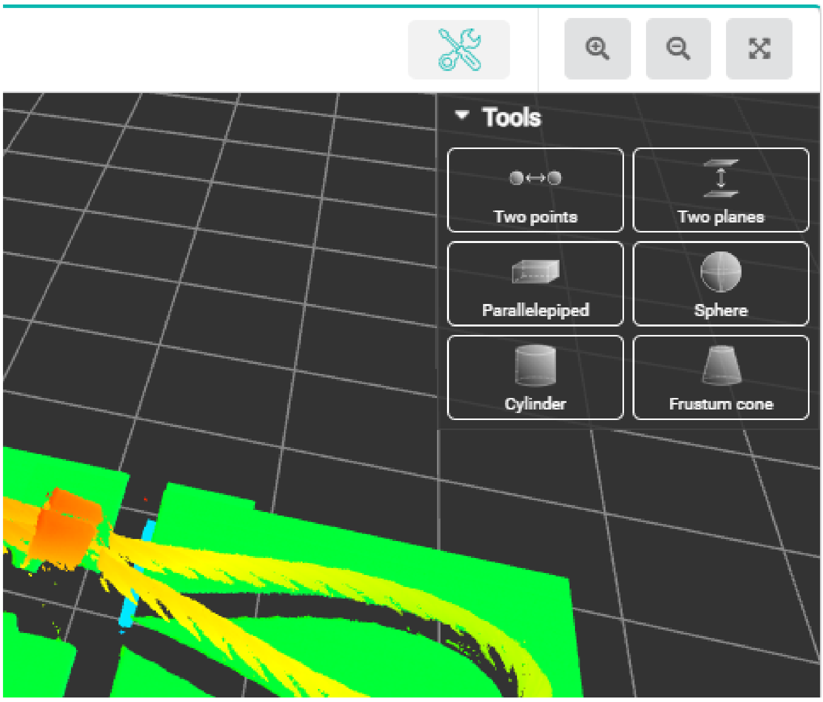
Once loaded, the point cloud is automatically colored based on the height of the points, allowing you to visually assess the vertical distribution of the data.



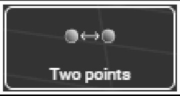
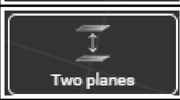
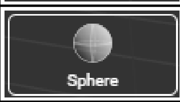
23.4.5. Point cloud measurement tools

Basic measurement tools are available in 3D viewing mode for point cloud analysis. Users can measure distances between points, determine the height of objects, and perform other simple geometric measurements directly on the loaded point cloud.

In the upper right corner, there is a button for selecting manual measurement tools.



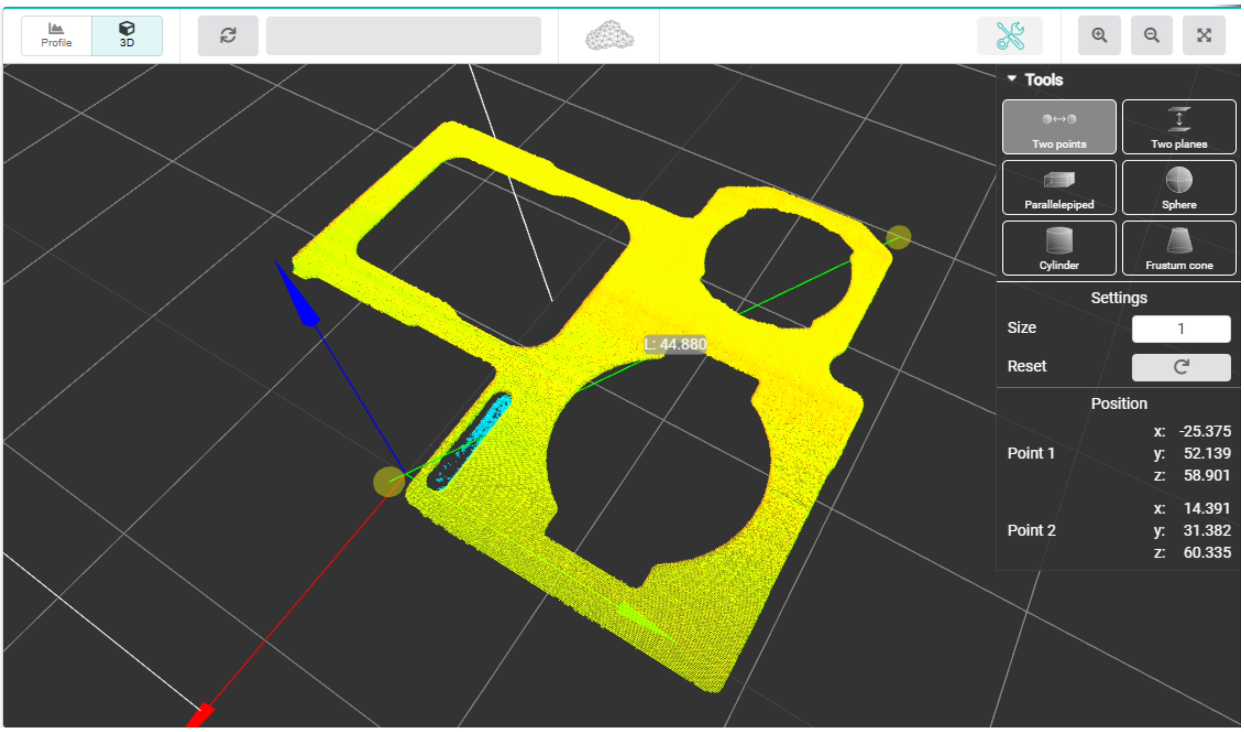
Available measurement tools:

Tool	Icon	Description
Two points		Measuring the distance between two arbitrarily specified points in the cloud.
Two planes		Determining the distance or angle between two given parallel planes.
Parallelepiped		Construction and measurement of a bounding volume in the form of a parallelepiped.
Sphere		Approximation of a spherical shape by points and determination of its parameters.
Cylinder		Constructing a cylinder from selected points and measuring its height and radius.
Truncated cone		Analysis of cone-shaped objects with determination of base radii and height.

Below is a description of each measurement tool, explaining its operating principle and providing a practical application example.

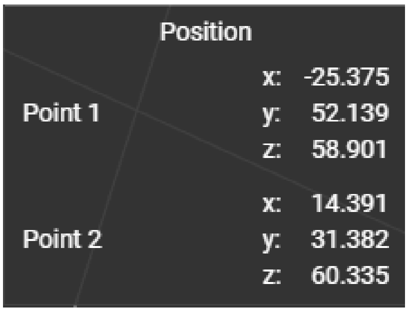
1) Two Points

This tool allows you to measure the distance between two specified points in space. Once activated, the user sequentially adjusts the position of the two points, and the linear distance between them is automatically calculated. This tool is convenient for quickly assessing the linear dimensions of objects.



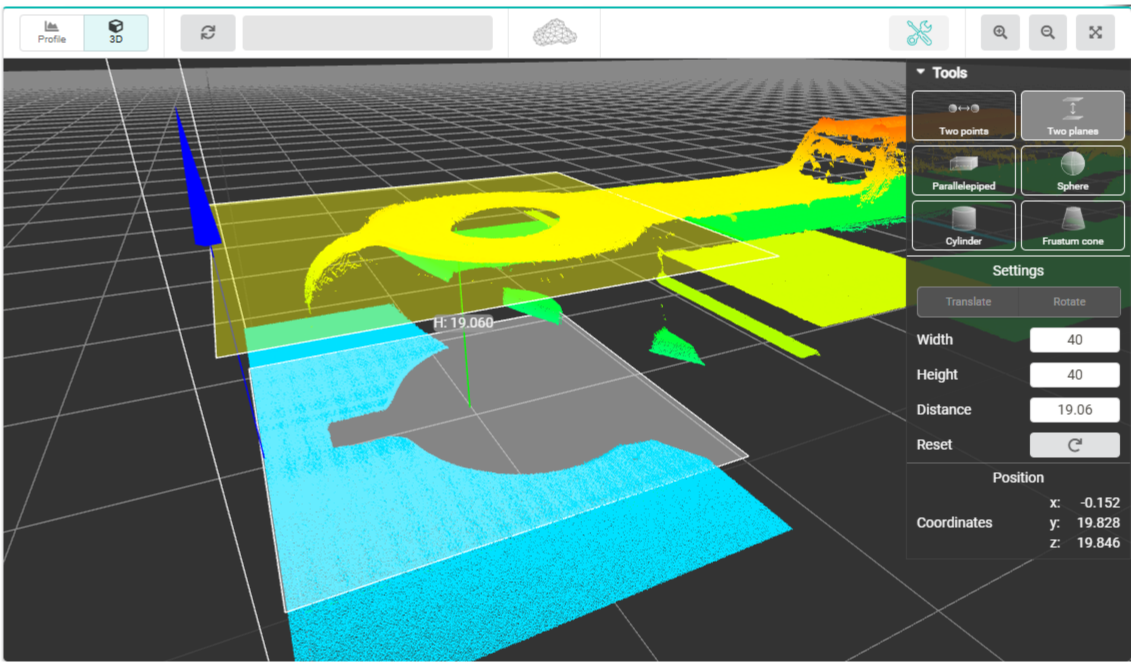
When you select a point in 3D space, arrows appear that align with the coordinate axes (X, Y, Z). Hovering the cursor over one of the arrows allows you to move the point along the corresponding axis by holding down the mouse button. The following settings are available in the tool's parameters: changing the size of the displayed point and resetting the point's coordinates to their original values.

The current coordinates of active points are displayed at the bottom of the window. When a point's position changes, the coordinates are automatically recalculated and updated on the screen.

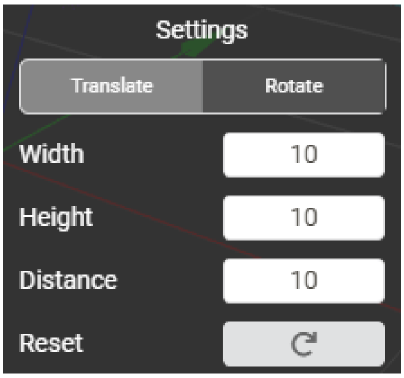


2) Two Planes

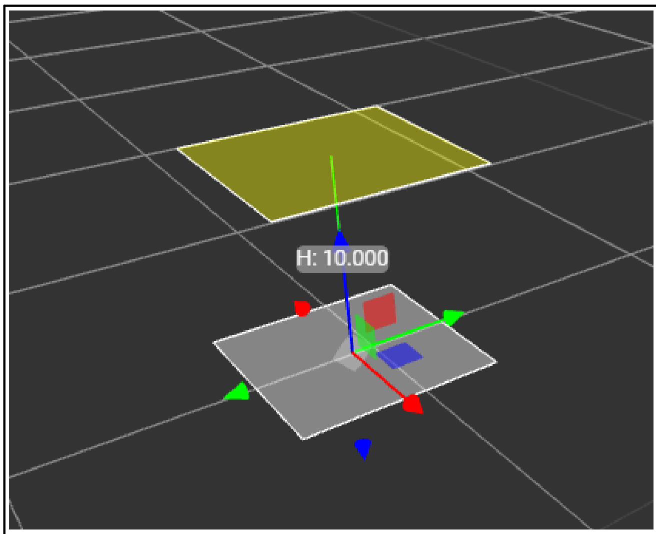
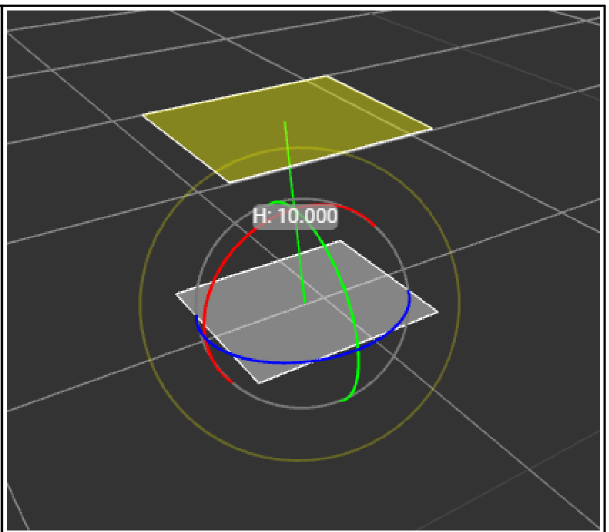
This tool is designed to measure the distance between two parallel planes manually adjusted to selected areas of the point cloud. This method provides a more accurate and convenient measurement compared to the Two Points tool, as the use of planes allows for the unambiguous determination of the perpendicular direction for distance measurement.



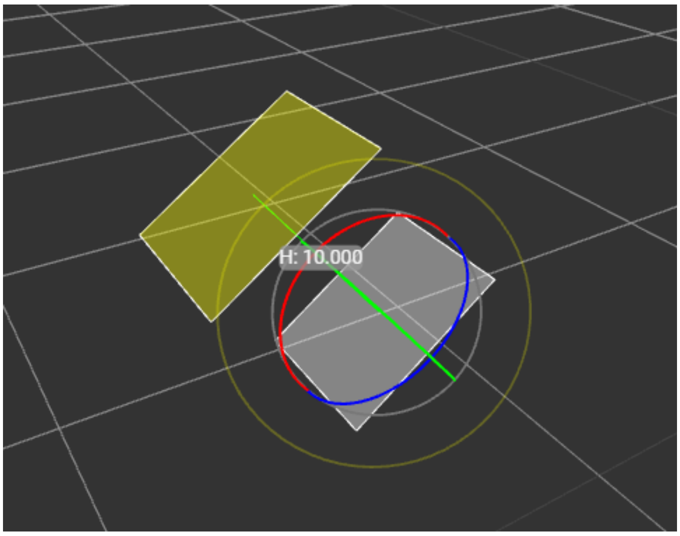
The settings area allows you to select a control mode (movement or rotation), change the size of planes, set the distance between them, and reset the parameters to their original values.



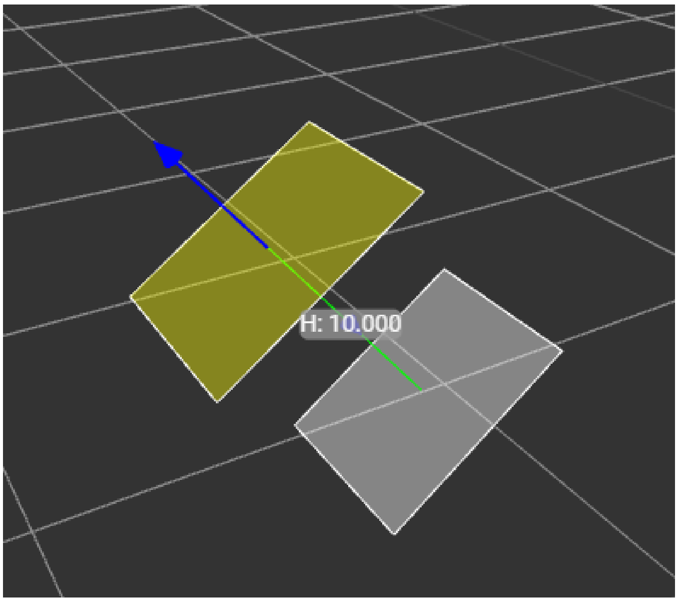
One of the planes acts as a reference and is highlighted in white. When this plane is selected, control elements are displayed: arrows for movement (**Translate** mode) and circles for rotation (**Rotate** mode).

	
Translate mode	**Rotate** mode

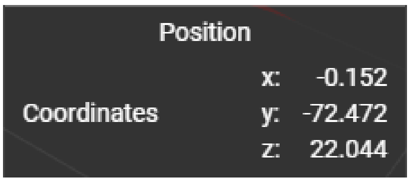
Rotation in **Rotate** mode is performed relative to the center of the reference plane.



The second plane, highlighted in yellow, always remains parallel to the reference plane. When this plane is selected, only the distance between planes perpendicular to the reference plane can be adjusted.

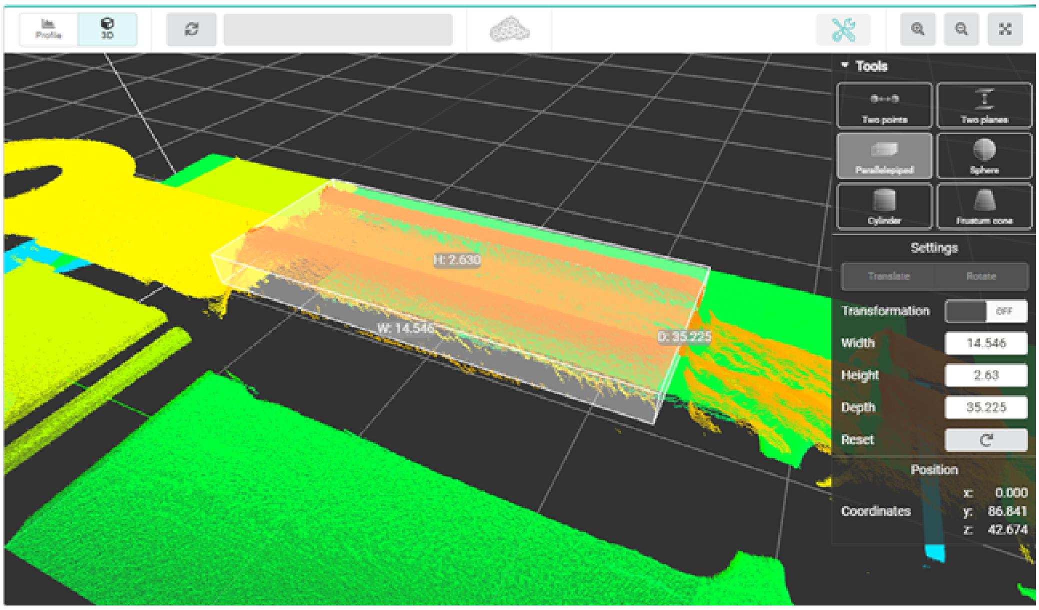


The current coordinates of the center of the reference plane are displayed at the bottom of the window..

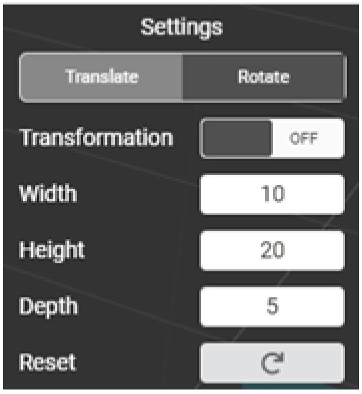


3) Parallelepiped

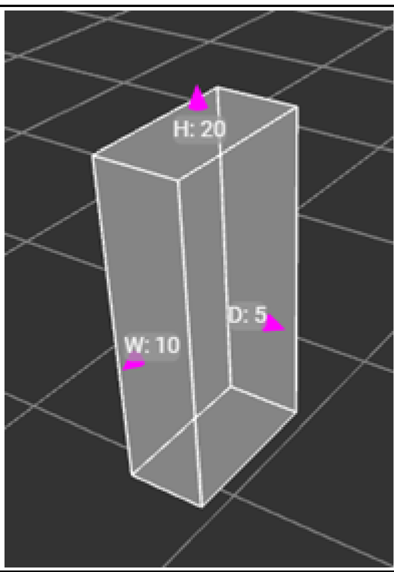
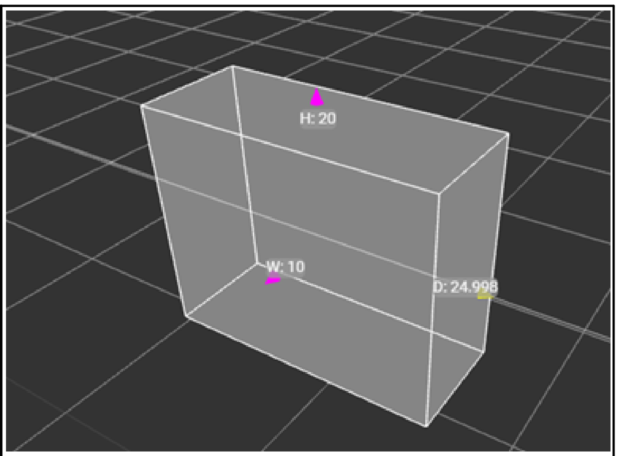
This tool is designed to construct a bounding volume in the shape of a parallelepiped around a selected area of the point cloud. It allows you to quickly estimate the dimensions and sizes of the object bounded by the parallelepiped. The parallelepiped can be manually adjusted in size and position to precisely fit the shape of the object.



The settings area allows you to: select a control mode (move or rotate), enable transformation mode, adjust dimensions (width, height, depth), and reset parameters to their original values.



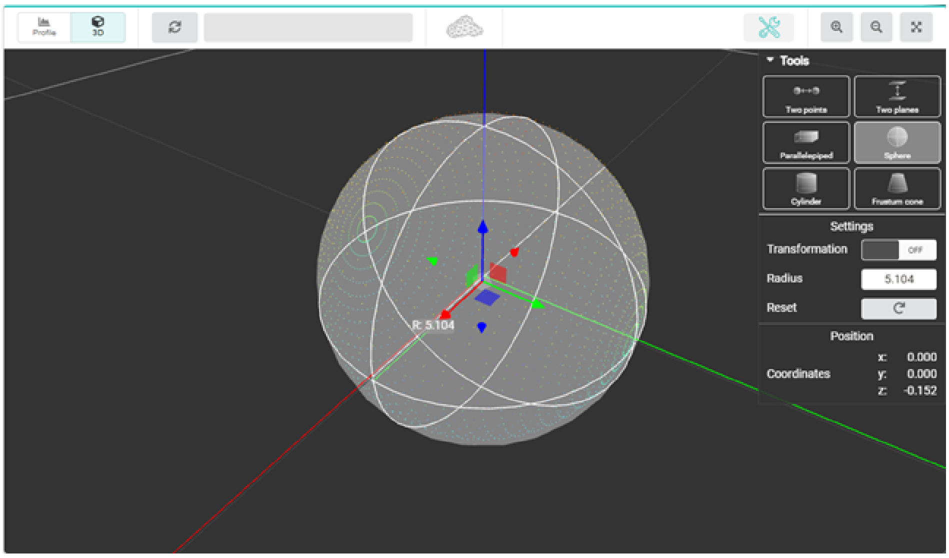
When a parallelepiped is selected, coordinate axes for translation (**Translate** mode) and circles for rotation (**Rotate** mode) are displayed.

	
Translate mode	**Rotate** mode

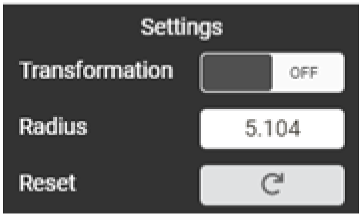
The coordinates of the center of the parallelepiped's base are displayed at the bottom of the window.

4) Sphere

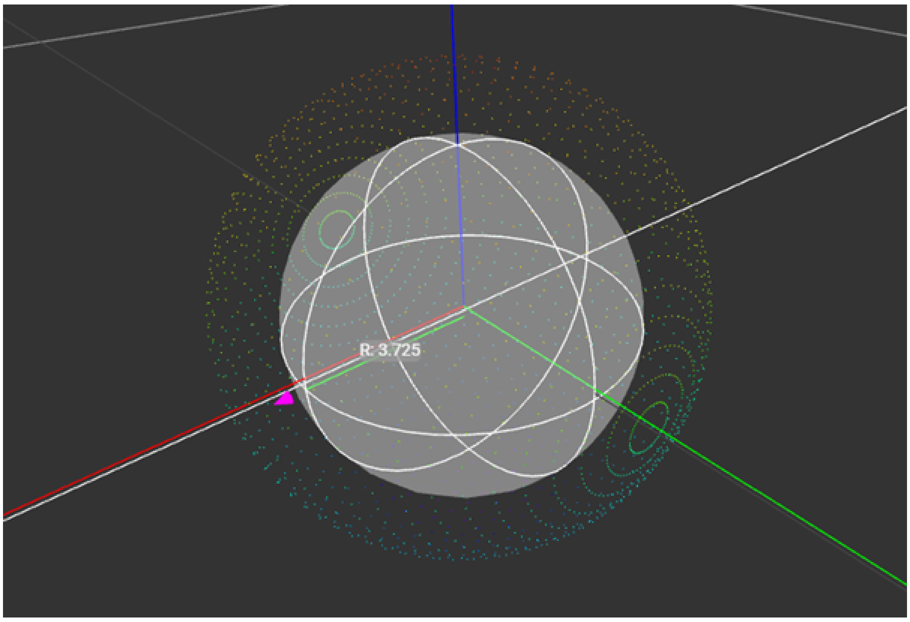
This tool is designed to approximate the shape of a point cloud object using a sphere. It allows you to determine the sphere's parameters – the center and radius. It is used to estimate the size and shape of objects that are close to spherical.



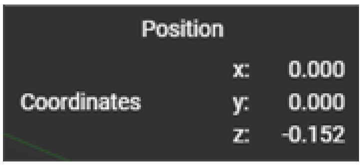
The settings area allows you to enable transformation mode, adjust the sphere radius, and reset the parameters to their original values.



When a sphere is selected, arrows appear along the coordinate axes for moving it. In transform mode, an additional marker (arrow) appears for changing the sphere's radius.

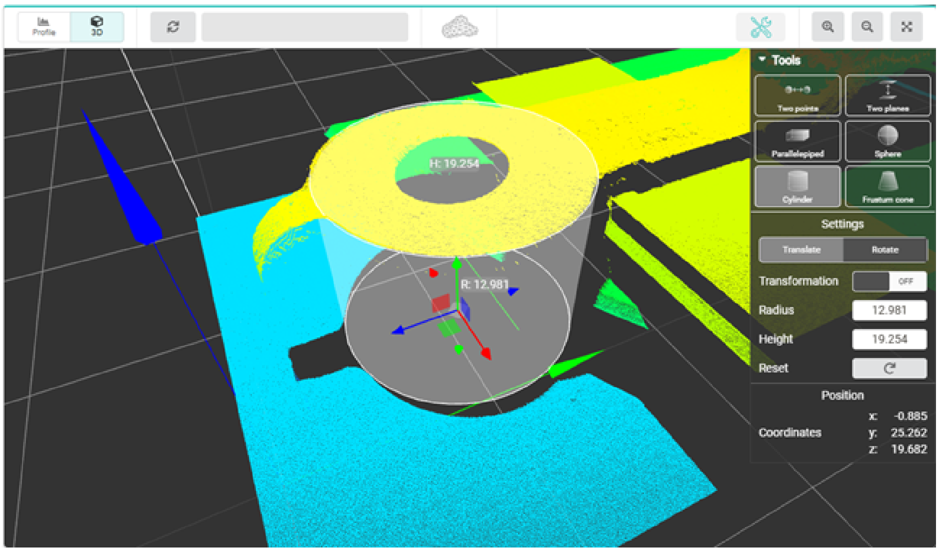


The coordinates of the center of the sphere are displayed at the bottom of the window.

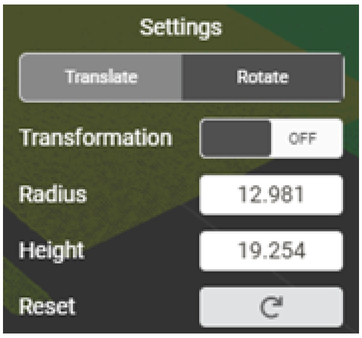


5) Cylinder

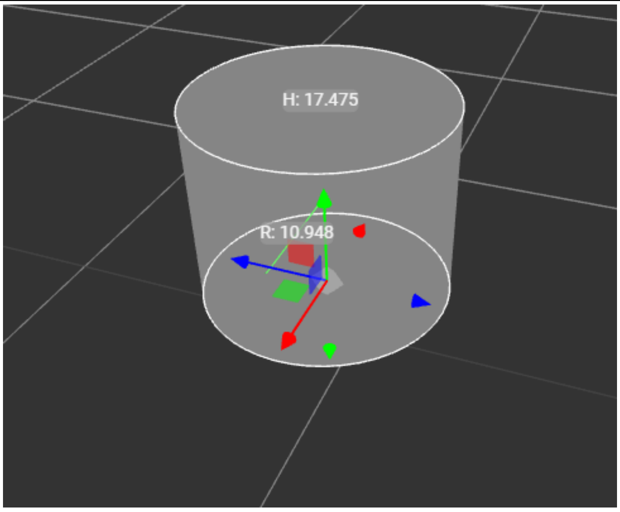
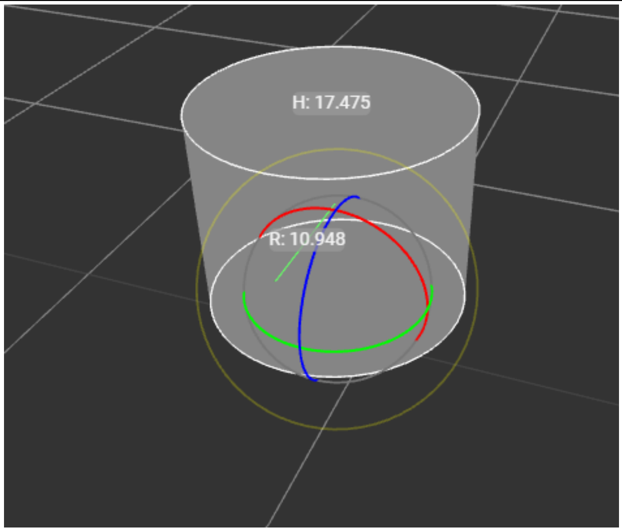
This tool is designed to approximate a selected area of a point cloud with a cylindrical model. It allows you to define the cylinder's parameters, including its base center, height, and radius. It is used to measure and analyze cylindrical objects.



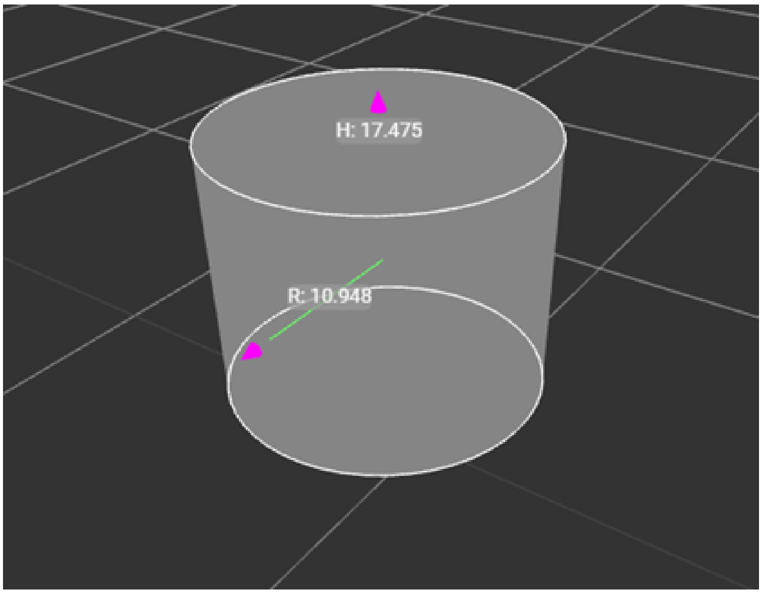
The settings area allows you to: select a control mode (move or rotate), enable transformation mode, adjust dimensions (radius and height), and reset parameters to their original values.



When a cylinder is selected, arrows are displayed along the coordinate axes for translation (**Translate** mode) and circles for rotation (**Rotate** mode).

	
Translate	Rotate

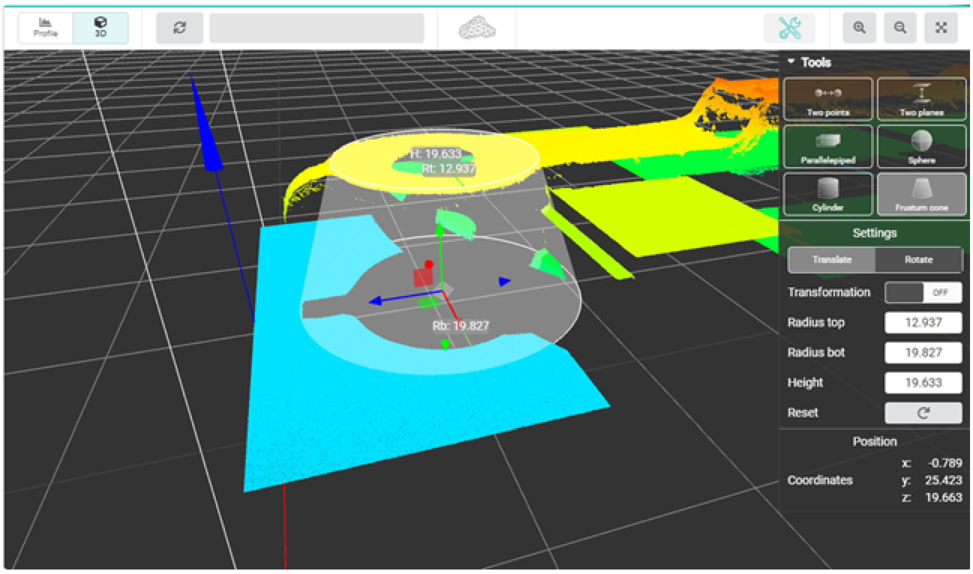
When you turn on the transformation mode, markers appear for changing the radius and height of the cylinder.



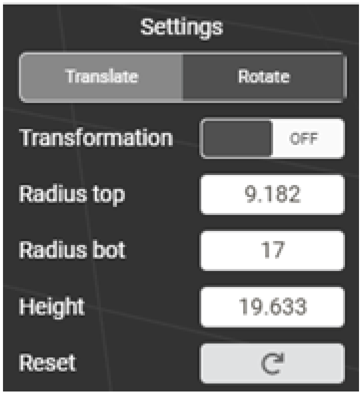
The coordinates of the center of the cylinder base are displayed at the bottom of the window.

6) Truncated Cone

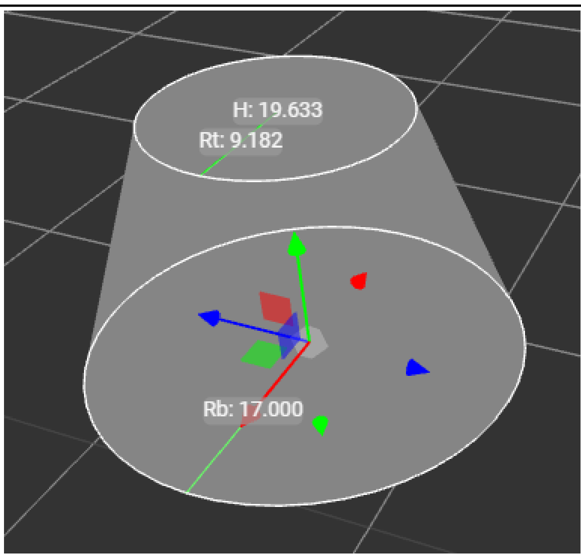
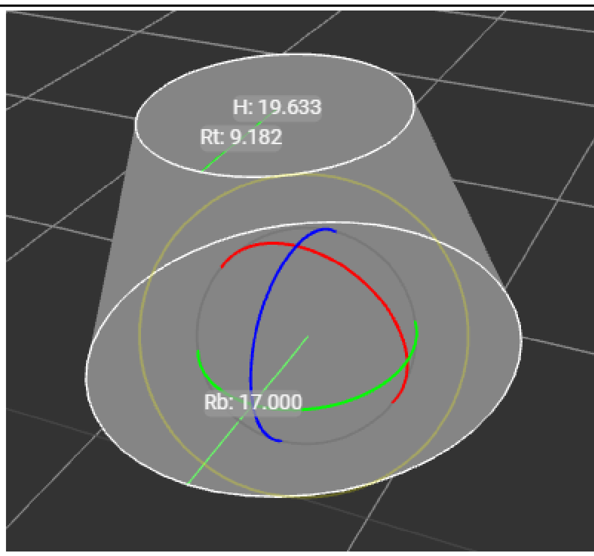
This tool is designed to approximate a selected region of a point cloud with a truncated cone model. It allows you to determine the parameters of the truncated cone, such as the base radii and height. It is used to analyze and measure cone-shaped objects.



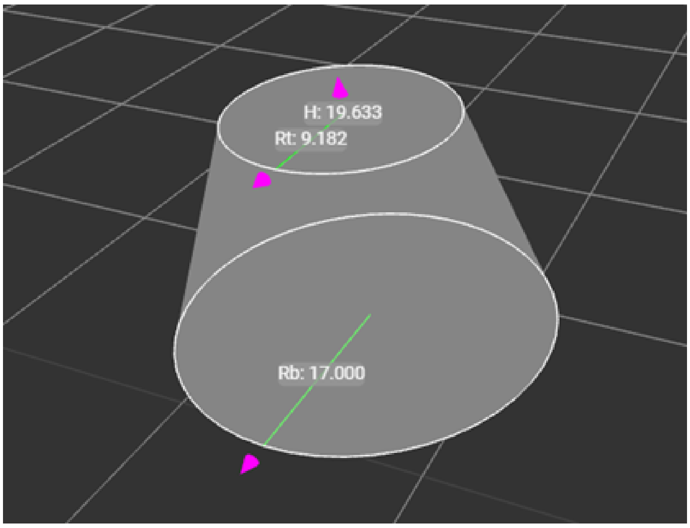
The settings area allows you to: select a control mode (move or rotate), enable transformation mode, adjust dimensions (radii of the upper and lower bases, height), and reset parameters to their original values.



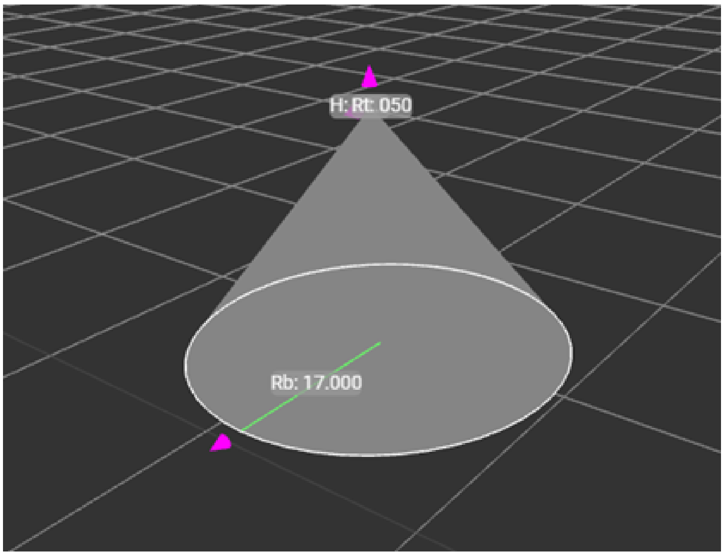
When a truncated cone is selected, arrows are displayed along the coordinate axes for translation (Translate mode) and circles for rotation (Rotate mode).

	
Translate	Rotate

In the transformation mode, markers appear for changing the radii of the lower and upper bases of the truncated cone, as well as for adjusting its height.

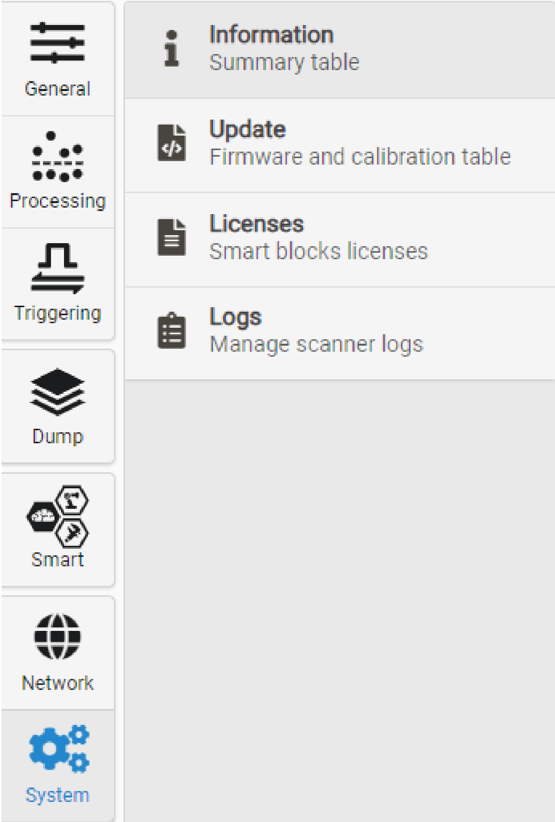


If the radius of the upper base of a truncated cone is zero, the shape is transformed into a regular cone.



The coordinates of the center of the lower base of the cone are displayed at the bottom of the window.

24. "System" tab



24.1. "Information" section

The **Information** section contains general information about the scanner.

Device information	
Operating time after start	00h 24m 31s
Total operating time	970h 54m 06s
Laser operating time	00h 36m 41s
CMOS-sensor temperature	64.6 °C
Name	RF627Smart Laser Scann
Model	RF627Smart
Serial	190101
Wavelength	660 nm
Firmware version	2.16.1
Hardware version	18.6.20.0
Calibration date and time	01.01.1970 03:00:00 (UTC +3)
Smart	Enabled
Working ranges	
Base Z (SMR)	76 mm
Range Z (MR)	100 mm
Range X Start (XSMR)	48 mm
Range X End (XEMR)	82 mm

In this section, you can change the scanner name displayed in the upper area of the web interface by entering a new name in the **Name** field and pressing **Enter**.

24.2. "Update" section

This section is intended to update the scanner firmware and the calibration table.

24.2.1. Updating and saving the firmware

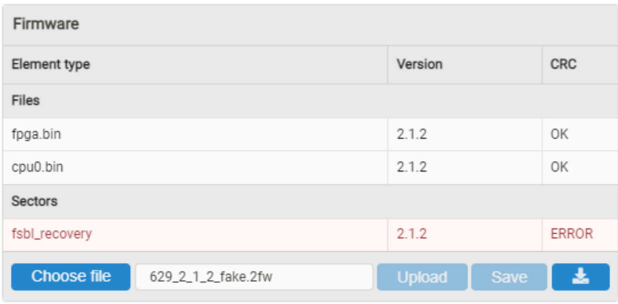
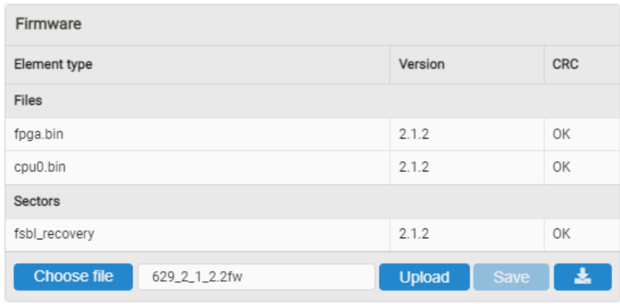
The firmware update file is provided by the manufacturer as new functions are implemented and bugs are fixed. The latest firmware versions are available here:

https://cloud.riftek.com/index.php/apps/files/?dir=/RF627_Firmware

Procedure:

1) Click **Choose file** and select the firmware file in the «.2fw» format.
2) Click **Upload** to upload the selected file.

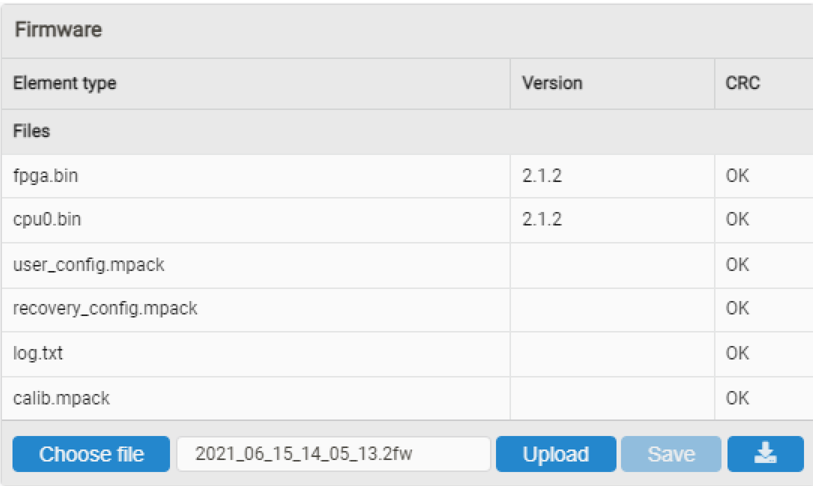
If there were some failures during the firmware upload, a checksum mismatch error will appear. In this case, click **Upload** again.

	
Checksum error	Successful upload

3) Click **Start** to start the update process.

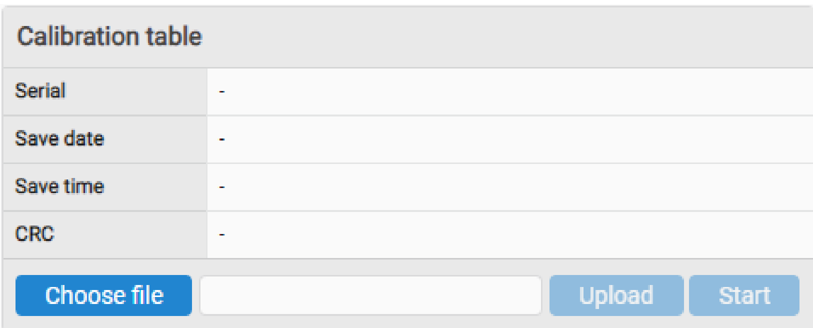
If the IP address settings haven't been changed after completing the firmware update process and restarting the scanner, the web interface will automatically reboot without waiting for the timer to expire. If the network settings have been changed, the web interface will reboot with the default IP address (192.168.1.30) after the timer expires.

To restore settings after saving incorrect parameters, update errors and other cases, it is possible to save the complete internal state of the scanner. Clicking the button will generate a file containing the complete state of the scanner. Firmware recovery is performed in the same way as updating.



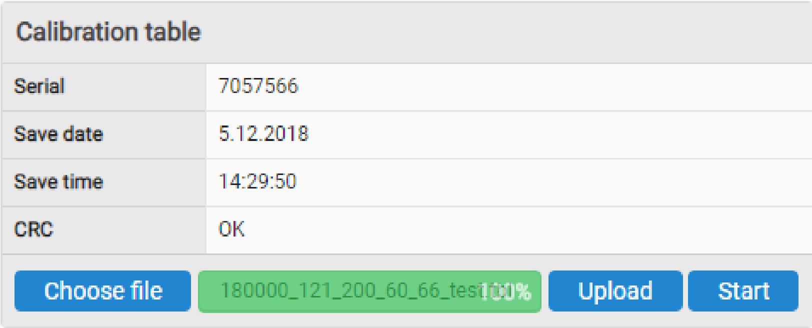
Recovering the saved firmware

24.2.2. Updating the calibration table



Procedure:

1) Click **Choose File** and select the calibration table file.
2) Click **Upload** to upload the selected file.

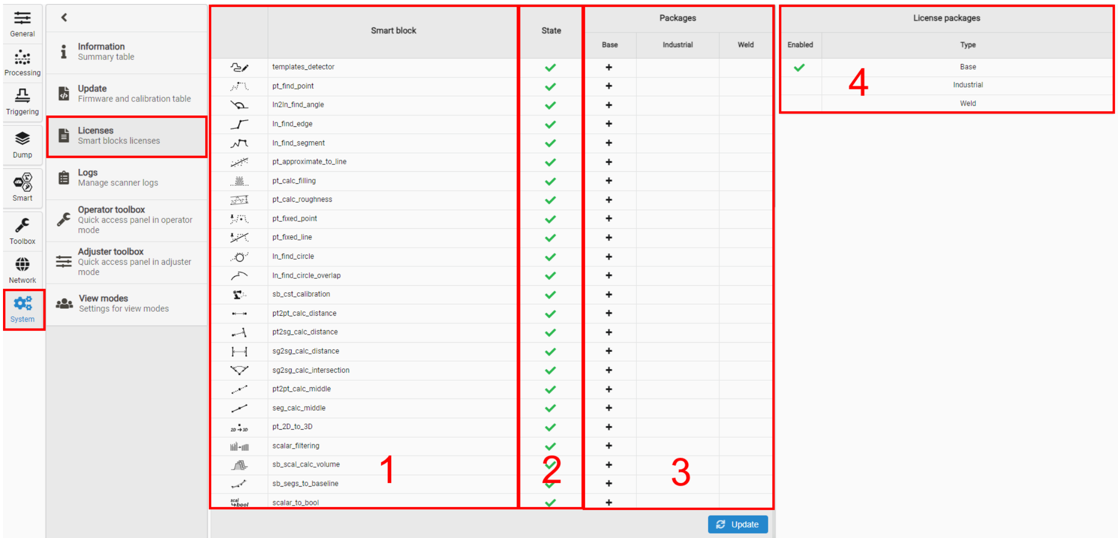


3) Click **Start** to start the update process.

24.3. "Licenses" section

License management is carried out on the basis of license packages, optionally (except for the "Base" package) included by the manufacturer. Each smart block can be included in one or more license packages.

Package	Description
Base	Contains smart blocks for selecting profile primitives (points, segments, etc.), smart blocks for their mathematical and statistical processing (distances, filtering, etc.), smart blocks for transmitting/receiving data in the form of tcp and udp packets.
Industrial	Contains smart blocks of the “Base” package, as well as smart blocks for transmitting/receiving data via industrial protocols (EthernetIP, ModbusTCP), smart blocks for controlling industrial robots.
Weld Tracking	Contains smart blocks of the “Industrial” package, as well as smart blocks for weld seam tracking in real time.
Weld Seam Inspection	Contains smart blocks of the “Industrial” package, as well as smart blocks for monitoring the geometric parameters of the weld seam.



Area 1 contains the smart blocks available in the scanner firmware, area 2 shows the license status for each block, area 3 - contents of the selected package, area 4 - packages types.

✔ - smart block is available;

✖ - no license.

To enable the license package:

1. Send a request to info@riftek.com.
2. Pay the cost of the required license package.

3. Coordinate the time and provide remote access (using free remote desktop software) to the scanner manufacturer to activate the license package.

24.4. "Logs" section

The **Logs** section displays information about the scanner operation that can be used to detect possible errors.

Time		Message
00:00:00	[INFO]	|===|
00:00:00	[INFO]	|===============STARTING 2D LASER SCANNER=============|
00:00:00	[INFO]	|===|
00:00:00	[----------]	System monitor module
00:00:00	[RUN]	Setup errors handlers
00:00:00	[INFO]	Success
00:00:00	[RUN]	Init GPIO_PS for leds and button(s)
00:00:00	[INFO]	Success
00:00:00	[RUN]	Init CPU temperature reader
00:00:00	[INFO]	Success
00:00:00	[RUN]	Init temperature sensors reader
00:00:00	[INFO]	Success
00:00:00	[RUN]	System monitor thread
00:00:00	[INFO]	Success
00:00:00	[----------]	
00:00:00	[----------]	File system
00:00:00	[RUN]	Low level init
00:00:00	[INFO]	Success
00:00:00	[RUN]	Mount file system
00:00:00	[INFO]	Success
00:00:00	[INFO]	Success
00:00:00	[----------]	
00:00:00	[----------]	Device EEPROM module
00:00:00	[RUN]	Physical init EEPROM
00:00:00	[INFO]	I2C clock tet to: 100000
00:00:00	[INFO]	Success

Save LOG to internal memory 181/181 Download Save

This tab contains information about the operations performed and their order. To get the log file, click the **Download** button.

Data can be exported in *.txt and *.csv formats. You can select the format after clicking the **Save** button.

The **Save LOG to internal memory** option enables saving the log file to the internal memory of the scanner.

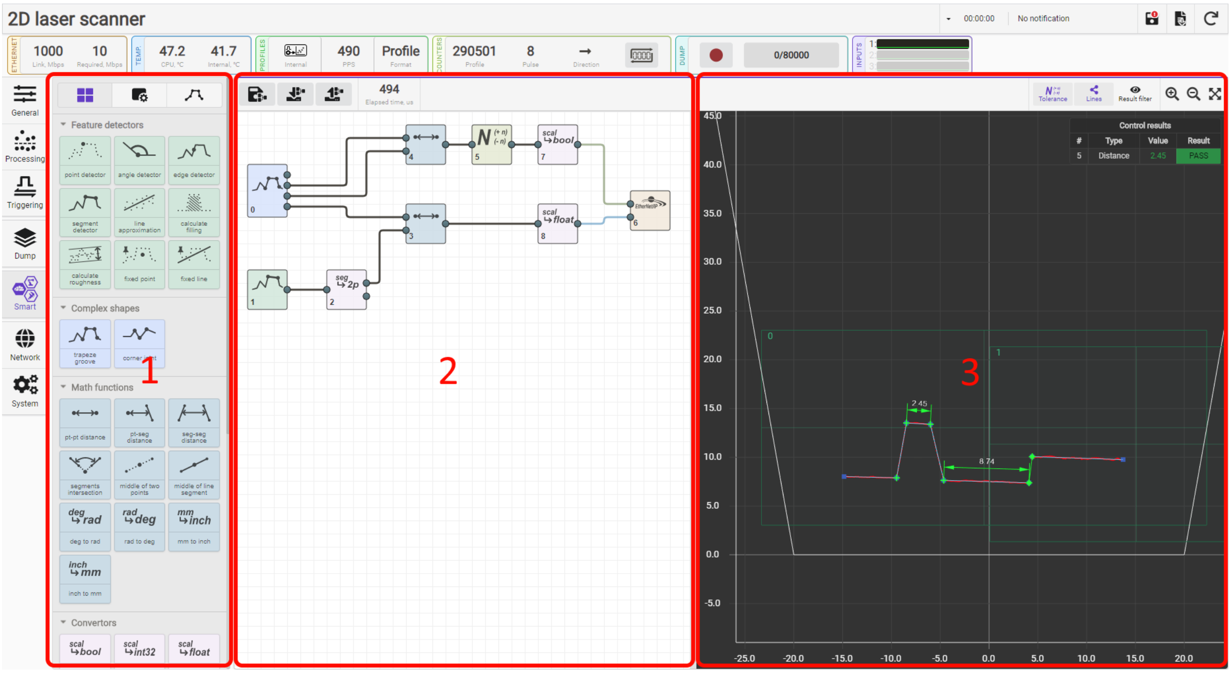
25. "Smart" tab

This chapter only applies to Smart scanners.
The **Smart** tab is designed to implement the smart functions of the scanner.
Smart functions include:

- Creating an algorithm for measuring various geometrical and statistical quantities of the controlled profile.
- Performing measurements in real time according to a given algorithm.
- Processing of measurement results and automatic decision-making about their being within acceptable limits (control of tolerances).
- Transmitting measurement results via industrial (Ethernet/IP, Modbus TCP) and simplified (UDP) network protocols.
- Forming control actions at the physical outputs of the scanner.

