ibaPDA v8.5.0

New Features

08.11.2023
iba AG

Table of contents

1 General remarks

1.1 Supported Windows Operating Systems

For **Windows 8.1 (x86/x64)**, **Windows Server 2012 (x64)** and **Windows Server 2012 R2 (x64)** Microsoft ended the Extended Support period on Jan 10, resp. Oct 10, 2023. No security updates are provided by Microsoft anymore since then.

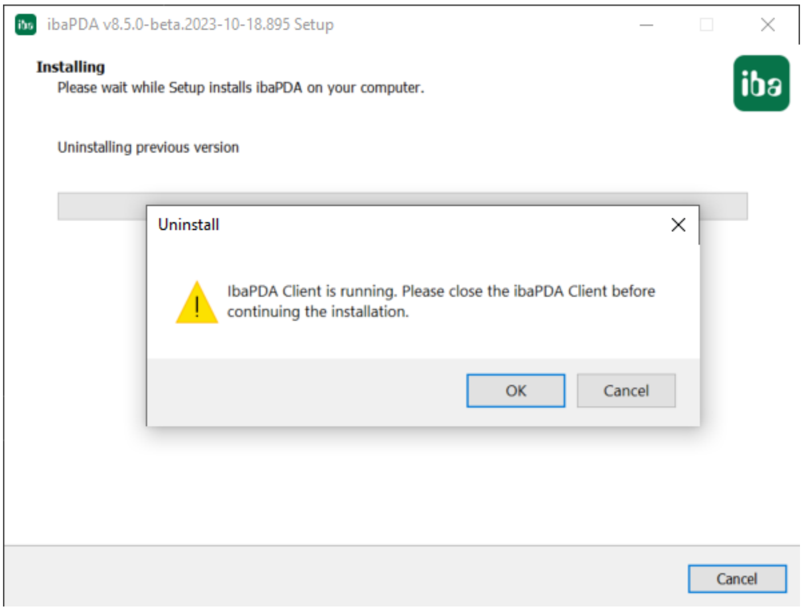
Although installation is technically still possible, ibaPDA v8.5.0 and higher no longer support these operating systems. The possibility to install on these operation systems can be omitted without any further notice.

The following operating systems are currently supported:

- Windows 10 (x86/x64)
- Windows 11 (x64)
- Windows Server 2016 (x64)
- Windows Server 2019 (x64)
- Windows Server 2022 (x64)

2 Installer command line switch: ForceClientClose

When you update an existing ibaPDA installation with a new version then the existing ibaPDA is first uninstalled. During this uninstall it is checked if the ibaPDA client is running. If it is still running then a message box appears asking you to close the ibaPDA client manually.



ibaPDA v8.5.0-beta.2023-10-18.895 Setup

Installing

Please wait while Setup installs ibaPDA on your computer.

Uninstalling previous version

Uninstall

IbaPDA Client is running. Please close the ibaPDA Client before continuing the installation.

This message box is not shown if the installation is run in silent mode. The installation then just fails.

There is now a new command line switch for the installer: /ForceClientClose. When this switch is used then the installer will close the client automatically. This can lead to the loss of any changes to the configuration of the running client. The switch can be used in scripts used to install new ibaPDA versions automatically.

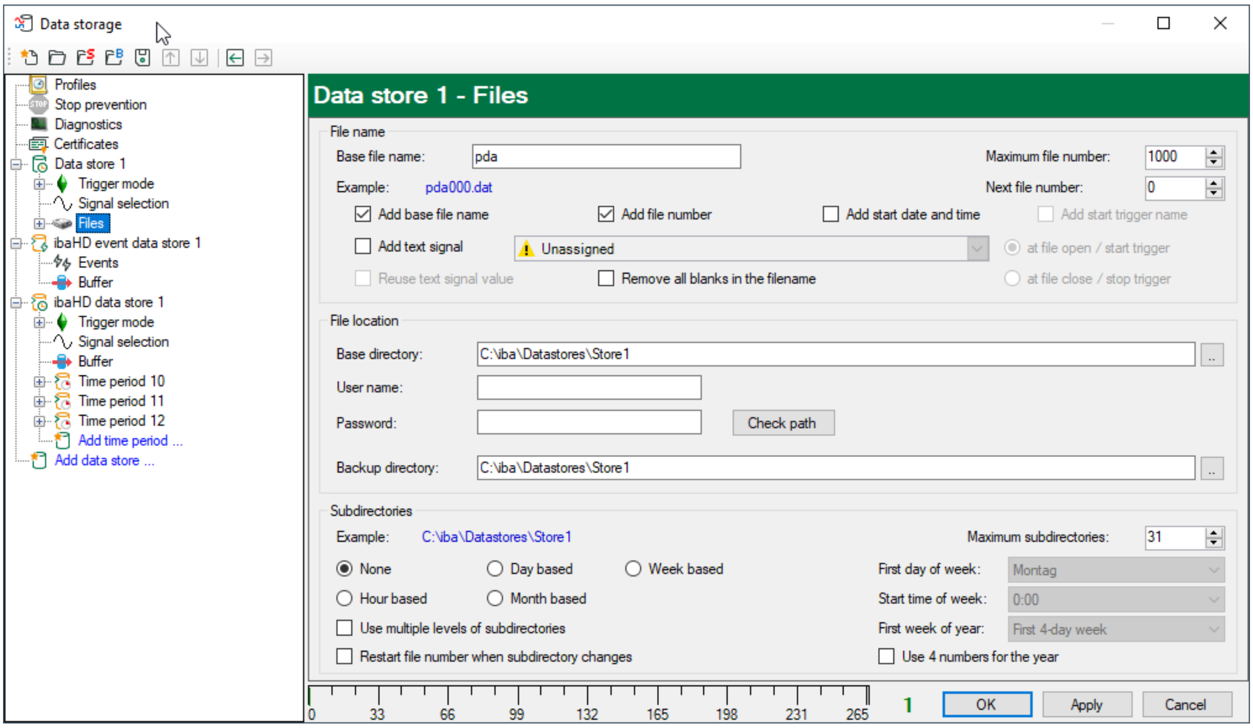
3 Default base directory changes

The default base directories of the following elements were changed.

3.1 Timebased data store

The default base directory was changed from *C:\dat* to *C:\iba\Datastores*.

For each timebased data store a sub folder will be created (Store1, Store2, etc.)



Data store 1 - Files

File name

Base file name: pda

Maximum file number: 1000

Example: pda000.dat

Next file number: 0

☑ Add base file name ☑ Add file number □ Add start date and time □ Add start trigger name

□ Add text signal ⚠ Unassigned ● at file open / start trigger

□ Reuse text signal value □ Remove all blanks in the filename ○ at file close / stop trigger

File location

Base directory: C:\iba\Datastores\Store1

User name:

Password: Check path

Backup directory: C:\iba\Datastores\Store1

Subdirectories

Example: C:\iba\Datastores\Store1

Maximum subdirectories: 31

● None ○ Day based ○ Week based

First day of week: Montag

○ Hour based ○ Month based

Start time of week: 0:00

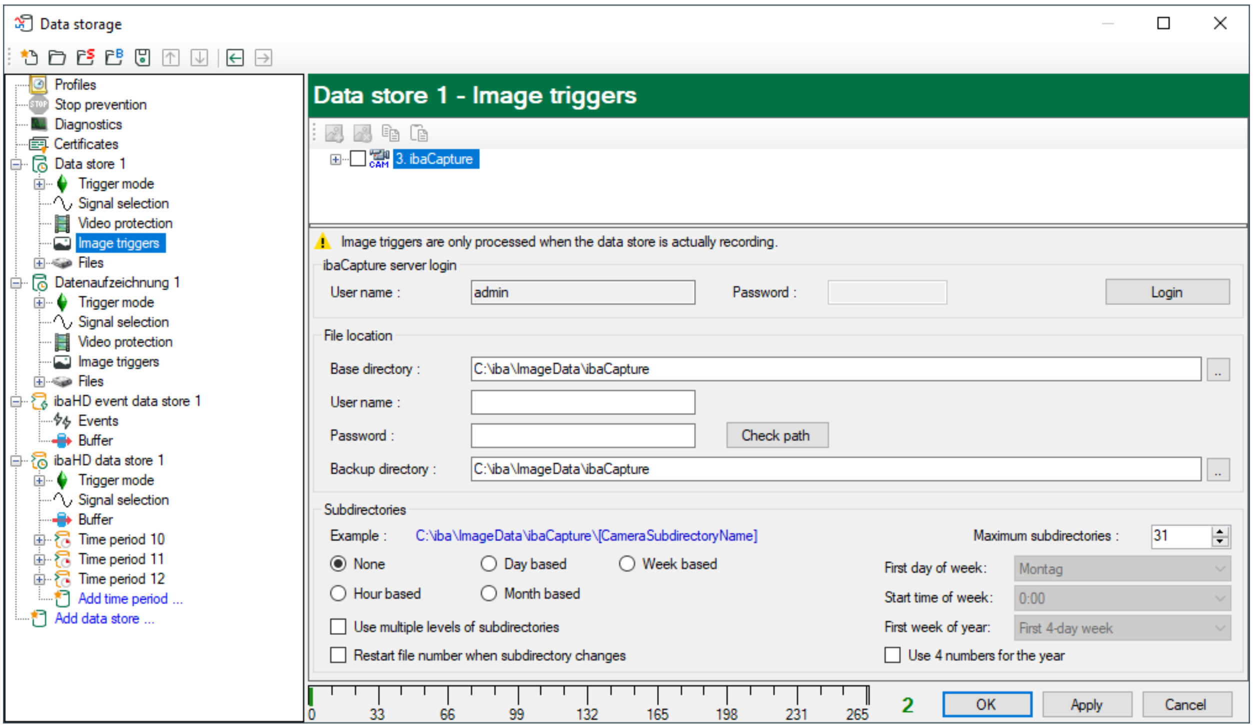
□ Use multiple levels of subdirectories

First week of year: First 4-day week

□ Restart file number when subdirectory changes □ Use 4 numbers for the year

3.2 Image triggers

The default base directory was changed from *C:\dat* to *C:\iba\ImageData*.



