New features in ibaPDA v6.29.0

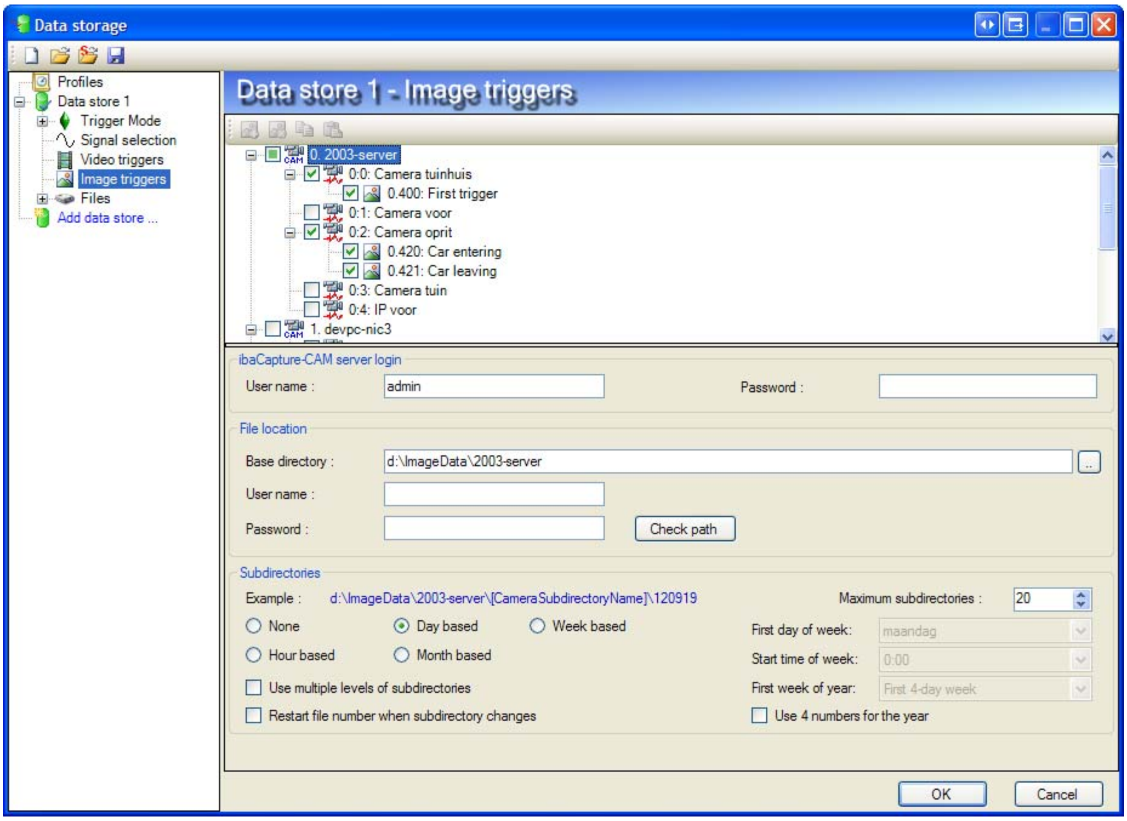
1 ibaCapture-CAM : Image trigger feature

Image triggers allow you to save snapshots from a video stream at specific moments. The image triggers are defined in the data store configuration. The “Image triggers” node will be available when there are ibaCapture-CAM modules in the I/O configuration.

The tree shows all video servers and the active cameras connected to them. You can add a video trigger to a camera by selecting the camera in the tree and clicking the "Add" button in the tool bar or using the context menu (right click on camera).

By setting / removing a checkmark you can enable / disable an image trigger. A camera can have up to 10 image triggers.

The image trigger configuration is set up at three levels: video server, camera and trigger level itself.



- □ 0. 2003-server
  - ☑ 0:0: Camera tuinhuis
    - ☑ 0.400: First trigger
  - □ 0:1: Camera voor
  - ☑ 0:2: Camera oprit
    - ☑ 0.420: Car entering
    - ☑ 0.421: Car leaving
  - □ 0:3: Camera tuin
  - □ 0:4: IP voor
- □ 1. devpc-nic3

ibaCapture-CAM server login

User name : admin

Password :

File location

Base directory : d:\ImageData\2003-server

User name :

Password :

