RIFTEK
Sensors & Instruments

LASER SCANNERS

RF627 Series,
RF627Smart Series

User's manual

Contents

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.

2. CE compliance

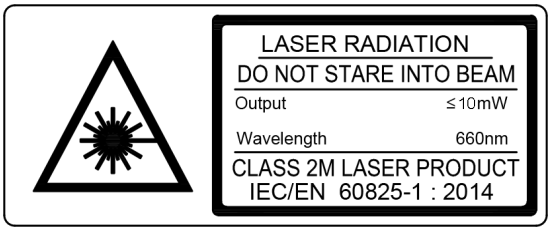
Laser scanners have been developed for use in industry and meet the requirements of the following Directives:

- 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

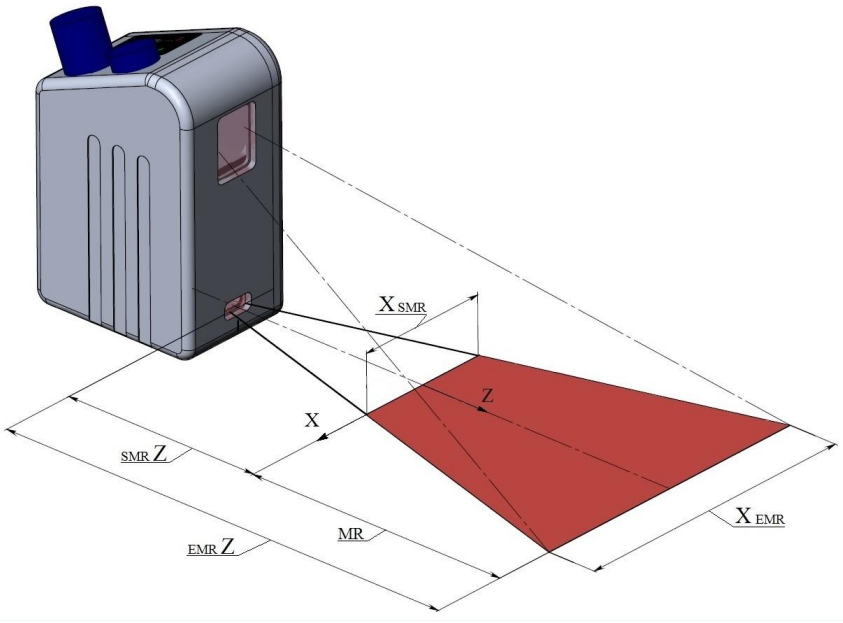
Laser scanners are designed for non-contact measuring and checking of surface profile, position, displacement, dimensions, sorting and sensing of technological objects, 3D models construction.

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. The image of object outline thus formed is analyzed by a FPGA and signal processor, which calculates 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 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.

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.

There are two operating modes in the full working range: Basic mode with the frequency of 485 Hz (profiles/second) and DS mode with the frequency of 938 Hz.

In addition, you can use the ROI function, which makes it possible to increase the working frequency of the scanner in the limited working range up to 5096 Hz in Basic mode and up to 6800 Hz in DS mode.

The **Smart** version of the scanner (RF627Smart) makes 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), as well as to the logic outputs of the scanner in order to control the actuators and notify about product suitability.

NOTE: By purchasing the RF627 scanner, you get the opportunity to use all the functions of the RF627Smart scanner for 100 hours. The trial period can be extended upon request. To use all functions of the RF627Smart scanner for an unlimited time, you must purchase a license (it is possible to purchase a license for certain functions). For more details, refer to par. 23.3.

7. Basic technical data

7.1. Specification

Sampling rate, accuracy, resolution	
Nominal sampling rate (full working range), not less	485 profiles/s (standard mode), 921 profiles/s (DS mode)
Maximum sampling rate (ROI mode)	4884 profiles/s, 6379 profiles/s (DS mode)
Linearity (measurement error), Z axis	±0.05% of the range (standard mode), ±0.1% of the range (DS mode)
Linearity (measurement error), X axis	±0.2% of the range
Resolution, Z axis	0.01% of the range (standard mode), 0.02% of the range (DS mode)
Resolution, X axis	648 or 1296 points (programmable value)
Laser	
660 nm or 405 nm or 450 nm or 808 nm Class 2M according to IEC/EN 60825-1:2014	
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	6 W (without a built-in heater)
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

7.2. Working ranges and dimensions

Range	MR, mm	smrZ, mm	emrZ, mm	Xsmr, mm	Xemr, mm	Size, mm	Weight, g
25/10-8/11	10	25	35	8	11	Figure 1	0.37
65/25-20/22	25	65	90	20	22	Figure 2	0.6
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		

Range	MR, mm	SMR, mm	EMR, mm	Xsmr, mm	Xemr, mm	Size, mm		Weight, g
180/250-170/278	250	180	430	170	278	Figure 3	L=326 mm	2
190/300-160/300	300	190	490	160	300		L=283 mm	1.9
220/300-203/330	300	220	520	203	330		L=374 mm	2.1
260/400-210/400	400	260	660	210	400		L=350 mm	2.2
325/500-268/500	500	325	825	268	500		L=415 mm	2.3
400/600-320/600	600	400	1000	320	600		L=490 mm	2.4
475/700-374/700	700	475	1175	374	700		L=558 mm	2.5
545/800-425/800	800	545	1345	425	800		L=627 mm	2.6
615/900-480/900	900	615	1515	480	900		L=696 mm	2.7
690/1000-535/1000	1000	690	1690	535	1000		L=765 mm	2.8
620/1165-430/1010	1165	620	1785	430	1010		L=554 mm	2.5

Detailed CAD documentation (2D and 3D) is available here:
https://riftek.com/upload/iblock/757/2D_CAD.rar
https://riftek.com/upload/iblock/94e/RF627_3D.zip

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. Overall and mounting dimensions of the scanners are shown in the Figures below. The housing has fastening holes for installing the scanner on the equipment. The sensors, shown in Figure 3, are equipped with an adjustable support that allows you to implement three options for mounting the scanner.

The scanner has two connectors, **Reset** button and LED indicators.

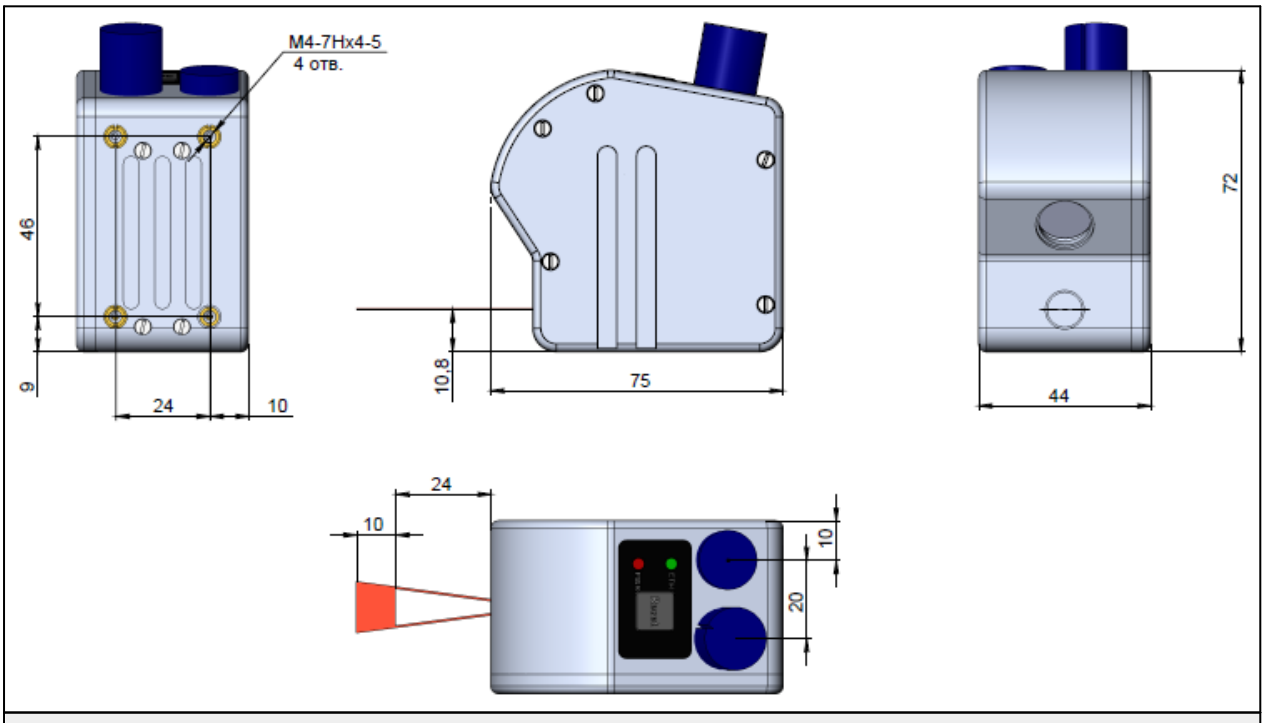


Figure 1

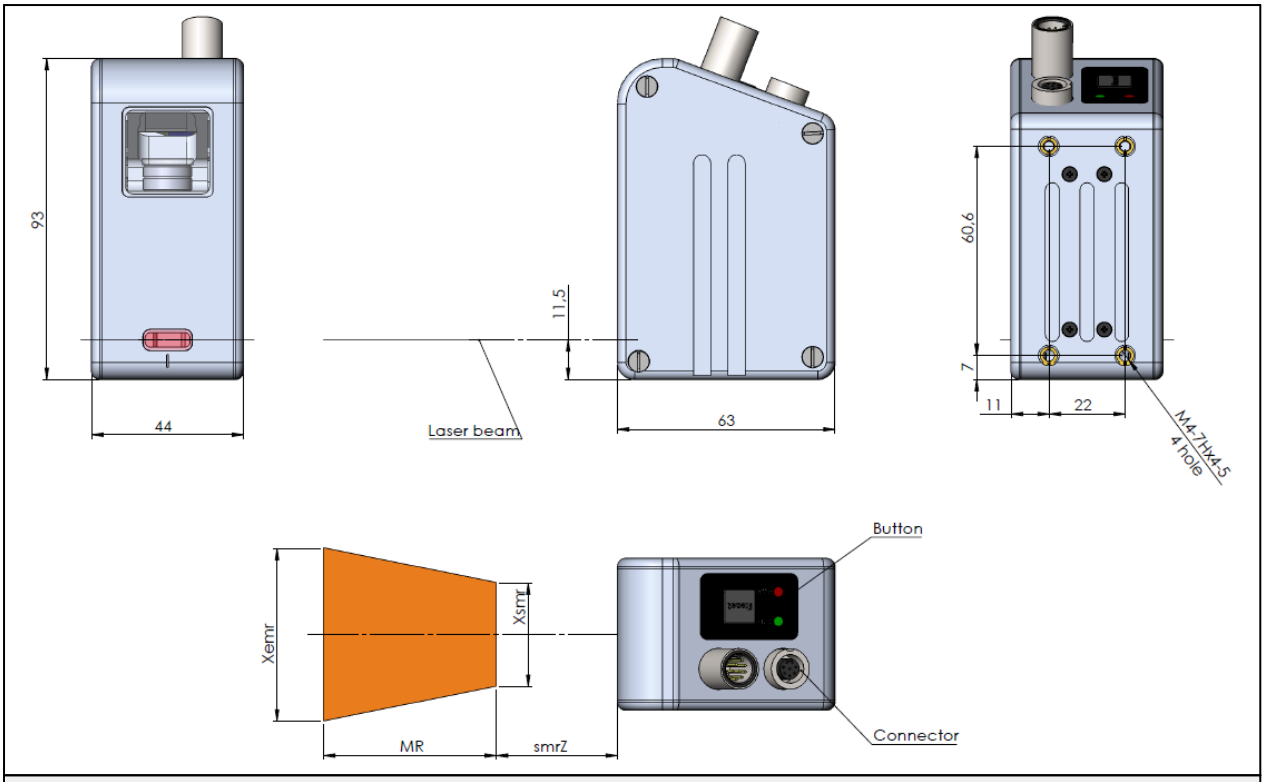


Figure 2

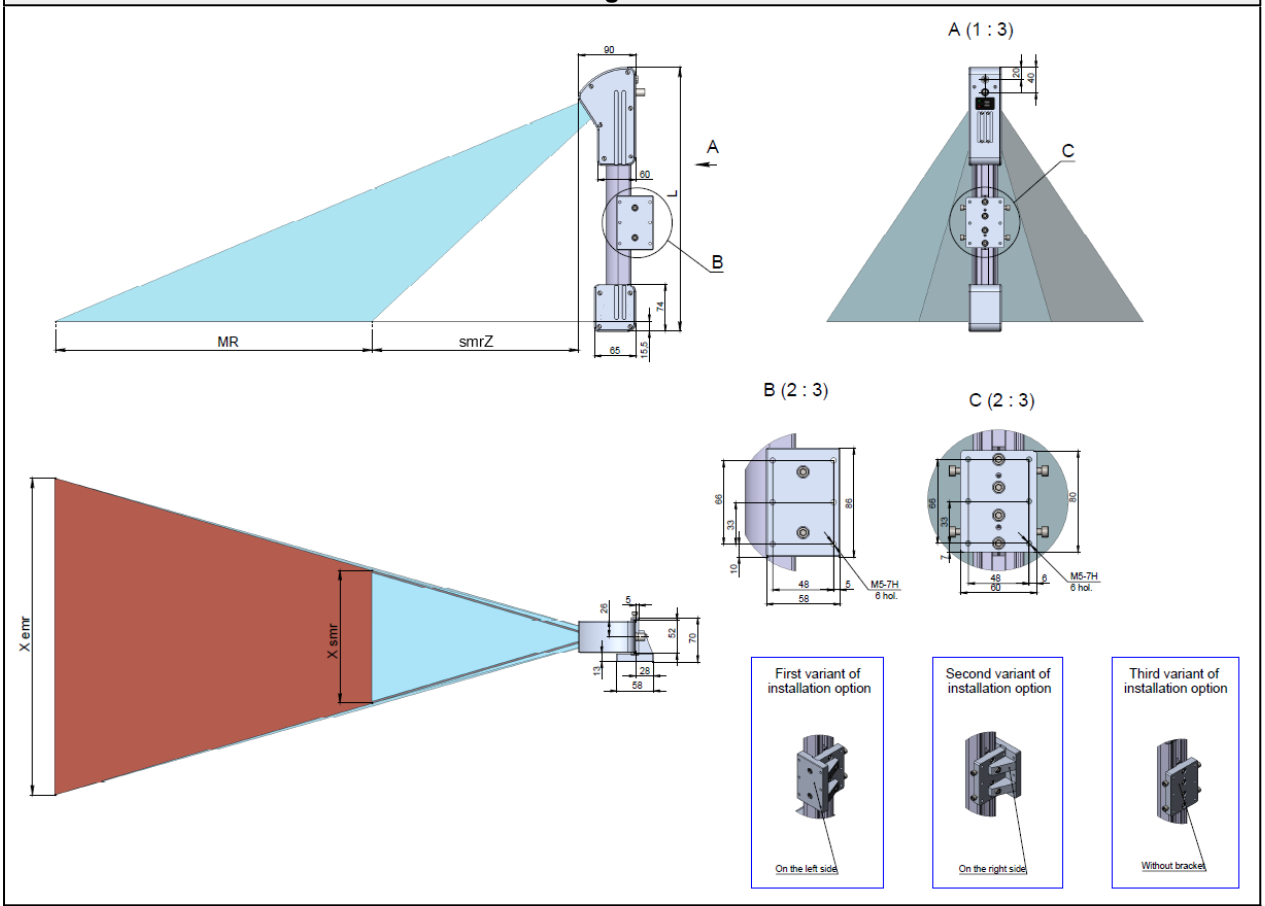


Figure 3

8. Example of item designation when ordering

RF627.(WAVE)-smrZ/MR-Xsmr/Xemr-M(R)-H--AK-EW-AC

Symbol	Description
(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 option.
AC	Built-in air cooling system. Ordering a water cooling system requires consultation with the manufacturer.

Example: RF627BLUE-70/50-30/42-5 – Scanner with a blue laser, smrZ - 70 mm, MR - 50 mm, Xsmr - 30 mm, Xemr - 42 mm, cable length - 5 m.
Note: Overall dimensions of the sensors with the AK, EW and AC options are given in Annex 3. Detailed documentation (2D and 3D) is available for download on RIFTEK's official website.

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.

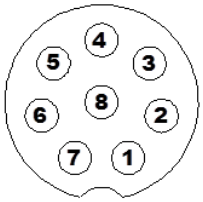
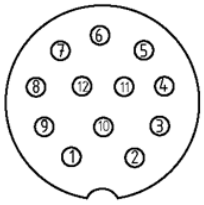
10. Connection

Scanners come with two cables:
1) Ethernet cable;
2) Power cable with synchronization lines and outputs.

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.

10.1. Designation of connector contacts

The scanner is equipped 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:

#	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:

#	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 («-»)

10.2. Cables

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	Description
Black	OUT1-
Gray/pink	IN3-
Red/blue	IN3+
Gray	IN2-
Pink	IN2+
White	NEXT_LAS_OFF
Green	IN1+
Yellow	IN1-
Violet	OUT1+
Red	VIN
Blue	GND
Brown	0V

10.3. 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.

The scanner can be controlled in three ways:

1. Through the embedded web interface (see a description below).
2. Through software developed by the user on the basis of 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, see:

- 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

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

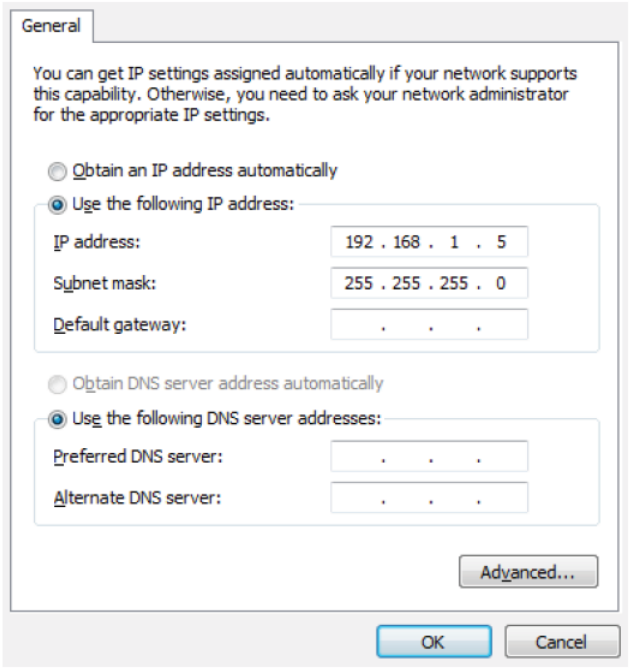
12. Network configuration and the first connection

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.

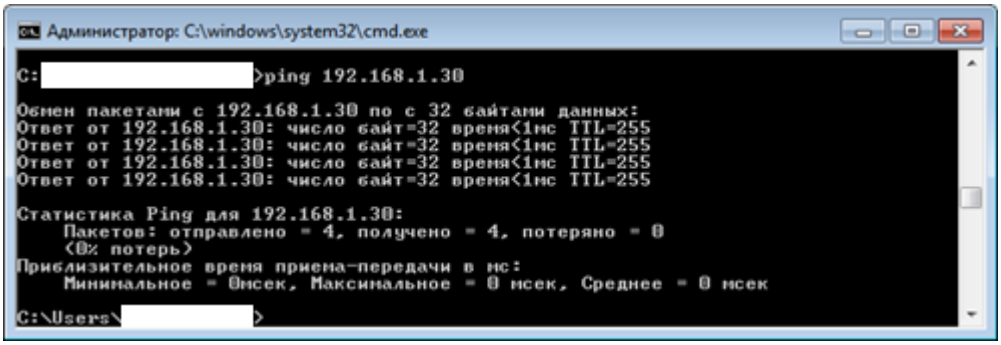
NOTE: Ethernet Jumbo frames are not supported.

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 booting and the Ethernet interface is initializing (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:

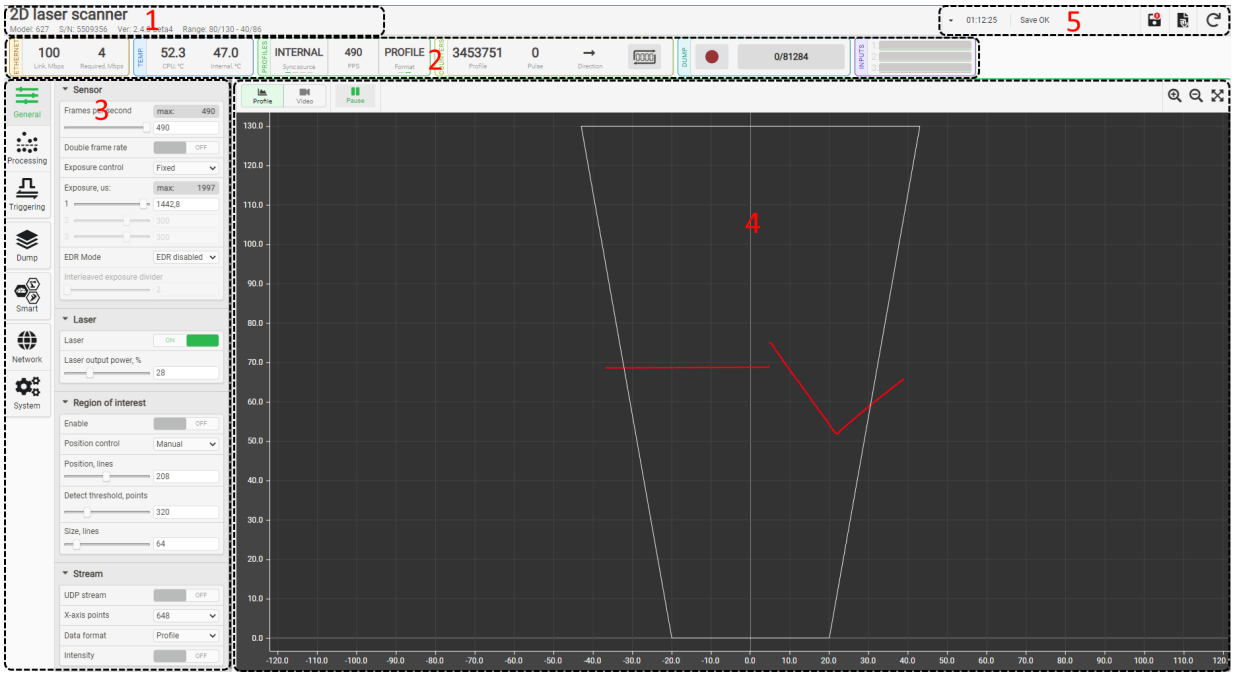


The scanner is ready to operate.

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

13. Web page

The web page is intended to check the operation and configure parameters of the scanner. To access the web page, enter the IP address of the scanner into the address bar of the web browser:



The web page is divided into five areas:

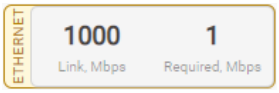
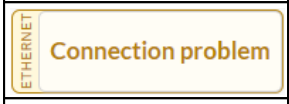
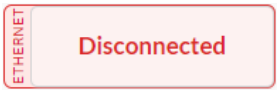
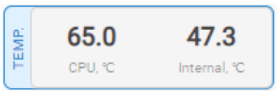
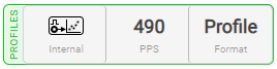
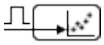
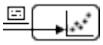
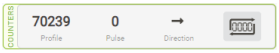
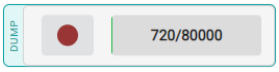
1. Scanner name, scanner model, serial number, firmware version and measuring ranges.

2. Scanner status indicators.

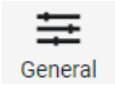
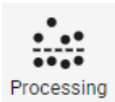
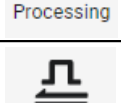
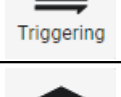
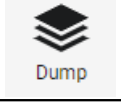
3. Parameterization tabs.
4. Visualization area.
5. Control buttons and notifications.

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:

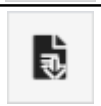
Group	Icon	Description
Ethernet		If the connection is established, the **Link** field and the connection speed value will be displayed. The **Required** field displays the recommended connection speed required for correct operation.
		This message appears when there are delays in the network during data 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. Synchronization sources: • **Internal**  - Synchronization by the internal generator of the scanner. • **External**  - Synchronization by the external trigger. • **Soft**  - Synchronization by the software request. View of buttons when hovering over the cursor: 
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 level of internal memory for recording profiles and the record button (**Record**). Recording is possible only for calibrated profiles (**Data format > Profile**), otherwise the record button is not active.
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:

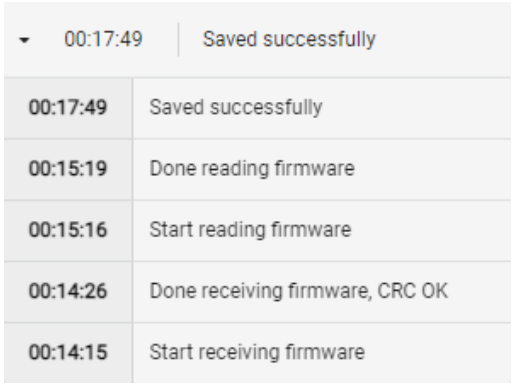
Tab	Icon	Description
General		General scanner settings (CMOS sensor parameters, ROI parameters, laser control, data stream control).
Profile		Profile extraction settings.
Triggering		Settings of input channels of the scanner (triggering modes) and output channels for synchronizing the operation of several 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. 15.1.

Area 5 is located in the upper right corner and contains the notification area and the control buttons.

Button	Name	Description
	**Save configuration**	Save settings to the flash memory of the scanner.
		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. 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.

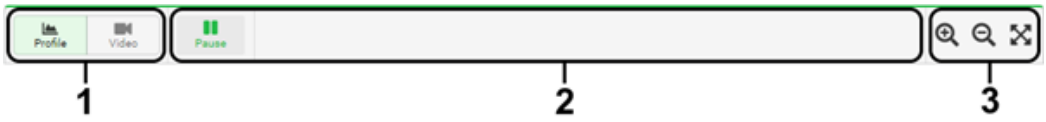
15. Results display area

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.

15.1. 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	Profile	Displaying the profile on a 2D grid.
Video	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 15.

Area 3 contains the following buttons:

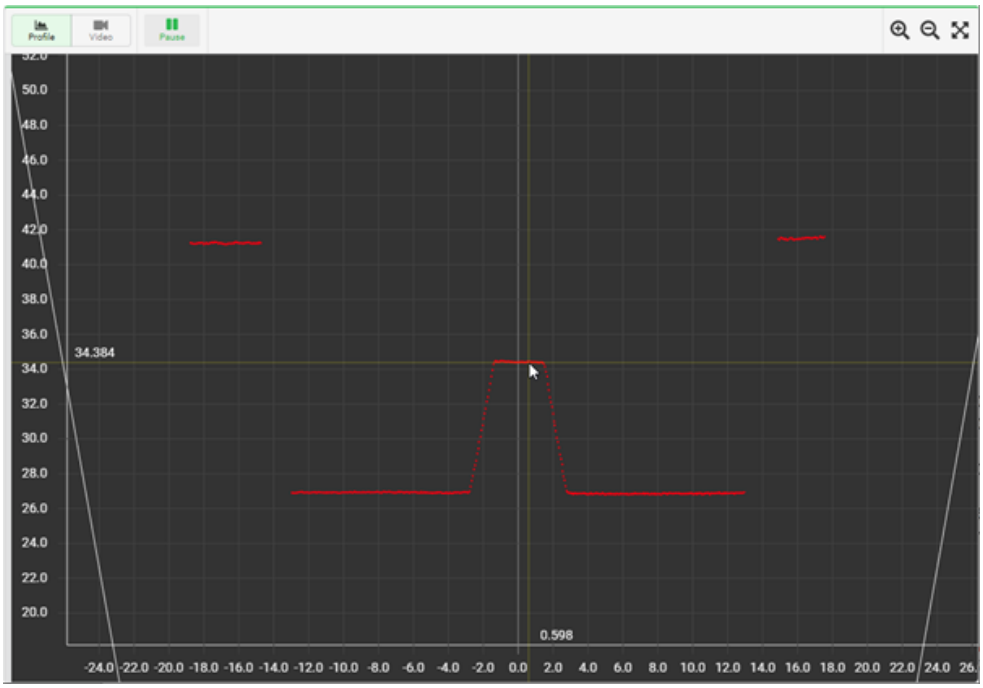
Icon	Description
	Zoom in.
	Zoom out.
	Reset zoom. NOTE: Returning the image to its original scale is also possible by double-clicking the left mouse button in the display area.

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

15.2. Display modes

15.2.1. Profile mode

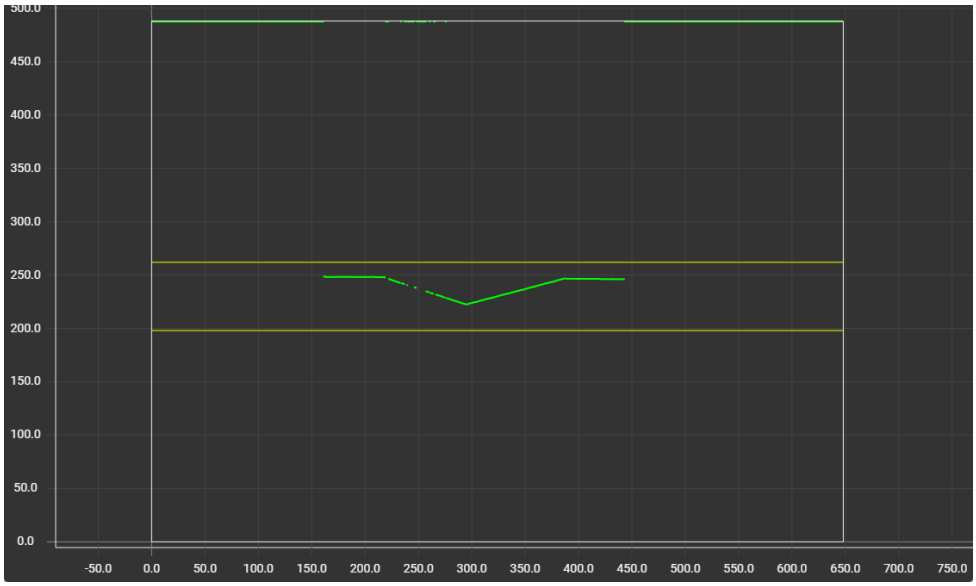
The **Profile** mode is intended to view a two-dimensional profile on the grid. The vertical axis corresponds to the Z coordinate of the scanner, the horizontal axis corresponds to the X coordinate of the scanner.



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.



15.2.2. 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).

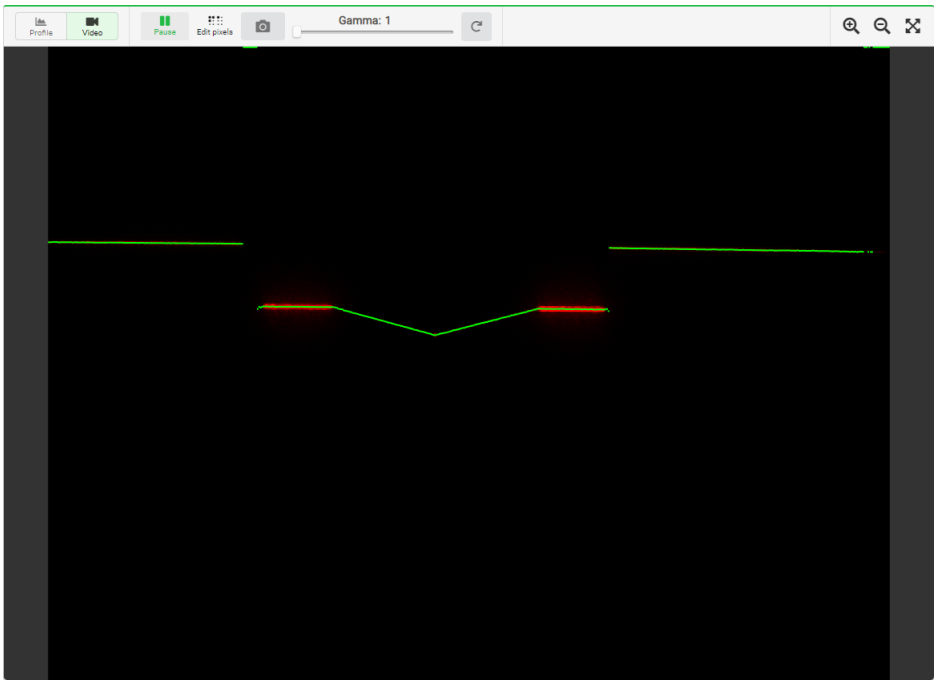
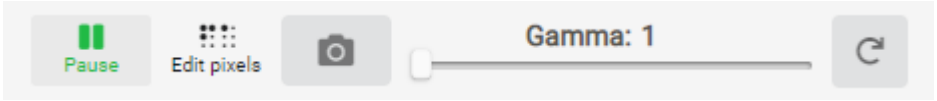


Image transfer speed is determined by computer performance (average value is about 15 frames/s).

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.

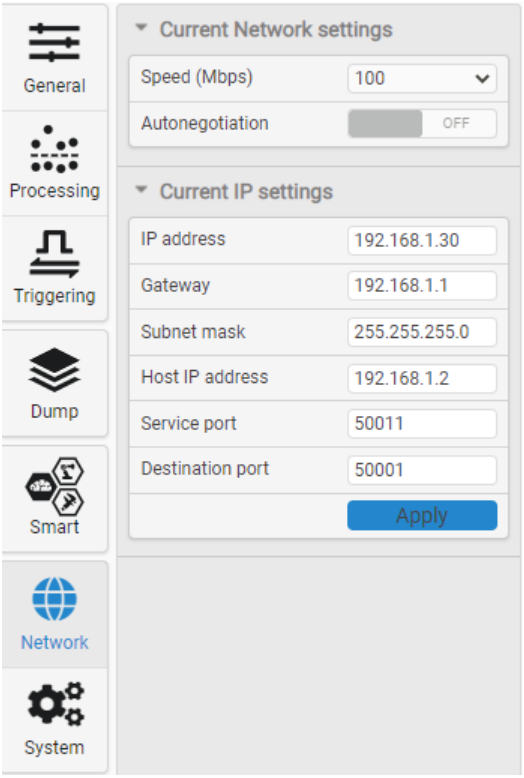
16. Setting parameters

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.).