Data storage

- Profiles
- Stop prevention
- Diagnostics
- Certificates
- Data store 1
  - Trigger mode
  - Signal selection
  - Video protection
  - Image triggers
  - Files
- Datenaufzeichnung 1
  - Trigger mode
  - Signal selection
  - Video protection
  - Image triggers
  - Files
- ibaHD event data store 1
  - Events
  - Buffer
- ibaHD data store 1
  - Trigger mode
  - Signal selection
  - Buffer
  - Time period 10
  - Time period 11
  - Time period 12
  - Add time period ...
- Add data store ...

Data store 1 - Image triggers

□ 3. ibaCapture

Image triggers are only processed when the data store is actually recording.

ibaCapture server login

User name : admin Password : Login

File location

Base directory : C:\iba\ImageData\ibaCapture

User name :

Password : Check path

Backup directory : C:\iba\ImageData\ibaCapture

Subdirectories

Example : C:\iba\ImageData\ibaCapture\[CameraSubdirectoryName] Maximum subdirectories : 31

● None ○ Day based ○ Week based First day of week: Montag

○ Hour based ○ Month based Start time of week: 0:00

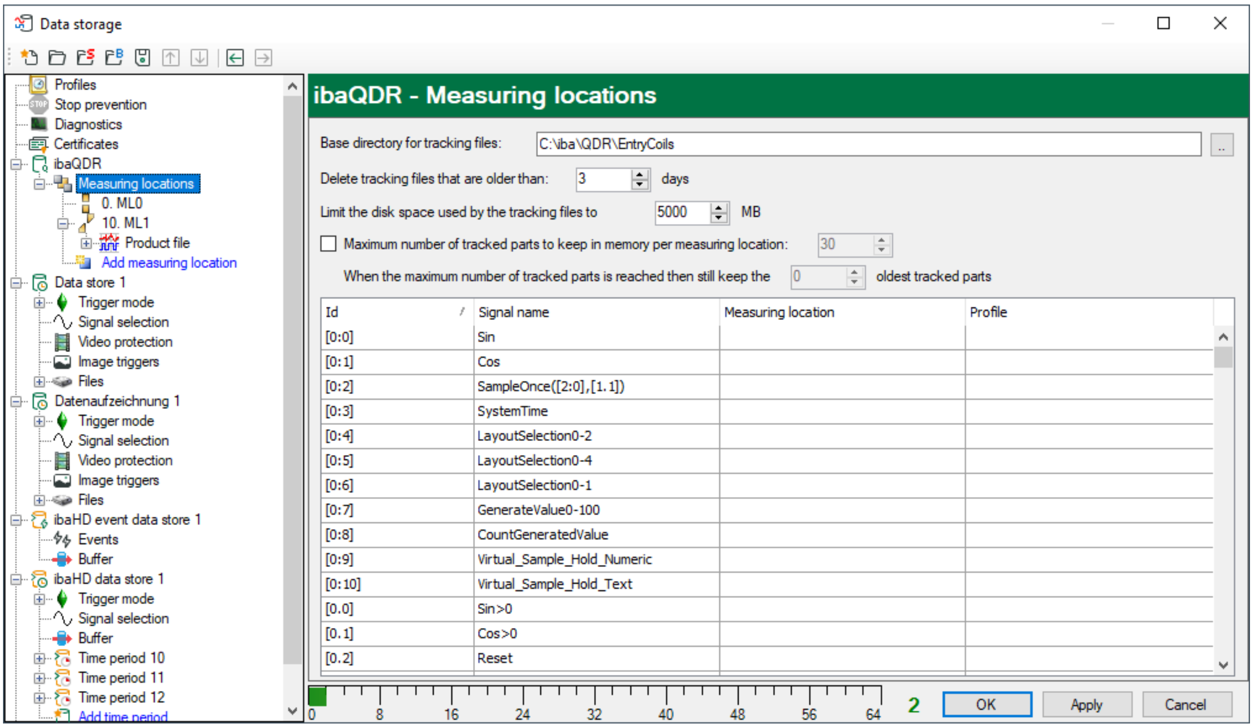
□ Use multiple levels of subdirectories First week of year: First 4-day week

□ Restart file number when subdirectory changes □ Use 4 numbers for the year

3.3 ibaQDR data store

3.3.1 Measuring locations

The default base directory was changed from *C:\dat* to *C:\iba\QDR\EntryCoils*.



Data storage

- Profiles
- Stop prevention
- Diagnostics
- Certificates
- ibaQDR
  - Measuring locations
    - 0. ML0
    - 10. ML1
      - Product file
    - Add measuring location
- Data store 1
  - Trigger mode
  - Signal selection
  - Video protection
  - Image triggers
  - Files
- Datenaufzeichnung 1
  - Trigger mode
  - Signal selection
  - Video protection
  - Image triggers
  - Files
- ibaHD event data store 1
  - Events
  - Buffer
- ibaHD data store 1
  - Trigger mode
  - Signal selection
  - Buffer
  - Time period 10
  - Time period 11
  - Time period 12
  - Add time period

ibaQDR - Measuring locations

Base directory for tracking files: C:\iba\QDR\EntryCoils

Delete tracking files that are older than: 3 days

Limit the disk space used by the tracking files to 5000 MB

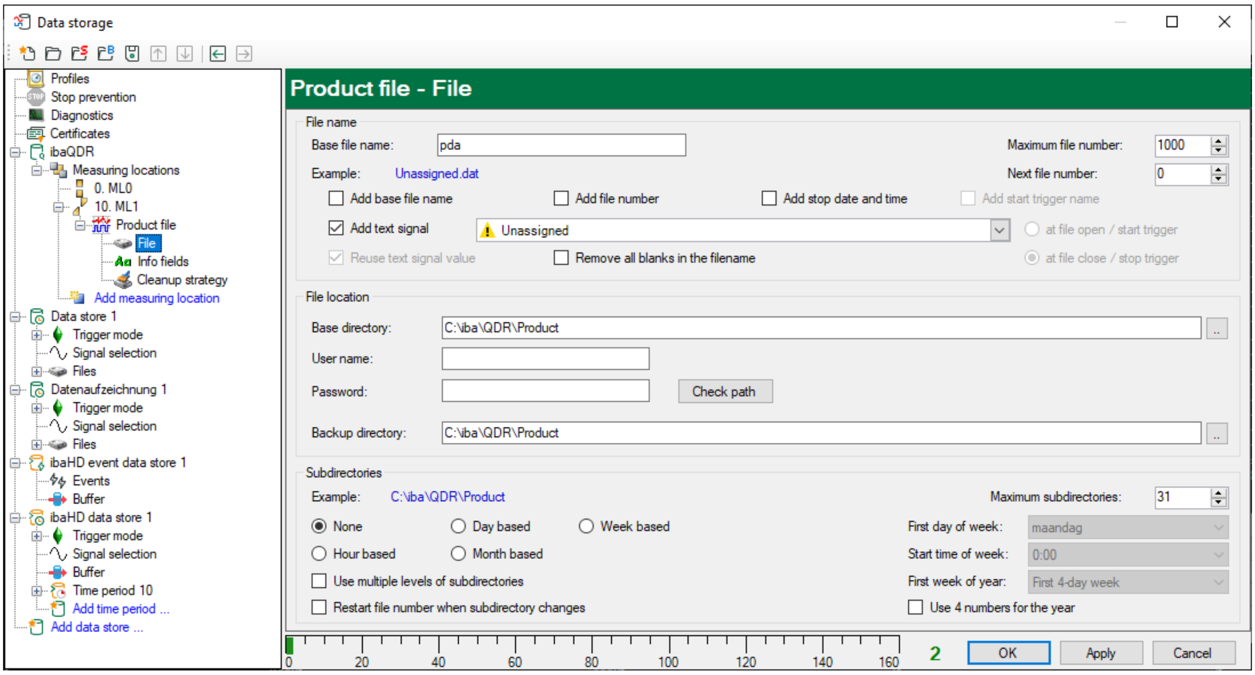
□ Maximum number of tracked parts to keep in memory per measuring location: 30

When the maximum number of tracked parts is reached then still keep the 0 oldest tracked parts

Id	Signal name	Measuring location	Profile
[0:0]	Sin		
[0:1]	Cos		
[0:2]	SampleOnce([2:0],[1.1])		
[0:3]	SystemTime		
[0:4]	LayoutSelection0-2		
[0:5]	LayoutSelection0-4		
[0:6]	LayoutSelection0-1		
[0:7]	GenerateValue0-100		
[0:8]	CountGeneratedValue		
[0:9]	Virtual_Sample_Hold_Numeric		
[0:10]	Virtual_Sample_Hold_Text		
[0.0]	Sin>0		
[0.1]	Cos>0		
[0.2]	Reset		

3.3.2 Product file

The default base directory for product files was changed from *C:\dat* to *C:\iba\QDR\Product*.



Data storage

- Profiles
- Stop prevention
- Diagnostics
- Certificates
- ibaQDR
  - Measuring locations
    - 0. ML0
    - 10. ML1
      - Product file
        - File
        - Info fields
        - Cleanup strategy
    - Add measuring location
- Data store 1
  - Trigger mode
  - Signal selection
  - Files
- Datenaufzeichnung 1
  - Trigger mode
  - Signal selection
  - Files
- ibaHD event data store 1
  - Events
  - Buffer
- ibaHD data store 1
  - Trigger mode
  - Signal selection
  - Buffer
  - Time period 10
  - Add time period ...
- Add data store ...