Check path

Subdirectories

Example : d:\ImageData\2003-server\[CameraSubdirectoryName]\120919

Maximum subdirectories : 20

○ None ● Day based ○ Week based

○ Hour based ○ Month based

□ Use multiple levels of subdirectories

□ Restart file number when subdirectory changes

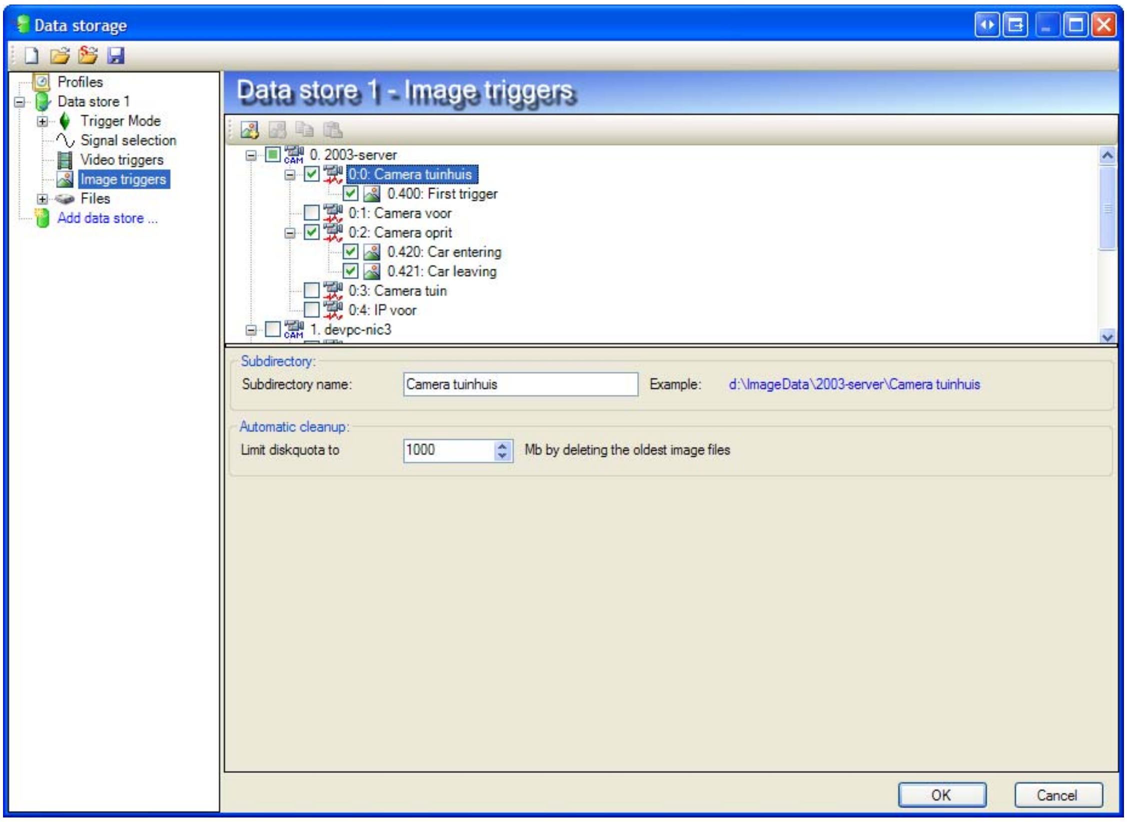
First day of week: maandag

Start time of week: 0:00

First week of year: First 4-day week

□ Use 4 numbers for the year

On the video server level you can specify the base directory and the subdirectory structure which will be created under the camera’s base directory. This is very similar to the data store DAT file directory setup. You can also specify the user account used to logon to the video server.



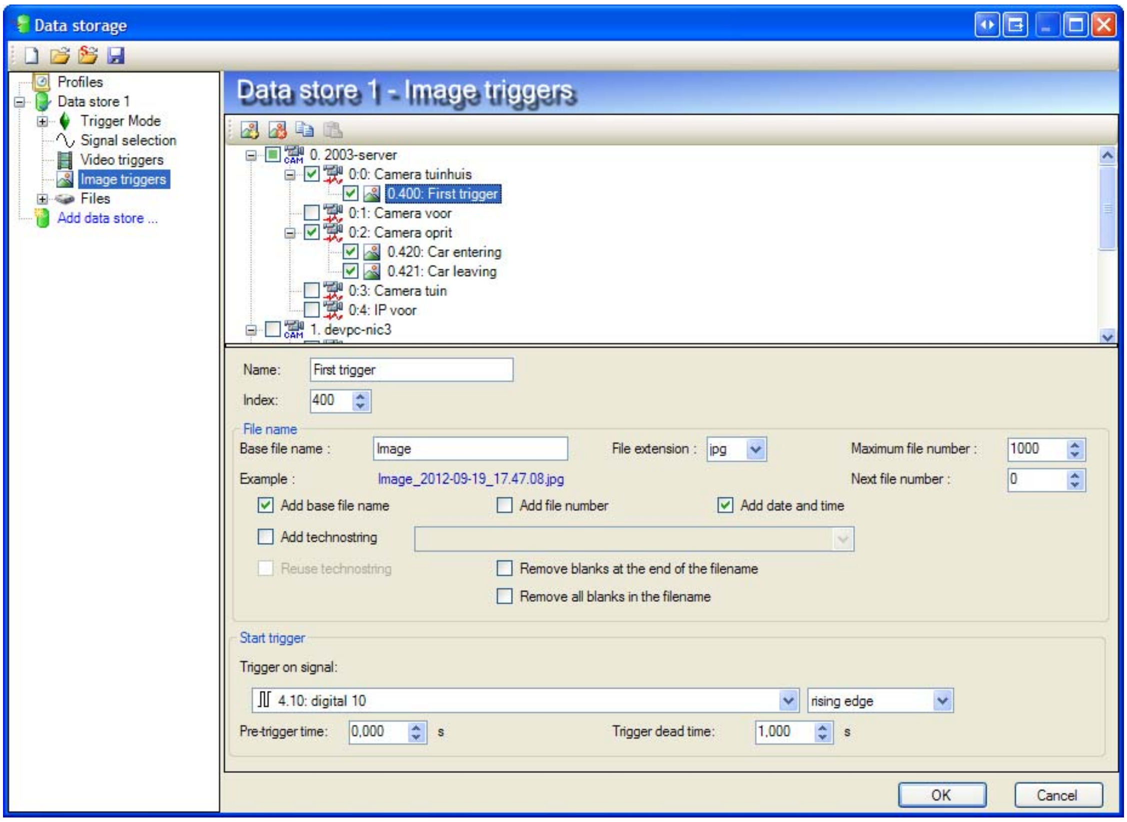
Subdirectory:

Subdirectory name: Camera tuinhuis Example: d:\ImageData\2003-server\Camera tuinhuis

Automatic cleanup:

Limit diskquota to 1000 Mb by deleting the oldest image files

On the camera level you can specify the subdirectory name for the camera and the disk quota. The disk quota is like the disk quota of the data store only checked every 15 minutes. So you should set the limit to the maximum value you want minus the approximate amount of data that is generated within 15 minutes.