To ensure the simplicity and ease of use of smart functions, the concept of a “computation graph” (hereinafter referred to as the graph), formed by the user to solve a specific problem, is applied. A graph is an ordered sequence of operations performed by a scanner. This sequence is presented in the form of smart blocks and links between them. When the structure of the graph is changed, its ordering is automatically performed (i.e., determining the order of performing calculations). **Restriction:** cyclic links are not allowed in the graph.

The main window of the web interface with an active **Smart** tab:



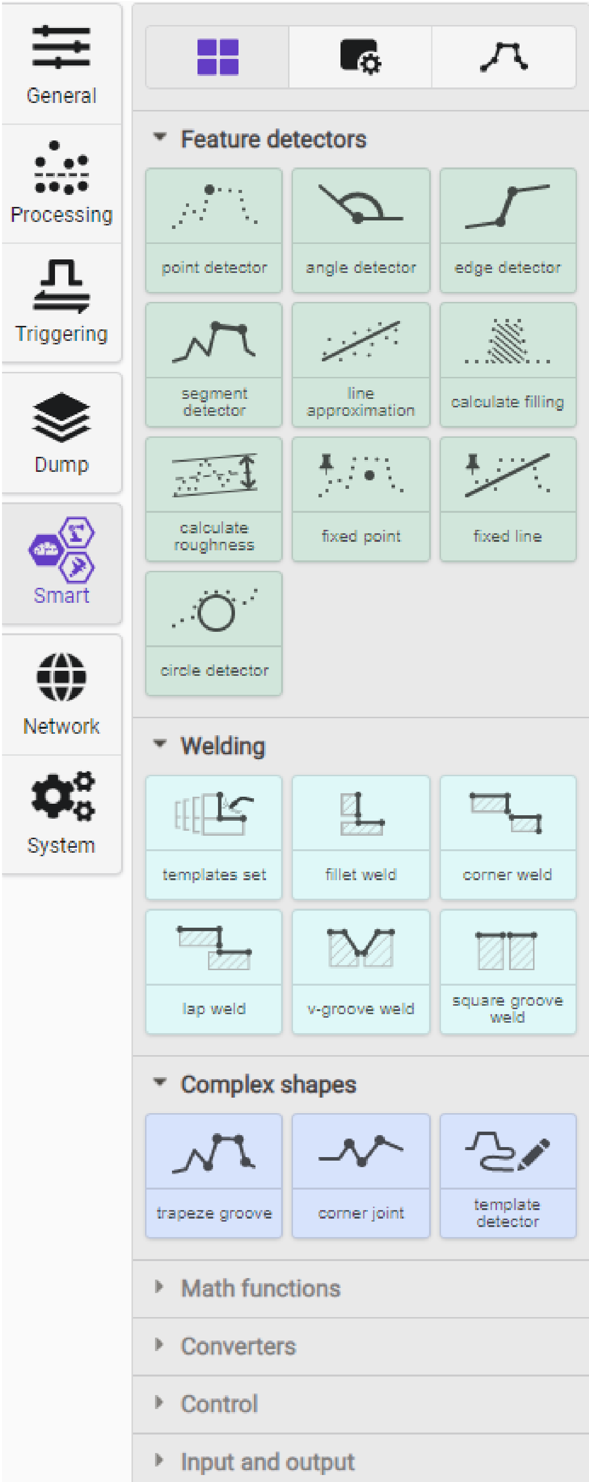
Designations:
1 - Smart blocks and parameters area.
2 - Graph creation area.
3 - Measurement results display area.

25.1. Smart blocks and parameters

This area is intended for displaying a set of smart blocks, setting parameters of blocks added to the graph, and setting parameters for approximating a profile by a set of segments.

The area contains three tabs:

- **Smart blocks** - a set of smart blocks grouped by functionality.
- **Block settings** - parameters of the block selected on the graph.
- **Profile approximation** - parameters of profile approximation by segments.



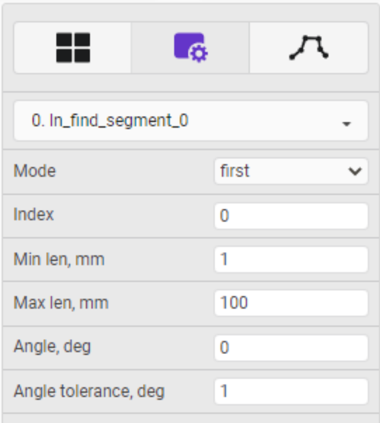
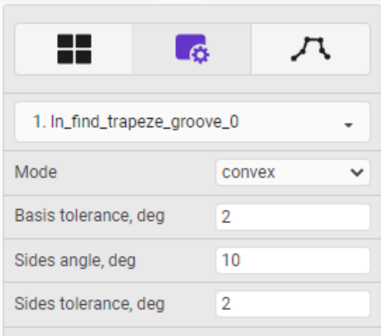
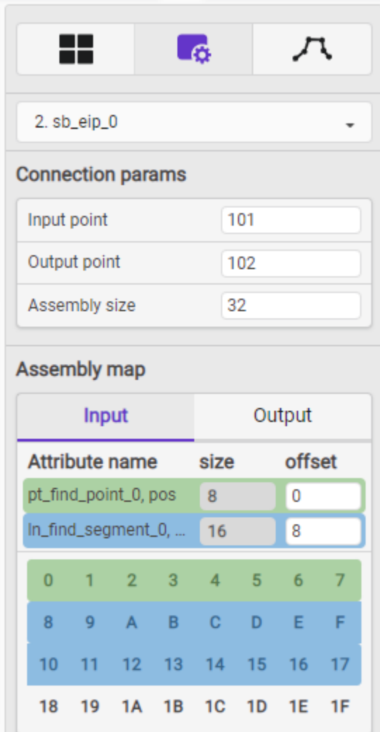
25.1.1. "Smart Blocks" tab

The tab contains available smart blocks. The pictogram on the smart block schematically shows the function it performs, while the output data (the result of the block operation) is shown in bold. Examples:

<table>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>the smart block outputs a point</td>
<td>the smart block outputs a segment</td>
<td>the smart block outputs a line</td>
</tr>
</table>

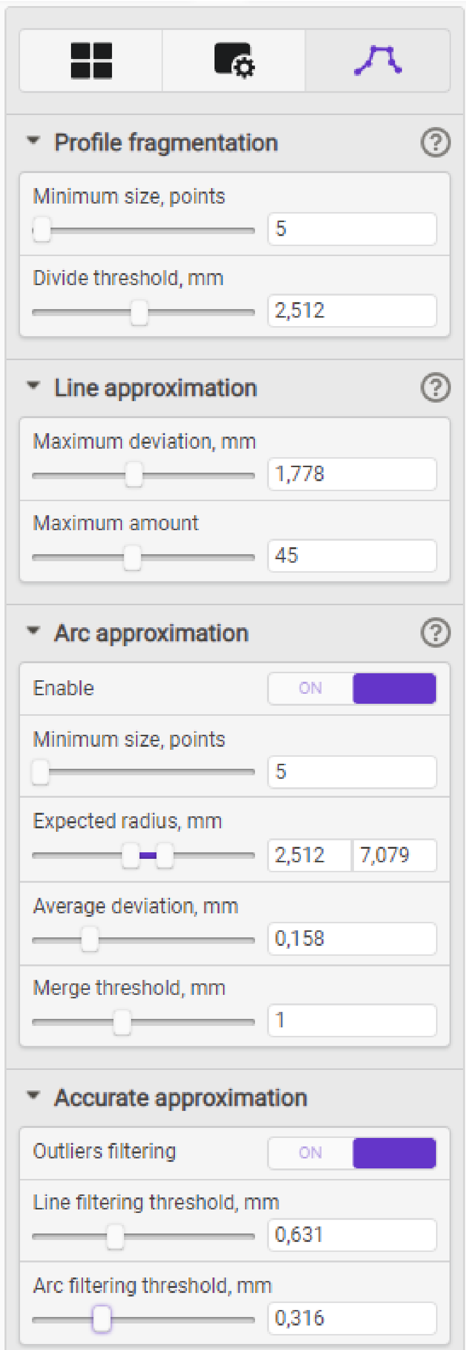
25.1.2. "Block Settings" tab

The tab provides access to the settings of the block selected on the graph. Examples:

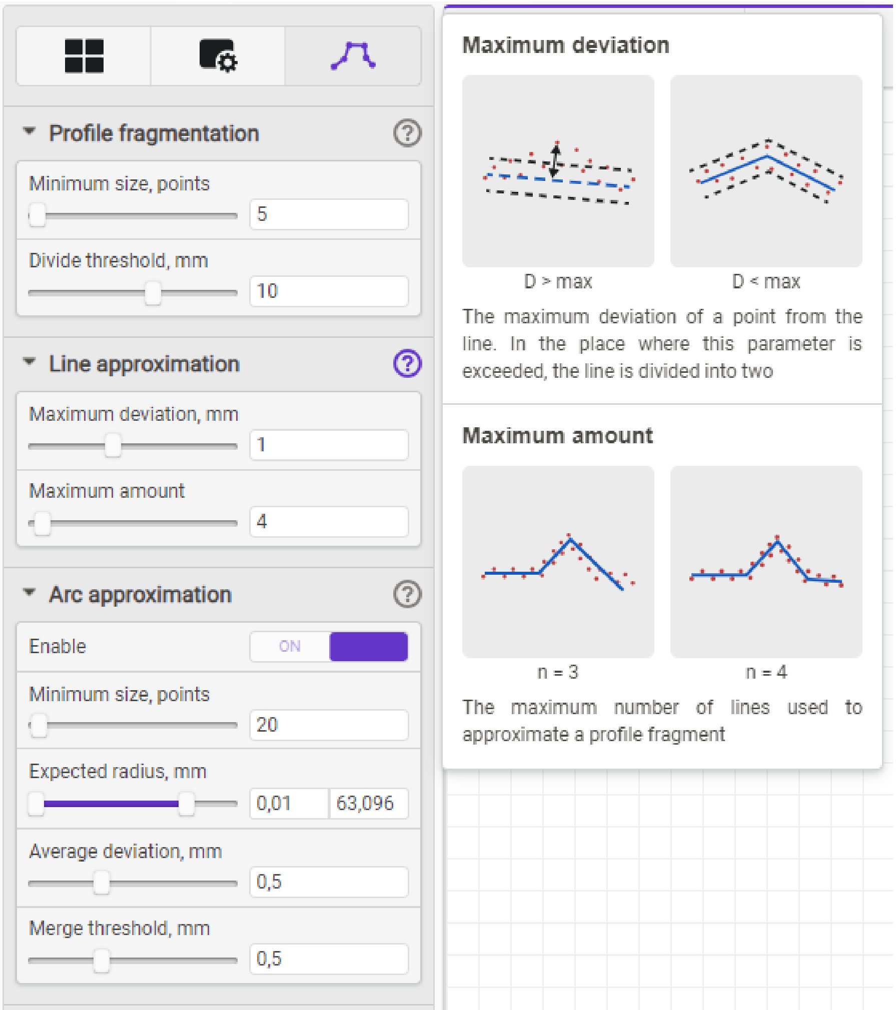
<table>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>“segment detector” smart block settings</td>
<td>“trapeze groove” smart block settings</td>
<td>“eip” smart block settings</td>
</tr>
</table>

25.1.3. "Profile Approximation" tab

The tab provides access to the parameters of the profile approximation algorithm.



To display a description of parameters, click on the question mark next to the section name.



25.2. Creating a Smart function

The Smart function is created in two stages:
Stage 1 - Profile approximation.
Stage 2 - Building a graph.

25.2.1. Stage 1. Profile approximation

Profile approximation is the first stage in configuring the smart function of the scanner. The "Profile Approximation" tab provides access to the parameters of the profile approximation algorithm. The stability and accuracy of the measurement results directly depend on the quality of the approximation. The optimal approximation is achieved when each straight line on the profile forms a segment, not a broken line, and a circle (arc) is approximated by a circular arc. In difficult cases, when the profile features do not allow approximating a circle with an arc with sufficient accuracy, approximation by a set of segments is possible, followed by approximation by a circle with a special smart block.

To perform the approximation, it is necessary to place a sample of the controlled object in the scanner's field of view and achieve the required profile quality in accordance with the procedures described in Section 19.

The profile approximation algorithm can be divided into three sequentially performed stages:

1. Splitting the entire set of profile points into fragments.
2. Splitting each fragment into a set of approximating line segments and arcs.
3. Clarification of approximating line segments and arcs.

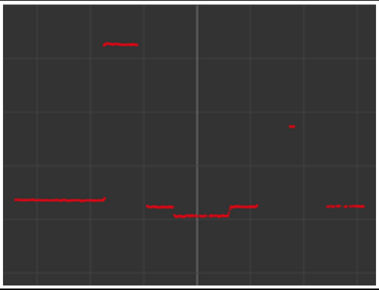
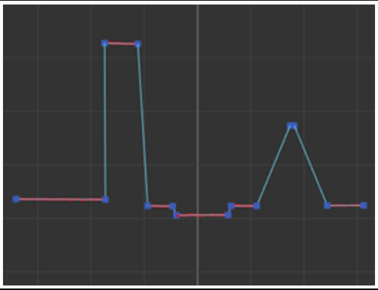
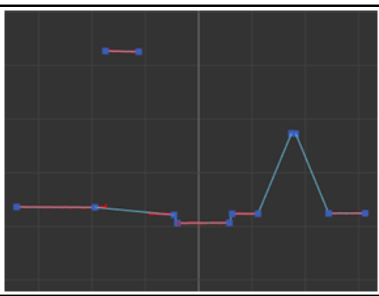
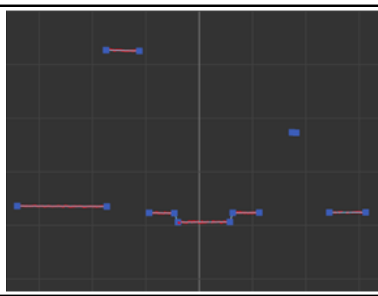
25.2.1.1. Splitting profile points into fragments

The procedure of splitting into fragments is intended to combine the points belonging to one surface, but separated by random outliers or features of the scanned object. At this stage, the profile fragments containing less than a specified number of points are excluded from further processing.

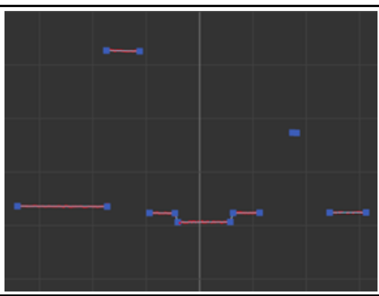
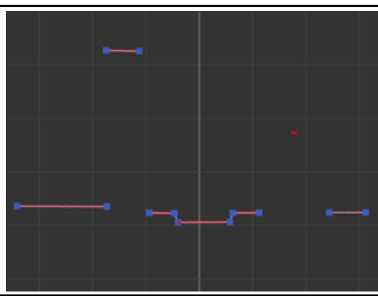
The following parameters from **Profile segmentation** section affect the splitting results:

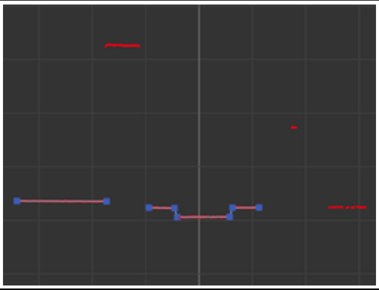
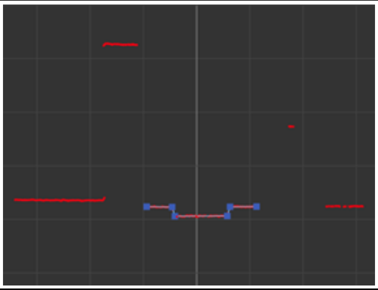
- **Minimum size, points** - the minimum required number of points in a profile fragment for its participation in further processing.
- **Divide threshold, mm** - the minimum distance in millimeters between two successive profile points (left to right) required to combine these points into a single fragment.

The splitting is done automatically by manipulating the specified parameters. The control of merging the points into fragments is carried out visually according to the results of approximating the profile by segments. Namely, if the ends of successive segments are not connected, then they belong to different contours. Examples:

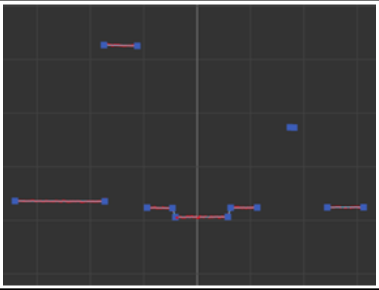
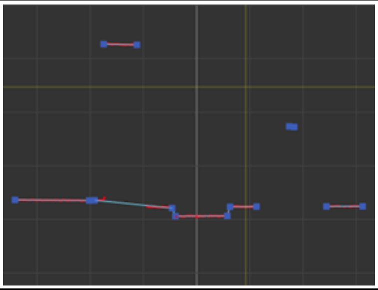
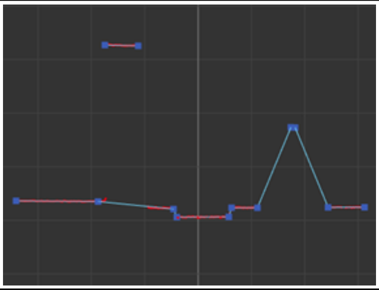
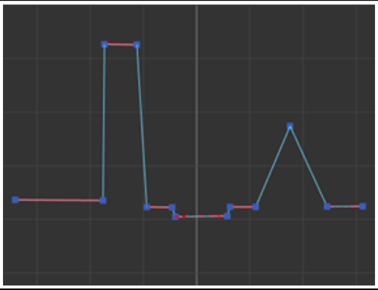
	
Source profile	The profile is split into 1 fragments - all segments are connected
	
The profile is split into 2 fragments - the contour from above consists of one segment, and the contour from below consists of 10 segments (the shortest one forms the apex of the triangle)	The profile is split into 5 fragments (the shortest one forms the apex of the triangle)

Examples of the influence of parameters on splitting a profile into contours.The **Minimum size** parameter:

	
Minimum size = 5 points	Minimum size = 10 points

	
Minimum size = 60 points	Minimum size = 150 points

Examples of the influence of parameters on splitting a profile into fragments. The **Divide threshold** parameter:

	
Divide threshold = 4 mm	Divide threshold = 6 mm
	
Divide threshold = 10 mm	Divide threshold = 15 mm

25.2.1.2. Splitting each fragment into a set of approximating line segments and arcs

Each fragment is divided into approximation elements: segments and arcs, while the approximation parameters are set separately for segments and for arcs.

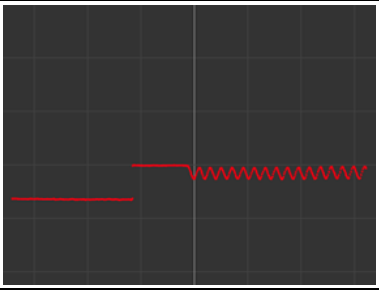
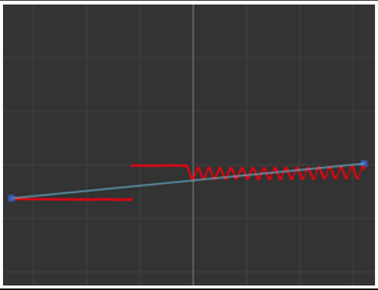
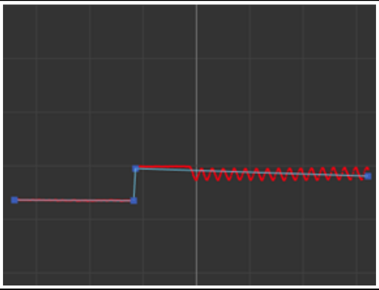
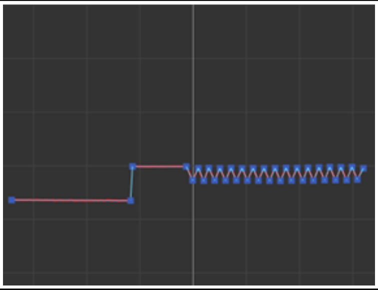
25.2.1.2.1. Approximation by segments

Approximation parameters:

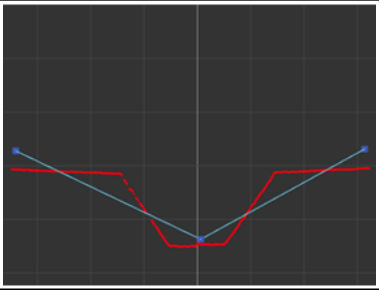
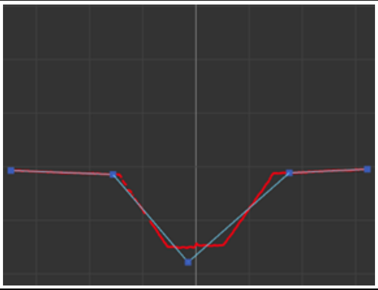
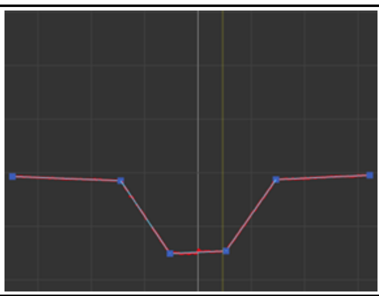
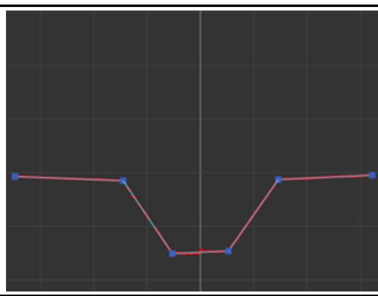
- **Maximum deviation, mm** - the maximum permissible distance in mm between the profile point and the approximating segment. If the next point does not meet this criterion, a new segment is formed. Thus, this parameter affects the approximation granularity.
- **Maximum amount** - the maximum permissible number of lines in the fragment, which acts as a limitation for the approximation algorithm. If splitting requires more lines than specified by this parameter, the **Maximum deviation** parameter value is ignored and splitting stops.

The splitting is done automatically by manipulating the specified parameters. The control of splitting is carried out visually.

Examples of the influence of parameters on splitting a fragment into a set of segments. The **Maximum deviation** parameter:

	
Source profile	Maximum deviation = 2 mm
	
Maximum deviation = 1 mm	Maximum deviation = 0.1 mm

Examples of the influence of parameters on splitting a fragment into a set of segments. The **Maximum amount** parameter:

	
Maximum amount=2	Maximum amount = 4
	
Maximum amount = 5	Maximum amount = 10. New lines were not added, since all approximating lines meet the criterion **Maximum deviation** = 1 mm

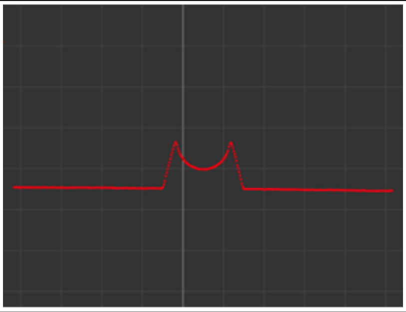
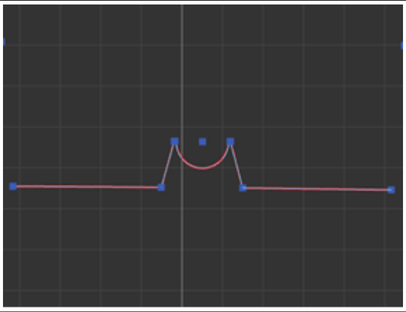
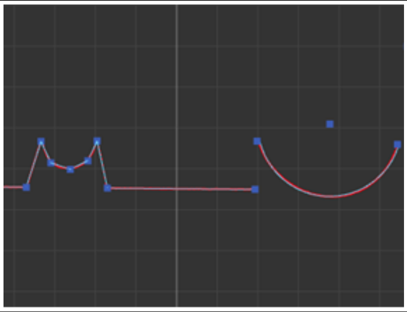
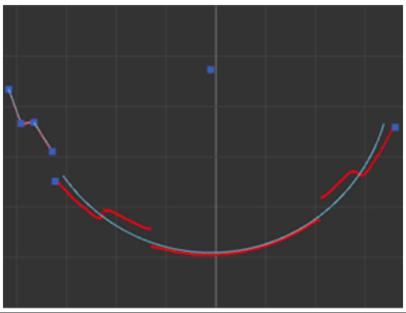
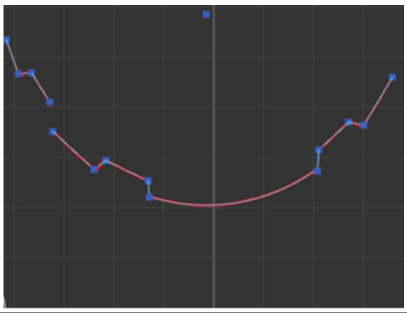
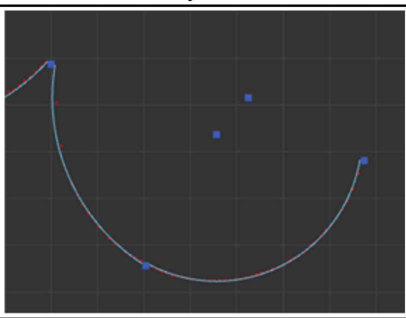
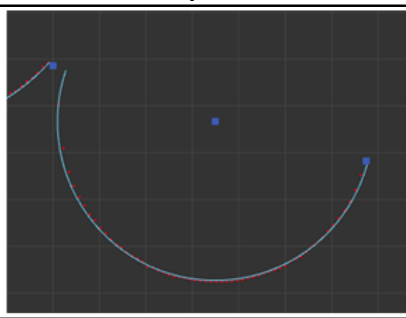
25.2.1.2.2. Approximation by arcs

Approximation parameters:

- **Enable** - enables the use of arcs when approximating the profile. If the parameter is **OFF**, then the profile will be approximated only by segments.
- **Minimum size, points** - sets the minimum required number of points that must form an arc. If a profile element contains fewer points, it will be approximated by a line segment.

- **Expected radius, mm** - sets the minimum and maximum radii of the arc circle. If, as a result of the approximation of a profile element, an arc is obtained, the radius of the circle of which does not fit into the specified limits, then this element will be approximated by a segment.
- **Average deviation, mm** - the admissible value of the average (by points) error of approximating a profile element by an arc. If this parameter is exceeded, the element is approximated by a segment.
- **Merge threshold, mm** - the threshold for combining consecutive (adjacent) arcs into one. It sets the maximum deviation of the centers of the circles of arcs and their radii for combining them into one arc with averaging the parameters.

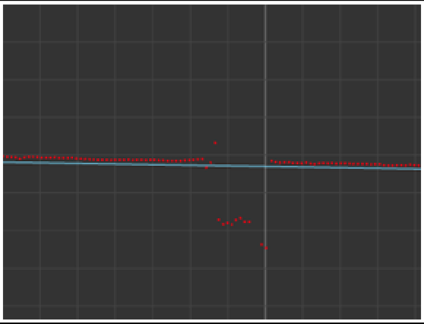
Influence of parameters on the approximation of the profile by arcs:

	
Initial profile	Enable = OFF
	
Enable = ON, Min size = 15, Expected radius = 3...36 mm, Accuracy = 0.1 mm	Expected radius = 8...36 mm
	
Accuracy = 0.5 mm	Accuracy = 0.3 mm
	
Merge threshold = 0.5 mm	Merge threshold = 2 mm

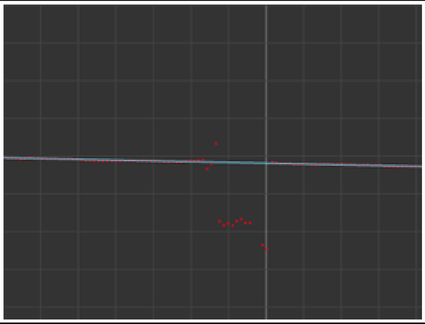
25.2.1.3. Filtering

Accurate approximation of segments and arcs is achieved by eliminating points that strongly deviate from the average approximation values. Such points, as a rule, are outliers lying on glare, re-reflections, spurious illumination, and lead to a decrease in the approximation accuracy. Exclusion of outliers (for both segments and arcs) is performed when the **Outliers filtering** parameter is enabled. In this case, the permissible deviation of a point from the mean is determined by the **Line filtering threshold** and **Arc filtering threshold** parameters for segments and for arcs, respectively.

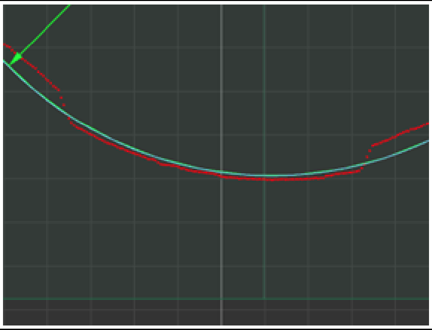
Influence of approximation parameters on the result:



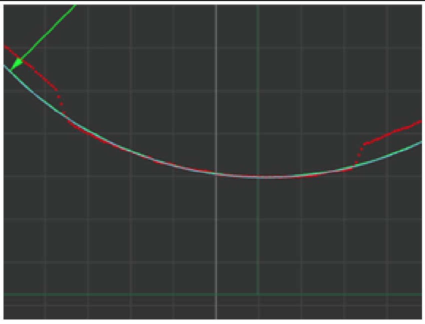
Outliers filtering = OFF



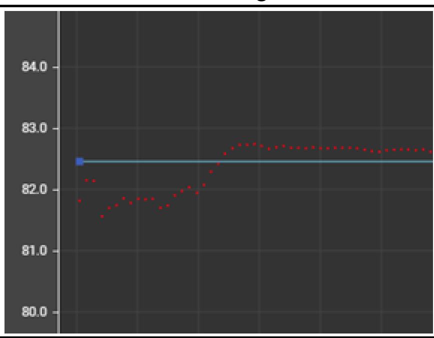
Outliers filtering = ON



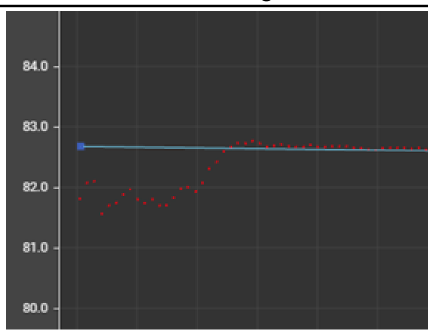
Outliers filtering = OFF



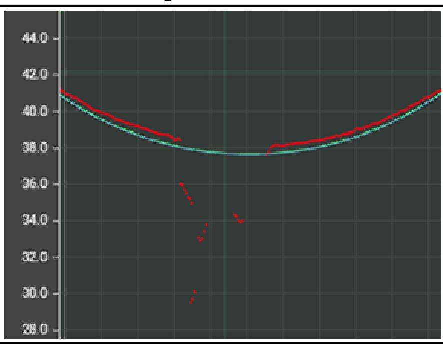
Outliers filtering = ON



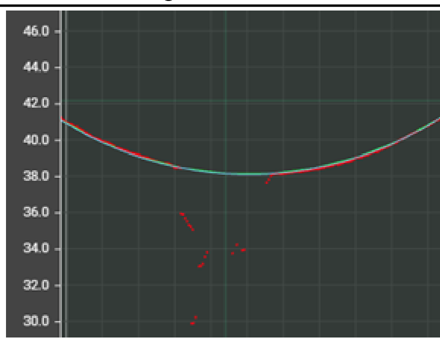
Line filtering threshold = 1.5 mm



Line filtering threshold = 0.5 mm



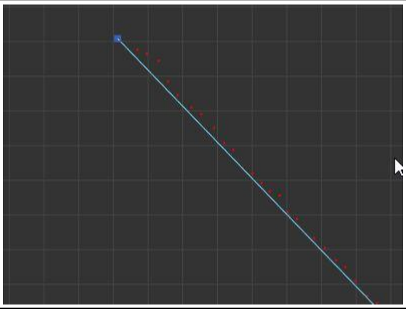
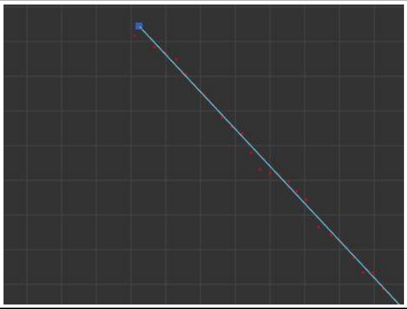
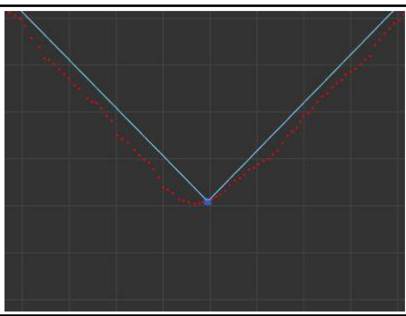
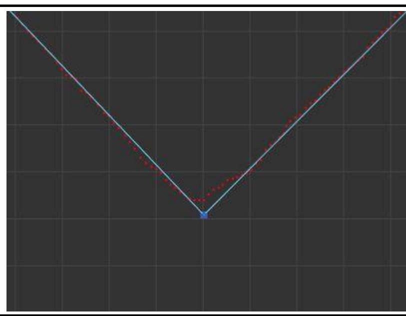
Arc filtering threshold = 15 mm



Arc filtering threshold = 1 mm

25.2.1.4. Clarification of approximating line segments and arcs

This stage includes the precise determination of the coordinates of the beginning and end of the line segments that approximate the profile, and the coordinates of the intersections of the segments. This stage does not contain any parameters.

	
Approximation is disabled - the beginning of the line segment lies at the first point of the profile, which does not allow accurate approximation of all points	Approximation is enabled - the beginning of the line segment is recalculated taking into account the first point of the profile and the optimal approximation of all points of the profile
	
Approximation is disabled - the intersection of the line segments forming the angle lies on the profile point and is displaced relative to the true vertex of the angle	Approximation is enabled - the intersection of the line segments forming the angle is recalculated and lies on the extension of the sides of the angle, which corresponds to the true vertex of the angle

25.2.2. Stage 2. Building a graph

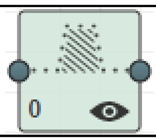
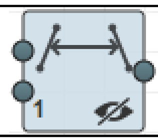
Upon completion of the profile approximation procedure, it is necessary to build a computational graph.

The computation graph is a set of blocks and connections between them. It is recommended (but not necessary) to orient the graph horizontally - smart blocks that extract features from the profile (points, line segments, etc.) are placed on the left. Processing blocks are placed in the middle. On the right are blocks for transmitting measurement results to external systems and receiving messages from external systems.

In the current revision of the firmware, it is possible to save only one graph in the internal memory of the scanner - it is the basic graph, which is loaded when you turn on the scanner and starts working automatically. The graph can be saved to a computer for use in other scanners of the “Smart” series or for use in the future, i.e. the saved graph can be loaded into the scanner as a basic one.

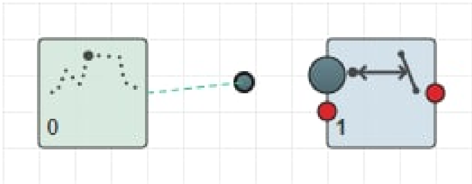
The graph building area also displays the time (in microseconds) spent on profile approximation and graph rendering. If this time is less than 1/(frequency of profiles in seconds), each profile is processed, if greater - some profiles may be skipped and not processed.

Each block of the graph has a unique (within the graph) identifier (number), displayed in the lower right corner of the block. This identifier allows you to quickly match the block and the search area in which it operates. In addition, at the bottom of the block there are some quick access elements for block control. Examples:

	
"Calculate filling" block: identifier "0", results display "On"	"Seg-seg distance" block: identifier "1", results display "Off"

To place a block on the graph, click on it or move it from the **Smart blocks** area to the graph building area.

To make a connection between blocks, drag the output of one block to the input of another block (or several blocks). For convenience, the inputs of blocks to which a connection can be made are increased in size, and inputs to which a connection cannot be made are shown in red:



When the block is placed in the graph area, the search area appears in the measurement results display area. The search area is intended to specify the area in which the selected block operates. You can move and resize the search area with the mouse.

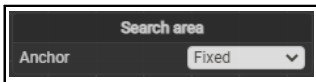
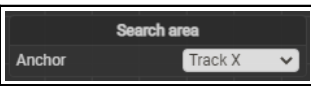
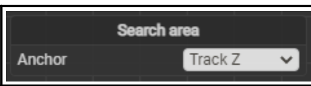
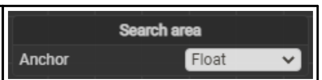
25.2.2.1. Results display area

The area is intended for visual control and customization of smart block search areas. The area also displays the result of profile approximation (a set of approximating line segments) and the results of smart block operation.

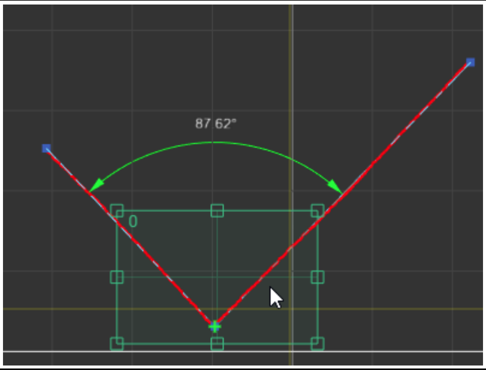
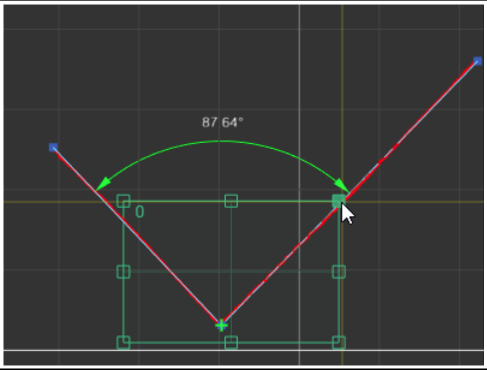
As noted above, some blocks have search areas within which the functions of the block are performed. The use of search areas makes it possible to exclude noise, sensor flare and other factors from calculations. The search area can be:

- fixed,
- tracking along the X coordinate,
- tracking along the Z coordinate, or
- floating, i.e. tracking along both coordinates.

The mode of moving the search area is selected for each area separately in the menu that appears when the area is specified:

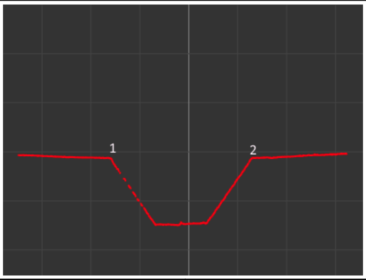
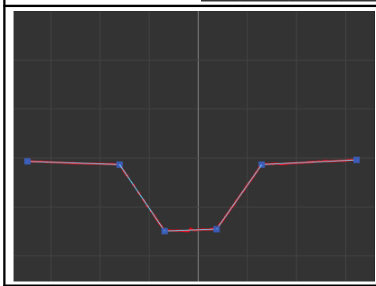
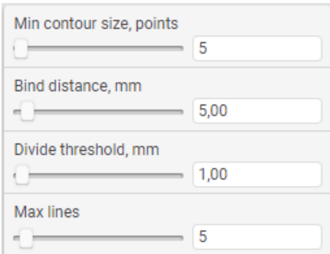
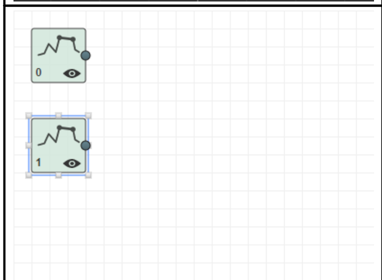
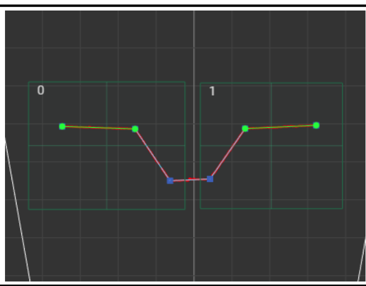
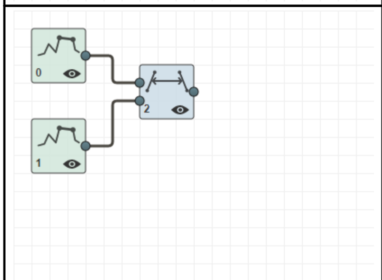
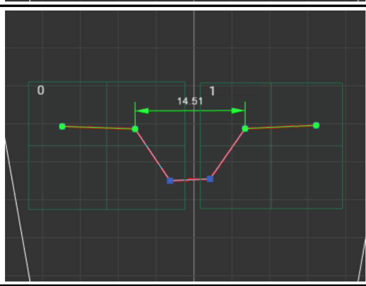
			
Fixed area	Tracking along the X coordinate	Tracking along the Z coordinate	Floating area

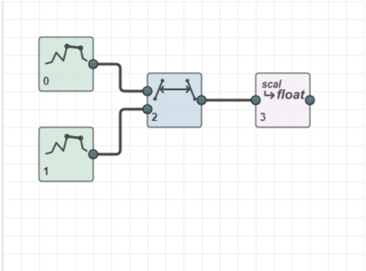
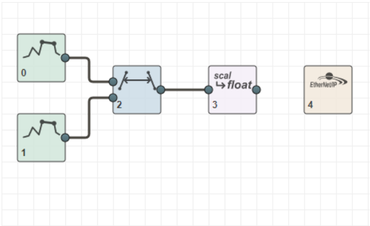
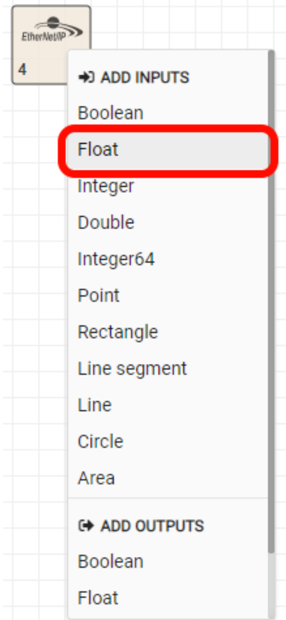
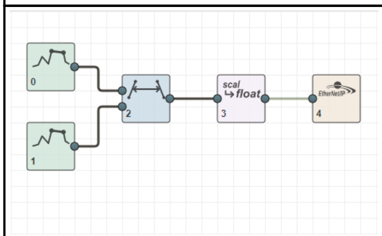
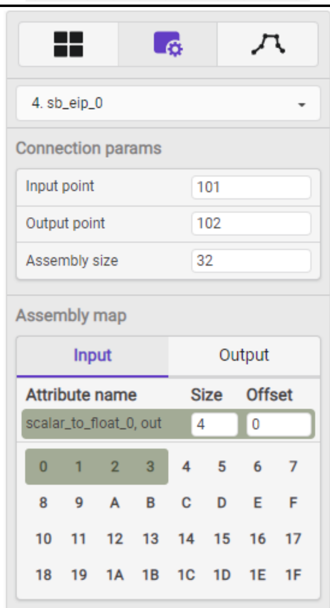
In any mode of moving an area, the user can move and resize the area. Moving is done with the right mouse button (click on the search area and move the mouse). Resizing is carried out using special rectangles located along the perimeter of the search area:

	
Moving the search area	Resizing the search area

25.2.2.2. Example of building a graph

As an illustration of the process of building a graph for solving a specific problem, let's find the distance between points 1 and 2 (the wide base of the trapezoid) on the profile and transfer it to an external controller via the Ethernet/IP protocol.

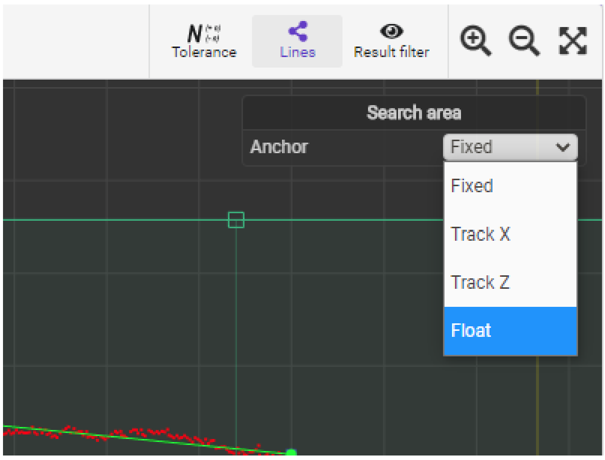
		Source profile. It is necessary to find the distance between points 1 and 2.
		Configure the algorithm for approximating the profile by segments to obtain a stable splitting in accordance with the profile shape. See par. 25.2.1.
		Place two “segment detector” blocks in the graph building area to select the left and right segments. Segments are highlighted by search areas. In the parameters (see par. 24.1.2) for block "0" select "Mode" > "first", for block "1" - "Mode" > "last" (the first block works with the first segment, the second block works with the last segment).
		Place the “seg-seg distance” block in the graph building area. This block is designed to calculate the distance between the right point of the first segment and the left point of the second segment. In the parameters (see par. 25.1.2) for block “2” select “Mode”> “end <-> begin”. Connect the outputs of blocks "0" and "1" with the inputs of block "2".

		Place the “scal-> float” block in the graph building area to convert the scalar value (internal data type) of the distance to the floating point format (float 4 bytes).
		Place the “eip” block in the graph building area. This block is designed to transfer the calculated value via the Ethernet/IP protocol. Using the context menu of the “eip” block, add an input of the “float” type.
		After connecting the output of the “scal-> float” block with the input of the “eip” block, the parameters of the “eip” block will display the placement of data from the block input in the input assembly (this data is sent to the network). The location of the data within the assembly can be changed by dragging.

25.2.3. How it works

The part to be inspected is placed in the scanner's field of view. The graph is calculated for each part profile generated by the scanner.

First of all, the first and last line segments are selected on the profile, using the search areas of the “segment detector” blocks. Each of the search areas can be either "tracking" (along X, along Z or along both coordinates at once) or "fixed" (by default). The behavior for each search area can be set by a parameter that appears when you click on the corresponding search area:



Search area behavior selection menu
(fixed, tracking along X, tracking along Z, tracking along both coordinates)

If the search areas are the tracking ones, then when moving the part in the working range of the scanner, they automatically change their position so that the detected segment is in the center of the search area. When the part disappears (there are no segments), the tracking search areas return to the coordinates specified by the user when setting up the graph (i.e., to the initial position).

The selected segments are sent to the "seg-seg distance" block configured to calculate the distance between the right point of the first segment and the left point of the last segment. If one of the segments (or both) is not detected, the block returns a value with the flag "result is not valid" (the internal representation of the distance of SDT_SCALAR type is used, see par. 25.3.1).

Then the internal representation of the distance is converted to the general type "float" ("scal-> float" block) for further output to the assembly (in Ethernet/IP terminology). If the distance is "not valid" at the input, the result will be "NaN" (Not a Number).

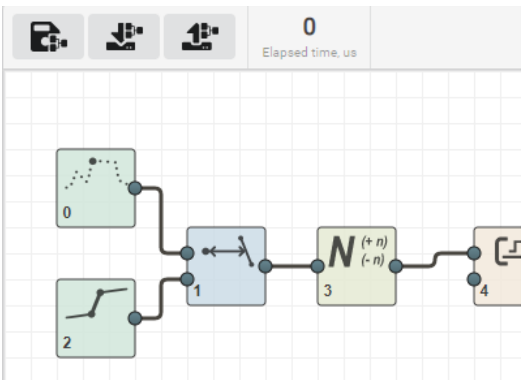
The last block ("eip") receives the distance of the general type "float", and places it in the input assembly, which sends the data to the network.

As a result, an external system ("adapter" - in Ethernet/IP terminology) can connect to the scanner and get the calculated value of the distance.

Video demonstration: https://youtu.be/-KvKu5MQ6JM

25.2.4. Saving and loading Smart functions

To save/load a Smart function, use the corresponding buttons located in the upper part of the graph building area:



Button	Assignment
	Saving the current graph in the non-volatile memory of the scanner. The graph saved in nonvolatile memory is automatically loaded when the scanner is turned on.
	Downloading the current graph from the scanner for saving it to the PC. The saved graph can later be used on other scanners of the "Smart" series.
	Uploading a graph from the PC into the scanner. The uploaded graph replaces the current graph and starts calculating automatically.

25.3. Smart blocks

25.3.1. Data types

Each smart block operates with a certain type (several types) of data representing measurement results, logic signals, etc. Byte order (unless otherwise noted) is LITTLE-ENDIAN. Description of data types is given in the table:

Name	Type	Description
internal types		Internal types are used to transfer information within a graph. As a rule, they are composite (contain several fields) and in general should not be used to input and output data from/to external systems (EthernetIP, UDP, etc.).
SDT_SCALAR	scalar_t	Scalar. It represents a single measurement/signal value. In the current revision, it corresponds to the "float" data type. In future revisions, this may change.

SDT_POINT	point_t	Point. In the current revision, it has the following structure:
		{
		float x;
		float z;
		}

SDT_RECT	rect_t	Rectangle. In the current revision, it has the following structure:
		{
		point_t topLeft;
		float w;
		float h;
		}

SDT_SEGMENT	segment_t	Line segment. In the current revision, it has the following structure:
		{
		point_t p1;
		point_t p2;
		}

SDT_LINE	line_t	Line. In the current revision, it has the following structure:
		{
		float a;
		float b;
		float c;
		}

SDT_CIRCLE	circle_t	Circle. In the current revision, it has the following structure:
		{
		point_t center;
		float r;
		}

SDT_POINT_3D	point_3d_t	Point in 3D space. In the current revision, it has the following structure:
		{
		double x, y, z;
		double dummy = 1.0;
		}

SDT_EULER_3D	euler_3d_t	Euler angles in 3D space. In the current revision, it has the following structure: { euler_3d_order_e order;

<table>
<tr><th>Name</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td>double ax, ay, az;
}
typedef enum{
EULER_XYZ = 0 << 0 | 1 << 2 | 2 << 4,
EULER_XZY = 0 << 0 | 2 << 2 | 1 << 4,
EULER_YZX = 1 << 0 | 2 << 2 | 0 << 4,
EULER_YXZ = 1 << 0 | 0 << 2 | 2 << 4,
EULER_ZXY = 2 << 0 | 0 << 2 | 1 << 4,
EULER_ZYX = 2 << 0 | 1 << 2 | 0 << 4
}euler_3d_order_e;</td></tr>
<tr><td>SDT_POSE_3D</td><td>pose_3d_t</td><td>The position of an external device (for example, TCP robot) in 3D space. Sets the position and tilt angles. In the current revision, it has the following structure:
{
rot_order_e order;
double x, y, z, w, p, r;
}
order - specifies the order in which the rotation is applied. Valid values: ROT_XYZ_WPR = 0, ROT_ZYX_RPW = 1.</td></tr>
<tr><td>SDT_CST_3D</td><td>cst_3d_t</td><td>Data for coordinate system transformation. Makes it possible to convert 2D coordinates of objects (points, lines, etc.) into 3D coordinates. It is formed after calibration by the “cst calibration” smart block. In the current revision, it has the following structure:
{
matrix_4x4_t m;
pose_3d_t pose;
}
m - 4x4 matrix of "double" elements.</td></tr>
<tr><td>common types</td><td colspan="2">Common types are used to transmit data to external (in relation to the scanner) devices and receive data from them. They are used in conjunction with special conversion blocks.</td></tr>
<tr><td>SDT_BOOL</td><td>bool_t</td><td>A logical value that has two mutually exclusive states "TRUE" and "FALSE". It corresponds to the "uint8" type with the following encoding scheme:
0 - “FALSE”;
other - “TRUE”.</td></tr>
<tr><td>SDT_FLOAT</td><td>float</td><td>Single-precision floating-point value (size - 4 bytes).</td></tr>
<tr><td>SDT_INT16</td><td>int16_t</td><td>Signed integer value (size - 2 bytes).</td></tr>
<tr><td>SDT_INT32</td><td>int32_t</td><td>Signed integer value (size - 4 bytes).</td></tr>
<tr><td>SDT_DOUBLE</td><td>double</td><td>Double-precision floating-point value (size - 8 bytes).</td></tr>
<tr><td>SDT_INT64</td><td>int64_t</td><td>Signed integer value (size - 8 bytes).</td></tr>
</table>

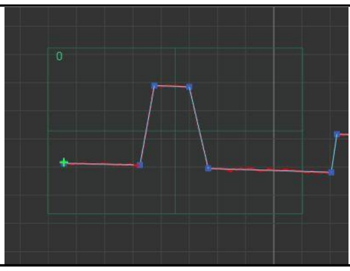
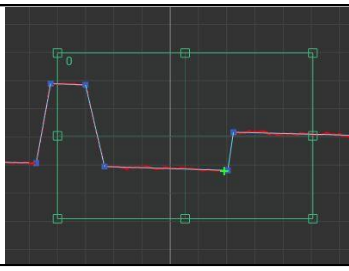
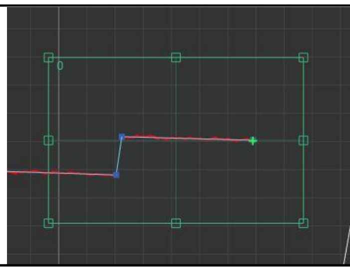
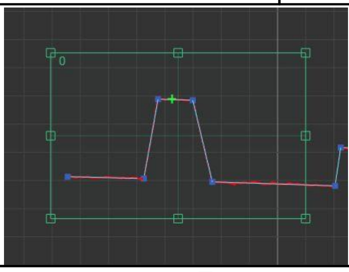
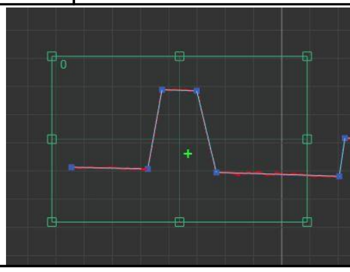
25.3.2. Sections

Smart blocks are grouped into the following sections:

1. "Feature detectors" - Smart blocks designed to extract primitives (points, lines, angles, etc.) from a profile.
2. "Math functions" - Smart blocks that perform mathematical operations on primitives (calculation of distances and angles, conversion of units of measurement, etc.).
3. "Welding" - Smart blocks designed to solve welding tasks, such as tracking weld grooves, etc.
4. "Converters" - Smart blocks for performing conversions (conversion of types, conversion of units of measurement, merging of primitives, decomposition of primitives, etc.).
5. "Control" - Smart blocks for controlling the measured values within tolerances.
6. "Base IO" - Smart blocks for data exchange with external systems (available without a license).