Product file - File

File name

Base file name: pda

Maximum file number: 1000

Example: Unassigned.dat

Next file number: 0

□ Add base file name □ Add file number □ Add stop date and time □ Add start trigger name

☑ Add text signal: Unassigned ○ at file open / start trigger

☑ Reuse text signal value □ Remove all blanks in the filename ● at file close / stop trigger

File location

Base directory: C:\iba\QDR\Product

User name:

Password:

Check path

Backup directory: C:\iba\QDR\Product

Subdirectories

Example: C:\iba\QDR\Product

Maximum subdirectories: 31

● None ○ Day based ○ Week based

○ Hour based ○ Month based

First day of week: maandag

Start time of week: 0:00

First week of year: First 4-day week

□ Use multiple levels of subdirectories

□ Restart file number when subdirectory changes

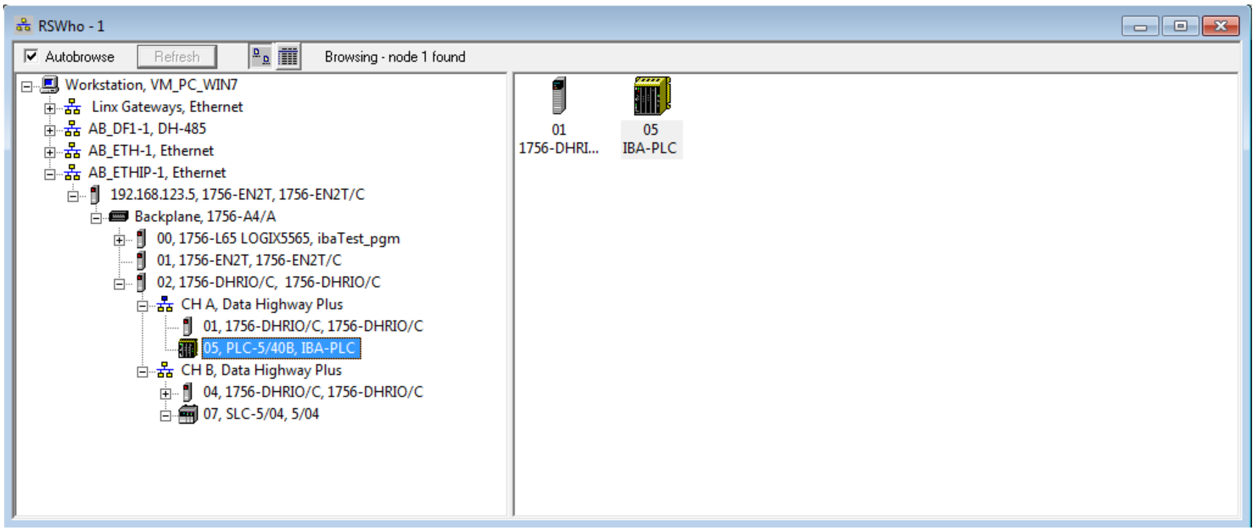
□ Use 4 numbers for the year

4 Routing support in AB-Xplorer

AB-Xplorer is used to measure data from Allen-Bradley/Rockwell PLC-5, SLC-500 and MicroLogix controllers. AB-Xplorer supports the following connection methods:

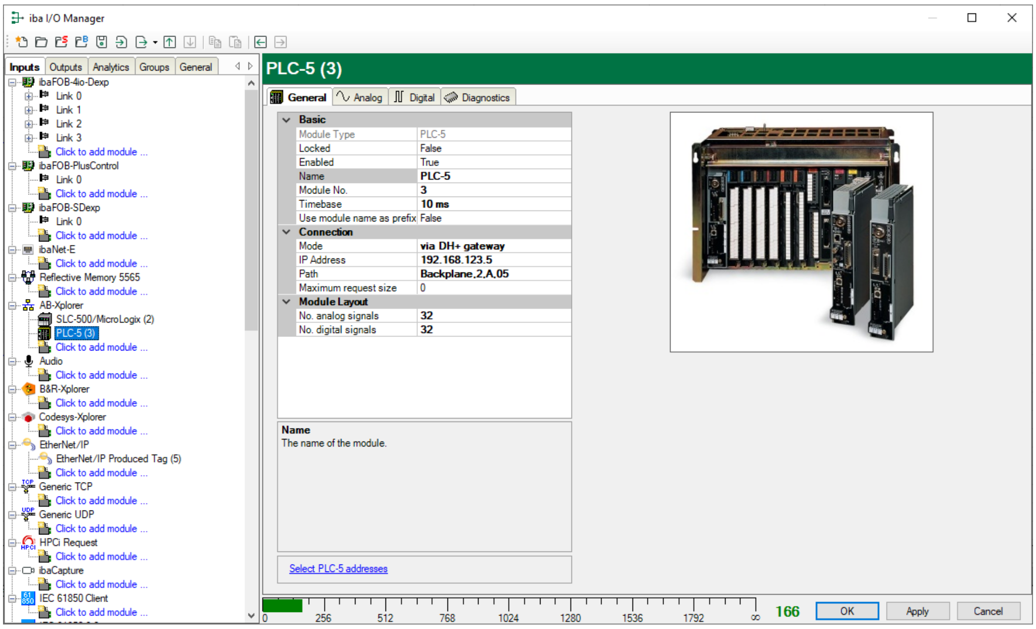
1. Direct Ethernet connection. This can be used for controllers with Ethernet onboard like e.g. the SLC-5/05 and PLC-5 with 1785-ENET card installed.
2. Connection via a 1761-NET-ENI adapter. The adapter should be connected to the DF1 port of SLC-500, PLC-5 and MicroLogix controllers.
3. EtherNet/IP connection. The MicroLogix 1100 and 1400 controllers support EtherNet/IP directly.

In ibaPDA v8.5.0 a fourth connection method is added: via DH+ gateway. An SLC-5/04 or PLC-5 controller can be connected via DH+ (Data Highway Plus) to a ControlLogix controller. IbaPDA will then communicate with the SLC-5/04 or PLC-5 via the ControlLogix controller.



```
Workstation, VM_PC_WIN7
  Linx Gateways, Ethernet
  AB_DF1-1, DH-485
  AB_ETH-1, Ethernet
  AB_ETHIP-1, Ethernet
    192.168.123.5, 1756-EN2T, 1756-EN2T/C
      Backplane, 1756-A4/A
        00, 1756-L65 LOGIX5565, ibaTest_pgm
        01, 1756-EN2T, 1756-EN2T/C
        02, 1756-DHRIO/C, 1756-DHRIO/C
          CH A, Data Highway Plus
            01, 1756-DHRIO/C, 1756-DHRIO/C
            05, PLC-5/40B, IBA-PLC
          CH B, Data Highway Plus
            04, 1756-DHRIO/C, 1756-DHRIO/C
            07, SLC-5/04, 5/04
```

In the screenshot you see that the ControlLogix at IP address 192.168.123.5 has a 1756-DHRIO card in slot 2. This card has 2 DH+ channels. On channel A a PLC-5/40B with DH+ address 5 is connected. On channel B an SLC-5/04 with DH+ address 7 is connected.



PLC-5 (3)

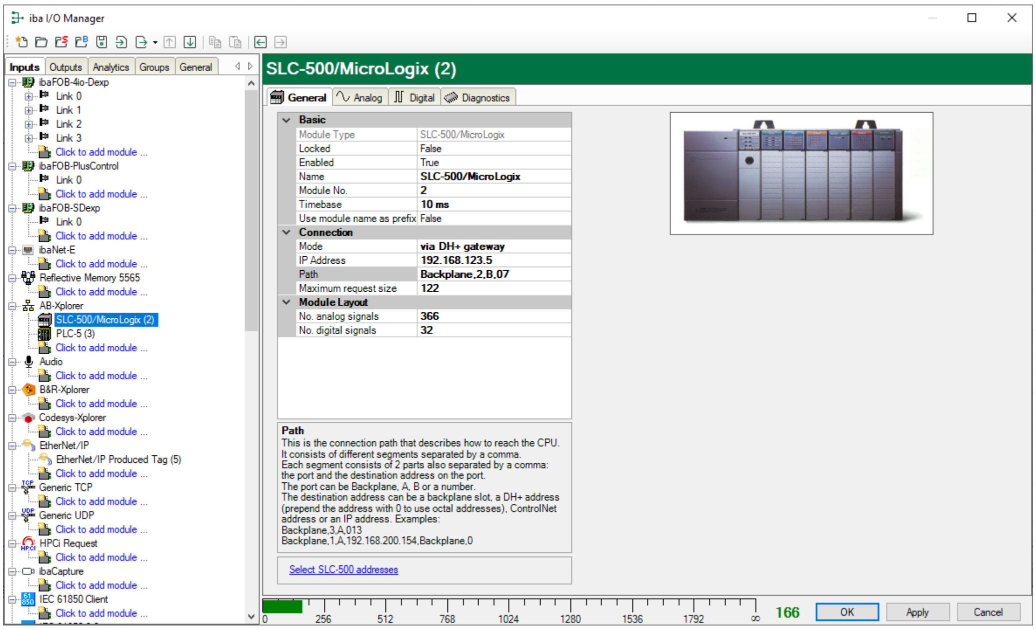
Basic	
Module Type	PLC-5
Locked	False
Enabled	True
Name	**PLC-5**
Module No.	**3**
Timebase	**10 ms**
Use module name as prefix	False
Connection	
Mode	**via DH+ gateway**
IP Address	**192.168.123.5**
Path	**Backplane,2,A,05**
Maximum request size	0
Module Layout	
No. analog signals	**32**
No. digital signals	**32**

Name
The name of the module.

Select PLC-5 addresses

If you want to measure data from the PLC-5 then you can add a PLC-5 module in ibaPDA. As *Mode* you select *"via DH+ gateway"*. The *IP Address* should be the IP address of the ControlLogix controller. The *Path* describes how to reach the PLC-5 from the ControlLogix. It consists of different segments separated by a comma. Each segment consists of 2 parts also separated by a comma: the port and the destination address on the port. The port can be Backplane, A, B or a number. The destination address can be a backplane slot, a DH+ address (prepend the address with 0 to use octal addresses), ControlNet address or an IP address.

In our case the *Path* should be *Backplane,2,A,05*. The first segment *Backplane,2* means that we go to the backplane and take slot 2. This way we reach the 1756-DHRIO in slot 2 of the ControlLogix. The second segment *A,05* means that we take channel A on the 1756-DHRIO and go to the device with DH+ address 5, which is the PLC-5. Addresses on DH+ are usually expressed in octal. Therefor you should prepend the address with a zero, so 05 instead of 5.



SLC-500/MicroLogix (2)

Basic	
Module Type	SLC-500/MicroLogix
Locked	False
Enabled	True
Name	**SLC-500/MicroLogix**
Module No.	**2**
Timebase	**10 ms**
Use module name as prefix	False
Connection	
Mode	**via DH+ gateway**
IP Address	**192.168.123.5**
Path	**Backplane,2,B,07**
Maximum request size	**122**
Module Layout	
No. analog signals	**366**
No. digital signals	**32**

Path
This is the connection path that describes how to reach the CPU.
It consists of different segments separated by a comma.
Each segment consists of 2 parts also separated by a comma:
the port and the destination address on the port.
The port can be Backplane, A, B or a number.
The destination address can be a backplane slot, a DH+ address
(prepend the address with 0 to use octal addresses), ControlNet
address or an IP address. Examples:
Backplane,3,A,013
Backplane,1,A,192.168.200.154,Backplane,0

Select SLC-500 addresses

The SLC-5/04 from the example can be reached via *Path Backplane,2,B,07*. *Backplane,2* brings us again to the 1756-DHRIO. *B,07* goes to octal address 7 on channel B, which is the SLC-5/04.