Data store 1 - Image triggers

- 0. 2003-server
  - ☑ 0:0: Camera tuinhuis
    - ☑ 0.400: First trigger
  - ☐ 0:1: Camera voor
  - ☑ 0:2: Camera oprit
    - ☑ 0.420: Car entering
    - ☑ 0.421: Car leaving
  - ☐ 0:3: Camera tuin
  - ☐ 0:4: IP voor
- ☐ 1. devpc-nic3

Name: First trigger

Index: 400

File name

Base file name : Image | File extension : jpg | Maximum file number : 1000

Example : Image_2012-09-19_17.47.08.jpg | Next file number : 0

☑ Add base file name ☐ Add file number ☑ Add date and time

☐ Add technostring

☐ Reuse technostring ☐ Remove blanks at the end of the filename

☐ Remove all blanks in the filename

Start trigger

Trigger on signal: 4.10: digital 10 | rising edge

Pre-trigger time: 0,000 s | Trigger dead time: 1,000 s

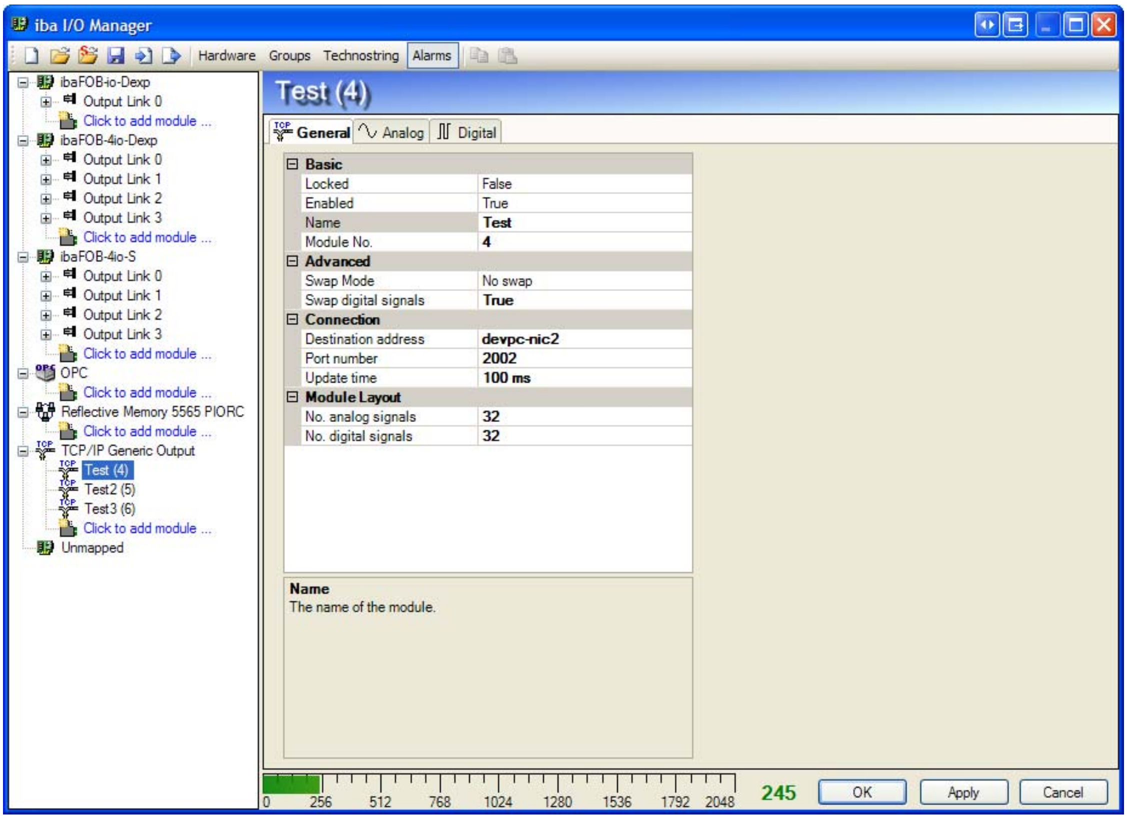
On the trigger level you can change the name of trigger and set its index. The trigger index defines the order of the triggers for one camera. You can specify how the image file name will be generated. The filename options are the same as the options you have for the standard dat file.

The start trigger determines when the snapshot will be taken. You can trigger on the edge of a digital signal or on a level crossing of an analog signal. It is possible to set a pre-trigger time. The snapshot will be taken at the trigger timestamp minus the pre-trigger time. The trigger dead time determines the minimum amount of time between two images.

The image triggers are also saved in the dat file as image channels. An image channel contains the file names of the generated images. A future ibaAnalyzer will then be able to show the generated images.

2 Generic TCP/IP output interface

This new interface allows pda to send out values via TCP/IP. It uses the same license as the generic TCP/IP interface. Each module you add to the TCP/IP generic output interface corresponds with one TCP/IP connection.