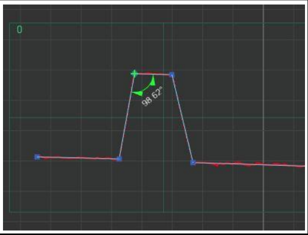
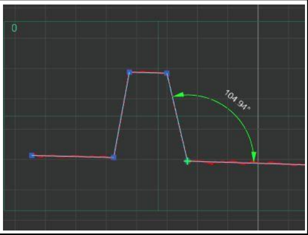
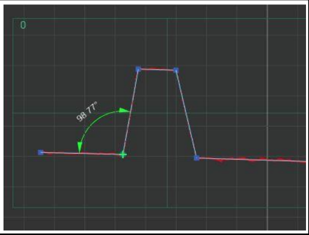
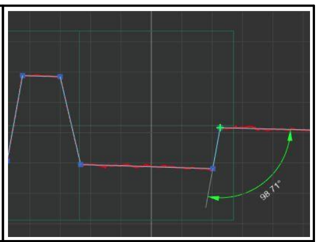
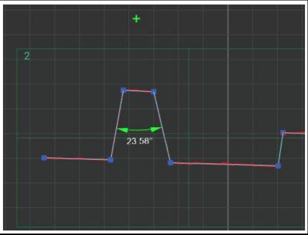
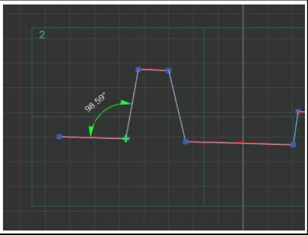
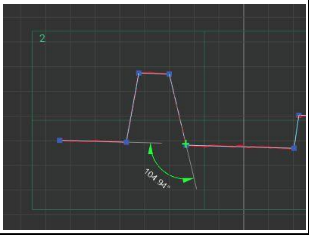
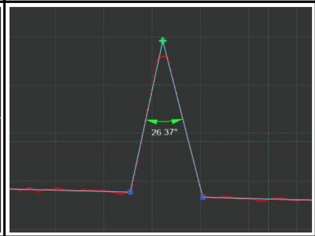
7. "Industrial IO" - Smart blocks for data exchange with industrial automation systems (available under license).
8. "Robots IO" - Smart blocks for data exchange with industrial robot controllers (available under license).

25.3.2.1. "Feature detectors" section

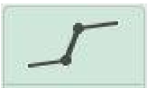
	“point detector” - searching for a profile point.		
Parameters:	“Mode”	min X	The profile point with the minimum X coordinate.
		min Z	The profile point with the minimum Z coordinate.
		max X	The profile point with the maximum X coordinate.
		max Z	The profile point with the maximum Z coordinate.
		average	The profile point with averaged coordinates.
Examples:			
			
min X	min Z	max X	
			
max Z	average		
Outputs:	“pos”	SDT_POINT	Point coordinates.
	“angle detector” - searching for the angle between two line segments of the profile.		
Parameters:	“Mode”	top	The angle whose vertex has the maximum Z coordinate.
		bottom	The angle whose vertex has the minimum Z coordinate.
		left	The angle whose vertex has the minimum X coordinate.
		right	The angle whose vertex has the maximum X coordinate.
		value	The first angle satisfying the "Angle average" and "Angle tolerance" parameters.
	“Type”	basic	Basic angle.
		suplement	Supplementary angle.
		explement	Explementary angle.
		sup-exp	Supplementary explementary angle.

	"Angle value"	0...179	Angle value.
	"Angle tolerance"	0...89	The maximum permissible deviation (in both directions) from the desired angle.

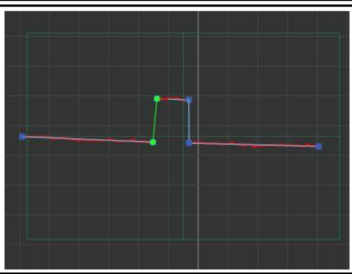
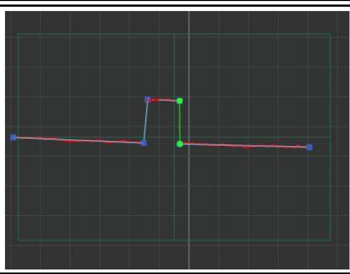
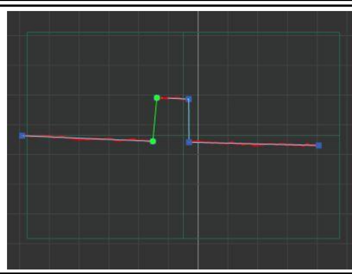
Examples:

			
top	bottom	left	right
			
"Angle value" = 24, "Angle tolerance" = 2	"Angle value" = 99, "Angle tolerance" = 2	"Angle value" = 105, "Angle tolerance" = 2	"Angle value" = 25, "Angle tolerance" = 2

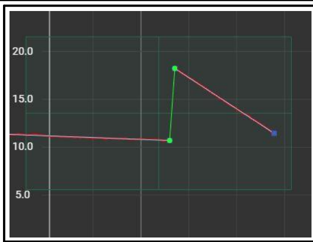
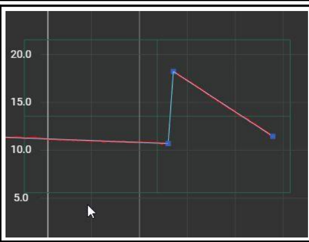
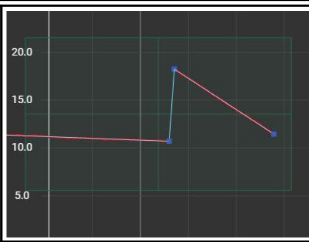
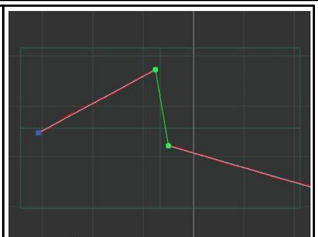
Outputs:	"pos"	SDT_POINT	Vertex coordinates.
	"angle"	SDT_SCALAR	The angle value in degrees.

 edge detector	"edge detector" - searching for the difference in height between two segments of the profile.

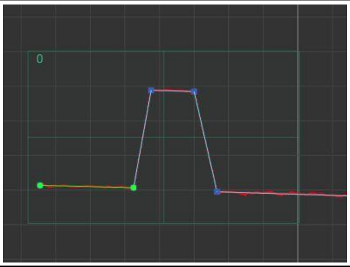
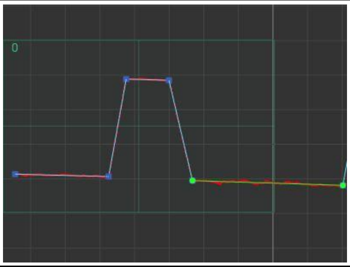
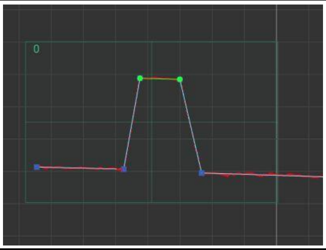
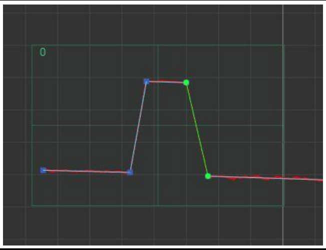
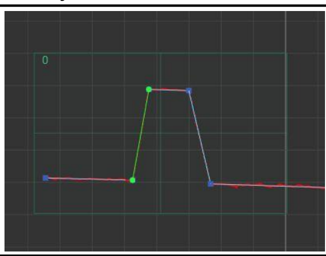
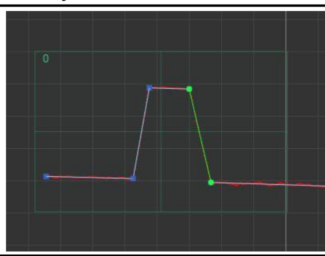
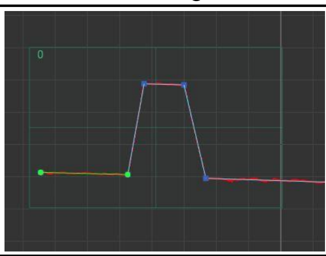
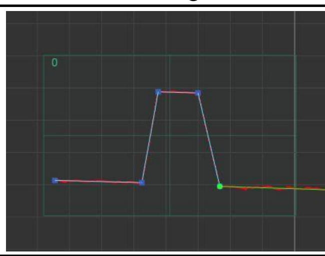
Parameters:	"Mode"	rise	Height difference "rise".
		fall	Height difference "fall".
		any	Any direction of height difference. The choice is made by height difference.

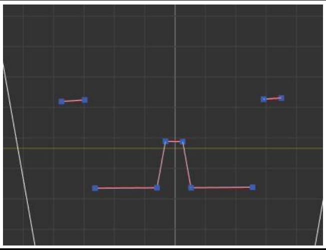
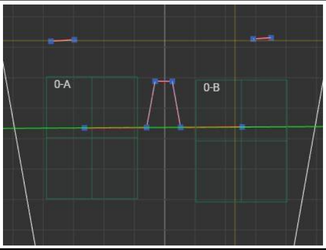
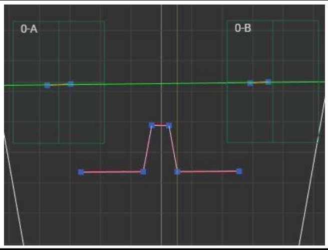
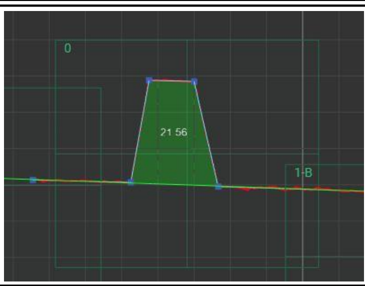
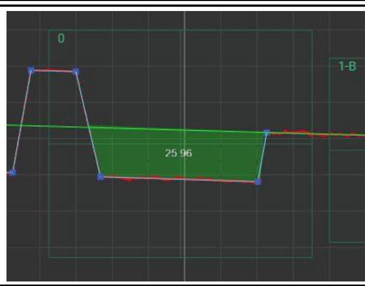
		
rise	fall	any

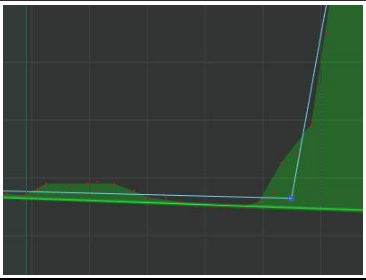
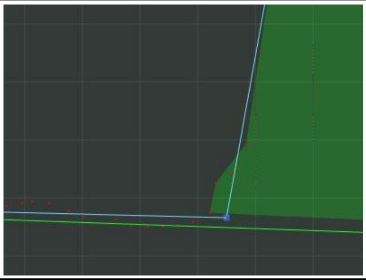
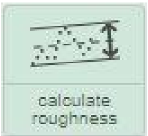
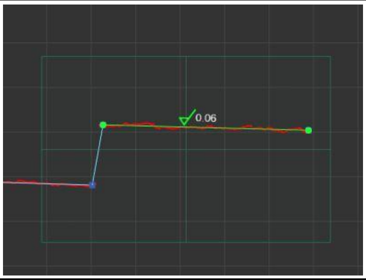
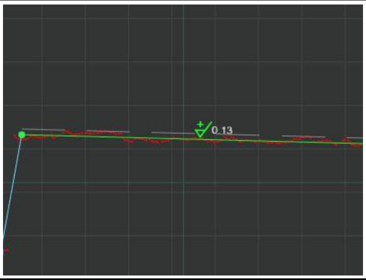
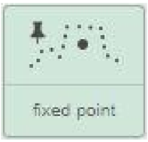
	"Min step", mm	0.01...1000	The minimum allowable height difference for detection. The direction is not taken into account.
	"Angle tolerance", deg	0...45	The maximum permissible angle between two segments forming a difference in height.

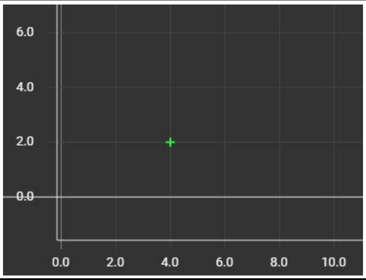
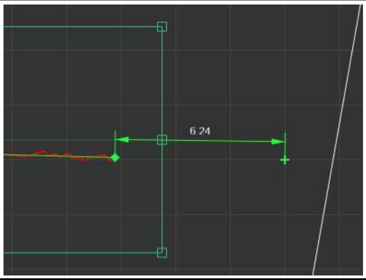
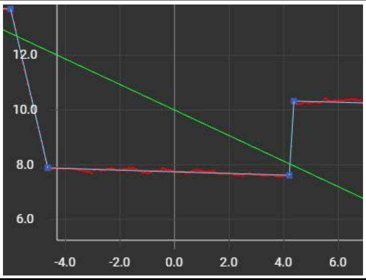
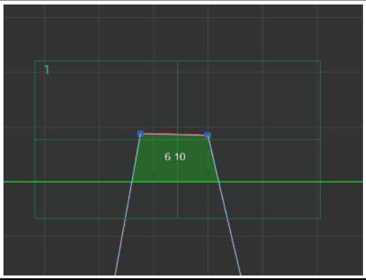
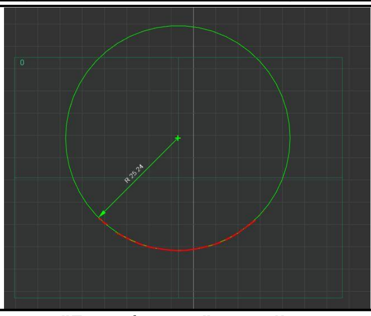
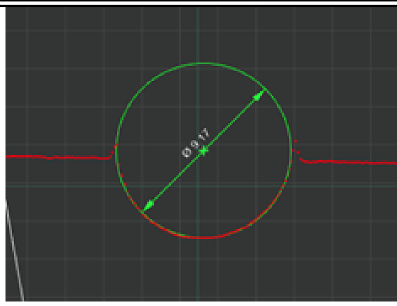
			
"Min step" = 5 mm "Angle tolerance" = 35°	"Min step" = 10 mm "Angle tolerance" = 35°	"Min step" = 10 mm "Angle tolerance" = 30°	"Min step" = 5 mm "Angle tolerance" = 45°

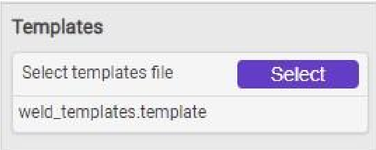
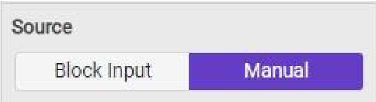
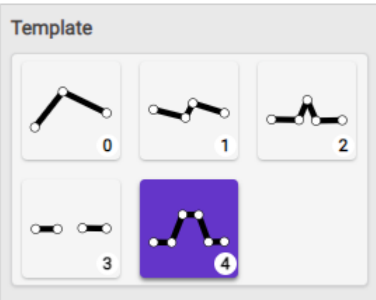
Outputs:	“edge”	SDT_SEGMENT	The segment corresponding to the height difference.
	“segment detector” - searching for a line segment on the profile. It is performed from left to right (from negative X values to positive X values).		
Parameters:	“Mode”	first	The leftmost segment.
		last	The rightmost segment.
		by index	The segment with the index specified by the “Index” parameter.
		rise edge	The central segment of the "edge" with the "Rise" height difference.
		fall edge	The central segment of the "edge" with the "Fall" height difference.
		first, constr.	The first segment that simultaneously satisfies the conditions of the parameters “Min len”, “Max len”, “Angle” and “Angle tolerance”.
		last, constr.	The last segment that simultaneously satisfies the conditions of the parameters “Min len”, “Max len”, “Angle” and “Angle tolerance”.

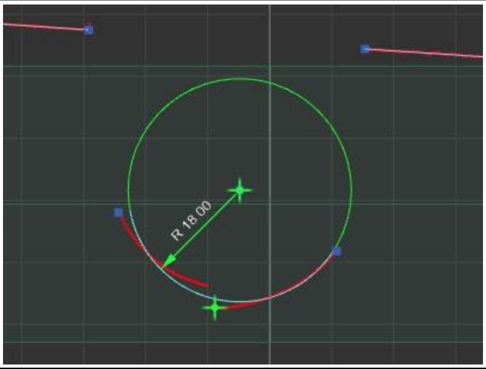
	
first	last
	
by index, “Index” = 2	by index, “Index” = 3
	
rise edge	fall edge
	
first, constr	last, constr

	"Index"	0...256	The segment index used in the "Mode" = "by index".
	"Min len", mm	0.01...1000	The minimum segment length used in modes "Mode" = "first, constr." and “Mode” = “last, constr.”.
	"Max len", mm	0.01...1000	The maximum segment length used in modes "Mode" = "first, constr." and “Mode” = “last, constr.”.
	"Angle", deg	-90...90	The angle of inclination of the segment relative to the horizontal axis. It is used in modes "Mode" = "first, constr." and “Mode” = “last, constr.”
	"Angle tolerance", deg	0...89	The tolerance (in both directions) of the angle of inclination of the segment relative to the horizontal axis. It is used in modes "Mode" = "first, constr." and “Mode” = “last, constr.”
Outputs:	“seg”	SDT_SEGMENT	The segment corresponding to the block parameters.
	“line approximation” - line approximation of profile points (in two areas).		
			
original profile without approximation	approximation of lower fragments	approximation of upper fragments	
Outputs:	“line”	SDT_LINE	The line that approximates the points located in areas.
	“calculate filling” - calculating the total area of the profile deviation from the base input line.		
Parameters:	“Mode”	above	Take into account the points lying above the baseline.
		below	Take into account the points lying below the baseline.
			
above	below		
Parameters:	“Threshold”, mm	0.01...100	The threshold of the deviation from the baseline. The deviations less than the threshold are not taken into account - they are considered noise.

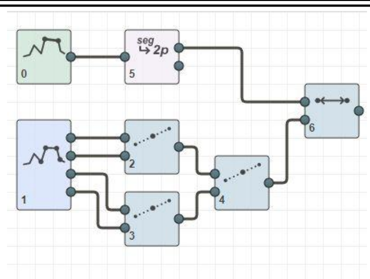
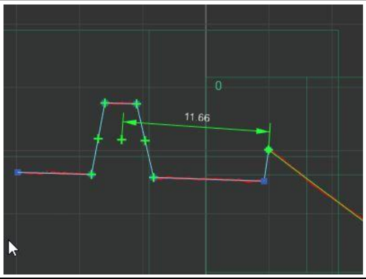
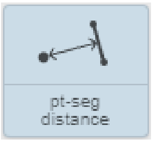
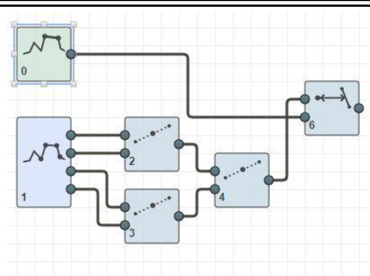
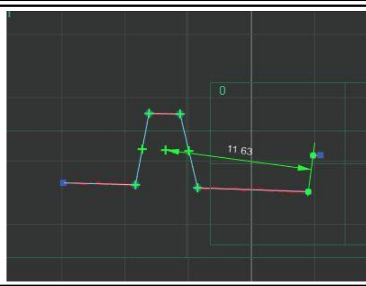
			
“Threshold” = 0.01 mm		“Threshold” = 0.1 mm	
Inputs:	“in1”	SDT_LINE	The baseline against which the deviations are analyzed.
Outputs:	“area”	SDT_SCALAR	The total area of deviations, mm².
	“calculate roughness” - calculating the roughness of the profile relative to the input baseline.		
Parameters:	“Mode”	std. dev.	The standard deviation.
		pos dev	The maximum positive deviation (upward relative to the base segment).
		neg dev	The maximum negative deviation (downward relative to the baseline).
			
std. dev.	pos dev	neg dev	
	“Left margin”, mm	0...100	An offset from the left edge of the segment whose points are not used in the calculation.
	“Right margin”, mm	0...100	An offset from the right edge of the segment whose points are not used in the calculation.
Inputs:	“in1”	SDT_SEGMENT	The base segment with respect to which the roughness is analyzed.
Outputs:	“value”	SDT_SCALAR	The roughness value, mm².
	“fixed point” - the fixed point (the position does not depend on the profile).		
Parameters:	“Position by X”, mm	0.01...1000	The position of the point along the X axis.
	“Position by Z”, mm	0.01...1000	The position of the point along the Z axis.

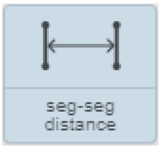
			
“Position by X” = 4 mm, “Position by Z” = 2 mm		Measuring the distance between the profile point and the fixed point	
Outputs:	“pos”	SDT_POINT	The point with given coordinates.
	“fixed line” - the fixed line (the position does not depend on the profile).		
Parameters:	“Slope”, deg	-90...90	The slope of the line relative to the horizontal axis.
	“Position by Z”, mm	0.01...1000	The position of the line relative to the Z axis.
			
“Slope” = 25°, “Position by Z” = 10 mm		Measuring the filling area relative to the fixed line	
Outputs:	“line”	SDT_LINE	The line with the specified parameters.
	“circle detector” - searching for a circle on a profile.		
Parameters:	"Mode"	arc	Detection from approximated profile arcs.
		segments	Detection from approximated profile segments.
	"Result type"	radius	The output is a radius.
		diameter	The output is a diameter.
	"Expected value", mm	0.01...1000	Circle radius/diameter.
	"Tolerance", mm	0.01...1000	Radius/diameter tolerance (both directions).
			
"Result type" = radius "Expected value" = 25 mm "Tolerance" = 5 mm		"Result type" = diameter "Expected value" = 10 mm "Tolerance" = 1 mm	

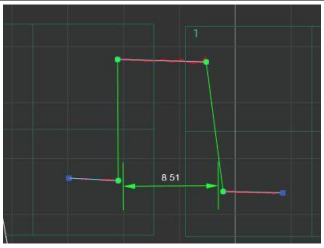
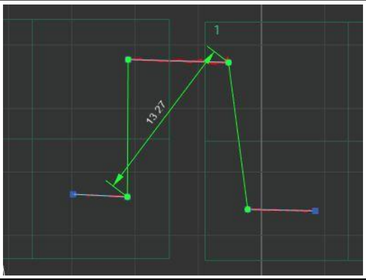
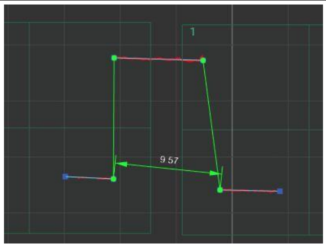
Outputs:	"result"	SDT_SCALAR	The value of the radius/diameter of the circle in mm.
	"center"	SDT_POINT	Circle center coordinates.
	"templates detector" - searching for patterns in the results of profile approximation. How to work with template files, see Annex 5.		
Parameters:			Selecting a file with a set of templates that was prepared in the template editor.
			Template switching mode.
		Block input	The template number is set by the block input (an external system can control the template number).
		Manual	The template number is set manually by the operator using the web interface.
			Template selection area in "Manual" mode.
Inputs:	"idx"	SDT_INT32	Template index (in "Block input" mode).
Outputs:	"det"	SDT_BOOL	Selected template detection flag.
	"idx"	SDT_INT32	Selected template index.
	dynamic	SDT_POINT, SDT_SEGMENT	They are created after loading a file with a set of templates.
	"overlap detector" - searching for a step on an arc.		
Parameters:	"Radius", mm	0.01...1000	Expected arc radius.
	"Tolerance", mm	0.01...1000	Tolerance for the expected radius (in both directions).
	"Radius step", mm	0.01...1000	Amplitude of change (step) of the radius.
	"Angle offset", deg	-360…360	Additional angular displacement of the direction by a step, allowing to take into account the position of the scanner in the system.

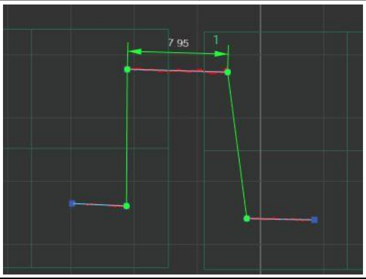
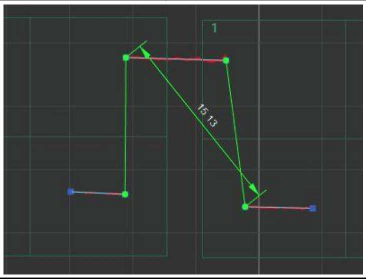
	 "Radius" = 20 mm, "Tolerance" = 10 mm, "Radius step" = 0.5 mm, "Angle offset" = 0°		
Outputs:	"radius"	SDT_SCALAR	The actual value of the arc radius.
	“center”	SDT_POINT	Arc center coordinates.
	"overlap”	SDT_POINT	Step midpoint coordinates.
	"angle"	SDT_SCALAR	Angular direction from the center of the arc.

25.3.2.2. "Math functions" section

	“point to point distance” - calculating the distance between two points arriving at the block inputs.		
			
Measuring the distance between the center of the trapezoid and the start of the line segment			
Inputs:	“in1”	SDT_POINT	Point #1.
	“in2”	SDT_POINT	Point #2.
Outputs:	“dist”	SDT_SCALAR	The distance between points in mm.
	“point to segment distance” - calculating the distance between a point and a segment arriving at the block inputs.		
			
Measuring the distance between the center of the trapezoid and a vertical line segment			
Inputs:	“in1”	SDT_POINT	Point.
	“in2”	SDT_SEGMENT	Segment.
Outputs:	“dist”	SDT_SCALAR	The distance between a point and a segment in mm.

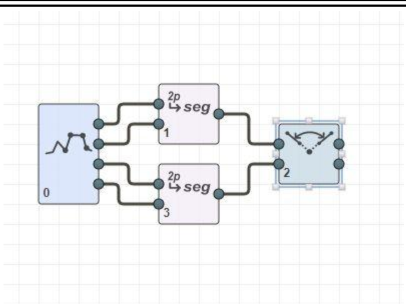
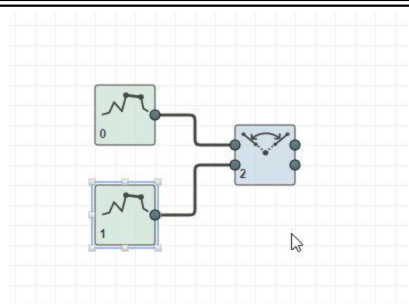
	"segment to segment distance" - calculating the distance between two segments arriving at the block inputs.		
Parameters:	"Mode"	perpendicular	The perpendicular distance between parallel segments.
		begin<->begin	The distance between the beginnings of the segments (leftmost points).
		begin<->end	The distance between the beginning of one segment (leftmost point) and the end of another segment (rightmost point).
		end<->begin	The distance between the end of one segment (rightmost point) and the beginning of another segment (leftmost point).
		end<->end	The distance between the ends of the segments (rightmost points).

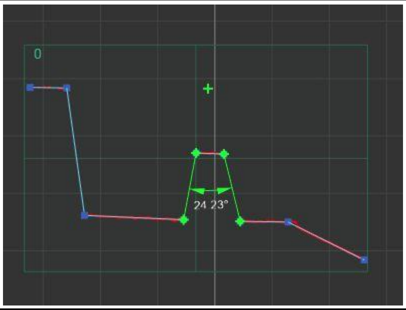
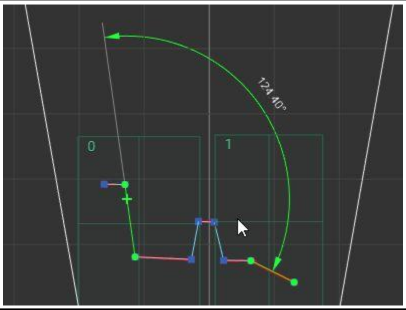
		
"Mode" = perpendicular	"Mode" = begin<->begin	"Mode" = begin<->end

	
"Mode" = end<->begin	"Mode" = end<->end

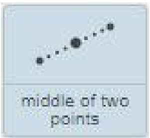
Inputs:	"in1"	SDT_SEGMENT	Segment #1.
	"in2"	SDT_SEGMENT	Segment #2.
Outputs:	"dist"	SDT_SCALAR	The distance between segments in mm.

	"segments intersections" - calculating the center of intersection of segments and the angle between them.

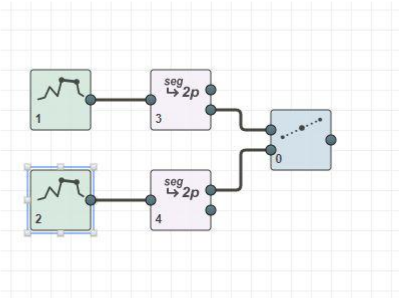
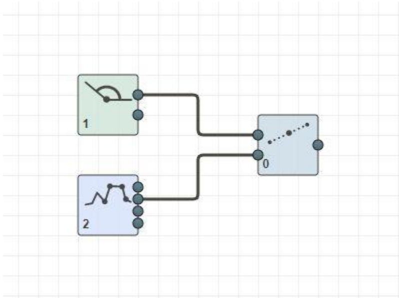
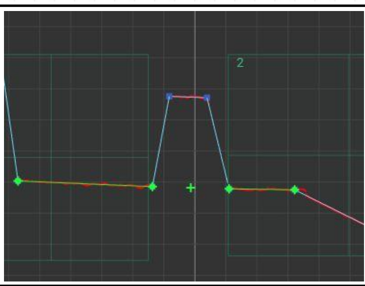
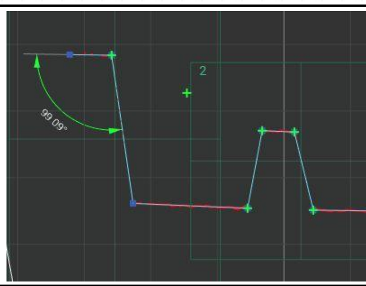
	

	
Measuring the angle and determining the point of intersection of the side segments of the trapezoid	Measuring the angle and determining the point of intersection of arbitrary segments on the profile

Inputs:	“in1”	SDT_SEGMENT	Segment #1.
	“in2”	SDT_SEGMENT	Segment #2.
Outputs:	“point”	SDT_POINT	The intersection point of two segments.
	"angle"	SDT_SCALAR	The angle between segments in degrees.

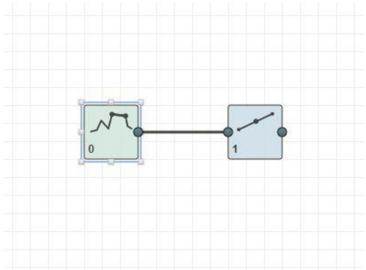
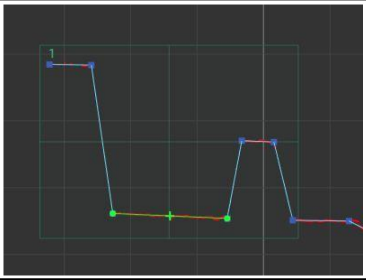
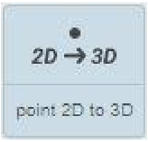
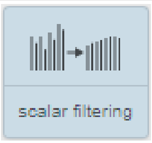


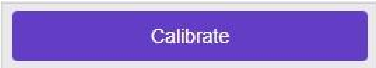
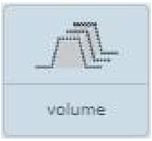
“middle of two points” - calculating the midpoint between two points.

	
	
Calculating the midpoint between the endpoints of two segments	Calculating the midpoint between the vertex of the angle and one of the points of the trapezoid

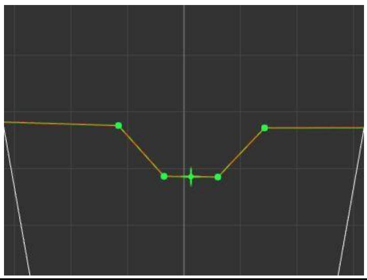
Inputs:	“in1”	SDT_POINT	Point #1.
	“in2”	SDT_POINT	Point #2.
Outputs:	“middle”	SDT_POINT	Midpoint.

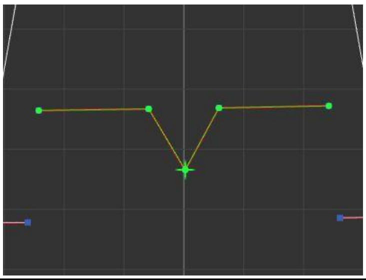
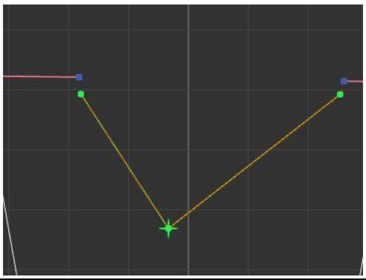
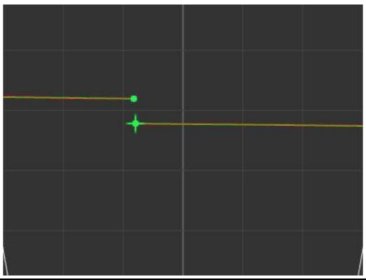
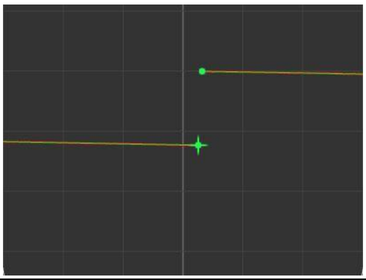
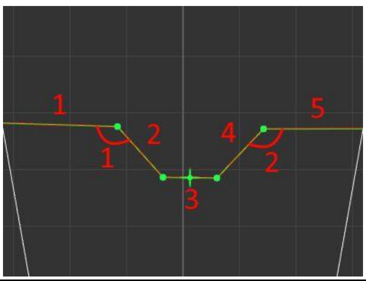
“middle of line segment” - calculating the midpoint of the line segment.

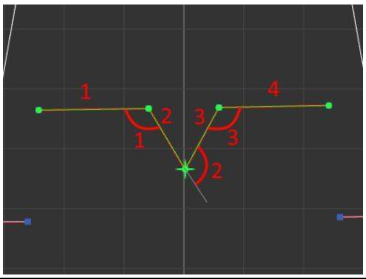
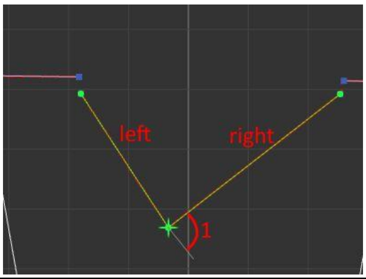
			
Calculating the center of an arbitrary segment			
Inputs:	"in1"	SDT_SEGMENT	Segment.
Outputs:	"middle"	SDT_POINT	Midpoint.
	"point 2D to 3D" - converting a point from the local 2D coordinate system of the scanner to the 3D coordinate system of an external device. The conversion is performed according to the following expressions: $X = X_0 + x*A_0 + y*A_1$ $Y = Y_0 + x*A_2 + y*A_3$ $Z = Z_0 + x*A_4 + y*A_5$ where: X, Y, Z - coordinates of a point in the 3D coordinate system of an external device; X_0, Y_0, Z_0 - calibration offsets; A[6] - rotation matrix coefficients; x, y - coordinates of a point in the 2D coordinate system of the scanner.		
Inputs:	"in"	SDT_POINT	Point.
Outputs:	"x"	SDT_SCALAR	The X coordinate in the 3D coordinate system of the external device.
	"y"	SDT_SCALAR	The Y coordinate in the 3D coordinate system of the external device.
	"z"	SDT_SCALAR	The Z coordinate in the 3D coordinate system of the external device.
	"scalar filtering" - filtering incoming scalar values. Pre-filtering is performed by the median filter specified by the "Median filter" parameter. Smoothing of values can be performed by simple averaging or bilateral filter (parameters "Smoothing filter" and "Filter size").		
Parameters:	"Median filter"	disabled	Median filtering is not performed.
		3 values	Median filtering by 3 values.
		5 values	Median filtering by 5 values.
		7 values	Median filtering by 7 values.
	"Smoothing filter"	average	Smoothing (if the "Filter size" parameter != disabled) is performed by the averaging filter.
		bilateral	Smoothing (if the "Filter size" parameter != disabled) is performed by the bilateral filter.
	"Filter size"	disabled	Smoothing is not performed.
		3 values	Smoothing by 3 values.
		5 values	Smoothing by 5 values.
		7 values	Smoothing by 7 values.
		9 values	Smoothing by 9 values.
		11 values	Smoothing by 11 values.
		13 values	Smoothing by 13 values.
		15 values	Smoothing by 15 values.
Inputs:	"in1"	SDT_SCALAR	Input value for filtering.
Outputs:	"out"	SDT_SCALAR	Filtered value.

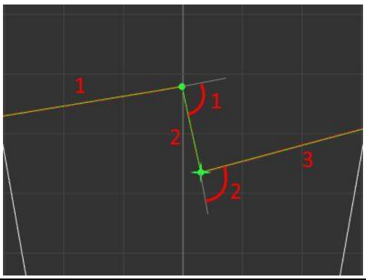
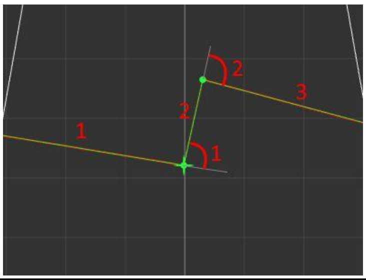
	"cst calibration" - calibration (collection of initial data and calculation) of the matrix for transforming the coordinate system from 2D to 3D, taking into account the position of the scanner on the robot. All calibration actions are performed in a separate window and are described in Annex 8.		
Parameters:			Clicking this button opens the calibration window.
Inputs:	"pose"	SDT_POSE_3D	The current TCP position of the external device (robot).
Outputs:	"cst"	SDT_CST_3D	The data needed to perform the coordinate system transformation.
	"volume" - volume calculation. The volume is calculated by successive summation of the multiplication results of the areas of individual sections and the distance traveled by the object, which was measured by the encoder.		
Parameters:	Encoder step", mm	0.001…100	Encoder step in mm.
Inputs:	"area"	SDT_SCALAR	The input value of the section area (as a rule, it should be obtained from the output of the "calculate filling" block).
Outputs:	"cst"	SDT_SCALAR	The output value of the volume (it is updated after the object disappears from the working range of the scanner).
	"segments to baseline" - recalculation of one common approximated line for several segments. The number of segments is arbitrary and is determined by the number of added dynamic inputs.		
Inputs:	dynamic	SDT_SEGMENT	Input segments.
Outputs:	"baseline"	SDT_LINE	A line approximating the segments.

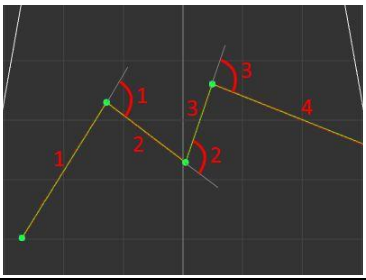
25.3.2.3. "Welding" section

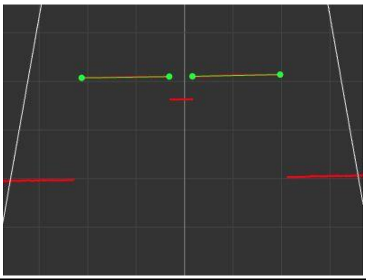
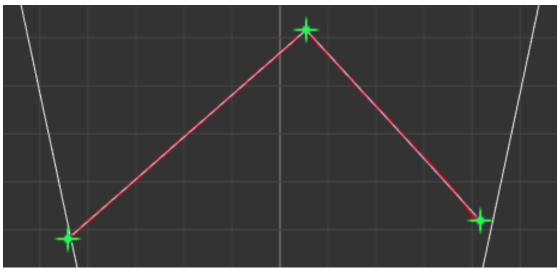
	"templates set" - a set of templates for robotic welding. The template can be set by the user through the block parameters or by an external system using a special block input.		
Parameters:	"General"		Group of general block parameters.
	"Seam type"	trapeze groove	Trapezoidal groove detection: 
		v-groove	V-groove detection:

			
		fillet	Fillet weld detection: 
		lap left	Lap joint detection (the high part is on the left): 
		lap right	Lap joint detection (the high part is on the right): 
	"Trapeze groove":		Group of parameters for the "trapeze groove" template. Designation of lines and angles: 
	"Min len (segment 1)", mm "Max len (segment 1)", mm	0.1...1000 0.1...1000	Minimum and maximum length of segment #1.
	"Min len (segment 2)", mm "Max len (segment 2)", mm	0.1...100 0.1...100	Minimum and maximum length of segment #2.
	"Min len (segment 3)", mm "Max len (segment 3)", mm	0.1...100 0.1...100	Minimum and maximum length of segment #3.

	“Min len (segment 4)”, mm “Max len (segment 4)”, mm	0.1...100 0.1...100	Minimum and maximum length of segment #4.
	“Min len (segment 5)”, mm “Max len (segment 5)”, mm	0.1...1000 0.1...1000	Minimum and maximum length of segment #5.
	“Angle #1”, deg “Angle #1 tolerance”, deg	-90…-10 0...45	Value and permissible deviation of angle #1.
	“Angle #2”, deg “Angle #2 tolerance”, deg	-90…-10 0...45	Value and permissible deviation of angle #2.
	“V-groove”:		Group of parameters for the "v-groove" template. Designation of lines and angles: 
	“Min len (segment 1)”, mm “Max len (segment 1)”, mm	0.1...1000 0.1...1000	Minimum and maximum length of segment #1.
	“Min len (segment 2)”, mm “Max len (segment 2)”, mm	0.1...100 0.1...100	Minimum and maximum length of segment #2.
	“Min len (segment 3)”, mm “Max len (segment 3)”, mm	0.1...100 0.1...100	Minimum and maximum length of segment #3.
	“Min len (segment 4)”, mm “Max len (segment 4)”, mm	0.1...1000 0.1...1000	Minimum and maximum length of segment #4.
	“Angle #1”, deg “Angle #1 tolerance”, deg	-90…-10 0...45	Value and permissible deviation of angle #1.
	“Angle #2”, deg “Angle #2 tolerance”, deg	50…150 0...89	Value and permissible deviation of angle #2.
	“Angle #3”, deg “Angle #3 tolerance”, deg	-90…-10 0...45	Value and permissible deviation of angle #3.
	“Fillet”:		Group of parameters for the "fillet" template. Designation of lines and angles: 
	“Min len (left)”, mm “Max len (left)”, mm	0.1...1000 0.1...1000	Minimum and maximum length of the segment to the left.
	“Min len (right)”, mm “Max len (right)”, mm	0.1...1000 0.1...1000	Minimum and maximum length of the segment to the right.
	“Max distance”, mm	0.1...100	Maximum allowable distance between the end of the left segment and the beginning of the right one. Minimum allowable distance is 0.
	“Angle”, deg “Angle tolerance”, deg	50…150 0...89	Value and permissible deviation of angle #1.
	“Lap left”:		Group of parameters for the "lap left" template. Designation of lines and angles:

			
	"Min len (segment 1)", mm "Max len (segment 1)", mm	0.1...1000 0.1...1000	Minimum and maximum length of segment #1.
	"Min len (segment 2)", mm "Max len (segment 2)", mm	0.1...100 0.1...100	Minimum and maximum length of segment #2.
	"Min len (segment 3)", mm "Max len (segment 3)", mm	0.1...1000 0.1...1000	Minimum and maximum length of segment #3.
	"Angle #1", deg "Angle #1 tolerance", deg	-150…-30 0...89	Value and permissible deviation of angle #1.
	"Angle #2", deg "Angle #2 tolerance", deg	30…150 0...89	Value and permissible deviation of angle #2.
	"Lap right":		Group of parameters for the "lap right" template. Designation of lines and angles: 
	"Min len (segment 1)", mm "Max len (segment 1)", mm	0.1...1000 0.1...1000	Minimum and maximum length of segment #1.
	"Min len (segment 2)", mm "Max len (segment 2)", mm	0.1...100 0.1...100	Minimum and maximum length of segment #2.
	"Min len (segment 3)", mm "Max len (segment 3)", mm	0.1...1000 0.1...1000	Minimum and maximum length of segment #3.
	"Angle #1", deg "Angle #1 tolerance", deg	30…150 0...89	Value and permissible deviation of angle #1.
	"Angle #2", deg "Angle #2 tolerance", deg	-150…30 0...89	Value and permissible deviation of angle #2.
Inputs:	"idx"	SDT_INT	Template index. The order is the same as in this document: 0 - "trapeze groove", etc.
Outputs:	"det"	SDT_BOOL	Flag of successful template detection (the weld seam is recognized, the correct data is output).
	"pt 1"	SDT_POINT	The coordinates of the first point.
	"fillet weld" - the block parameters are similar to the parameters of the corresponding template of the "templates set" block.		
Outputs:	"det"	SDT_BOOL	Flag of successful template detection (the joint is recognized, the correct data is output).
	"left segment"	SDT_SEGMENT	The left segment of the corner.
	"right segment"	SDT_SEGMENT	The right segment of the corner.

	“corner weld”: 		
Parameters:	“Min len (segment 1)”, mm “Max len (segment 1)”, mm	0.1...1000 0.1...1000	Minimum and maximum length of segment #1.
	“Min len (segment 2)”, mm “Max len (segment 2)”, mm	0.1...100 0.1...100	Minimum and maximum length of segment #2.
	“Min len (segment 3)”, mm “Max len (segment 3)”, mm	0.1...100 0.1...100	Minimum and maximum length of segment #3.
	“Min len (segment 4)”, mm “Max len (segment 4)”, mm	0.1...1000 0.1...1000	Minimum and maximum length of segment #4.
	“Max distance”, mm	0.1...100	Maximum allowable distance between the end of the left segment and the beginning of the right one. Minimum allowable distance is 0.
	“Angle #1”, deg “Angle #1 tolerance”, deg	-150…-50 0...89	Value and permissible deviation of angle #1.
	“Angle #2”, deg “Angle #2 tolerance”, deg	50…150 0...89	Value and permissible deviation of angle #2.
	“Angle #3”, deg “Angle #3 tolerance”, deg	-150…-50 0...89	Value and permissible deviation of angle #3.
Outputs:	“det”	SDT_BOOL	Flag of successful template detection (the joint is recognized, the correct data is output).
	“segment #1”	SDT_SEGMENT	Segment #1.
	“segment #2”	SDT_SEGMENT	Segment #2.
	“segment #3”	SDT_SEGMENT	Segment #3.
	“segment #4”	SDT_SEGMENT	Segment #4.
	“lap weld” - block parameters are the same as parameters of the “lap left” template of the “templates set” block.		
Outputs:	“det”	SDT_BOOL	Flag of successful template detection (the joint is recognized, the correct data is output).
	“segment #1”	SDT_SEGMENT	Segment #1.
	“segment #2”	SDT_SEGMENT	Segment #2.
	“segment #3”	SDT_SEGMENT	Segment #3.
	“v-groove weld” - block parameters are the same as parameters of the corresponding template of the “templates set” block.		
Outputs:	“det”	SDT_BOOL	Flag of successful template detection (the joint is recognized, the correct data is output).

	“segment #1”	SDT_SEGMENT	Segment #1.
	“segment #2”	SDT_SEGMENT	Segment #2.
	“segment #3”	SDT_SEGMENT	Segment #3.
	“segment #4”	SDT_SEGMENT	Segment #4.
	“square groove weld”: 		
Parameters:	“Min len (left segment)”, mm “Max len (left segment)”, mm	0.1...1000 0.1...1000	Minimum and maximum length of the segment to the left.
	“Min len (right segment)”, mm “Max len (right segment)”, mm	0.1...100 0.1...100	Minimum and maximum length of the segment to the right.
	“Min distance”, mm “Max distance”, mm	0...100 0.1...100	Minimum and maximum allowable distance between the end of the left segment and the beginning of the right one.
	“Angle”, deg “Angle tolerance”, deg	-150…-50 0...89	Value and permissible deviation of the angle between the segments.
Outputs:	“det”	SDT_BOOL	Flag of successful template detection (the joint is recognized, the correct data is output).
	“left segment”	SDT_SEGMENT	Segment to the left.
	“right segment”	SDT_SEGMENT	Segment to the right.
	“3-pt tracking (by points)” - tracking of the weld groove at three points formed by the profile:  The output values of the smart block are the points and angles (i.e. poses) to which the actuator (welding robot) must move in order to travel along the welding path. Points are issued sequentially, ahead of time. The tracking process is displayed in 3D in a special window, and this window can also be used for visual analysis of the weld groove. A detailed description of this smart block is given in the User's Manual for the "Laser Seam Tracking System for Welding Automation. RF627Weld-Smart.".		
Inputs:	“cst”	SDT_CST_3D	Data from the scanner calibration block with an actuator.
	“point #1”	SDT_POINT	"Left" point of the weld groove.
	“point #2”	SDT_POINT	"Central" point of the weld groove.
	“point #3”	SDT_POINT	"Right" point of the weld groove.
	“enabled”	SDT_BOOL	"Smart block enabled" flag.

	“accuracy”	SDT_FLOAT	Actuator movement accuracy (mm).
	“step”	SDT_FLOAT	Step of picking points along the weld groove and step of issuing points (mm).
	“torch offset”	SDT_FLOAT	Offset of the TCP relative to the weld groove in the direction perpendicular to the groove (offset from the groove) in mm.
	“torch rotation”	SDT_EULER_3D	Corrections to the angular position of the tool relative to the flange (in order to take into account the bending of the tool, for example, a burner) in rad.
Outputs:	“pose”	SDT_POSE_3D	Output poses for an actuator (for example, a welding robot).
	“detected”	SDT_BOOL	"Weld groove detected" flag.
	“tracking”	SDT_BOOL	"Weld groove tracking" flag. It is set to TRUE while the block is enabled and the block can issue the points to the robot, i.e. the end of the weld groove has not been reached.
	“3-pt tracking (by velocity)” - tracking of the weld groove at three points formed by the profile (similar to the “3-pt tracking (by points)” smart block). The output values of the smart block are the linear and angular velocities (in the form of a pose) with which the actuator (welding robot) must travel along the welding path. The tracking process is displayed in 3D in a special window, and this window can also be used for visual analysis of the weld groove. A detailed description of this smart block is given in the User's Manual for the "Laser Seam Tracking System for Welding Automation. RF627Weld-Smart.".		
Inputs:	“cst”	SDT_CST_3D	Data from the scanner calibration block with an actuator.
	“point #1”	SDT_POINT	"Left" point of the weld groove.
	“point #2”	SDT_POINT	"Central" point of the weld groove.
	“point #3”	SDT_POINT	"Right" point of the weld groove.
	“enabled”	SDT_BOOL	"Smart block enabled" flag.
	“accuracy”	SDT_FLOAT	Actuator movement accuracy (mm).
	“step”	SDT_FLOAT	Step of picking points along the weld groove (mm).
	“torch offset”	SDT_FLOAT	Offset of the TCP relative to the weld groove in the direction perpendicular to the groove (offset from the groove) in mm.
	“torch rotation”	SDT_EULER_3D	Corrections to the angular position of the tool relative to the flange (in order to take into account the bending of the tool, for example, a burner) in rad.
Outputs:	“pose”	SDT_POSE_3D	Output velocities for an actuator (for example, a welding robot).
	“detected”	SDT_BOOL	"Weld groove detected" flag.
	“tracking”	SDT_BOOL	"Weld groove tracking" flag. It is set to TRUE while the block is enabled and the block can issue the points to the robot, i.e. the end of the weld groove has not been reached.
	“weld seam detector”– Weld center detection. Currently, the unit detects weld centers only on the pipe surface, as it uses a basic circular approximation. A detailed description of the unit is provided in the RF627Weld-Smart welding kit manual.		
Inputs:	"left bordedr"	SDT_FLOAT	The left border to the left of which the seam is guaranteed to be absent

	"right border"	SDT_FLOAT	The right border to the right of which the seam is guaranteed to be absent
	"sliding window width"	SDT_FLOAT	The width of the sliding window when constructing a complex characteristic for seam detection
	"sliding window step"	SDT_FLOAT	The sliding window offset step when constructing a complex characteristic for seam detection
Outputs:	"pose"	SDT_POINT	Coordinates of the detected seam
	"det"	SDT_BOOL	Weld detection flag

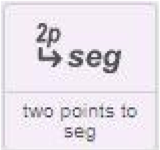
25.3.2.4. "Converters" section

	"scalar to bool" - converting a scalar value to a boolean value. The conversion is performed according to the following rule: the scalar value is greater than "0" - "TRUE", otherwise - "FALSE".			
	Inputs:	"in"	SDT_SCALAR	Scalar value.
	Outputs:	"out"	SDT_BOOL	Boolean value.
	"scalar to int16" - converting a scalar value to an integer value (size - 2 bytes). The conversion is performed rounding down to the smallest (in absolute value) integer.			
	Inputs:	"in"	SDT_SCALAR	Scalar value.
	Outputs:	"out"	SDT_INT16	Integer value.
	"scalar to int32" - converting a scalar value to an integer value (size - 4 bytes). The conversion is performed rounding down to the smallest (in absolute value) integer.			
	Inputs:	"in"	SDT_SCALAR	Scalar value.
	Outputs:	"out"	SDT_INT	Integer value.
	"scalar to int32" - converting a scalar value to a floating-point value (single precision).			
	Inputs:	"in"	SDT_SCALAR	Scalar value.
	Outputs:	"out"	SDT_FLOAT	Floating-point value.
	"scalar to double" - converting a scalar value to a floating-point value (double precision).			
	Inputs:	"in"	SDT_SCALAR	Scalar value.
	Outputs:	"out"	SDT_DOUBLE	Floating-point value.
	"scalar to int64" - converting a scalar value to an integer value (size - 8 bytes). The conversion is performed with rounding to the smallest (in absolute value) integer value.			
	Inputs:	"in"	SDT_SCALAR	Scalar value.
	Outputs:	"out"	SDT_INT64	Integer value.
	"bool to scalar" - converting a boolean value to a scalar value. The conversion is performed according to the following rule: if the boolean value is "TRUE", then the scalar value is "1", otherwise - "0".			
	Inputs:	"in"	SDT_BOOL	Boolean value.
	Outputs:	"out"	SDT_SCALAR	Scalar value.