The *Maximum request size* property can be used to limit the number of words that can be read in one request. The message size on DH+ is limited. The size can be maximum 122 words. So it is recommended to fill this in in the *Maximum request size* property.

Consecutive addresses from a single file e.g. N7:0 to N7:19 can be read in a single command. Non-consecutive addresses or addresses from different files require multiple commands. The more commands required, the slower the communication will be.

5 Value input control

Starting with this version, ibaQPanel provides a new value input control that replaces the text input control.

The value input control is an improved version of the text input control with more features and configuration options, such as

- Support for multiline
- Full-featured standalone button to apply pending changes
- Improved settings for less complicated and more convenient configuration

Remark:
To support backwards compatibility, the text input control can still be used in the ibaQPanel. However, it can no longer be added to the designer.

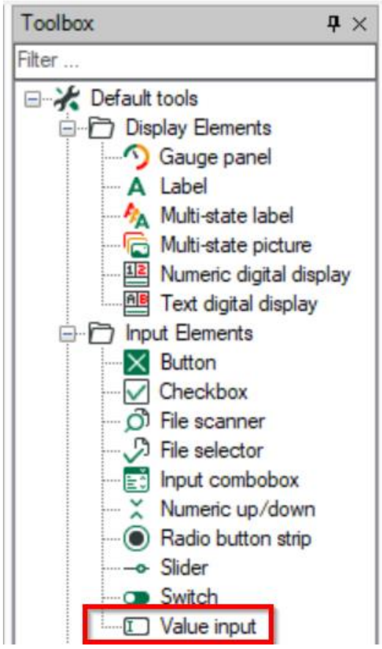
Notable changes

- When adding a value input control from the toolbar, the apply button is not created by default.

  To add the apply button, use the *Add apply button* option in the context menu.
- In the settings, the text mode option (input only, input == output and input <> output) is replaced with individual properties.
- To convert text input controls to value input control, use the *Convert to value input* option in the context menu

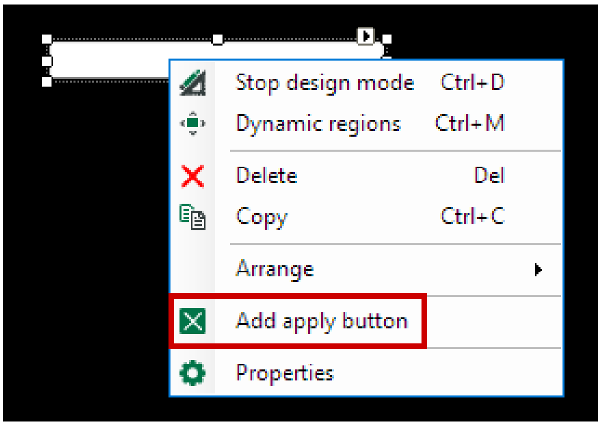
5.1 Designer

A new value input control can be added using the *Value input control* entry in the toolbox.

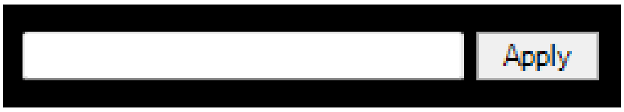


5.1.1 Apply button

The apply button is not created by default. However, it can be added using the *Add apply button* option in the context menu of the value input.



Once clicked, an *Apply* button appears next to the value input control.



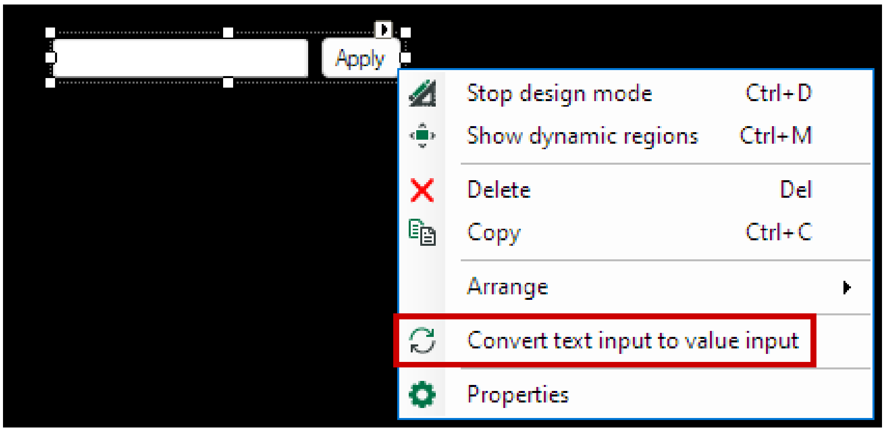
The button is automatically linked to the selected value input control and can be used to apply pending changes.

It’s also possible to select multiple value input controls at once. In this case, the *apply* button is added to each selected control.

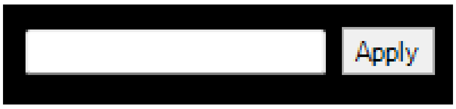
5.1.2 Converting text input controls to value input controls

The designer provides a convenient way to convert obsolete text input controls into improved value input controls.

When right-clicking a text input control, the option *Convert text input to value input* can be selected. This will convert the selected text input info a value input.

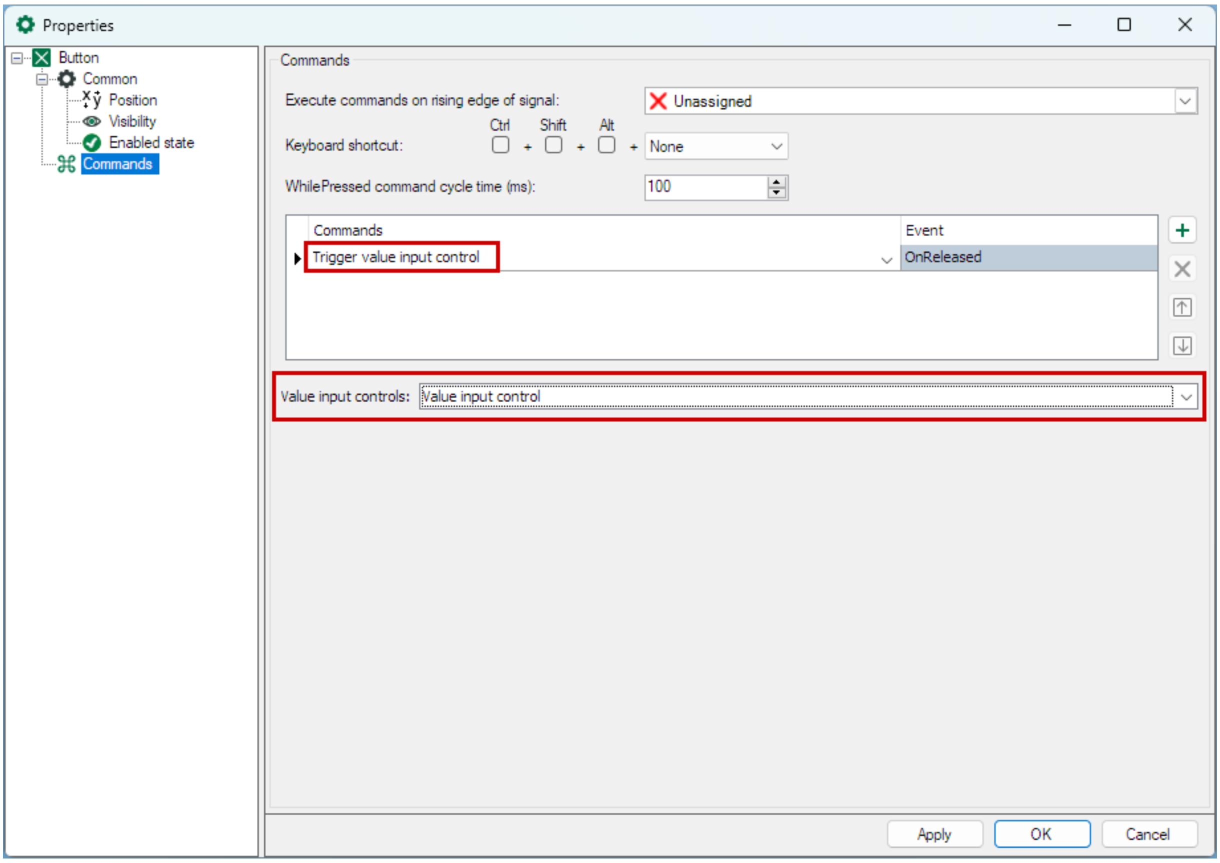


All settings of the text input are taken over into the value input, as well as the position and the size of the text input control.



5.2 Trigger value input button command

To link a button to value input controls, the commands have been extended with a new *Trigger value input control* command.



Commands

Execute commands on rising edge of signal: ☒ Unassigned

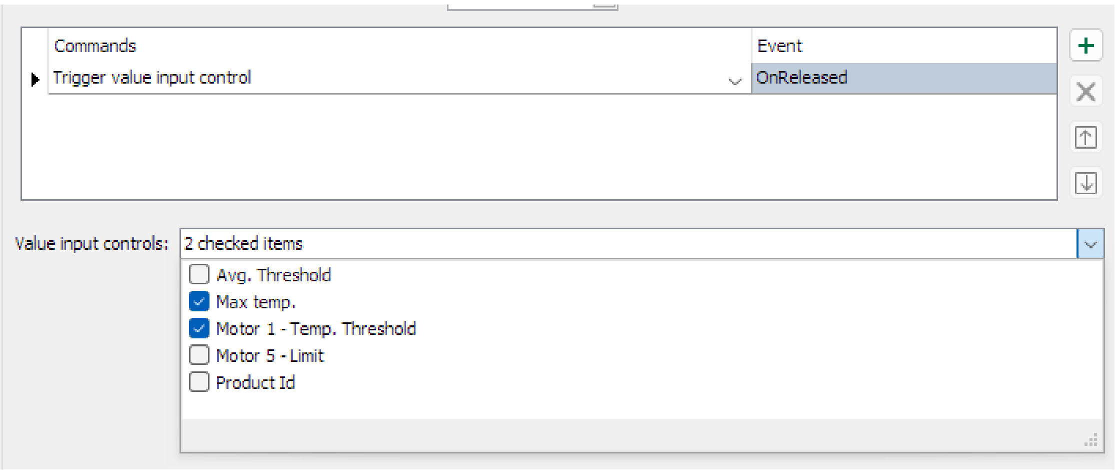
Keyboard shortcut: Ctrl □ + Shift □ + Alt □ + None

WhilePressed command cycle time (ms): 100

Commands	Event
Trigger value input control	OnReleased

Value input controls: Value input control

When selecting the new command, a list with all available value input controls of the current ibaQPanel appears. One or more entries can be selected.