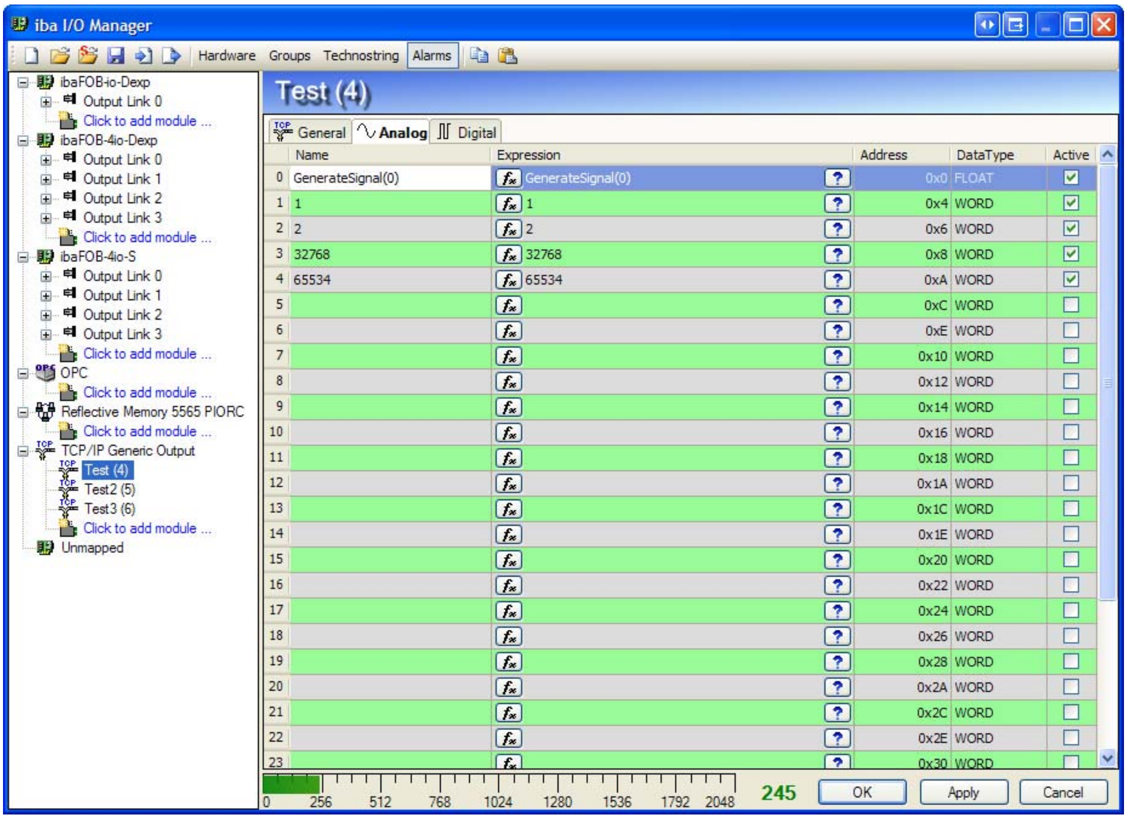


Basic	
Locked	False
Enabled	True
Name	Test
Module No.	4
Advanced	
Swap Mode	No swap
Swap digital signals	True
Connection	
Destination address	devpc-nic2
Port number	2002
Update time	100 ms
Module Layout	
No. analog signals	32
No. digital signals	32

Name
The name of the module.

You have to set the destination address and port number you want to connect to. The update time determines how fast pda will send messages. If the update time is set to 0 then pda will send the message as soon as it receives the data. Like for all output modules pda can only generate data every 50ms or slower if there are modules with a slower timebase.

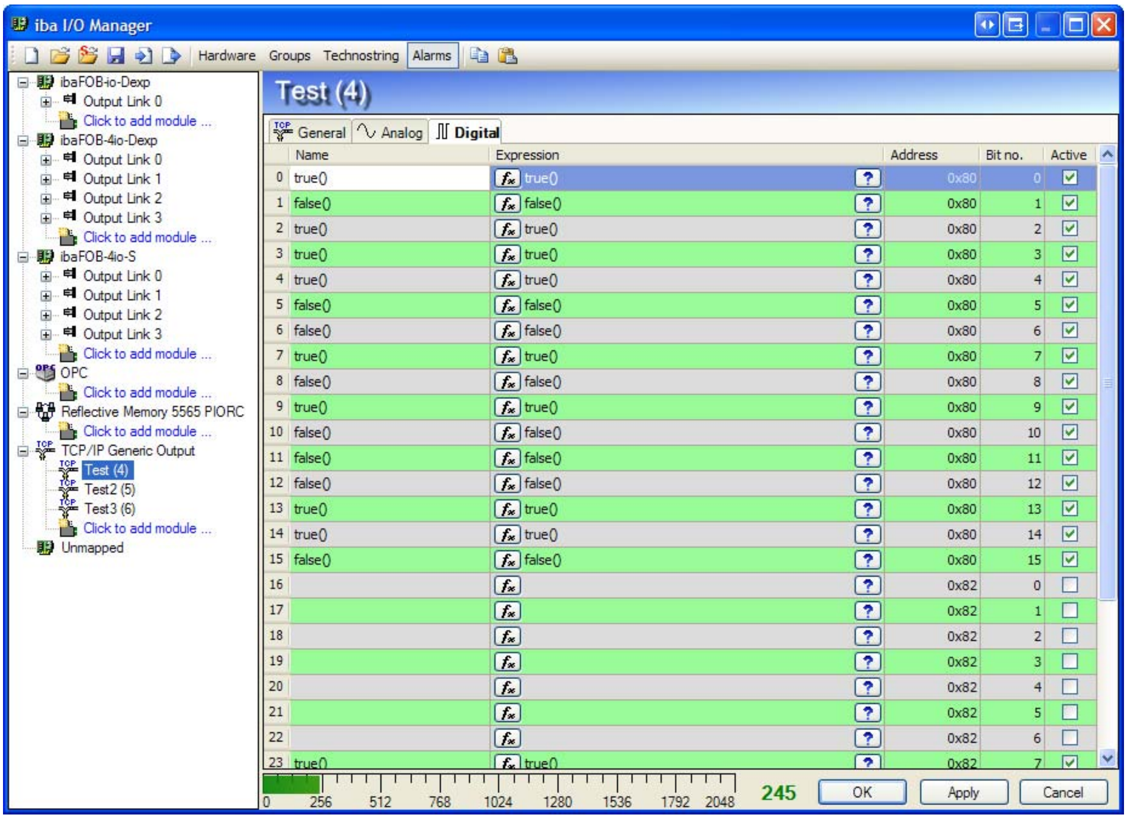
You can change the number of analog and digital signals. The swap mode can also be changed in case the other side of the connection expects a different byte order than little-endian. The digital signals are packed in 16 bit integers. These integers are also swapped when the “Swap digital signals” property is set.