17. "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.

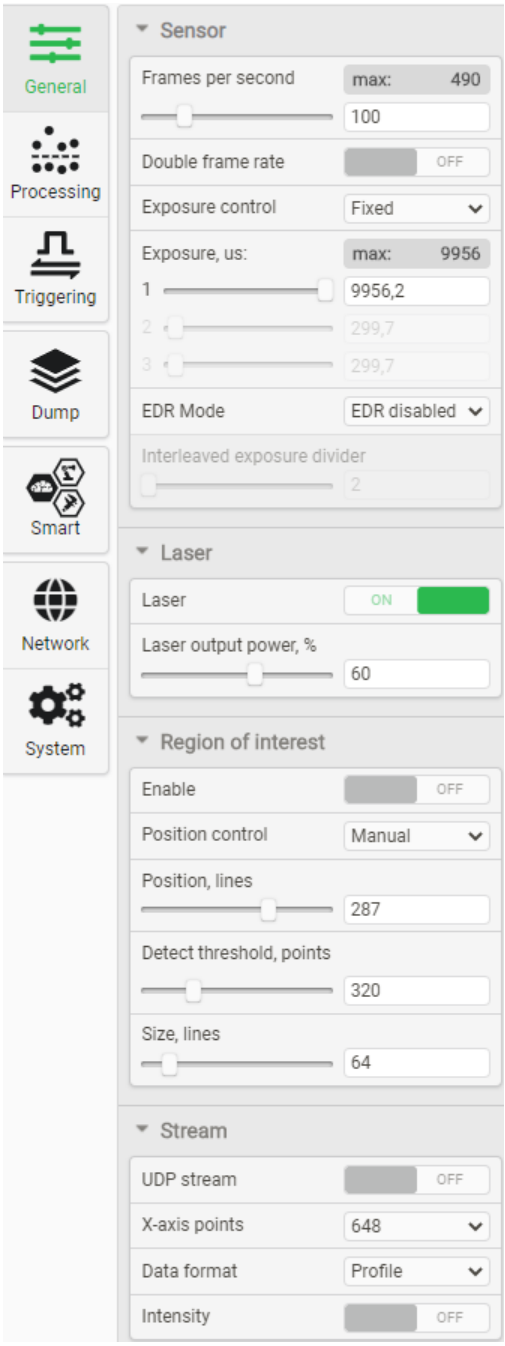
Current IP settings:

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.

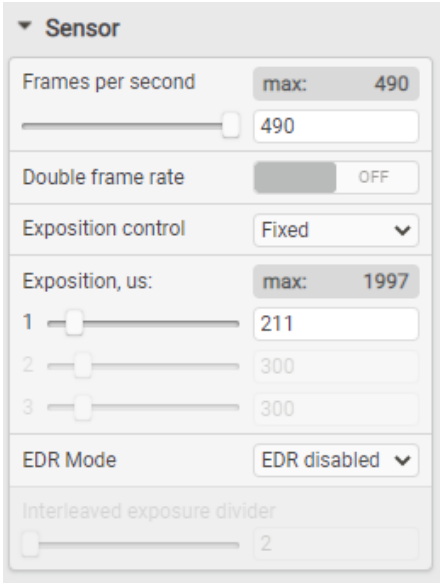
In order for the changes to take effect, it is necessary to click the **Apply** button.

18. "General" tab. General parameters

To configure the general parameters of the scanner, go to the **General** tab:



18.1. CMOS sensor parameters



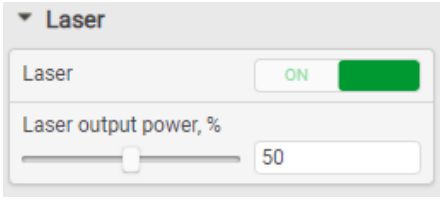
Parameters:

Parameter	Factory value	Description
Frames per second	485	The current number of profiles (frames) per second that the scanner processes and transmits.
Double frame rate (DS mode)	OFF	Enable / disable the double frame rate mode: • **ON** - Enabled, the scanner works in DS mode. • **OFF** - Disabled, the scanner works in standard mode. **Note:** In this mode, the linearity of the scanner for Z is reduced from ±0.05% to ±0.1% of the measuring range for Z.
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_exposureAdjust" 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** – 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	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 DS modes, 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.
EDR Mode	Disabled	Expanding the dynamic range of the CMOS sensor. Possible options: • **EDR disabled** – Disabled.

Parameter	Factory value	Description
		• **Column EDR** – The expansion of the dynamic range is achieved due to different exposure times for the even and odd columns of the CMOS sensor. For odd columns, the exposure time is lower. It is used in the control of complex objects containing areas with different reflectivity. Exposure time reduction is determined by the **Interleaved exposure divider** coefficient. • **Piecewise linear EDR** – The expansion of the dynamic range is achieved through the use of a piecewise linear response of the CMOS sensor.
Interleaved exposure divider	5	Exposure time reduction coefficient for odd columns of the CMOS sensor. It determines how many times the exposure time for odd columns is reduced relative to the main exposure time. It is available only when the **Column EDR** mode is enabled.

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.

18.2. 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%. **Note:** The laser output power is adjustable only in manual mode.

18.3. 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.

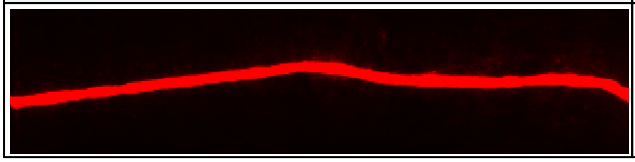
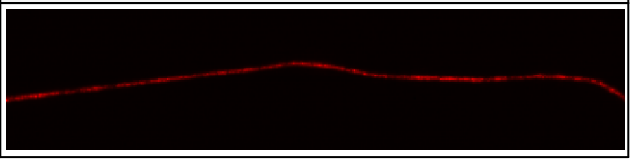
Since the exposure time cannot exceed the frame duration, it is necessary to set the required frame rate (**FPS** parameter) before setting the exposure time.

18.3.1. 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. 15.).

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. 18.5.). 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
	

18.3.2. Multiple exposure mode and EDR mode

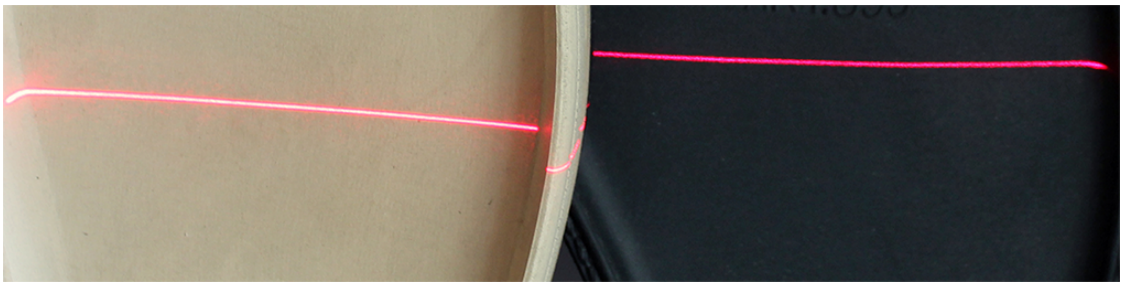
Multiple exposure mode (**Exposition control > 2 exposures, 3 exposures**), as well as **EDR** mode are intended to expand the dynamic range of the scanner. These modes are used when the objects (or the surfaces of one object) located in the field of view of the scanner have different reflective abilities.

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.

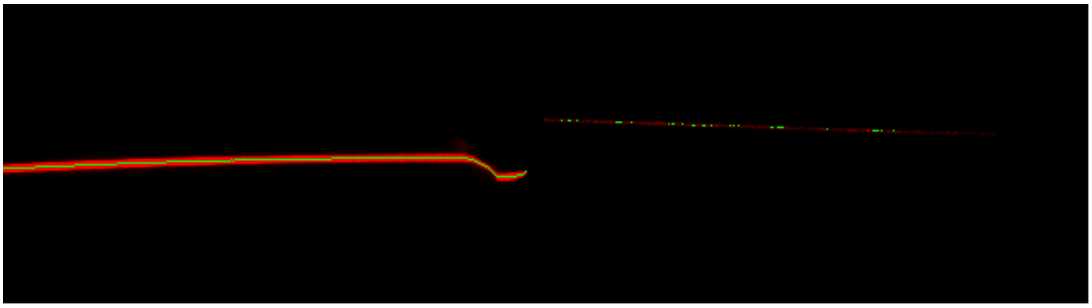
NOTE: In this mode, the frequency of profile output decreases in proportion to the number of exposures.

When using **EDR**, depending on the selected mode, different exposure times are set for even and odd columns or the uneven sensitivity of the CMOS sensor is used. The final profile is formed by combining two profiles. The frequency of profiles does not change.

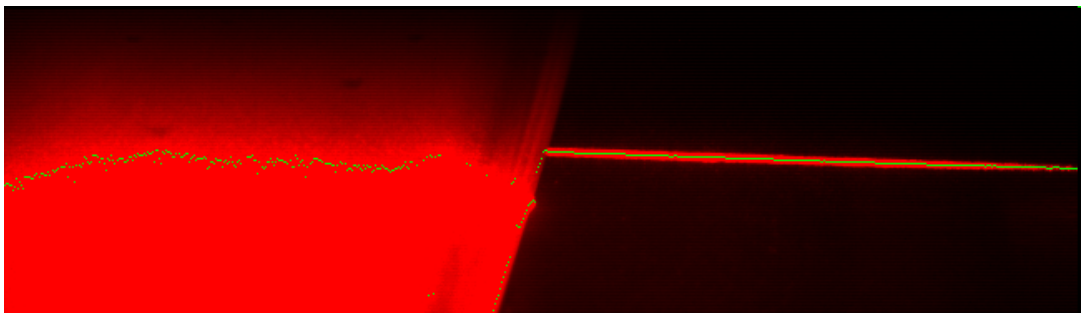
Example:



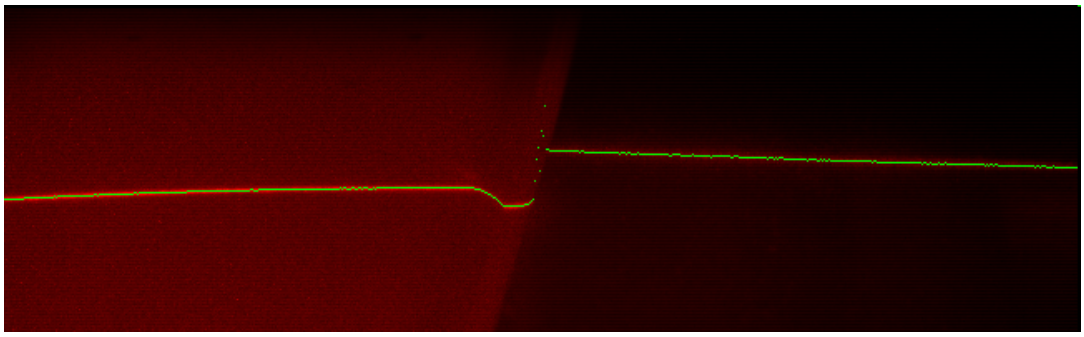
The scanner sees two objects: a light object and a dark object.



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.



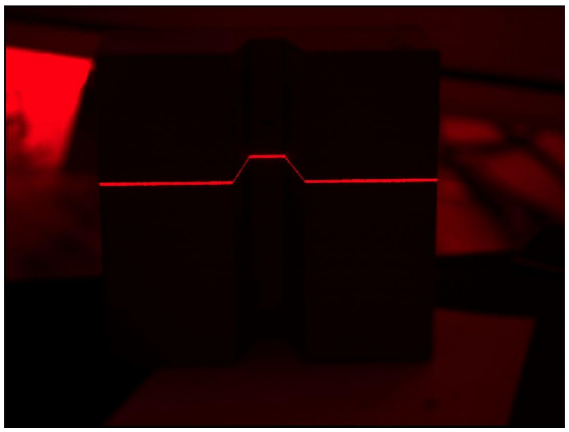
When increasing the exposure time, the profile of the dark object is well-defined, but the profile of the light one is not. You need to record the exposure time for the dark object.

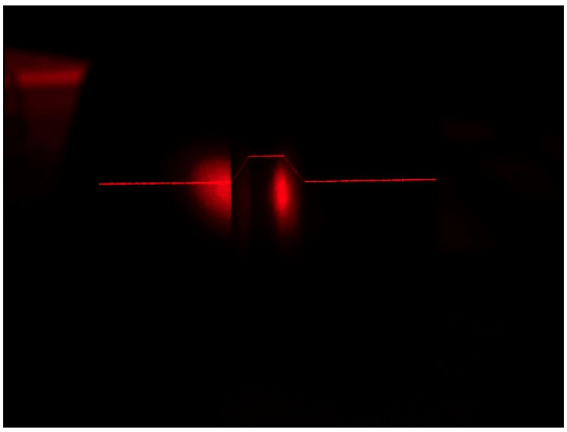


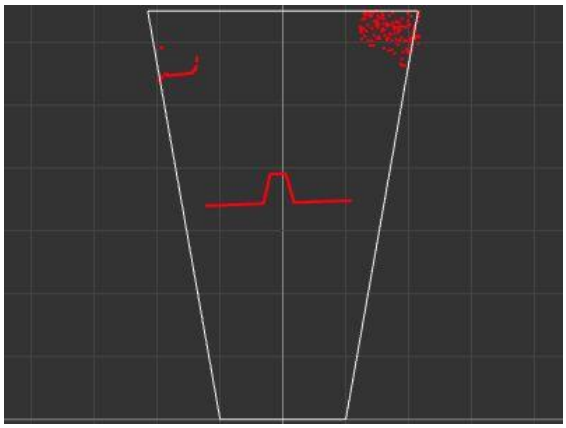
Select the multiple exposure mode and specify the recorded values of the exposure time, or select the EDR mode with the corresponding exposure times for the columns. As a result, you get a high-quality image and a profile of a complex object.

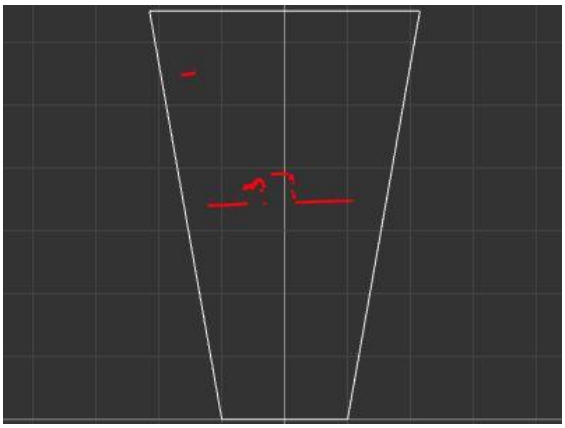
18.3.3. Removing background light from extraneous light sources

This mode is recommended when intense radiation reflected from extraneous sources (sun, lighting, etc.) enters the field of view of the scanner.

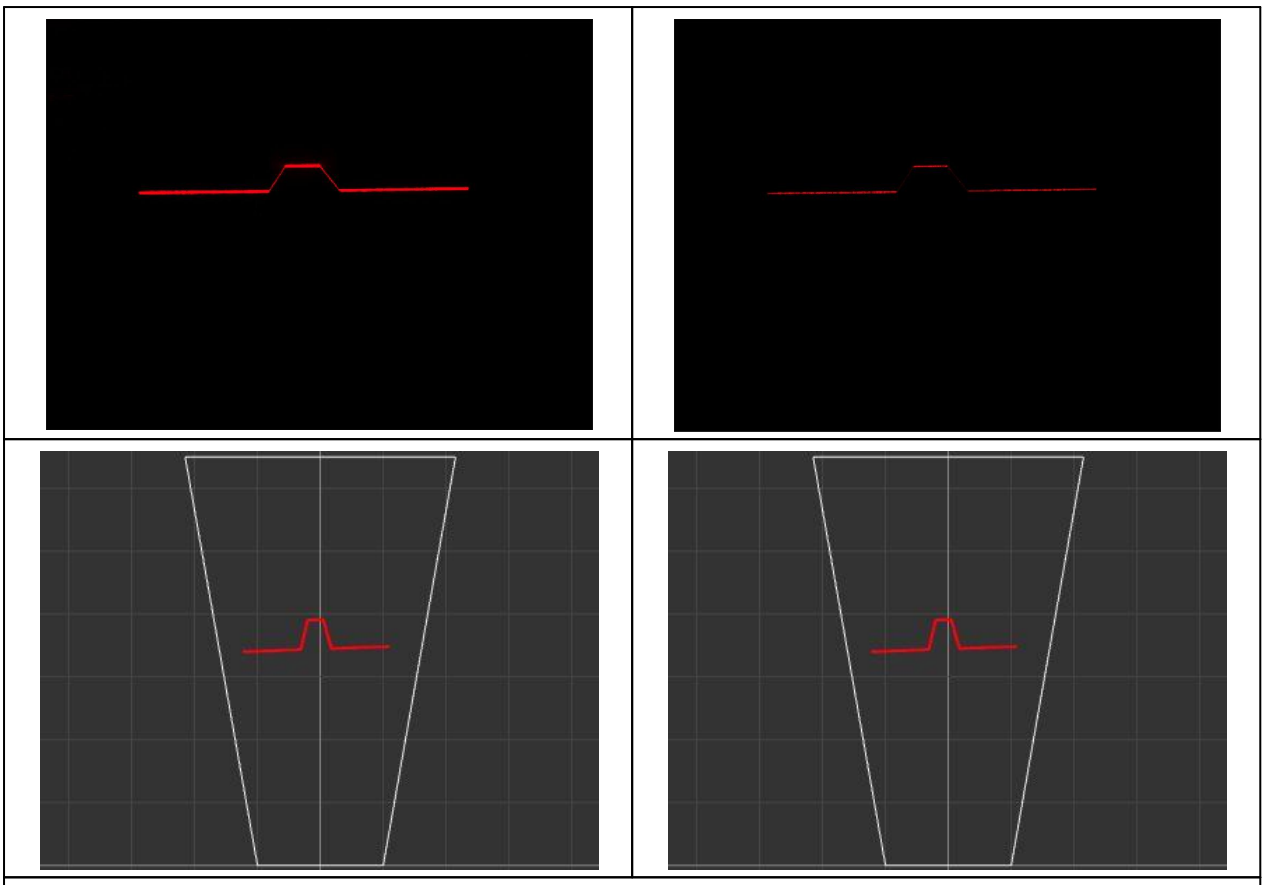








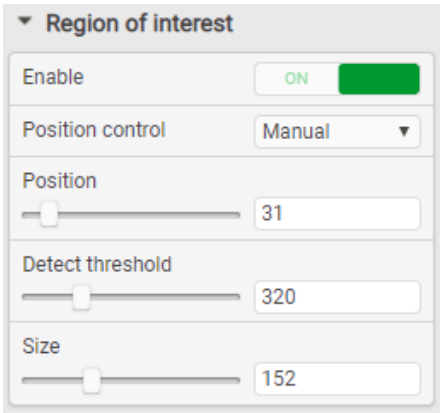
Examples of images and profiles with light and glare on the scanned surface.



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.

18.4. ROI 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):

ROI size, (% of MR) / (lines)	DS mode disabled	DS mode enabled
100% / 488	485	921
65% / 320	720	1335
41% / 200	1100	1965
19% / 96	2029	3325
13% / 64	2741	4223
6.5% / 32	4223	5788
5% / 24	4884	6379

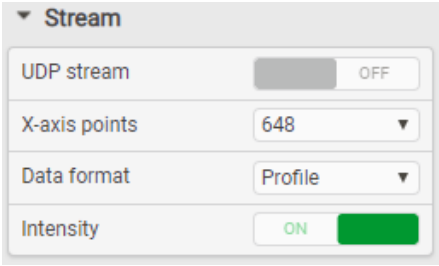
Parameters:

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 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).

18.5. 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.

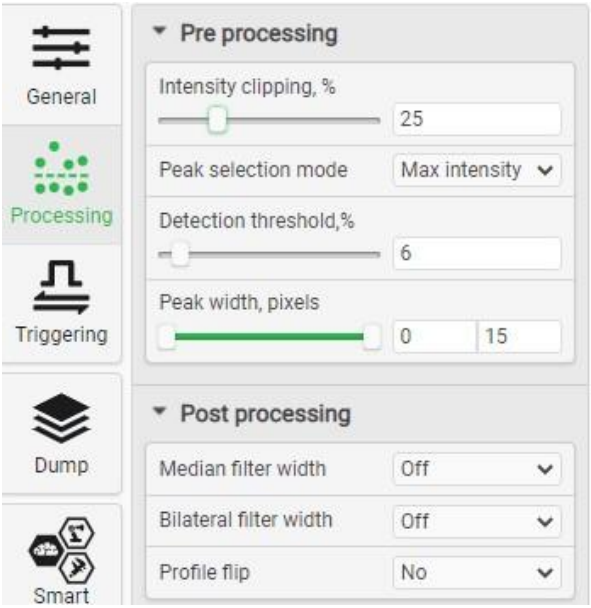


Parameters:

Parameter	Factory value	Description
UDP stream	ON	Enable/disable the UDP data stream.
X-axis points	1296	The number of points along the X coordinate (648 or 1296).
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.

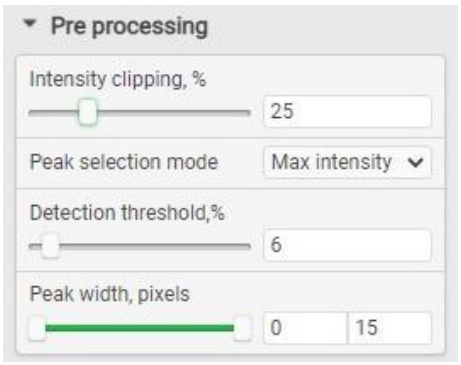
19. "Processing" tab. Profile extraction settings

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).



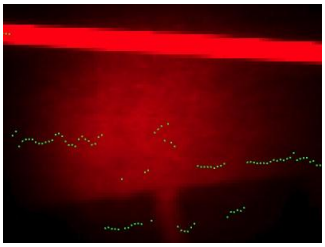
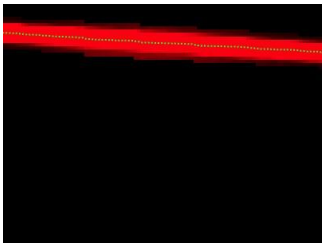
19.1. "Pre processing" section. Profile extraction parameters

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 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.

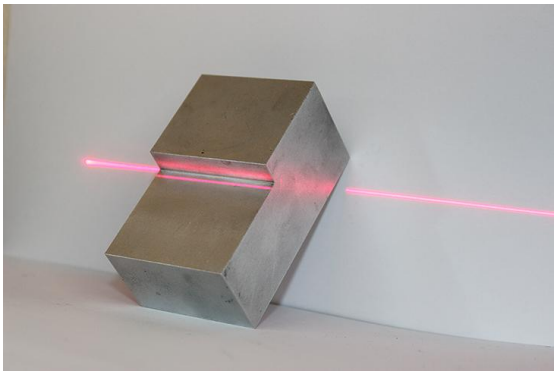
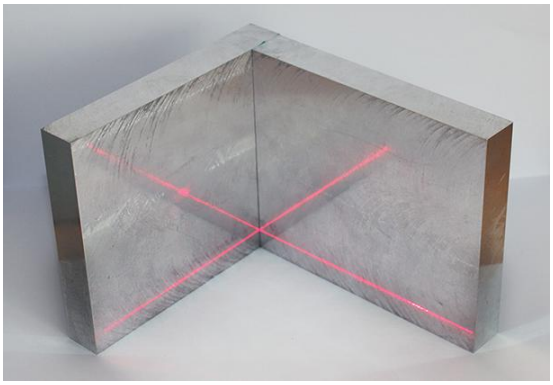
Parameter	Factory value	Description
		 "Intensity clipping" = 4%  "Intensity clipping" = 70%
Peak selection mode (see par. 19.1.1.)	Max intensity	The algorithm for determining the peak brightness in the image column to obtain the profile point. It is used to suppress false images resulting from multiple 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.

19.1.1. "Peak selection mode" parameter

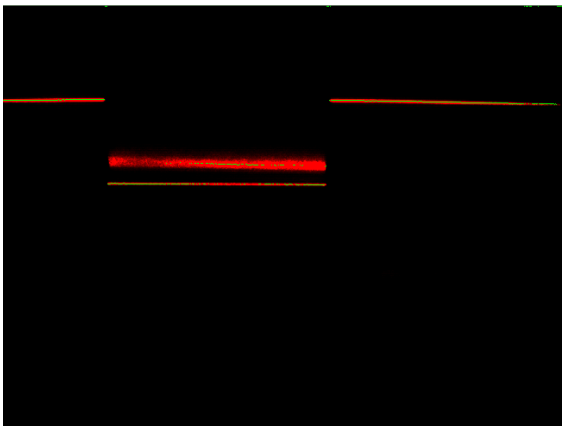
The **Peak selection mode** parameter defines the algorithm for detecting the brightness peak in the column of the CMOS sensor to obtain the profile point. Changing this parameter helps to correctly extract the profile in the case of laser beam re-reflections from the object surface or in the case of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

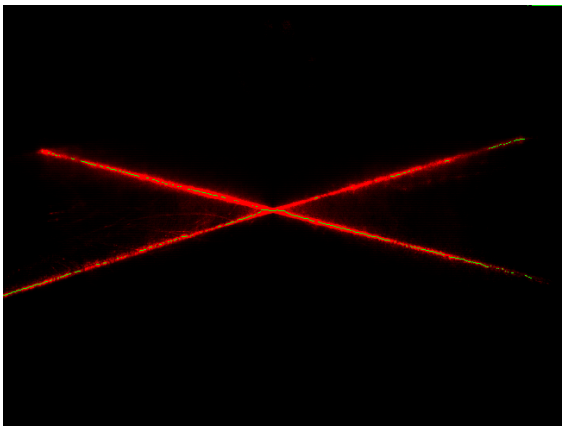
The intensity of the re-reflected beam or brightening from external light sources can sometimes exceed the intensity of the laser line. In this case, you can use the modes with an indication of a more specific detection point.

Example:

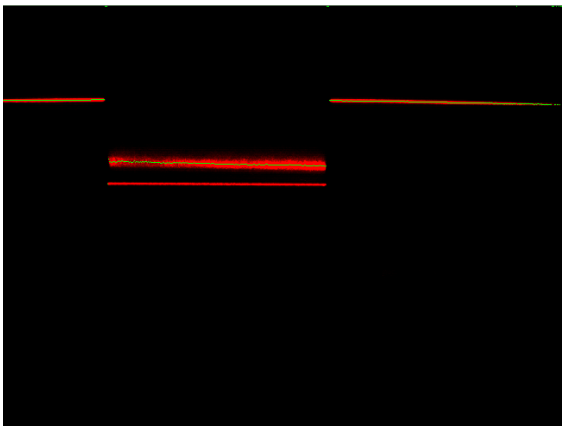



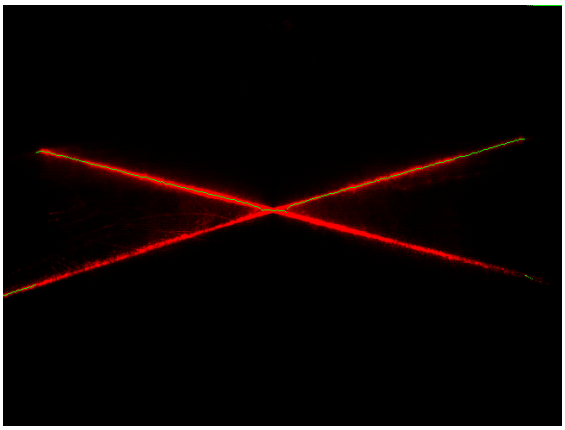
Re-reflections of a laser beam on the object having a complex profile.



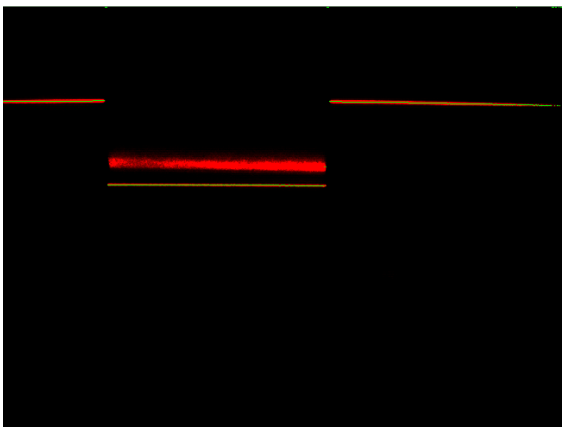


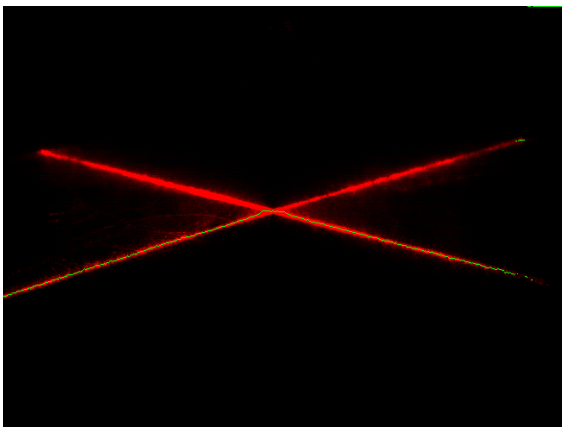
The **Max Intensity** value determines the selection of the profile point based on the maximum brightness of the image in the CMOS sensor column. The brightness of the re-reflected signal may be greater than the brightness of the original signal. The scanner incorrectly selects the profile, placing it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

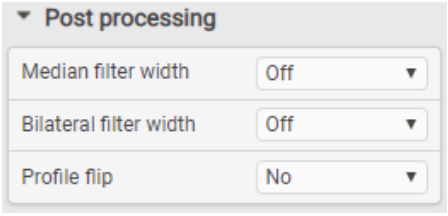




The **Last** value determines the selection of the last peak in the CMOS sensor column. The scanner selects a profile by a real signal.

19.2. "Post processing" section. Filtering

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

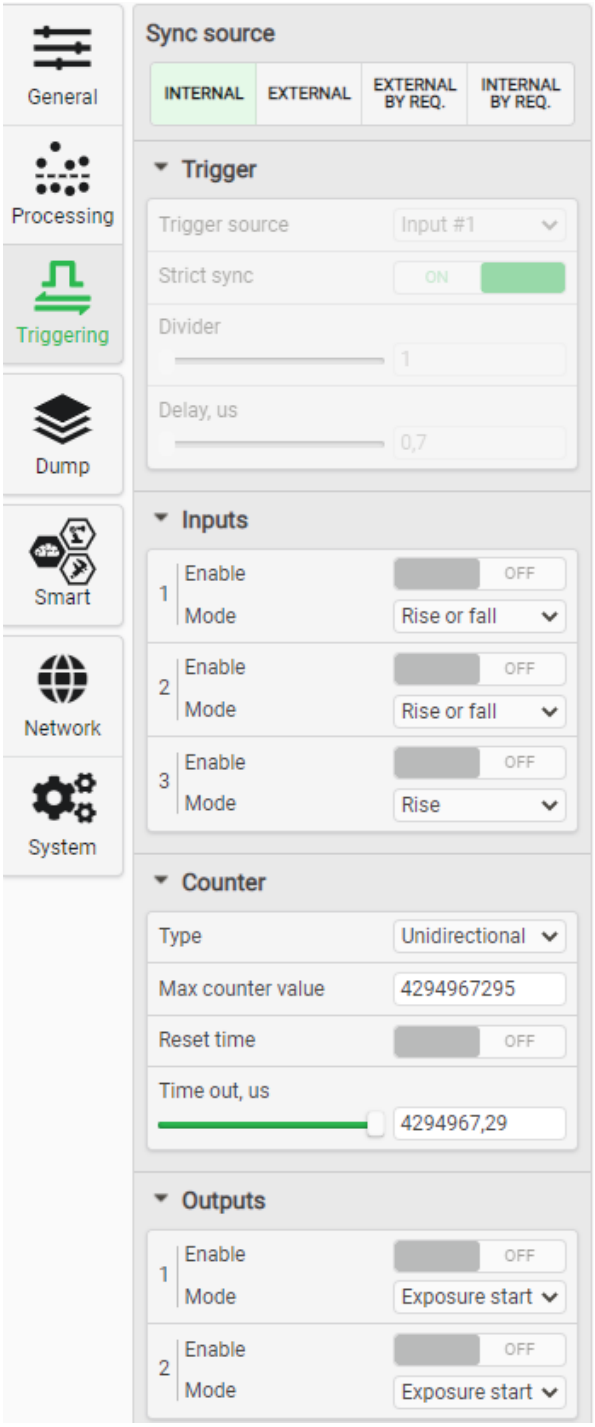


Parameters:

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.