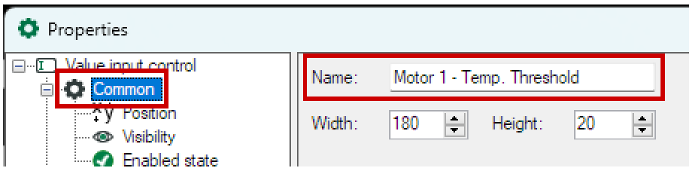


Commands	Event
Trigger value input control	OnReleased

Value input controls: 2 checked items

- □ Avg. Threshold
- ☑ Max temp.
- ☑ Motor 1 - Temp. Threshold
- □ Motor 5 - Limit
- □ Product Id

The name of each entry is defined by the *Name* property of the respective value input control.



Name: Motor 1 - Temp. Threshold

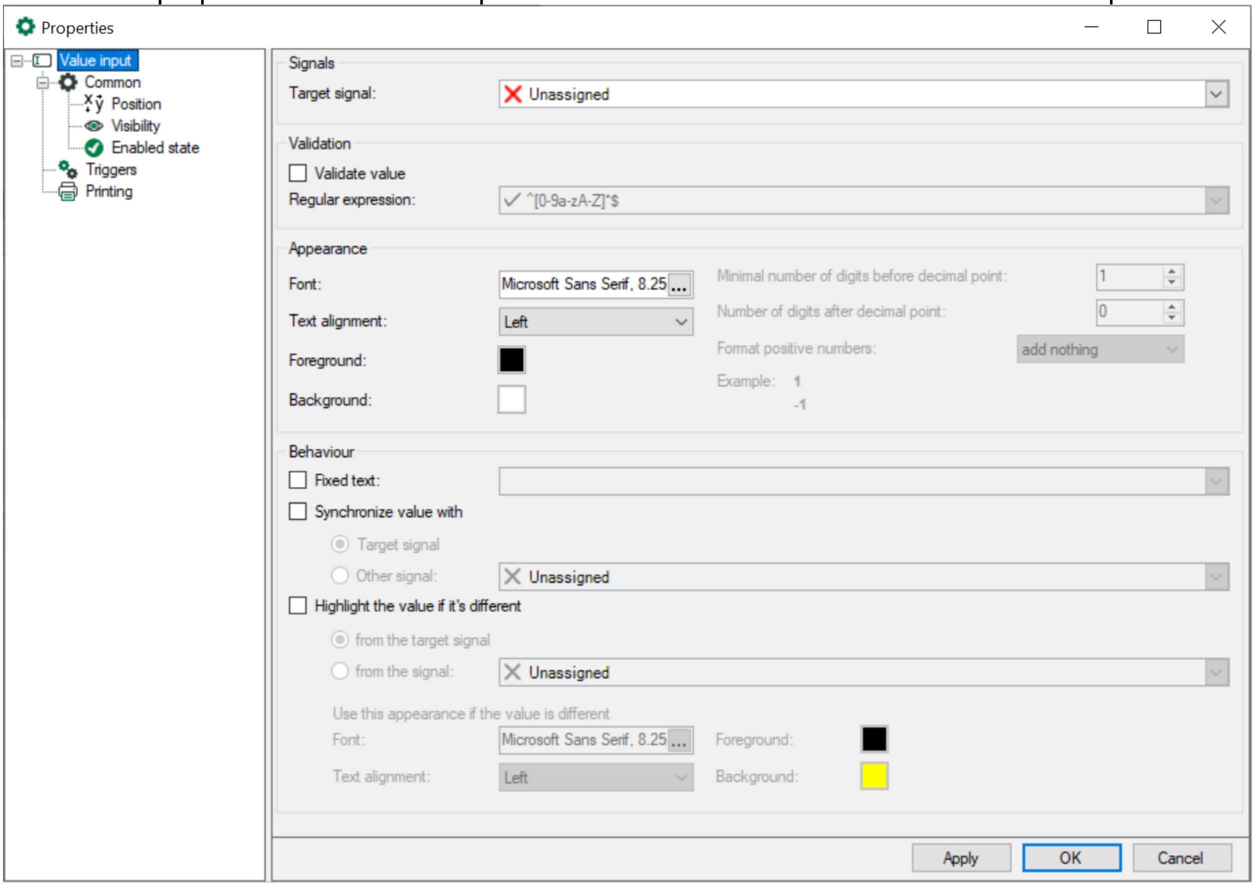
Width: 180 Height: 20

Remark:
It’s recommended to use meaningful names for the controls. Otherwise, it’s difficult to identify the intended value input controls.

When clicking the button, each selected value input control is triggered and the pending changes are saved.

5.3 Properties

Most of the properties of the value input control have been taken over from the text input



Properties

- Value input
  - Common
    - Position
    - Visibility
    - Enabled state
  - Triggers
  - Printing

Signals

Target signal: ☒ Unassigned

Validation

□ Validate value

Regular expression: ✓ ^[0-9a-zA-Z]*$

Appearance

Font: Microsoft Sans Serif, 8.25 ...

Text alignment: Left

Foreground: ■

Background: □

Minimal number of digits before decimal point: 1

Number of digits after decimal point: 0

Format positive numbers: add nothing

Example: 1
-1

Behaviour

□ Fixed text:

□ Synchronize value with

● Target signal

○ Other signal: ✕ Unassigned

□ Highlight the value if it’s different

● from the target signal

○ from the signal: ✕ Unassigned

Use this appearance if the value is different

Font: Microsoft Sans Serif, 8.25 ...

Foreground: ■

Text alignment: Left

Background: 🟨

Remark:
In this section, the term *value* refers to the text currently displayed in the value input control. This can either be a value entered by the user or a value set externally by a signal.

Target signal

The target signal can either be an ibaQPanel text input or an ibaQPanel input.

When applying the value with <ENTER>, the apply button or a trigger, it’s saved in the destination.

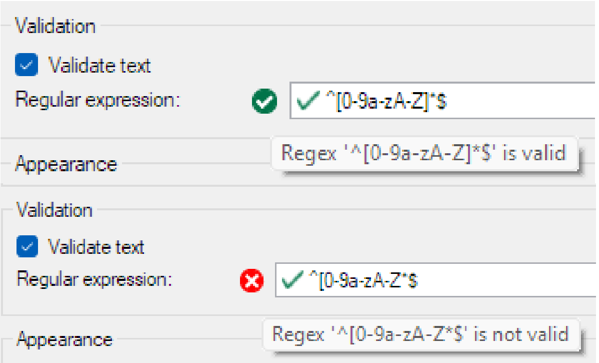
Validate text

The value of a value input control can be validated with a regular expression. This is mainly used to ensure that it only contains defined, valid characters.

Either a static regex can be entered or a signal containing a regex can be selected.

Static regex

The default pattern **^[0-9a-zA-Z]*$** allows any characters from a-z, A-Z and 0-9. Special characters are not allowed. When the pattern is changed, an icon and the tooltip indicate whether the entered pattern is valid.



Validation

☑ Validate text

Regular expression: ✓ ^[0-9a-zA-Z]*$

Regex '^[0-9a-zA-Z]*$' is valid

Appearance

Validation

☑ Validate text

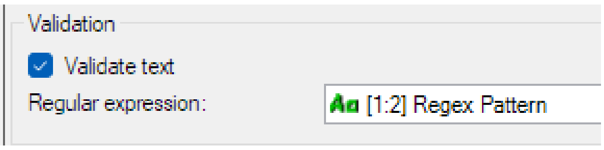
Regular expression: ✗ ^[0-9a-zA-Z*$

Regex '^[0-9a-zA-Z*$' is not valid

Appearance

Signal

Besides using a static text as the pattern, a signal can be used as well.



Validation

☑ Validate text

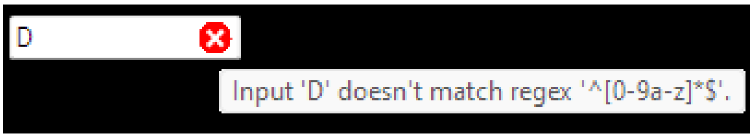
Regular expression: [1:2] Regex Pattern

The signal contains the pattern that is used to validate the text. This is especially useful if the pattern has to be changed dynamically.

Value input behaviour

If the validation is activated and the entered value doesn't match the pattern, the value cannot be applied.

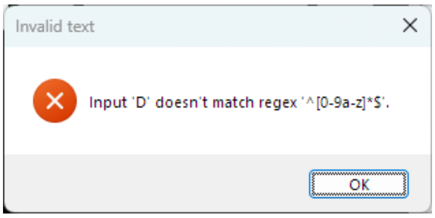
This is indicated by an error icon with a tooltip.



D

Input 'D' doesn't match regex '^[0-9a-z]*$'.

When attempting to apply the value, the error is also shown in a message box.

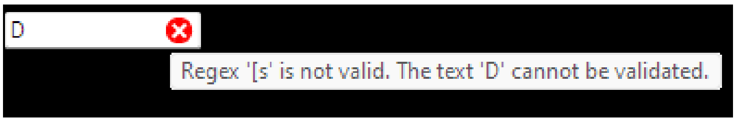


Invalid text

Input 'D' doesn't match regex '^[0-9a-z]*$'.

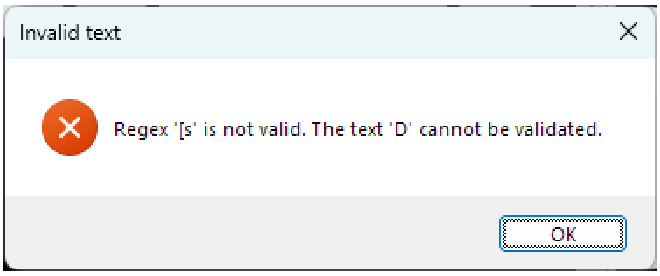
When a signal is selected as the source of the pattern, the signal may contain an invalid pattern.