	Name	Expression	Address	DataType	Active
0	GenerateSignal(0)	GenerateSignal(0)	0x0	FLOAT	☑
1	1	1	0x4	WORD	☑
2	2	2	0x6	WORD	☑
3	32768	32768	0x8	WORD	☑
4	65534	65534	0xA	WORD	☑
5			0xC	WORD	☐
6			0xE	WORD	☐
7			0x10	WORD	☐
8			0x12	WORD	☐
9			0x14	WORD	☐
10			0x16	WORD	☐
11			0x18	WORD	☐
12			0x1A	WORD	☐
13			0x1C	WORD	☐
14			0x1E	WORD	☐
15			0x20	WORD	☐
16			0x22	WORD	☐
17			0x24	WORD	☐
18			0x26	WORD	☐
19			0x28	WORD	☐
20			0x2A	WORD	☐
21			0x2C	WORD	☐
22			0x2E	WORD	☐
23			0x30	WORD	☐

Each analog signal has these properties:

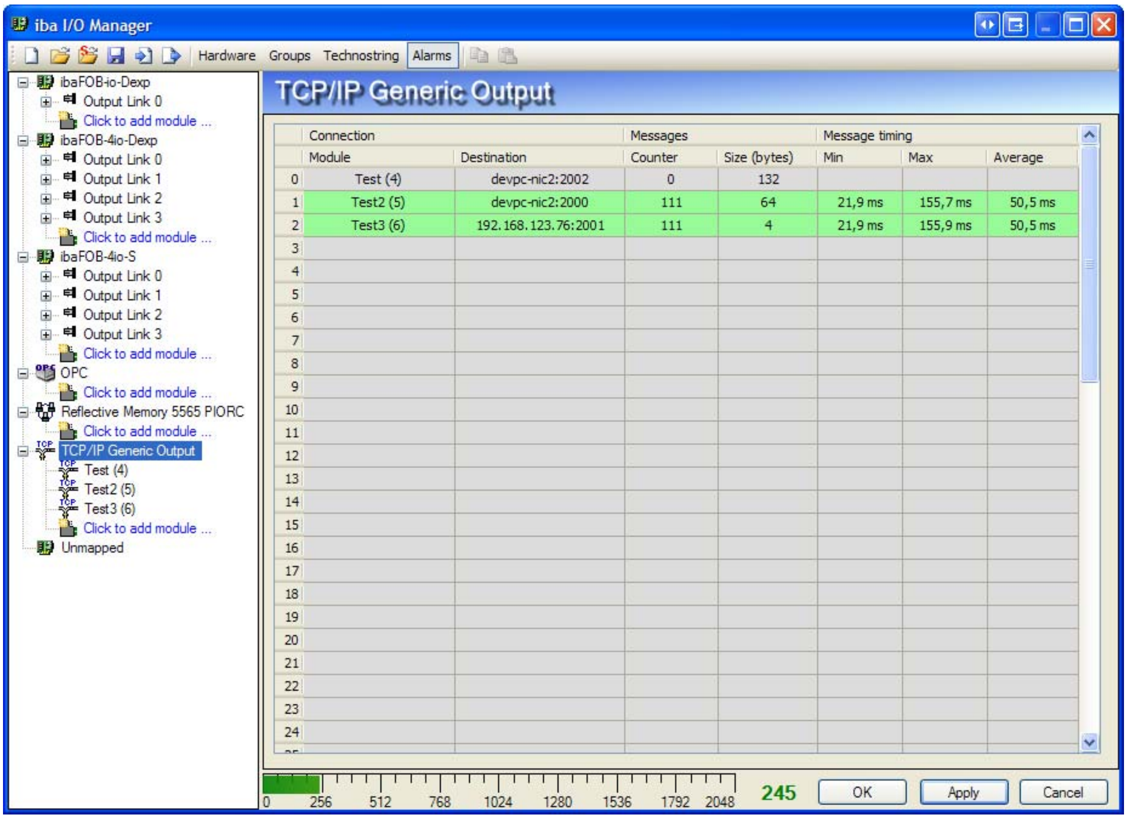
- Name: This is the signal name. It is only used for information.
- Expression: This determines the data that will be generated for the signal.
- Address: This is the address within the TCP message where the data for this signal will be written.
- DataType: This is the datatype of the signal. The possible datatypes are BYTE, INT, WORD, DINT, DWORD and FLOAT.
- Active: If the signal is active then the data generated by the expression will be written to the TCP message. If the signal is inactive then zero will be written to the TCP message.



	Name	Expression	Address	Bit no.	Active
0	true()	true()	0x80	0	☑
1	false()	false()	0x80	1	☑
2	true()	true()	0x80	2	☑
3	true()	true()	0x80	3	☑
4	true()	true()	0x80	4	☑
5	false()	false()	0x80	5	☑
6	false()	false()	0x80	6	☑
7	true()	true()	0x80	7	☑
8	false()	false()	0x80	8	☑
9	true()	true()	0x80	9	☑
10	false()	false()	0x80	10	☑
11	false()	false()	0x80	11	☑
12	false()	false()	0x80	12	☑
13	true()	true()	0x80	13	☑
14	true()	true()	0x80	14	☑
15	false()	false()	0x80	15	☑
16			0x82	0	☐
17			0x82	1	☐
18			0x82	2	☐
19			0x82	3	☐
20			0x82	4	☐
21			0x82	5	☐
22			0x82	6	☐
23	true()	true()	0x82	7	☑

Each digital signal has these properties:

- Name: This is the signal name. It is only used for information.
- Expression: This determines the data that will be generated for the signal.
- Address: This is the address within the TCP message where the data for this signal will be written.
- Bit no: This is the bit within the 16 bit integer that will be written. The bit number goes from 0 to 15.
- Active: If the signal is active then the data generated by the expression will be written to the TCP message. If the signal is inactive then zero will be written to the TCP message.