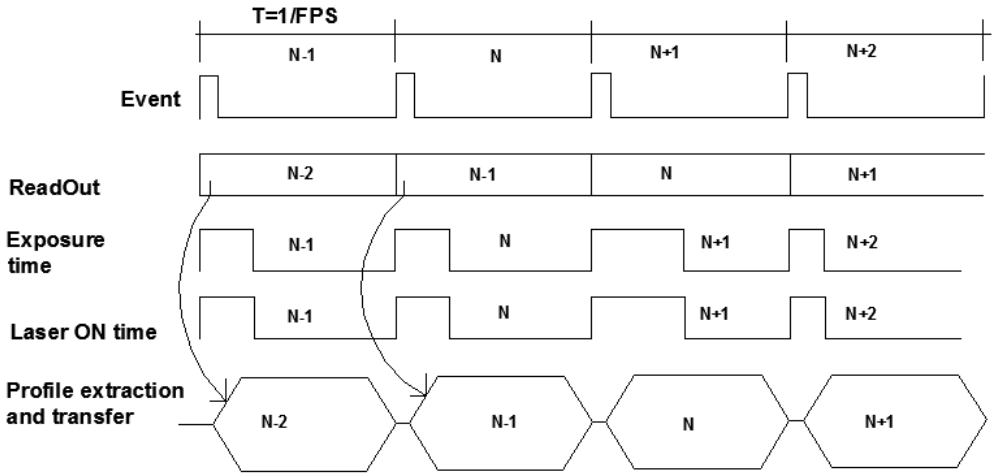
20. "Triggering" tab. Triggering modes

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



20.1. 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:

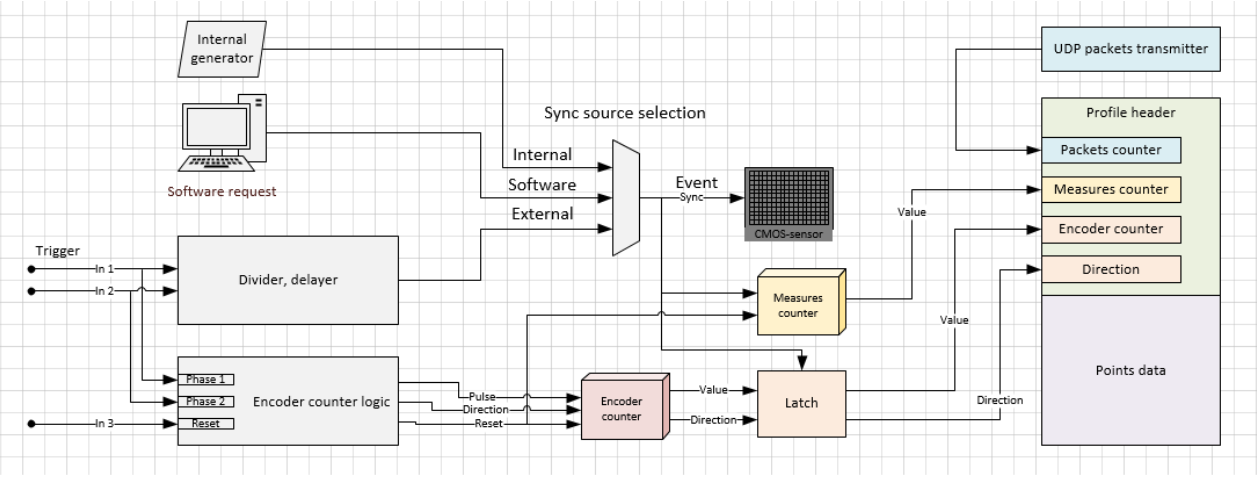
T	Frame (profile) period.
FPS	Frame (profile) rate.
N-1, N...	Frame (profile) numbers.
Event	Event that triggers the measurement cycle of obtaining a single frame (profile).
Exposure time	Exposure time of the image sensor.
Laser ON time	Time during which the laser is turned on.
Profile extraction and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

20.2. 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:

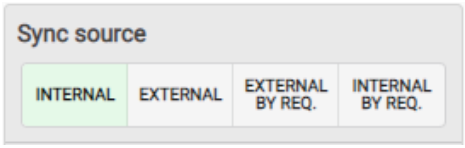
1. The maximum processed frequency at inputs #1, #2 and #3 is 10 MHz. If the event arriving rate is higher than the FPS, the measurement is started at the closest synchronization event after the end of the current cycle. The minimum allowable pulse duration is 40 ns. When using the input divider (**Divider**),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

20.3. Selecting a source of synchronization events

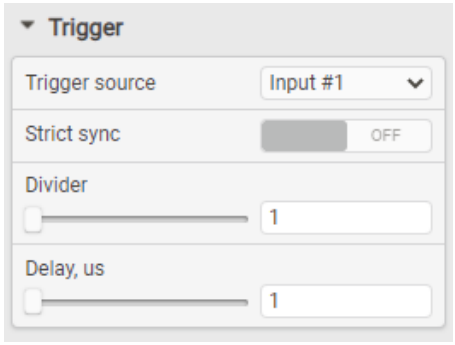
To select the source of synchronization events, use the **Sync source** section of the **Triggering** tab:



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.

20.4. 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.



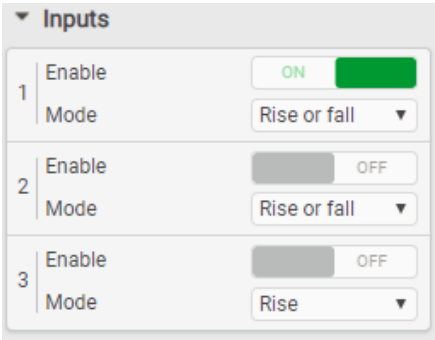
Parameters:

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

Parameter	Factory value	Description
		Divider is "2", it applies for every second signal at the input, etc. Setting the **Divider** parameter allows, for example, to match the 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).

20.4.1. 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.

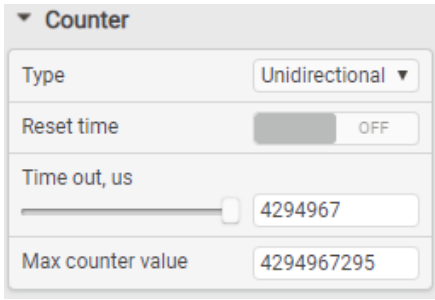
Input #3 of the scanner is designed to connect the reset signal of the measurement counters and the encoder.

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.

20.4.2. 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). • 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.

20.4.3. Examples

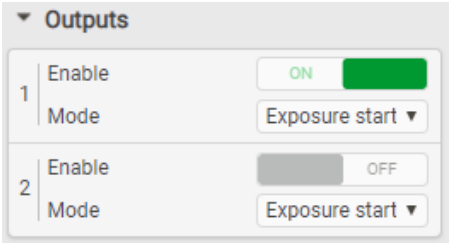
Examples of trigger settings:

#	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.

#	Event source	How it works	Options	How to install
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, taking into account the set division ratio. The direction indication (Dir) is transmitted in the data packet.	• Triggering the measurement on the rise of the pulse. • Triggering the measurement on the fall of the pulse. • Divider.	• Connect the Step signal to input #1. • 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**.

20.5. 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.

21. "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).

21.1. Synchronous measurements

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

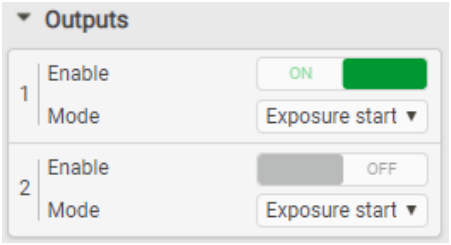
NOTE: The mode number is the line number in the table of trigger settings (see par. 20.4.3.).

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.

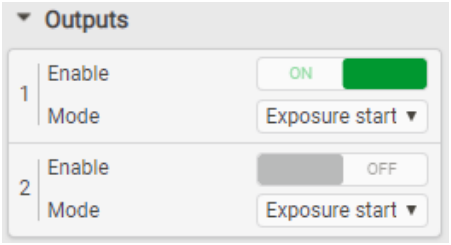


- The other scanners (Slave) are switched to mode #3, the operating mode is "Rise".
- The Master output is connected to Input #1 of all Slave scanners.

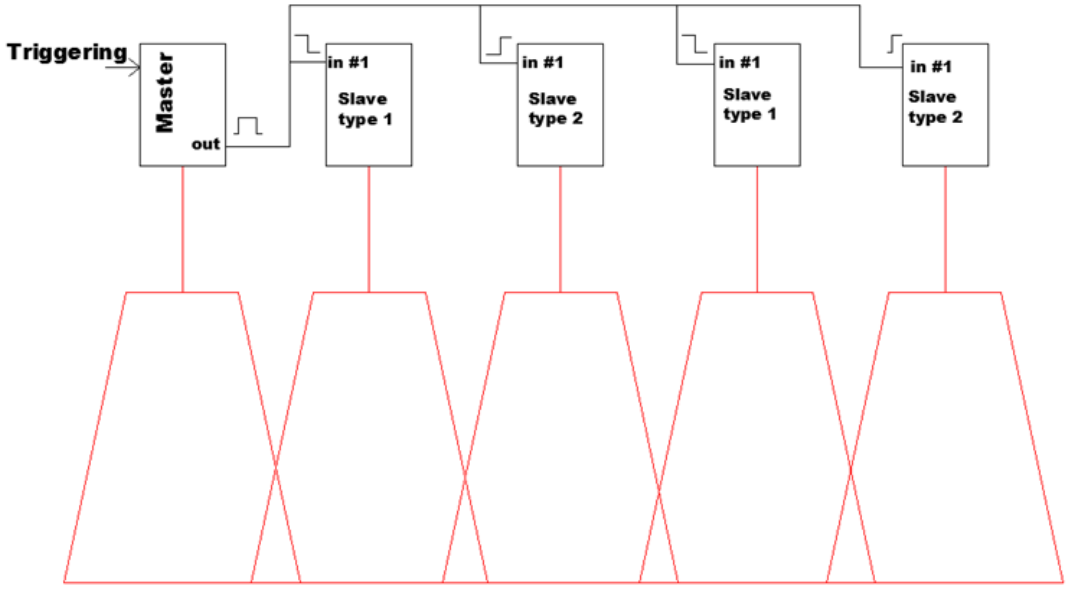
21.2. Asynchronous measurements

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.

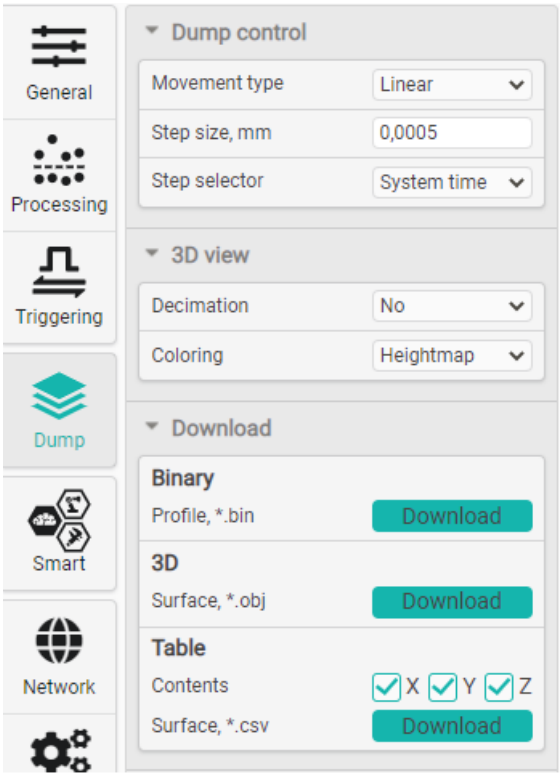


As a result, the lasers of the scanners of the "Master + Slave type 2" group and the "Slave type 1" group will alternately turn on.

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

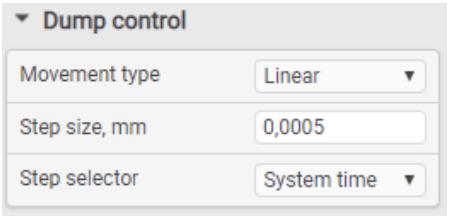
22. "Dump" tab. Accumulated profiles parameters

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



22.1. "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
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.

Parameter	Factory value	Description
		• **Step counter** – Encoder counter. • **Measurement counter** – Internal measurement counter.

22.2. "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. **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**).

22.3. "Download" section. Downloading profiles

This section is intended for saving profiles in various formats. For more details, go to par. 22.4.3 "Export of accumulated profiles".

22.4. Operations with profiles

22.4.1. 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. 20).

22.4.2. 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. 22.1.):

- 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.

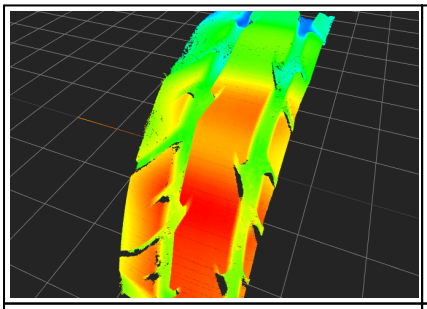
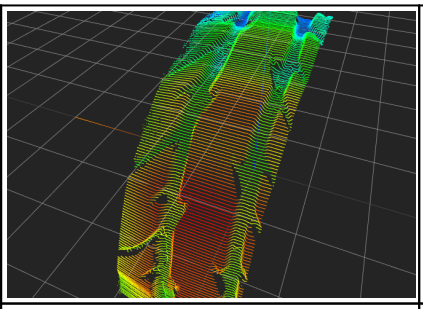
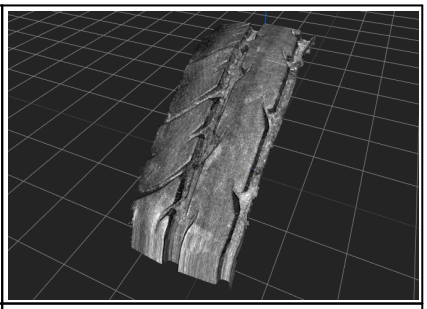
After configuring the display parameters, it is necessary to click the refresh button  . After that, the data will be downloaded from the scanner and a point cloud will appear.

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. 18.5.). Otherwise, the intensity of all points will be zero (black color).

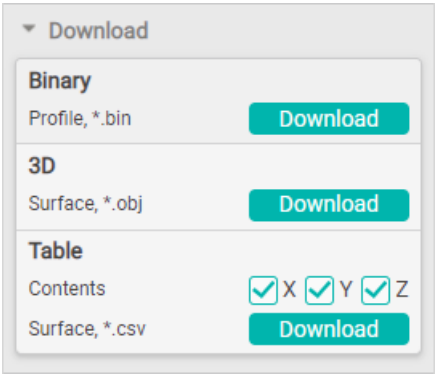
		
Point cloud without decimation and with coloring by height	Point cloud (decimation coefficient is set to 5)	Point cloud with coloring by intensity

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.

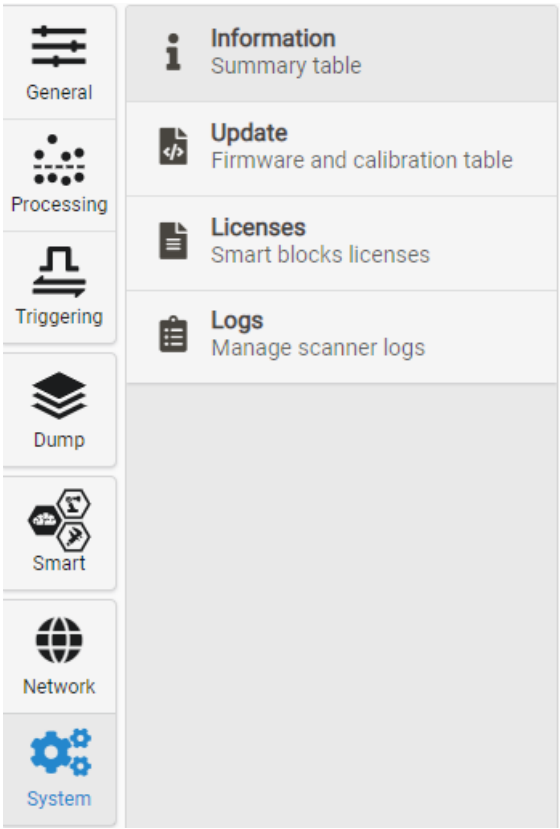
22.4.3. Export of accumulated profiles

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. "System" tab



23.1. "Information" section

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

Device information	
Work	64h 19m 09s
Total	1585h 23m 33s
CMOS-sensor temperature, °C	47.9
Name	2D laser scanner
Model	Laser scanner
Serial	5509356
Wavelength	650 nm
Firmware version	2.4.0-beta7
Hardware version	18.6.20.0
Calibration date and time	1.1.1970 3:0:0 (UTC +3)
Working ranges	
Base Z (SMR)	80 mm
Range Z (MR)	130 mm
Range X Start (XSMR)	40 mm
Range X End (XEMR)	86 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**.

23.2. "Update" section

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

23.2.1. 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.

Firmware		
Element type	Version	CRC
Files		
fpga.bin	2.1.2	OK
cpu0.bin	2.1.2	OK
Sectors		
fsbl_recovery	2.1.2	ERROR
Choose file	629_2_1_2_fake.2fw	Upload Save

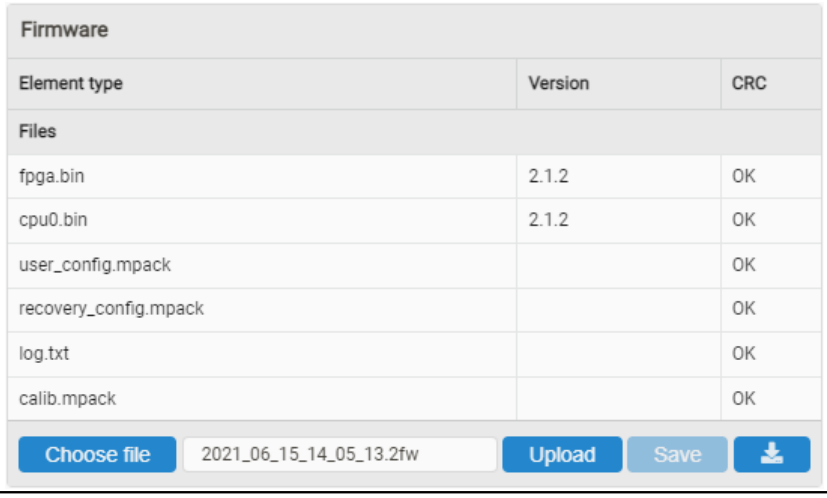
Checksum error

Firmware		
Element type	Version	CRC
Files		
fpga.bin	2.1.2	OK
cpu0.bin	2.1.2	OK
Sectors		
fsbl_recovery	2.1.2	OK
Choose file	629_2_1_2.2fw	Upload Save

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.

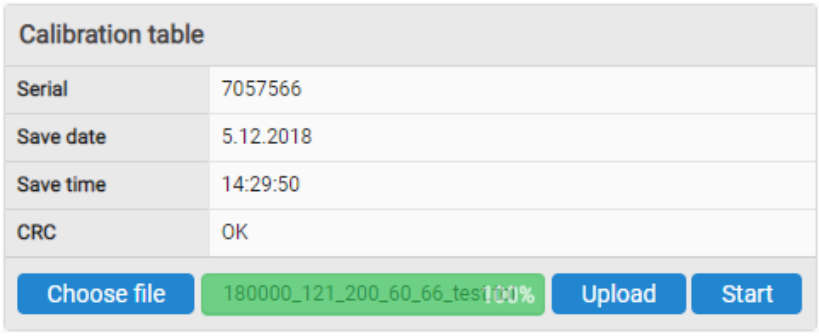


Recovering the saved firmware

23.2.2. Updating the calibration table

Procedure:

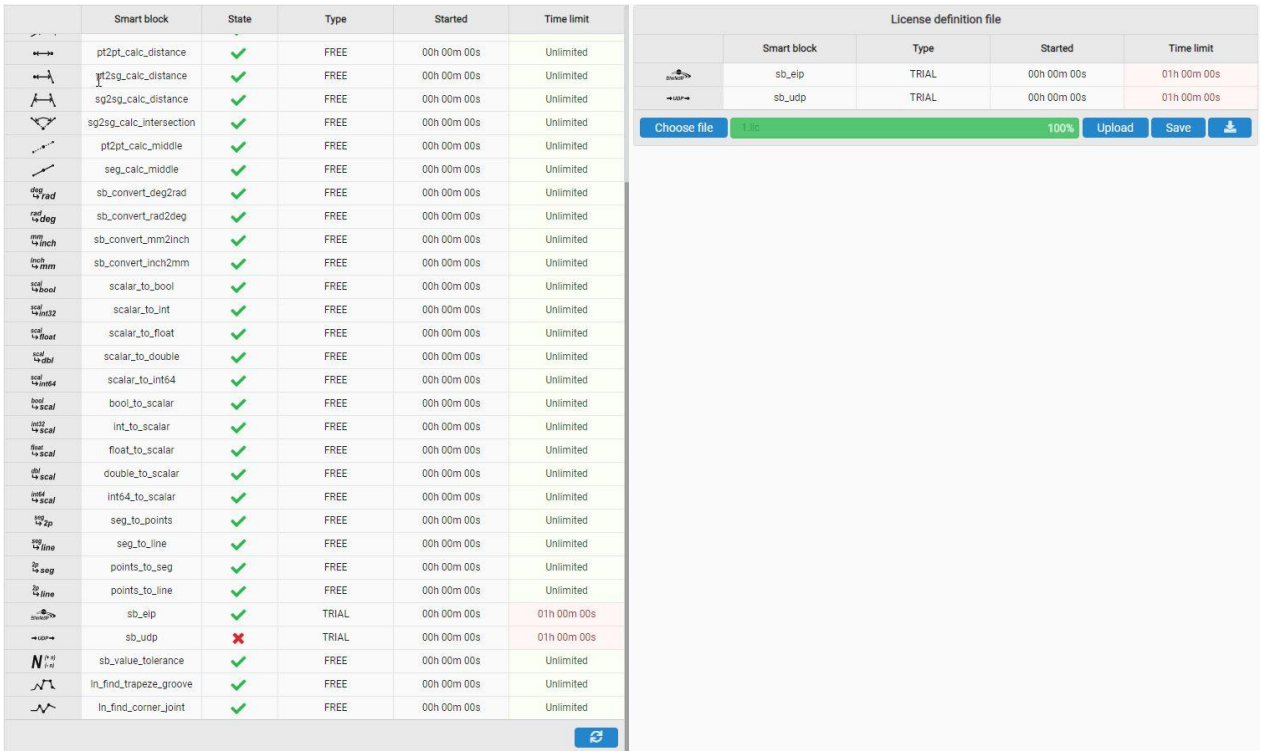
1) Click **Choose File** and select the calibration table file.
2) Click **Upload** to upload the selected file.



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

23.3. "Licenses" section

The **Licenses** section is designed to display the status and manage licenses for "smart blocks" in the **Smart** tab.

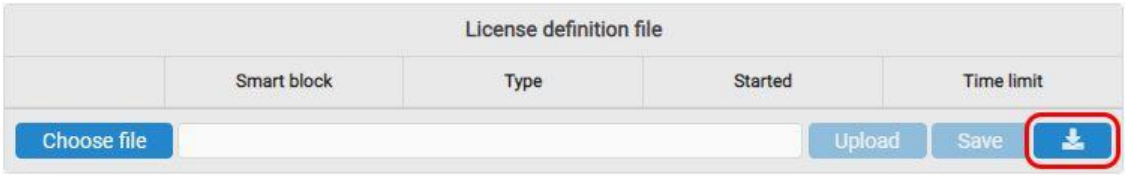


On the left side of the section, you can see a list of blocks available in the scanner firmware and the license status for each block:

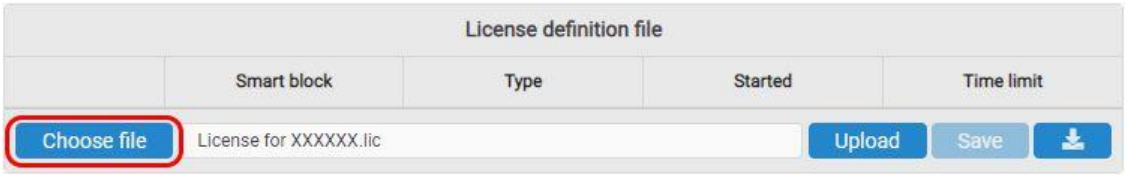
Parameter	Description	
State	The current license status for the block.	
	✓	All block functions are available. The following license types can have this status: • FREE - the block does not require a license; • TRIAL - the trial period has not expired; • FULL - the block license was purchased.
	✗	The block does not function, or has limited functionality. The following license types can have this status: • TRIAL - trial period expired.
Type	License type.	
	FREE	The block does not require a license. All block functions are available.
	TRIAL	The block has a trial period, after which the functionality is limited. The trial period starts from the moment the block is placed on the computation graph and continues for the time specified by the "Time limit" parameter.
	FULL	The block license was purchased.
Started	The running time when the block was first placed on the computation graph. A trial period starts from this time.	
Time limit	The trial period, after which the functionality of the block is limited or the block stops working.	

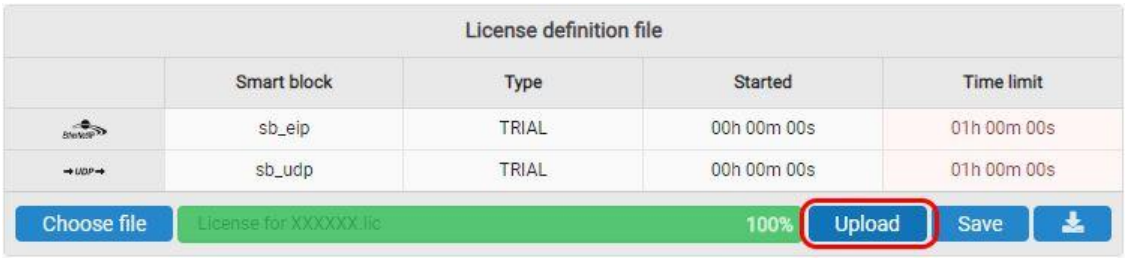
How to purchase a license for a smart block:

1. Download the license definition file from the scanner (**Download** button):



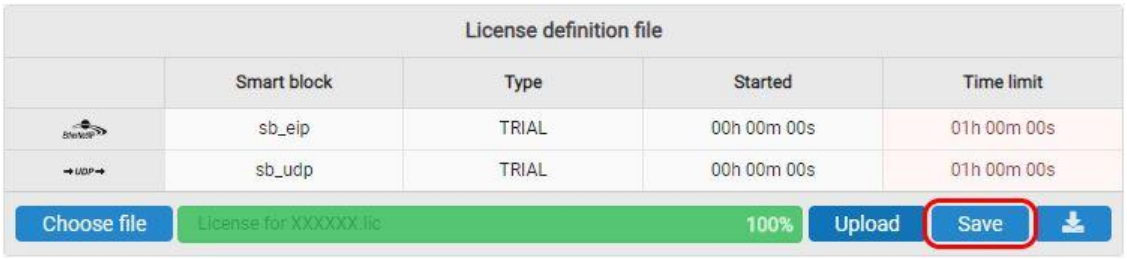
2. Send the downloaded file to the scanner manufacturer, pay for the license. After confirming payment, the manufacturer will send you a file, which must be uploaded to the scanner (it will be necessary to select the file by clicking the **Choose file** button, and then upload it to the scanner using the **Upload** button):





3. After uploading the file to the scanner, an internal check will be performed and the results will be displayed. In the screenshot above, the file was verified successfully and contains data on the trial period extension (license type: TRIAL, the start time has been reset, the new duration is 1 hour). It is necessary to make sure that the received file contains purchased licenses. In case of any discrepancies, contact the technical support service of the scanner manufacturer.

4. Save the license to the internal memory of the scanner by clicking the **Save** button:

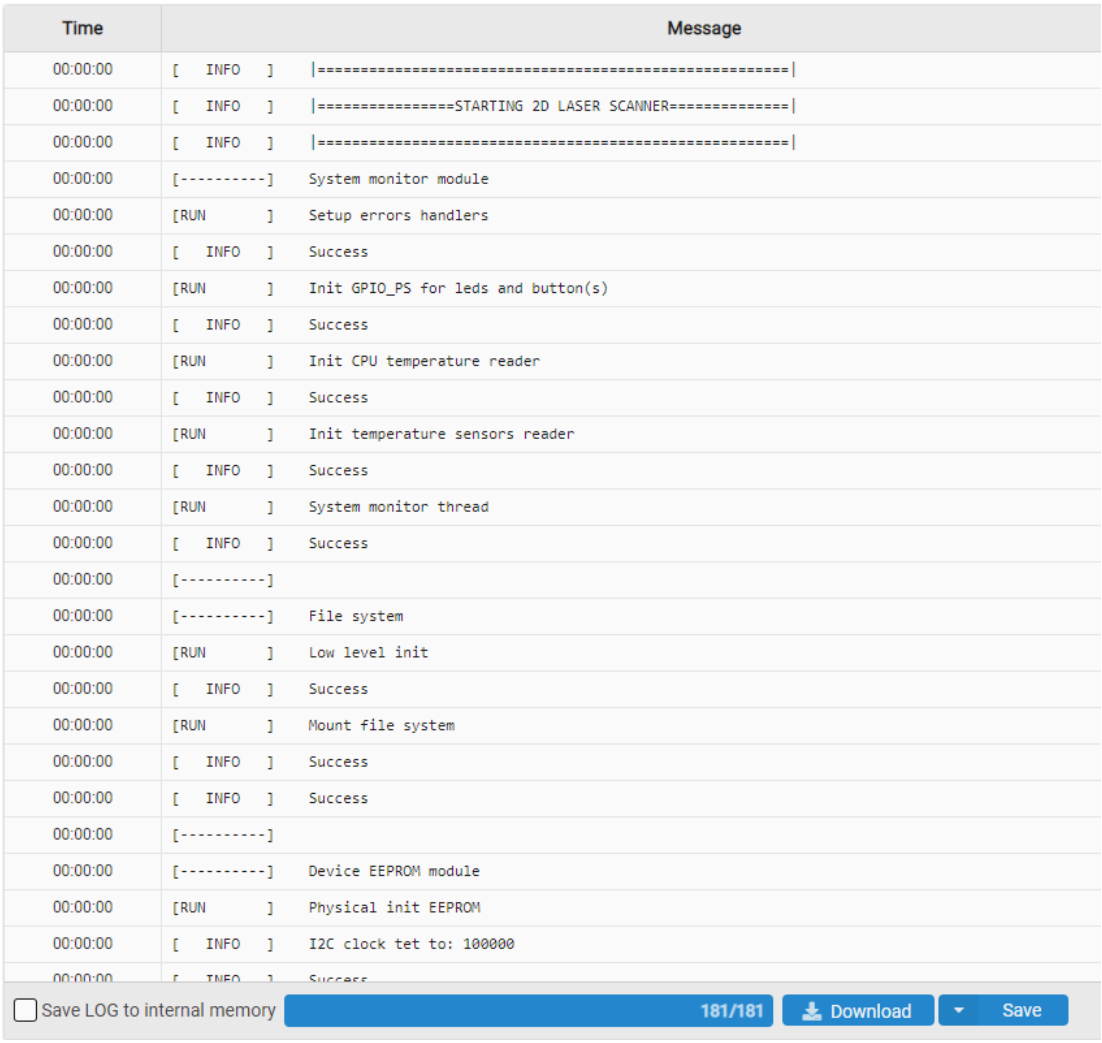


5. After saving the license, it is necessary to make sure that the changes are correct:

	Smart block	State	Type	Started	Time limit
	pt2pt_calc_distance	✓	FREE	00h 00m 00s	Unlimited
	pt2sg_calc_distance	✓	FREE	00h 00m 00s	Unlimited
	sg2sg_calc_distance	✓	FREE	00h 00m 00s	Unlimited
	sg2sg_calc_intersection	✓	FREE	00h 00m 00s	Unlimited
	pt2pt_calc_middle	✓	FREE	00h 00m 00s	Unlimited
	seg_calc_middle	✓	FREE	00h 00m 00s	Unlimited
deg → rad	sb_convert_deg2rad	✓	FREE	00h 00m 00s	Unlimited
rad → deg	sb_convert_rad2deg	✓	FREE	00h 00m 00s	Unlimited
mm → inch	sb_convert_mm2inch	✓	FREE	00h 00m 00s	Unlimited
inch → mm	sb_convert_inch2mm	✓	FREE	00h 00m 00s	Unlimited
scal → bool	scalar_to_bool	✓	FREE	00h 00m 00s	Unlimited
scal → int32	scalar_to_int	✓	FREE	00h 00m 00s	Unlimited
scal → float	scalar_to_float	✓	FREE	00h 00m 00s	Unlimited
scal → dbl	scalar_to_double	✓	FREE	00h 00m 00s	Unlimited
scal → int64	scalar_to_int64	✓	FREE	00h 00m 00s	Unlimited
bool → scal	bool_to_scalar	✓	FREE	00h 00m 00s	Unlimited
int32 → scal	int_to_scalar	✓	FREE	00h 00m 00s	Unlimited
float → scal	float_to_scalar	✓	FREE	00h 00m 00s	Unlimited
dbl → scal	double_to_scalar	✓	FREE	00h 00m 00s	Unlimited
int64 → scal	int64_to_scalar	✓	FREE	00h 00m 00s	Unlimited
seg → 2p	seg_to_points	✓	FREE	00h 00m 00s	Unlimited
seg → line	seg_to_line	✓	FREE	00h 00m 00s	Unlimited
2p → seg	points_to_seg	✓	FREE	00h 00m 00s	Unlimited
2p → line	points_to_line	✓	FREE	00h 00m 00s	Unlimited
EtherNet/IP	sb_eip	✓	TRIAL	00h 00m 00s	01h 00m 00s
UDP	sb_udp	✓	TRIAL	00h 00m 00s	01h 00m 00s
N (+n) (-n)	sb_value_tolerance	✓	FREE	00h 00m 00s	Unlimited
	ln_find_trapeze_groove	✓	FREE	00h 00m 00s	Unlimited
	ln_find_corner_joint	✓	FREE	00h 00m 00s	Unlimited

23.4. "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

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.