In this case, a more detailed error is shown.



D

Regex '[s' is not valid. The text 'D' cannot be validated.



Invalid text

Regex '[s' is not valid. The text 'D' cannot be validated.

Appearance

Various display settings can be changed to customize the appearance.

The precision settings are only used if the current value is numeric.

Fixed text

If *fixed text* is activated, the value input control is read only and cannot be modified. The value is set to the defined text.

Synchronize value

This option allows the value input control to synchronize the current value with either the destination or a selected signal. This way, the value input control is always updated with the latest value of the selected source.

If there's a pending change, e.g the value input control is focused or the value has been changed without applying it, the value will not be synchronized to prevent overwriting the user's changes.

Highlight the value

In some cases it's useful to change the appearance if the value is different from either the destination or a specific signal.

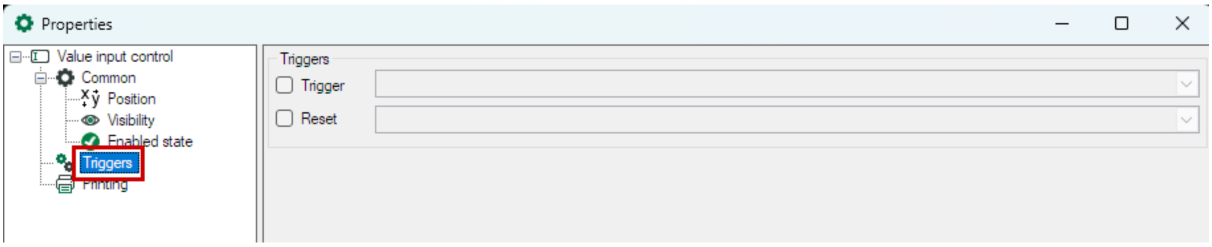
Remark:
In both cases, the current value of the value input control is used for the comparison. This can be the latest value of the selected synchronization source or a uncomitted value entered by the user.

The appearance can be set using the available font settings, text alignment or foreground and background color.

As soon as the value matches the selected comparison source, the changed appearance is reset.

5.3.1 Triggers

As for the text input control, a *trigger* and a *reset* signal can be defined for a value input control in the *Triggers* node.



Trigger

The current pending changes are saved if a rising edge of the selected *trigger* signal is detected.

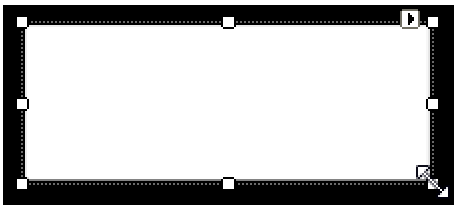
Reset

If a rising edge of the *reset* signal is detected, pending changes are reset to the value of the destination signal.

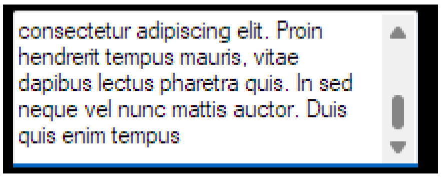
If the *synchronize value* option is activated, the value is reset to the selected synchronization source.

5.4 Multiline

The new multiline feature of the value input control can be activated by simply changing the size of the control.



A vertical scrollbar is added automatically if the text exceeds the height.



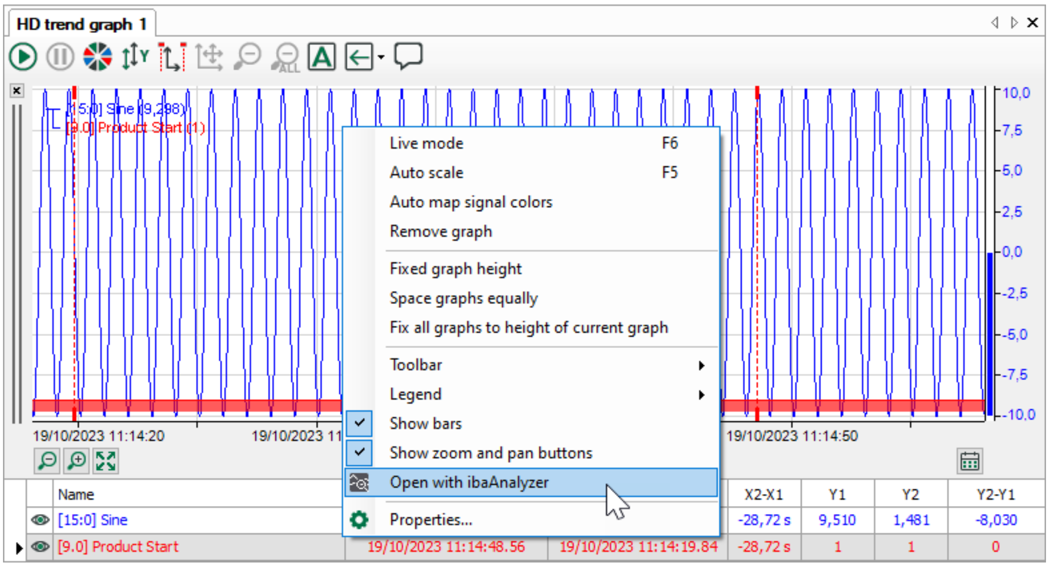
consectetur adipiscing elit. Proin hendrerit tempus mauris, vitae dapibus lectus pharetra quis. In sed neque vel nunc mattis auctor. Duis quis enim tempus

Remark:
The <RETURN> key applies the current value.
To enter a new line, use <CTRL> + <RETURN> or <SHIFT> + <RETURN>.

6 ibaHD related improvements

6.1 ibaHD trend graph

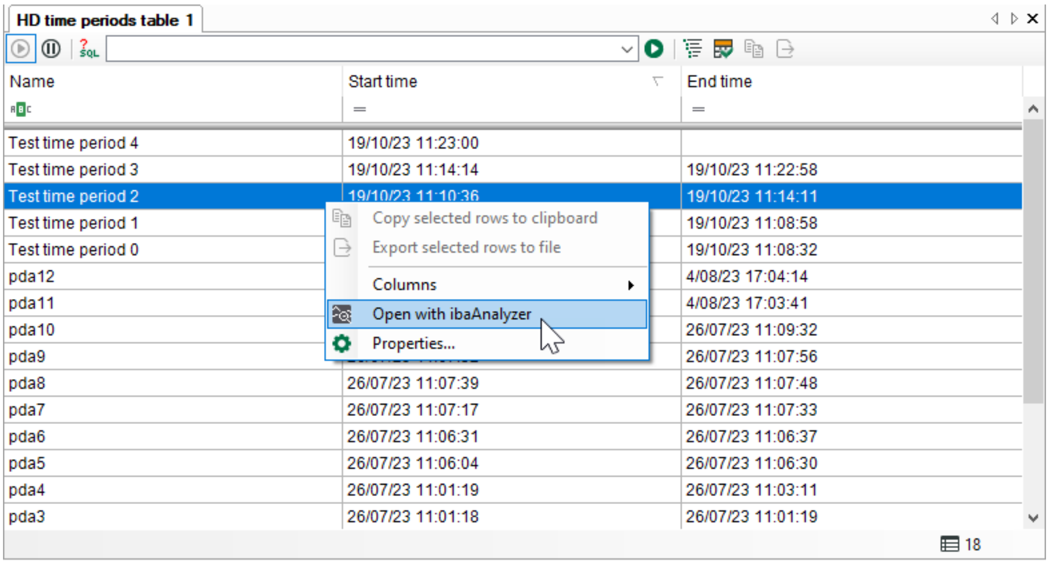
When the ibaHD trend graph is in pause then it is possible to open the displayed time range in ibaAnalyzer. When you right-click on a graph the context menu has a new command “Open with ibaAnalyzer” for this.



Name			X2-X1	Y1	Y2	Y2-Y1
[15:0] Sine			-28,72 s	9,510	1,481	-8,030
[9.0] Product Start	19/10/2023 11:14:48.56	19/10/2023 11:14:19.84	-28,72 s	1	1	0

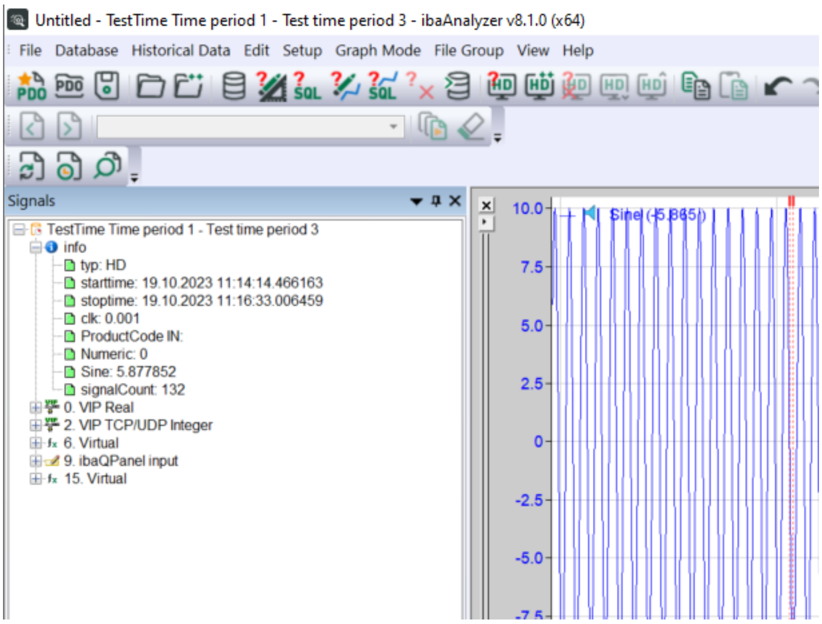
6.2 ibaHD time periods table

When you select a row in the time periods table you can now open it in ibaAnalyzer. Right-click the row to open the context menu and use the new command “Open with ibaAnalyzer”.