	“int16 to scalar” - converting an integer value (size - 2 bytes) to a scalar value.			
	Inputs:	“in”	SDT_INT16	Integer value.
	Outputs:	“out”	SDT_SCALAR	Scalar value.
	“int32 to scalar” - converting an integer value (size - 4 bytes) to a scalar value.			
	Inputs:	“in”	SDT_INT32	Integer value.
	Outputs:	“out”	SDT_SCALAR	Scalar value.
	“float to scalar” - converting a single-precision floating-point value to a scalar value.			
	Inputs:	“in”	SDT_FLOAT	Floating-point value.
	Outputs:	“out”	SDT_SCALAR	Scalar value.
	“double to scalar” - converting a double-precision floating-point value to a scalar value.			
	Inputs:	“in”	SDT_DOUBLE	Floating-point value.
	Outputs:	“out”	SDT_SCALAR	Scalar value.
	“int64 to scalar” - converting an integer value (size - 8 bytes) to a scalar value.			
	Inputs:	“in”	SDT_INT64	Integer value.
	Outputs:	“out”	SDT_SCALAR	Scalar value.
	“deg to rad” - converting a scalar value in degrees to a scalar value in radians.			
	Inputs:	“in”	SDT_SCALAR	Scalar in degrees.
	Outputs:	“result”	SDT_SCALAR	Scalar in radians.
	“rad to deg” - converting a scalar value in radians to a scalar value in degrees.			
	Inputs:	“in”	SDT_SCALAR	Scalar in radians.
	Outputs:	“result”	SDT_SCALAR	Scalar in degrees.
	“mm to inch” - converting a scalar value in millimeters to a scalar value in inches.			
	Inputs:	“in”	SDT_SCALAR	Scalar in mm.
	Outputs:	“result”	SDT_SCALAR	Scalar in inches.
..				
	“inch to mm” - converting a scalar value in inches to a scalar value in millimeters.			
	Inputs:	“in”	SDT_SCALAR	Scalar in inches.
	Outputs:	“result”	SDT_SCALAR	Scalar in mm.

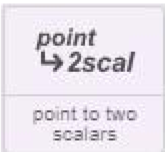
	"seg to two points" - converting a line segment to two points corresponding to the ends of the segment.			
	Inputs:	"in"	SDT_SEGMENT	Line segment.
	Outputs:	"left"	SDT_POINT	The point corresponding to the left end of the segment (the smaller X-coordinate).
		"right"	SDT_POINT	The point corresponding to the right end of the segment (the greater X-coordinate).

	"seg to line" - converting a line segment to a line with a corresponding slope and offset.			
	Inputs:	"in"	SDT_SEGMENT	Line segment.
	Outputs:	"out"	SDT_LINE	Line.

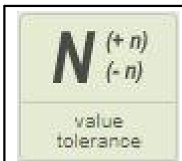
	"two points to seg" - converting two points corresponding to the ends of a segment to a line segment.			
	Inputs:	"left"	SDT_POINT	The point corresponding to the left end of the segment (the smaller X-coordinate).
		"right"	SDT_POINT	The point corresponding to the right end of the segment (the greater X-coordinate).
	Outputs:	"out"	SDT_SEGMENT	Line segment.

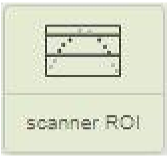
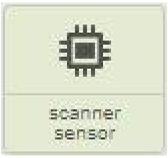
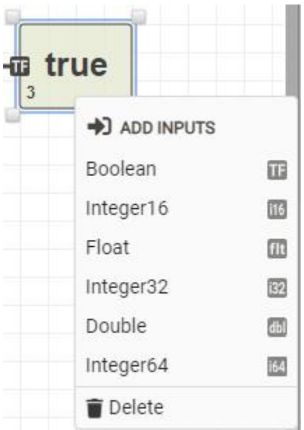
	"two points to line" - converting two points to a line.			
	Inputs:	"left"	SDT_POINT	The point corresponding to the left end of the segment (the smaller X-coordinate).
		"right"	SDT_POINT	The point corresponding to the right end of the segment (the greater X-coordinate).
	Outputs:	"out"	SDT_LINE	Line.

	"line to seg" - converting a line to a segment.			
	Parameters:	"Left point x", mm	-1000.0…1000.0	The x-coordinate of the left point.
		"Right point x", mm	-1000.0…1000.0	The x-coordinate of the right point.
	Inputs:	"in"	SDT_LINE	The input line to be converted to a segment.
	Outputs:	"out"	SDT_SEGMENT	Segment.

	"point to two scalars" - decomposition of a point into two scalars.			
	Inputs:	"in"	SDT_POINT	The point whose coordinates will be at the outputs of the smart block.
	Outputs:	"x"	SDT_SCALAR	The x-coordinate of the input point.
		"z"	SDT_SCALAR	The z-coordinate of the input point.

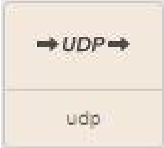
25.3.2.5. "Control" section

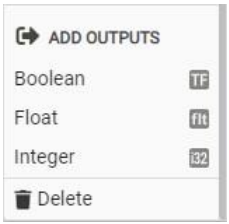
	"value tolerance" - checking the input scalar value for falling into the range specified by the parameters.			
	Inputs:	"in"	SDT_SCALAR	The checked value.
	Outputs:	"result"	SDT_SCALAR	The check result.

	"scanner laser" - control of the laser installed in the scanner.			
	Inputs:	"enable"	SDT_BOOL	Turning the laser on (TRUE) or turning it off (FALSE).
		"result"	SDT_INT	Laser output power, %.
	"scanner ROI" - control of the region of interest that the scanner processes.			
	Inputs:	"enable"	SDT_BOOL	Enable / disable the region of interest.
		"pos"	SDT_FLOAT	ROI position in mm.
		"size"	SDT_FLOAT	ROI size in mm.
	"scanner sensor" - control of the parameters of the CMOS sensor installed in the scanner.			
	Inputs:	"pps"	SDT_INT	The required number of profiles per second (may be limited by the scanner operation mode).
		"expose1"	SDT_INT	Frame exposure time in µs.
	"scanner encoder" - control of the counters (encoder) of the scanner and obtaining its current values.			
	Inputs:	"rst"	SDT_BOOL	Reset counters to "0" (if the input value is TRUE).
	Outputs:	"profile"	SDT_INT32	The current profile number.
		"pulse"	SDT_INT32	The number of pulses at the physical inputs of the scanner (encoder counter).
		"dir"	SDT_BOOL	Direction of movement.
	"picoc script" - editing and execution of the script written in the PicoC language. A detailed description of the block is provided in Annex 7.			
	Parameters:	"Execute"	ON/OFF	Execute the script (ON) or stop the script execution (OFF).
	Inputs:	Inputs are created by the user using the context menu.		
	Outputs:	Outputs are created by the user using the context menu.		
	"values monitor" - viewing current values. This block allows the user to view the current values of the signals on the graph in real time.			
	Inputs:	Inputs are created and deleted by the user using the context menu. 		

25.3.2.6. "Base IO" section

The following smart blocks are available in the basic configuration of **Smart** scanners.

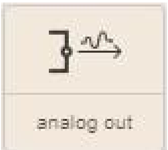
	"UDP" - transmitting and receiving data over UDP (User Datagram Protocol).		
Parameters:	"Output datagram", bytes	8...16384	The size of the sent datagram. It will contain data from the inputs of the block.
	"Destination IP"	XXX.XXX.XXX.XXX	The IP address of the host to which the datagram is sent.
	"Destination port"	1...65535	The port number of the host to which the datagram is sent.
	"Input datagram", bytes	8...16384	The size of the received datagram. It must contain data for the block outputs.
	"Receive IP"	XXX.XXX.XXX.XXX	The IP address of the scanner. It is set in the general settings of the scanner.
	"Receive port"	1...65535	The port number of the scanner to listen for incoming datagrams.
	"Port map"	Send	Distribution of block inputs over the sent datagram. The input values will be arranged in accordance with this parameter. 
	"Port map"	Receive	Distribution of block outputs in the received datagram. The output values must be arranged in accordance with this parameter. 
Inputs:	Inputs are created by the user using the context menu.		

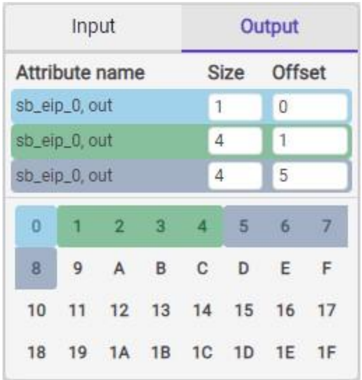
			
Outputs:	Outputs are created by the user using the context menu. 		
	"tcp server" - transmitting and receiving data over TCP (Transmission Control Protocol). The scanner is a server and waits for clients to connect (no more than four clients at a time). In the current version, the FIN packet must be sent when disconnecting a client. The size of the sent packet is always equal to the value of the "Send buffer size" parameter.		
Parameters:	"Listen port"	1...65535	Scanner port number to which clients are expected to connect.
	"Send buffer size", bytes	1...16384	Send buffer size (the maximum amount of data that the sent packet can contain).
	"Send buffer size", bytes	1...16384	Receive buffer size (the maximum amount of data that can be read in the received packet). Larger packets will be ignored.
	"Port map"	Send	Distribution of block inputs over the sent buffer. The input values will be arranged according to this parameter. 
	"Port map"	Receive	Distribution of block outputs in the received packet. The output values must be arranged according to this parameter.

			
Inputs:	Inputs are created by the user using the context menu.		
Outputs:	Outputs are created by the user using the context menu.		

25.3.2.7. "Industrial IO" section

The following smart blocks are available in the **Industrial** configuration (in addition to the **Base** configuration) of **Smart** scanners.

	"phys out" - output of results to physical outputs of the device. Since physical outputs can only be in two mutually exclusive states ("TRUE", "FALSE"), the following conversion rule applies: if the value of the input scalar is greater than "0", then the output is "TRUE", otherwise - "FALSE".		
Inputs:	"phys_out_1"	SDT_SCALAR	The value transmitted to physical output #1.
	"phys_out_2"	SDT_SCALAR	The value transmitted to physical output #2.
	"analog out" - output of analog signals. This block must be used with analog output devices manufactured by Riftek. Output signal amplitude scaling is available for each channel (minimum and maximum possible values are set).		
Parameters:	"Baud rate", bits/s	9600…92160	The baud rate for the analog signal output device.
	"Channel #N min"	-3.4x10^38… 3.4x10^38	Minimum output voltage for channel N.
	"Channel #N max"	-3.4x10^38… 3.4x10^38	Maximum output voltage for channel N.
Inputs:	"analog_out_N"	SDT_SCALAR	Output signal amplitude for channel N.
	"Etnernet/IP" - transmitting and receiving data over Ethernet/IP. Only one instance of this block can be placed on the graph.		
Parameters:	"Input point"	1...256	Input assembly number (according to EIP specification).
	"Output point"	1...256	Output assembly number (according to EIP specification).
	"Assembly size"	1...512	Assembly size in bytes.
	"Assembly map"	Input	Distribution of block inputs over the input assembly. The values of the inputs will be arranged in accordance with this parameter.

			
	"Assembly map"	Output	Distribution of block outputs over the output assembly. The values of the outputs must be arranged in accordance with this parameter. 
Inputs:	Inputs are created by the user using the context menu.		
Outputs:	"state"	SDT_BOOL	Connection flag.
	Outputs are created by the user using the context menu.		
Modbus TCP modbus_tcp	"ModbusTCP" - transmitting and receiving data over ModbusTCP. The addresses of the objects ("Coils", "Discrete inputs", "Input registers", "Holding registers") are independent and can overlap. Inputs and outputs of the block are generated dynamically using the context menu. Boolean inputs are always located in the "Discrete inputs" object, and Boolean outputs - in the "Coils" object. Inputs of other available types will be located in the "Input registers" object, outputs - in the "Holding registers" object. For inputs and outputs other than Boolean (the "Input registers" and "Holding registers" objects), data size conversion is possible. For example, the type of the block input is Float and occupies 4 bytes, but the responder can only work with the Float16 type of 2 bytes. The user can specify that it is necessary to write data to the output register (2 bytes in size according to the Modbus specification) with a reduction to a size of 2 bytes:  Only one instance of this block can be placed on the graph.		
Parameters:			
Coils:	"Address"	0...65535	The starting address of the object.
	"Count"	0...1968	The number of elements.
	"Assembly map"	Output	Distribution of block outputs over the output assembly. The values of the outputs must be arranged in accordance with this parameter.

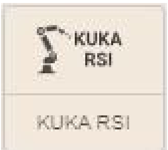
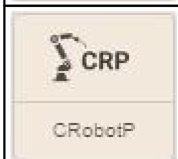
			
Discrete inputs:	“Address”	0...65535	The starting address of the object.
	“Count”	0...2000	The number of elements.
	“Assembly map”	Input	Distribution of block inputs over the input assembly. The values of the inputs will be arranged in accordance with this parameter. 
Input registers:	“Address”	0...65535	The starting address of the object.
	“Count”	0...125	The number of elements.
	“Assembly map”	Input	Distribution of block inputs over the input assembly. The values of the inputs will be arranged in accordance with this parameter. 
Holding registers:	“Address”	0...65535	The starting address of the object.
	“Count”	0...123	The number of elements.
	“Assembly map”	Output	Distribution of block outputs over the output assembly. The values of the outputs must be arranged in accordance with this parameter. 
Inputs:	Inputs are created by the user using the context menu.		
Outputs:	Outputs are created by the user using the context menu.		

25.3.2.8. "Robot IO" section

The following smart blocks are available in **Industrial** configuration (in addition to the **Base** configuration) of **Smart** scanners. A detailed description of the blocks and protocols is given in the User's Manual for "Laser seam tracking system for welding automation":

https://riftek.com/upload/medialibrary/870/kz28mn7xa378395yt720ueeu0c9dlaks/Laser_Seam_Tracking_System_for_Welding_Automation_eng.pdf

	“robot protocol HND1” - data exchange with robots using the HND1 protocol.		
Parameters:	“Destination IP”	XXX.XXX.XXX.XXX	The IP address of the robot (or other device with which data should be exchanged).
	“Destination port”	1...65535	The network port number of the robot (or other device with which data should be exchanged).

	"Listen port"	1...65535	The network port number of the scanner to listen for incoming packets.
	"Swap X<->Y"	true/false	Swapping the X and Y coordinates of points.
	"Flip X-axis"	on/off	Flipping coordinates along the X axis (relative to 0). It is performed after applying the "Swap X<->Y" parameter.
	"Flip Y-axis"	on/off	Flipping coordinates along the Y axis (relative to 0). It is performed after applying the "Swap X<->Y" parameter.
	"Offset along X-axis, mm"	-1000...1000	Offset of coordinates along the X axis. It is performed after applying the "Flip Y-axis" parameter.
	"Offset along Y-axis, mm"	-1000...1000	Offset of coordinates along the Y axis. It is performed after applying the "Flip Y-axis" parameter.
Inputs:	"det"	SDT_BOOL	Boolean flag for template detection (correctness of all output points).
	"point #1"	SDT_POINT	Point #1, the coordinates of which are transmitted in the packet with the measurement results.
	"point #2"	SDT_POINT	Point #2, the coordinates of which are transmitted in the packet with the measurement results.
	"point #3"	SDT_POINT	Point #3, the coordinates of which are transmitted in the packet with the measurement results.
Outputs:	"idx"	SDT_INT	The index of the welding template to be used.
	"R691 protocol" - data exchange and control of Fanuc robots via the R691 (Universal sensor interface) protocol.		
	"KUKA RSI" - data exchange and control of KUKA robots via a customizable protocol with the RSI (Robot Sensor Interface) module.		
	"JAKA" - data exchange and control of JAKA robots.		
	"P3 protocol" - data exchange and control of robots via the P3 protocol (based on Ethernet/IP).		
	"Rozum Robotics" - data exchange and control of Rozum Robotics robots.		
	"CrobotP" - data exchange and control of CRP robots.		

UR Universal Robots RTDE	"Universal Robots RTDE" - data exchange and control of UR robots via the RTDE (Real-Time Data Exchange) protocol.

26. Maintenance

Laser scanners are virtually maintenance free. As these are optical systems, they are sensitive to dust and sputter on the front windows. Cleaning is best done with a soft cloth. Do not use scratching cleaners or other aggressive media.

It is necessary to remove fingerprints from the windows, because fingerprints degrade the quality of profiles.

In order to remove fingerprints or grease, clean the windows with 20 % alcohol and soft paper.

27. Troubleshooting

Problem	Cause	Solution
Laser is off	No power supply (or less than 9 V).	Check the power supply.
	Power cable or Ethernet are not connected.	Check the cables connection.
	Scanner electronics failure.	Contact the technical support.
No scanners on the network	No power supply (or less than 9 V).	Check the power supply.
	Ethernet cable or/and power cable are not connected.	Check the cables.
	Incorrect settings of the network card of the PC.	Check the network card configuration (see par. 12.1).
	Scanner freezes.	Reboot the scanner.
	Scanner electronics failure.	Contact the technical support.
No profile	Low exposure time.	Check the exposure time.
	The object is beyond the working range of the scanner.	Install the object within the working range of the scanner.
	ROI mode is enabled and the object is beyond the set ROI area.	Check the ROI settings.
Incorrect profile	Scanner windows are not clean.	Clean the windows as described in par. 26.
	Incorrect scanner settings.	Check settings.
	Measurements are taken in locations close to powerful light sources.	Do not use the scanner in locations close to powerful light sources.
Incorrect profile reflection and distortions in measurements	May occur when the current firmware version is under 20190717 and you update it to the firmware version from 20190717 to 20191112 (provided that the "Image Flip" option was used during calibration).	Update the firmware to a version later than 20191113. To restore the profile orientation, contact the technical support.

28. Annex 1. Recovery mode

The **Recovery** mode is intended to restore the scanner operability in case of hardware failures or after incorrect user actions.

To activate this mode, it is necessary to turn on the scanner with the **Reset** button pressed and continue to hold this button pressed for at least 10 seconds.

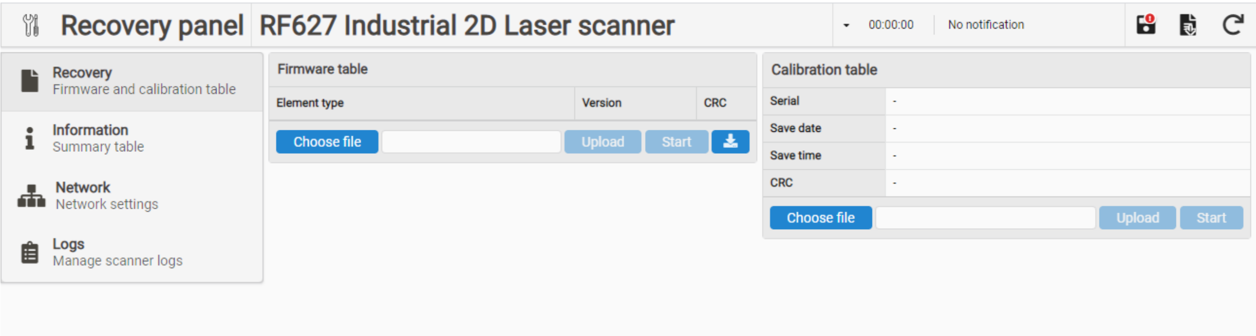
In this mode, the **PWR** indicator displays the SOS signal (three short - three long - three short), which means that the scanner is in **Recovery** mode now.

After turning off the scanner, it will operate in basic mode when turned on again.

In **Recovery** mode, when you enter the IP address of the scanner in the address bar of the browser, a simplified web page will be loaded. In this page, you can perform the following actions:

– view general scanner settings;
– update the firmware of the scanner;
– view and, if necessary, change the network settings;
– view the log file.

The WEB page in **Recovery** mode is shown below:



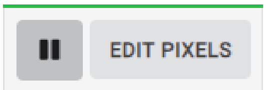
The controls in the upper part are the same as in the main web page. The sections of the web page in Recovery mode correspond to the modes of the main web interface.

Recovery mode	Main interface	Section
Recovery	"Update" section, "System" tab	24.2. "Update" section.
Information	"Information" section, "System" tab	24.1. "Information" section.
Network	"Network" tab	18. "Network" tab. Network parameters.
Logs	"Logs" section, "System" tab	24.4. "Logs" section.

29. Annex 2. Editing defective pixels

During the operation of the scanner, defective pixels may appear in the CMOS sensor, which significantly distort the profile extracted from the image. The procedure for marking defective pixels is given below. After marking, the signal value of the defective pixel is automatically calculated as the result of interpolating the signal of adjacent pixels.

The button for enabling the mode of editing defective pixels of the CMOS sensor (**EDIT PIXELS** button) is located in the web interface of the scanner in the area of additional display parameters next to the button for stopping / starting the video stream.



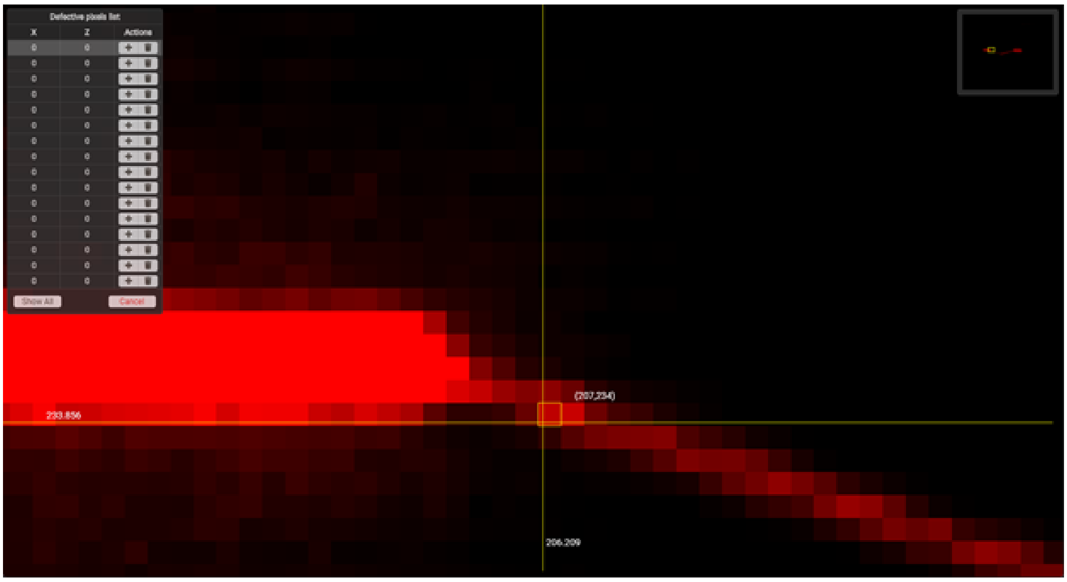
When this mode is enabled, a window with a list of defective pixels is displayed.

Defective pixels list

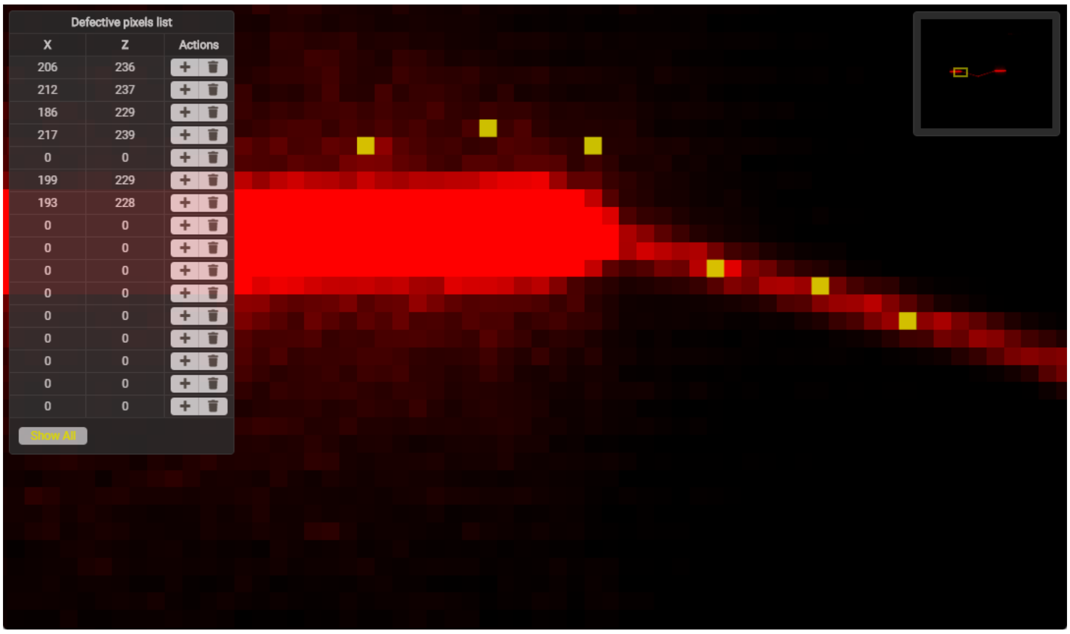
X	Z	Actions
0	0	
0	0	
0	0	
0	0	
0	0	
0	0	
0	0	
0	0	
0	0	
0	0	
0	0	
0	0	
0	0	
0	0	
0	0	
0	0	

Show All

To add a pixel to the list, click  in an empty row of the table (X and Z coordinates are equal to zero), and then left-click on the defective pixel in the image. The cursor automatically selects the current pixel with an indication of its coordinates. To exit this mode, click 
.



To clear the table row (cancel the defective pixel interpolation), click in the required row. After clicking Show All, all the defective pixels added to the table will be selected.



30. Annex 3. Web API

Using the Web API, the user can obtain information about the device, read or write the parameter value. In addition, the device can execute some commands through the Web API. For a complete list of supported commands, see the command description. The Web API examples use the device's factory IP address and present commands as they would be typed into the address bar of the browser. If the device's IP address has been changed, the device's current IP address should be used.

30.1. General device information

/hello - getting general information about the device in JSON format.

- GET:
  - 192.168.1.30/hello

/api/v1/config/commands - getting a list of commands supported by the device. The formalized description will contain the command name, web API accessibility, command ID, access mode.

- GET:
  - 192.168.1.30/api/v1/config/commands

/api/v1/config/returnCodes - getting a textual description of the codes of the operation results and errors returned by the device.

- GET:
  - 192.168.1.30/api/v1/config/returnCodes

30.2. Reading and writing parameters

/api/v1/config/params - getting general information about all device parameters in JSON format. The formalized description of the parameter will contain its name, type, access mode, index in the parameter array, offset for binary data, parameter data size, current value, default value, minimum and maximum values, parameter value step, maximum number of elements (for arrays).

- GET:
  - 192.168.1.30/api/v1/config/params

/api/v1/config/params/values - reading and writing device parameters. For reading, you can request specific parameters by name or index. To write a parameter, it is necessary to form a “PUT” request with the “parameter_name:value” parameters.

- GET:
  - 192.168.1.30/api/v1/config/params/values
  - 192.168.1.30/api/v1/config/params/values?
    name=fact_general_hardwareVer&index=120
- PUT:
  - 192.168.1.30/api/v1/config/params/values?
    user_sensor_framerate=100&user_sensor_exposure1=100000

30.3. Saving and restoring settings. Rebooting the device

/api/v1/config/params/save - saving the current values of the device parameters in the non-volatile memory of the device in the user area. The saved values will be used the next time the device is turned on.

- GET:
  - 192.168.1.30/api/v1/config/params/save

/api/v1/config/params/restore/save - saving the current values of the device parameters in the recovery area. These parameters will be applied if the parameters from the user area are damaged.

- GET:
  - 192.168.1.30/api/v1/config/params/restore/save

/api/v1/config/params/restore/load - loading device parameters from the recovery area. The loaded values will be written to the user area, the device will automatically reboot.

- GET:
  - 192.168.1.30/api/v1/config/params/restore/load

/api/v1/reboot - reboot the device. The parameters will be loaded from the user area (if they are not damaged).

- GET:
  - 192.168.1.30/api/v1/reboot

30.4. Getting information from the device log file

/api/v1/log - receiving a log file of the device operation with a full description of the records.

- GET:
  - 192.168.1.30/api/v1/log

/api/v1/log/content - receiving a log file of the device operation in a shortened, easy-to-read form.

- GET:
  - 192.168.1.30/api/v1/log/content

30.5. Authorization

/api/v1/authorization - authorization as a manufacturer. This allows you to edit the factory parameters of the device. Using the “GET” request, you need to get a token for which a key must be generated. The key must be sent to the device in the “PUT” request.

- GET:
  - 192.168.1.30/api/v1/authorization
- PUT:
  - 192.168.1.30/api/v1/authorization?key=230d84e16c0dae529098f1f1bb4debb3a6db3c870c4699245e651c06b714deb35a4d0a43a99f5ea0cc771a0e189c190a

30.6. Profile request

/api/v1/profile/capture - request for making measurements (obtaining a profile). It is available only in “Software, external” and “Software, internal” modes.

- GET:
  - 192.168.1.30/api/v1/profile/capture - request for 1 measurement;
  - 192.168.1.30/api/v1/profile/capture?count=100 - request for 100 measurements.

30.7. Smart

/api/v1/smart/description - getting a description of block groups, data types of the “Smart” module and an array of blocks implemented in this firmware.

- GET:
  - 192.168.1.30/api/v1/smart/description

/api/v1/smart/graph/results - getting the results of the graph blocks operation and the profile on which the calculation was performed.

- GET:
  - 192.168.1.30/api/v1/smart/graph/results

/api/v1/smart/block/read - getting a list of graph blocks with their parameters.
- GET:
  - 192.168.1.30/api/v1/smart/block/read

31. Annex 4. "Template detector" smart block and Template Editor

31.1. General information

The "templates detector" smart block is designed to detect templates in the profile (templates are stored in a file that must be loaded through the smart block parameters). The template can be selected in the following ways:

- by the operator using the web interface,
- by an external system through a special input of the smart block,
- automatically by the criterion of the greatest similarity (not available in this revision).

A set of templates is formed by the user using a special editor, described below, and saved to a file with the ".template" extension. The scanner can contain several files, each file can contain several templates. The template can be based on the current profile approximation or a user-drawn sequence of segments.

31.2. Template structure and search principle

The template is:

- A set of elements (**Element**) identical to those obtained after profile approximation: segments and arcs. Each element includes a description of the profile part (**Part**), used only for the graphical display of the element in the editor.
- A set of self constraints and relative constraints that make it possible to establish a relationship between elements and select only those elements that satisfy the constraints.
- A description of output values of the template set (**Outputs**). The output values can be (at the user's choice) segments, or their points (start, end, middle), or points of intersection of the segments.
- A description of options (**Variant**) that determine the acceptable absence of elements (for example, in a U-shaped template, the middle segment may disappear).
- An icon automatically generated or uploaded by the user, which is displayed in the settings of the "templates detector" smart block.

In the current firmware version, the number of variants, elements and constraints cannot exceed the following values:

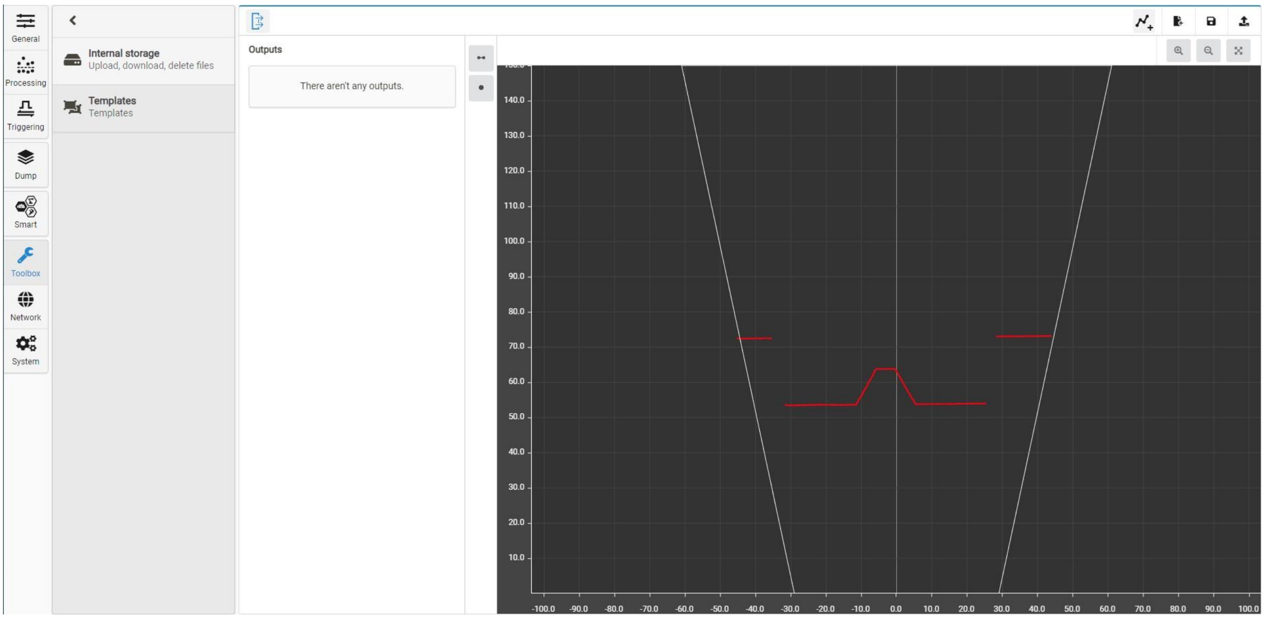
Maximum number of template set outputs (number of smart block outputs)	8
Maximum number of elements in a template	16
Maximum number of constraints for each template	64
Maximum number of variants for each template	8
Maximum icon size, bytes	65536
Maximum icon size, pixels	64x64

The search for a template in a profile is based on a sequential check of self constraints and relative constraints for the elements of the profile approximation. The search starts for the first variant, and if at least one constraint is not met, the check stops and the next variant will be checked. If the last variant of the template is reached and the template is not found, it is considered that the template was not found in the profile. The “det” output of the block will be set to “FALSE”, the outputs of the block will have invalid values of the segments (or arcs - in the future).

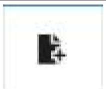
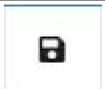
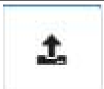
If each element of the template meets the constraints, the template is considered found. The “det” output of the block will be set to “TRUE” and the block outputs will receive data according to their type and settings.

31.3. Template editor

To open the template editor, select **Toolbox > Templates**:



If templates have not been previously created or loaded from a file, the editor will not display any information. The user has access to:

Creating a new set of templates	
Saving the created or edited template set to a file in the non-volatile memory of the scanner	
Loading a set of templates from the non-volatile memory of the scanner	

31.3.1. Working with the template editor

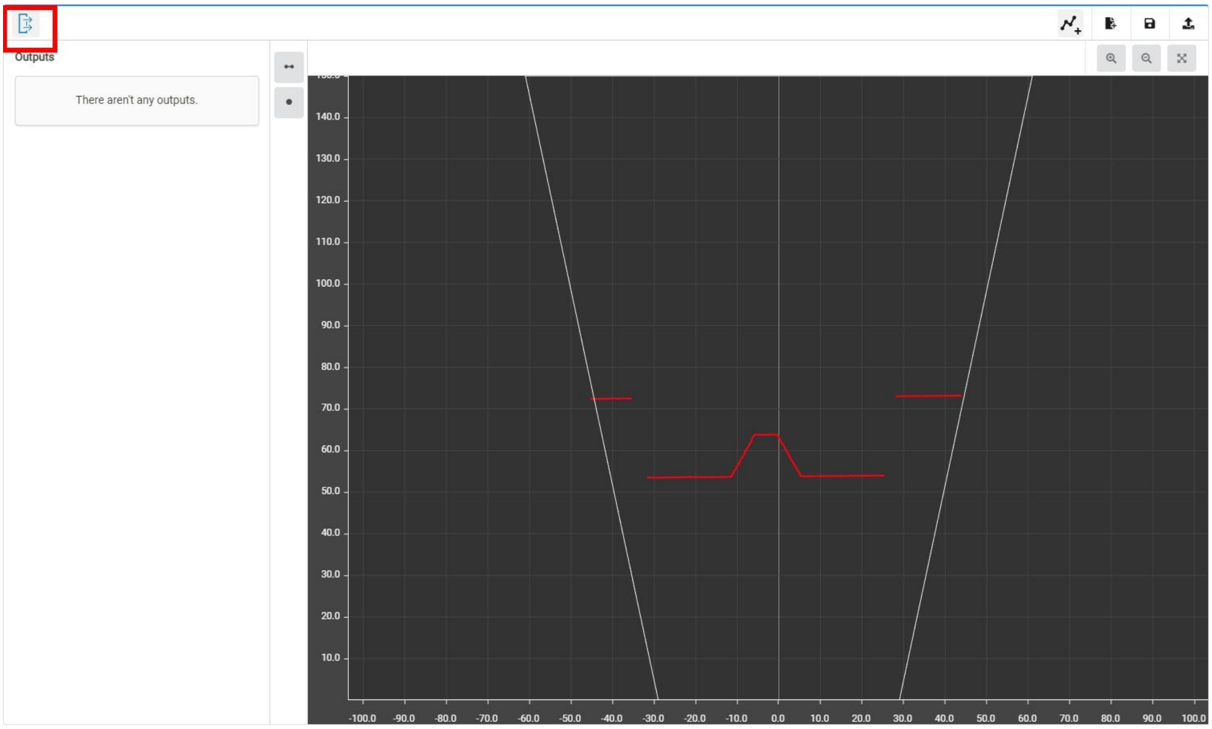
To create a set of templates, follow these steps:

1. Determine the number and types of template set outputs. The created outputs will define the outputs of the "templates detector" smart block. Create the template set outputs.
2. Create all templates one by one:
   a. add a template to a set,
   b. create template elements based on the results of approximation or draw them manually,
   c. add self constraints and mutual constraints for template elements,
   d. add and customize template variants (if necessary),
   e. configure template outputs (associate points and lines with template set outputs created in step 1),
   f. if necessary, replace the generated template icon with a custom one.

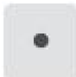
3. Save the set of templates as a file to the non-volatile memory of the scanner (the file can later be used for other scanners).

31.3.1.1. Creating template set outputs

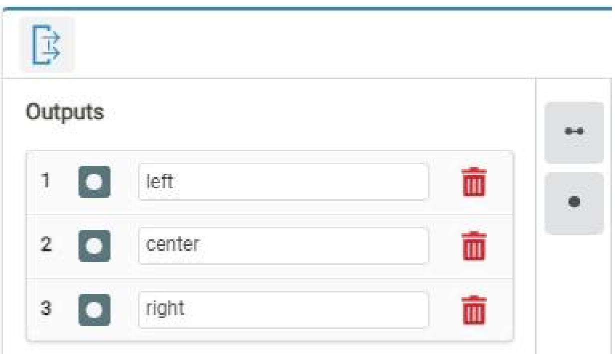
The template set outputs define the outputs of the "templates detector" smart block. Two types of outputs are available: point and line. To access the controls for template set outputs, click the **Template set outputs** button:



To create an output, click the corresponding button:

"line" output	
"point" output	

Outputs are created with a default name (the name can be changed). In the example below, three “point” outputs have been created and their default names have been changed:



31.3.1.2. Adding a template to a set

To add a template to a set, click the **Add template** button:

Templates will be created sequentially. The displayed numbering starts from “1”, however, when selecting a template by an external system (for example, a robot) using the input of the “templates detector” smart block, an index starting from “0” must be specified:

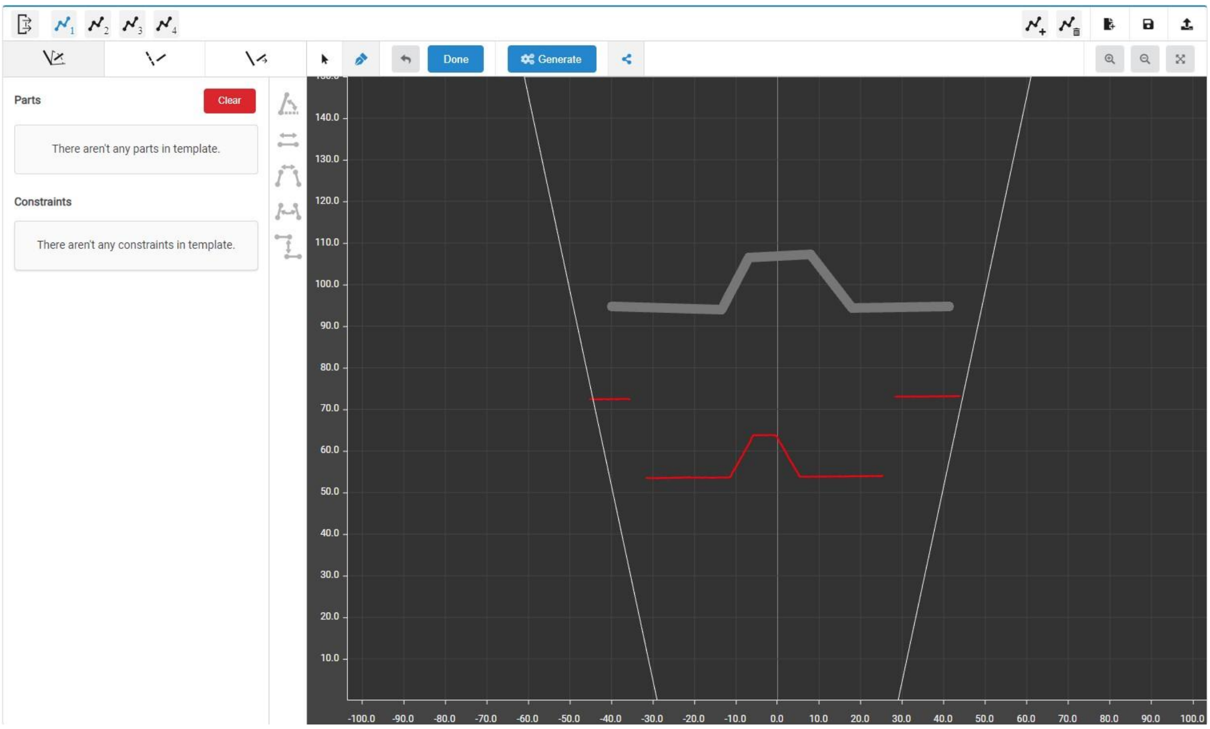
31.3.1.3. Creating template elements

Template elements can be created by the user by sequential drawing of segments, or automatically based on the results of approximation of the current profile.

31.3.1.3.1. Creation of template elements by the user

This mode is activated by clicking the button with the pen symbol:

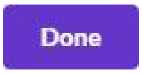
When you click on the coordinate grid, points forming a polyline appear:



IMPORTANT: The shape of the polyline without specified relative constraints (angles between segments and distances between segments) does not matter when searching for a template and only sets the number of elements in the template. The shape of the template is only taken into account when applying relative constraints.

Deleting the drawn segments is done by clicking the following button:

To create the template element, click **Done**:



The drawn elements will be displayed in the list of template elements and will be transferred to the scanner, and the search for these elements in the profile will start.

Parts — Clear

	X	Z	X	Z
1	-31.51	53.48	-11.48	53.63
2	-11.48	53.63	-5.78	63.79
3	-5.78	63.79	-0.43	63.74
4	-0.43	63.74	5.37	53.77
5	5.37	53.77	25.24	53.97

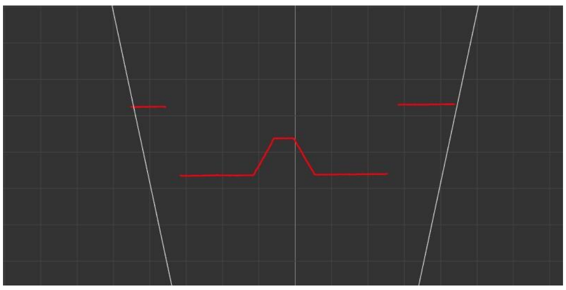
It is necessary to take into account that if no self / relative constraints are specified, the first segments in the profile will be found by the number of elements in the template.

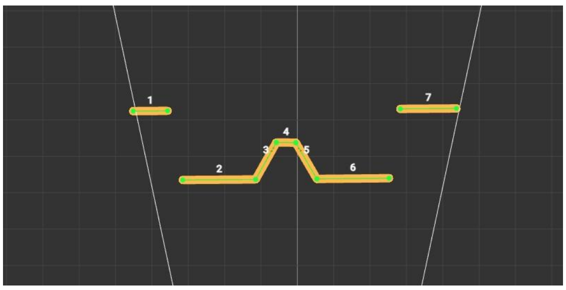
When template elements are deleted (by clicking the Clear button or the trash can icon), all associated template outputs will be reset (become unassociated). Associating (assigning) template outputs to its elements is described in par. "Assigning template output values".

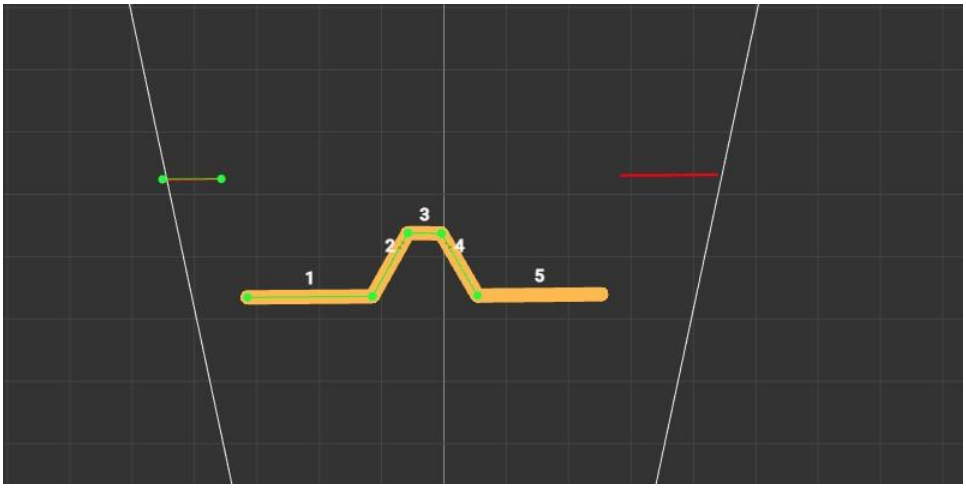
31.3.1.3.2. Automatic creation of template elements

To automatically generate template elements, click **Generate**:

The template elements will be created based on the results of the current profile approximation. Extra segments can be deleted by selecting each of them and clicking the  button.







The note about the shape of the polyline (formed from template elements) in the previous paragraph should be taken into account in this case as well.

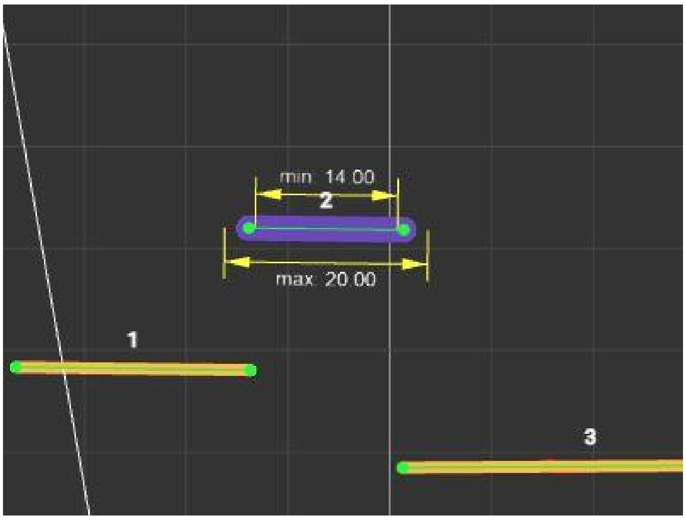
31.3.1.4. Creating constraints

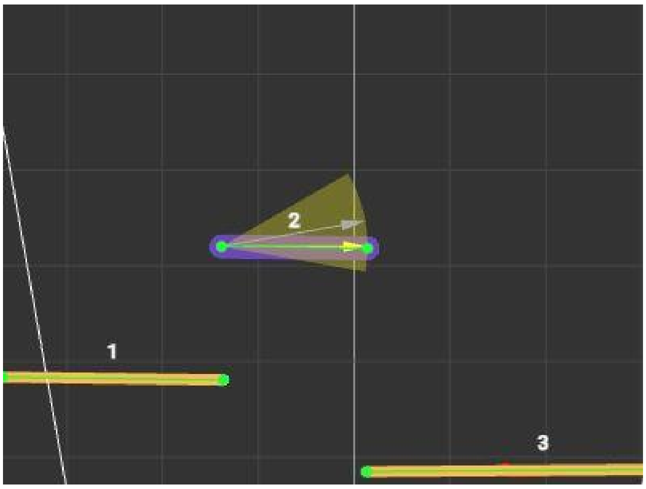
After creating the template elements, it is necessary to add constraints that make it possible to take into account the shape of the template, i.e. permissible change of elements and permissible ratio between pairs of elements.

There are two types of constraints: self constraints and relative constraints.

31.3.1.4.1. Self constraints

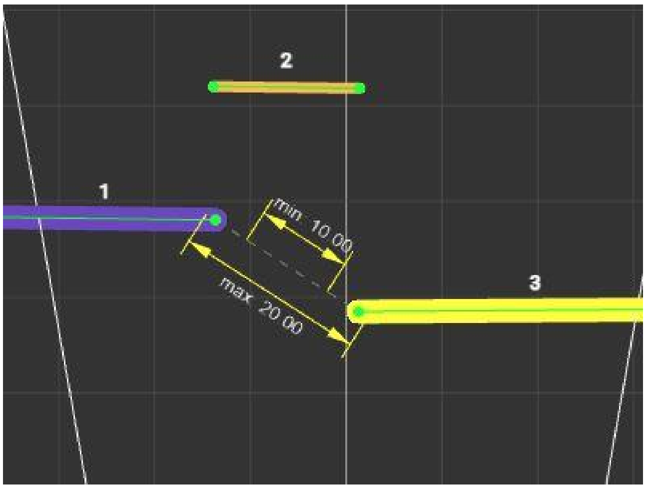
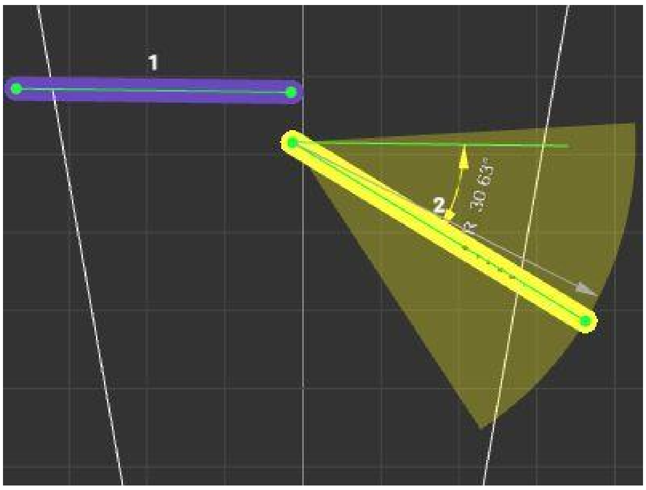
Self constraints are applied directly to the template element. The following self constraints are provided:

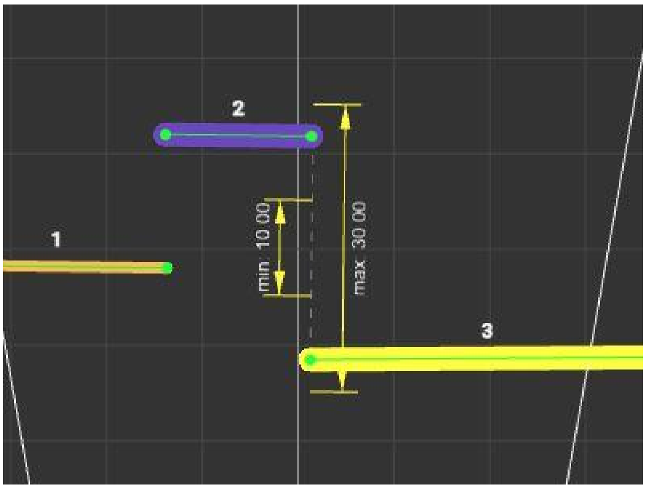
Length	The length of the element (sets the minimum and maximum allowed length of the element). Parameters: • min - minimum allowable length; • max - maximum allowable length. 
Angle	The angle of inclination relative to the horizontal axis. Parameters: • value - target angle relative to the horizontal axis (positive values define counterclockwise direction); • tolerance - deviation from the target angle (on both sides of the target angle).