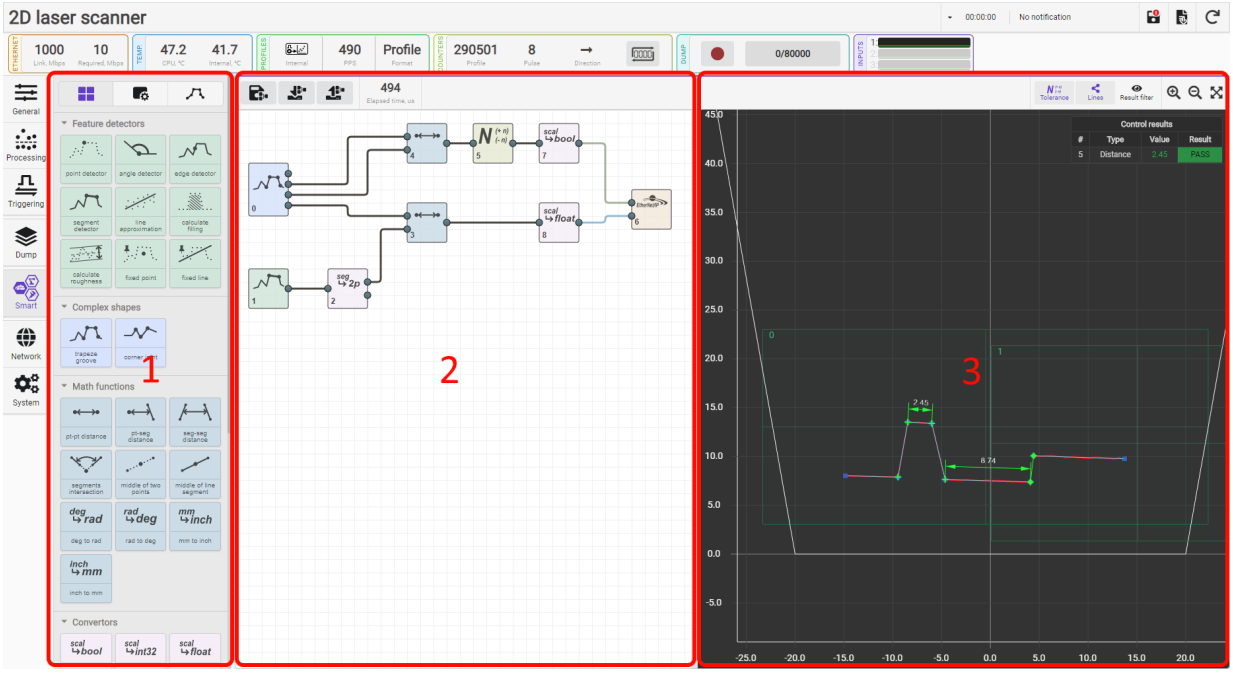
24. "Smart" tab

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:



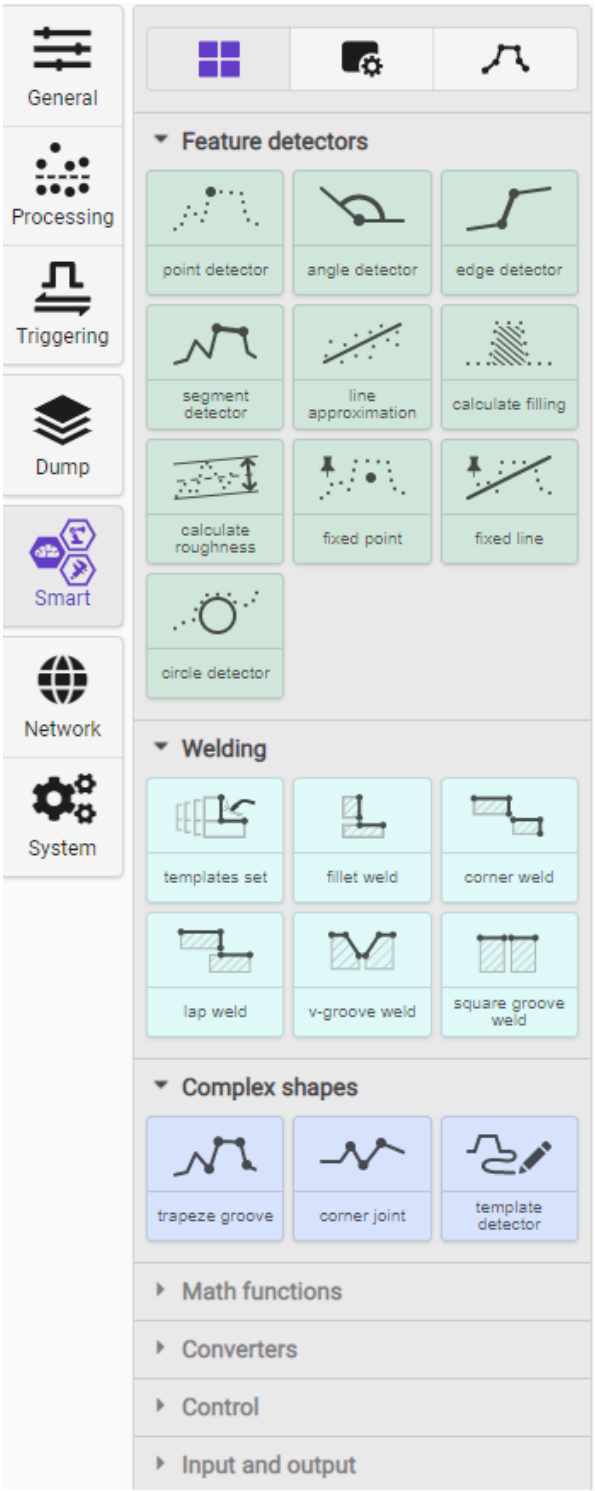
where:
1 - smart blocks and parameters area;
2 - graph creation area;
3 - measurement results display area.

24.1. 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.



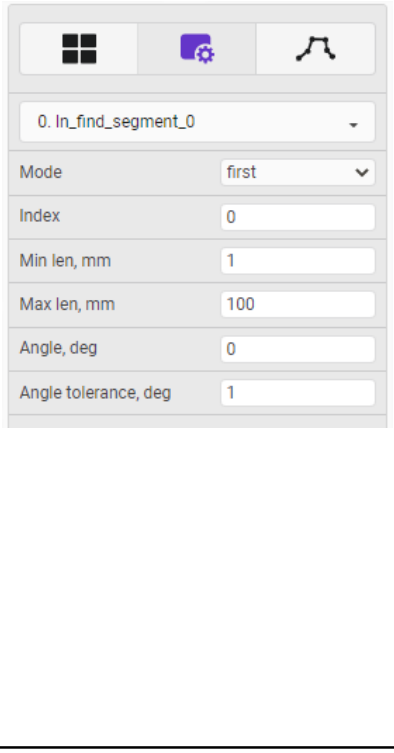
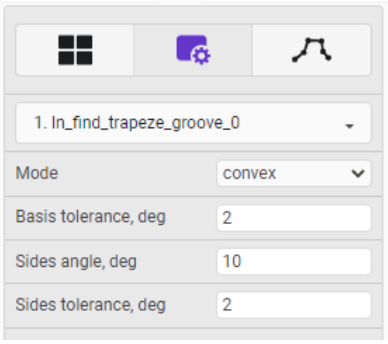
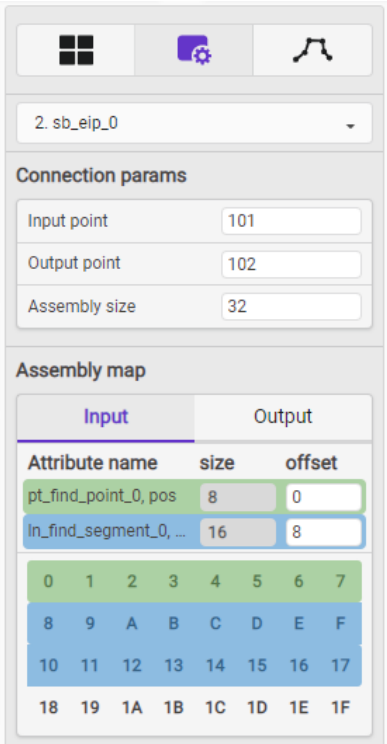
24.1.1. "Smart Blocks" tab

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:

		
the smart block outputs a point	**the smart block outputs a segment**	**the smart block outputs a line**

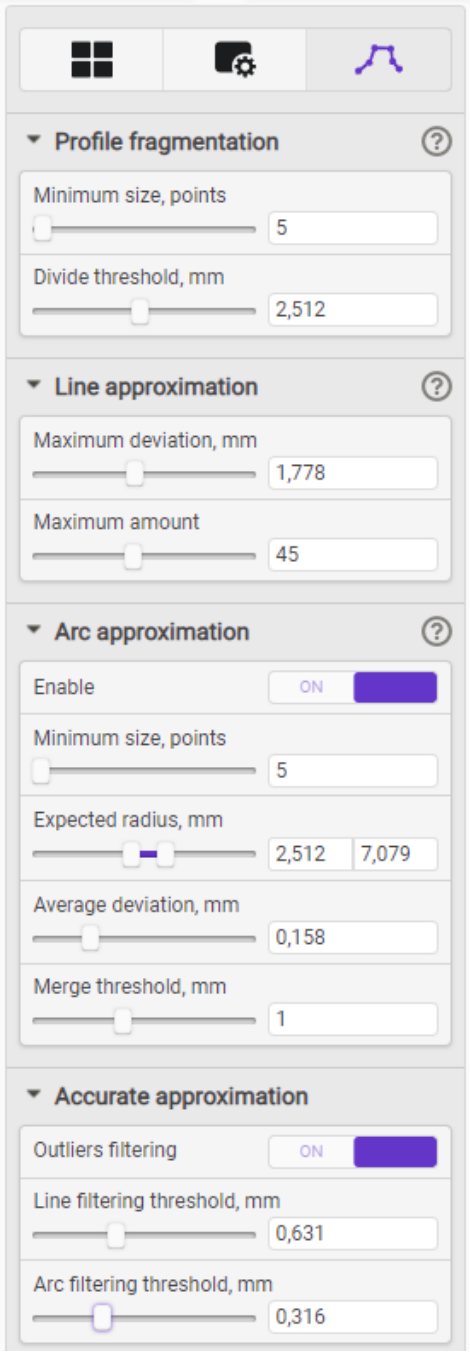
24.1.2. "Block Settings" tab

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

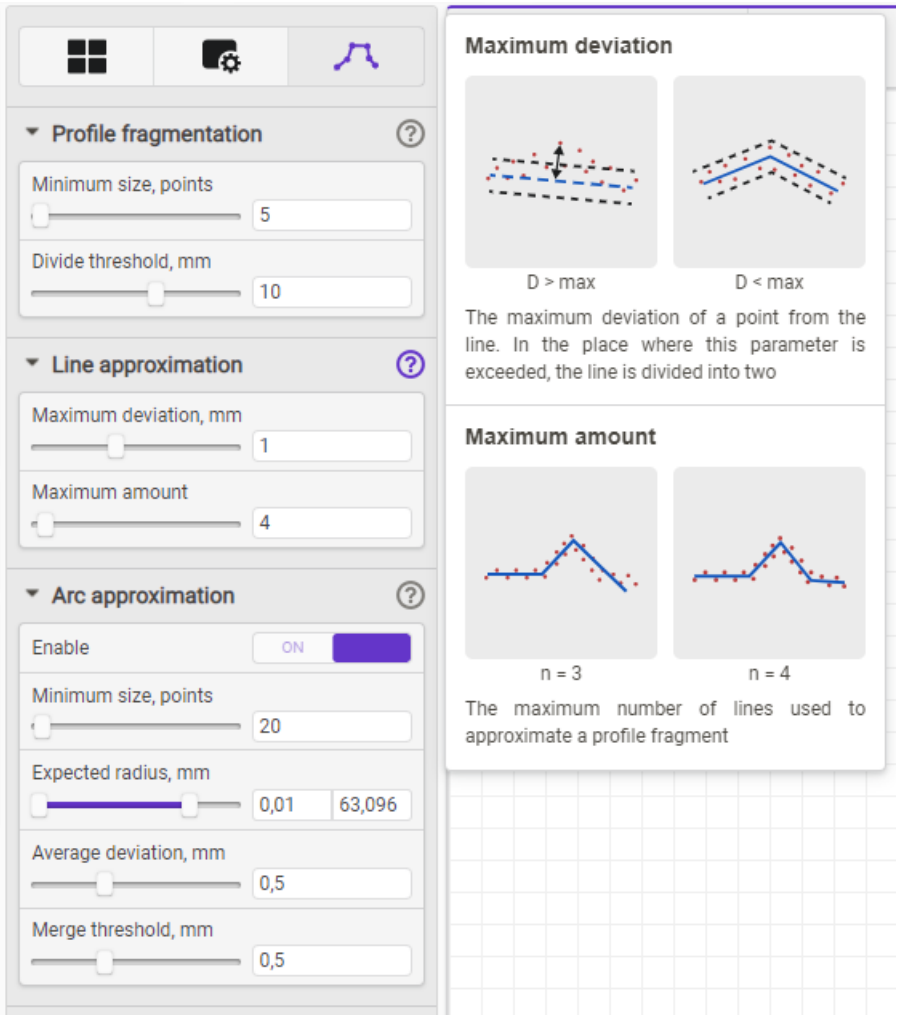
		
“segment detector” smart block settings	**“trapeze groove” smart block settings**	**“eip” smart block settings**

24.1.3. "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.



24.2. Creating a Smart function

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

24.2.1. 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 18.

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.

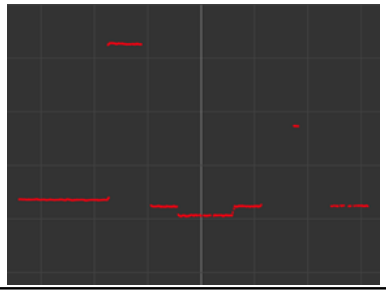
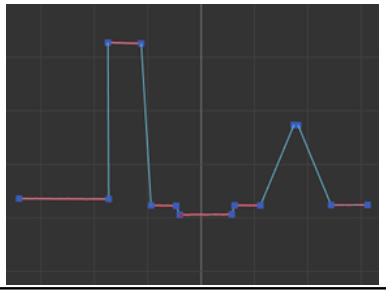
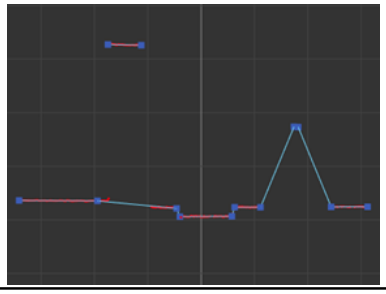
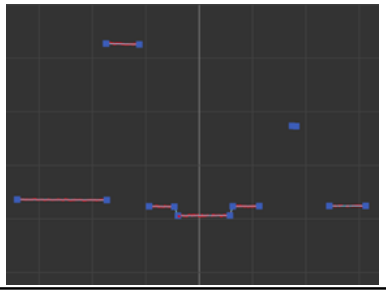
24.2.1.1. Splitting profile points into fragments

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.

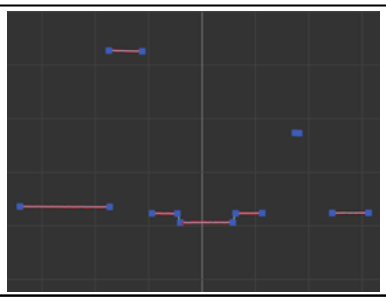
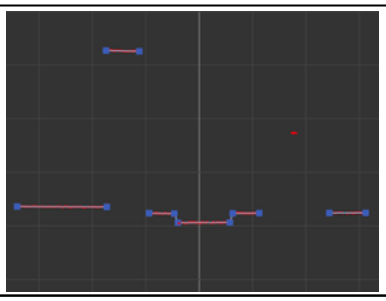
The following parameters from **Profile segmentation** section affect the splitting results:

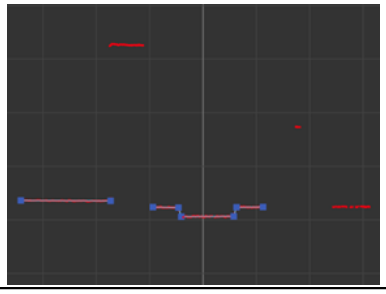
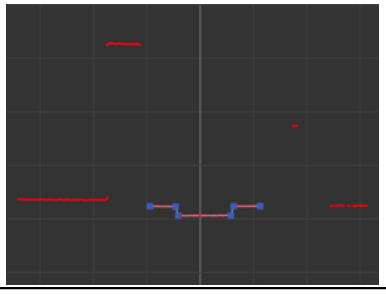
- **Minimum size, points** - the minimum required number of points in a profile fragment for its participation in further processing.
- **Divide threshold, mm** - the minimum distance in millimeters between two successive profile points (left to right) required to combine these points into a single fragment.

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:

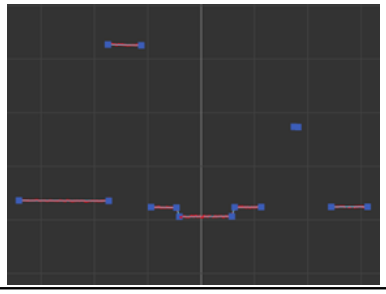
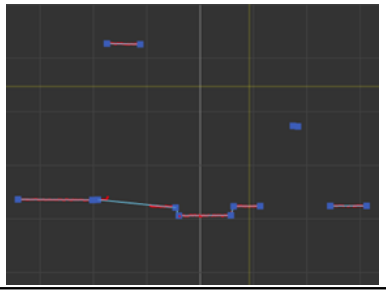
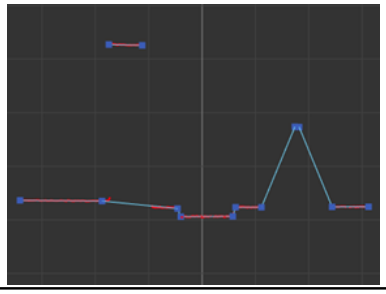
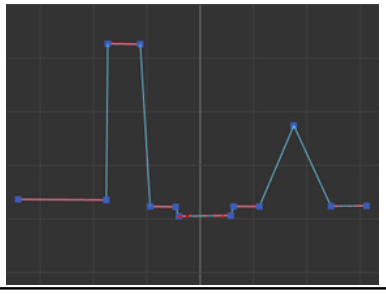
	
Source profile	The profile is split into 1 fragments - all segments are connected
	
The profile is split into 2 fragments - the contour from above consists of one segment, and the contour from below consists of 10 segments (the shortest one forms the apex of the triangle)	The profile is split into 5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

24.2.1.2. 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.

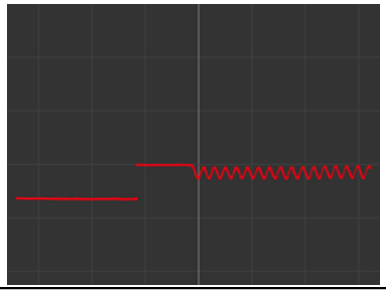
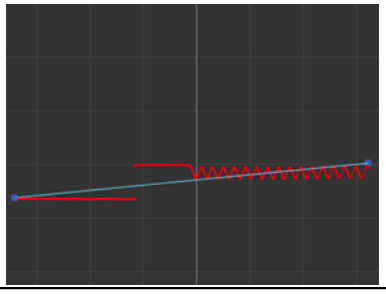
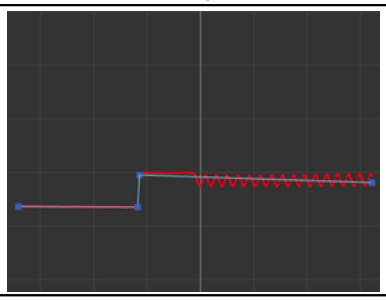
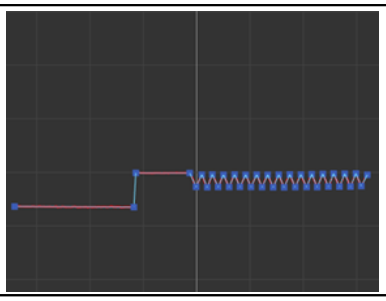
24.2.1.2.1. Approximation by segments

Approximation parameters:

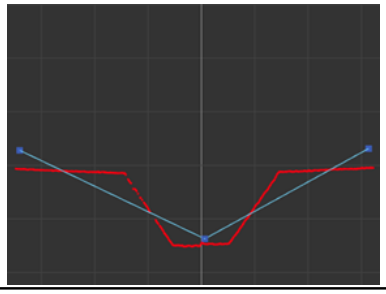
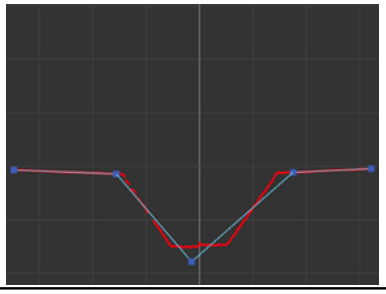
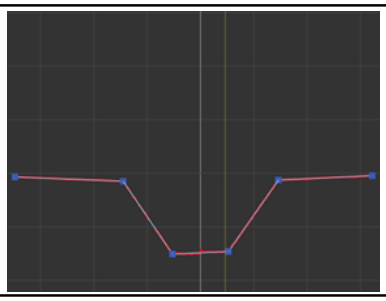
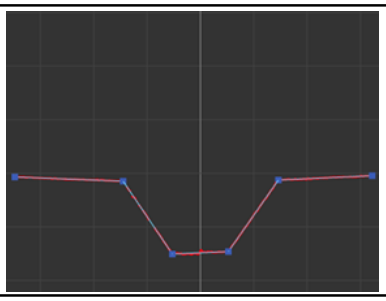
- **Maximum deviation, mm** - the maximum permissible distance in mm between the profile point and the approximating segment. If the next point does not meet this criterion, a new segment is formed. Thus, this parameter affects the approximation granularity.
- **Maximum amount** - the maximum permissible 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

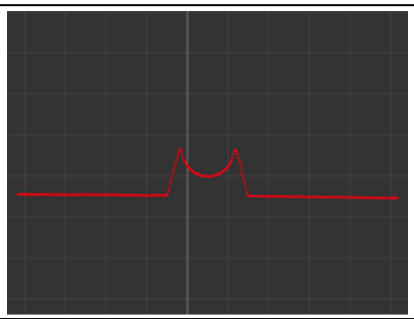
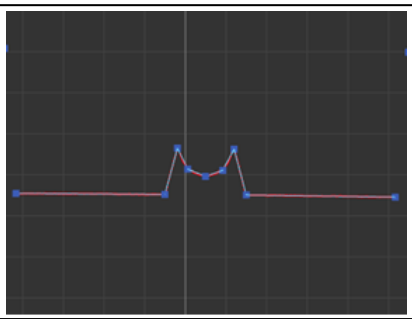
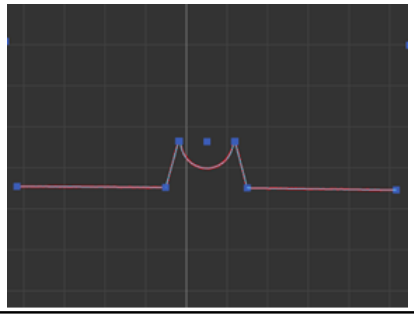
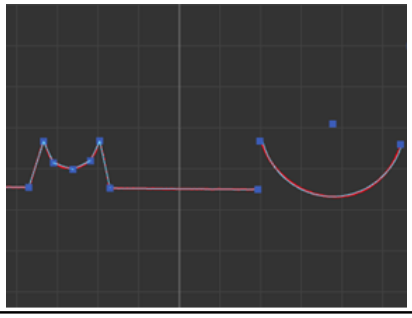
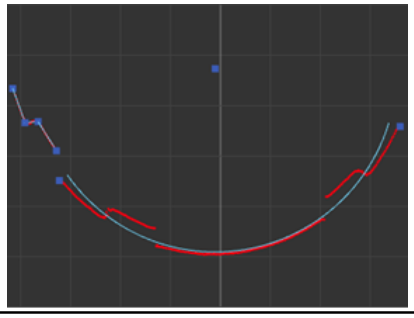
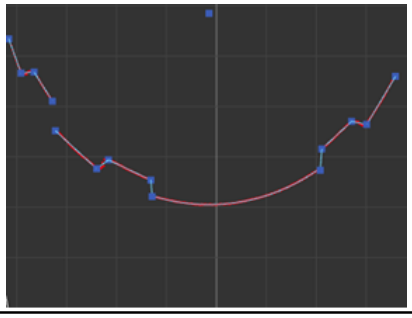
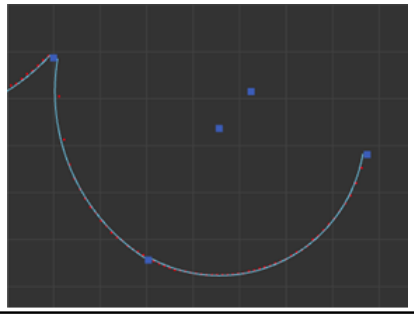
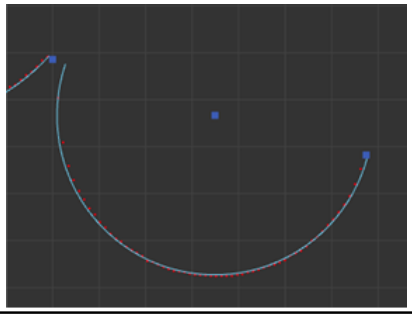
24.2.1.2.2. Approximation by arcs

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.

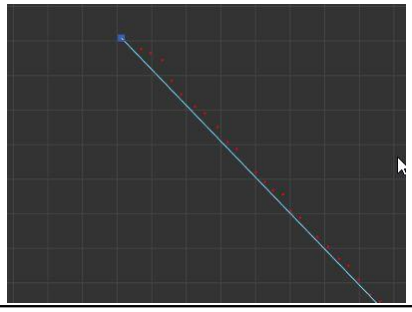
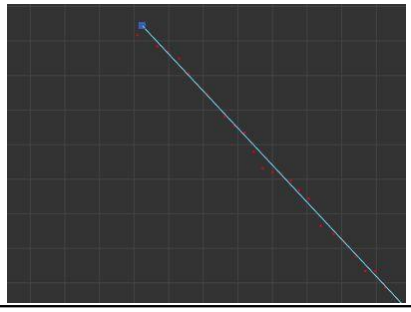
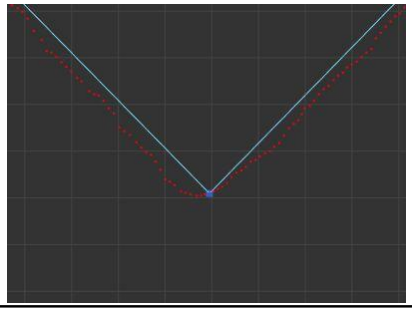
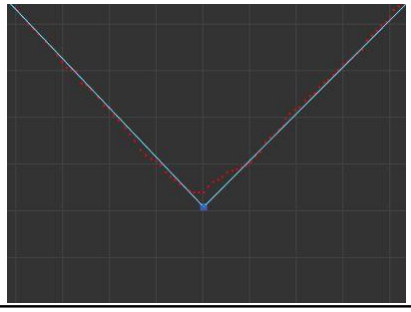
- **Expected radius, mm** - sets the minimum and maximum radii of the arc circle. If, as a result of the approximation of a profile element, an arc is obtained, the radius of the circle of which does not fit into the specified limits, then this element will be approximated by a segment.
- **Average deviation, 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

24.2.1.3. 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

24.2.2. Stage 2. Building a graph

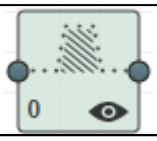
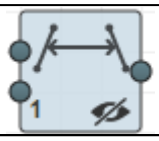
Upon completion of the profile approximation procedure, it is necessary to build a computational graph.

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.

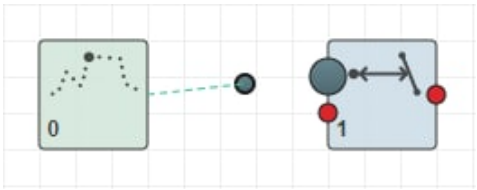
The graph building area also displays the time (in microseconds) spent on profile approximation and graph rendering. If this time is less than 1/(frequency of profiles in seconds), each profile is processed, if greater - some profiles may be skipped and not processed.

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.

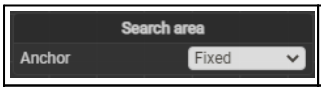
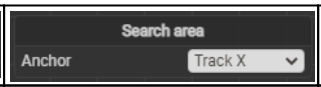
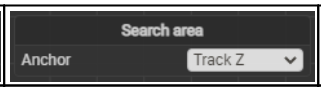
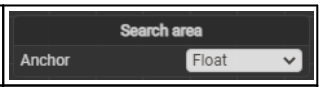
24.2.2.1. Results display area

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.

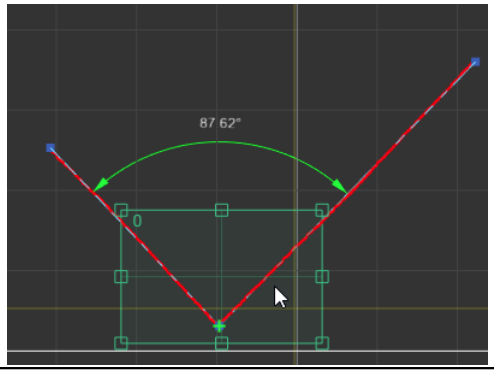
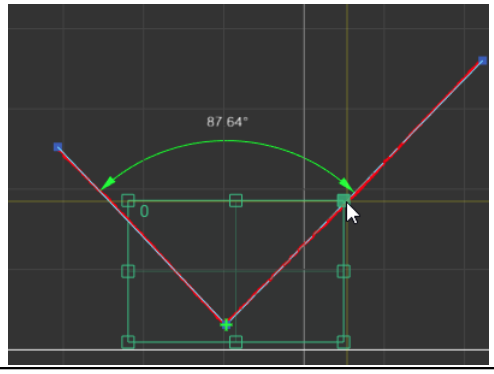
As noted above, some blocks have search areas within which the functions of the block are performed. The use of search areas makes it possible to exclude noise, sensor flare and other factors from calculations. The search area can be:

- 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:

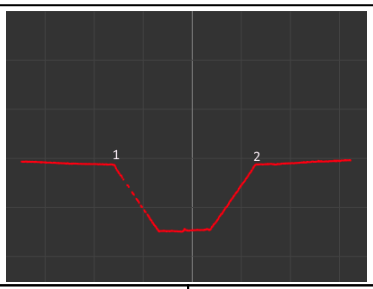
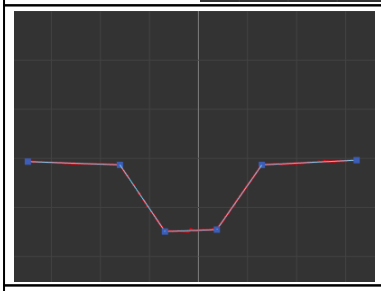
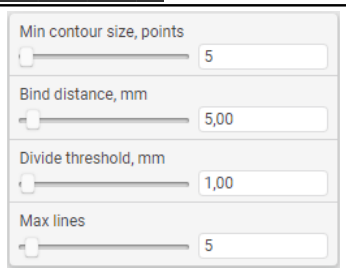
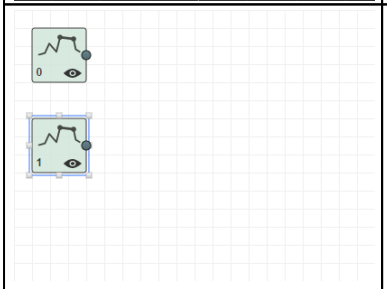
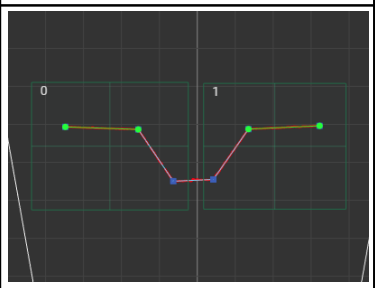
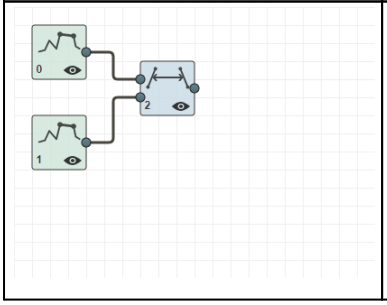
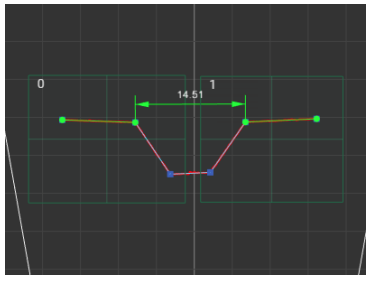
			
Fixed area	Tracking along the X coordinate	Tracking along the Z coordinate	Floating area

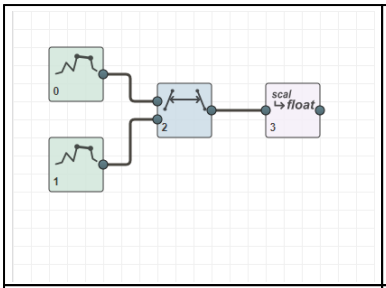
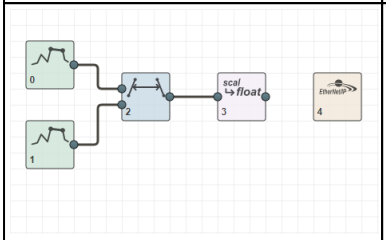
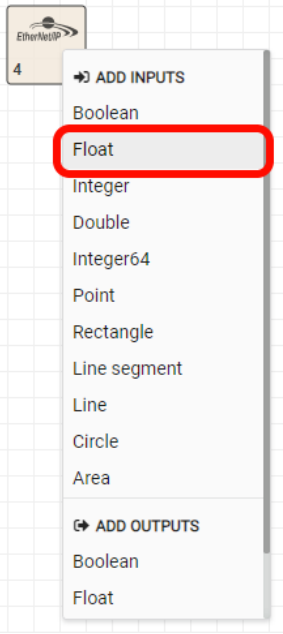
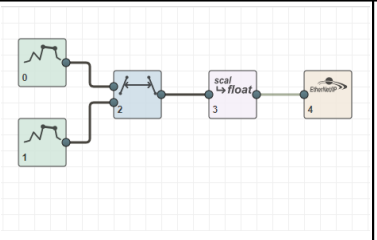
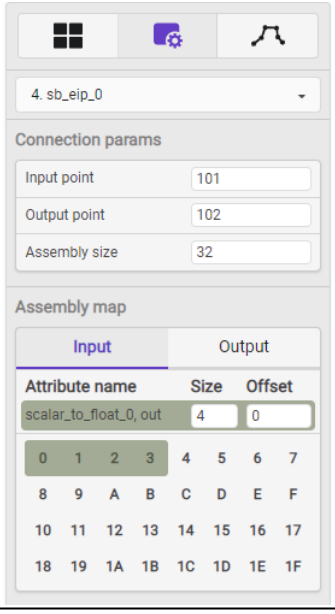
In any mode of moving an area, the user can move and resize the area. Moving is done with the right mouse button (click on the search area and move the mouse). Resizing is carried out using special rectangles located along the perimeter of the search area:

	
Moving the search area	Resizing the search area

24.2.2.2. 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.