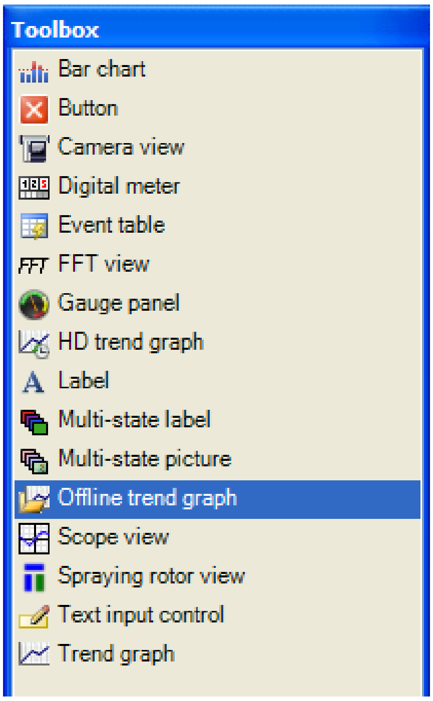


TCP/IP Generic Output

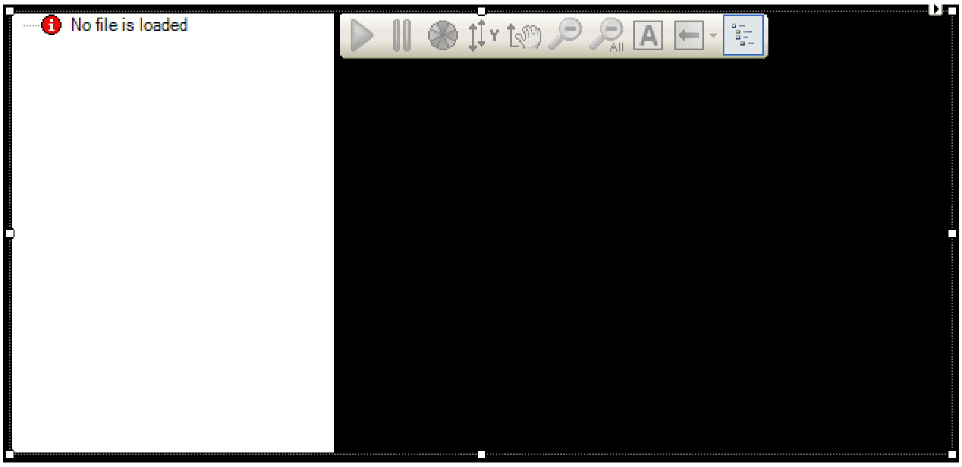
	Connection		Messages		Message timing		
	Module	Destination	Counter	Size (bytes)	Min	Max	Average
0	Test (4)	devpc-nic2:2002	0	132			
1	Test2 (5)	devpc-nic2:2000	111	64	21,9 ms	155,7 ms	50,5 ms
2	Test3 (6)	192.168.123.76:2001	111	4	21,9 ms	155,9 ms	50,5 ms
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							
21							
22							
23							
24							

When you select the interface in the tree then you get a table with diagnostic values about the configured connections. Each row corresponds to a connection/module. A row is green if the connection is established and it is gray when the connection is broken. The module column shows which module corresponds with the row. The destination shows the address and port number of the other side of the connection. The message counter shows how many messages have been sent successfully. The size shows the size in bytes of each message. The message size is determined by the maximum offset of all signals configured in the module. The message timing columns show the time between 2 messages. You can see the minimum, maximum and average time. The average time is taken over 100 messages.