31.3.1.4.2. Relative constraints

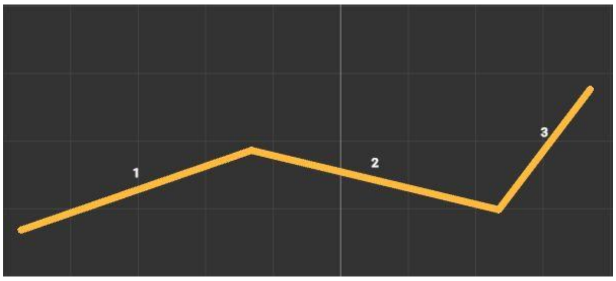
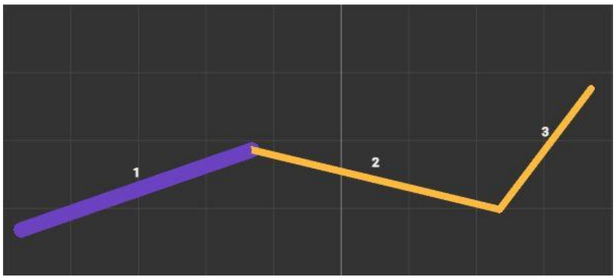
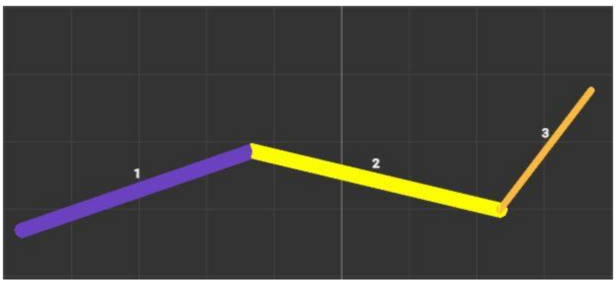
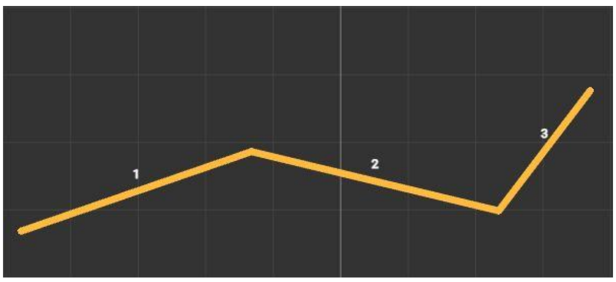
Relative constraints are applied only to a pair of elements and are set from the main element to the relative one. The following relative constraints are provided:

Distance 	The shortest distance from the end point of the main element to the start point of the relative element:  Parameters: • min - minimum allowable distance; • max - maximum allowable distance.
Angle 	The angle between the main and relative elements (the green line indicates the direction of the main element at the beginning of the relative element): 

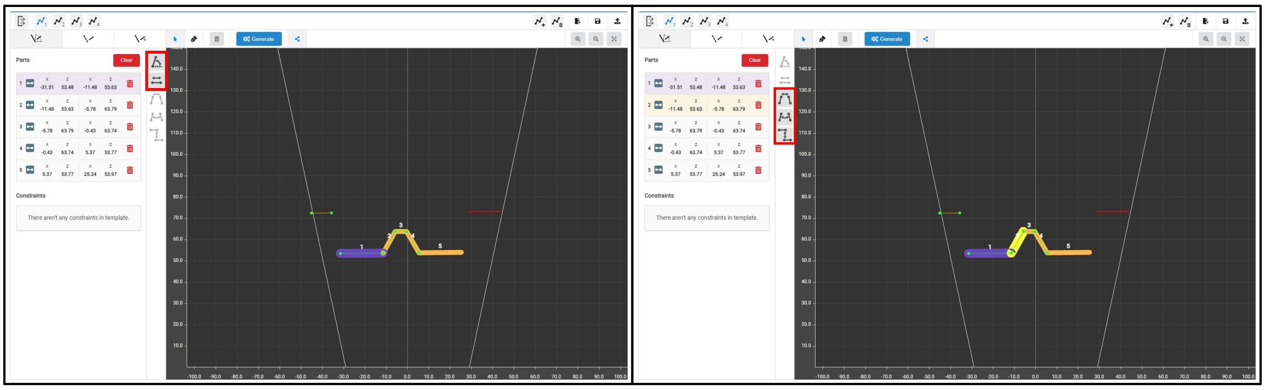
	Parameters: • value - target angle between base elements and relative elements (positive values define the counterclockwise direction); • tolerance - deviation from the target angle (determined on both sides of the target angle).
Distance ortho 	The distance between the starting point of the relative element and the starting point of the main element in the perpendicular direction (the perpendicular is set by the main element):  Parameters: • min - minimum allowable distance in perpendicular direction; • max - maximum allowable distance in perpendicular direction.

31.3.1.5. Adding constraints to a template

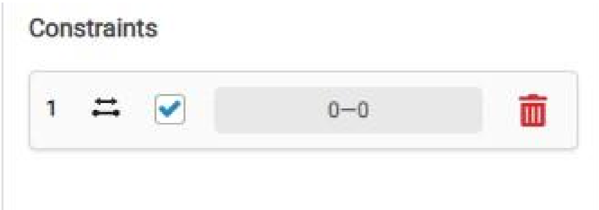
To add a constraint, select the main and relative lines. When you click on a line, the main line is selected first (highlighted in purple). If the main line is already selected, then the next click selects the relative line (highlighted in yellow). When clicking outside the lines of the template, the current selection is cleared.

 	
Original template.	Clicking on the first element of the template. The first element is selected as the main one.
 	
Clicking on the second element of the template. The second element is selected as the relative one.	Clicking outside the elements of the template. The selection has been cleared.

After selecting elements, buttons for adding constraints appear in the template settings area:

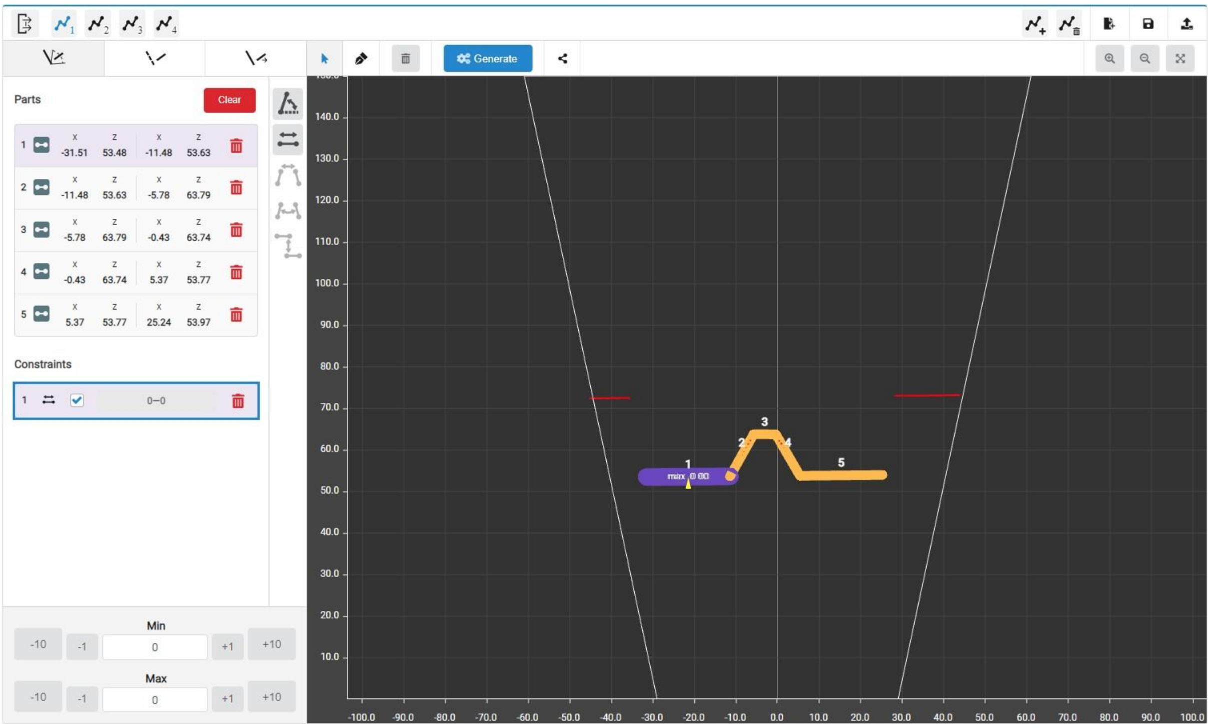


After selecting the required constraint, it will be displayed in the list of constraints:

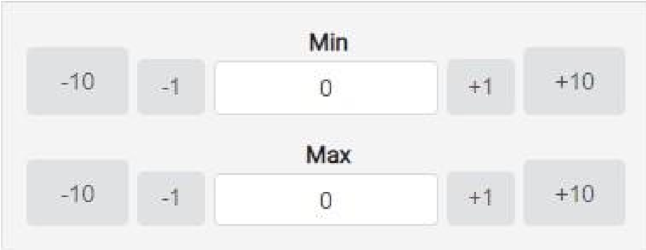


For self constraints, the number of the element for which they apply is indicated. For relative constraints, the numbers of elements for which these constraints apply are indicated. The "checkbox" element in the constraint line enables or disables checking of this constraint when searching for a template. The numeric fields define the constraint parameters and depend on the constraint type.

To delete a constraint, click 🗑 for the constraint you need to delete. Clicking on a constraint in the list selects the constraint. The template displays a graphical visualization of the selected constraint. At the bottom of the panel, an area for configuring the constraint parameters appears:

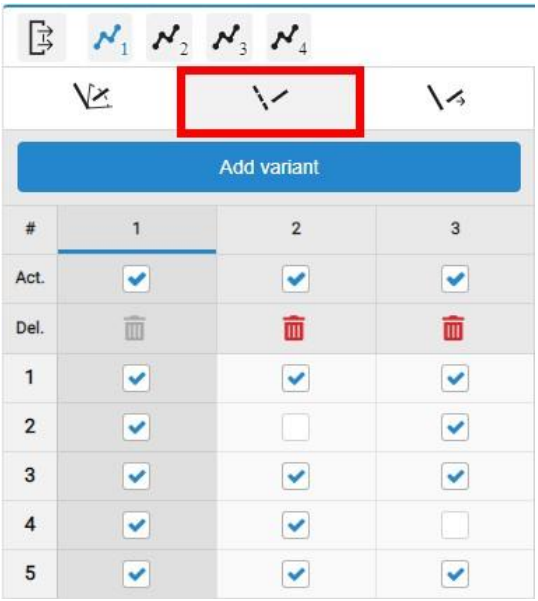


Configuring the selected constraint consists in setting its parameters, the number of which is determined by the type of constraint. To quickly set the required values, the panel contains buttons for increasing and decreasing the current value by 1 and 10 units:



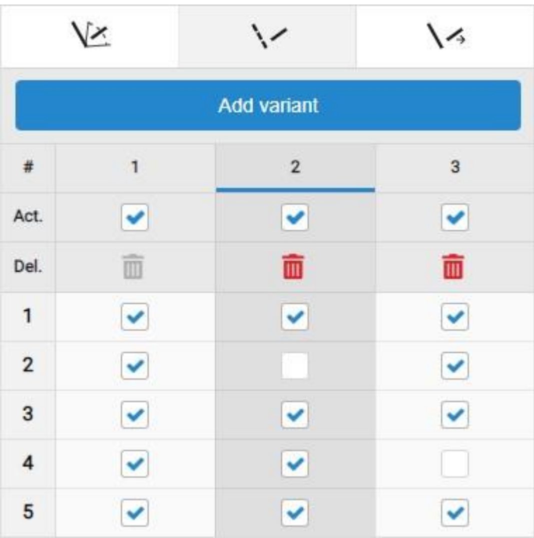
31.3.1.6. Creating and customizing template variants

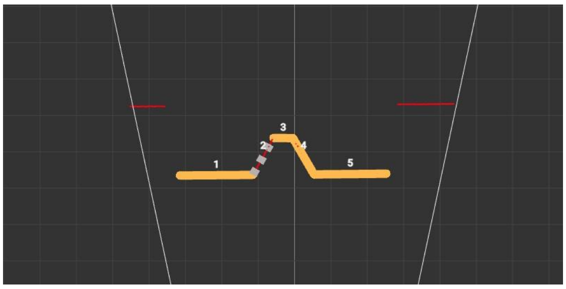
The template variant determines which elements of the template may be missing. When displaying variants, the number of the element is shown along the vertical axis, and the number of the variant - along the horizontal axis. Variant #1 always exists, it must contain all elements of the template

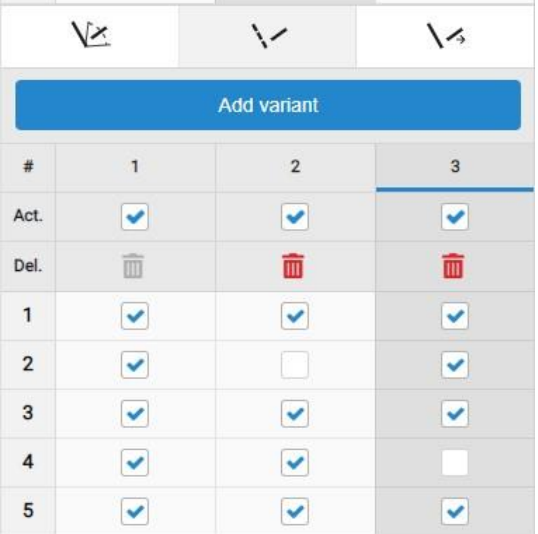


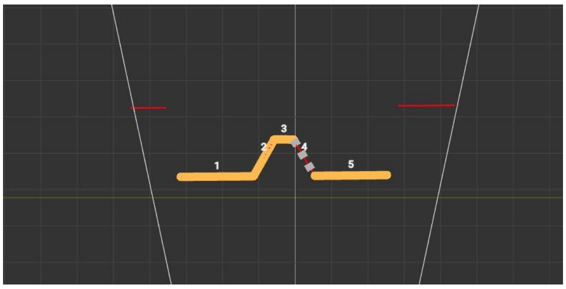
The checkbox in the **Act.** column indicates whether this variant is used in matching. The ✖ button deletes the corresponding variant.

Clicking on the variant number selects that variant for display. Inactive elements of the template are displayed with a dashed line - this means that the element is not necessary for this variant of the template:

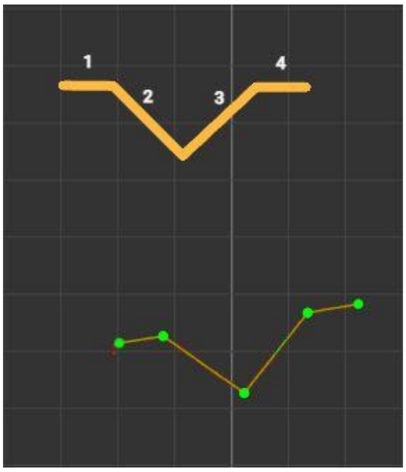
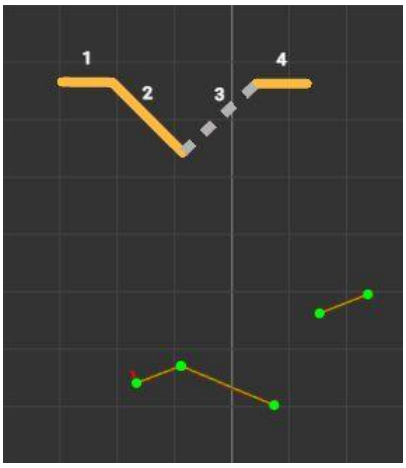
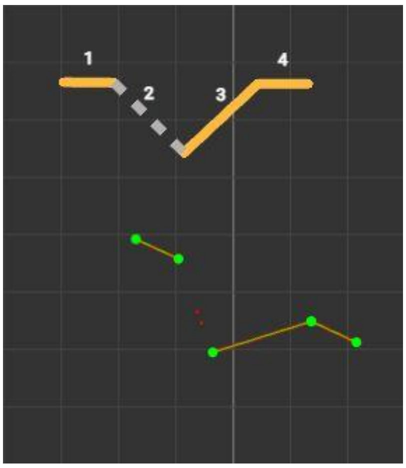






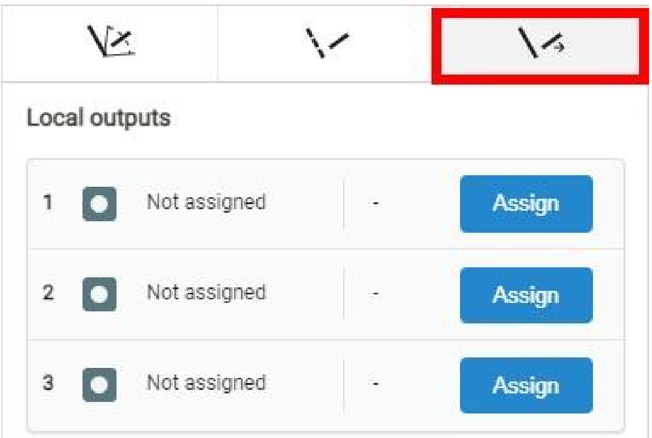


Examples of different variants of the same template:

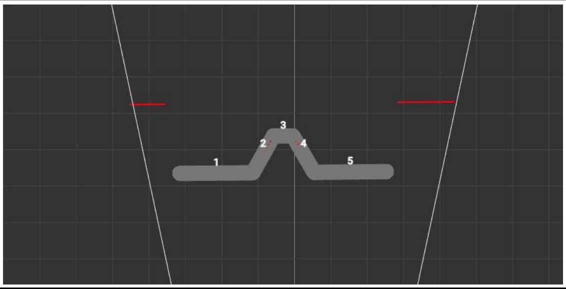
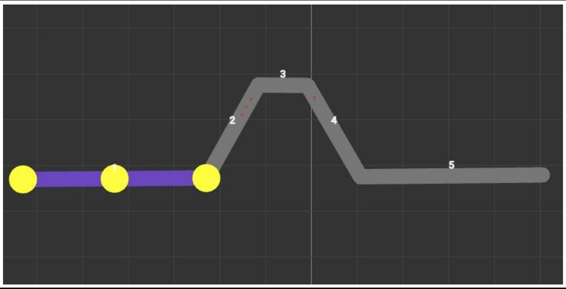
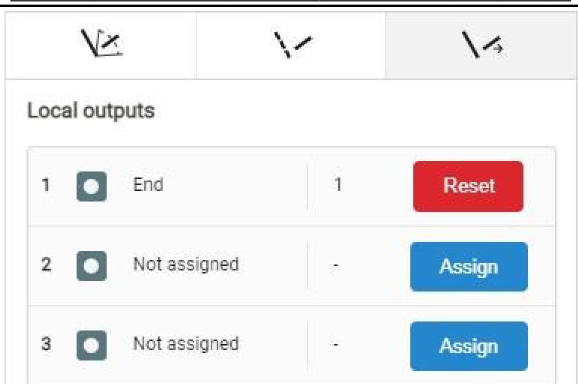
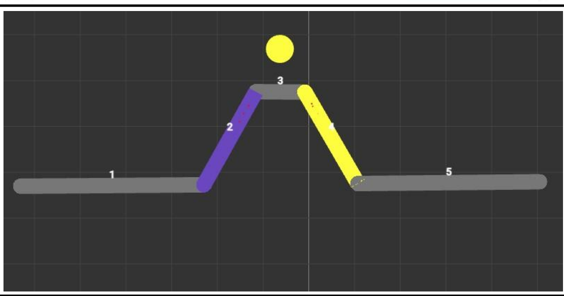
		
Variant 1 - All lines are active	Variant 2 -. Line 3 is not active	Variant 3 - Line 2 is not active

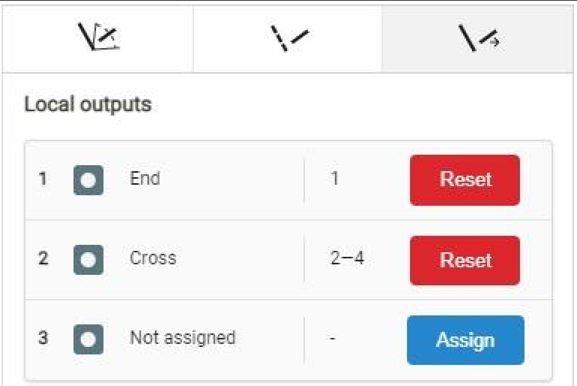
31.3.1.7. Assigning template output values

After creating the template elements, constraints and variants, it is necessary to assign the output values of the template, which will be sent to the outputs of the “templates detector” smart block when the template is detected:



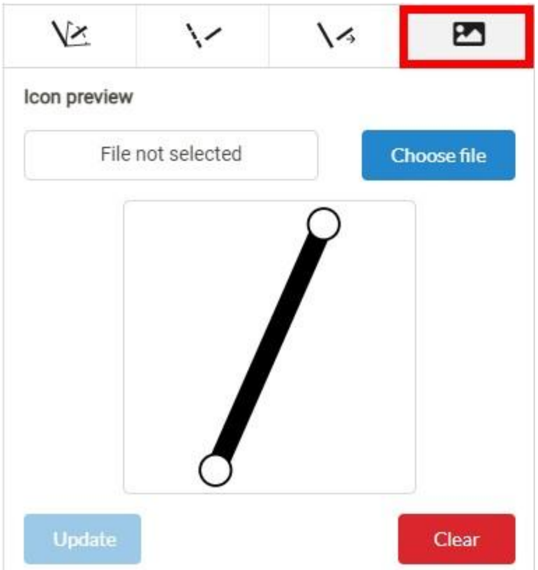
To assign output data to an output, click the **Assign** button next to the corresponding output. After that, the variants available for assignment will be displayed:

<table>
<tr><td></td><td>After clicking the “Assign” button, the elements of the template are highlighted in gray. You must select a segment.</td></tr>
<tr><td></td><td>Segment #1 is selected - three points are available for assignment (the beginning of the segment, the middle of the segment and the end of the segment).</td></tr>
<tr><td></td><td>After clicking on the yellow circle on the right side of the segment (the end of the segment), the template output is assigned to the end point of segment #1.</td></tr>
<tr><td></td><td>Option to select the point of intersection of two segments (segment #2 and segment #4 are sequentially selected).</td></tr>
</table>

	After clicking on the yellow circle (point of intersection), the point of intersection of segments #2 and #4 is assigned to the template output.

31.3.1.8. Assigning a custom icon to a template

Each template can be assigned a custom icon, which is displayed in the settings of the “template detector” smart block. When creating a template, an icon is generated automatically based on the elements of the template. To replace the generated icon, select the icons tab in the template settings:

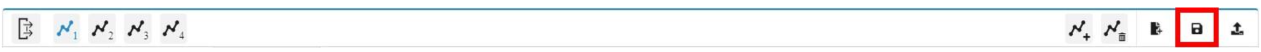


Click the **Choose file** button and select the icon file (taking into account the constraints specified in par. “Template structure and search principle”). The downloaded icon will be displayed immediately. To assign the icon to the template, click the **Update** button, which will change the icon in the list of templates:

	
Automatically generated icon of template #15	Custom icon of template #15

31.3.1.9. Saving a set of templates to the non-volatile memory of the scanner

The template set must be saved as a file with the “.template” extension. To save, click the **Save template set** button:



You will see a window with a list of existing files, where you can select a file and overwrite it, or enter a new file name and save it.

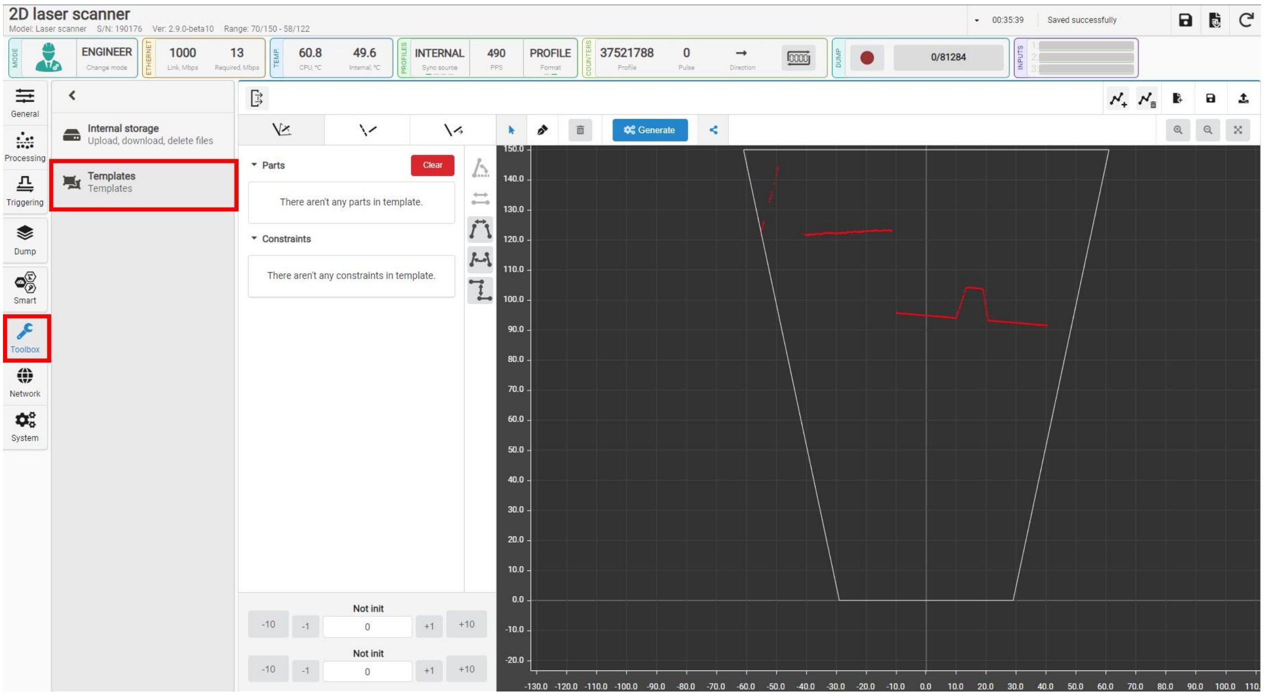
31.3.2. Example of creating a custom template set

For an example of creating a custom template set, a part with a trapezoid cut will be used.

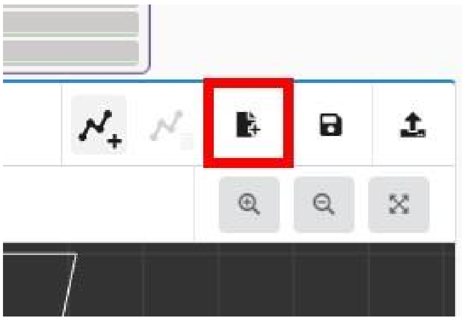
31.3.2.1. Step 1. Creating the template set file

The “templates detector” smart blocks use files specially prepared in the web interface as initial data for template detection. These files have the “.template” extension and are stored in the non-volatile memory of the scanner (the files can be used with other scanners).

To create a new template file (as well as to edit existing ones), go to the **Templates** section:



Click the “New templates file” button (located in the upper right corner), if the template file has not been opened or created before, this step can be skipped:

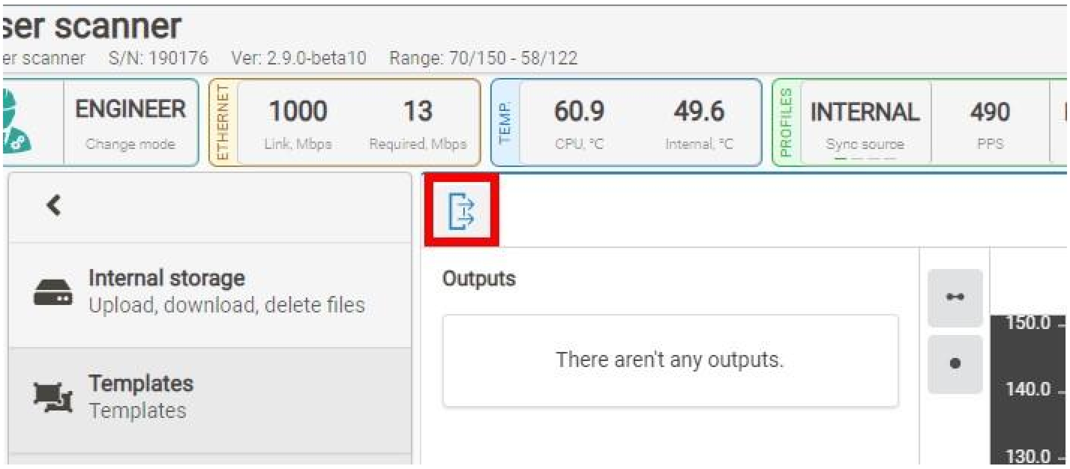


31.3.2.2. Step 2. Adding template set outputs

Each template set has a set of outputs specified by the user during the creation of the template set file. Based on these outputs, the outputs of the “templates detector” smart block will be created in the computation graph interface to connect smart blocks (for example, calculations of intersections, average, etc.).

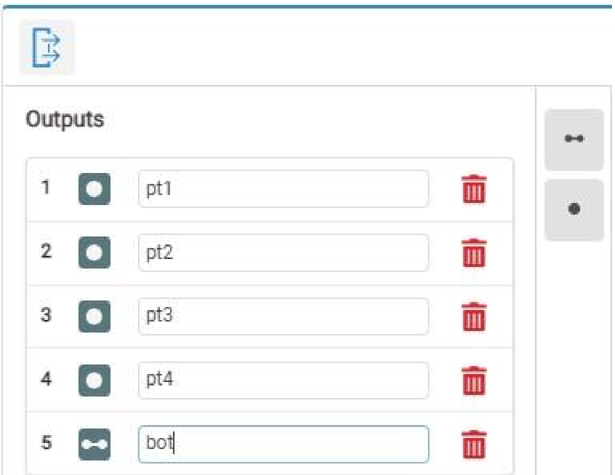
The number of outputs and their types depend on the intended post-processing of the template detection results. It is necessary to pay attention to the fact that the outputs are the same for all templates of the set, but for each individual template a specific output value can be set (for example, the coordinates of end point of the segment or the point of intersection of the segments).

To add/delete outputs, go to the corresponding section:



In the example below, let's add the following set of outputs: 4 points that will correspond to the cut-out angles and a segment (the bottom of the cut-out):

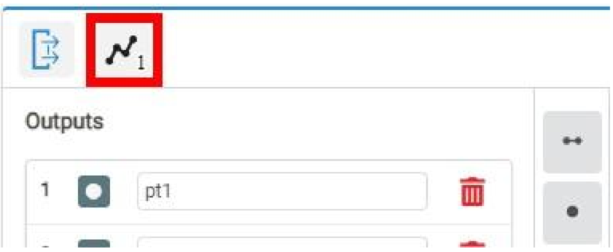
- Add 4 points by clicking the "add point" button .
- Add a segment by clicking the "add segment" button .



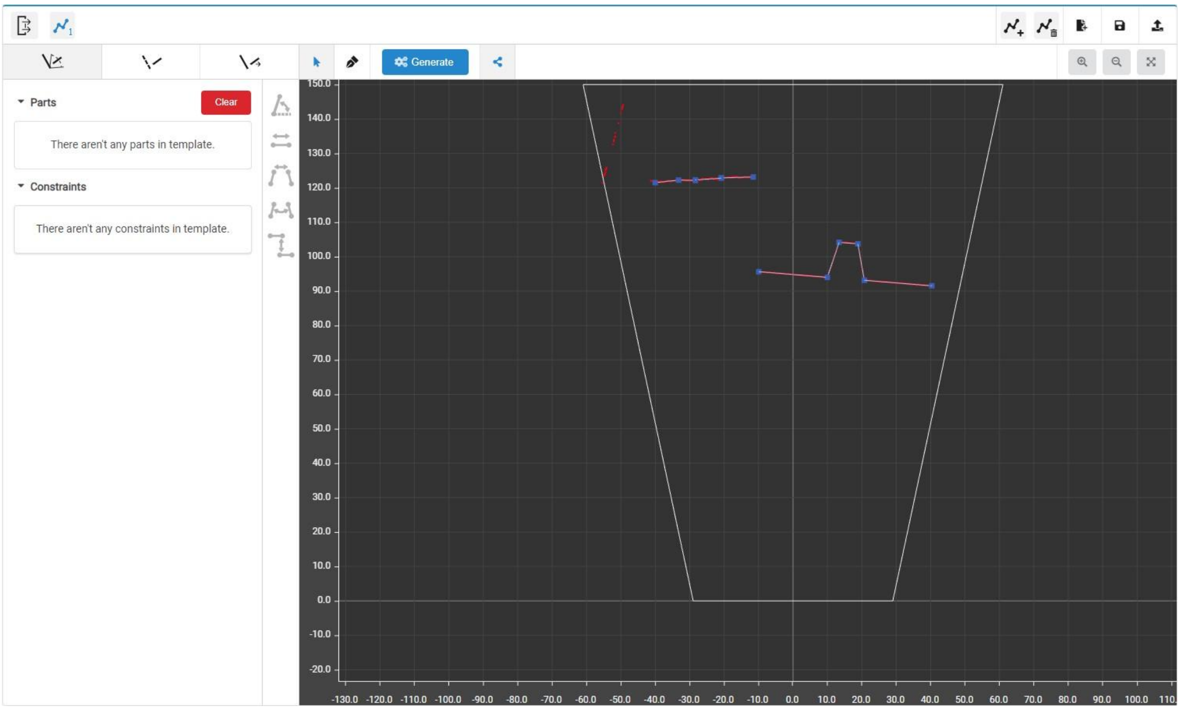
31.3.2.3. Step 3. Adding a template

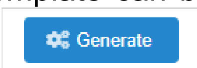
Each template consists of a set of segments (in future releases, also arcs), a set of self constraints, and a set of relative constraints.

To add a template to the set, click the “add template” button .

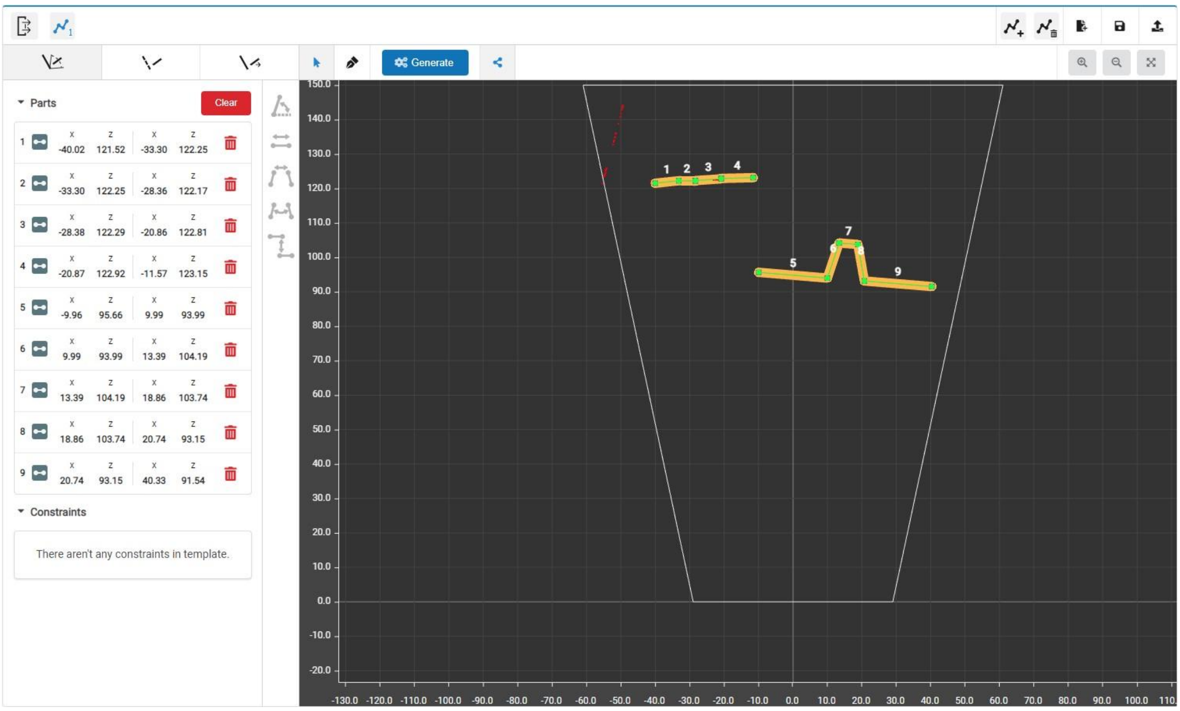


Click the added template button to go to the template editing section:

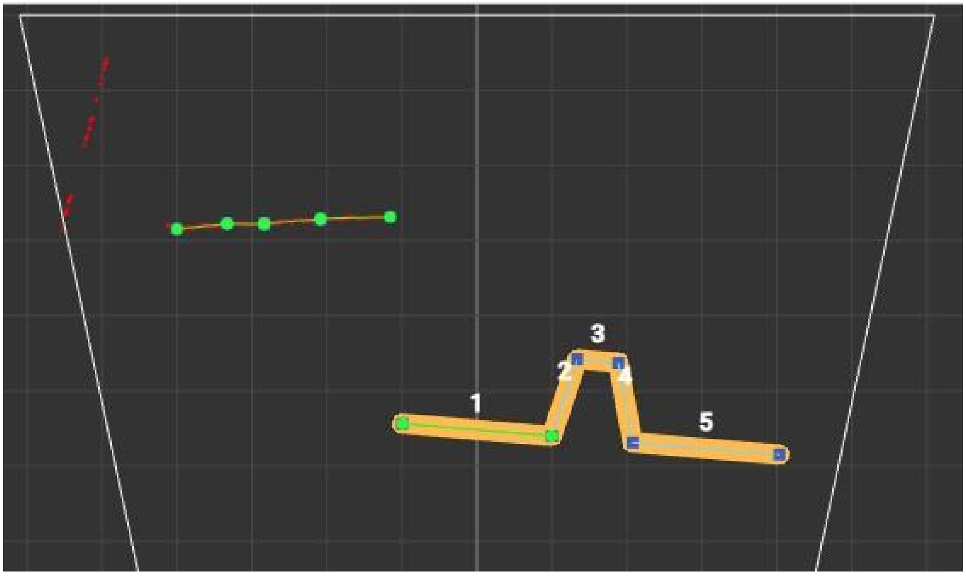


Adding segments to the template can be done either manually, using the  button, or automatically, using the 

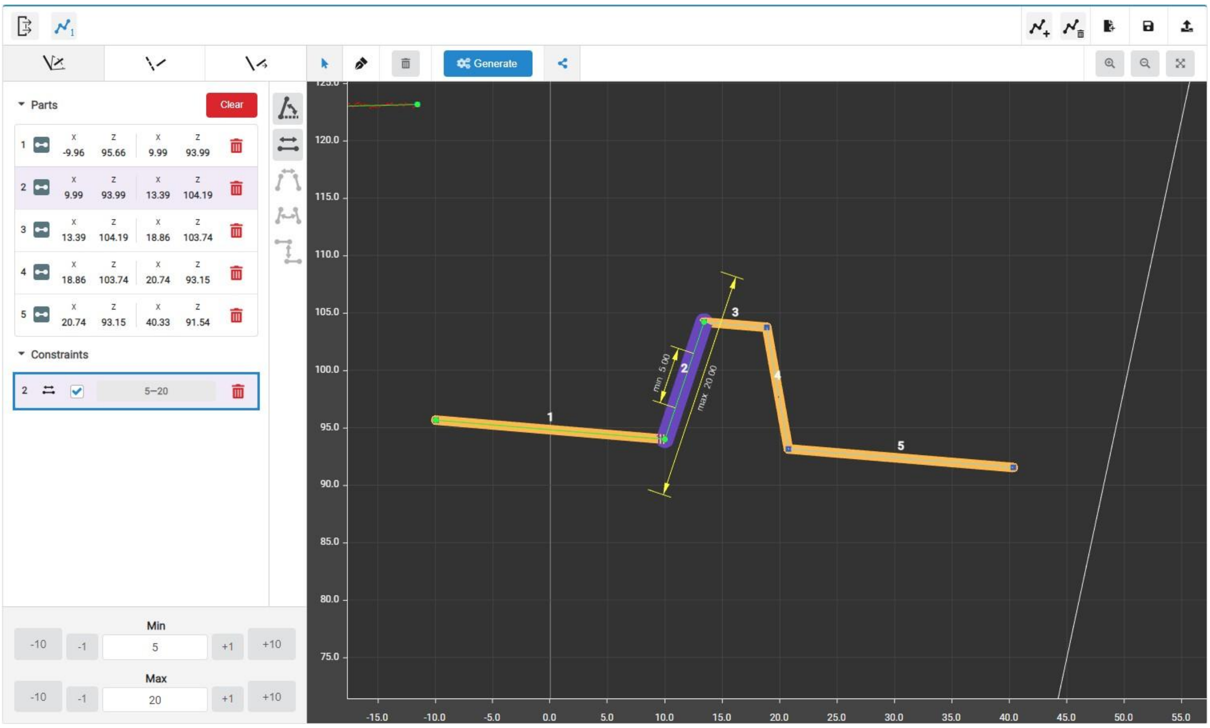
button. Using automatic generation, we get a set of segments containing segments that do not belong to the desired template (1, 2, 3, 4), as well as those that belong to the template (5, 6, 7, 8, 9):



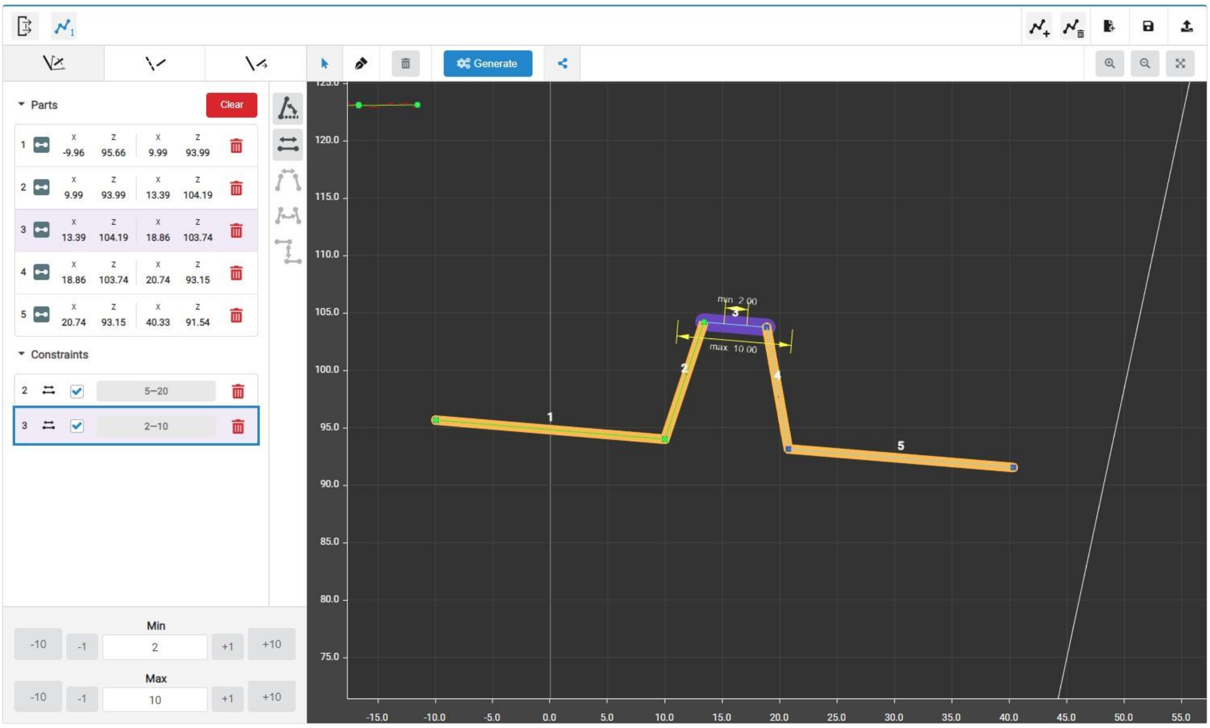
The segments that do not belong to the desired template must be deleted - select them with the left mouse button and press the **Delete** key on the keyboard or the  symbol in the line of the segment to be deleted. Only those segments that form the template should remain:



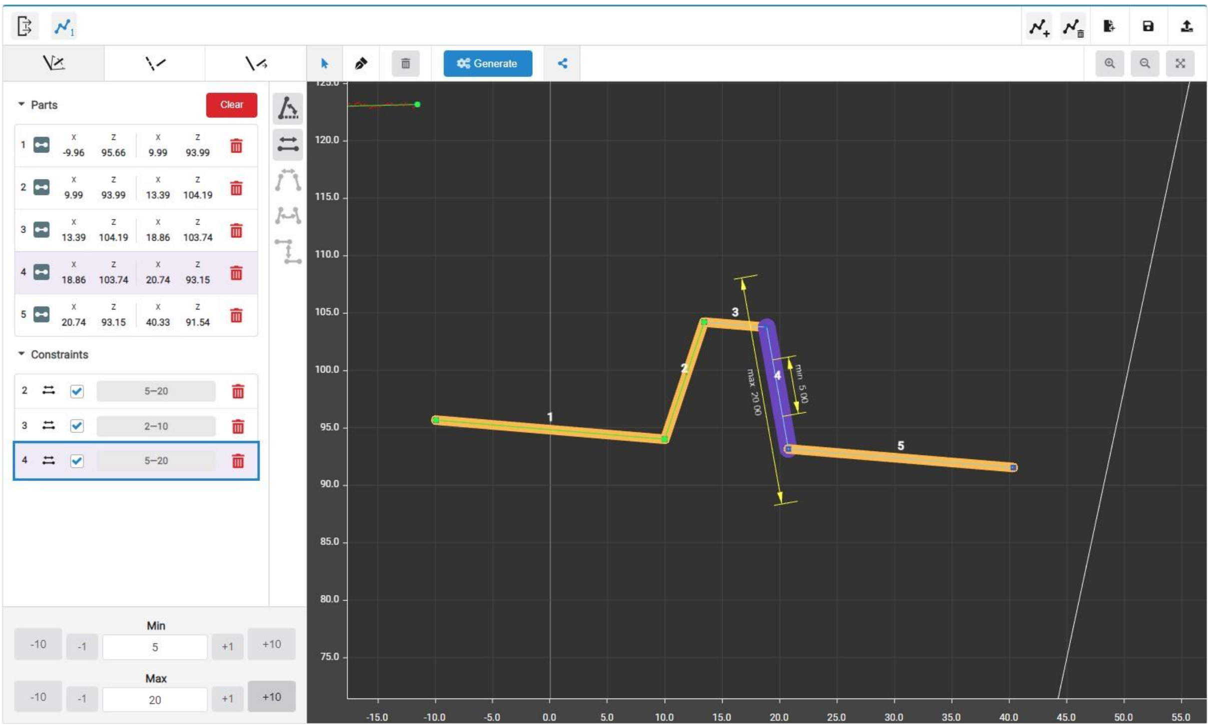
To detect a template in a profile, self constraints and relative constraints must be added. In the given example, it is necessary to apply constraints to the length of segments 2, 3 and 4 - let's set self constraints for them.



Segment 2 cannot be shorter than 5 mm and longer than 20 mm.

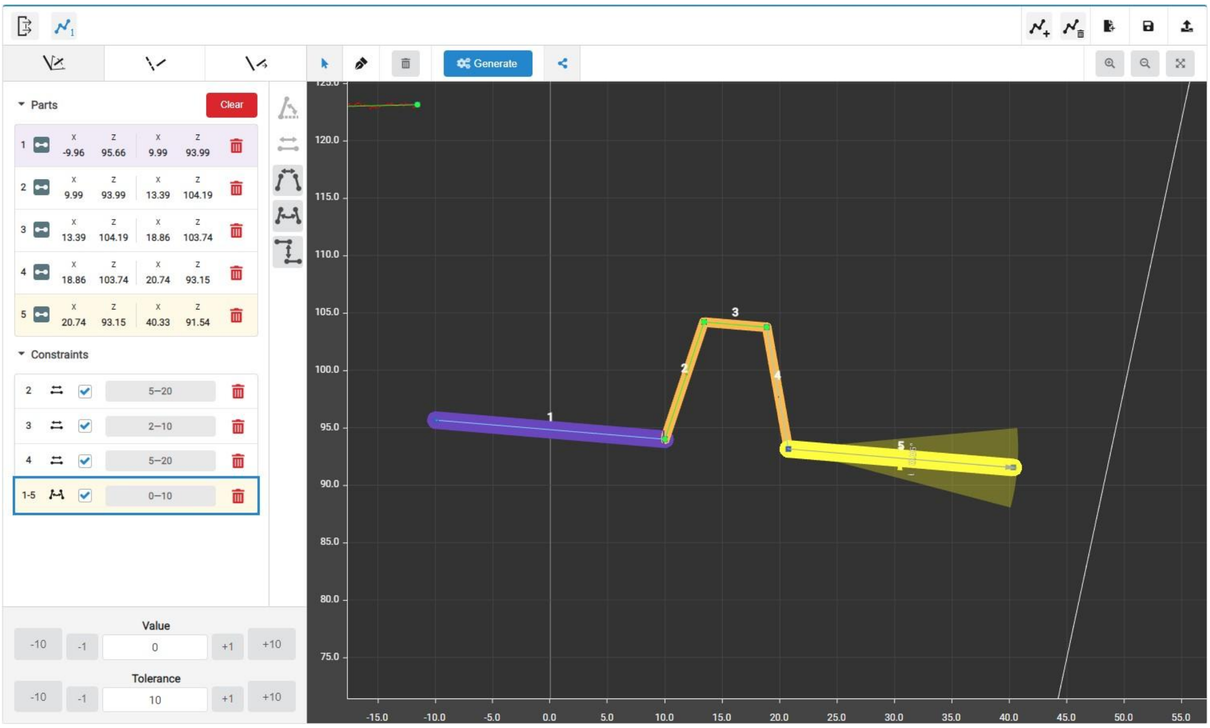


Segment 3 cannot be shorter than 2 mm and longer than 10 mm.

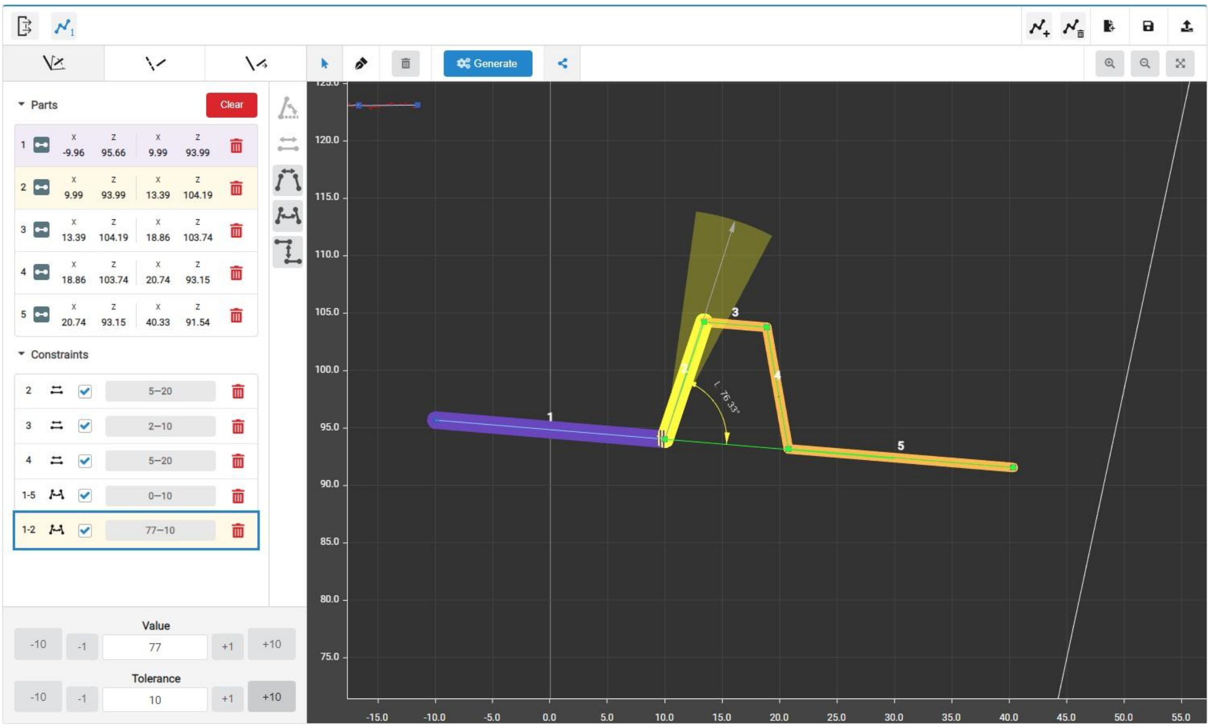


Segment 4 cannot be shorter than 5 mm and longer than 20 mm.