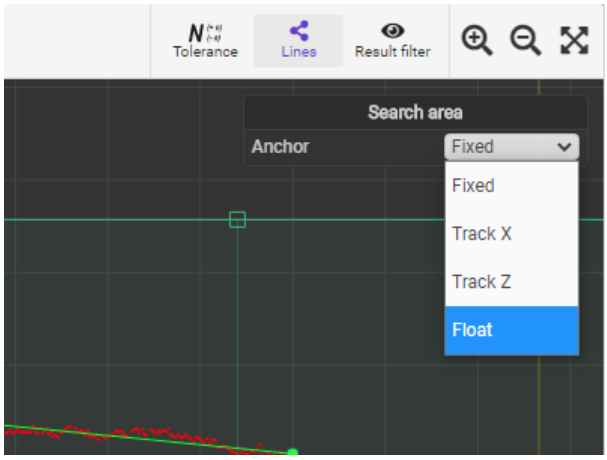
		Source profile. It is necessary to find the distance between points 1 and 2.
		Configure the algorithm for approximating the profile by segments to obtain a stable splitting in accordance with the profile shape. See par. 24.2.1.
		Place two “segment detector” blocks in the graph building area to select 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. 24.1.2) 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.

24.2.3. 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. 24.3.1.).

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).

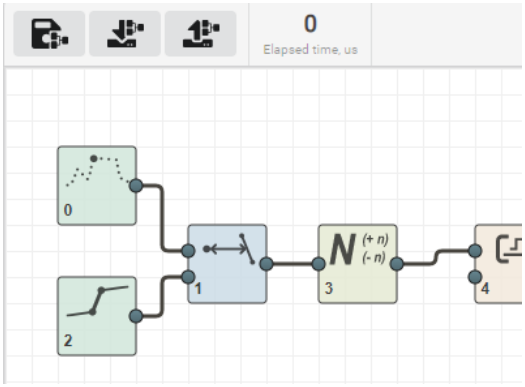
The last block (“eip”) receives the distance of the general type "float", and places it in the input assembly, which sends the data to the network.

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

24.2.4. 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.

Button	Assignment
	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.

24.3. Smart blocks

24.3.1. 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;
		}

common types	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.	
SDT_BOOL	bool_t	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”.
SDT_FLOAT	float	Single-precision floating-point value (size - 4 bytes).

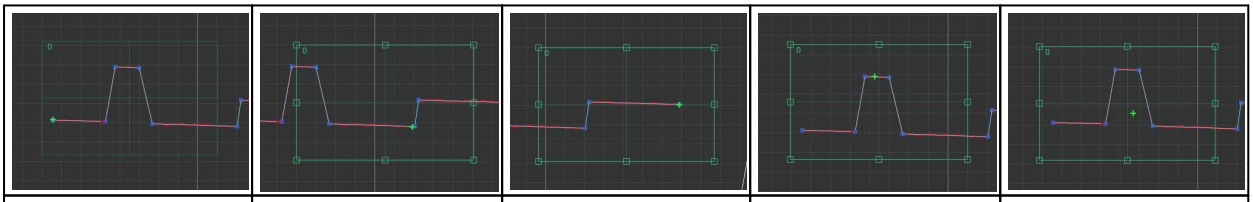
Name	Type	Description
SDT_INT	int32_t	Signed integer value (size - 4 bytes).
SDT_DOUBLE	double	Double-precision floating-point value (size - 8 bytes).
SDT_INT64	int64_t	Signed integer value (size - 8 bytes).

24.3.2. Sections

Smart blocks are grouped into the following sections:

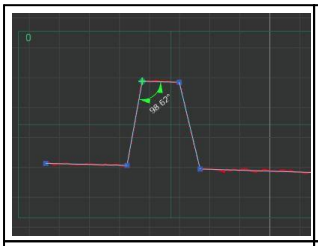
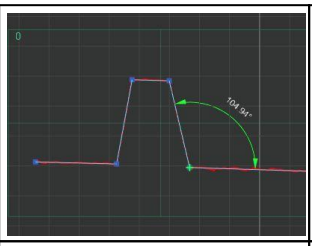
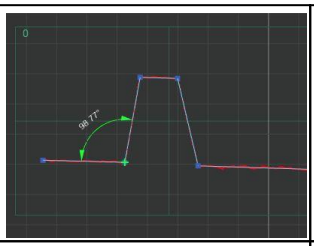
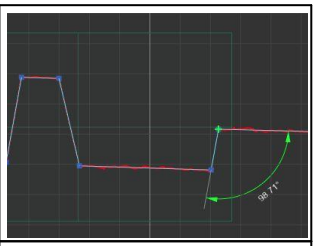
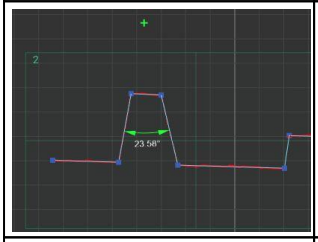
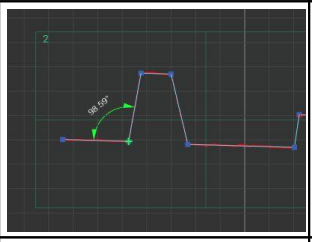
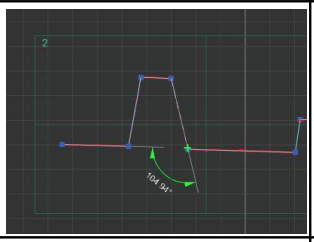
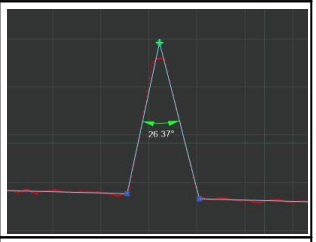
1. "Feature detectors" - Smart blocks designed to extract primitives (points, lines, angles, etc.) from a profile.
2. "Welding" - Smart blocks designed to solve welding tasks, such as detecting a welding joint and measuring its parameters.
3. "Complex shapes" - Smart blocks that detect complex-shaped elements (trapezoids, corner joints, etc.), taking into account the peculiarities of the shape of the detected element.
4. "Math functions" - Smart blocks that perform mathematical operations on primitives (calculation of distances and angles, conversion of units of measurement, etc.).
5. "Converters" - Smart blocks for performing conversions (conversion of types, conversion of units of measurement, merging of primitives, decomposition of primitives, etc.).
6. "Control" - Smart blocks for controlling the measured values within tolerances.
7. "Input and output" - Smart blocks for outputting measurement results and other signals and entering information from external systems into the graph.

24.3.2.1. "Feature detectors" section

<table>
<tr><td> 
</td><td colspan="3">“point detector” - searching for a profile point.</td></tr>
<tr><td rowspan="5">Parameters:</td><td rowspan="5">“Mode”</td><td>min X</td><td>The profile point with the minimum X coordinate.</td></tr>
<tr><td>min Z</td><td>The profile point with the minimum Z coordinate.</td></tr>
<tr><td>max X</td><td>The profile point with the maximum X coordinate.</td></tr>
<tr><td>max Z</td><td>The profile point with the maximum Z coordinate.</td></tr>
<tr><td>average</td><td>The profile point with averaged coordinates.</td></tr>
<tr><td colspan="4">Examples:
 
min X | min Z | max X | max Z | average</td></tr>
<tr><td>Outputs:</td><td>“pos”</td><td>SDT_POINT</td><td>Point coordinates.</td></tr>
<tr><td colspan="4"></td></tr>
<tr><td> 
</td><td colspan="3">“angle detector” - searching for the angle between two line segments of the profile.</td></tr>
<tr><td>Parameters:</td><td>“Mode”</td><td>top</td><td>The angle whose vertex has the maximum Z coordinate.</td></tr>
</table>

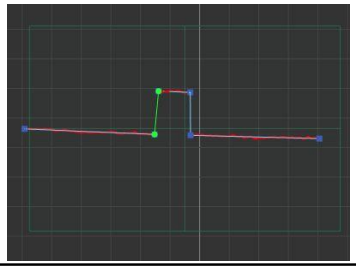
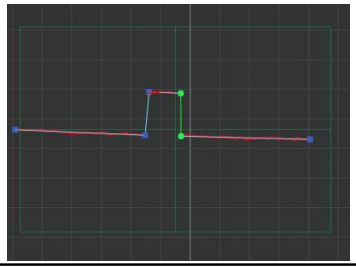
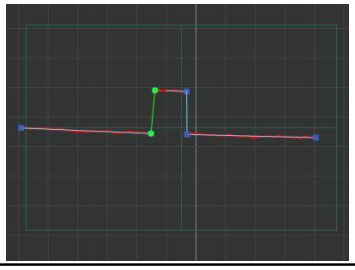
		bottom	The angle whose vertex has the minimum Z coordinate.
		left	The angle whose vertex has the minimum X coordinate.
		right	The angle whose vertex has the maximum X coordinate.
		value	The first angle satisfying the "Angle average" and "Angle tolerance" parameters.
	“Angle value”	0...179	The desired 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.

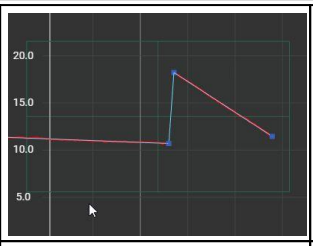
	“edge detector” - searching for the difference in height between two segments of the profile.		
Parameters:	“Mode”	rise	Height difference “rise”.
		fall	Height difference “fall”.
		any	Any direction of height difference. The choice is made by height difference.

		
rise	fall	any

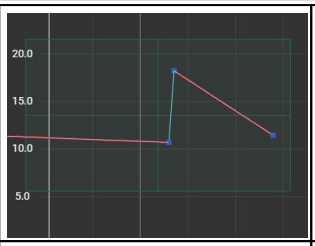
	“Min step”, mm	0,01...1000	The minimum allowable height difference for detection. The direction is not taken into account.

	“Angle tolerance”, deg	0...45	The maximum permissible angle between two segments forming a difference in height.

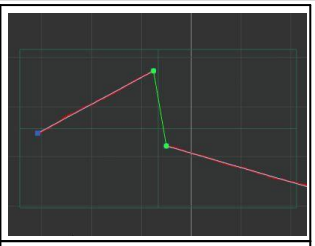
“Min step” = 5 mm
“Angle tolerance” = 35°



“Min step” = 10 mm
“Angle tolerance” = 35°



“Min step” = 10 mm
“Angle tolerance” = 30°

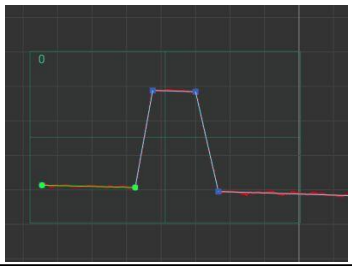


“Min step” = 5 mm
“Angle tolerance” = 45°

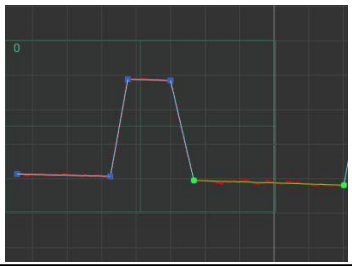
Outputs:	“edge”	SDT_SEGMENT	The segment corresponding to the height difference.

“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).

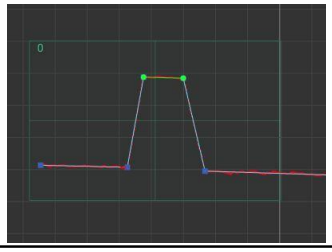
Parameters:	“Mode”	first	The leftmost segment.
		last	The rightmost segment.
		by index	The segment with the index specified by the “Index” parameter.
		rise edge	The central segment of the "edge" with the "Rise" height difference.
		fall edge	The central segment of the "edge" with the "Fall" height 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”.



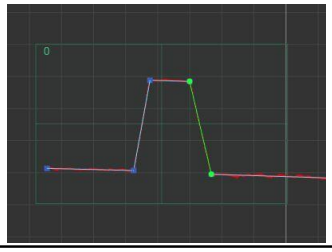
first



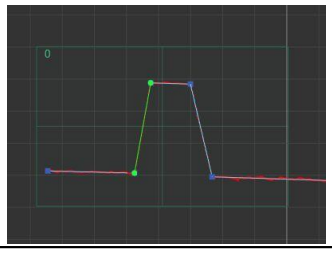
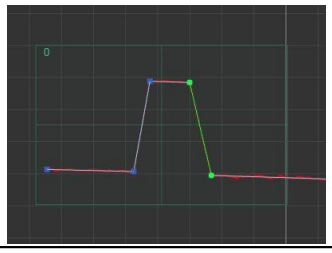
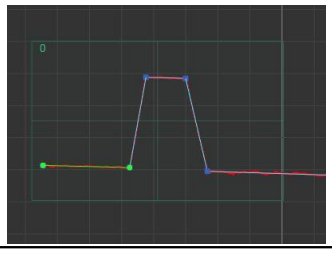
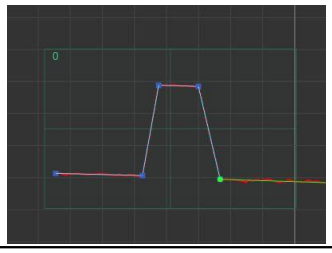
last



by index, “Index” = 2

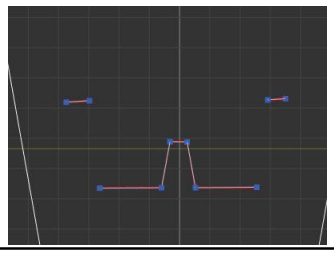
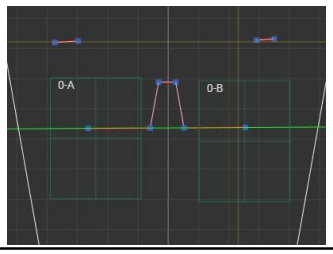
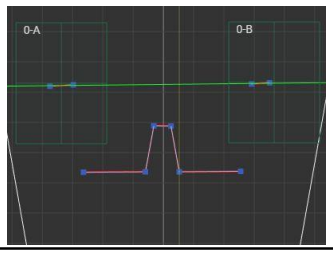


by index, “Index” = 3

	
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.

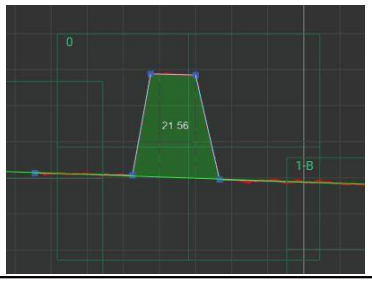
	“line approximation” - line approximation of profile points (in two areas).

		
original profile without approximation	approximation of lower fragments	approximation of upper fragments

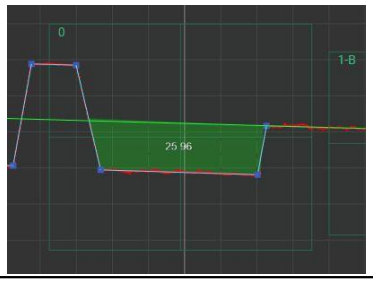
Outputs:	“line”	SDT_LINE	The line that approximates the points located in areas.

	“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.

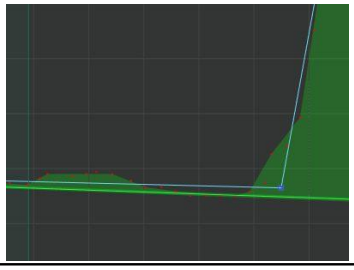


above

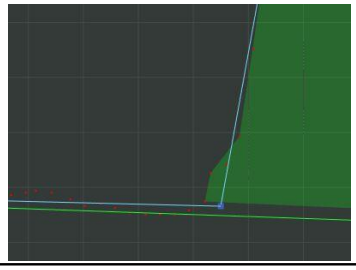


below

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.

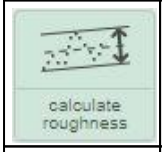


“Threshold” = 0.01 mm



“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, mm^2.



“calculate roughness” - calculating the roughness of the profile relative to the input baseline.

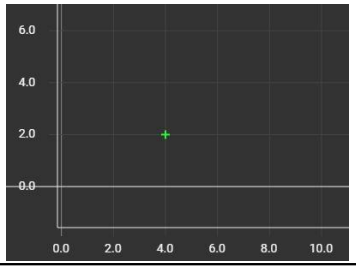
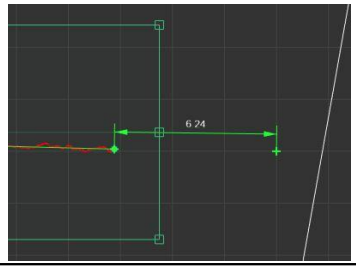
Parameters:	“Mode”	std. dev.	The standard deviation.
		pos dev	The maximum positive deviation (upward relative to the base segment).
		neg dev	The maximum negative deviation (downward relative to the baseline).

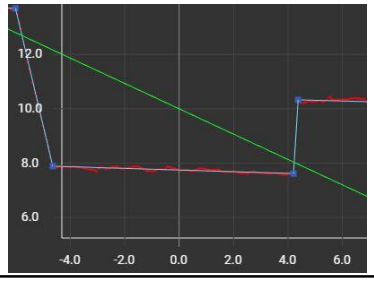
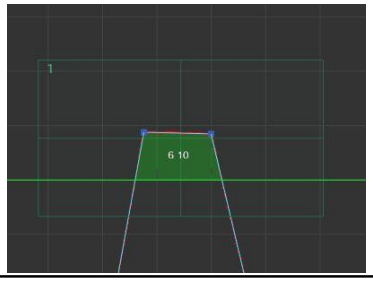
std. dev.

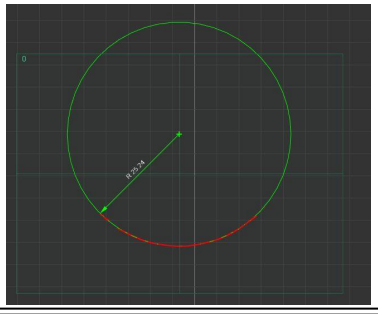
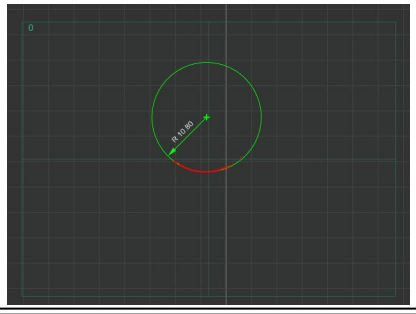
pos dev

neg dev

Inputs:	“in1”	SDT_SEGMENT	The base segment with respect to which the roughness is analyzed.
Outputs:	“value”	SDT_SCALAR	The roughness value, mm^2.

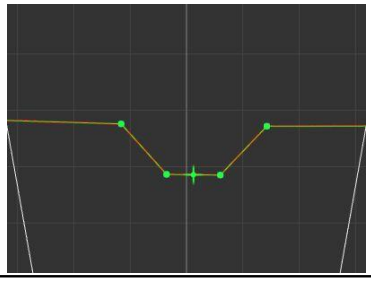
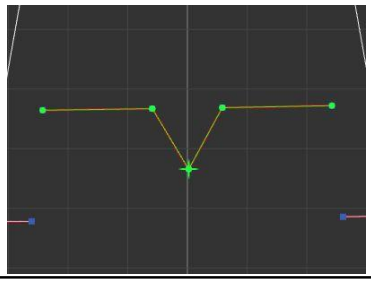
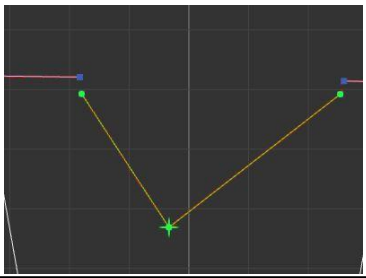
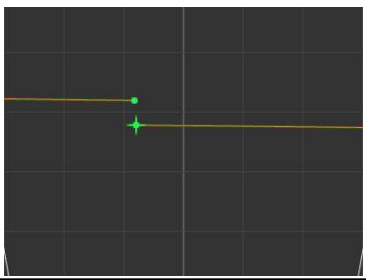
	“fixed point” - the fixed point (the position does not depend on the profile).		
Parameters:	“Position by X”, mm	0,01...1000	The position of the point along the X axis.
	“Position by Z”, mm	0,01...1000	The position of the point along the Z axis.
			
“Position by X” = 4 mm, “Position by Z” = 2 mm		Measuring the distance between the profile point and the fixed point	
Outputs:	“pos”	SDT_POINT	The point with given coordinates.

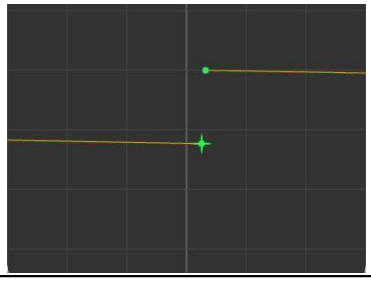
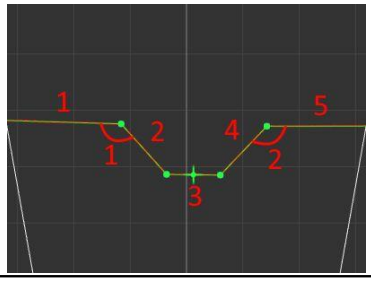
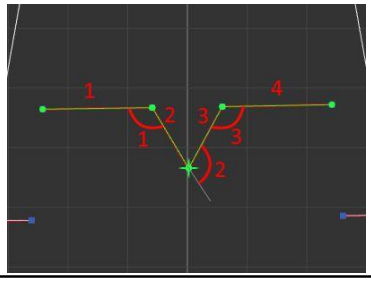
	“fixed line” - the fixed line (the position does not depend on the profile).		
Parameters:	“Slope”, deg	-90...90	The slope of the line relative to the horizontal axis.
	“Position by Z”, mm	0,01...1000	The position of the line relative to the Z axis.
			
“Slope” = 25°, “Position by Z” = 10 mm		Measuring the filling area relative to the fixed line	
Outputs:	“line”	SDT_LINE	The line with the specified parameters.

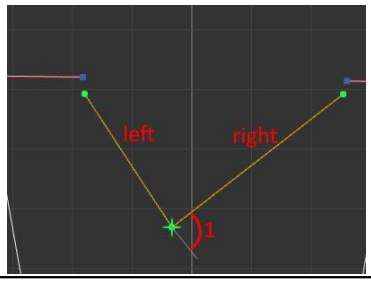
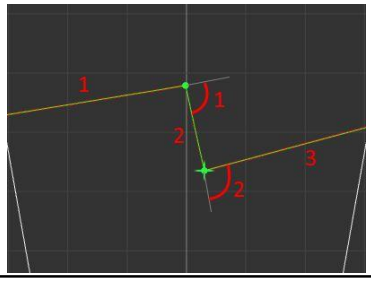
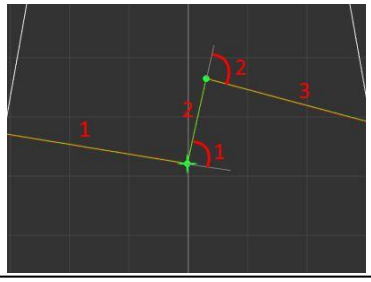
	“circle detector” - searching for a circle on a profile.		
Parameters:	"Radius value", mm	0,01...1000	The circle radius.
	"Radius tolerance", mm	0,01...1000	The radius tolerance (in both directions).
			

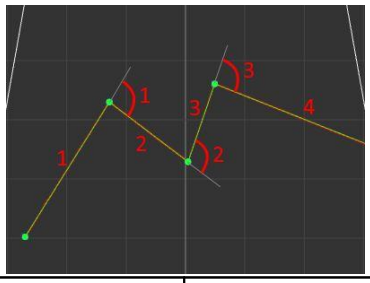
	"Radius value" = 25 mm, "Radius tolerance" = 5 mm		"Radius value" = 10 mm, "Radius tolerance" = 1 mm
Outputs:	"radius"	SDT_SCALAR	The circle radius in mm.
	“center”	SDT_POINT	Circle center coordinates.

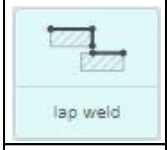
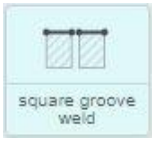
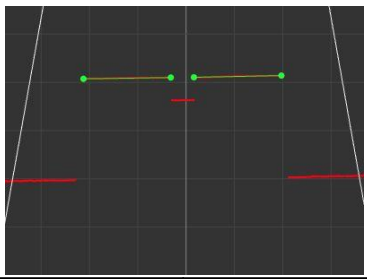
24.3.2.2. "Welding" section

	“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: 

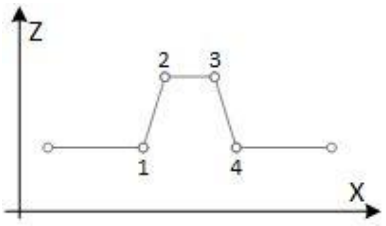
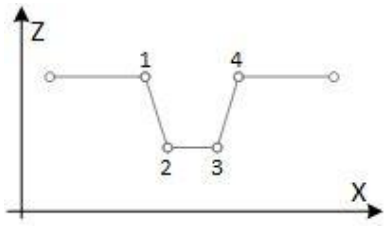
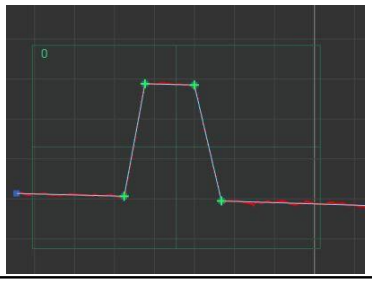
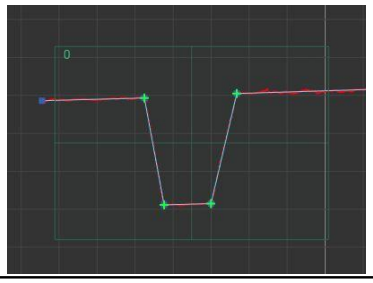
	“Min len (segment 1)”, mm “Max len (segment 1)”, mm	0,1…1000 0,1…1000	Minimum and maximum length of segment #1.
	“Min len (segment 2)”, mm “Max len (segment 2)”, mm	0,1…100 0,1…100	Minimum and maximum length of segment #2.
	“Min len (segment 3)”, mm “Max len (segment 3)”, mm	0,1…1000 0,1…1000	Minimum 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).

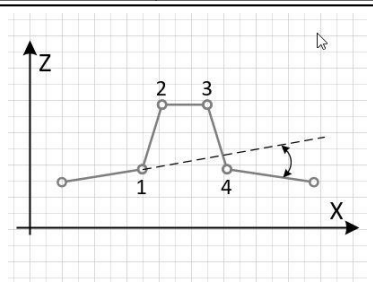
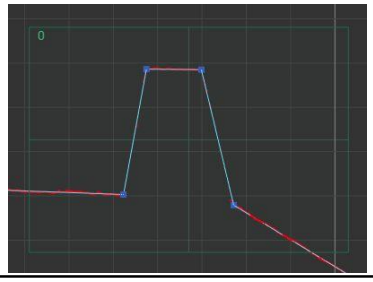
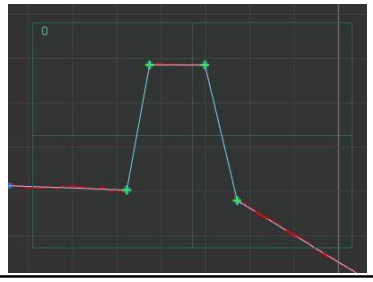
	“left segment”	SDT_SEGMENT	Segment to the left.
	“right segment”	SDT_SEGMENT	Segment to the right.

24.3.2.3. "Complex shapes" section

	“trapeze groove” - searching for a trapezoidal groove formed by five segments (one segment is located to the left of the trapezoid, another segment is located to the right of the trapezoid and three segments make up the trapezoid).		
Parameters:	“Mode”	convex	Convex trapezoid.
		concave	Concave trapezoid (hollow).

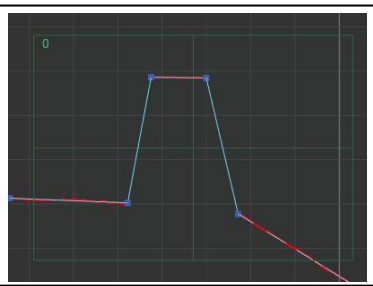
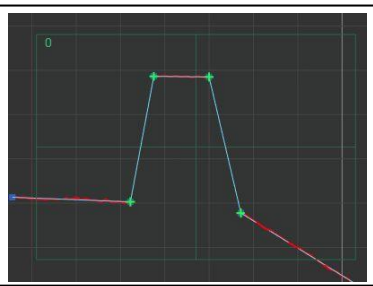
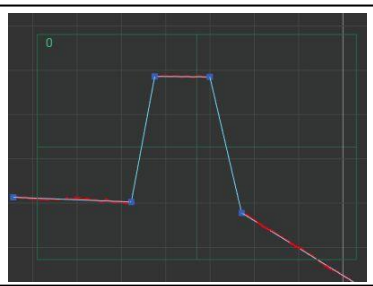
	
	
convex	concave

	“Basis tolerance”, deg	0...60	The permissible angle between segments to the left and right of the trapezoid.

		
“Basis tolerance”	“Basis tolerance” = 5°, trapezoid is not detected	“Basis tolerance” = 30°, trapezoid is detected

	“Sides angle”, deg	0...90	The angle between the segments that form the sides of the trapezoid.
	“Sides tolerance”, deg	0...60	The maximum permissible deviation of the angle (in both directions) between the segments that form the sides of the trapezoid.

The actual value of the angle between the sides of the scanned trapezoid is 23.5°.

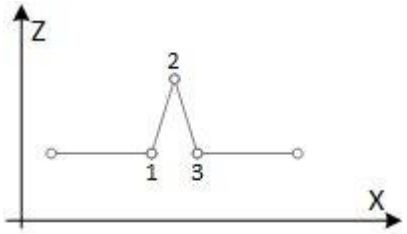
		

“Sides angle” = 21°, “Sides tolerance” = 2°, trapezoid is not detected	“Sides angle” = 22°, “Sides tolerance” = 2°, trapezoid is detected	Sides angle” = 21°, “Sides tolerance” = 1°, trapezoid is not detected

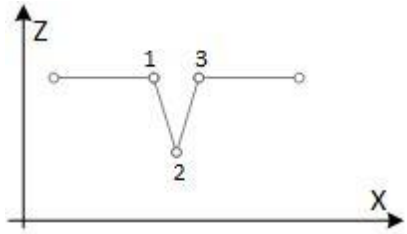
Outputs:	“point_1”	SDT_POINT	Coordinates of point #1.
	“point_2”	SDT_POINT	Coordinates of point #2.
	“point_3”	SDT_POINT	Coordinates of point #3.
	“point_4”	SDT_POINT	Coordinates of point #4.

“corner joint” - searching for a corner joint formed by 4 segments (one segment is located to the left of the corner joint, another segment is located to the right of the corner joint, two segments form the corner joint).

Parameters:	“Mode”	convex	The convex corner joint.
		concave	The concave corner joint.

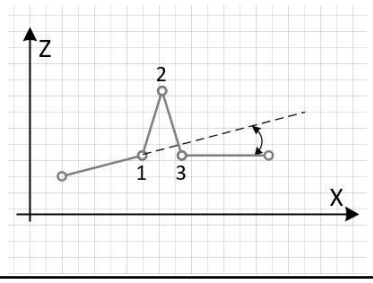


convex

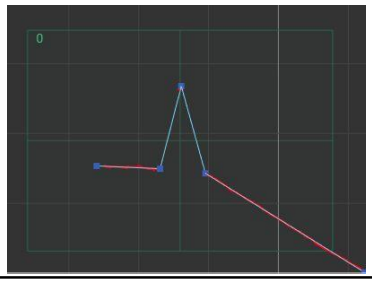


concave

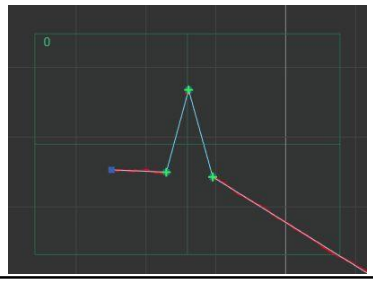
	“Basis tolerance”, deg	0...60	The permissible angle between the segments to the left and right of the corner joint.



“Basis tolerance”



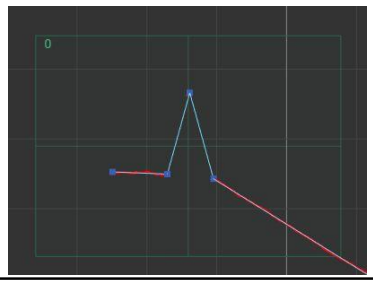
“Basis tolerance” = 5°, the corner joint is not detected



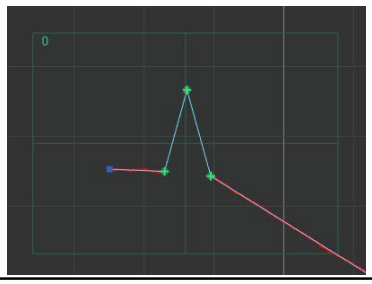
“Basis tolerance” = 30°, the corner joint is detected

	“Angle value”, deg	0...89	The angle between the segments that form the corner.
	“Angle tolerance”, deg	0...60	The maximum permissible deviation of the angle (in both directions) between the segments that form the corner.

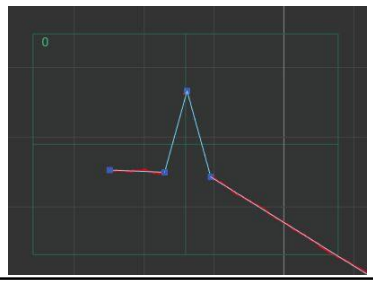
The actual value of the angle is 30.7°.