3 Offline trend view

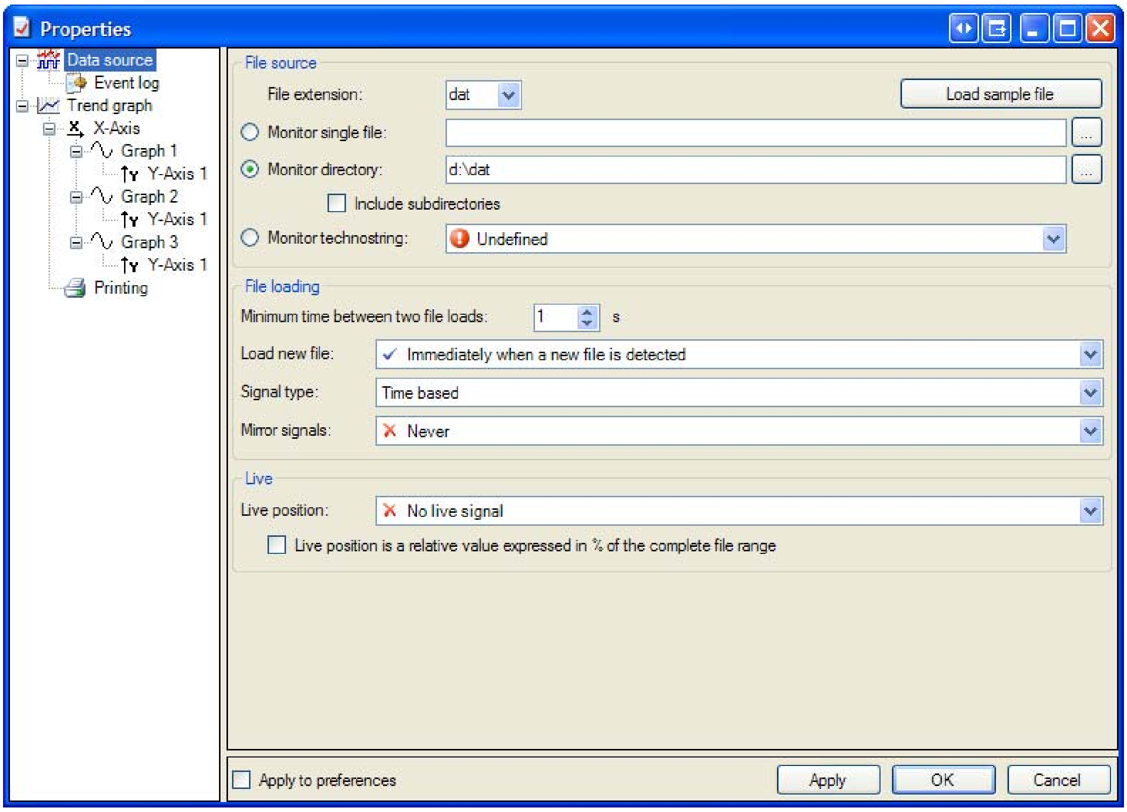


In QPanel there is a new control called the offline trend graph. It behaves like the normal trend graph but the data is coming from a dat file.



No file is loaded

The control itself has 2 parts. On the left there is a signal tree that shows the signals from the currently loaded file. On the right there is a trend graph. The visibility of the signal tree can be changed via the context menu of the signal tree, via the context menu of the trend graph and via a button on the trend graph toolbar.



File source

File extension: dat

○ Monitor single file:

● Monitor directory: d:\dat

□ Include subdirectories

○ Monitor technostring: Undefined

File loading

Minimum time between two file loads: 1 s

Load new file: ✓ Immediately when a new file is detected

Signal type: Time based

Mirror signals: ✗ Never

Live

Live position: ✗ No live signal

□ Live position is a relative value expressed in % of the complete file range

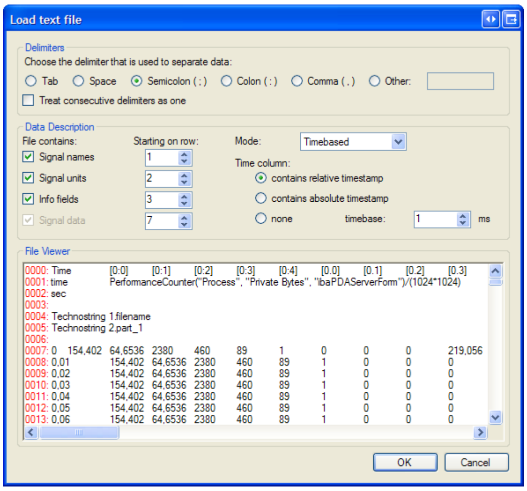
□ Apply to preferences

Initially there is no file loaded so we have to configure which file to load. Open the properties of the offline trend graph via the context menu. The first node is the data source node. There are three sections: file source, file loading and live.

In the file source section you first have to decide what file types you want to load. The offline trend graph supports iba dat files and text files. In the rest of the section you can select how the trend graph determines the next file to load. There are 3 options:

- Monitor single file: In this mode the trend graph reloads the same file each time its contents are changed.
- Monitor directory: In this mode the trend graph will load the most recent file within a directory and optionally its subdirectories.
- Monitor technostring: In this mode the trend graph receives the complete file path from a technostring. Each time the technostring changes the trend graph will try to load the file.

There is also a button to load a sample file immediately. If you have selected a text file then the text file settings form will open. Here you can specify how a text file should be parsed.



Load text file

Delimiters

Choose the delimiter that is used to separate data:

○ Tab ○ Space ● Semicolon (;) ○ Colon (:) ○ Comma (,) ○ Other:

☐ Treat consecutive delimiters as one

Data Description

File contains:	Starting on row:
☑ Signal names	1
☑ Signal units	2
☑ Info fields	3
☑ Signal data	7

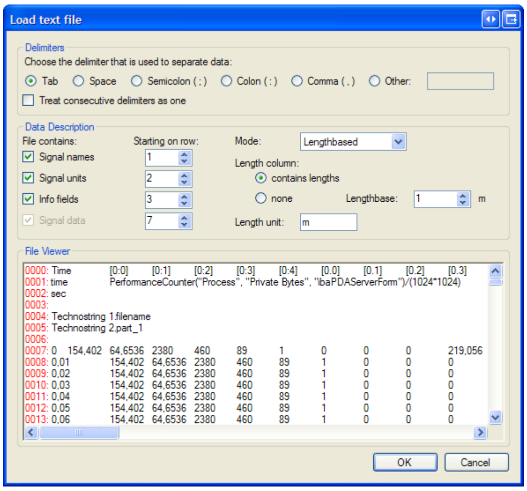
Mode: Timebased

Time column:
● contains relative timestamp
○ contains absolute timestamp
○ none timebase: 1 ms