Name	Start time	End time
Test time period 4	19/10/23 11:23:00	
Test time period 3	19/10/23 11:14:14	19/10/23 11:22:58
Test time period 2	19/10/23 11:10:36	19/10/23 11:14:11
Test time period 1		19/10/23 11:08:58
Test time period 0		19/10/23 11:08:32
pda12		4/08/23 17:04:14
pda11		4/08/23 17:03:41
pda10		26/07/23 11:09:32
pda9		26/07/23 11:07:56
pda8	26/07/23 11:07:39	26/07/23 11:07:48
pda7	26/07/23 11:07:17	26/07/23 11:07:33
pda6	26/07/23 11:06:31	26/07/23 11:06:37
pda5	26/07/23 11:06:04	26/07/23 11:06:30
pda4	26/07/23 11:01:19	26/07/23 11:03:11
pda3	26/07/23 11:01:18	26/07/23 11:01:19

If ibaAnalyzer v8.1.0 or later is installed then the time period will be opened and the time period info fields will be visible. Also the time period name will be used as file name.



Untitled - TestTime Time period 1 - Test time period 3 - ibaAnalyzer v8.1.0 (x64)

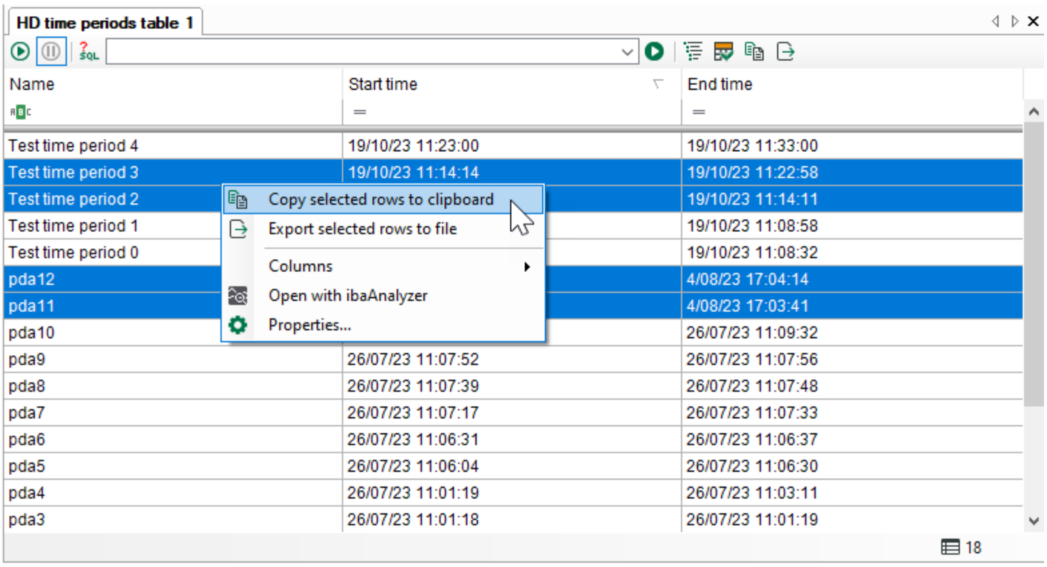
File Database Historical Data Edit Setup Graph Mode File Group View Help
Signals
Sine (-5.865)
10.0
7.5
5.0
2.5
0
-2.5
-5.0
-7.5

TestTime Time period 1 - Test time period 3
- info
  - typ: HD
  - starttime: 19.10.2023 11:14:14.466163
  - stoptime: 19.10.2023 11:16:33.006459
  - clk: 0.001
  - ProductCode IN:
  - Numeric: 0
  - Sine: 5.877852
  - signalCount: 132
- 0. VIP Real
- 2. VIP TCP/UDP Integer
- 6. Virtual
- 9. ibaQPanel input
- 15. Virtual

If an older ibaAnalyzer is installed then just the time range will be opened and no time period info fields will be queried because those ibaAnalyzers don't have time period support. The name of the file will just contain the start and end time, no time period name.

When a time period without an end time is opened in ibaAnalyzer then the current time will be used as end time.

In pause mode you can select multiple rows. When multiple rows are selected and you use the "Open with ibaAnalyzer" command then each row is opened as a separate file in a single ibaAnalyzer.



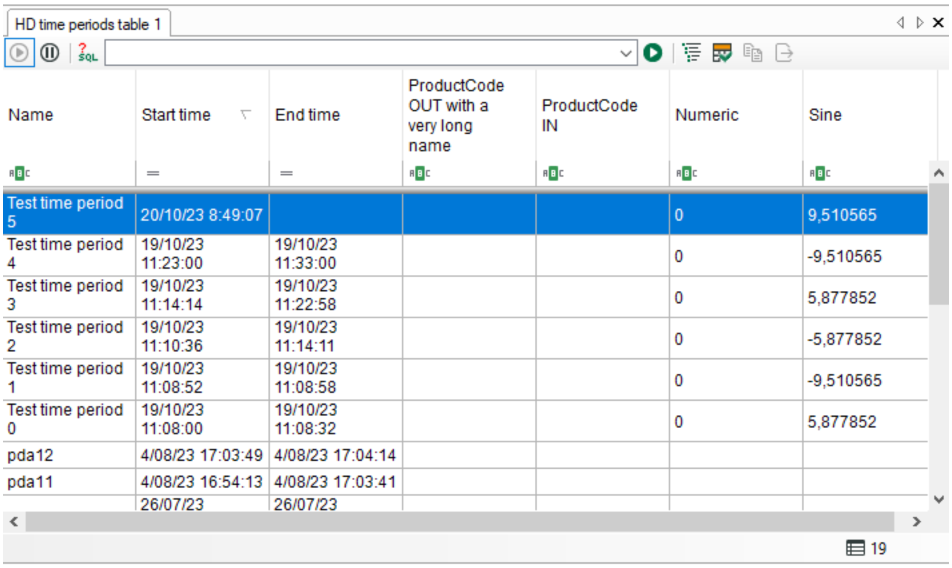
HD time periods table 1

Name	Start time	End time
Test time period 4	19/10/23 11:23:00	19/10/23 11:33:00
Test time period 3	19/10/23 11:14:14	19/10/23 11:22:58
Test time period 2		19/10/23 11:14:11
Test time period 1		19/10/23 11:08:58
Test time period 0		19/10/23 11:08:32
pda12		4/08/23 17:04:14
pda11		4/08/23 17:03:41
pda10		26/07/23 11:09:32
pda9	26/07/23 11:07:52	26/07/23 11:07:56
pda8	26/07/23 11:07:39	26/07/23 11:07:48
pda7	26/07/23 11:07:17	26/07/23 11:07:33
pda6	26/07/23 11:06:31	26/07/23 11:06:37
pda5	26/07/23 11:06:04	26/07/23 11:06:30
pda4	26/07/23 11:01:19	26/07/23 11:03:11
pda3	26/07/23 11:01:18	26/07/23 11:01:19

In pause mode there are 2 extra context menu commands:

- Copy selected rows to clipboard: This will copy the displayed text of the selected rows to the clipboard. Columns are separated by tabs and rows are separated by new lines.
- Export selected rows to file: This allows you to export the values of the selected rows to an Excel file (.xlsx or .xls) or a text file.

These commands can also be triggered by the last two buttons in the toolbar.

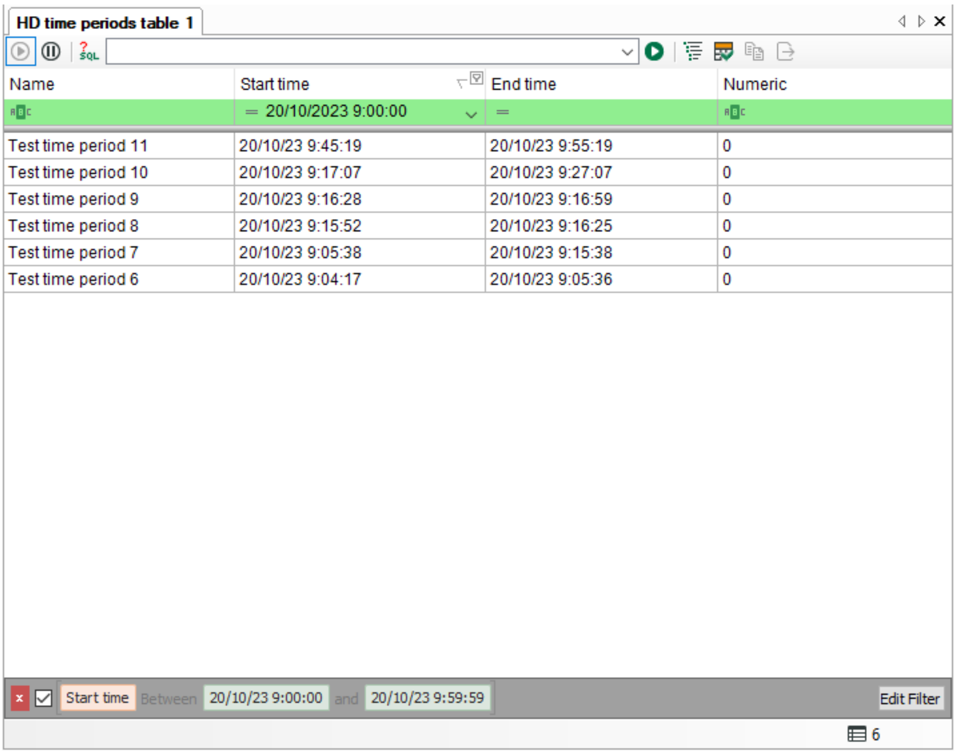


HD time periods table 1

Name	Start time	End time	ProductCode OUT with a very long name	ProductCode IN	Numeric	Sine
Test time period 5	20/10/23 8:49:07				0	9,510565
Test time period 4	19/10/23 11:23:00	19/10/23 11:33:00			0	-9,510565
Test time period 3	19/10/23 11:14:14	19/10/23 11:22:58			0	5,877852
Test time period 2	19/10/23 11:10:36	19/10/23 11:14:11			0	-5,877852
Test time period 1	19/10/23 11:08:52	19/10/23 11:08:58			0	-9,510565
Test time period 0	19/10/23 11:08:00	19/10/23 11:08:32			0	5,877852
pda12	4/08/23 17:03:49	4/08/23 17:04:14				
pda11	4/08/23 16:54:13	4/08/23 17:03:41				
	26/07/23	26/07/23				

19

The time period column header height will be adjusted automatically so that the column name is visible. Word wrap is applied. If a single word is too long for the width of the column then it will be truncated and … is added. This also applies to the normal rows. The height will be changed automatically. When a cell value contains multiple lines then these will now all be shown.