“Angle value” = 25°, “Angle tolerance” = 2°, the corner is not detected



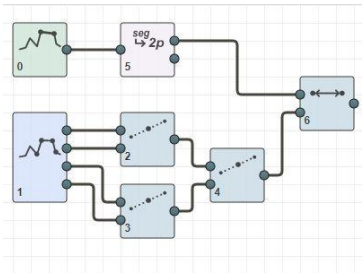
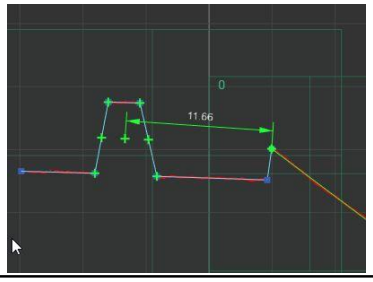
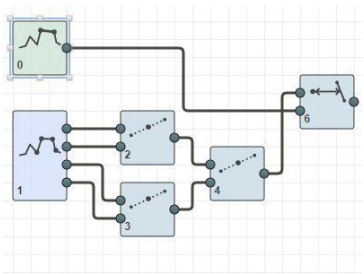
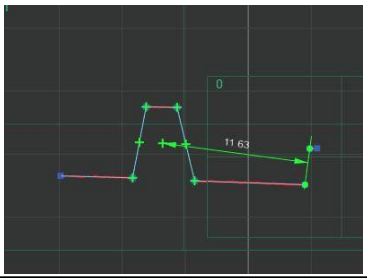
“Angle value” = 29°, “Angle tolerance” = 2°, the corner is detected



“Angle value” = 29°, “Angle tolerance” = 1°, the corner is not detected

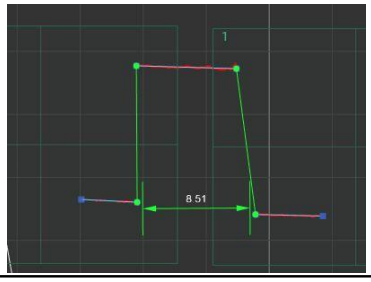
Outputs:	“point_1”	SDT_POINT	Coordinates of point #1.
	“point_2”	SDT_POINT	Coordinates of point #2.
	“point_3”	SDT_POINT	Coordinates of point #3.
	“template detector” - detection of a custom template.		
Parameters:	“Edit”		Opening the template editor. The procedure for creating a custom template is described in Annex 6.
Outputs:	“det”	SDT_BOOL	Boolean flag of template detection.
	“out”	SDT_SEGMENT	Dynamically generated outputs for template segments.

24.3.2.4. "Math functions" section

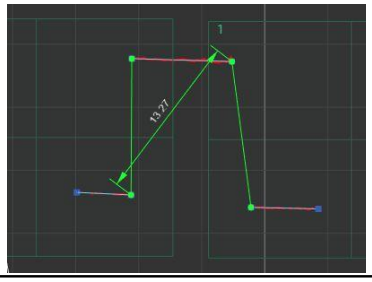
	“point to point distance” - calculating the distance between two points arriving at the block inputs.		
			
Measuring the distance between the center of the trapezoid and the start of the line segment			
Inputs:	“in1”	SDT_POINT	Point #1.
	“in2”	SDT_POINT	Point #2.
Outputs:	“dist”	SDT_SCALAR	The distance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

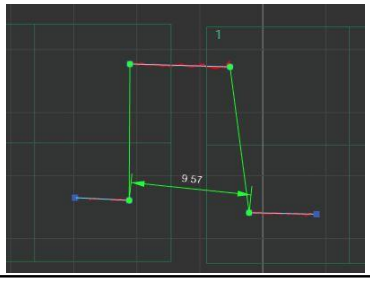
Parameters:	"Mode"	perpendicular	The perpendicular distance between parallel segments.
		begin<->begin	The distance between the beginnings of the segments (leftmost points).
		begin<->end	The distance between the beginning of one segment (leftmost point) and the end of another segment (rightmost point).
		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).



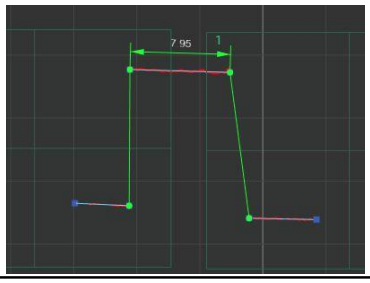
"Mode" = perpendicular



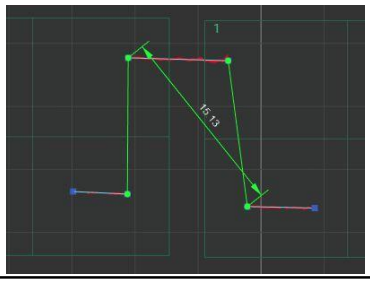
"Mode" = begin<->begin



"Mode" = begin<->end



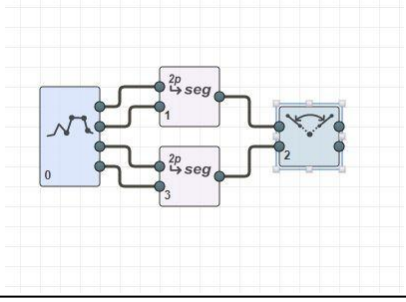
"Mode" = end<->begin

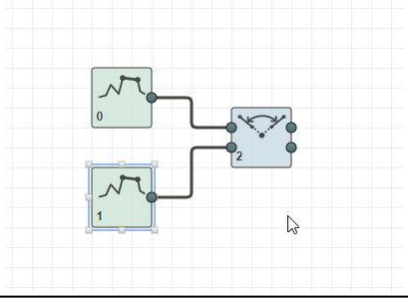


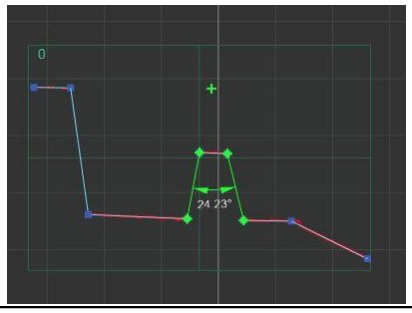
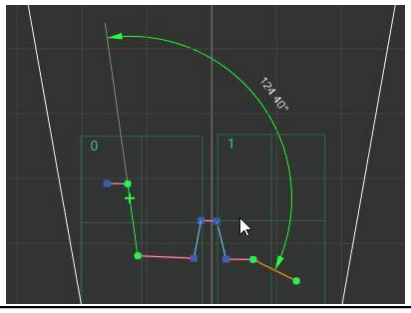
"Mode" = end<->end

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

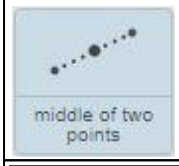
"segments intersections" - calculating the center of intersection of segments and the angle between them.

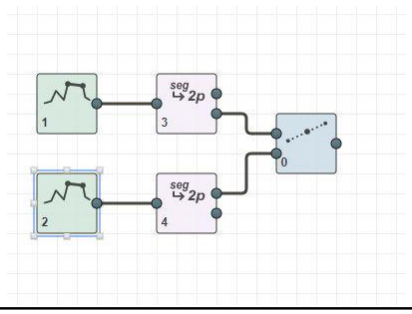
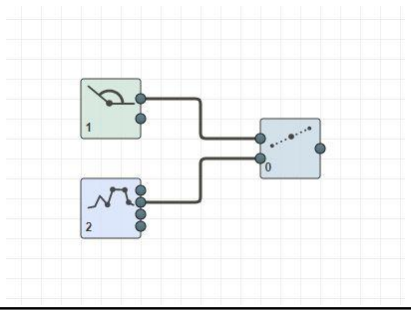
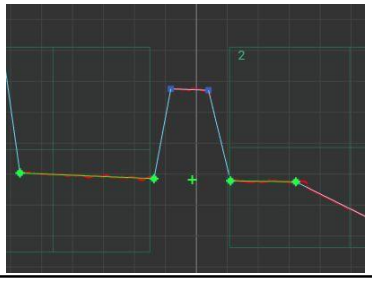
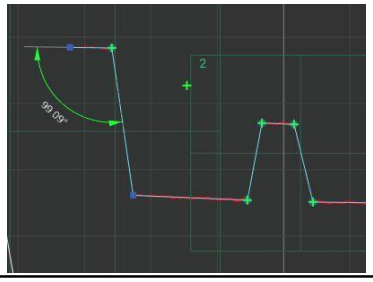




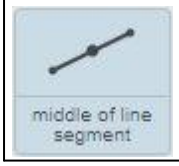
	
Measuring the angle and determining the point of intersection of the side segments of the trapezoid	Measuring the angle and determining the point of intersection of arbitrary segments on the profile

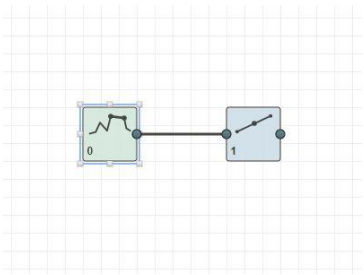
Inputs:	“in1”	SDT_SEGMENT	Segment #1.
	“in2”	SDT_SEGMENT	Segment #2.
Outputs:	“point”	SDT_POINT	The intersection point of two segments.
	"angle"	SDT_SCALAR	The angle between segments in degrees.

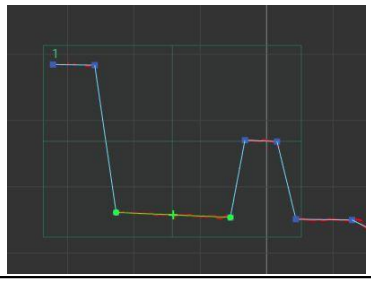
	“middle of two points” - calculating the midpoint between two points.

	
	
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:

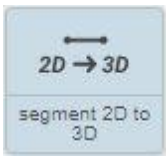
X, Y, Z - coordinates of a point in the 3D coordinate system of an external device;

X_0, Y_0, Z_0 - calibration offsets;

A[6] - rotation matrix coefficients;

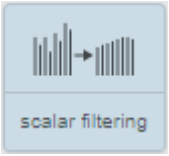
x, y - coordinates of a point in the 2D coordinate system of the scanner.

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.

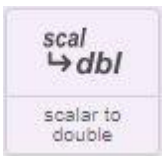


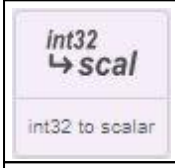
“segment 2D to 3D” - converting the line segment from the local 2D coordinate system of the scanner to the 3D coordinate system of an external device. The conversion procedure is performed for the segment endpoints and is the same as for “point 2D to 3D”.

Inputs:	“in”	SDT_SEGMENT	Segment.
Outputs:	“x1”	SDT_SCALAR	The X coordinate of the left point in the 3D coordinate system of the external device.
	“y1”	SDT_SCALAR	The Y coordinate of the left point in the 3D coordinate system of the external device.
	“z1”	SDT_SCALAR	The Z coordinate of the left point in the 3D coordinate system of the external device.
	“x2”	SDT_SCALAR	The X coordinate of the right point in the 3D coordinate system of the external device.
	“y2”	SDT_SCALAR	The Y coordinate of the right point in the 3D coordinate system of the external device.
	“z2”	SDT_SCALAR	The Z coordinate of the right point in the 3D coordinate system of the external device.

 scalar filtering	“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.

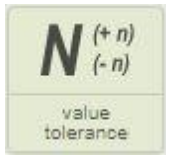
24.3.2.5. "Converters" section

 scal ↳bool scalar to bool	“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.
 scal ↳int32 scalar to int32	“scalar to int32” - converting a scalar value to an integer value (size - 4 bytes). The conversion is performed with rounding to the smallest (in absolute value) integer value.			
	Inputs:	“in”	SDT_SCALAR	Scalar value.
	Outputs:	“out”	SDT_INT	Integer value.
 scal ↳float scalar to float	“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.
 scal ↳dbl scalar to double	“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.
	“int32 to scalar” - converting an integer value (size - 4 bytes) to a scalar value.			
	Inputs:	“in”	SDT_INT	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.			
	Входы:	“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.
..				

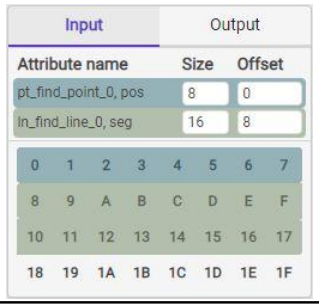
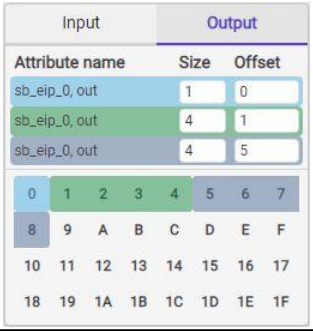
	“inch to mm” - converting a scalar value in inches to a scalar value in millimeters.			
	Inputs:	“in”	SDT_SCALAR	Scalar in inches.
	Outputs:	“result”	SDT_SCALAR	Scalar in mm.
	“seg to two points” - converting a line segment to two points corresponding to the ends of the segment.			
	Inputs:	“in”	SDT_SEGMENT	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.

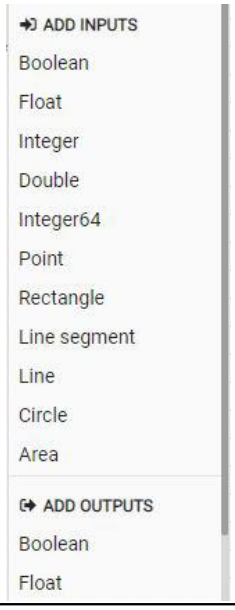
24.3.2.6. "Control" section

	“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.

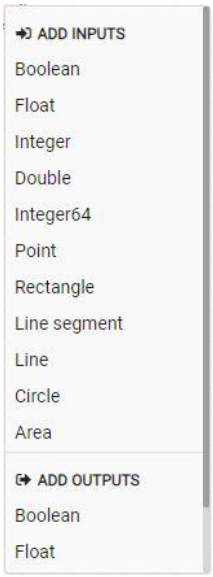
scanner sensor	“scanner sensor” - control of the parameters of the CMOS sensor installed in the scanner.			
	Inputs:	“pps”	SDT_INT	The required number of profiles per second (may be limited by the scanner operation mode).
		“expose1”	SDT_INT	Frame exposure time in µs.

24.3.2.7. "Input and output" section

EtherNet/IP eip	“Etnernet/IP” - transmitting and receiving data over Ethernet/IP. Only one instance of this block is allowed 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 located in accordance with this parameter.
			
	“Assembly map”	Output	Distribution of block outputs over the output assembly. The values of the outputs must be located in accordance with this parameter.
			
Inputs:	The inputs are created by the user using the context menu.		

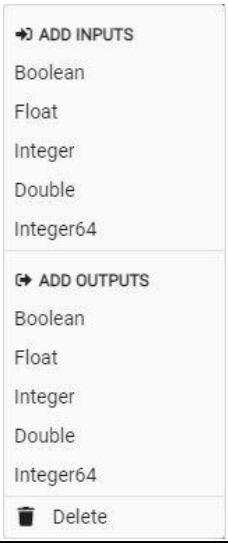
<table>
<tr><td></td><td colspan="3"> 
</td></tr>
<tr><td>Outputs:</td><td colspan="3">The outputs are created by the user using the context menu.</td></tr>
<tr><td colspan="4"></td></tr>
<tr><td> 
</td><td colspan="3">“UDP” - transmitting and receiving data over UDP (User Datagram Protocol).</td></tr>
<tr><td rowspan="9">Parameters:</td><td>“Output datagram”, bytes</td><td>8...16384</td><td>The size of the sent datagram. This datagram will contain data from the inputs of the block.</td></tr>
<tr><td>“Destination IP”</td><td>XXX.XXX.XXX.XXX</td><td>The IP address of the host to which the datagram is sent.</td></tr>
<tr><td>“Destination port”</td><td>1...65535</td><td>The port number of the host to which the datagram is sent.</td></tr>
<tr><td>“Input datagram”, bytes</td><td>8...16384</td><td>The size of the received datagram. This datagram must contain data for the block outputs.</td></tr>
<tr><td>“Receive IP”</td><td>XXX.XXX.XXX.XXX</td><td>The IP address of the scanner. It is set in the general settings of the scanner.</td></tr>
<tr><td>“Receive port”</td><td>1...65535</td><td>The port number of the scanner listening for incoming datagrams.</td></tr>
<tr><td rowspan="2">“Port map”</td><td>Send</td><td>Distribution of block inputs over the sent datagram. The values of the inputs will be located in accordance with this parameter.</td></tr>
<tr><td colspan="2"> 
</td></tr>
<tr><td>“Port map”</td><td>Receive</td><td>Distribution of block outputs in the received datagram. The output values must be located in accordance with this parameter.</td></tr>
</table>

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



phys out	“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.
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 is allowed 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 located 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 located 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 located 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 located in accordance with this parameter.
			
Inputs:	The inputs are created by the user using the context menu.		

	
Outputs:	The outputs are created by the user using the context menu.
	“robot protocol HND1” - data exchange with robots using the HND1 protocol. A description of the protocol is given in Annex 5.

Parameters:	“Destination IP”	XXX.XXX.XXX.XXX	The IP address of the robot (or other device with which data is exchanged).
	“Destination port”	1...65535	The network port number of the robot (or other device with which data is exchanged).
	“Listen port”	1...65535	The network port number of the scanner listening for incoming packets.
	“Swap X<->Y”	true/false	Swap the X and Y coordinates of points.
	“Flip X-axis”	on/off	Flip (relative to 0) coordinates along the X axis. It is performed after applying the “Swap X<->Y” parameter.
	“Flip Y-axis”	on/off	Flip (relative to 0) coordinates along the Y axis. 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 of template detection.
	“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.

25. 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.

26. 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. 25.
	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.

27. 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.

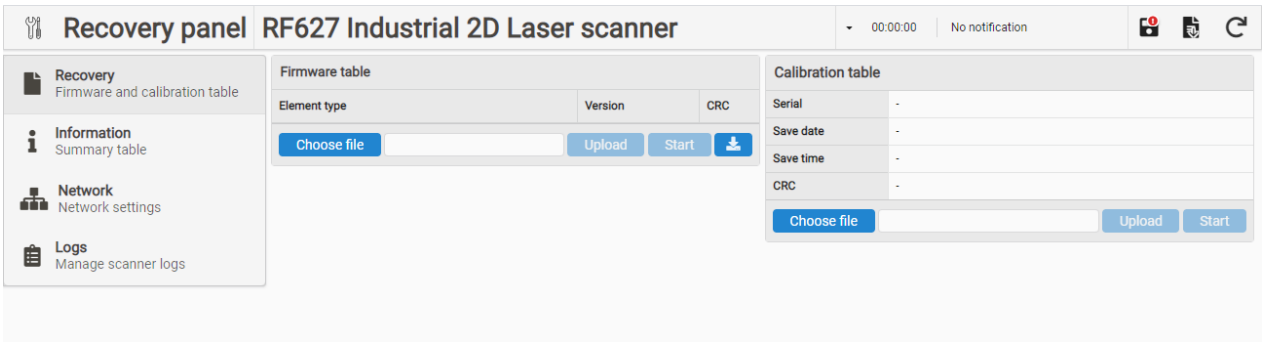
In this mode, the **PWR** indicator displays the SOS signal (three short - three long - three short), which means that the scanner is in **Recovery** mode now.

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:



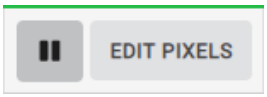
The controls in the upper part are the same as in the main web page. The sections of the web page in Recovery mode correspond to the modes of the main web interface.

Recovery mode	Main interface	Section
Recovery	"Update" section, "System" tab	23.2. "Update" section.
Information	"Information" section, "System" tab	23.1. "Information" section.
Network	"Network" tab	17. "Network" tab. Network parameters.
Logs	"Logs" section, "System" tab	23.4. "Logs" section.

28. Annex 2. Editing defective pixels

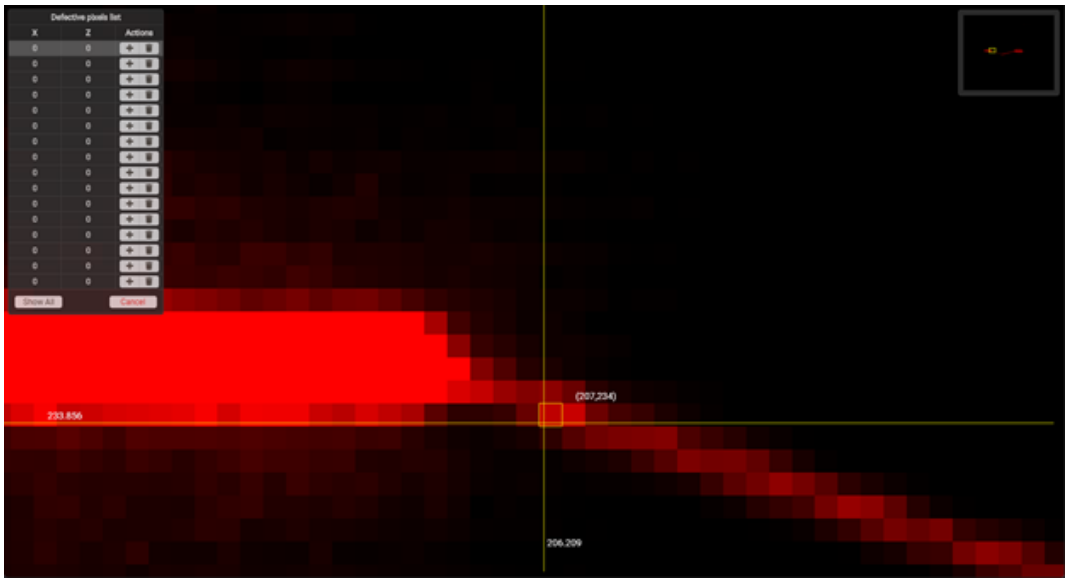
During the operation of the scanner, defective pixels may appear in the CMOS sensor, which significantly distort the profile extracted from the image. The procedure for marking defective pixels is given below. After marking, the signal value of the defective pixel is automatically calculated as the result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

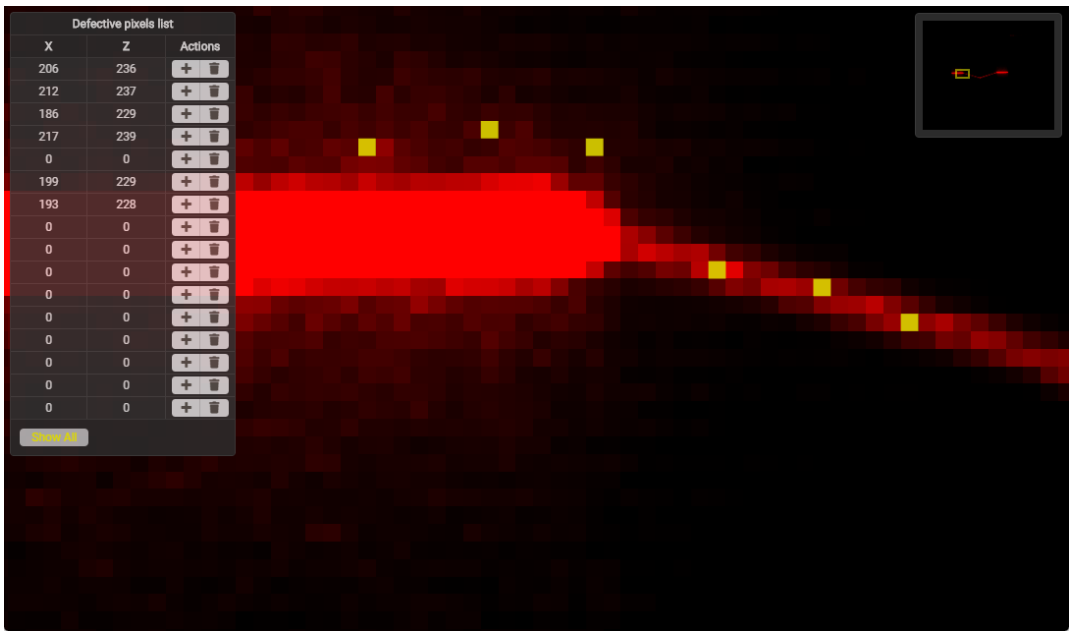


When this mode is enabled, a window with a list of defective pixels is displayed.

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 Cancel.



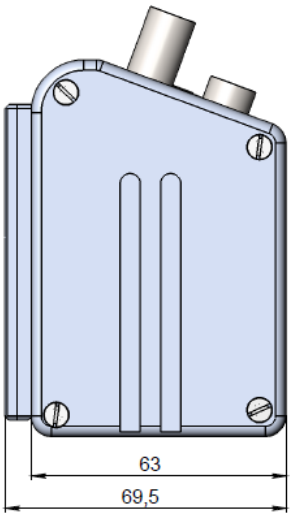
To clear the table row (cancel the defective pixel interpolation), click in the required row. After clicking Show All, all the defective pixels added to the table will be selected.

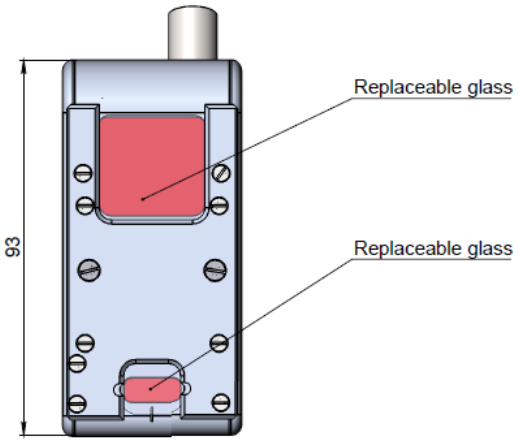


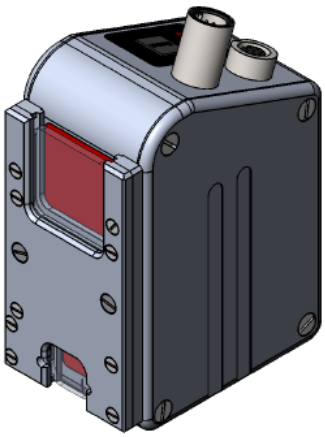
29. Annex 3. Overall and mounting dimensions of scanners with options

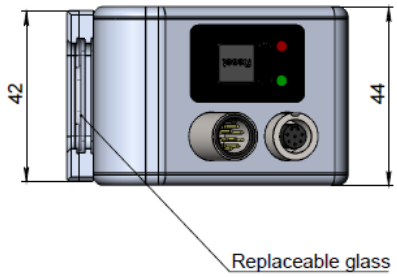
29.1. Example of a scanner with replaceable protective windows, EW option

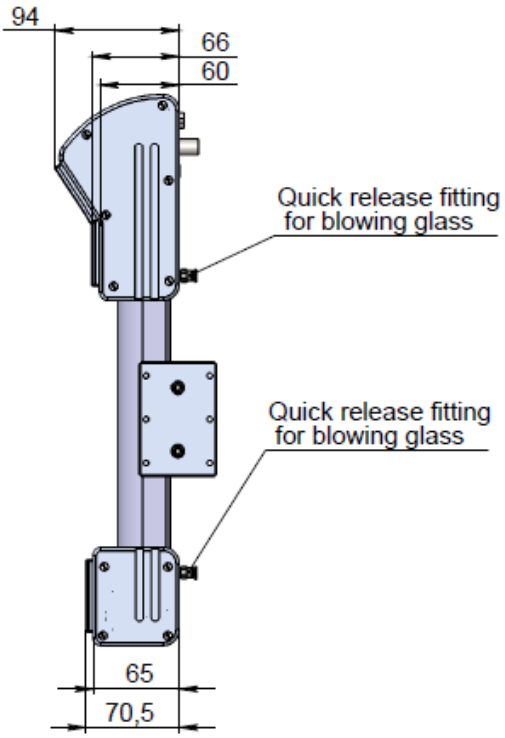
The scanners with replaceable protective windows:

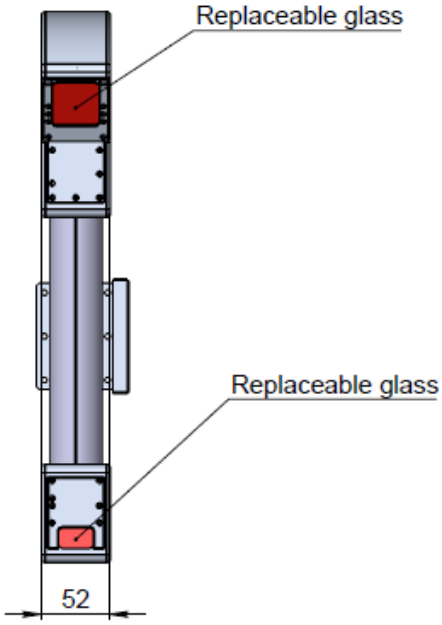


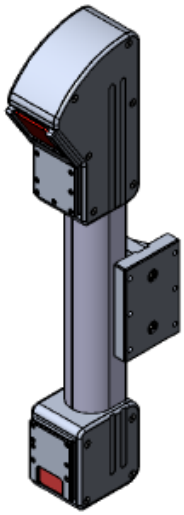


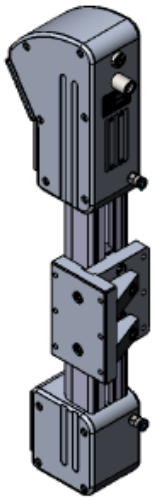





94
66
60
Quick release fitting
for blowing glass
Quick release fitting
for blowing glass
65
70,5

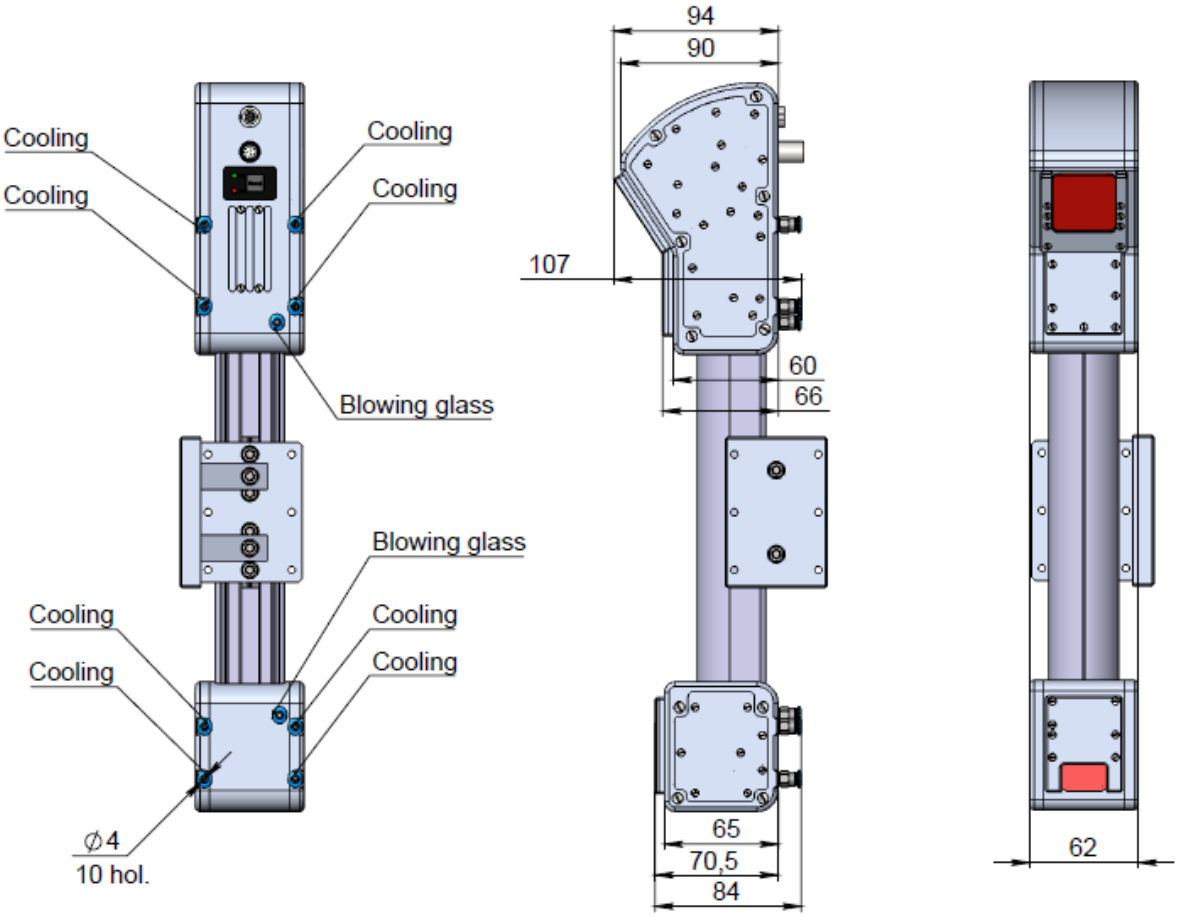

Replaceable glass
Replaceable glass
52





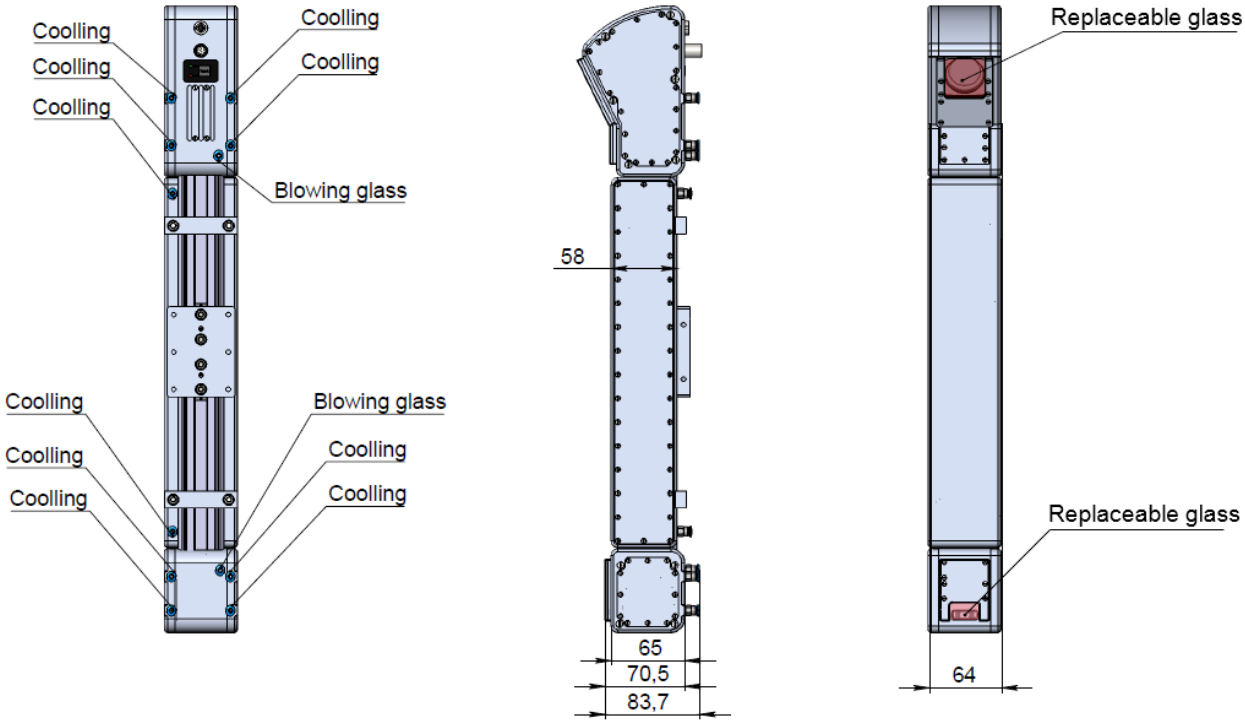
29.2. Example of a scanner with air cooling, AK-EW-AC option

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



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

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



30. Annex 4. Web API

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 5. HND1 protocol, version 1.0.