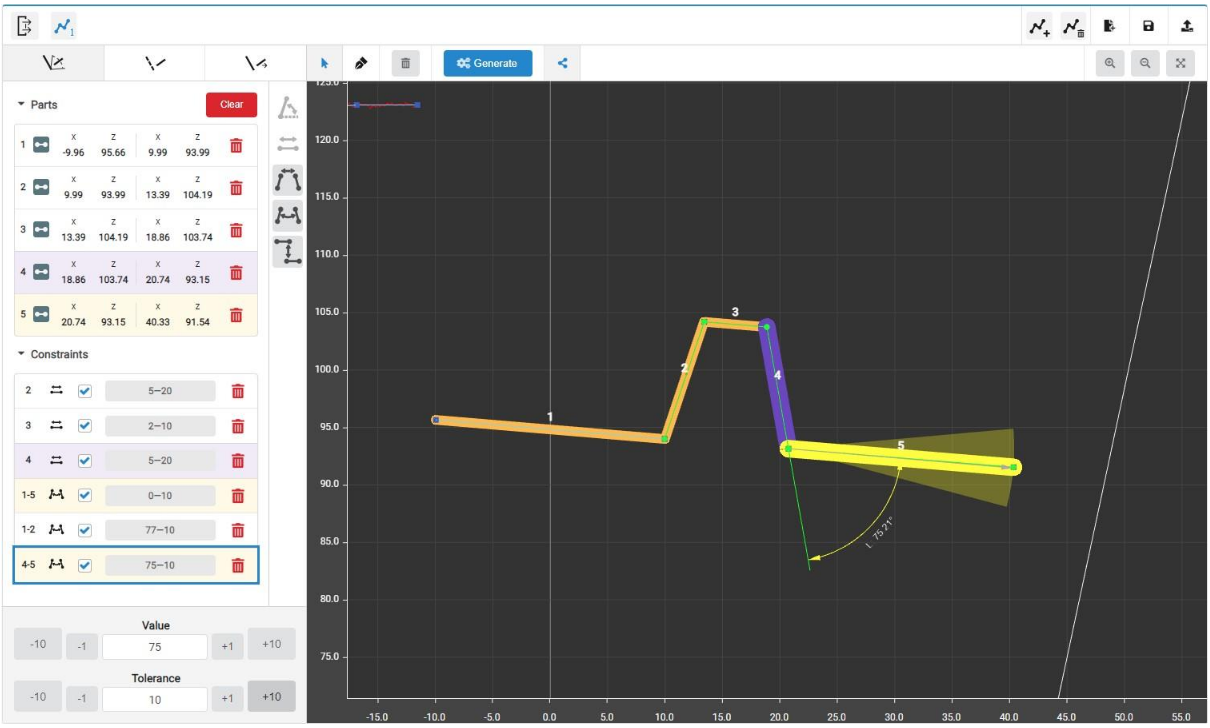
To account for the shape of the template, the following relative constraints must be added:



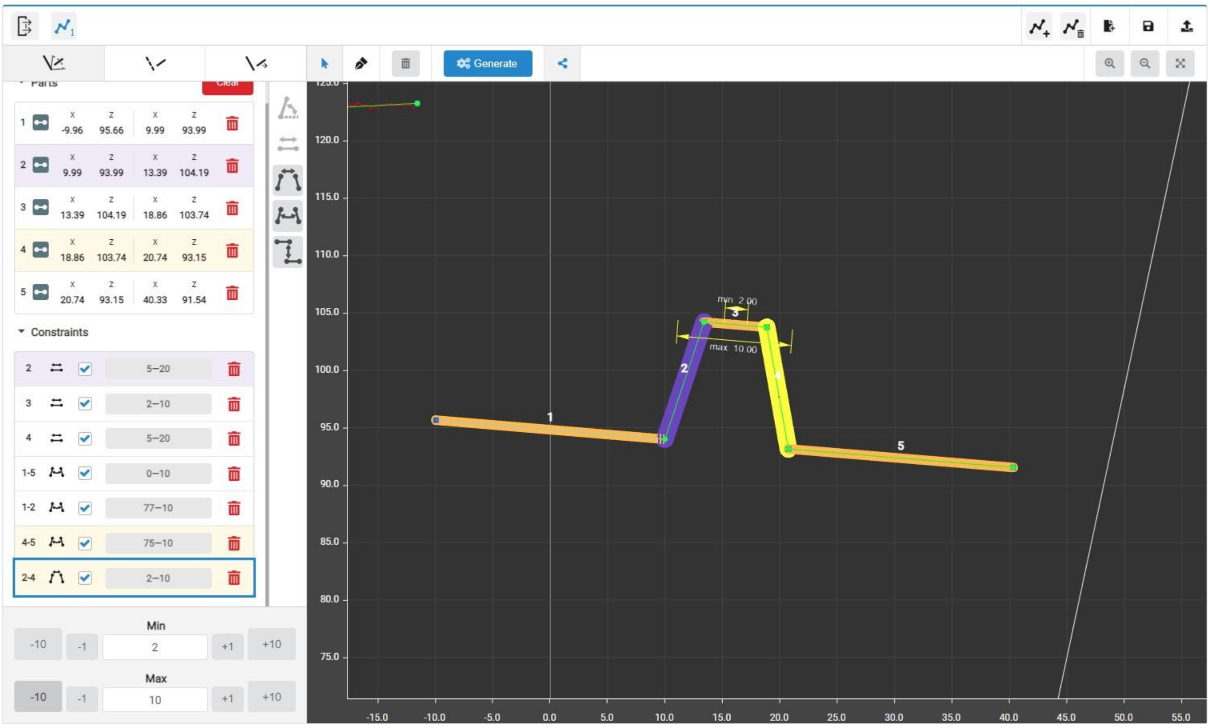
The angle between segments 1 and 5 must not exceed ±5° (angle amplitude is 10°).



The angle between segments 1 and 2 must be 77° with a tolerance of 5° in any direction.



The angle between segments 4 and 5 must be 75° with a tolerance of 5° in any direction.

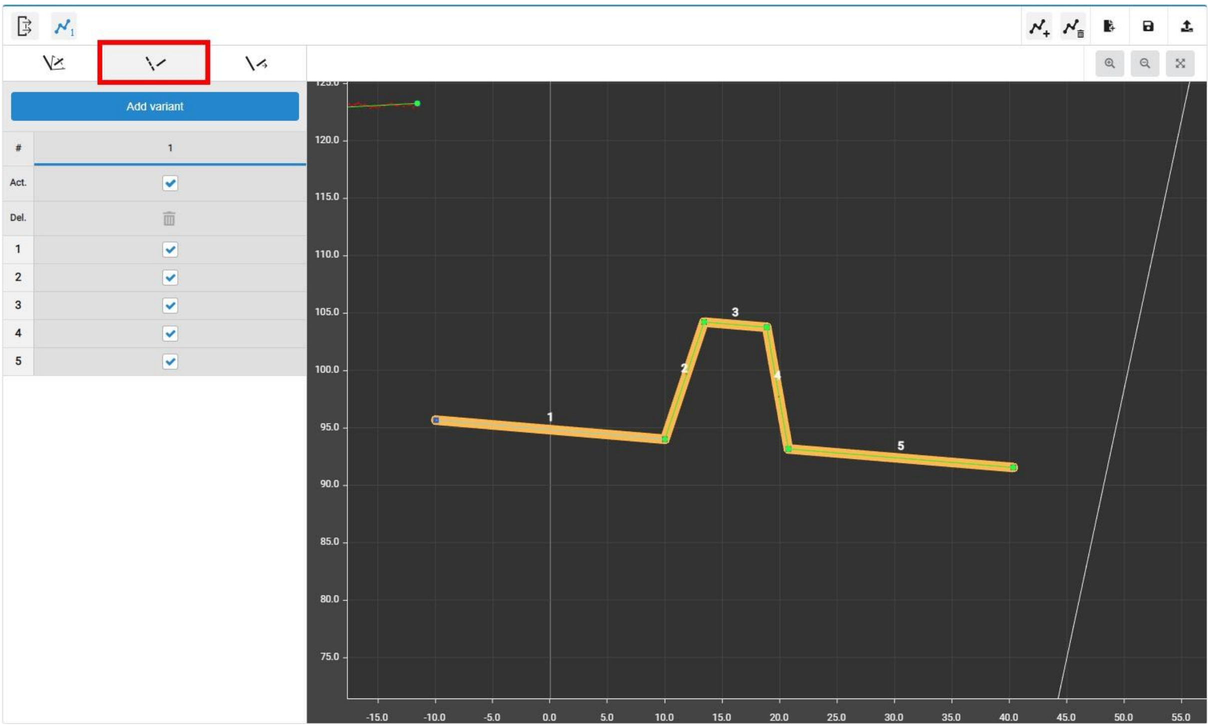


The distance between the end of segment 2 and the beginning of segment 4 must be at least 2 mm and not more than 10 mm.

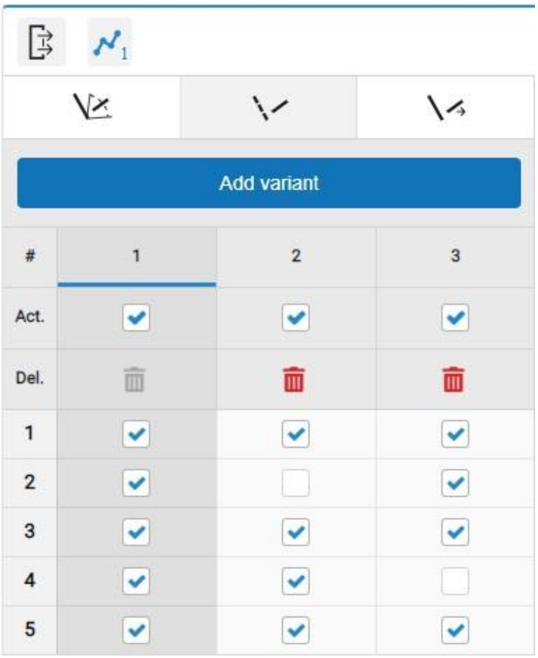
31.3.2.4. Step 4: Adding template variants (if necessary)

Template variants make it possible to take into account the possible absence of segments (or arcs) in the profile.

Template variants can be edited in a special section:



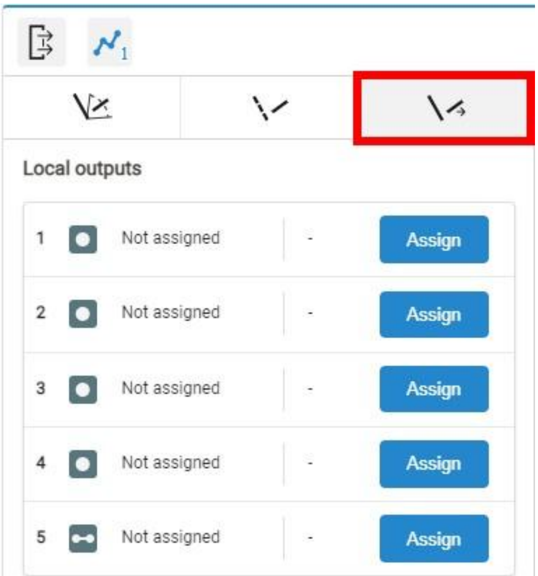
For the cutout shape used in the example, segments 2 and 4 may disappear, so it is advisable to add variants in order to detect the template in such cases:



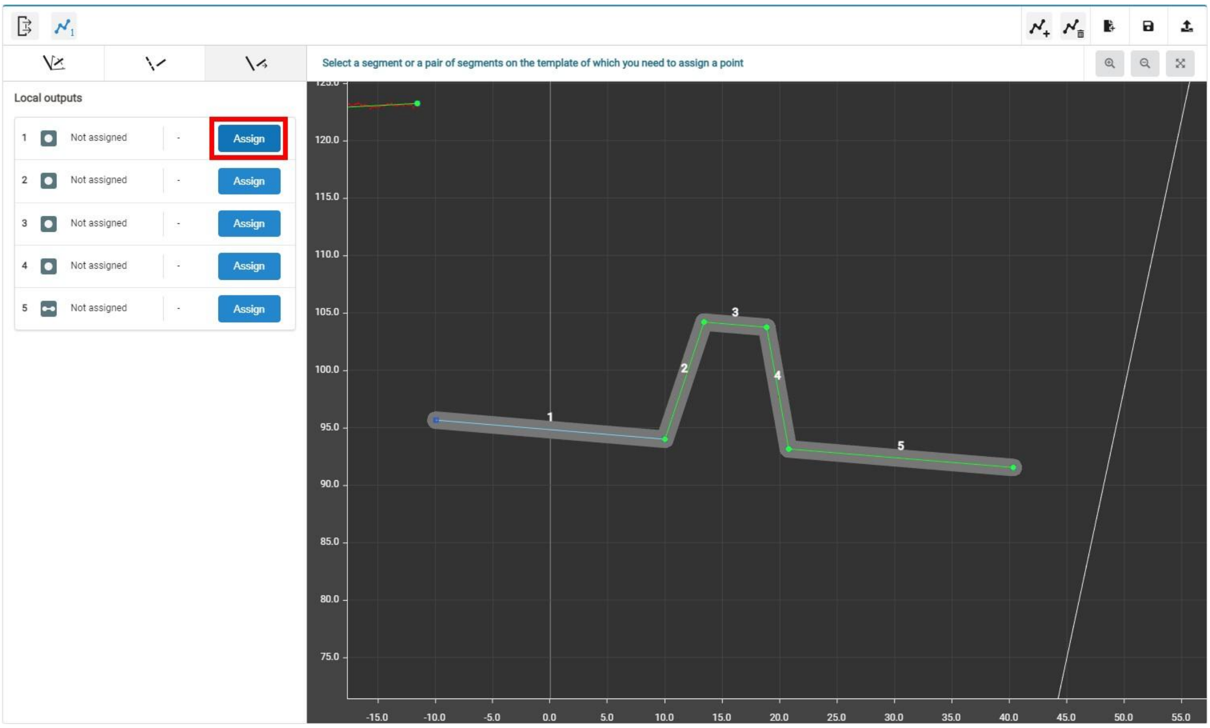
31.3.2.5. Step 5. Assigning template outputs

In the given example, four points and one segment should come at the output of the template set. For this particular template, these should be the corner points and the bottom of the cutout (as mentioned at the beginning when analyzing the required outputs).

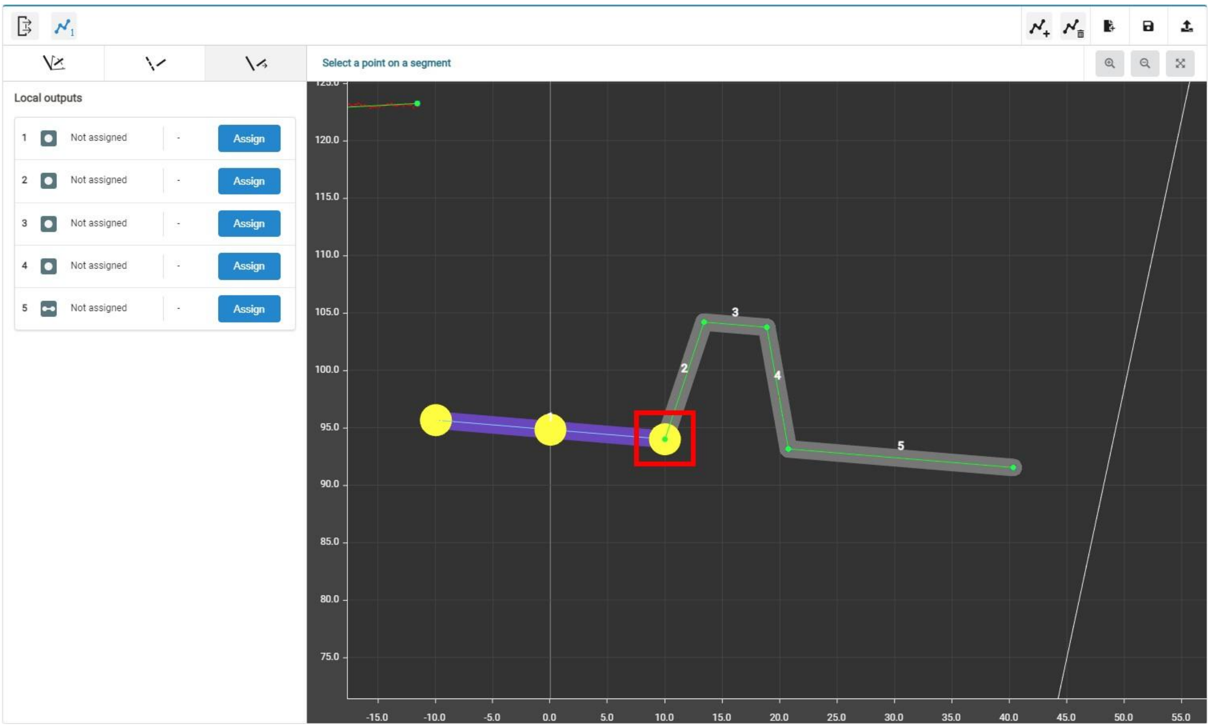
The assignment is made in the **Local outputs** section:

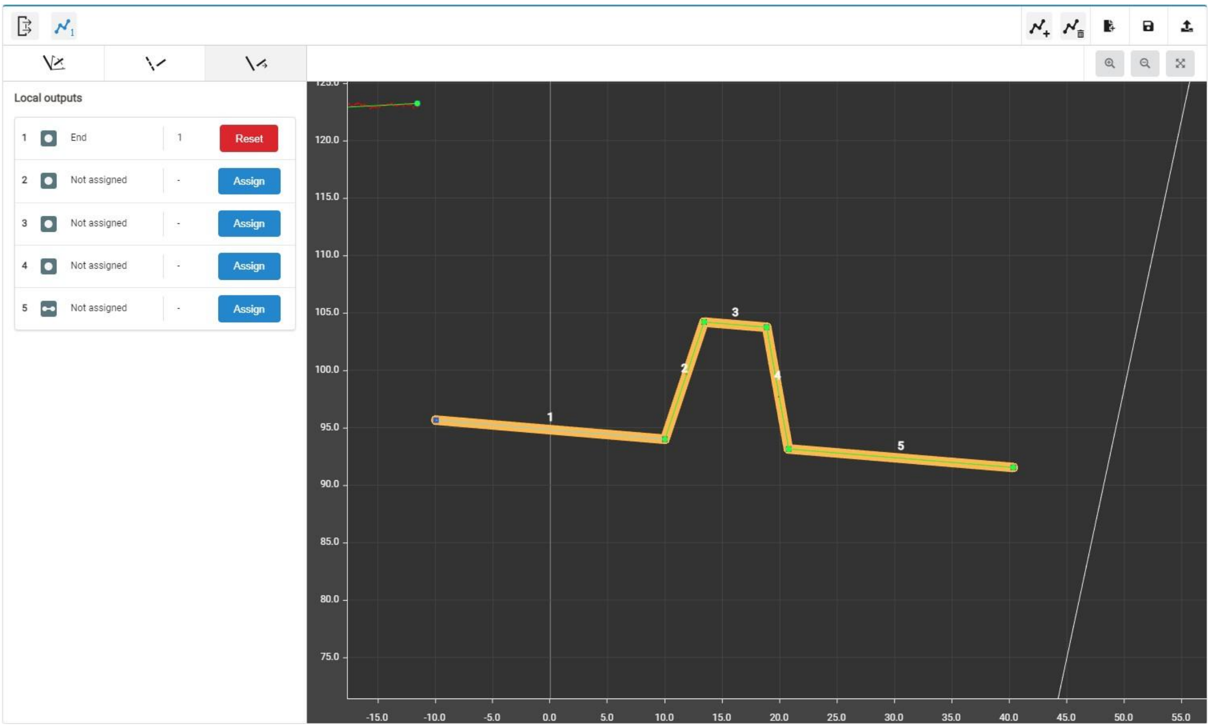


Assign the outputs of the template by clicking the **Assign** button.



Add the first point (the other three points are added in the same way):

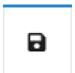




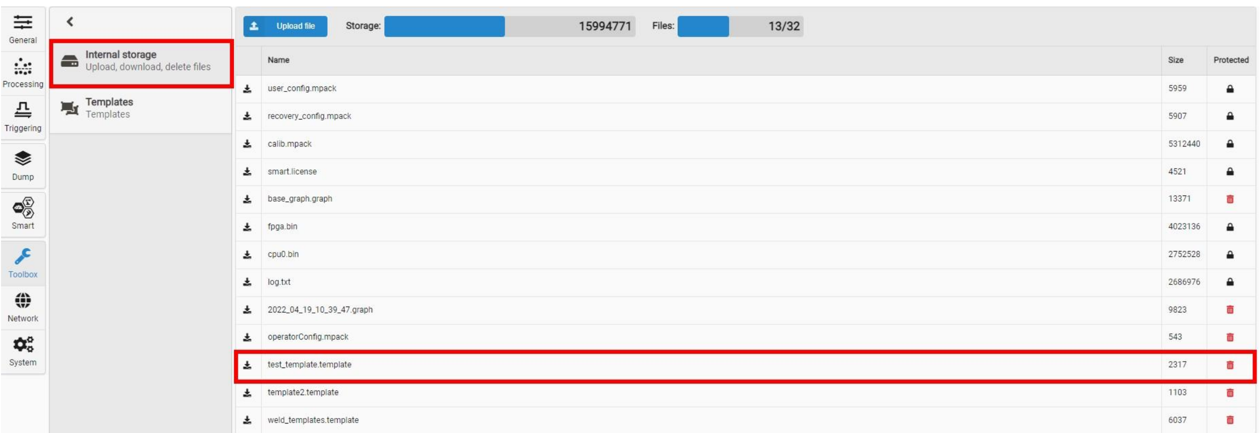
All points and segment added:

31.3.2.6. Step 6. Saving the template file to the non-volatile memory of the scanner

Template files must be saved in the non-volatile memory of the scanner and assigned to the “templates detector” smart block.

To save the template file, click the  button and enter the file name:

Click the **Save** button to save the template file. You can check the existence of the saved file in the **Internal storage** section:



32. Annex 5. "C-script" smart block

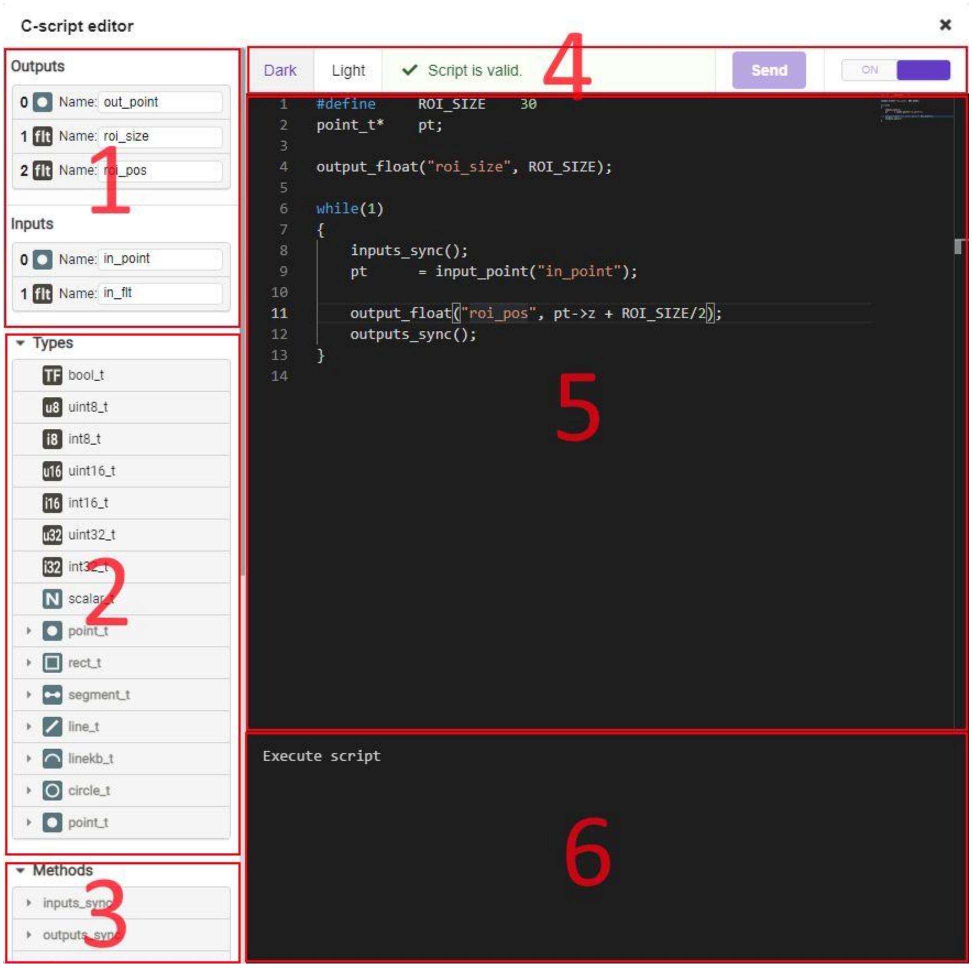
32.1. General information

The “C-script” smart block is intended for editing and executing custom scripts in the C-like language “rfc”. The script is validated before execution - in case of errors, information about them will be displayed in the script editor console. Interaction with other smart blocks is carried out using the input and output dynamically created ports. Ports are addressed by name, which can be set via the port context menu or in the editor.

Existing restrictions:

- the preprocessor is implemented with restrictions, the directives “#define”, “#if”, “#ifdef”, “#else”, “#endif” should be used with caution;
- declaration of functions is only supported by the ANSI standard;
- function pointers are not supported;
- the following declarations are ignored: static, extern, volatile, register and auto;
- structures and unions must be declared globally, bit fields are not supported.

A special editor is provided for creating and editing script texts:



The editor window is divided into the following areas:

1. Inputs and outputs of the smart block with specifying the I/O data type and name. The user can change the name, taking into account that only ASCII characters are allowed and the length of the name should not exceed 60 characters.
2. List of data types supported by the script.

3. List of special methods. By clicking on a method, its prototype will be inserted into the script editor.
4. Editor theme, script validation and execution.
5. Script editing area.
6. Console for displaying errors and messages.

32.2. Supported data types

The set of data types available to the user includes basic types, extended types, and special data types.

The basic types are the standard C language types:

Type	Size, byte	Min	Max
char	1	-128	127
unsigned char	1	0	255
short int	2	-32768	32767
short unsigned int	2	0	65535
int	4	-2147483648	2147483647
unsigned int	4	0	4294967295
long int	8	−(2^63 - 1)	2^63 - 1
long unsigned int	8	0	2^64 - 1
float	4	±1.5 * 10^(−45)	±3.4 * 10^38

Extended data types: bool_t, uint8_t, int8_t, uint16_t, int16_t, uint32_t, int32_t.

Special types are used to work with the input and output ports of the block. They are consistent with the data types used inside the computation graph:

Type	Size, byte	Description
scalar_t	4	Scalar value. In the current version, it is represented by the "float" data type.
point_t	8	Point coordinates: { float x; float z; }
rect_t	16	Rectangle parameters: { point_t topLeft; float w; float h; }
segment_t	16	Line segment: { point_t p1; point_t p2; }
line_t	12	Infinite line specified by abc coefficients: { float a; float b; float c; }
linekb_t	9 (12 with alignment)	Infinite line specified by slope factor k and offset b. In addition, there is a line validity flag "valid", the value of which is: 1 - the line is not strictly vertical (k != infinity), 0 - the line is not valid - cannot be represented by this expression. { float k; float b; bool_t valid; }
circle_t	12	Circle specified by center coordinates and radius: { point_t center;

Type	Size, byte	Description
		float r; }
arc_t	25 (28 with alignment)	Arc specified by start and end points, center coordinates, circumscribing circle radius, and convex/concave flag: { point_t p1; point_t p2; point_t center; float r; bool_t convex; }

point_3d_t	12	Point in 3D space:
		{
		float x;
		float y;
		float z;
		}

quat_3d_t	16	Rotation quaternion in 3D space:
		{
		float w;
		float x;
		float y;
		float z;
		}

euler_3d_t	16	Euler angles of rotation in 3D space:
		{
		uint32_t order;
		float ax;
		float ay;
		float az;
		}
		“order” - order of angles:
		EULER_XYZ = 36,
		EULER_XZY = 24,
		EULER_YZX = 9,
		EULER_YXZ = 33,
		EULER_ZXY = 18,
		EULER_ZYX = 6

pose_3d_t	28	Pose in 3D space that sets the position and angles of inclination of the actuator (for example, the TCP robot):
		{
		point_3d_t pos;
		euler_3d_t rot;
		}

cst_3d_t	92	Current position of the actuator (TCP robot) and data for coordinate system transformation:
		{
		float m[4][4];
		pose_3d_t pose;
		}

32.3. Supported methods

32.3.1. Basic methods

ctype.h		
int isalnum(int)	int isalpha(int)	int isblank(int)
int iscntrl(int)	int isdigit(int)	int isgraph(int)
int islower(int)	int isprint(int)	int ispunct(int)
int isspace(int)	int isupper(int)	int isxdigit(int)
int tolower(int)	int toupper(int)	int isascii(int)
int toascii(int)		
math.h		
float acos(float)	float asin(float)	float atan(float)
float atan2(float, float)	float ceil(float)	float cos(float)
float cosh(float)	float exp(float)	float fabs(float)
float floor(float)	float fmod(float, float)	float frexp(float, int *)
float ldexp(float, int)	float log(float)	float log10(float)
float modf(float, float *)	float pow(float, float)	float round(float)
float sin(float)	float sinh(float)	float sqrt(float)
float tan(float)	float tanh(float)	

32.3.2. Special methods

Special methods do not require the connection of additional modules.

32.3.2.1. System methods

void inputs_sync()
- Synchronization of data at all inputs of the smart block. The script execution will be suspended awaiting the appearance of information from the previous blocks of the graph at all inputs of the block. Unconnected inputs are ignored.

void outputs_sync()
- Synchronization of data at all outputs of the smart block. The graph calculation will be suspended until the execution of this command, which ensures that the calculation of subsequent blocks is started only after the information appears at all outputs of the block.

void sleep_us(unsigned int val)
- Suspend script execution for the specified time (microseconds). Minimum value - 100 μs, step - 100 μs. It is recommended to use it in cycles to provide processor time to the internal threads of the scanner.

32.3.2.2. Methods for working with input ports

bool_t input_bool(char* portName, bool_t* val);
bool_t input_float(char* portName, float* val);
bool_t input_int16(char* portName, int16_t* val);
bool_t input_int32(char* portName, int32_t* val);
bool_t input_scalar(char* portName, scalar_t* val);
bool_t input_point(char* portName, point_t* val);
bool_t input_rect(char* portName, rect_t* val);
bool_t input_segment(char* portName, segment_t* val);

bool_t input_line(char* portName, line_t* val);
bool_t input_circle(char* portName, circle_t* val);
bool_t input_arc(char* portName, arc_t* val);
bool_t input_pose3d(char* portName, pose_3d_t* val);
bool_t input_cst3d(char* portName, cst_3d_t* val);
bool_t input_point3d(char* portName, point_3d_t* val);
bool_t input_euler3d(char* portName, euler_3d_t* val);
- Reading the data of the input port named "portName". If there is no port with this name, type mismatch or invalid data, the return value will be “FALSE” and “val” should not be used.

32.3.2.3. Methods for working with output ports

void output_bool(char* portName, bool_t val, bool_t valid);
void output_float(char* portName, float val, bool_t valid);
void output_int16(char* portName, int16_t val, bool_t valid);
void output_int32(char* portName, int32_t val, bool_t valid);
void output_scalar(char* portName, scalar_t val, bool_t valid);
void output_point(char* portName, point_t* val, bool_t valid);
void output_rect(char* portName, rect_t* val, bool_t valid);
void output_segment(char* portName, segment_t* val, bool_t valid);
void output_line(char* portName, line_t* val, bool_t valid);
void output_circle(char* portName, circle_t* val, bool_t valid);
void output_arc(char* portName, arc_t* val, bool_t valid);
void output_pose3d(char* portName, pose_3d_t* val, bool_t valid);
void output_cst3d(char* portName, cst_3d_t* val, bool_t valid);
void output_point3d(char* portName, point_3d_t* val, bool_t valid);
void output_euler3d(char* portName, euler_3d_t* val, bool_t valid);
- Writing data to the output port named "portName". The data validity flag is also set. If there is no port with this name or if the type does not match, the value written will be invalid.

32.3.2.4. Mathematical methods for speeding up calculations in 3D space

float m3d_distance_pt_to_pt(point_3d_t* pt1, point_3d_t* pt2);
- Calculating the Euclidean distance between two points.

void m3d_pose_fill_xyz_zyx(float x, float y, float z, float ax, float ay, float az, pose_3d_t* pose);
- Filling the fields of the "pose" structure with the passed values. The order in which Euler angles are applied is set to EULER_ZYX.

void m3d_pose_difference(pose_3d_t* pose1, pose_3d_t* pose2, pose_3d_t* result);
- Calculating the difference between two poses.

void m3d_pose_scale(pose_3d_t* pose, float factorPos, float factorRot);
- Pose scaling. Multiplying spatial coordinates by "factorPos" and angular coordinates by "factorRot".

void m3d_pose_clamp(pose_3d_t* pose, float minPos, float maxPos, float minRot, float maxRot);
- Pose clamping. Spatial coordinates that have values less than “minPos” will have the value “minPos”. Spatial coordinates that have values greater than the value “maxPos” will have the value “maxPos”. Similarly with the angular coordinates having the values “minRot” and “maxRot”.

bool_t m3d_pose_isequal(pose_3d_t* pose1, pose_3d_t* pose2, float tolerancePos, float toleranceRot);

- Checking the identity of two poses, taking into account the allowable deviations: “tolerancePos” - the maximum allowable deviation of any of the spatial coordinates, “toleranceRot” - the maximum allowable deviation of any of the angular coordinates.

void m3d_pose_interpolate_lin(pose_3d_t* pose1, pose_3d_t* pose2, pose_3d_t* result, float factor);

- Linear interpolation of the intermediate pose between "pose1" and "pose2" at the normalized distance ("factor"). If “factor” = 0, the output value will be “pose1”; if “factor” = 1, the output value will be “pose2”.

void m3d_pose_limit_movement(pose_3d_t* start, pose_3d_t* end, pose_3d_t* limits);

- Limitation of spatial and angular movement from the “start” pose to the “end” pose. Limits are set by the “limits” pose, which sets the maximum allowable movement along each of the axes. The output pose is placed at “end”.

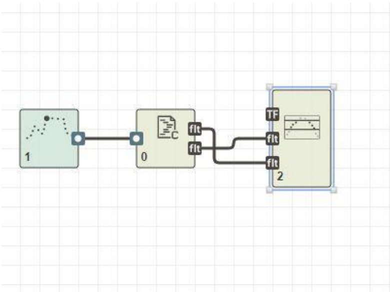
void m3d_pose_scale_movement(pose_3d_t* start, pose_3d_t* end, float factor);

- Scaling spatial and angular movement from the “start” pose to the “end” pose. The multiplier is set by the “factor” coefficient. The output pose is placed at “end”.

void m3d_euler_fill_zyx(float ax, float ay, float az, euler_3d_t* result);

- Filling a structure containing Euler angles with the passed values. The order in which angles are applied is EULER_ZYX.

32.4. Examples of scripts

<table>
<tr><th colspan="2">ROI position control</th></tr>
<tr><td></td><td>#define ROI_SIZE 30
point_t* pt;

output_float("roi_size", ROI_SIZE);

while(1)
{
inputs_sync();
pt = input_point("point");

output_float("roi_pos", pt->z + ROI_SIZE/2);
outputs_sync();
}</td></tr>
</table>

33. Annex 6. Calibration of the scanner relative to the robot

To convert the coordinates of points from 2D (a surface lying within the working range of the scanner) to 3D (the space associated with the robot), it is necessary to perform the calibration procedure and obtain data for coordinate system transformation.

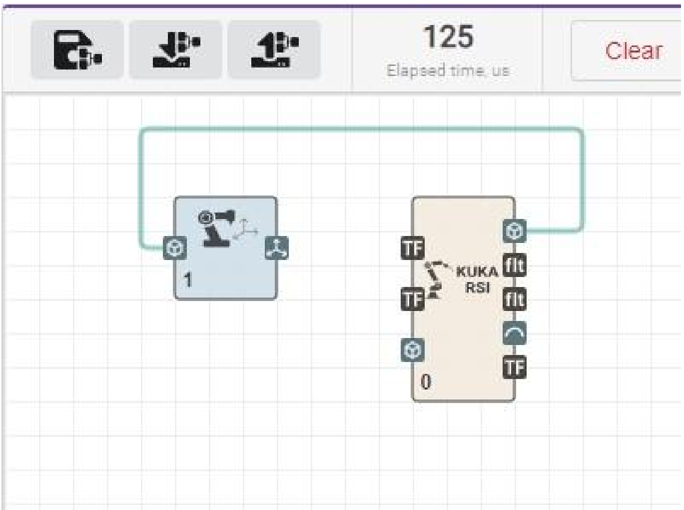
Calibration of the scanner against the robot must be performed after the TCP calibration.

The calibration procedure includes the following steps:

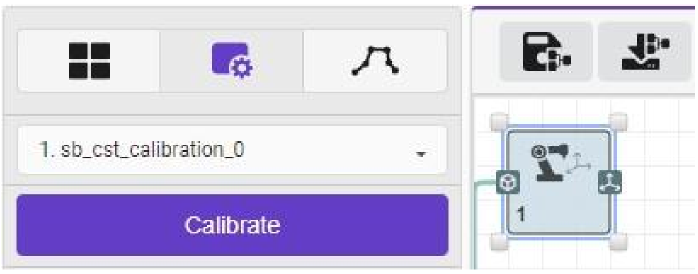
1. Data sampling.
2. Calculation of the coordinate transformation matrix.
3. Manual correction (if necessary).
4. Calibration accuracy verification: by plane and/or by 3D point. The user can choose any method. Using both methods improves reliability.

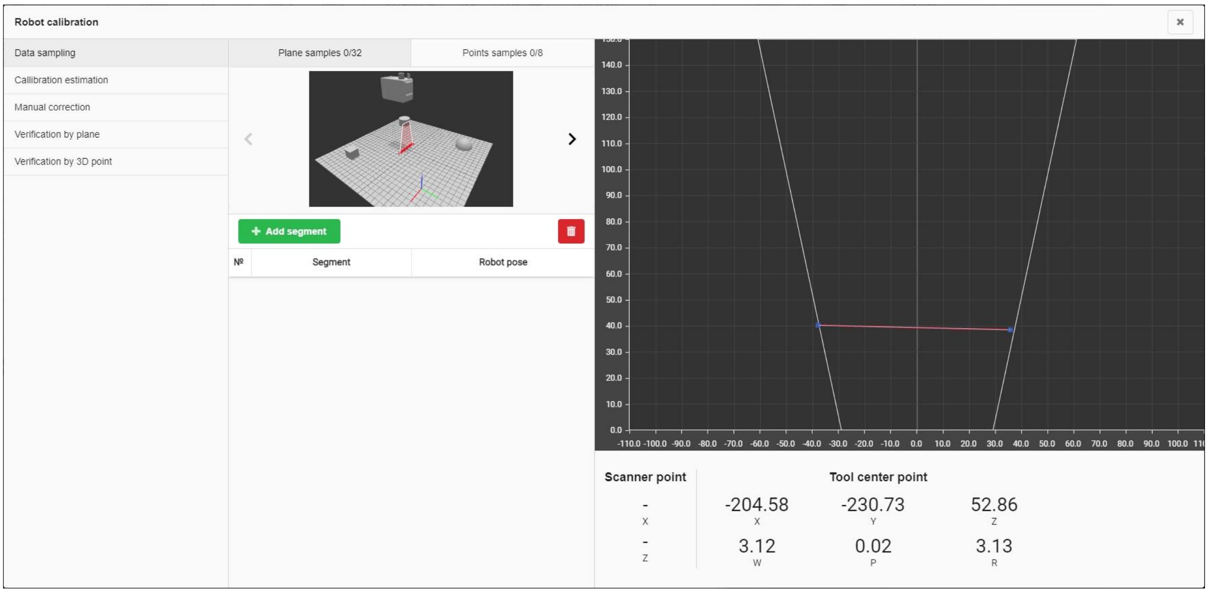
The minimum required set of blocks on the graph includes the “cst calibration” smart block and the smart block for data exchange with an external system from which TCP coordinates will be transmitted (for example, “udp”, “robot protocol P3”, etc.).

Example: graph for performing calibration with the Kuka robot using the RSI protocol.



To enter the calibration interface, click the **Calibrate** button in the parameters of the “cst calibration” smart block:





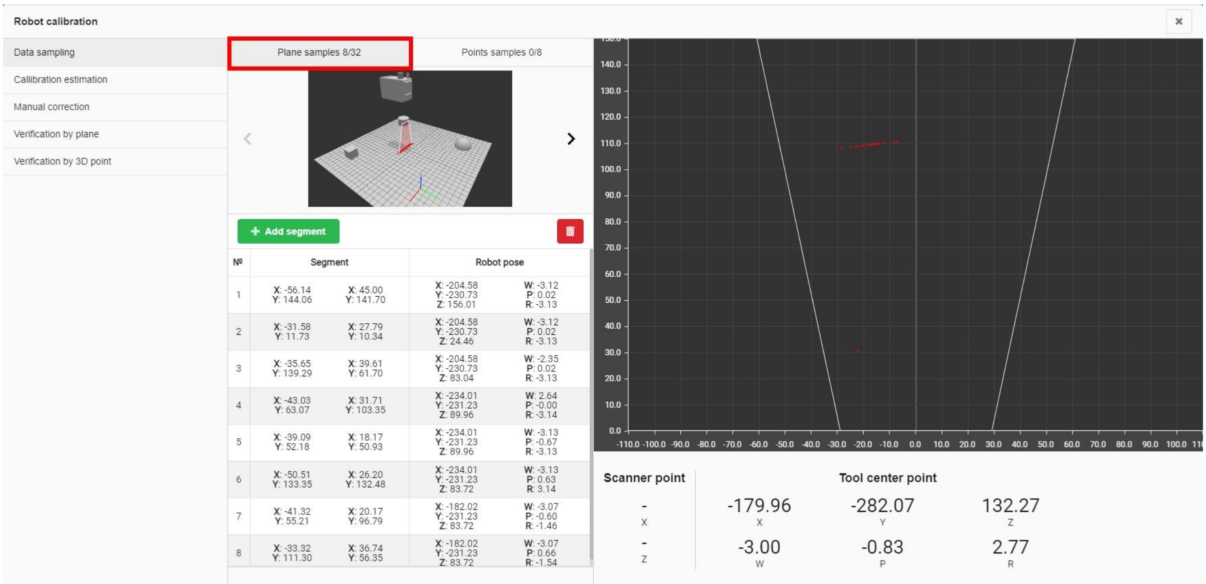
33.1. Data sampling

Data sampling requires a calibration plane, which can be any matte metal flat surface that provides a profile without gaps or rises in all calibration positions (see below). In addition, a calibration needle (used to calibrate the TCP of the robot) can optionally be used. The TCP coordinates of the robot must be transmitted to the scanner in the coordinate system that will be used in the future for specific tasks. The accuracy of the resulting calibration directly depends on the accuracy of the TCP calibration of the robot.

Data sampling by plane.

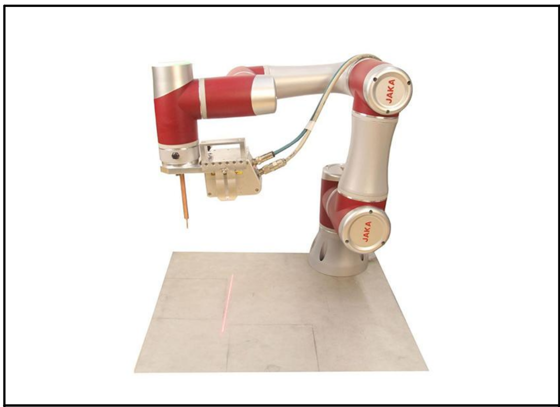
The procedure provides obtaining the basic data for the algorithm for calculating the coordinate transformation matrix, taking into account both the spatial displacement of the scanner relative to the TCP, and its inclination in all planes.

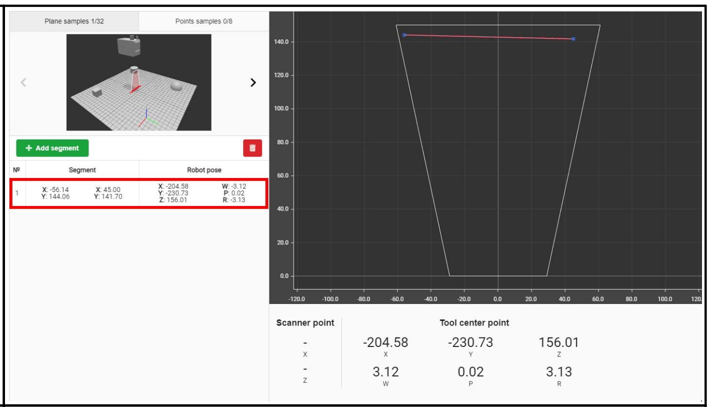
Select **Data sampling > Plane samples**:

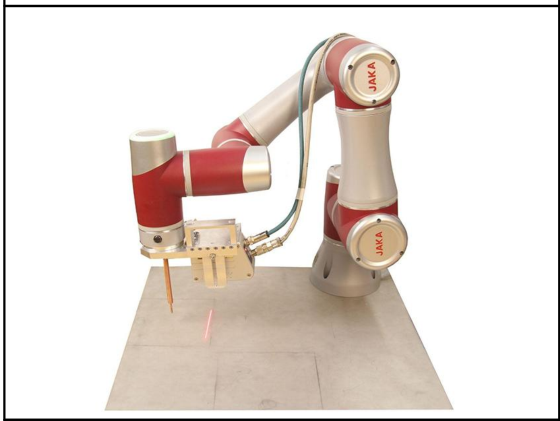


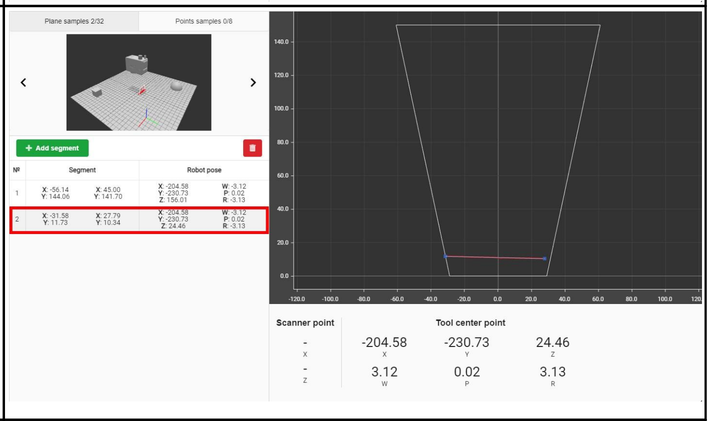
It is not allowed to change the position and inclination of the calibration plane during data acquisition. The calibration plane can be manufactured by the customer, taking into account the requirements for calibration accuracy - the calibration result cannot be more accurate than the deviation of the manufactured plane from the ideal one.

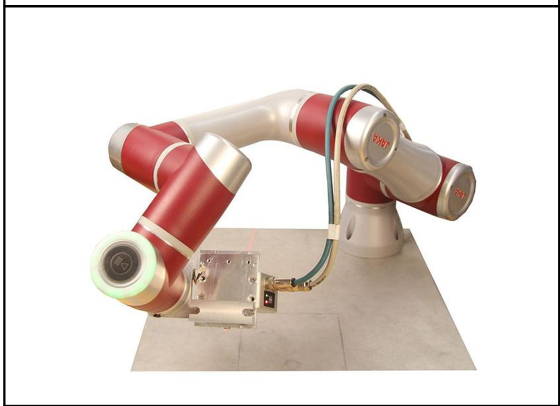
The initial data for the calibration plane are several different positions of the robot (at least eight), at which the position of the robot TCP and the coordinates of the segment are fixed in the calibration interface. The calibration plane must be within the working range of the scanner, the profile must be approximated by one segment. The next position of the robot is added to the set by clicking the **Add segment** button:

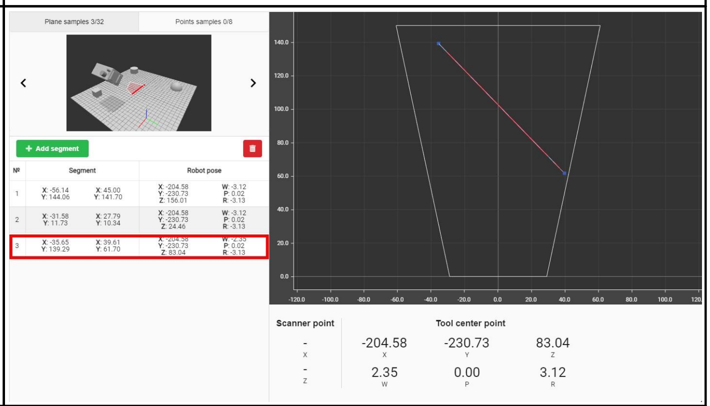


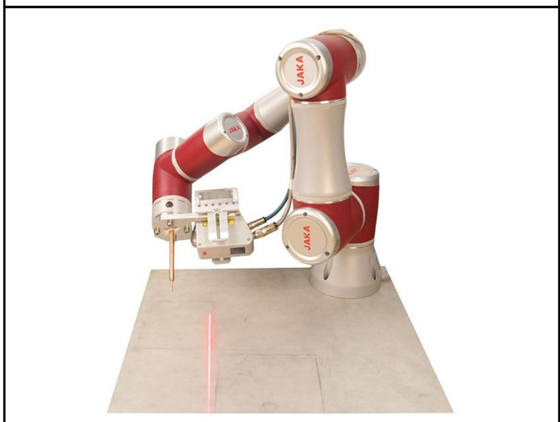


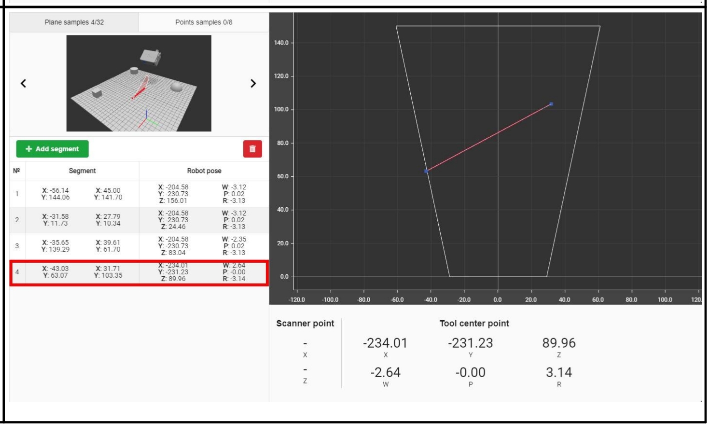


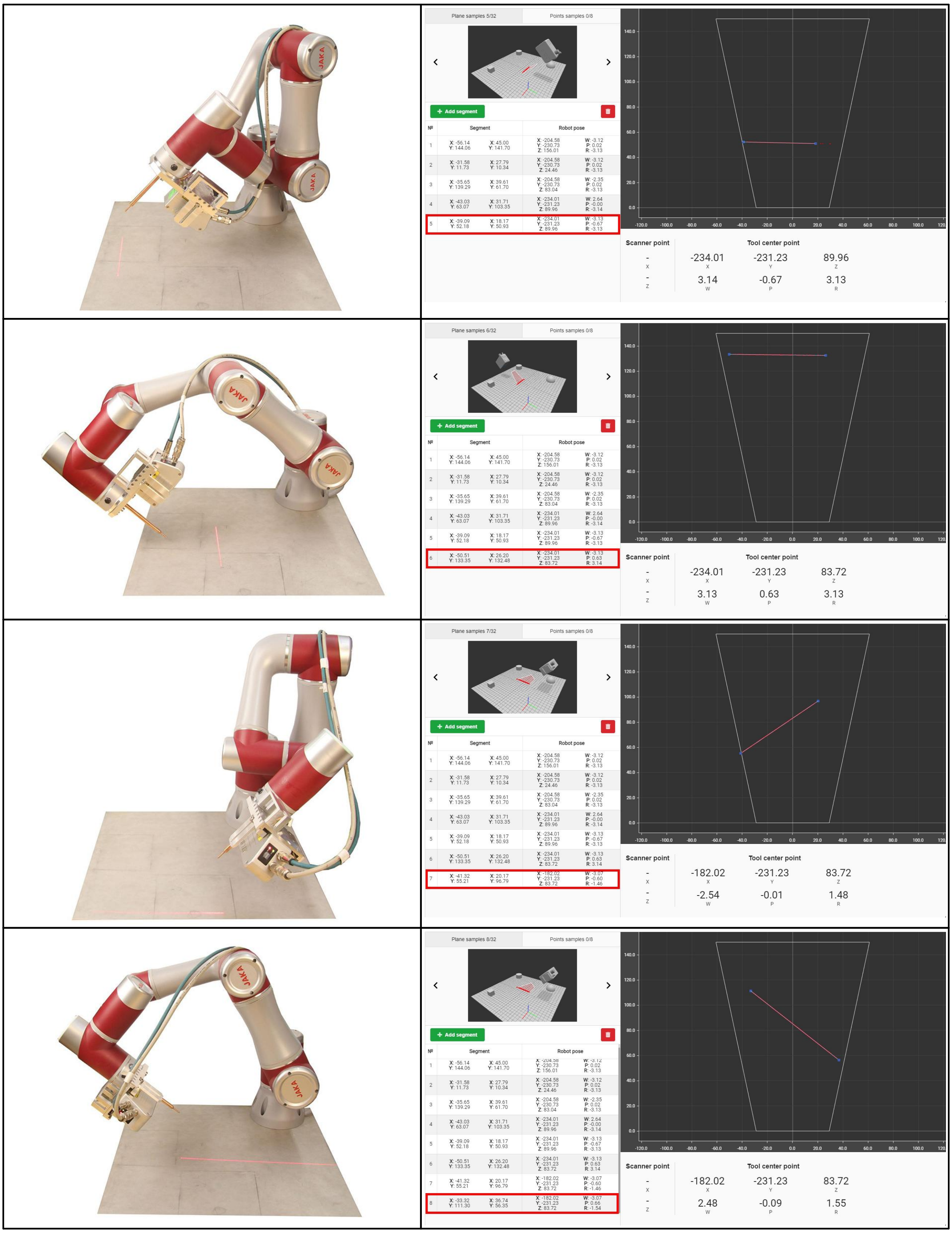








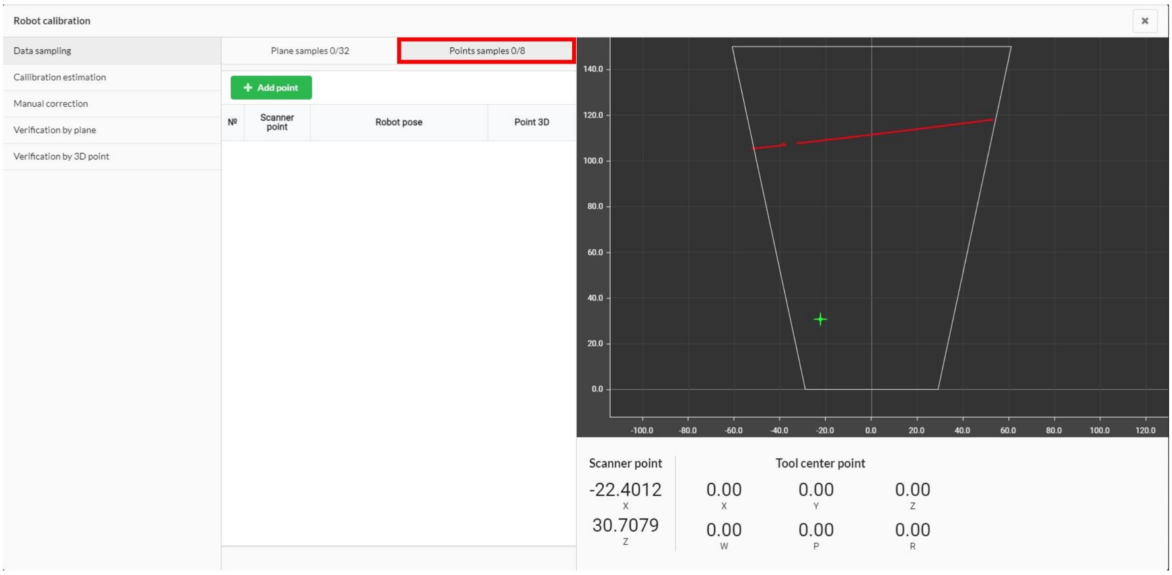




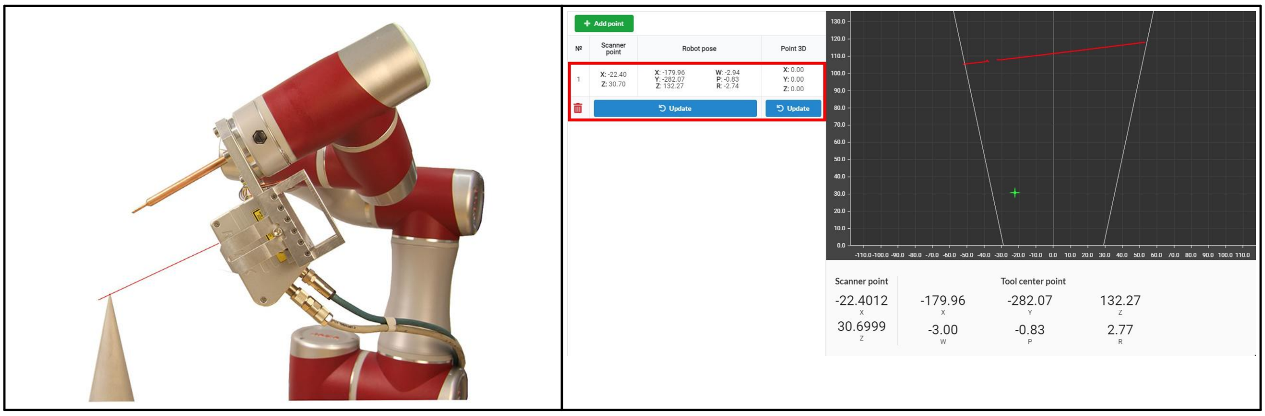
Data sampling by points.