HD time periods table 1

Name	Start time	End time	Numeric
	= 20/10/2023 9:00:00	=	
Test time period 11	20/10/23 9:45:19	20/10/23 9:55:19	0
Test time period 10	20/10/23 9:17:07	20/10/23 9:27:07	0
Test time period 9	20/10/23 9:16:28	20/10/23 9:16:59	0
Test time period 8	20/10/23 9:15:52	20/10/23 9:16:25	0
Test time period 7	20/10/23 9:05:38	20/10/23 9:15:38	0
Test time period 6	20/10/23 9:04:17	20/10/23 9:05:36	0

☒ ☑ Start time Between 20/10/23 9:00:00 and 20/10/23 9:59:59 Edit Filter

6

The start time and end time columns have now better filtering capabilities. You can use the equals, does not equal, less than and greater than operators. The equals and does not equal operators actually work with a time interval. If you enter just the date and set the time to 00:00:00 then all time periods within that day will be shown. If you enter 10:00:00 as time then all time periods between 10:00:00 and 10:59:59 on that day will be shown. So all 00 parts in the time are wildcards. This way you can filter a day, an hour, a minute and a second.

6.3 ibaHD event table

Like the time periods table, the ibaHD event table can copy selected rows to clipboard and can export selected rows to file. It can also automatically adjust the height of all rows. The filtering of the time column is also improved.

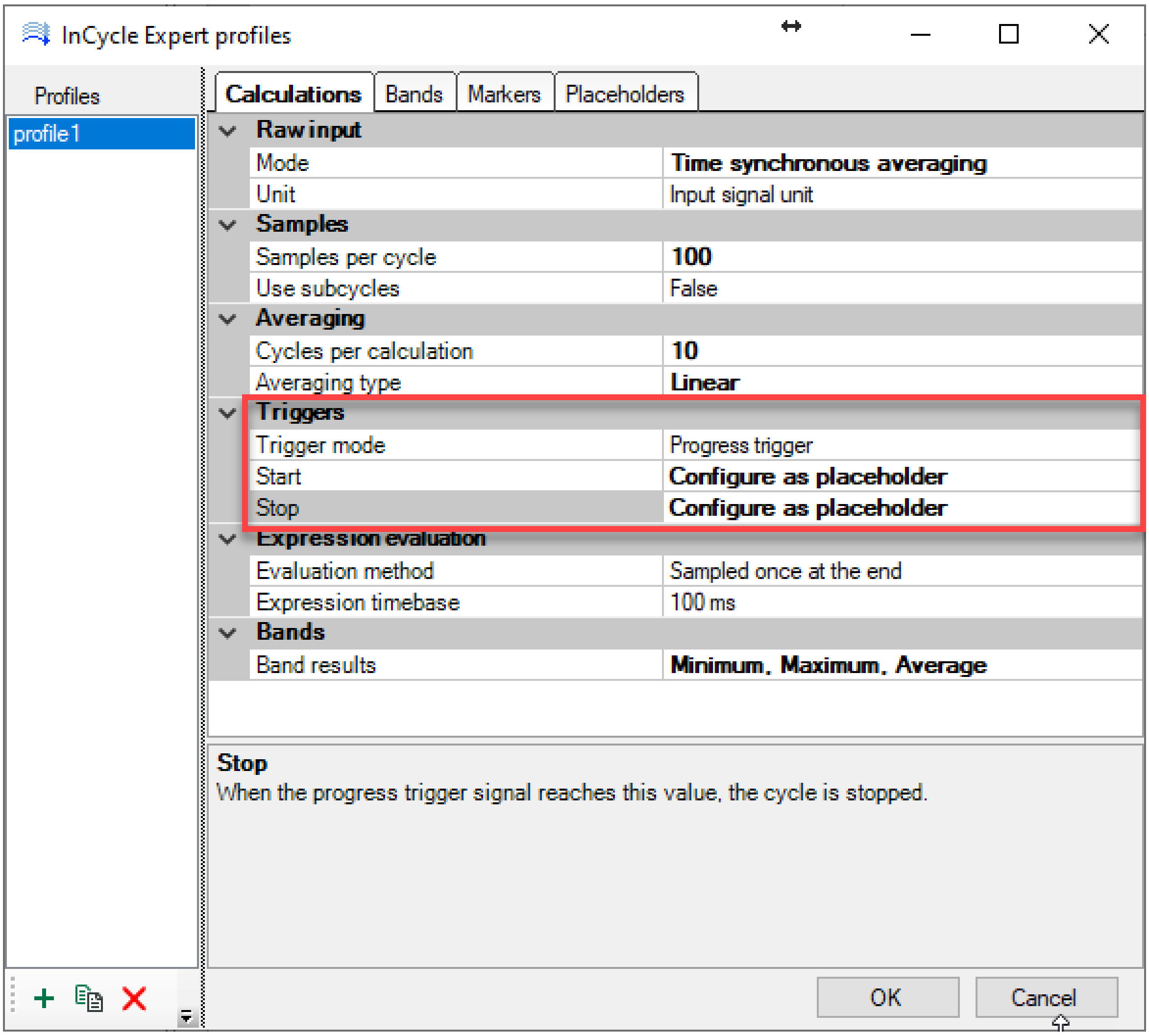
7 Progress trigger for InCycle modules

The InCycle Expert, InCycle Auto-Adapting and InCycle ONNX modules each have a profile where the Mode and Trigger mode must be configured. When the Mode is set to Time synchronous averaging, there are three possible options for the Trigger Mode:

- Pulse trigger
- Start and stop trigger
- Progress trigger

This last option, progress trigger, is new. If you select this option, the triggering and the sampling are determined by a numeric progress signal. During a cycle, this progress signal must increase from a start value to a stop value. When the start value is crossed, a new cycle starts. When the stop value is reached, the cycle ends.

The start and stop value can be configured in the profile itself or a placeholder can be used, which must be assigned in the general tab of the module.

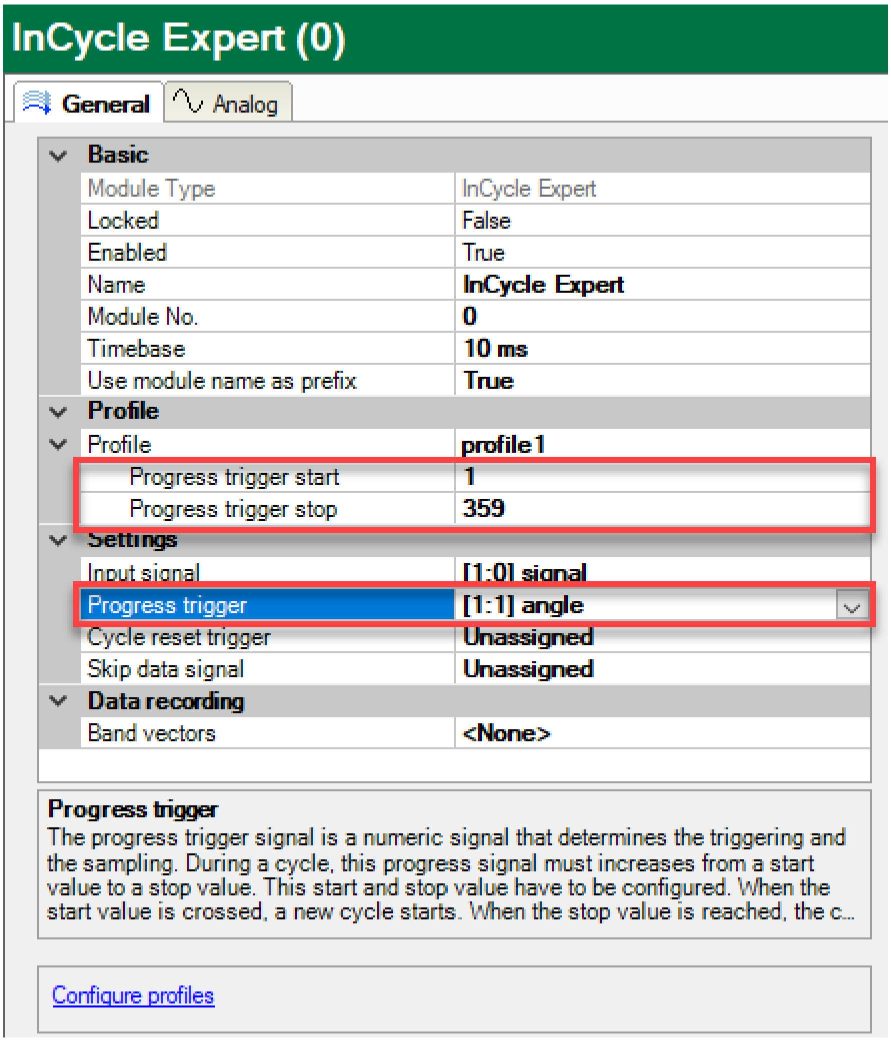


Raw input	
Mode	**Time synchronous averaging**
Unit	Input signal unit
Samples	
Samples per cycle	**100**
Use subcycles	False
Averaging	
Cycles per calculation	**10**
Averaging type	**Linear**
Triggers	
Trigger mode	Progress trigger
Start	**Configure as placeholder**
Stop	**Configure as placeholder**
Expression evaluation	
Evaluation method	Sampled once at the end
Expression timebase	100 ms
Bands	
Band results	**Minimum, Maximum, Average**

Stop
When the progress trigger signal reaches this value, the cycle is stopped.

The rate of increase of the progress signal determines the sampling speed. So, this speed is dynamic during a cycle. If the progress signal rises fast during a part of a cycle, more samples are taken in that part of the cycle.

The progress trigger signal must be assigned in the general tab. It can be a length signal or a signal representing the actual angle of a rotating element.



InCycle Expert (0)

Basic	
Module Type	InCycle Expert
Locked	False
Enabled	True
Name	**InCycle Expert**
Module No.	**0**
Timebase	**10 ms**
Use module name as prefix	**True**
Profile	
Profile	**profile1**
Progress trigger start	**1**
Progress trigger stop	**359**
Settings	
Input signal	**[1:0] signal**
Progress trigger	**[1:1] angle**
Cycle reset trigger	**Unassigned**
Skip data signal	**Unassigned**
Data recording	
Band vectors	**<None>**

Progress trigger
The progress trigger signal is a numeric signal that determines the triggering and the sampling. During a cycle, this progress signal must increases from a start value to a stop value. This start and stop value have to be configured. When the start value is crossed, a new cycle starts. When the stop value is reached, the c...

Configure profiles