31.1. Ethernet interface - link layer

Используемый протокол транспортного уровня - UDP.

Сканер, как правило, подключается к контроллеру робота или исполнительной системы и работает как подчиненное устройство. Двусторонний обмен данными осуществляется посылкой сканеру команды ведущим устройством и посылкой сканером ответа. Предусмотрены команды, разрешающие односторонний обмен, а именно посылку сканером результатов измерений до тех пор, пока не будет отдана команда остановиться.

Каждая команда и ответ состоят из заголовка (типа сообщения (команды) и длины последующих данных), за которым следуют непосредственно данные, специфичные для данной команды. Эта последовательность допускает передачу команд и ответов переменной длины. Для повышения эффективности в будущем может быть предусмотрена отправка и получение нескольких команд или ответов в одном пакете UDP. Получатель будет распаковывать и обрабатывать каждую команду в пакете в том порядке, в котором они размещены в пакете.

Слова (16 или 32-разрядные значения) отправляются в формате “little endian”.

31.2. Description of HND1 commands

31.2.1. Getting the protocol version

This command requests the protocol version supported by the scanner. The scanner will respond with the protocol version (two integers, major and minor version codes). The protocol version described in this document is indicated in the section title.

Command name: **MSG_GET_SENSOR_VERSION**

Scanner command:

Type		Length	
1	0	0	0

Parameters:

- none.

Scanner response:

Type		Length		major		minor	
1	0	0	0	1	0	0	0

Parameters:

- major: major version code;
- minor: minor version code.

31.2.2. Setting the laser intensity

This command sets the current laser intensity. It is possible to change the intensity of up to 4 lasers. Each 16-bit field defines the intensity in % (0 - minimum intensity, 100 - maximum possible intensity). The value changed by this command is not saved in the nonvolatile memory of the scanner.

Command name: **MSG_SET_LASERS_INTENSITY**

Scanner command:

Type		Length		intens0		intens1		intens2		intens3	
5	0	8	0	0	0	0	0	0	0	0	0

Parameters:
- intens0: the intensity of the first (main) laser;
- intens1: the intensity of the second (additional) laser;
- intens2: the intensity of the third (additional) laser;
- intens3: the intensity of the fourth (additional) laser.

Scanner response:

Type		Length	
5	0	0	0

Parameters:
- none.

31.2.3. Setting the exposure time

This command sets the exposure time of the frame by the CMOS sensor. The value must be transmitted in milliseconds. It is possible to set the exposure time up to 3 frames (in multiple exposure operation mode of the scanner).

Command name: **MSG_SET_SENSOR_EXPOSURE**

Scanner command:

Type		Length		exp0		exp1		exp2	
6	0	6	0	0	0	0	0	0	0

Parameters:
- exp0: exposure time for the first frame;
- exp1: exposure time for the second frame (in multiple exposure mode);
- exp2: exposure time for the third frame (in multiple exposure mode).

Scanner response:

Type		Length	
6	0	0	0

Parameters:
- none.

31.2.4. Turning on the laser

Turning on the laser. If the laser is already on, nothing will happen. The intensity specified in the scanner settings (using the MSG_SET_LASERS_INTENSITY command or some other way) will be applied. Please note that the forced shutdown of laser radiation for safety reasons (a special signal on the scanner connector) takes precedence over all other controls.

Command name: **MSG_SET_LASER_ON**

Scanner command:

Type		Length	
7	0	0	0

Parameters:
- none.

Scanner response:

Type		Length	
7	0	0	0

Parameters:
- none.

31.2.5. Turning off the laser

Turning off the laser. If the laser is already off, nothing will happen.
Command name: **MSG_SET_LASER_OFF**
Scanner command:

Type		Length	
8	0	0	0

Parameters:
- none.

Scanner response:

Type		Length	
8	0	0	0

Parameters:
- none.

31.2.6. Setting the region of interest (ROI)

This command allows the user to set the size and position of the working area of the CMOS sensor. Reducing the size of the area increases the operating frequency of the scanner.

The pair "X1, Y1" sets the top left position of the ROI, and the pair "X2, Y2" sets the bottom right position of the ROI.

Command name: **MSG_SET_SENSOR_ROI**
Scanner command:

Type		Length		X1		Y1		X2		Y2		NU		NU	
12	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0

Parameters:
- X1: the X coordinate of the top left point of the ROI - currently not used, will be ignored;
- Y1: the Y coordinate of the top left point of the ROI;
- X2: the X coordinate of the bottom right point of the ROI - currently not used, will be ignored;
- Y2: the Y coordinate of the bottom right point of the ROI;
- NU: not used.

Scanner response:

Type		Length	
12	0	0	0

Parameters:
- none.

31.2.7. Getting device status

This command is intended to get information about the status of the device.
Command name: **MSG_GET_SENSOR_STATUS**
Scanner command:

Type		Length	
8	0	0	0

Parameters:
- none.

Scanner response:

Type		Length		mode		pad1[0]		...	pad1[15]		temp1		temp2	
15	0	2	50	0	0	0	0	...	0	0	0	0	0	0

temp3		heater		pad2[0]		...	pad2[15]		gain		exp		NU	
0	0	0	0	0	0	...	0	0	0	0	0	0	0	0

X1		Y1		X2		Y2		pad3[0]		...	pad3[15]		las		int0
0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0

int1	int2	int3	NU	NU	NU	NU	pad4[0]		...	pad4[15]		seam		NU	
0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0

NU[40]	pad5[0]		...	pad5[63]		NU		pad6[0]		...	pad6[127]	
0	0	0	....	0	0	0	0	0	0	...	0	0

Parameters:
- mode: scanner operation mode (in the current version, it is always 0 - measurement output mode);
- pad1: separator-reserve;
- temp1: CPU temperature, the value is calculated as 100*(temperature in °C) + 10000;
- temp2: internal temperature of the scanner (sensor #1), the value is calculated as 100*(temperature in °C) + 10000;
- temp3: internal temperature of the scanner (sensor #2), the value is calculated as 100*(temperature in °C) + 10000;
- heater: heating state (in the current version - 0);
- pad2: separator-reserve;
- gain: amplification of the CMOS sensor signal (in the current version, it is always 0);
- exp: exposure time (for the first frame) in ms;
- NU: not used;
- X1: the X coordinate of the top left point of the ROI - currently not used, will be ignored;
- Y1: the Y coordinate of the top left point of the ROI;

- X2: the X coordinate of the bottom right point of the ROI - currently not used, will be ignored;
- Y2: the Y coordinate of the bottom right point of the ROI;
- pad3: separator-reserve;
- las: laser state (0 - off, 1 - on), this parameter is not affected by the hardware shutdown of the laser;
- int0: radiation intensity of laser #1 (main);
- int1: radiation intensity of laser #2 (additional);
- int2: radiation intensity of laser #3 (additional);;
- int3: radiation intensity of laser #4 (additional);
- pad4: separator-reserve;
- seam: template index;
- pad5: separator-reserve;
- pad6: separator-reserve.

31.2.8. Setting the welding template

This command sets the index of the template used for finding a welding joint and outputting its attributes.

Command name: **MSG_SET_BASIC_SEAM_TYPE**

Scanner command:

Type		Length		idx	
40	0	2	0	0	0

Parameters:
- idx: template index.

Scanner response:

Type		Length	
40	0	0	0

Parameters:
- none.

31.2.9. Getting the firmware version of the scanner

This command requests the firmware version of the scanner. The scanner will respond with the firmware version (three integers, major, minor and patch version codes).

Command name: **MSG_GET_FIRMWARE_VERSION**

Scanner command:

Type		Length	
100	0	0	0

Parameters:
- none.

Scanner response:

Type		Length		major		minor		patch	
100	0	6	0	2	0	3	0	3	0

Parameters:
- major: major version code;
- minor: minor version code;
- patch: patch version code.

31.2.10. Getting the temperature of the scanner

This command requests the internal temperature of the scanner. A sensor installed on the CPU is used, because the CPU is the most heat-generating element.

Command name: **MSG_GET_MAIN_BD_TEMP**

Scanner command:

Type		Length	
105	0	0	0

Parameters:

- none.

Scanner response:

Type		Length		temp	
105	0	2	0	0	0

Parameters:

- temp: CPU temperature, the value is calculated as 100*(temperature in °C) + 10000.

31.2.11. Start sending measurement results

This command allows sending measurement results (welding joint parameters). The index of the welding joint is specified by the MSG_SET_BASIC_SEAM_TYPE command. The list of parameters depends on the type of the selected joint. It is possible to send up to 16 points and 16 parameter values. Each point has a status indicating whether it is used for this type of joint and whether its data is valid.

Command name: **MSG_START_MEASUREMENT_SENDING_IN_MM**

Scanner command:

Type		Length	
150	0	0	0

Parameters:

- none.

Scanner response:

Type		Length	
150	0	0	0

Parameters:

- none.

Scanner message (sent after each measurement performed):

Type		Length		timestamp				pt[0].x				pt[0].y			
150	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

pt[0].st				...	pt[15].x				pt[15].y			
0	0	0	0	...	0	0	0	0	0	0	0	0

pt[15].st				prm[0].val				prm[0].st				…
0	0	0	0	0	0	0	0	0	0	0	0	…

prm[15].val				prm[15].st				pad[0]		…	pad[63]	
0	0	0	0	0	0	0	0	0	0	…	0	0

Parameters:
- timestamp: system timestamp of measurement results generation, ms;
- pt[0].x: X coordinate of point #1 (float, mm);
- pt[0].z: Z coordinate of point #1 (float, mm);
- pt[0].st: status of point #1;

...
- pt[15].x: X coordinate of point #16 (float, mm);
- pt[15].z: Z coordinate of point #16 (float, mm);
- pt[15].st: status of point #16;
- prm[0].val: value of parameter #1 (depends on the template) - not used in the current version;
- prm[0].st: status of parameter #1 - not used in the current version;

…
- prm[15].val: value of parameter #16 (depends on the template) - not used in the current version;
- prm[15].st: status of parameter #16 - not used in the current version;
- pad: separator-reserve.

Statuses (for the current version of the protocol): 0 - point/parameter data is up-to-date; 2 - point/parameter data is not up-to-date and should not be used.

31.2.12. Stop sending measurement results

This command stops sending measurement results.
Command name: **MSG_STOP_MEASUREMENT_SENDING_IN_MM**
Scanner command:

Type		Length	
151	0	0	0

Parameters:
- none.

Scanner response:

Type		Length	
151	0	0	0

Parameters:
- none.

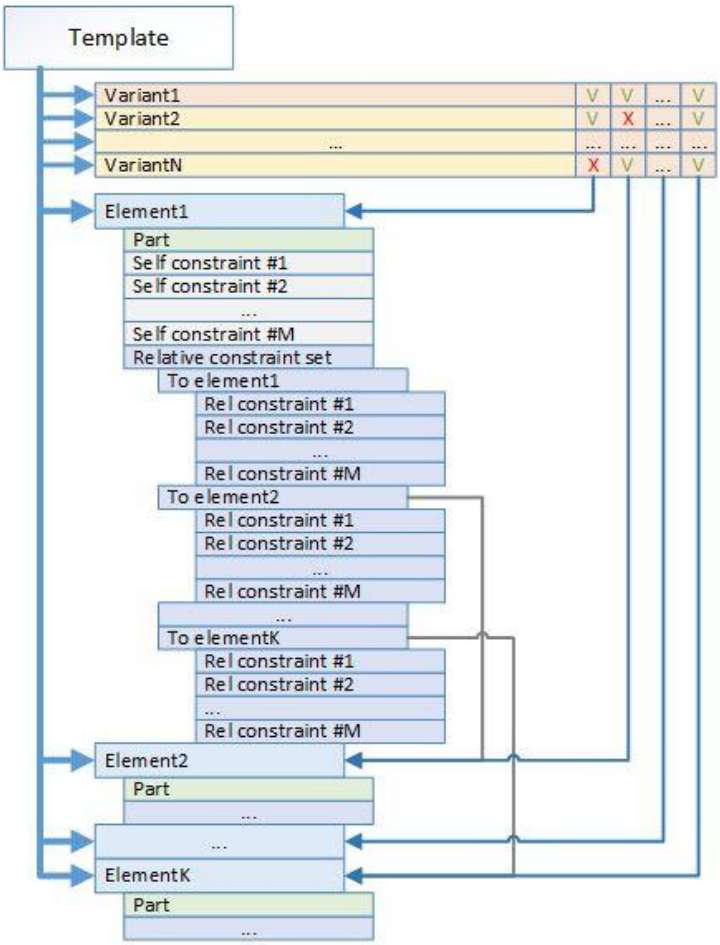
32. Annex 6. "Template detector" smart block and Template Editor

32.1. General information

The “template detector” smart block is designed to detect a user-defined template in the profile. The template is formed by the user using a special editor described below. The current profile approximation or a user-drawn sequence of line segments can be used as a basis for creating a template.

32.2. Template structure and search principle

A template is a set of elements identical to the elements obtained after approximating the profile. Each element includes a description of a part of the profile (used only for graphical display of an element in the editor), as well as a set of self constraints and relative constraints. In addition, the template contains a description of the variants that determine the acceptable absence of elements. The structure of the template can be represented as follows:



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

N	8	The maximum number of template variants. There is always variant #1 in the template - it contains all the elements.
K	16	The maximum number of elements in the template.
M	8	The maximum number of constraints for each element, separately for self constraints and relative constraints.

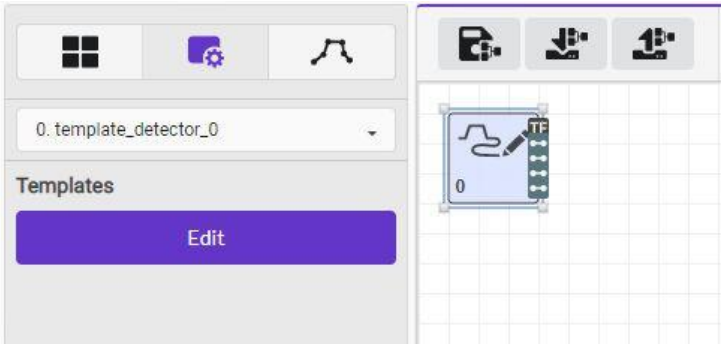
The search for a template in a profile is based on a sequential check of self constraints and relative constraints for the elements of the profile approximation. The search starts for the first variant, and if at least one constraint is not met, the check stops and the next variant will be checked. If 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 about segments (or arcs - in the future).

32.3. Template editor

32.3.1. Description of interface elements

To open the template editor, you need to open the parameters of the "template detector" smart block and click the **Edit** button.



A modal window of the visual template editor built into the web interface will appear:

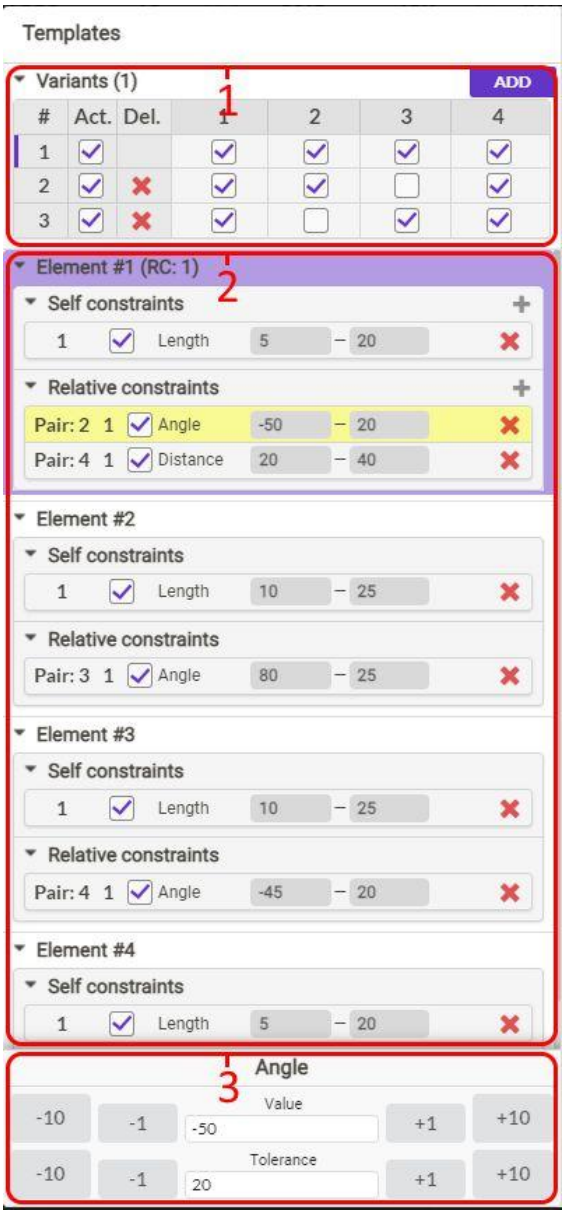
The template editor window is divided into the following functional areas:

1 - The area intended for displaying the template and setting its parameters.

2 - The area containing common controls.

3 - The area intended for visualizing a template and its constraints, visualizing a profile and displaying the template detection results for the current profile, and selecting the elements to be edited.

32.3.1.1. Area for displaying and configuring template parameters



This area contains three sections:

1 - Template variants. This section allows the user to add, remove, and change variants. Important: the first variant must always exist and include all elements of the template.

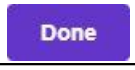
2 - A list of elements of the current template and the specified self and relative constraints. Clicking on the constraint displays it in the visualization area of the template, and also gives access to editing parameters.

3 - Constraint parameters.

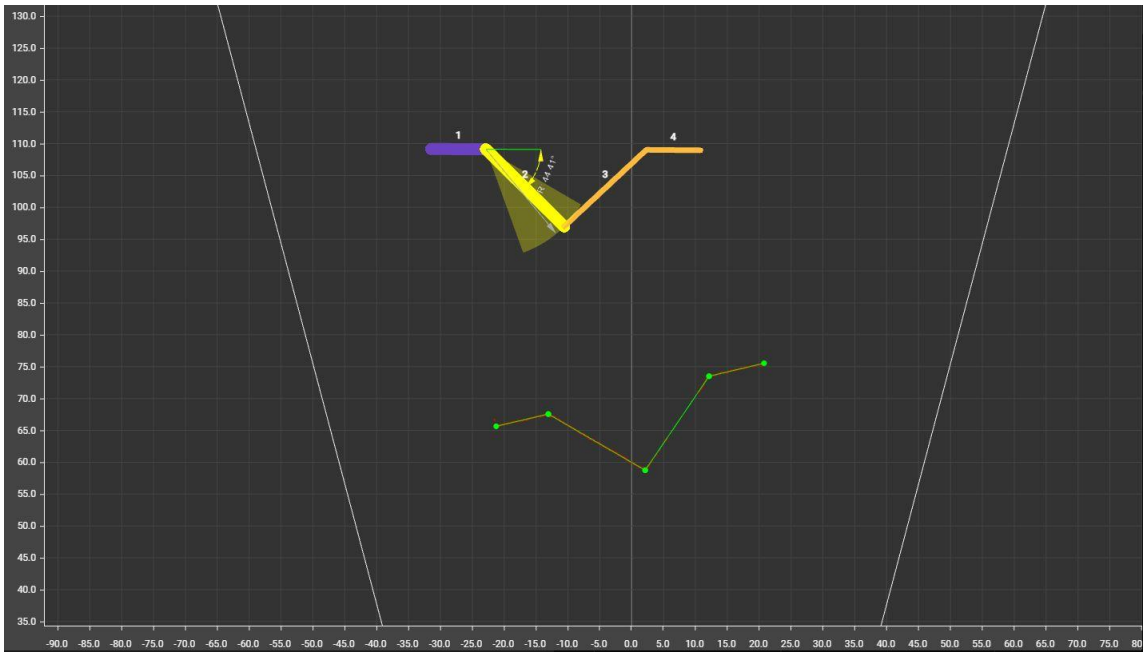
32.3.1.2. Area of controls

This area contains controls for creating, editing and deleting template elements.

Buttons:

Button	Assignment
	Selecting template elements. In this mode, the template element is selected by clicking on it.
	Drawing template elements. In this mode, adding a point for drawing template elements occurs when you click in the coordinate plane.
	Reset drawn lines in drawing mode.
	Create a line in drawing mode.
	Delete the selected template element.
	Automatic generation of template elements from approximated profile segments.
	Turn on/off the display of approximated profile elements.
	Clear the template.
	Save the template in *.json format.
	Upload the template to the scanner.
	Close the template editor.

32.3.1.3. Area for displaying the template and its constraints



The area is intended for:

- displaying template elements (gives an idea of the shape of the template);
- visualizing the selected constraint (selection is made in the area of template parameters);
- displaying profile points;
- displaying the results of profile approximation (by clicking the **Lines** button).

If a template is detected in the current profile, the position of the template will be shown with green lines.

32.3.2. Working with the template editor

The Template Editor allows the user to create, view, change, and clean up a template.

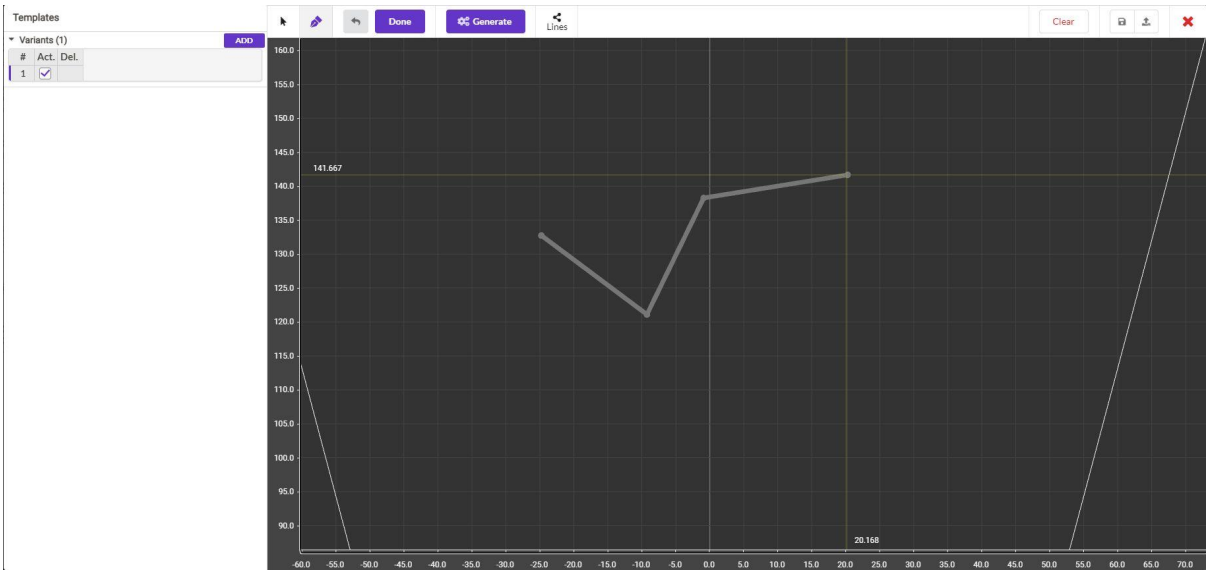
32.3.2.1. Creating template elements

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.

32.3.2.1.1. Creation of template elements by the user

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

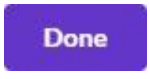
When you click on the coordinate grid, points forming a polyline appear:



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:

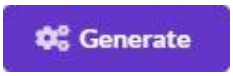
To create the template element, click **Done**:



The created elements are sent to the scanner and the search for these elements starts. 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.

32.3.2.1.2. Automatic creation of template elements

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



In this case, template elements are created based on the results of approximation of the current profile. 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.

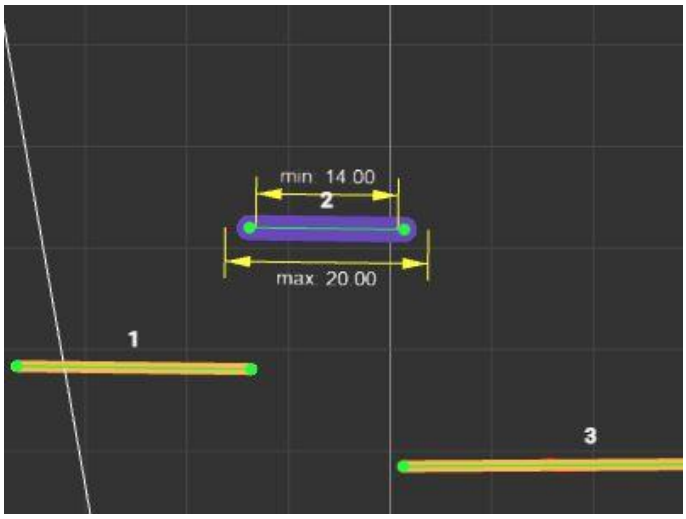
32.3.2.2. Creating constraints

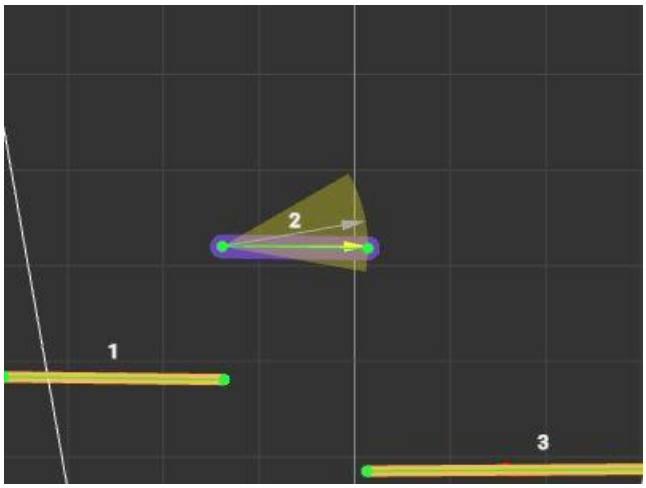
After creating the template elements, it is necessary to add constraints that make it possible to take into account the shape of the template, i.e. permissible change of elements and permissible ratio between pairs of elements.

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

32.3.2.2.1. Self constraints

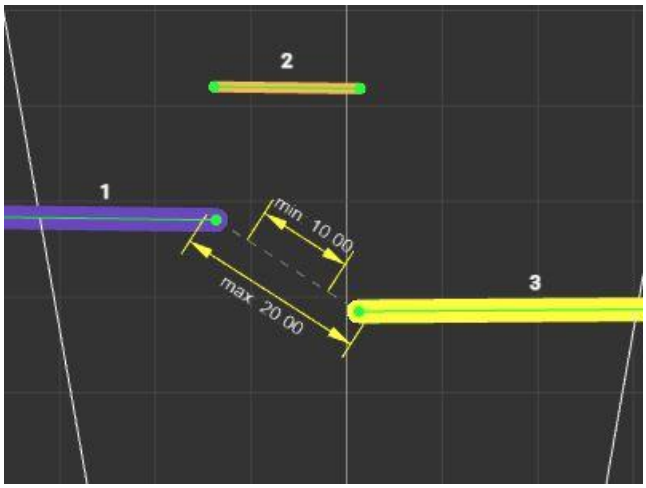
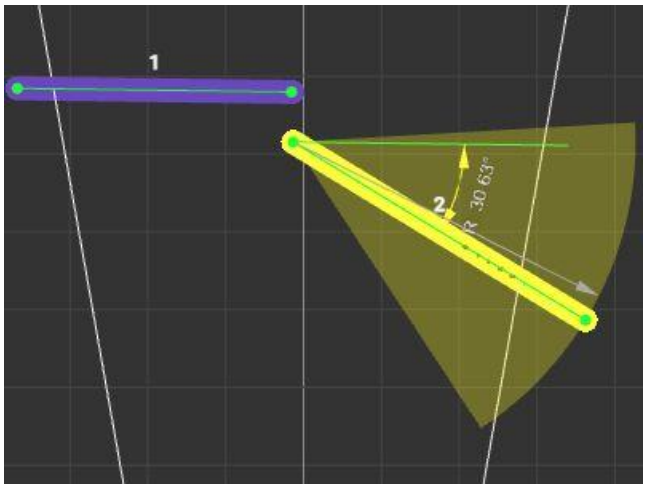
Self constraints are applied directly to the template element. The following self constraints are provided:

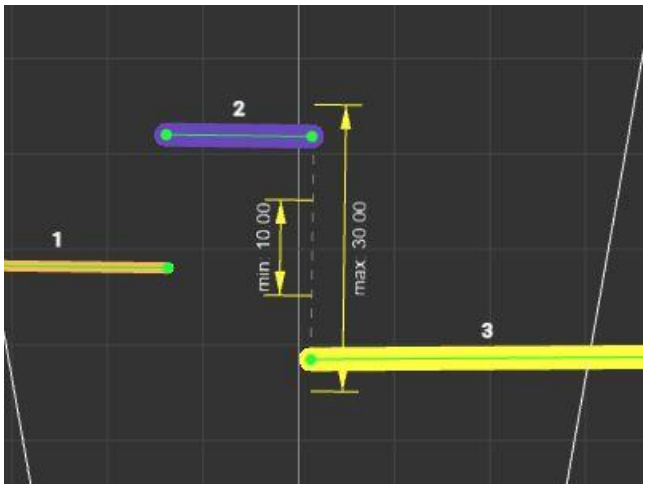
Length	The length of the element (sets the minimum and maximum allowed length of the element). Parameters: • min - minimum allowable length; • max - maximum allowable length. 
Angle	The angle of inclination relative to the horizontal axis. Parameters: • value - target angle relative to the horizontal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



32.3.2.2.2. Relative constraints

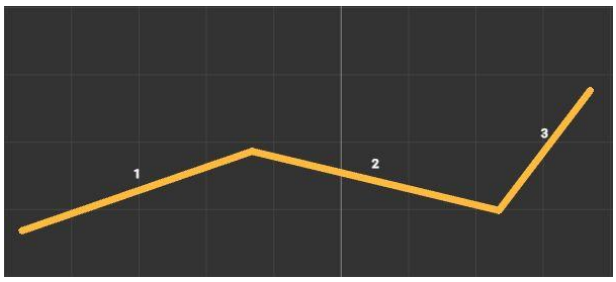
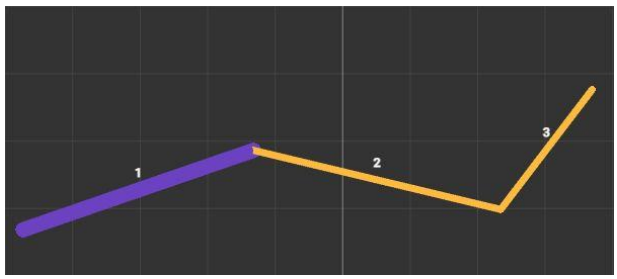
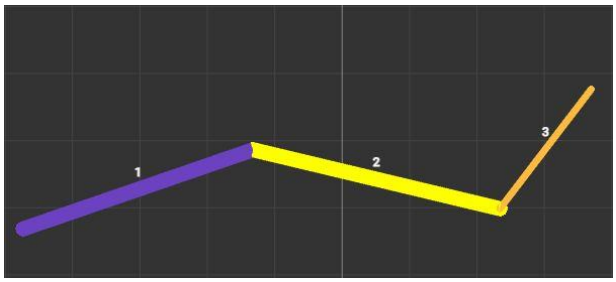
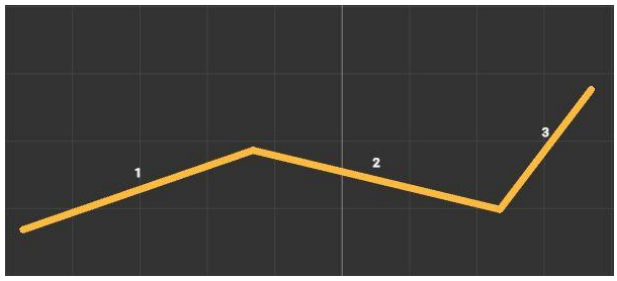
Relative constraints are applied only to a pair of elements and are set from the main element to the relative one. The following relative constraints are provided:

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 direction of the main element at the beginning of the relative element is indicated when the angle is denoted by a green line): 

	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.

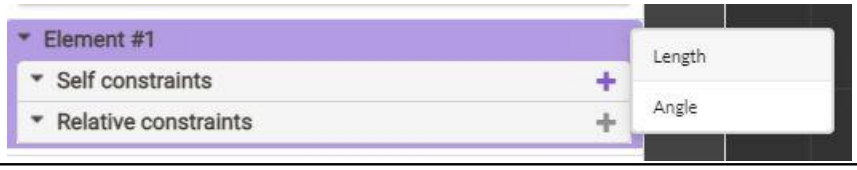
32.3.2.3. Adding constraints to a template

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:

When you click ➕, a drop-down list appears:



Adding a self constraint

Adding a relative constraint

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

For relative constraints, the template editor displays the number of the relative element for which this constraint is specified.

The checkbox next to the constraint enables or disables checking of this constraint when searching for a template.

The numeric fields define the parameters of the constraint and depend on the constraint type.

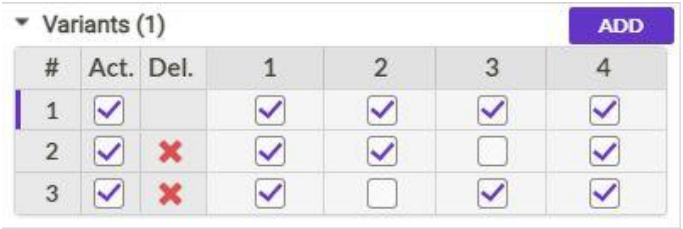
To delete a constraint, click ✖ for the constraint you need to delete.