Data sampling by points is optional and provides better convergence of the algorithm for calculating the coordinate transformation matrix (shorter calculation time). As a rule, one position of the calibration needle in space is sufficient.

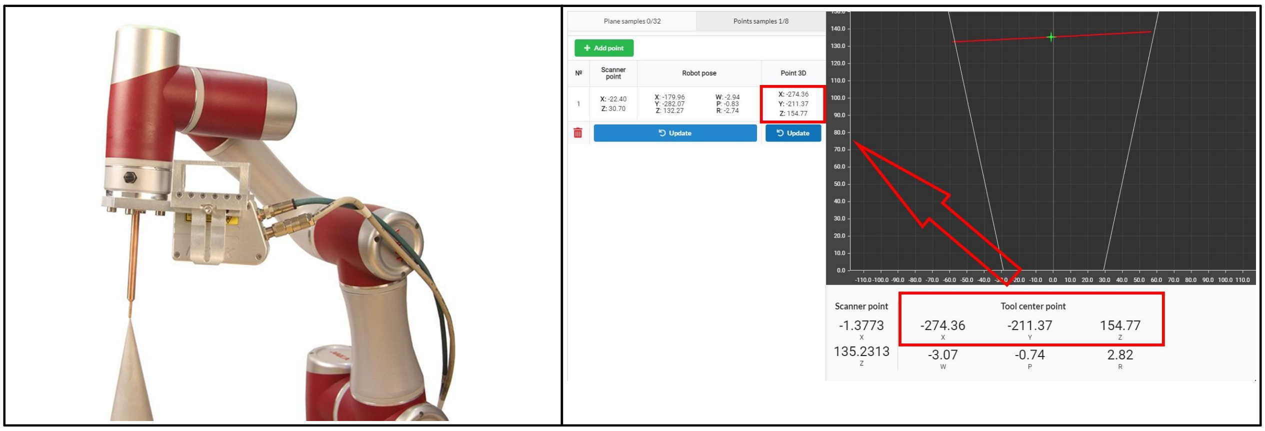
Select **Data sampling > Points samples**:



For each position of the calibration needle in space, the position of the needle point in the scanner coordinate system and the current position of the robot must be added to the initial data set (by clicking the **Add point** button):



After that, it is necessary to set the TCP of the robot at the tip of the calibration needle (similar to TCP calibration) and click the **Update** button in the **Point 3D** column, thereby matching the point in the coordinate system, taking into account the position of the robot, and the point in the robot coordinate system:

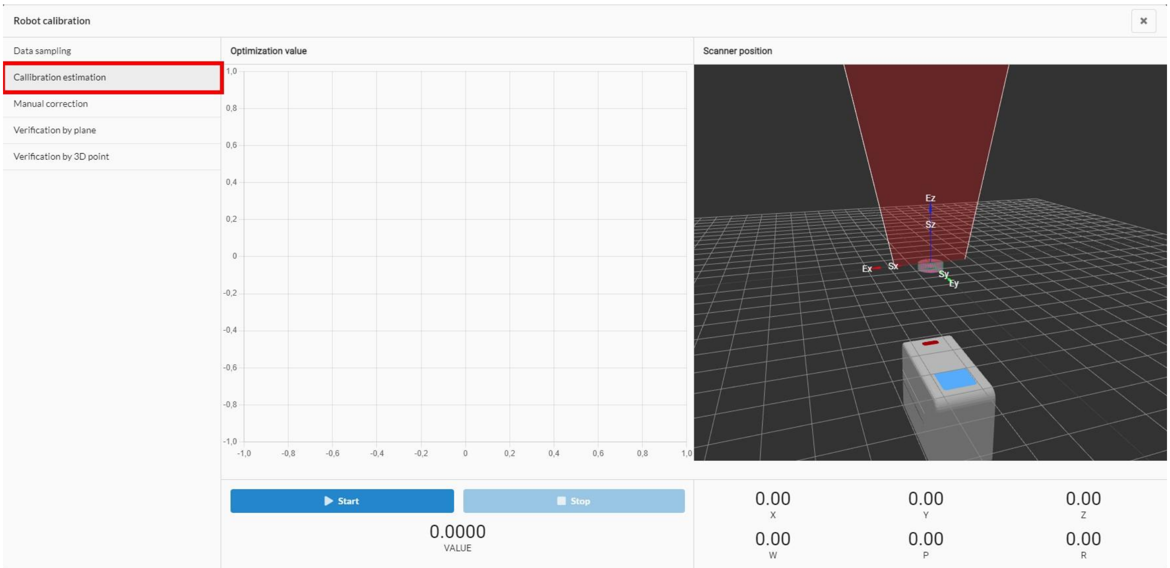


The collected initial data can be supplemented with new ones (taking into account that the calibration plane must be immobile) after performing calculations and accuracy checks. In addition, they are saved until the scanner is restarted.

33.2. Calculation of the coordinate transformation matrix

The collected data is used to calculate the position of the origin of the 2D coordinate system of the scanner relative to the TCP of the robot. The algorithm used for calculations is stochastic, so 10 runs are performed to find the optimal solution, from which the run with the best result is selected. The calculation can be stopped at any time and the best solution will be used as the result.

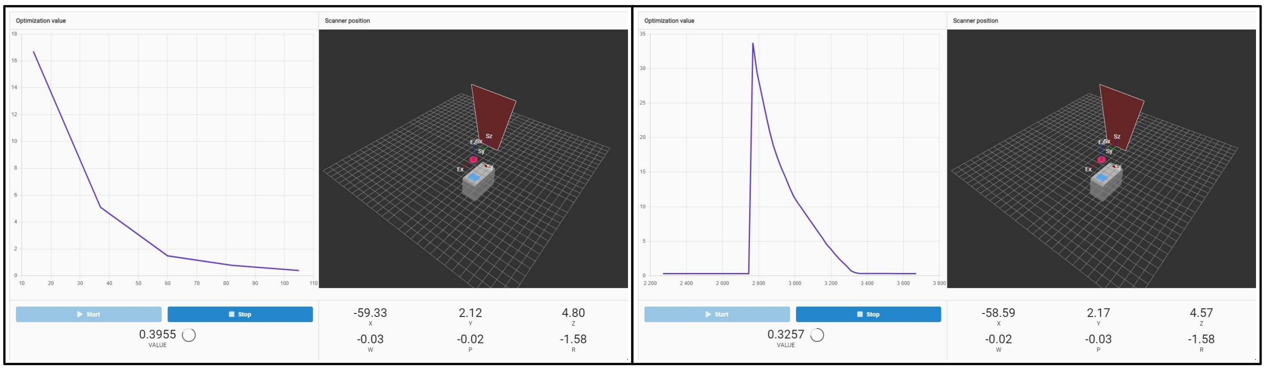
The interface is available on the **Calibration estimation** tab:



The left part of the window displays the error change graph, which characterizes the current accuracy of the coordinate system transformation relative to the source data, as well as the start and stop buttons for calculations.

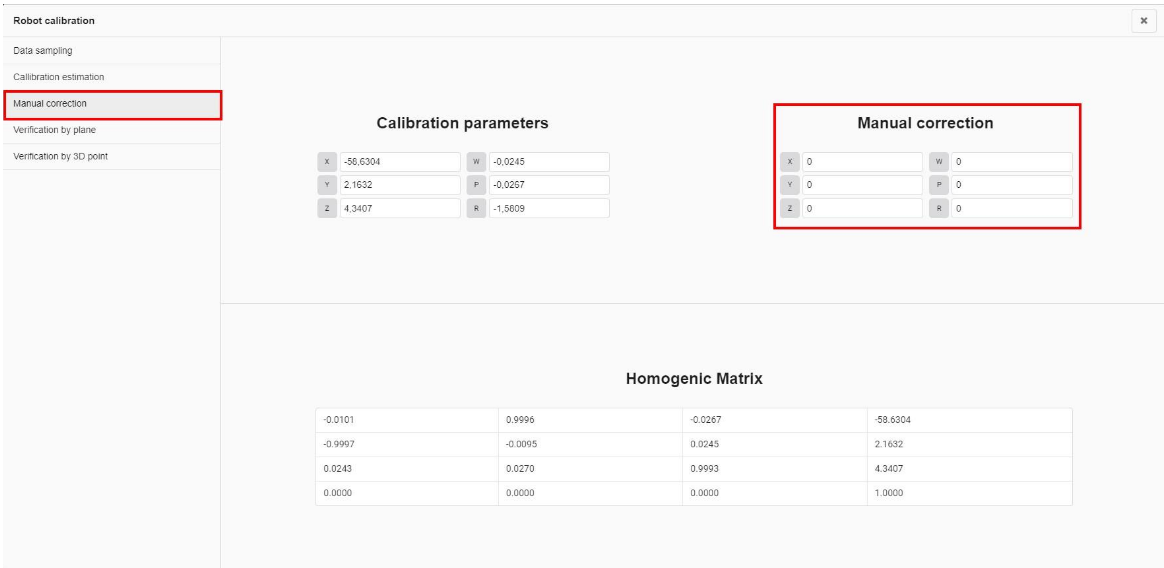
The right part displays the current position of the scanner relative to the TCP of the robot (the TCP of the robot is located at the origin of the coordinate system) and the calculated values of the spatial displacement and inclination of the scanner relative to the TCP of the robot.

During the calculation of the coordinate transformation matrix, the position of the scanner in the right window will approach the "true" position of its attachment to the robot, which allows the user to visually assess the correctness of the solution found by the algorithm:



33.3. Manual correction

If the engineer needs to correct the resulting coordinate transformation matrix, this can be done by editing the corresponding fields:



33.4. Calibration accuracy verification

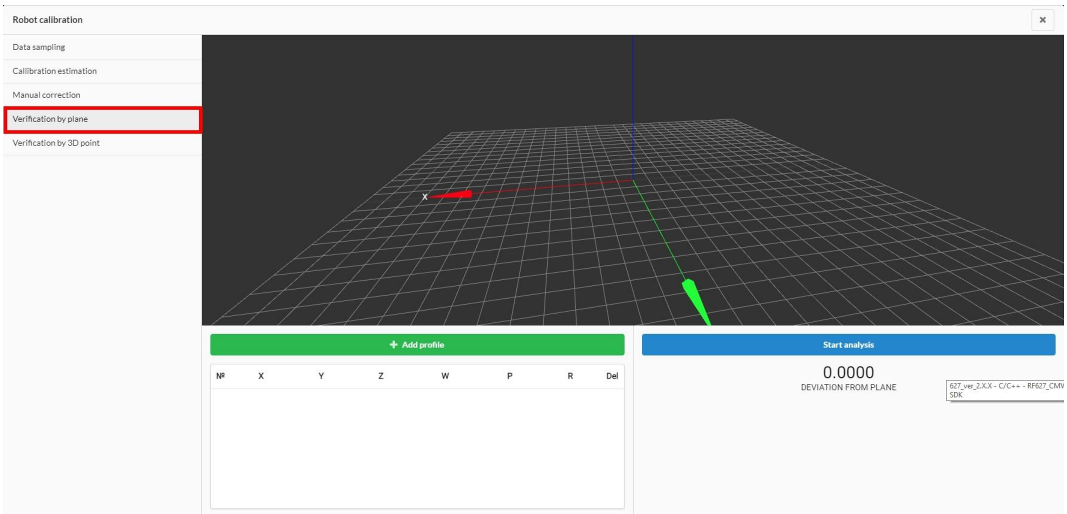
33.4.1. Verification by plane

The verification procedure includes two steps:

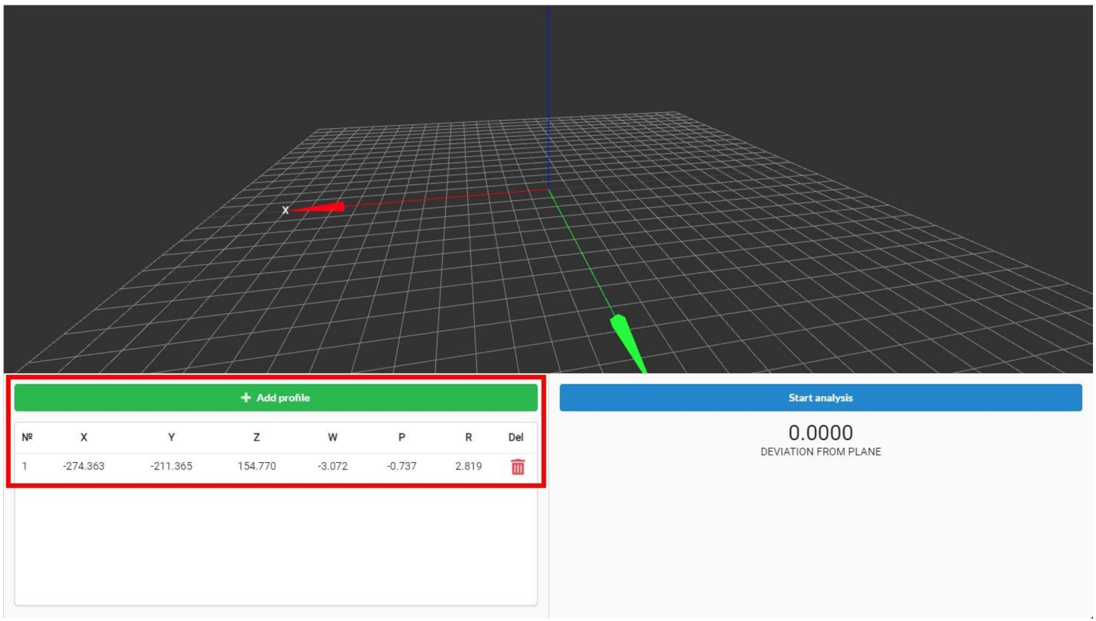
1. Collection of profiles in various locations of the robot with fixation of the coordinates and inclination of the TCP of the robot. The locations should differ in position and inclination. The greater the amplitude of changes, the more reliable the results obtained.
2. Approximation of the obtained data by a plane and calculation of the maximum deviation of a point in the profiles from the obtained plane.

Verification by plane is a quick calibration verification method because you can use the calibration plane and collect raw data quickly - it is only necessary to change the position of the robot. The results obtained by this verification method allow an assessment of the accuracy of the calibration in general.

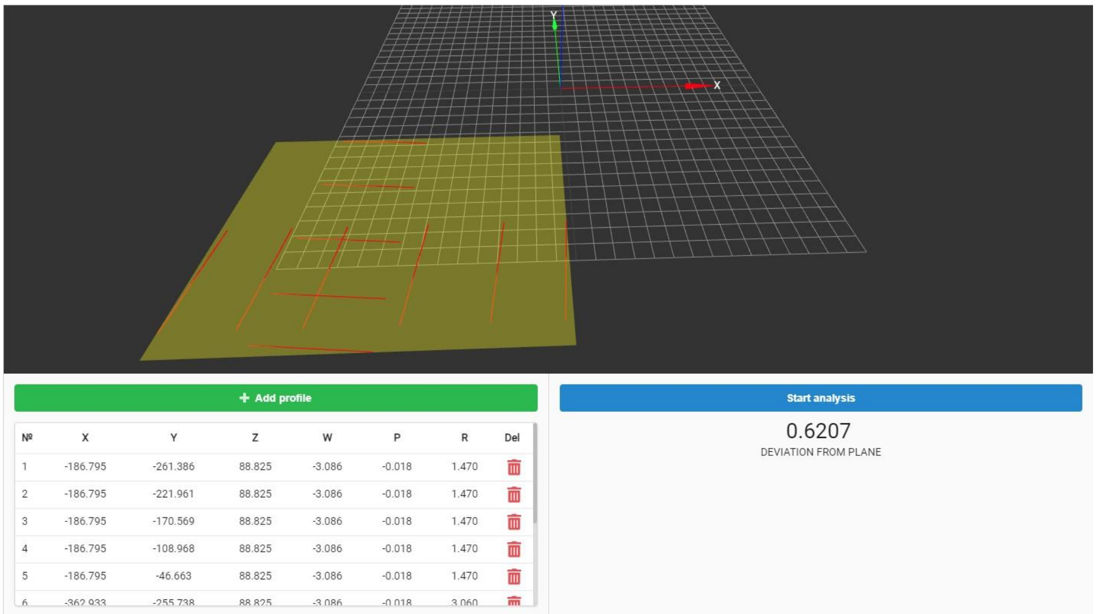
The interface is available on the **Verification by plane** tab:



To carry out the verification, it is necessary to collect several profiles (preferably 7-10) along the plane at different positions of the robot. The greater the amplitude of the change in the position of the robot, the more reliable the results will be. The position of the plane must remain unchanged during data collection. Adding the next profile is done by clicking the **Add profile** button, while both the profile and the position of the robot are fixed:



To perform calculations, click the **Start analysis** button. The entire set of profiles, taking into account the position of the robot, will be approximated by a plane:



In this case, the maximum distance from the profile points to the approximating plane is used as a measure of the coordinate transformation accuracy.

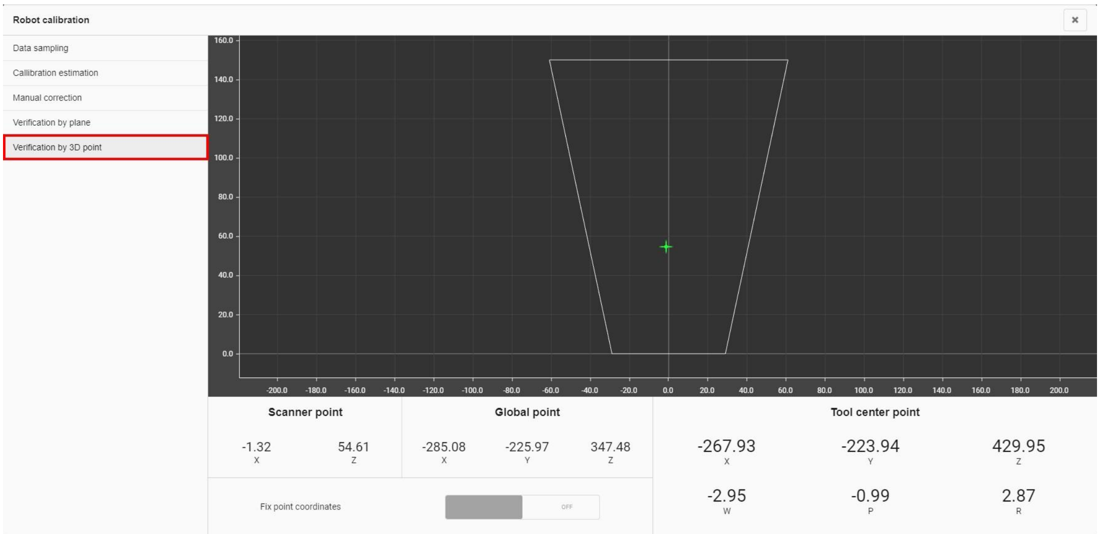
33.4.2. Verification by 3D point

The verification principle consists in comparing the calculated 3D coordinates of a point in space (corresponding to the tip of the calibration needle) and the actual coordinates of this point.

The verification includes two steps:

1. Placing the tip of the calibration needle in the working range of the scanner. Fixing the 3D coordinates of the vertex, calculated taking into account the position of the robot and the coordinate system transformation matrix, in the robot coordinate system.
2. Output of the TCP of the robot at the top of the calibration needle and verification of the obtained coordinates.

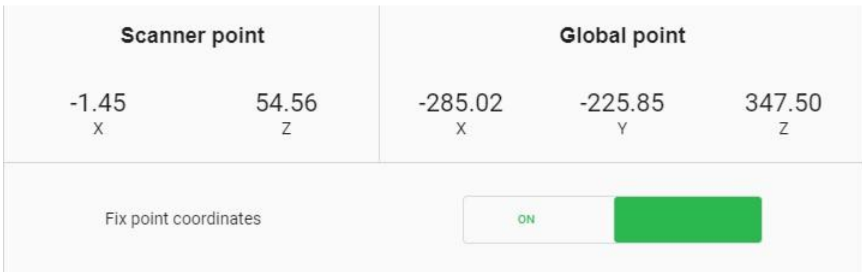
The interface is available on the **Verification by 3D point** tab:



The lower part of the window displays:

- In the **Scanner point** section - the current coordinates of the tip of the calibration needle in the 2D coordinate system of the scanner.
- In the **Global point** section - the current coordinates of the tip of the calibration needle in the 3D coordinate system of the robot, calculated on the basis of the coordinate transformation matrix.
- In the **Tool center point** section - the current coordinates of the TCP in the 3D coordinate system of the robot.

After placing the tip of the calibration needle in the working range of the scanner, a point (indicated by a green cross) will be detected, the 2D and 3D coordinates of which are displayed in the lower part of the window. It is necessary to fix the coordinates using the **Fix point coordinates** switch:



After fixing the coordinates, it is necessary to move the TCP of the robot to the tip of the calibration needle to obtain the actual coordinates in the robot coordinate system and compare the calculated coordinates (“Global point”) with the actual ones (“Tool center point”):

34. Annex 7. Overall and mounting dimensions

34.1. RF627Smart

RF627Smart-	Size		Weight, kg
25/10-8/11	Figure 7.1		0.37
65/25-20/22	Figure 7.2		0.6
75/50-30/41			
70/100-48/82			
70/150-58/122			
95/150-53/106			
82/200-60/150			
90/250-65/180			
180/250-170/278	Figure 7.3	L=326 mm	2
190/300-160/300		L=283 mm	1.9
220/300-203/330		L=374 mm	2.1
260/400-210/400		L=350 mm	2.2
325/500-268/500		L=415 mm	2.3
400/600-320/600		L=490 mm	2.4
475/700-374/700		L=558 mm	2.5
545/800-425/800		L=627 mm	2.6
615/900-480/900		L=696 mm	2.7
690/1000-535/1000		L=765 mm	2.8
620/1165-430/1010		L=554 mm	2.5

RF627Smart-	Size		Weight, kg
2/10-8/11	Figure 7.4		0.37
1/25-20/22	Figure 7.5	0.37	

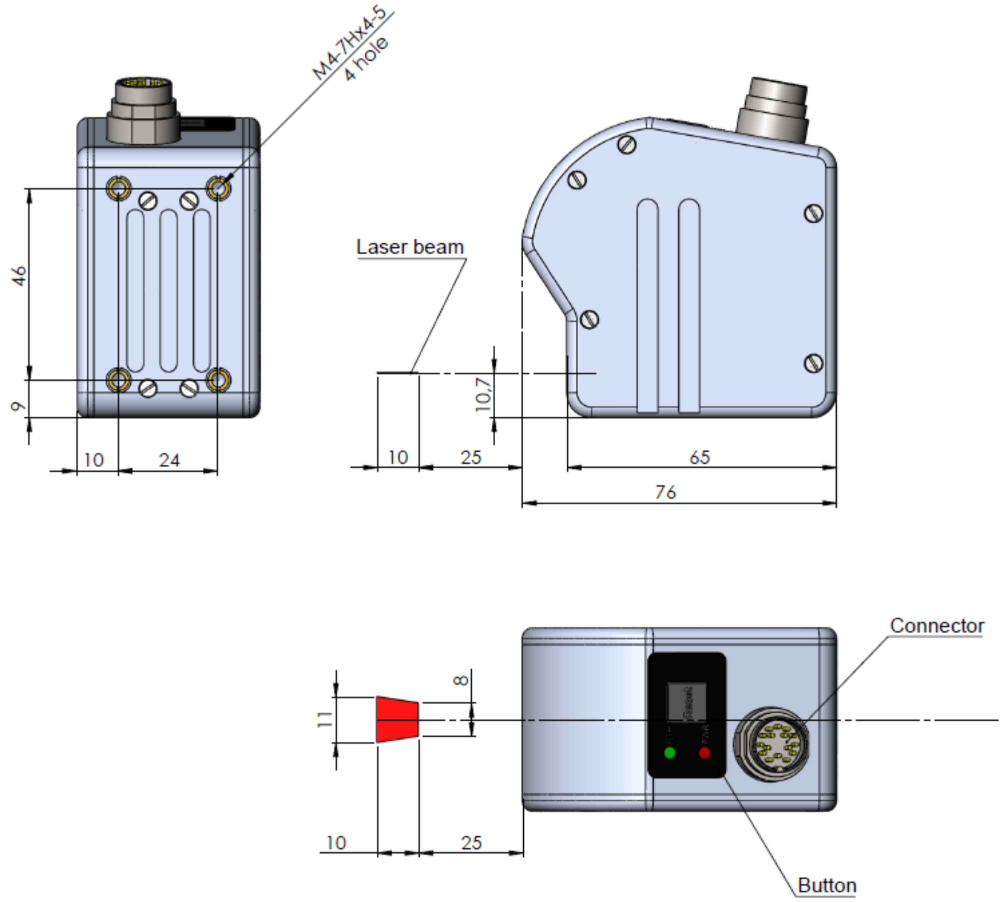


Figure 7.1

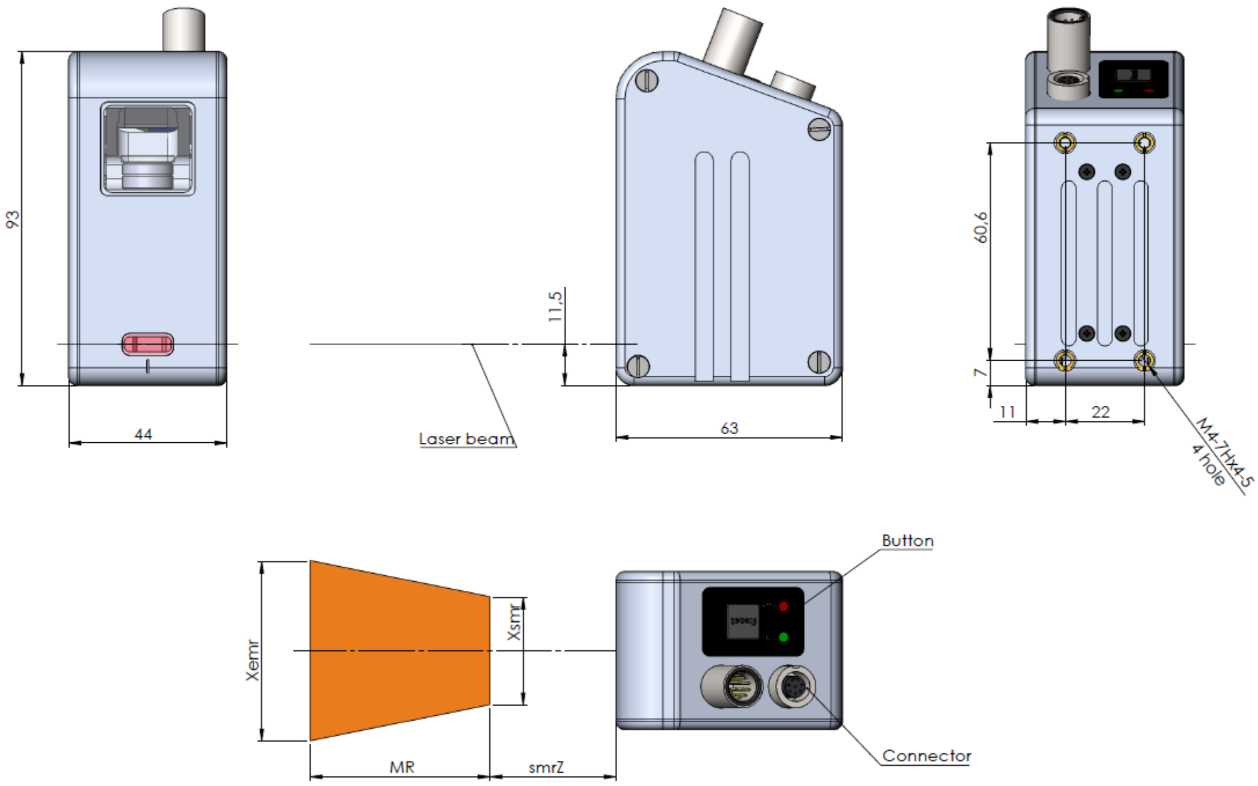


Figure 7.2

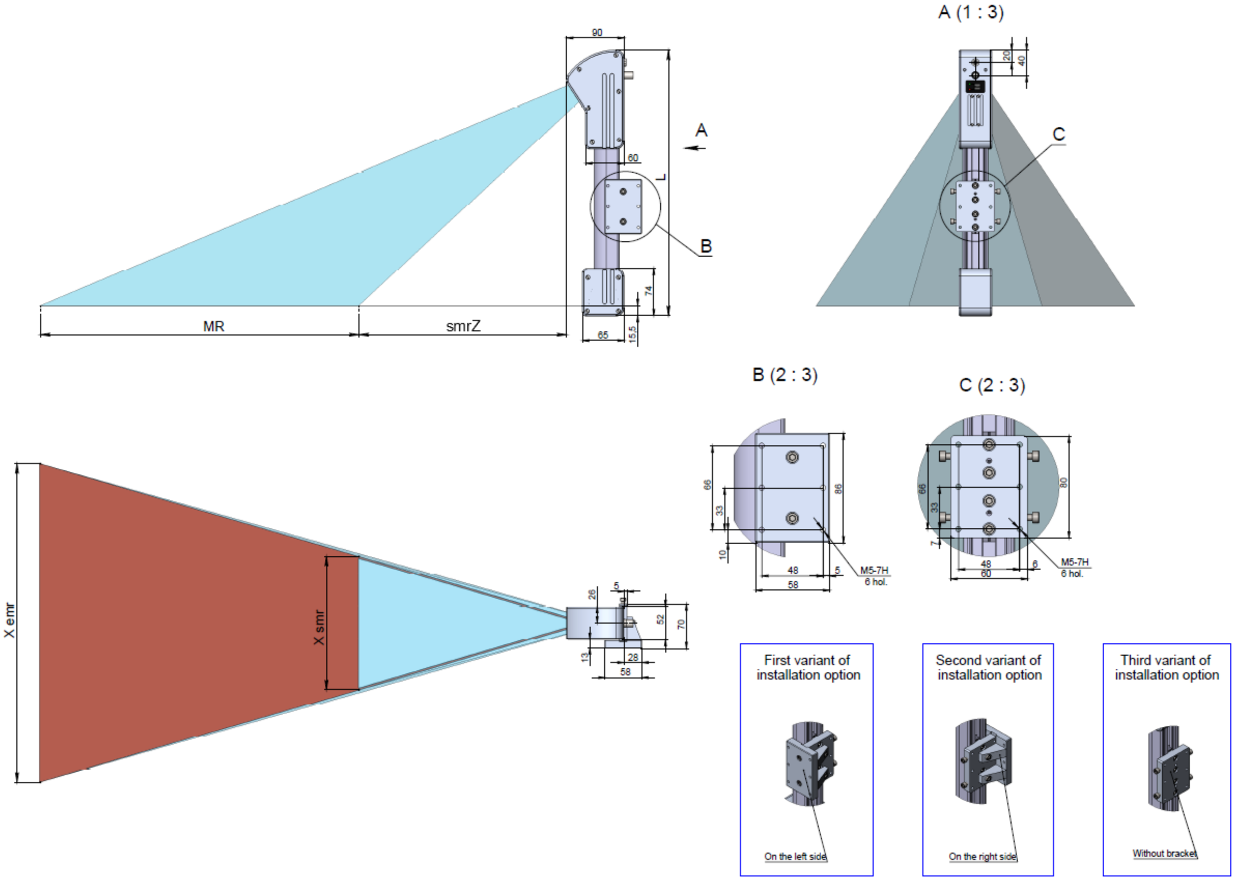


Figure 7.3

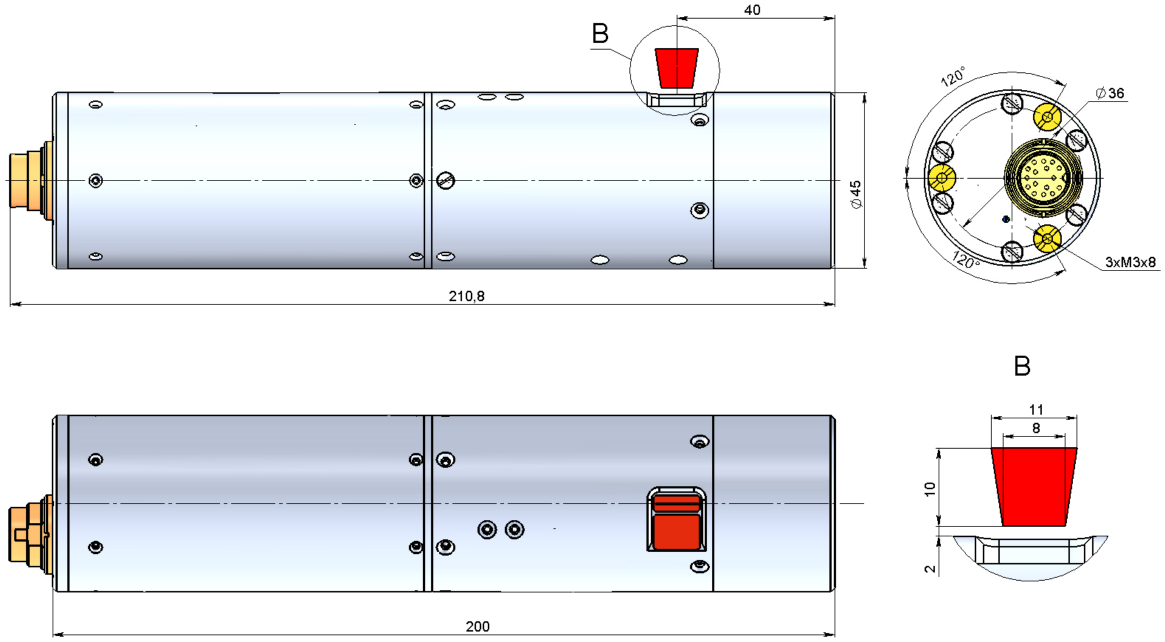


Figure 7.4

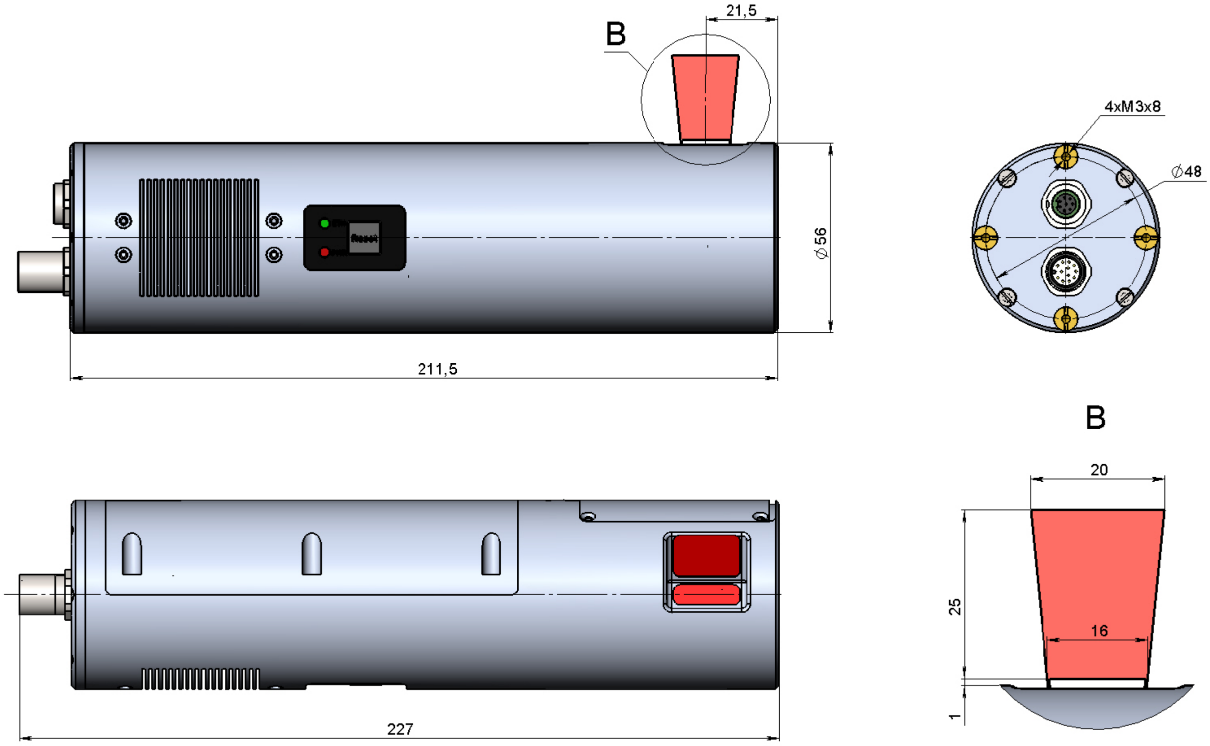


Figure 7.5

34.2. RF627BiSmart

RF627BISmart-	Size	Weight, kg
65/25-20/22	Figure 7.4	0.73
75/50-30/41		
70/100-48/82		
70/150-58/122		
95/150-53/106		
82/200-60/150		

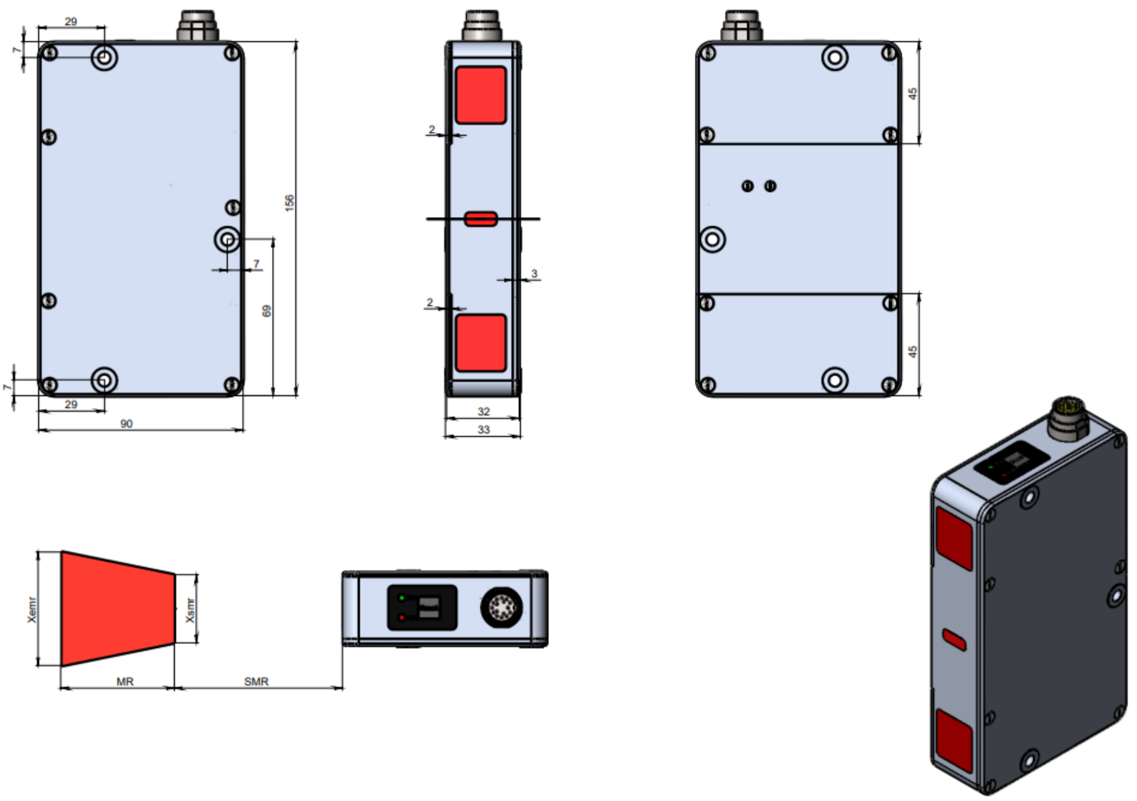


Figure 7.4

34.3. RF628Smart

RF628Smart	Size		Weight, kg
65/10-11/12	Figure 7.5	L=190 mm	1.38
60/50-36/50		L=173 mm	1.28
65/100-56/100		L=173 mm	1.13
90/150-70/140		L=185 mm	1.17
150/100-50/74		L=195 mm	1.19
150/150-64/112	Figure 7.6	L=200 mm	1.5
210/300-148/276		L=222 mm	1.57
285/400-198/376		L=260 mm	1.7
370/500-250/466		L=300 mm	1.82
450/600-300/556	Figure 7.7	L=340 mm	2.41
530/700-350/650		L=382 mm	2.47
610/800-400/744		L=420 mm	2.53
685/900-450/836		L=460 mm	2.59
765/1000-500/930		L=501 mm	2.65

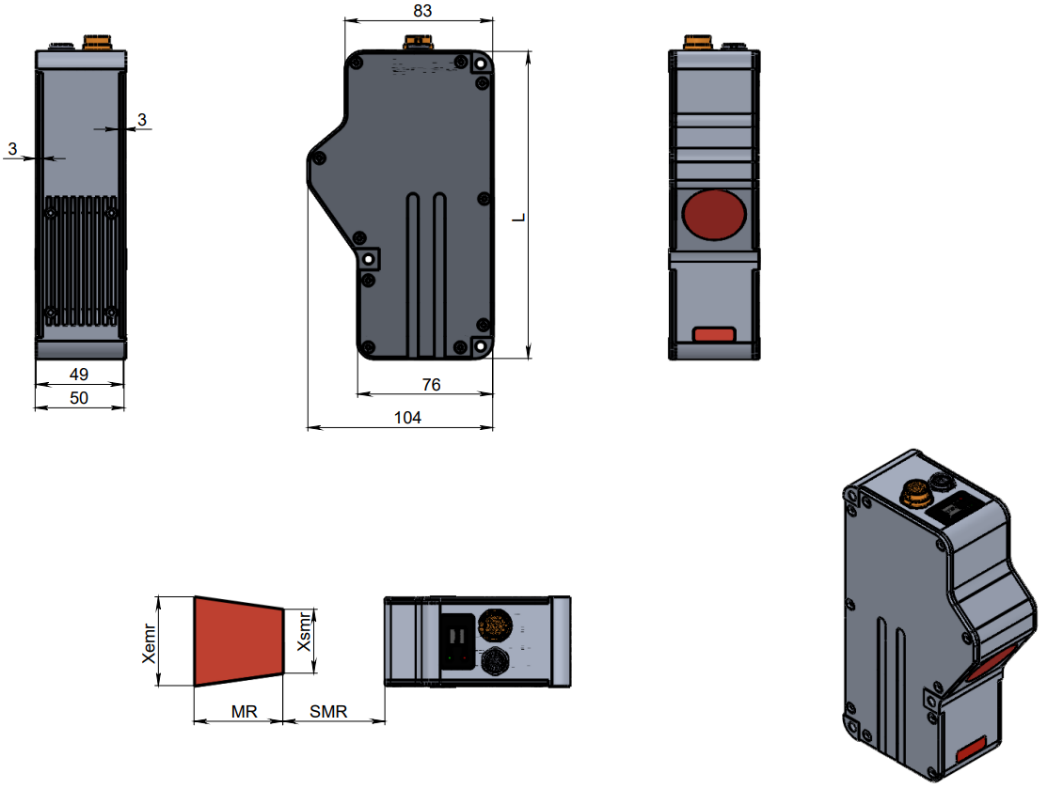


Figure 7.5

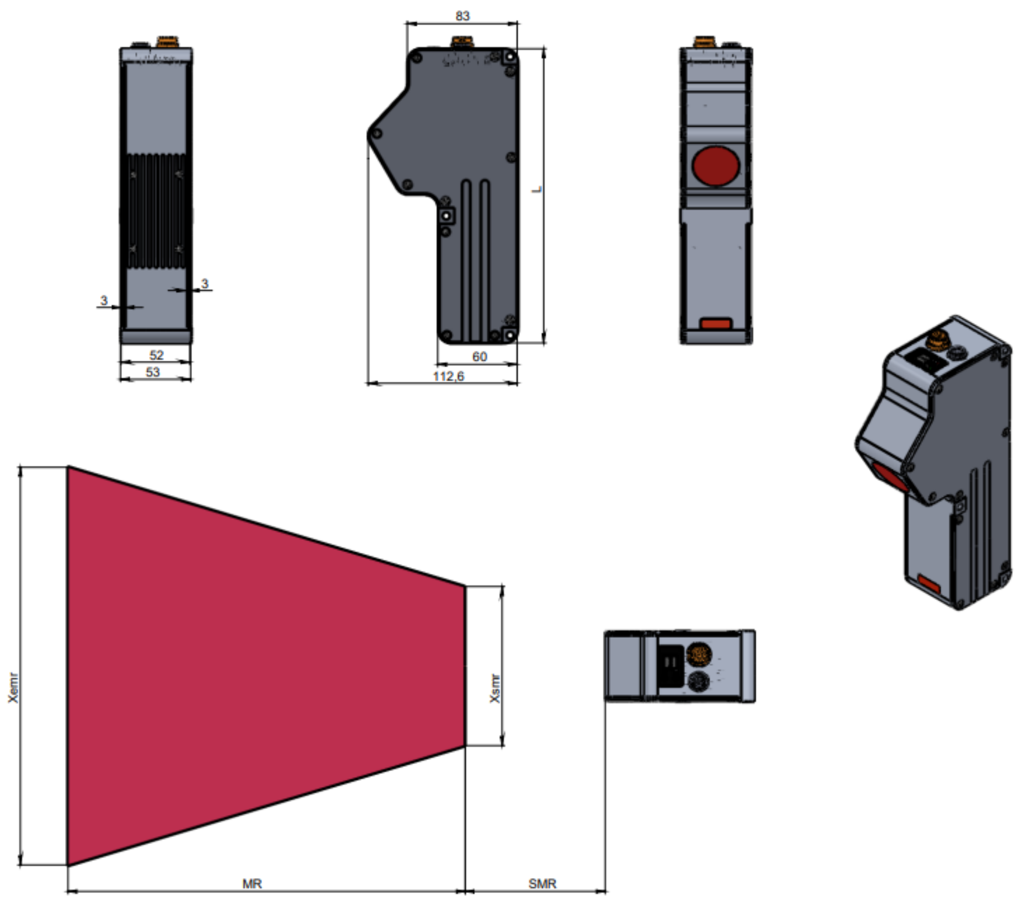


Figure 7.6

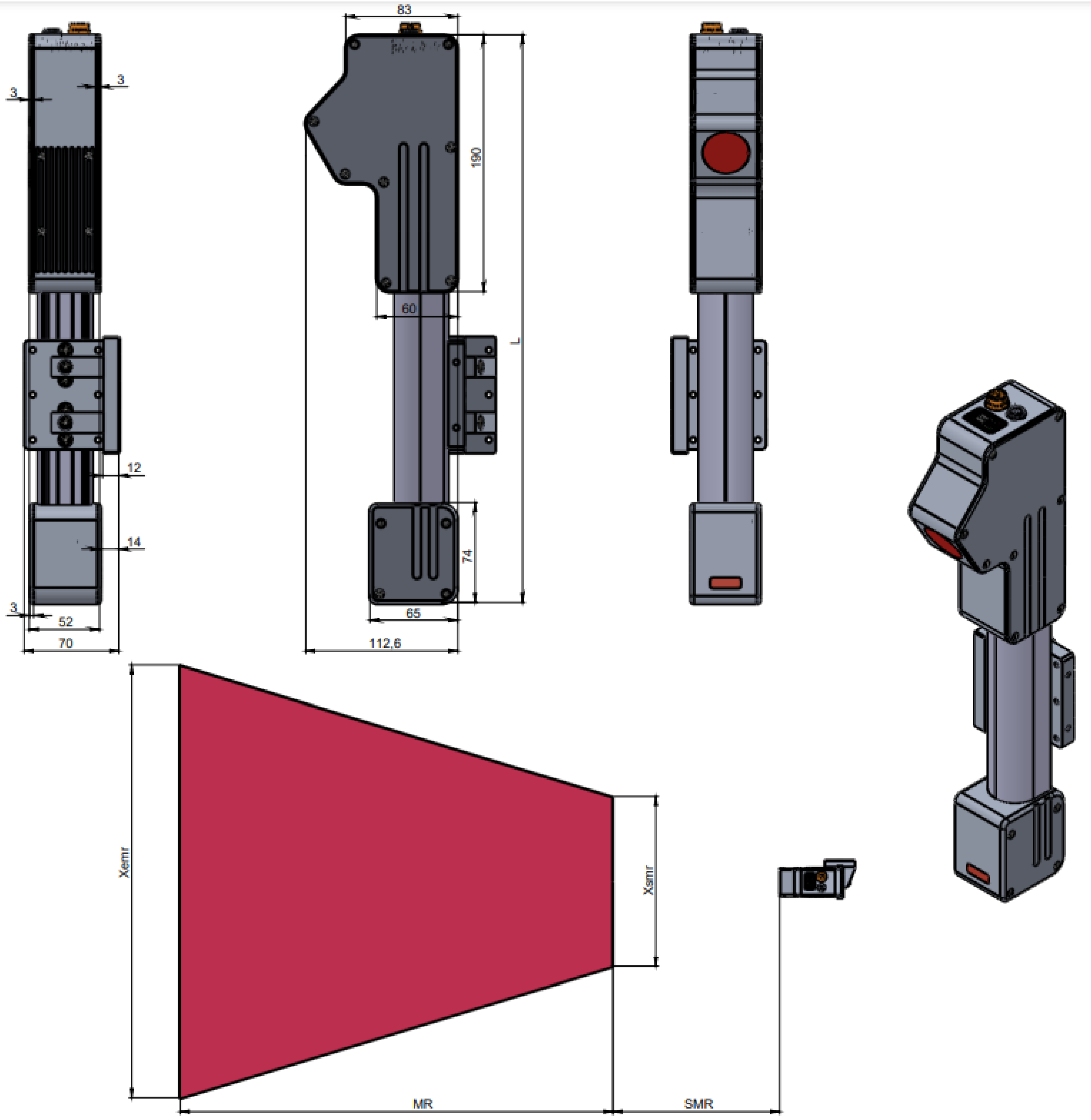


Figure 7.7

34.4. RF629Smart

RF629Smart	Size		Weight, kg
60/25-22/26	Figure 7.5	L=190 mm	1.38
60/50-36/50		L=173 mm	1.28
65/100-56/100		L=173 mm	1.13
90/150-70/140		L=185 mm	1.17
110/200-84/178		L=195 mm	1.19
95/250-100/250	Figure 7.6	L=200 mm	1.5
110/300-120/300		L=222 mm	1.57
145/400-158/400		L=260 mm	1.7
180/500-198/500		L=300 mm	1.82
230/600-236/600	Figure 7.7	L=340 mm	2.41
265/700-274/700		L=382 mm	2.47
310/800-314/800		L=420 mm	2.53
345/900-352/900		L=460 mm	2.59
385/1000-392/1000		L=501 mm	2.65

34.5. RF6292Smart

RF6292Smart	SIze		Weight, kg
70/5-24/24	Figure 7.5	L=190 mm	1.38
80/15-40/44		L=173 mm	1.28
95/25-70/81		L=173 mm	1.13
135/35-90/105		L=185 mm	1.17
170/45-110/130	Figure 7.6	L=200 mm	1.5
170/75-146/194		L=222 mm	1.57
220/90-200/256		L=260 mm	1.7
355/120-302/376	Figure 7.7	L=340 mm	2.41
455/170-400/500		L=420 mm	2.53
550/225-500/634		L=501 mm	2.65

34.6. Scanners with additional options

34.6.1. Example of a scanner with replaceable protective windows, EW option

The scanners with replaceable protective windows:

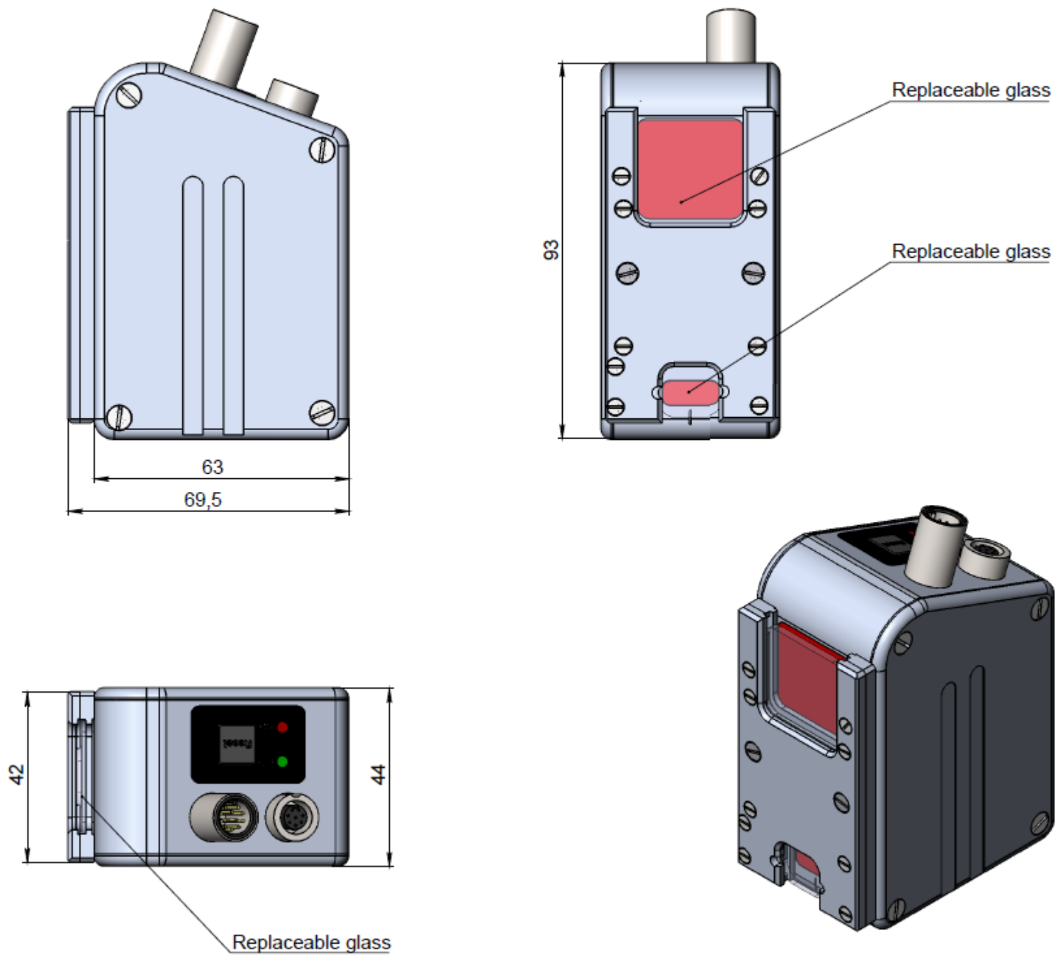


Figure 7.8

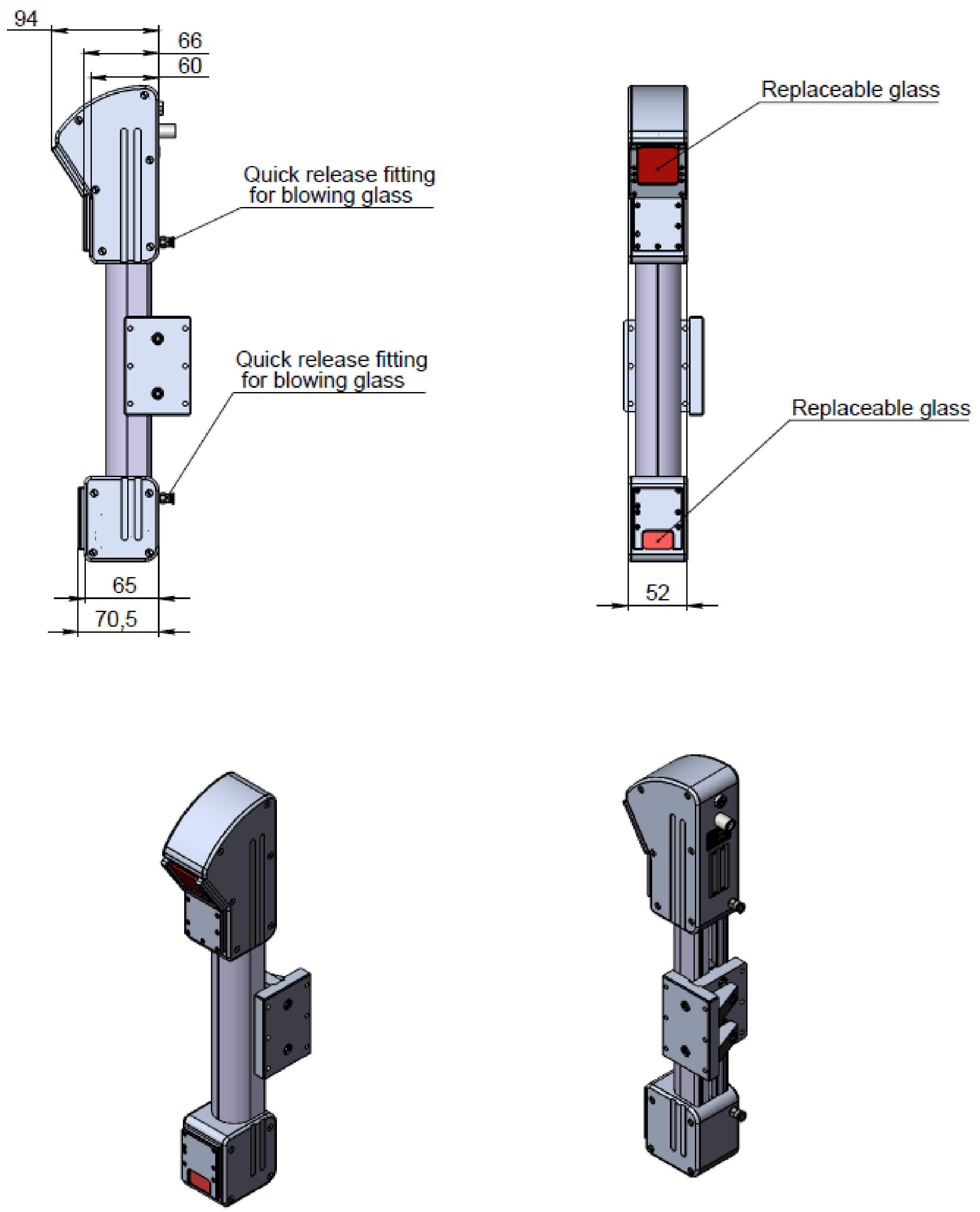

94
66
60
Quick release fitting
for blowing glass
Quick release fitting
for blowing glass
65
70,5
Replaceable glass
Replaceable glass
52

Figure 7.9

34.6.2. Example of a scanner with air cooling, AK-EW-AC option

The scanner with replaceable windows, air protection of windows and air cooling:

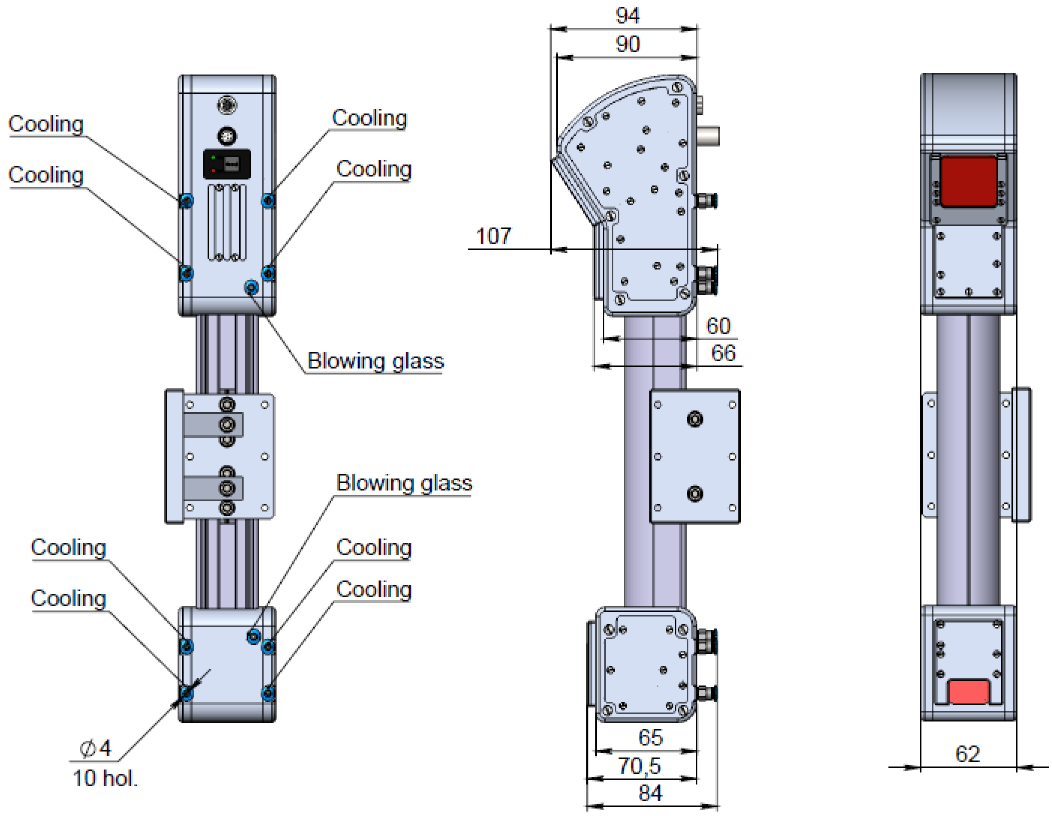


Figure 7.10

34.6.3. Example of a scanner with water cooling, AK-EW-AC option

The scanner with replaceable windows, air protection of windows and water cooling:

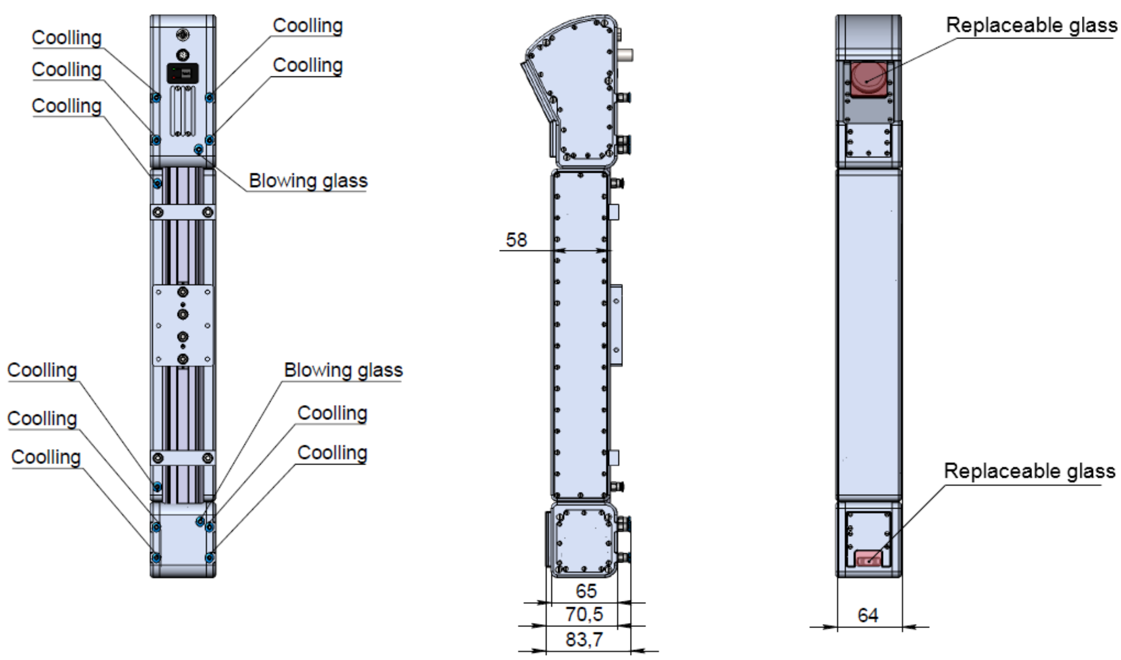
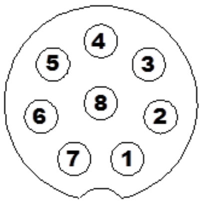
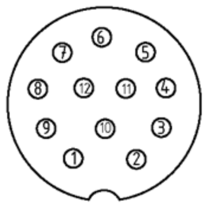


Figure 7.11

35. Annex 8. Connectors and cables

35.1. Pin assignment of connectors

The scanner comes with one or two connectors.
Scanner with two connectors:
1. Ethernet connector.
2. Multi-connector.
View from the side of connector contacts is shown below:

Connector 1 **Binder 712 Series, #09-0428-30-08**	**Connector 2** **SACC-DSI-M12MS-12CON-M12**
	

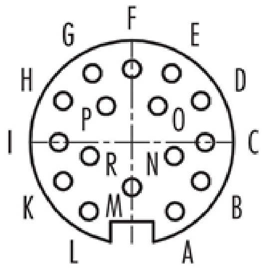
Designation of contacts is given in the tables below.
Connector 1:

Pin number	Assignment, 100baseTX	Assignment, 1000baseT
1		D4+
2		D3-
3		D3+
4	RX-	D2-
5	RX+	D2+
6	TX-	D1-
7	TX+	D1+
8		D4-

Connector 2:

Pin number	Assignment	Note
1	OUT1-	RS422
2	IN3-	RS422
3	IN3+	RS422
4	IN2-	RS422
5	IN2+	RS422
6	NEXT_LAS_OFF	Hardware laser on/off input. Hardware on/off means enabling/disabling laser radiation regardless of scanner settings.
7	IN1+	RS422
8	IN1-	RS422
9	OUT1+	RS422
10	VIN	+9...30V, 1A max
11	GND	Grounding
12	0V	0V power supply («-»)

Scanner with one connector.
View from the side of connector contacts is shown below:

Connector Binder 423 Series, #99-5456-15-16


Designation of contacts is given in the table below:

Pin number	Assignment	Note
A	D3- (RX-)	ETHERNET 1000baseT (100baseTX)
B	D3+ (RX+)	ETHERNET 1000baseT (100baseTX)
C	IN0-	RS422
D	IN0+	RS422
E	IN2-	RS422
F	D2- (TX-)	ETHERNET 1000baseT (100baseTX)
G	IN2-	RS422
H	IN1+	RS422
I	IN1-	RS422
K	D4-	ETHERNET 1000baseT (100baseTX)
L	D4+	ETHERNET 1000baseT (100baseTX)
M	VIN	+9…30V, 1A max
N	0V	0V power supply («-»)
O	D2+ (TX+)	ETHERNET 1000baseT (100baseTX)
P	D1- (TX-)	ETHERNET 1000baseT (100baseTX)
R	D1+ (TX+)	ETHERNET 1000baseT (100baseTX)

35.2. Cables

Scanner with two connectors.
Cable 1:

Pin number RJ45	Assignment 100baseTX	Assignment 1000baseT	Wire color
1	TX+	D1+	White/orange
2	TX-	D1-	Orange
3	RX+	D2+	White/green
4		D3+	Blue
5		D3-	White/blue
6	RX-	D2-	Green
7		D4+	White/brown
8		D4-	Brown

Cable 2 (free leads):

Wire color	Assignment
Black	OUT1-
Gray/pink	IN3-
Red/blue	IN3+
Gray	IN2-

Wire color	Assignment
Pink	IN2+
White	NEXT_LAS_OFF
Green	IN1+
Yellow	IN1-
Violet	OUT1+
Red	VIN
Blue	GND
Brown	0V

Scanner with one connector.
Cable (free leads):

Wire color	Assignment
Red	VIN
Blue	0V
Yellow/White	IN1+
Yellow/Brown	IN1-
Green/White	IN2+
Green/Brown	IN2-
Red/Blue	IN3+
Pink/Gray	IN3-
Green	D1+ (TX+)
Yellow	D1- (TX-)
Brown	D2+ (RX+)
White	D2- (RX-)
Violet	D3+
Black	D3-
Gray	D4+
Pink	D4-

RJ45 color code (optional, for the version with the RJ45 connector):

Pin number RJ45	Assignment 100baseTX	Assignment 1000baseT	Wire color
1	TX-	D1-	Yellow
2	TX+	D1+	Green
3	RX+	D2+	Brown
4		D3+	Violet
5		D3-	Black
6	RX-	D2+	White
7		D4+	Gray
8		D4-	Pink

36. Annex 9. ProfiTalk protocol

To make software development easier for users, the scanners support the proprietary ProfiTalk protocol, which provides the following benefits:

- Simple software implementation and support without using the SDK.
- Fast serialization and deserialization in one pass using the MessagePack binary data format.
- Fast integration and convenient debugging when using serialization and deserialization in the JSON format (in the next version);
- Abstraction and the ability to control the composition of transmitted data through the use of the DOM representation.
- Client-server architecture with the ability to connect multiple clients to the same scanner data sources (for example, several clients can simultaneously receive profiles and/or video frames, request or send scanner parameters).

The protocol is divided into the following modules:

1. “search_service” module - search for scanners on the network.
2. “commands_service” module - transmission of control commands.
3. “video_stream” module - video streaming.
4. “profiles_stream” module - profile streaming.
5. “smart” module - under development.

36.1. Specification

Technical specification of the protocol (version 1.0):

Module	Transport protocol	Default port number	Maximum number of simultaneous connections	Messaging type	Message format
“search_service”	UDP	51000	-	synchronous	MessagePack
“commands_service”	TCP	51001	4	synchronous	MessagePack
“profiles_stream”	TCP	51002	4	stream	MessagePack
“video_stream”	TCP	51003	4	stream	MessagePack
“smart”	TCP	-	-	-	-

36.2. Scanner support

Scanner model	Firmware version	Protocol specification version
RF627Smart	2.13.0	1.0
RF627Bi	3.3.0	1.0
RF628	-	-
RF629, RF6292	2.10.0	1.0

36.3. Message structure at the transport layer

For all modules (**except for the “search_service” module**), the following message structure at the transport layer is adopted:

Header - message body size (4 bytes, uint32_t, network byte order - from high to low)				Message body - serialized to MessagePack or JSON object		
B3	B2	B1	B0	A0	…	An

The header, which conveys the size of the message, is necessary for the correct division of the byte stream into messages and their deserialization.

The object is assembled after receiving the entire message: the length and the body of the message.

Since TCP is used and the protocol messages are quite simple, all requests must be executed synchronously. You cannot send the next request before receiving a response to the already sent request (in protocol specification version 1.0).

Multiple client connections are allowed (using multiple TCP connections). This is especially convenient when it is necessary to simultaneously work with requests that have significantly different execution and data transfer times, for example, receiving frames, reading/writing parameters, obtaining profiles, etc.

The “search_service” module does not require such a structure, because the UDP protocol is used and the messages are quite short.

36.4. Search for scanners on the network

In case it is necessary to find scanners available on the network (their IP addresses are unknown or for other reasons), you should use the “search_service” module, which is based on the UDP protocol. To do this, the serialized request must be sent to the port specified in the protocol specification (if it has not been changed in the scanner settings) and the broadcast address (255.255.255.255, XXX.XXX.XXX.255 is also recommended).

The request to search for all available scanners:

```
{	“request”: “SEARCH” }
```

The request to search with parameters:

```
{
	“request”: “SEARCH”,
	“serial”: 12345
}
{
	“request”: “SEARCH”,
	“name”: “2d laser scanner”
}
```

The response from the scanner is sent to the same address and port from which the request was received. If sent without a request (not provided in the current version), then 255.255.255.255 and XXX.XXX.XXX.255 are used as the address, the destination port is used from the scanner parameters (the same as the default one for the “search_service” module):

```
{
	“name”: “2d laser scanner”,
	“product_code”: “627”,
	“device_serial”: 12345,
	“hardware_id”: 4394025256,
	“firmware_version”: [134, 441, 234],
	“hardware_version”: 202012,
	“smr”: 123.35,
	“mr”: 54.23,
	“xsmr”: 8.432,
	“xemr”: 9.342,
	“ip4_addr”: 353265625464,
	“ip4_mask”: 96459394394394343,
	“ip4_gateway”: 6583434623623535,
```

"profitalk_commands_port": 51001,
"profitalk_video_port": 51003,
"profitalk_profiles_port": 51002,
}

The command to reset network settings to factory settings. It can be used if the TCP connection is not established, but UDP broadcast packets are received (must include the scanner serial number for identification):

{
"request": "RESET_NETWORK_PARAMETERS",
"serial": 12345
}

36.5. Transmission of control commands

The TCP protocol is used to transmit messages. The message consists of a command field and a payload field (this field can be omitted if not required). The payload field, if present, must contain the object. The command field specifies the handler to which the payload will be passed.

An example of changing the parameter value:

{
"request": "WRITE_PARAMETERS",
"payload":{
"fact_general_serial": 545746464,
"fact_general_workTime": 0,
"fact_general_authStatus": 1
}
}

Answer:

{
"result": "RF_GENERAL_FAULT",
"payload":{
"fact_general_serial": "RF_OK",
"fact_general_workTime": "RF_OK",
"fact_general_authStatus": "RF_WRITE_IMPOSSIBLE"
}
}

In this example, an error occurred when executing the command - the "fact_general_authStatus" parameter cannot be written, it has a "read-only" attribute, so the overall result is an error, and the payload contains the results of setting each of the parameters.

Example without error:

{
"request": "WRITE_PARAMETERS",
"payload":{
"user_sensor_framerate": 100
}
}

Answer:

{
"result": "RF_OK",
"payload":{

“user_sensor_framerate”: “RF_OK”
}
}

36.6. Commands

36.6.1. READ_PARAMETERS_DESCRIPTION

- Function: Obtaining general information about all device parameters. A formalized description of a parameter will contain its name, type, access mode, index in the parameter array, offset for binary data, parameter data size, current value, default value, minimum and maximum values, parameter value step, for arrays - the maximum number of elements.
- Access: “unlocked”.
- URI “GET”: "/api/v1/config/params".
- Request:

1
{ “request”: “READ_PARAMETERS_DESCRIPTION” }

- Answer (example):

```
{
   “result”: “RF_OK”,
   “payload”:{
      "byte_order": "little_endian",
      "factory": [
        {
            "name": "fact_general_firmwareVer",
            "type": "u32_arr_t",
            "access": "read_only",
            "index": 0,
            "offset": 0,
            "size": 12,
            "value": [2, 0, 0],
            "defaultValue": [2, 0, 0],
            "min": 0,
            "max": 4294967295,
            "step": 0,
            "maxCount": 3
        },
        {
            "name": "fact_general_hardwareVer",
            "type": "uint32_t",
            "access": "read_only",
            "index": 1,
            "offset": 12,
            "size": 4,
            "value": 302388224,
            "min": 0,
            "max": 4294967295,
            "step": 0,
            "defaultValue": 302388224
        },
        {
            "name": "fact_general_productCode",
            "type": "uint32_t",
            "access": "locked",
```

```
            "index": 2,
            "offset": 16,
            "size": 4,
            "value": 627,
            "min": 0,
            "max": 65535,
            "step": 0,
            "defaultValue": 627
        },
    ...
}
```


36.6.2. READ_PARAMETERS

- Function: Reading device parameter values. For reading, you can query specific parameters by name or index.
- Access: “unlocked”.
- URI “GET”: "/api/v1/config/params/values".

The Web API request format is described in section Web API, par. Reading and writing parameters.

- Request:

1

```
{
  “request”: “READ_PARAMETERS”,
   “payload”:{
      name: String,
      index: Number (uint32_t)
   }
}
```

Combining *name* and *index* parameters is supported. Only one parameter by name and/or one parameter by index is requested at a time.

2

```
{
  “request”: “READ_PARAMETERS”,
   “payload”:{
      names:{
         “parameter_1_name”,.
         “parameter_2_name”,
         ...,
      },
      indexes:{
         “parameter_3_index”,.
         “parameter_4_index”,
         ...,
      }
   }
}
```

Combining *name* and *index* parameters is supported.

- Answer (example):

```
{
   “result”: “RF_OK”,
   “payload”:{
      "fact_general_hardwareVer": 302388224,
      "user_trigger_counter_dir": 0
   }
```

}	
result	The result is in accordance with the response codes. If successful, “RF_OK”.
payload	Regardless of the type of request, the response has the same format.

36.6.3. WRITE_PARAMETERS

- Function: Writing device parameter values. To set the value, you can query specific parameters by name or index.
- Access: “unlocked”.
- URI “PUT”: "/api/v1/config/params/values".

The Web API request format is described in section Web API, par. Reading and writing parameters.

- Request:

```
{
  “request”: “WRITE_PARAMETERS”,
   “payload”:{
      “parameter_1_name”: “parameter_1_value”,
      “parameter_2_name”: “parameter_2_value”,
      ...
   }
}
```

- Answer (example):

```
{
   “result”: “RF_OK”,
   “payload”:{
      "user_sensor_framerate": "RF_OK",
      "user_sensor_exposure1": "RF_OK"
   }
}
```

result	The result is in accordance with the response codes. If successful, “RF_OK”. If an error occurs while setting the value, the overall result other than “RF_OK” will be returned.

36.6.4. SAVE_CURRENT_PARAMETERS

- Function: Saving the current values of device parameters in the user area of non-volatile memory. The saved values will be used the next time the device is turned on.
- Access: “unlocked”.
- URI “GET”: "/api/v1/config/params/save".
- Request:

```
{
  “request”: “SAVE_CURRENT_PARAMETERS”
}
```

- Answer (example):

```
{
   “result”: “RF_OK”
}
```

result	The result is in accordance with the response codes. If successful, “RF_OK”.

36.6.5. SAVE_RECOVERY_PARAMETERS

- Function: Saving the current device parameters to the recovery area. These parameters will be applied if the parameters from the user area are corrupted.
- Access: “unlocked”.
- URI “GET”: "/api/v1/config/params/recovery/save".
- Request:

```
{
  “request”: “SAVE_RECOVERY_PARAMETERS”
}
```

Answer (example):

```
{
   “result”: “RF_OK”
}
```

result	The result is in accordance with the response codes. If successful, “RF_OK”.

36.6.6. LOAD_RECOVERY_PARAMETERS

- Function: Downloading device parameter values from the recovery area. The downloaded values will be written to the user area and the device will automatically reboot.
- Access: “unlocked”.
- URI “GET”: "/api/v1/config/params/recovery/load".
- Request:

```
{
  “request”: “LOAD_RECOVERY_PARAMETERS”
}
```

Answer (example):

```
{
   “result”: “RF_OK”
}
```

result	The result is in accordance with the response codes. If successful, “RF_OK”.

36.6.7. REBOOT_DEVICE

- Function: Rebooting the device. All unsaved data and parameters will not be restored.
- Access: “unlocked”.
- URI “GET”: "/api/v1/reboot".
- Request:

```
{
  “request”: “REBOOT_DEVICE”
}
```

Answer (example):

```
{
   “result”: “RF_OK”
}
```

result	The result is in accordance with the response codes. If successful, “RF_OK”.

36.6.8. READ_PROFILES_DUMP

- Function: Obtaining a dump of profiles from the device. The dump is returned in several fragments (the number depends on the size of the dump), typically each fragment contains 10 profiles organized in an array. For convenience, the fragment contains the serial number (index) in the dump of the first profile in the array, a flag indicating whether this is the last fragment, and the array with the dump.
- Access: "unlocked".
- Request:

```
{
   "request": "READ_PROFILES_DUMP"
}
```

Answer (example):

```
{
  "result": String (*),
   "payload":{
      "index": Number (uint32_t),
      "last": Bool,
      "profiles": [
         {...},
         {...},
         {...},
      ]
   }
}
```

result	The result is in accordance with the response codes. If successful, "RF_OK".
payload: index	The number in the dump for the first profile in the "profiles" array.
payload: last	Flag of the last fragment of the dump.
payload: profiles	An array of dump profiles.

36.7. Profile transfer format

Profiles are transmitted without a request immediately after the frame is processed.

RAW format, 2024.01.08

```
{
  "format": "DATA_FORMAT_RAW_PROFILE",
   "discrete": number (float),
   "measure_index": number (uint32_t),
   "encoder_value": number (uint32_t),
   "encoder_dir": number (uint32_t),
   "profile": blob,
   "intensity": blob
}
```

format	The format of the transmitted profile. In this case, the subpixel position of the point in integer format. **To obtain a fractional value, it is necessary to divide the integer value by the value of the "discrete" field.**
discrete	The divisor value to obtain the fractional value of the point position.
measure_index	The serial number of the measurement (the frame on the basis of which the profile/profiles were calculated). In the future, several profiles can be calculated from one measurement, for example, with different detection or additional processing parameters.
encoder_value	The encoder value recorded in the middle of the exposure time of the frame from which the measurement was obtained.

encoder_dir	The direction of movement calculated from the encoder in the middle of the exposure time of the frame from which the measurement was obtained.
profile	Binary profile data - sequentially placed uint16_t positions of the point for each column of the CMOS sensor. To obtain the fractional value, see the description of the “format” field.
intensity	An optional field that is only transmitted when intensity transmission is enabled. Contains binary brightness data for each point in uint8_t format.

METRIC format, 2024.01.08

```
{
  “format”: “DATA_FORMAT_METRIC”,
   “scaling”: number (float),
   “measure_index”: number (uint32_t),
   “encoder_value”: number (uint32_t),
   “encoder_dir”: number (uint32_t),
   “profile”: blob,
   “intensity”: blob
}
```

format	The format of the transmitted profile. In this case, the metric (in mm) position of the point in integer format. **To obtain a fractional value, it is necessary to multiply the integer value by the value of the “scaling” field.**
scaling	The scale value to obtain the fractional value of the point position.
measure_index	The serial number of the measurement (the frame from which the profile/profiles are calculated). In the future, several profiles can be calculated from one measurement, for example, with different detection or additional processing parameters.
encoder_value	The encoder value recorded in the middle of the exposure time of the frame from which the measurement was obtained.
encoder_dir	The direction of movement calculated from the encoder in the middle of the exposure time of the frame from which the measurement was obtained.
profile	Binary profile data - sequentially placed pairs [int16_t, uint16_t] of the point position along X and Z. To obtain the fractional value, see the description of the “format” field.
intensity	An optional field that is only transmitted when intensity transmission is enabled. Contains binary brightness data for each point in uint8_t format.

36.8. Video frame transfer format

Video frames from which profiles are calculated are transmitted without a request. Since the video information stream is auxiliary, video frames are transmitted asynchronously relative to the profiles, i.e. one video frame can be transmitted to several transmitted profiles.

2024.01.08

```
{
  “fact_sensor_width”: number (uint32_t),
   “fact_sensor_height”: number (uint32_t),
   “frame_width”: number (uint32_t),
   “frame_height”: number (uint32_t),
   “user_roi_pos”: number (uint32_t),
   “user_roi_size”: number (uint32_t),
   “user_roi_active”: bool,
   “user_roi_enabled”: bool,
   “frame”: blob
}
```

fact_sensor_width	Physical horizontal size of the CMOS sensor (number of columns).
fact_sensor_height	Physical vertical size of the CMOS sensor (number of lines).

`frame_width`	Logical horizontal size of the CMOS sensor, taking into account the region of interest.
`frame_height`	Logical vertical size of the CMOS sensor, taking into account the region of interest.
`user_roi_pos`	Current vertical position of the region of interest (CMOS sensor line number).
`user_roi_size`	Current vertical size of the region of interest (number of CMOS sensor lines).
`user_roi_active`	ROI activity flag. The region of interest (ROI) can be enabled but not active - if a profile is not detected, the ROI activity flag is cleared and the entire frame is searched for the profile.
`user_roi_enabled`	A flag indicating that ROI is enabled.
`frame`	Binary frame data in uint8_t format per point. Binary data size: fact_sensor_width * frame_height.

To reduce the load on the Ethernet connection, a command is provided to set the frame sending frequency:

```
{
   “send_period_ms”: number (uint32_t)
}
```

`send_period_ms`	The period of sending frames in ms for the duration of the current connection (i.e. when the socket is closed and reconnected, the command must be sent again). **When connecting, the default value of 40 ms is used.**

37. Warranty policy

Warranty assurance for 2D Laser Scanners RF62x Series – 24 months from the date of shipping; warranty shelf-life – 12 months.

Warranty repair is not provided in the following cases:

- mechanical damage caused by impacts or falling from height,
- damage caused by opening the housing, incorrect connection, or absence of grounding.

38. Technical support

Technical support for issues related to incorrect work of the scanners and to problems with settings is free.

Technical support related to using the scanners is free. This kind of technical support includes consulting about ways to apply the scanner, and training to work with software tools and libraries.

Technical support for software developed by the customer is paid, and includes the possibility to add new features to software.

Technical support contacts:

- E-mail: info@riftek.com

39. Revisions

Date	Revision	Description
16.11.2018	1.0.0	Starting document.
28.12.2018	1.0.1	1. Added the ability to manually adjust the laser output power. 2. Added the description of Recovery mode, section 28. 3. Fixed minor inaccuracies in the description.
27.06.2019	1.0.2	1. Added eleven new scanner models with ranges (Z) from 250 to 1165 mm, par. 7.2. 2. Added settings that expand the dynamic range of scanners, par. 20.1. 3. Added profile filtering functions (median and bilateral filters), par. 20.2. 4. Added "Peak selection mode" function, par. 20.1.1. 5. Added the ability to include the point brightness values in the profile package, par. 19.3.

Date	Revision	Description
		6. Added the modes for profile accumulation, viewing and saving profiles, building 3D and brightness models, par. 16.2., 20.3. 7. Changed the firmware file format, par. 23.3. 8. Fixed minor inaccuracies in the description.
06.07.2020	2.0.0	1. Redesigned the web interface. 2. Added the multiple exposure mode. 3. Changed the measurement triggering system. 4. Added the ability to view oscillograms of signals at the scanner inputs. 5. Added the ability to edit defective pixels. 6. Added notifications in the web interface about important events in the scanner. 7. Added a description of the block diagram of the scanner's internal synchronization module.
04.01.2021	2.1.0	1. Added a detailed description of the Smart tab. 2. Added Annex 3. Overall and mounting dimensions of scanners with options. 3. Added Annex 4. Web API.
16.04.2021	2.1.1	1. Added profile approximation by arcs. 2. Terminology is clarified.
20.09.2021	2.1.2	1. Updated Section 11 "Ethernet interface and user software development". 2. Added a description of the "Intensity clipping" and "Peak width" parameters, par. 19.1. 3. Updated par. 23.2.1 "Updating and saving the firmware". 4. Updated par. 24.1.3 "Profile Approximation tab". 5. Added par. 24.2.1.3 "Clarification of approximating line segments and arcs". 6. Added a description of the "Scalar filtering" function, par. 24.3.2.3. 7. Added par. 24.3.2.2 "Welding section". 8. Added a description of the "template detector" smart block, par. 24.3.2.3. 9. Added a description of the "robot protocol HND1" smart block, par. 24.3.2.7. 10. Added Annex 5 "HND1 protocol, version 1.0." 11. Added Annex 6 "Template Detector smart block and Template Editor". 12. Added Annex 7 "C-script smart block". 13. Updated some screenshots and fixed minor inaccuracies.
05.09.2022	2.2.0	1. Added Section 12 "Compatibility of software versions". 2. Updated Section 14 "Web interface". 3. Added Section 15 "Operating modes". 4. Updated Section 26 "Smart tab". 5. Removed Annex "HND1 protocol, version 1.0". 6. Updated Annex 5 "Template Detector smart block and Template Editor". 7. Updated Annex 6 "C-script smart block". 8. Added Annex 7 "Calibration of the scanner relative to the robot".
14.08.2023	3.0.0	1. Updated: section 4. "General information", section 6 "Configurations, operating modes, options", section 7 "Basic technical data", section 8 "Example of item designation when ordering", section 10 "Connection", section 14 "Web interface operating modes", par. 19.1 "CMOS sensor parameters", par. 19.3.3 "Removing background light from extraneous light sources", par. 19.4 "ROI mode settings", par. 19.5 "Data stream control", par. 24.3 "Licenses section", section 25 "Smart tab". 2. Added: Annex 7 "Overall and mounting dimensions ", Annex 8. "Connectors and cables". 3. Removed: section 12 "Compatibility of software versions".
14.11.2023	4.0.0	Added description of RF627BiSmart, RF628, RF629 and RF6292.
22.02.2024	4.0.1	Added description of proprietary ProfiTalk protocol.
30.07.2024	5.0.0.	Changes for RF627Smart Series Scanners: 1. Operating frequency (485->520 Hz) and X resolution (640 (1280)->728 (1456)) have been changed. 2. The RF627BiSmart-27/10-8/11 binocular scanner model has been added. 3. **EDR Mode** has been excluded. 4. **Interleaved exposure mode** has been excluded.

Date	Revision	Description
		5. **DS Mode** has been excluded.
20.11.2025	6.0.0	1. Enabling Smart functions in RF628, RF629, and RF6292 scanners. 2. Increasing the ROI frequency in RF628, RF629, and RF6292 scanners to 21,500 Hz. 3. Adding a point cloud import function. 4. Adding point cloud measurement tools. 5. Added a Smart weld tracking block (longitudinal weld on a pipe) for managing non-destructive testing systems. 6. A multilingual web interface has been implemented (7 languages). 7. Added special sensors for scanning internal holes and threads.

40. Distributors

AUSTRALIA

Applied Measurement Australia Pty Ltd
RAILWAY INSTRUMENTS ONLY
Thornton Plaza, Unit 5,
27 Thornton Crescent, Mitcham
VIC 3132, Australia
Tel: +61 39874 5777
Fax: +61 39874 5888
sales@appliedmeasurement.com.au
www.appliedmeasurement.com.au

BELGIUM

Althen Sensors & Controls BV
Verrijn Stuartlaan 40, 2288 EL,
Rijswijk, Leidschendam
The Netherlands
Tel: +31 0 70 392 4421
Tel: +31 0 61 396 7830
Tel: +31 0 64 323 8393
sales@althen.nl
info@althen.nl
www.althensensors.com

BOSNIA AND HERZEGOVINA

ASCO RAIL sp. z o.o.
EXCLUSIVE REPRESENTATIVE FOR RAILWAY EQUIPMENT
ul. Wielowiejska 53, 44-120
Pyskowice, Poland
Tel: +48 32 230 45 70
Fax: + 48 32 233 21 34
biuro@ascorail.pl
export@ascorail.pl
www.ascorail.pl

BRAZIL

CAPI Controle e Automacao Ltda
Rua Itororo, 121, CEP 13466-240
Americana-SP, Brazil
Tel: +55 19 36047068
Fax: +55 19 34681791
capi@capicontrole.com.br
www.capicontrole.com.br

BULGARIA

ASCO RAIL sp. z o.o.
EXCLUSIVE REPRESENTATIVE FOR RAILWAY EQUIPMENT
ul. Wielowiejska 53, 44-120
Pyskowice, Poland
Tel: +48 32 230 45 70
Fax: + 48 32 233 21 34
biuro@ascorail.pl
export@ascorail.pl
www.ascorail.pl

CHILE

MOL INGENIERIA LTDA
EXCLUSIVE REPRESENTATIVE FOR RAILWAY EQUIPMENT
Republica de Honduras 11936
Las Condes, Santiago de Chile
Tel: +56 9 59200362
hconcha@molingenieria.com
www.molingenieria.com

CHILE

Verne SpA
Apoquindo 2818, oficina 31
Las Condes, Santiago, Chile
Tel: +56 2 228858633
info@verne.cl
jsaavedra@verne.cl
www.verne.cl

CHINA

Beijing Haiwei Lutong Technology Co., Ltd
Yard 1, Tianxing Street, Fangshan
District, Beijing, China
Tel: +86 10 8366 1866
Fax: +86 10 8366 1866
info@haiwlt.com
www.haiwlt.com

CHINA

Chongqing Wolf Industrial Technology Co., Ltd
Room 2307 / 2308, Light of City
international business building,
No. 19 Jiangnan Avenue, Nan'an
District, Chongqing, China
Tel: 023 62832618
Fax: 023 62832113
info@wolf-hk.com
www.wolf-hk.com

CHINA

Beijing Gemston Mechanical & Electrical Equipment Co., Ltd
RAILWAY INSTRUMENTS ONLY
Room 613, Anfu Mansion, Fengtai District, Beijing, China
Tel: +86 10 6765 0516
Fax: +86 10 6765 6966
Mobile: +86 137 1755 1423
dh0526@163.com
www.baoft.cn

CHINA

Xi'an Win-Success Automation Technology Co.,Ltd
Room 3-1-1039, Iduhui Building, No.11 Tangyan South Road
High-Tech Zone, Xi'an
Shaanxi PRC, China
Tel: +86 29 81106280
Fax: +86 29 81106285
Mob: +86 133 19271405
info@maxsenor.com
www.maxsensor.com

CHINA

Micron-Metrology co., Ltd
No.2, Kecheng Rd., Industrial Park District, Suzhou,
Jiangsu Province., China
Tel: 0512 65589760
Mob: +86 189 1806 9807
sales@micron-metrology.cn
www.micron-metrology.cn

CHINA

Zhenshangyou Technologies Co., Ltd
Rm 2205-2210, Zhongyou Hotel
1110 Nanshan Road, Nanshan District 518054 Shenzhen, China
Tel: +86 755-26528100/8011/8012
Fax: +86 755-26528210/26435640
info@51sensors.com
www.51sensors.com

CROATIA

ASCO RAIL sp. z o.o.
EXCLUSIVE REPRESENTATIVE FOR RAILWAY EQUIPMENT
ul. Wielowiejska 53, 44-120
Pyskowice, Poland
Tel: +48 32 230 45 70
Fax: + 48 32 233 21 34
biuro@ascorail.pl
export@ascorail.pl
www.ascorail.pl

CZECH REPUBLIC

ASCO RAIL sp. z o.o.
EXCLUSIVE REPRESENTATIVE FOR RAILWAY EQUIPMENT
ul. Wielowiejska 53, 44-120
Pyskowice, Poland
Tel: +48 32 230 45 70
Fax: + 48 32 233 21 34
biuro@ascorail.pl
export@ascorail.pl
www.ascorail.pl

DENMARK

BLConsult
Ryssbalt 294
95 291 Kalix, Sweden
Tel: +46 70 663 19 25
info@blconsult.se
www.blconsult.se

ESTONIA

FoodLab OU
Haabersti linnaosa, Astangu tn 52
13519 Eesti, Tallinn, Estonia
Tel: +372 56 363110
foodlab.ee@gmail.com

FINLAND

Kvalitest Industrial AB
EXCEPT FOR RAILWAY INSTRUMENTS
Ekbacksvagen 28,
16869 Bromma, Sweden
Tel: +46 0 76 525 5000
sales@kvalitest.com
www.kvalitest.com
www.kvalitest.se

FINLAND

TERASPYORA-STEELWHEEL OY
RAILWAY INSTRUMENTS ONLY
Juvan teollisuuskatu 28
FI-02920 ESPOO, Finland
Tel: +358 400 422 900
Fax: +358 9 2511 5510
steelwheel@steelwheel.fi
www.teraspyora.fi

FRANCE

BLET Measurement Group S.A.S.
1 avenue du President Georges Pompidou, 92500 Rueil Malmaison, France
Tel: + 33 0 1 80 88 57 85
Fax: +33 0 1 80 88 57 93
technique@blet-mesure.fr
www.blet-mesure.fr

GERMANY

Disynet GmbH
Breyeller Str. 2, 41379
Brueggen, Germany
Tel: +49 2157 8799 0
Fax: +49 2157 8799 22
disynet@sensoren.de
www.sensoren.de

GERMANY

Finger GmbH & Co. KG
OPTICAL MICROMETERS ONLY
Sapelloh 172, 31606
Warmsen, Germany
Tel: +49 5767 96020
Fax: +49 5767 93004
finger@finger-kg.de
www.finger-kg.de

GERMANY

ALTHEN GmbH MeЯ- und Sensortechnik
Dieselstrasse 2, 65779
Kelkheim, Germany
Tel: +49 0 6195 7 00 60
info@althen.de
www.althensensors.com/de/

HUNGARY

ASCO RAIL sp. z o.o.
EXCLUSIVE REPRESENTATIVE FOR RAILWAY EQUIPMENT
ul. Wielowiejska 53, 44-120
Pyskowice, Poland
Tel: +48 32 230 45 70
Fax: + 48 32 233 21 34
biuro@ascorail.pl
export@ascorail.pl
www.ascorail.pl

INDIA

Influx Big Data Solutions Pvt Ltd
No:2, Krishvi, Ground Floor,
Old Airport Road, Domlur,
Bangalore - 560071, India
Tel: +91 73 37748490
Tel: +91 94 48492380
milan@influxtechnology.com
support_india@influxtechnology.com
www.influxtechnology.com

INDIA

Paragon Instrumentation Engineers Pvt. Ltd.
RAILWAY INSTRUMENTS ONLY
200, Station Road,
Roorkee, 247 667, India
Tel: +91 1332 272394
tanuj@paragoninstruments.com
www.paragoninstruments.com

INDONESIA

PT. DHAYA BASWARA SANIYASA
Botanic Junction Blok H-9 NO. 7
Mega Kebon Jeruk, Joglo
Jakarta,11640, Indonesia
Tel: +62 21 2932 5859
management@ptdbs.co.id

ISRAEL

Nisso Dekalo Import Export LTD
1 David Hamelech Street
Herzlia 46661 Israel
Tel: +972 99577888
Fax: +972 99568860
nissodekaloltd@outlook.com
www.fly-supply.net
www.aircraft-partsupply.com

ITALY

FAE s.r.l.
Via Tertulliano, 41
20137 Milano, Italy
Tel: +39 02 55187133
Fax: +39 02 55187399
fae@fae.it
www.fae.it

JAPAN

Tokyo Instruments, Inc.
6-18-14 Nishikasai, Edogawa-ku,
Tokyo, 134-0088 Japan
Tel: +81 3 3686 4711
Fax: +81 3 3686 0831
f_kuribayashi@tokyoinst.co.jp
www.tokyoinst.co.jp

LATVIA

FoodLab OU
Haabersti linnaosa, Astangu tn 52
13519 Eesti, Tallinn, Estonia
Tel: +372 56363110
foodlab.ee@gmail.com

LUXEMBOURG

Althen Sensors & Controls BV
Verrijn Stuartlaan 40, 2288 EL,
Rijswijk, Leidschendam
The Netherlands
Tel: +31 0 70 392 4421
Tel: +31 0 61 396 7830
Tel: +31 0 64 323 8393
sales@althen.nl
info@althen.nl
www.althensensors.com

MALAYSIA

OptoCom InstruVentures
H-49-2, Jalan 5, Cosmoplex
Industrial Park, Bandar Baru
Salak Tinggi, Sepang, Malaysia
Tel: 603 8706 6806
Fax: 603 8706 6809
optocom@tm.net.my
www.optocom.com.my

MONTENEGRO

ASCO RAIL sp. z o.o.
EXCLUSIVE REPRESENTATIVE FOR RAILWAY EQUIPMENT
ul. Wielowiejska 53, 44-120
Pyskowice, Poland
Tel: +48 32 230 45 70
Fax: + 48 32 233 21 34
biuro@ascorail.pl
export@ascorail.pl
www.ascorail.pl

NETHERLANDS

Althen Sensors & Controls BV
Verrijn Stuartlaan 40, 2288 EL,
Rijswijk, Leidschendam
The Netherlands
Tel: +31 0 70 392 4421
Tel: +31 0 61 396 7830
Tel: +31 0 64 323 8393
sales@althen.nl
info@althen.nl
www.althensensors.com

NORWAY

BLConsult
Ryssbalt 294,
95 291 Kalix, Sweden
Tel: +46 70 663 19 25
info@blconsult.se
www.blconsult.se

NORWAY

Salitec AS
PB 468, N-1327
Lysaker, Norway
Tel: +47 23 891015
Fax: +47 92101005
mail@salitec.no
www.salitec.no

PERU

Verne Peru S.A.C.
Las Codornices 104,
Surquillo, Lima, Peru
Tel/fax: +51 992436734
info@verne.cl
www.verne.cl

POLAND

ASCO RAIL sp. z o.o.
EXCLUSIVE REPRESENTATIVE FOR RAILWAY EQUIPMENT
ul. Wielowiejska 53, 44-120
Pyskowice, Poland
Tel: +48 32 230 45 70
Fax: + 48 32 233 21 34
biuro@ascorail.pl
export@ascorail.pl
www.ascorail.pl

POLAND

RIFTEK EUROPE sp. z o.o.
ul. Domaniewska 17/19, 02-672
Warsaw, Poland
info@riftek.com
www.riftek.com

PORTUGAL

Campal Inovacoes Ferroviarias Lda.
Lagoas Park, Edificio 7, 1° Piso
Sul, 2740-244 Porto Salvo, Oeiras, Portugal
Tel: +351 21 584 4348
campal@campal.pt
www.campal.pt

SERBIA

ASCO RAIL sp. z o.o.
EXCLUSIVE REPRESENTATIVE FOR RAILWAY EQUIPMENT
ul. Wielowiejska 53, 44-120
Pyskowice, Poland
Tel: +48 32 230 45 70
Fax: + 48 32 233 21 34
biuro@ascorail.pl
export@ascorail.pl
www.ascorail.pl

SLOVAKIA

ASCO RAIL sp. z o.o.
EXCLUSIVE REPRESENTATIVE FOR RAILWAY EQUIPMENT
ul. Wielowiejska 53, 44-120
Pyskowice, Poland
Tel: +48 32 230 45 70
Fax: + 48 32 233 21 34
biuro@ascorail.pl
export@ascorail.pl
www.ascorail.pl

SLOVENIA

ASCO RAIL sp. z o.o.
EXCLUSIVE REPRESENTATIVE FOR RAILWAY EQUIPMENT
ul. Wielowiejska 53, 44-120
Pyskowice, Poland
Tel: +48 32 230 45 70
Fax: + 48 32 233 21 34
biuro@ascorail.pl
export@ascorail.pl
www.ascorail.pl

SOUTH KOREA

BS Holdings
B-201,Wonpogongwon 1ro,
59 Danwon-gu, Ansan-si,
Gyeonggi-do 15455, Republic of Korea
Tel: +82 31 411 5011
Fax: +82 31 411 5015
bsh5011@hanmail.net
www.lasersolution.co.kr

SOUTH KOREA

PROSEN. CO., LTD
M-1001, Songdo techno park IT center, 32, Songdogwahak-ro,
Yeonsu-gu, Incheon, 21984,
Republic of Korea
Tel: +82 32 811 3457
Fax: +82 32 232 7458
trade@prosen.co.kr
www.prosen.co.kr

SPAIN

IBERFLUID Instruments S.A.
C/ Botanica, 122, 08908
L'Hospitalet de Llobregat
Barcelona
Tel: +34 93 447 10 65
Fax: +34 93 334 05 24
myct@iberfluid.com
www.iberfluid.com

SWEDEN

BLConsult
Ryssbalt 294,
95 291 Kalix, Sweden
Tel: +46 70 663 19 25
info@blconsult.se
www.blconsult.se

SWEDEN

Kvalitest Industrial AB
EXCEPT FOR RAILWAY INSTRUMENTS
Ekbacksvagen 28,
16869 Bromma, Sweden
Tel: +46 0 76 525 5000
sales@kvalitest.com
www.kvalitest.com
www.kvalitest.se

SWITZERLAND

ID&T GmbH
Gewerbestrasse 12/a
8132 Egg (Zurich), Switzerland
Tel: + 41 44 994 92 32
Fax: + 41 44 994 92 34
info@idtlaser.com
www.idtlaser.com

THAILAND

Advantech Solution Co., Ltd.
20/170 Motorway Rd.,
Kwang Pravet, Khet Pravet,
Bangkok, Thailand 10250
Tel: +662 1848705
Fax: +662 1848708
sales@advantechsolution.com
www.advantechsolution.com

TURKEY

MAK Elektronik Malzeme Analiz ve Kalite Kontrol Cihazlari Dis Tic. Ltd. Sti.
Cenap Sahabettin Sokak, No:39,
34718 Kosuyolu - Kadikoy /
Istanbul - TURKEY
Tel: +90 216 402 10 34
Fax: +90 216 402 10 35
ulastac@metalografi.net
www.makelektronik.com.tr

TURKEY

TEKMA Muhendislik A.S.
Cevizli Mh. M. Kemal Cd.,
Hukukcular Towers,
A-Blok, No: 66-A/39
Kartal - Istanbul
Tel: +90 216 970 1318
Tel: +90 850 840 2334
info@tekma.eu
www.tekma.eu

UKRAINE

KODA
Frunze st. 22, 61002,
Harkov, Ukraine
Tel/Fax: +38 057 714 26 54
mail@koda.com.ua
www.koda.com.ua

UNITED KINGDOM, IRELAND

Althen UK
Northamptonshire
United Kingdom
Tel: +44 0 7823 921427
t.stoyles@althen.co.uk
www.althensensors.com
www.althencontrols.com

USA

Althen Sensors & Controls
2531 Bradley St., Oceanside, CA, 92056, USA
Tel: 858 633 3572
r.ream@althensensors.com

USA, CANADA, MEXICO

Acuity Products of Schmitt Industries, Inc.
2765 NW Nicolai Street
Portland, OR, 97210, USA
Tel: +1 503 227 7908
Fax: +1 503 223 1258
sales@acuitylaser.com
www.acuitylaser.com

USA, CANADA, MEXICO

International Electronic Machines Corporation
RAILWAY INSTRUMENTS ONLY
850 River Street, Troy,
New York, USA
Tel: +1 518 268-1636
Fax: +1 518 268-1639
marketing@iem.net
www.iem.net