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:

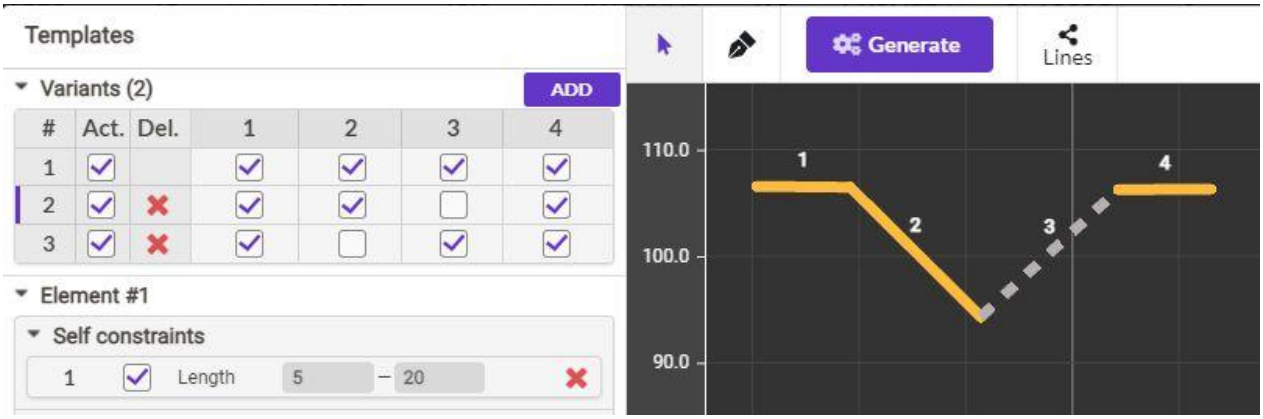
32.3.2.4. Creating and customizing template variants

The template variant determines which elements of the template may be missing. When displaying variants, the number of the variant is shown along the vertical axis and the number of the element along the horizontal axis:

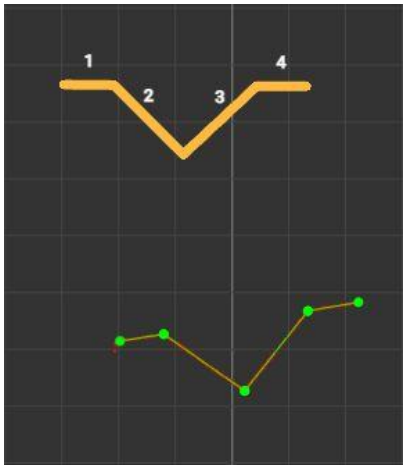


The checkbox in the **Act.** column indicates whether this variant is used in matching. The ✖ button deletes the corresponding variant.

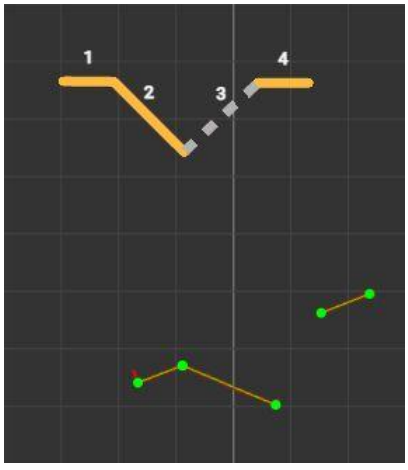
Clicking on the variant number selects that variant for display. Inactive elements of the template are displayed with a dashed line - this means that the element is not necessary for this variant of the template:



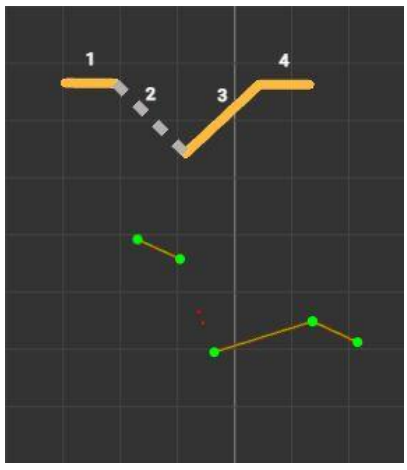
Examples of different variants of the same template:



Variant 1 - All lines are active



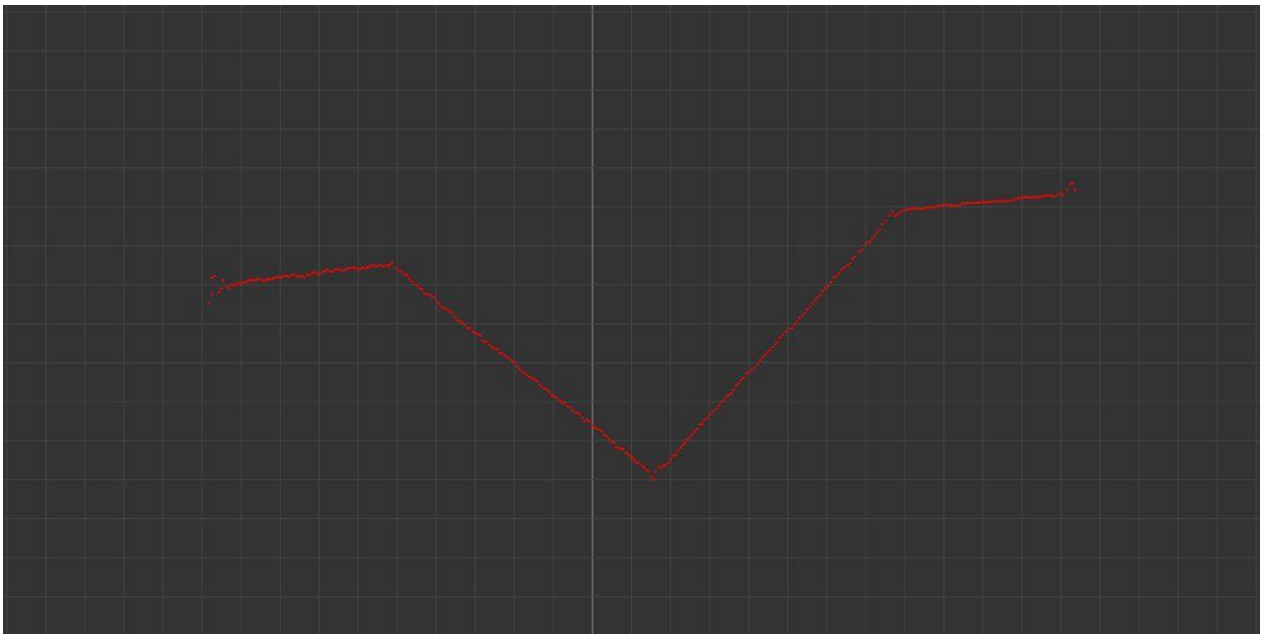
Variant 2 -. Line 3 is not active



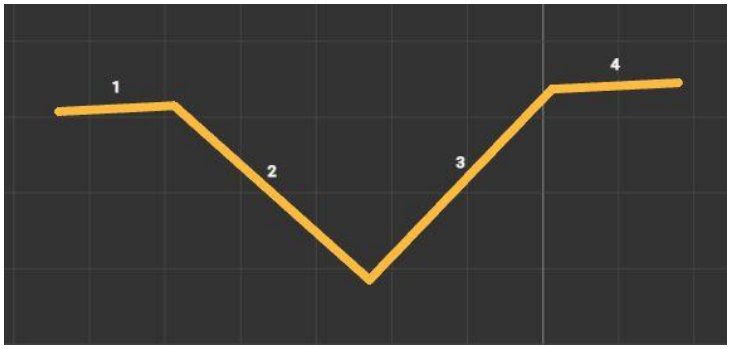
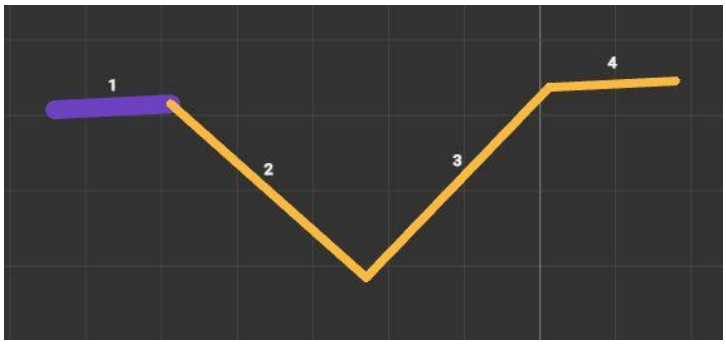
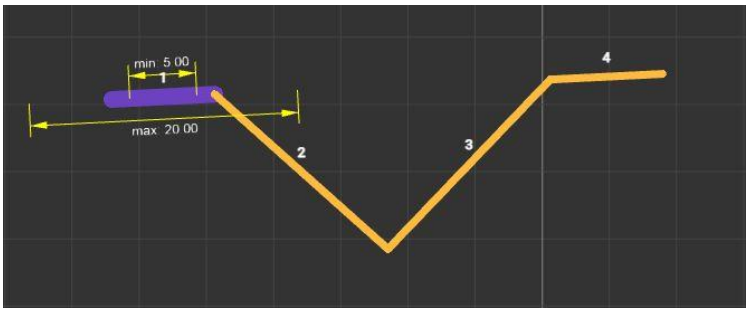
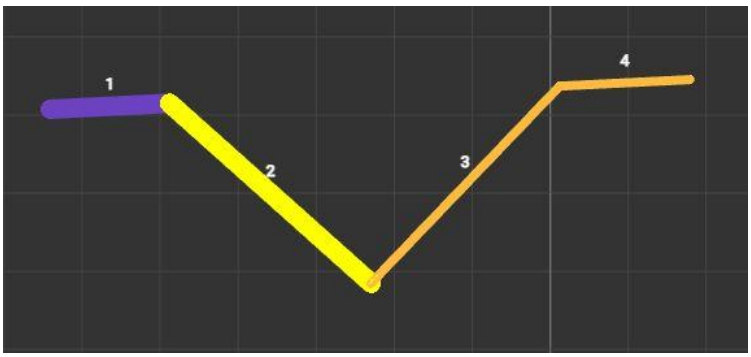
Variant 3 - Line 2 is not active

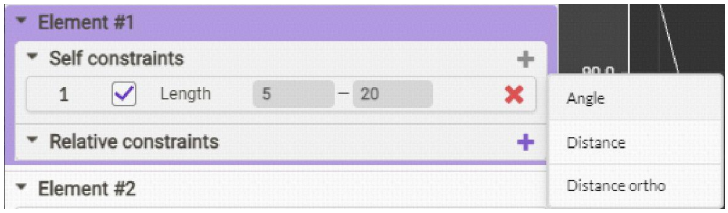
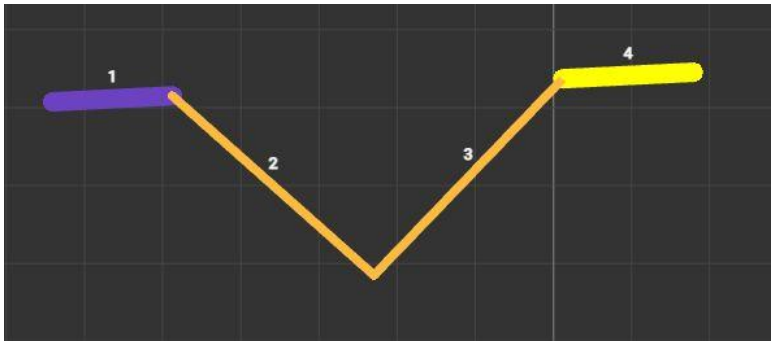
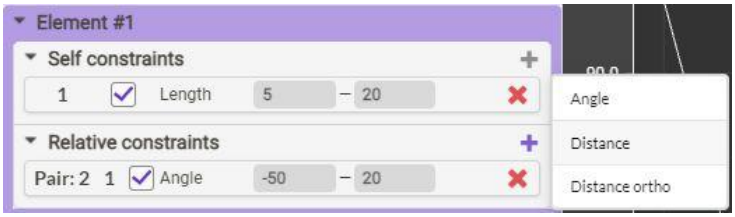
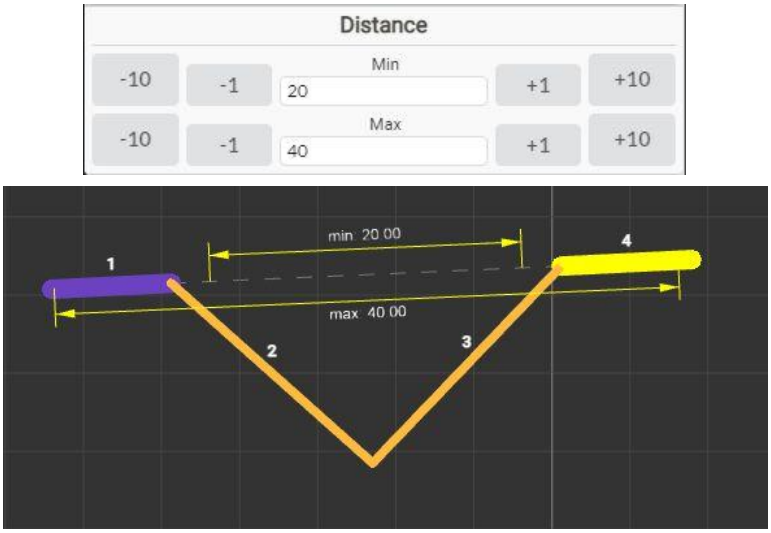
32.3.3. Example of creating a template

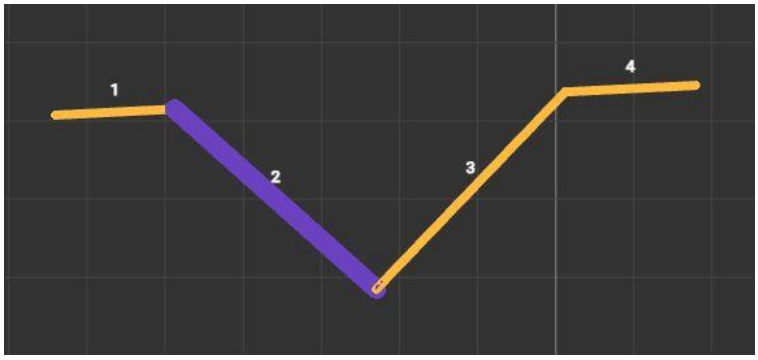
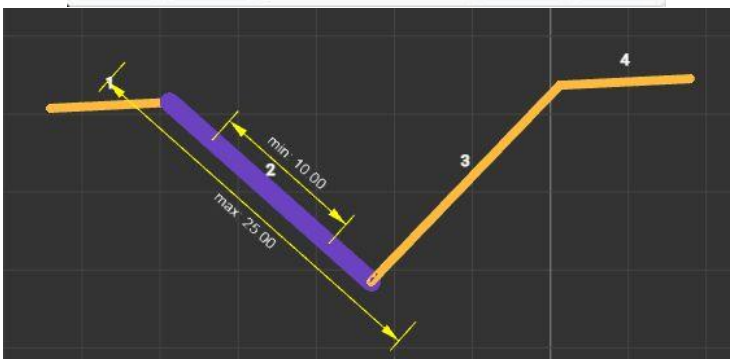
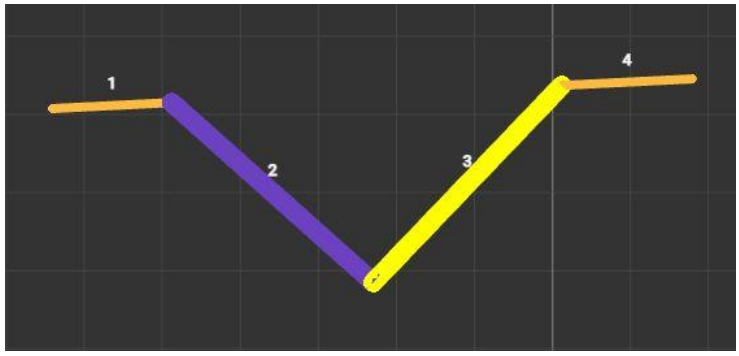
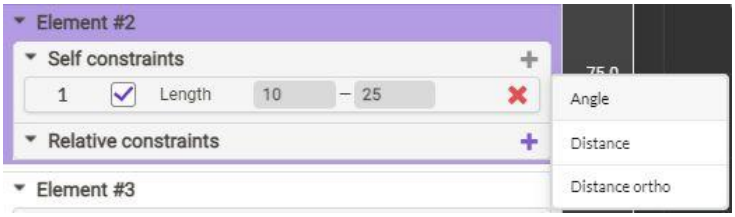
Below is a sequence of steps to create a template that describes the shape of the profile shown in the screenshot.

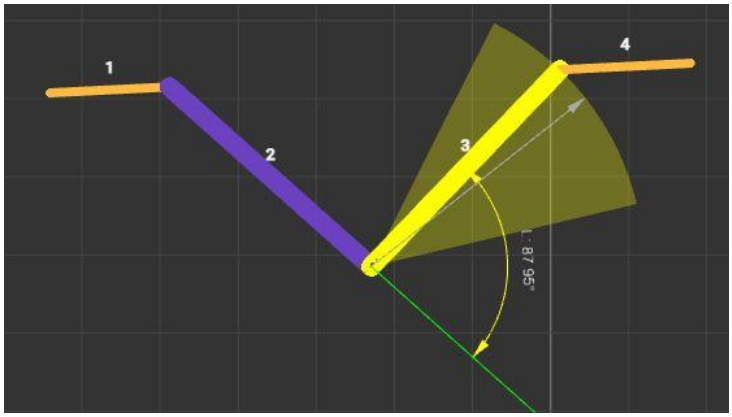
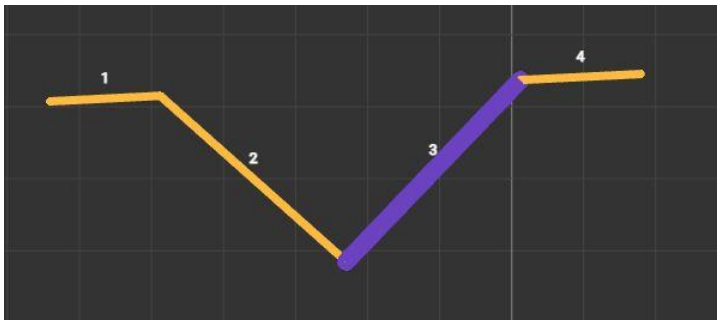
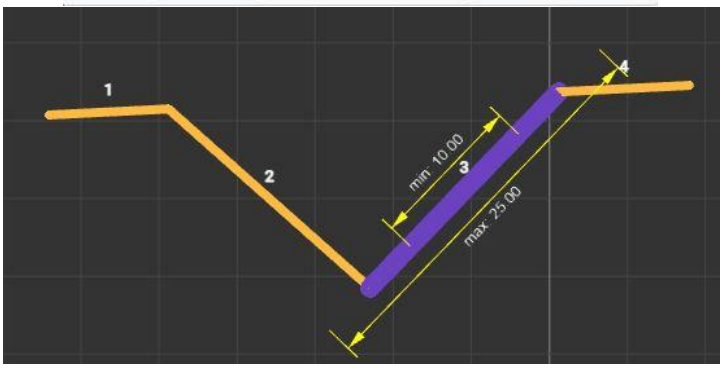
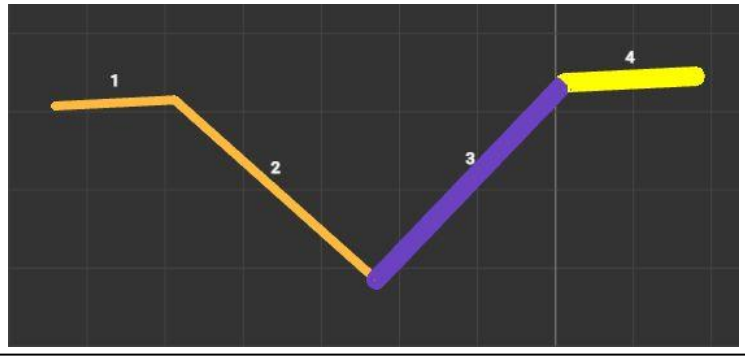


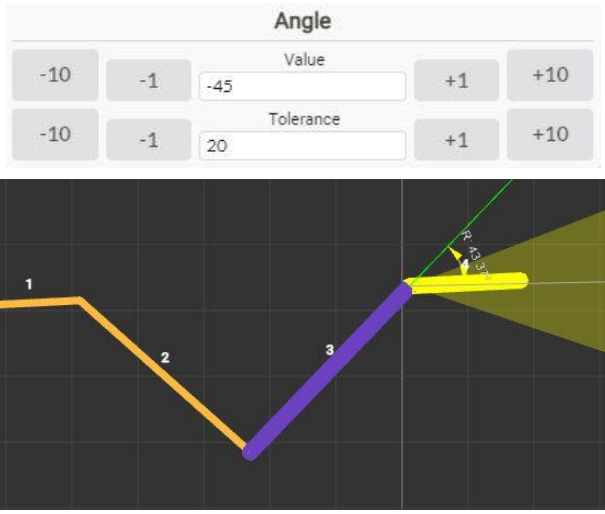
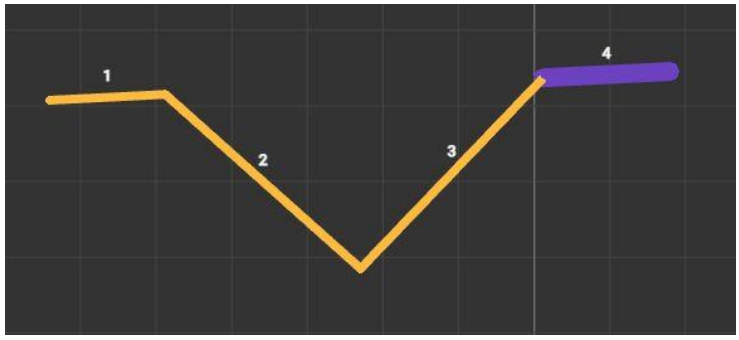
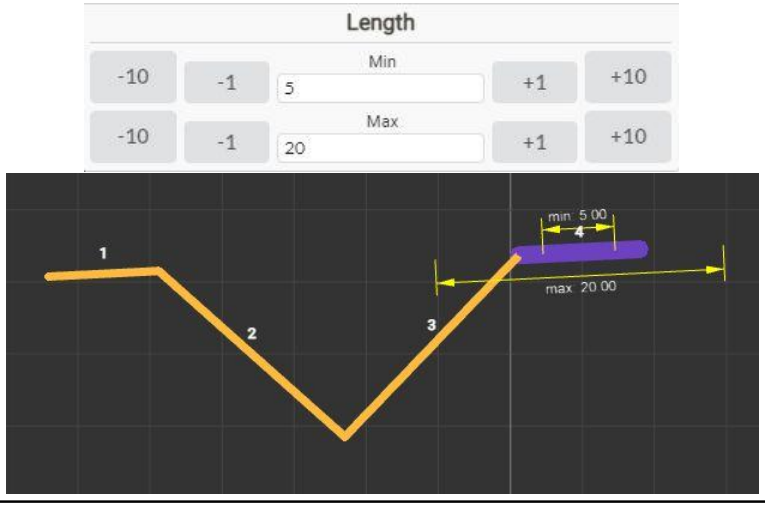
Follow these steps:

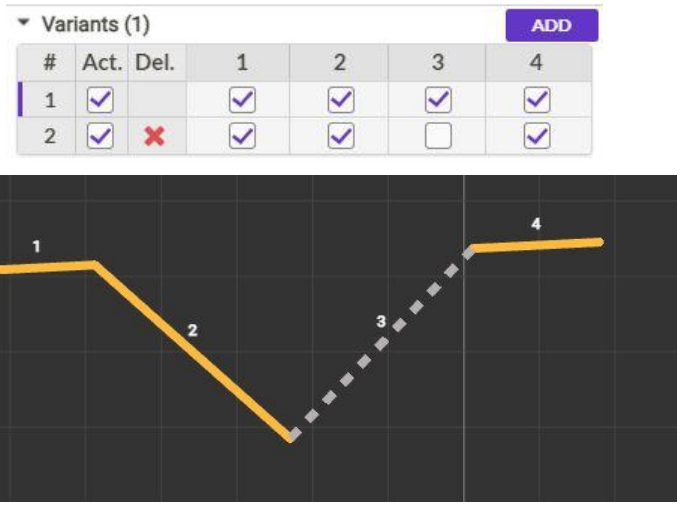
#	Description	Screenshot
	Creating template elements	
1	Click the **Generate** button to automatically create template elements or draw them manually according to the profile lines.	
	Adding constraints for element 1	
1	Select segment 1 - in this case, it will become the main element.	
2	In the "Self constraints" section of element 1, add a constraint of the "Length" type.	
3	Select the added constraint in the "Self constraints" list and specify the minimum and maximum length of the line (input fields will appear below the description of the template structure).	 
4	Select segment 1 and then immediately select segment 2 - in this case, it will become a relative element.	

#	Description	Screenshot
5	In the "Relative constraints" section of element 1, add a constraint of the "Angle" type.	
6	Select the added constraint in the "Relative constraints" list and specify the target angle and tolerance. When determining the target angle, it is necessary to roughly align the gray arrow with the relative element. The numerical value of the current angle between the elements is also shown. When specifying a tolerance, the displayed sector must include a relative element.	
7	Reset the previous selection by clicking in an empty area (which does not contain any elements of profile approximation). Select segment 1 and then immediately select segment 4 - in this case, it will become a relative element.	
8	In the "Relative constraints" section of element 1, add a constraint of the "Distance" type.	
9	Select the added constraint in the "Relative constraints" list and set the minimum and maximum distance values.	

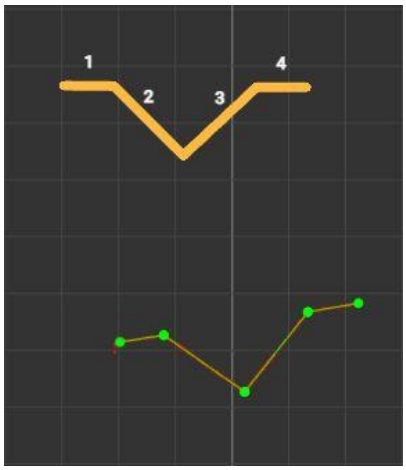
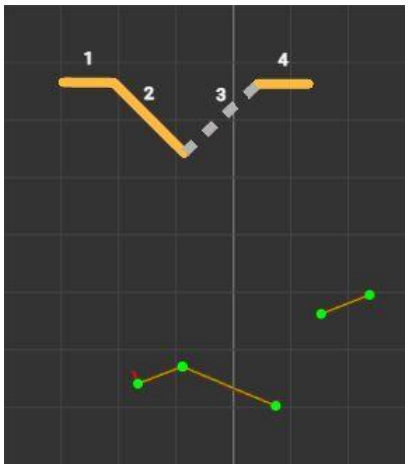
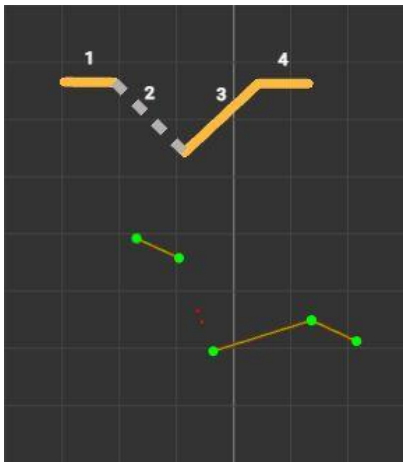
#	Description	Screenshot
	Adding constraints for element 2	
1	Reset the previous selection by clicking in an empty area (which does not contain any elements of profile approximation). Select segment 2 - in this case, it will become the main element.	
2	In the "Self constraints" section of element 2, add a constraint of the "Length" type.	
3	Select the added constraint in the "Self constraints" list and specify the minimum and maximum length of the segment.	 
4	Reset the previous selection. Select segment 2 and then immediately select segment 3 - in this case, it will become a relative element.	
5	In the "Relative constraints" section of element 2, add a constraint of the "Angle" type.	
6	Select the added constraint in the "Relative constraints" list and specify the target angle and tolerance.	

#	Description	Screenshot
		
	Adding constraints for element 3	
1	Reset the previous selection. Select segment 3 - in this case, it will become the main element.	
2	In the "Self constraints" section of element 3, add a constraint of the "Length" type.	
3	Select the added constraint in the "Self constraints" list and specify the minimum and maximum length of the segment.	 
4	Reset the previous selection. Select segment 3 and then immediately select segment 4 - in this case, it will become a relative element.	

#	Description	Screenshot
5	In the "Relative constraints" section of element 3, add a constraint of the "Angle" type.	
6	Select the added constraint in the "Relative constraints" list and specify the target angle and tolerance.	
Adding constraints for element 4		
1	Reset the previous selection. Select segment 4 - in this case, it will become the main element.	
2	In the "Self constraints" section of element 4, add a constraint of the "Length" type.	
3	Select the added constraint in the "Self constraints" list and specify the minimum and maximum length of the segment.	
Adding variants		

#	Description	Screenshot
1	Add a new variant by clicking the "ADD" button in the "Variants" section and make element 3 inactive (uncheck the box).	
2	Add a new variant by clicking the "ADD" button in the "Variants" section and make element 2 inactive (uncheck the box).	

As a result, the template will be detected taking into account all the variants:

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

33. Annex 7. "C-script" smart block

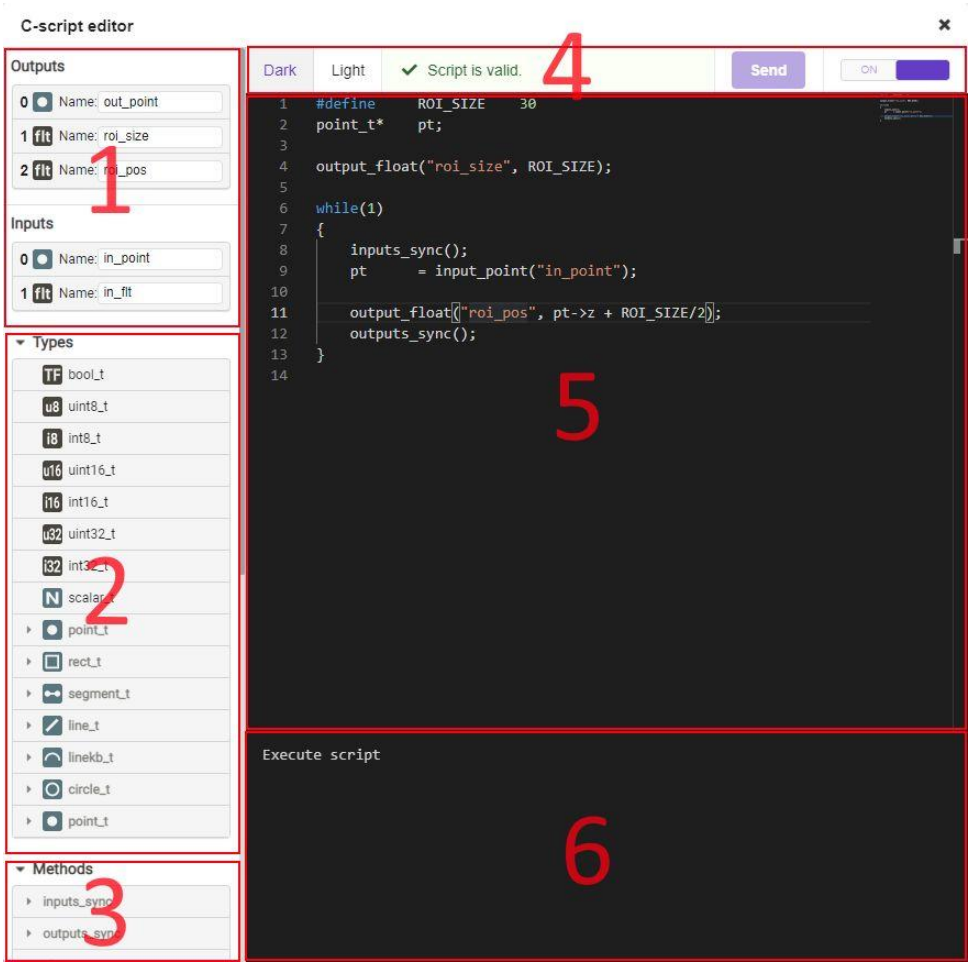
33.1. General information

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.

33.2. 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; }
circle_t	12	Circle specified by center coordinates and radius: { point_t center; 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;

Type	Size, byte	Description
		float r; bool_t convex; }

33.3. Supported methods

33.3.1. 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)	

33.3.2. Special methods

Special methods do not require the connection of additional modules.

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.

bool_t input_bool(char* portName)
float input_float(char* portName)
scalar_t input_scalar(char* portName)
point_t* input_point(char* portName)

rect_t* input_rect(char* portName)
segment_t* input_segment(char* portName)
line_t* input_line(char* portName)
circle_t* input_circle(char* portName)
arc_t* input_arc(char* portName)

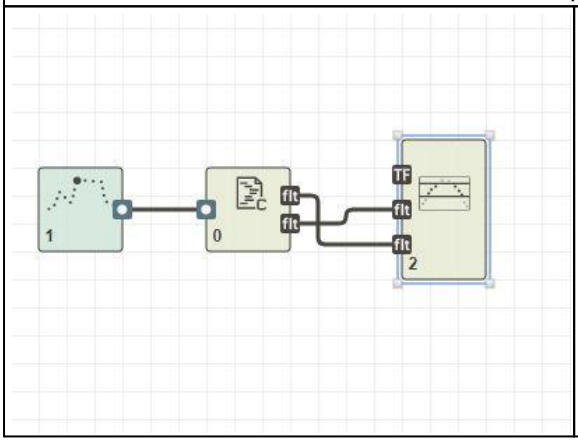
- Reading data from the input port named “portName”. If there is no port with this name or a type mismatch, the return value may be invalid.

void output_bool(char* portName, bool_t val)
void output_float(char* portName, float val)
void output_int(char* portName, int val)
void output_scalar(char* portName, scalar_t val)
void output_point(char* portName, point_t* val)
void output_rect(char* portName, rect_t* val)
void output_segment(char* portName, segment_t* val)
void output_line(char* portName, line_t* val)
void output_circle(char* portName, circle_t* val)
void output_arc(char* portName, arc_t* val)

- Writing data to the output port named “portName”. If there is no port with this name or a type mismatch, the value will be invalid.

33.4. Examples of scripts

ROI position control




```
#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();
}
```

34. Warranty policy

Warranty assurance for Laser Scanners RF627 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.

35. 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: support@riftek.com
- Skype: riftek_support

36. 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. 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.