File Viewer

```
0000: Time        [0:0]    [0:1]    [0:2]    [0:3]    [0:4]    [0.0]    [0.1]    [0.2]    [0.3]
0001: time        PerformanceCounter("Process", "Private Bytes", "ibaPDAServerForm")/(1024*1024)
0002: sec
0003:
0004: Technostring 1.filename
0005: Technostring 2.part_1
0006:
0007: 0   154,402  64,6536  2380     460      89       1        0        0        0        219,056
0008: 0,01        154,402  64,6536  2380     460      89       1        0        0        0
0009: 0,02        154,402  64,6536  2380     460      89       1        0        0        0
0010: 0,03        154,402  64,6536  2380     460      89       1        0        0        0
0011: 0,04        154,402  64,6536  2380     460      89       1        0        0        0
0012: 0,05        154,402  64,6536  2380     460      89       1        0        0        0
0013: 0,06        154,402  64,6536  2380     460      89       1        0        0        0
```

OK Cancel



Load text file

Delimiters

Choose the delimiter that is used to separate data:

● Tab ○ Space ○ Semicolon (;) ○ Colon (:) ○ Comma (,) ○ Other:

☐ Treat consecutive delimiters as one

Data Description

File contains:	Starting on row:
☑ Signal names	1
☑ Signal units	2
☑ Info fields	3
☑ Signal data	7

Mode: Lengthbased

Length column:
● contains lengths
○ none Lengthbase: 1 m

Length unit: m

File Viewer

```
0000: Time        [0:0]    [0:1]    [0:2]    [0:3]    [0:4]    [0.0]    [0.1]    [0.2]    [0.3]
0001: time        PerformanceCounter("Process", "Private Bytes", "ibaPDAServerForm")/(1024*1024)
0002: sec
0003:
0004: Technostring 1.filename
0005: Technostring 2.part_1
0006:
0007: 0   154,402  64,6536  2380     460      89       1        0        0        0        219,056
0008: 0,01        154,402  64,6536  2380     460      89       1        0        0        0
0009: 0,02        154,402  64,6536  2380     460      89       1        0        0        0
0010: 0,03        154,402  64,6536  2380     460      89       1        0        0        0
0011: 0,04        154,402  64,6536  2380     460      89       1        0        0        0
0012: 0,05        154,402  64,6536  2380     460      89       1        0        0        0
0013: 0,06        154,402  64,6536  2380     460      89       1        0        0        0
```

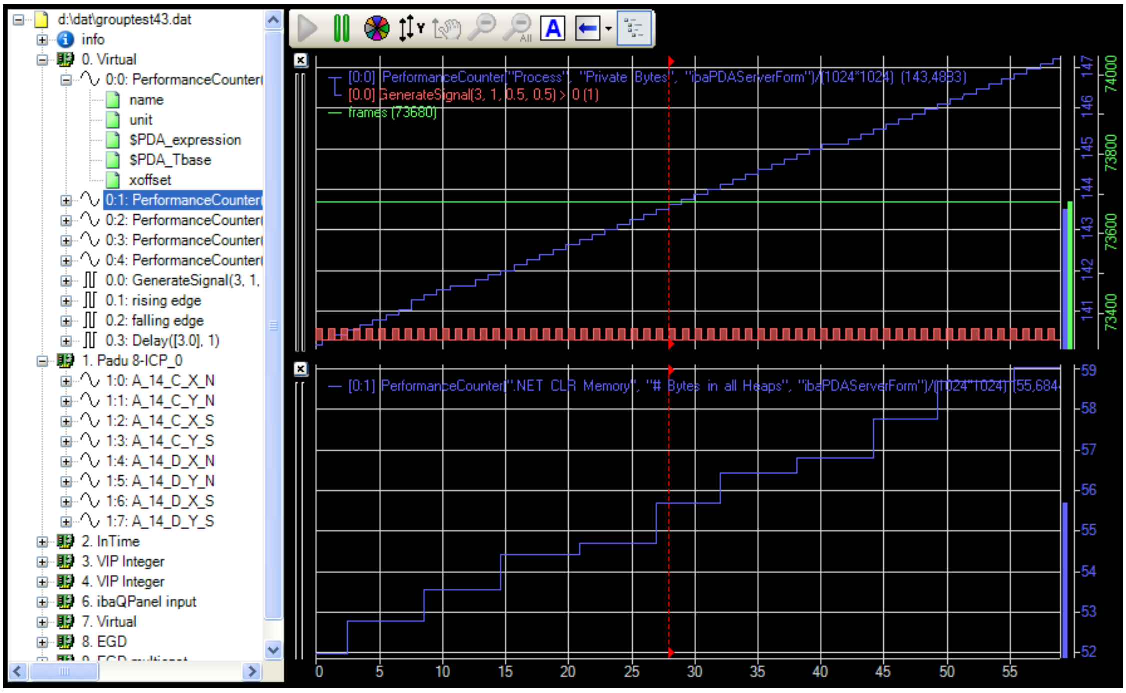
OK Cancel

This form shows you the first 1000 lines of the file in the file viewer section. The line numbers are shown in red. The first thing you have to select is the delimiter. In the data description section you can determine the row for signal names, signal units and info fields. You also have to specify on which row the data begins. Via the mode setting you can decide if the text file contains timebased or lengthbased data. If it contains timebased data then you can select if there is a time column or not. Two types of time are supported: relative time (in seconds) and absolute time (in the format used by the ibaAnalyzer export = day/month/year hour:minute:seconds.microseconds). If there is no time column then you have to set the timebase manually. In length mode you have to set if there is a length column or not. If there is no length column then you have to specify the length base. You can also set the length unit.

Back in in the file loading section of the datasource form you can determine when and how the newly found file is opened. First of all there is a minimum time between two file loads. The minimum time is 1s in order not to overload the PC. The new file can be loaded immediately or it can be loaded only when a rising edge on a digital trigger signal occurs. Next you have to select if you want to load time-based or length-based signals from the file. The signal data can be mirrored. You can never mirror, always mirror or let it determine by a digital signal. The trigger and mirror digital signals are signals coming from pda so NOT from the loaded dat file.

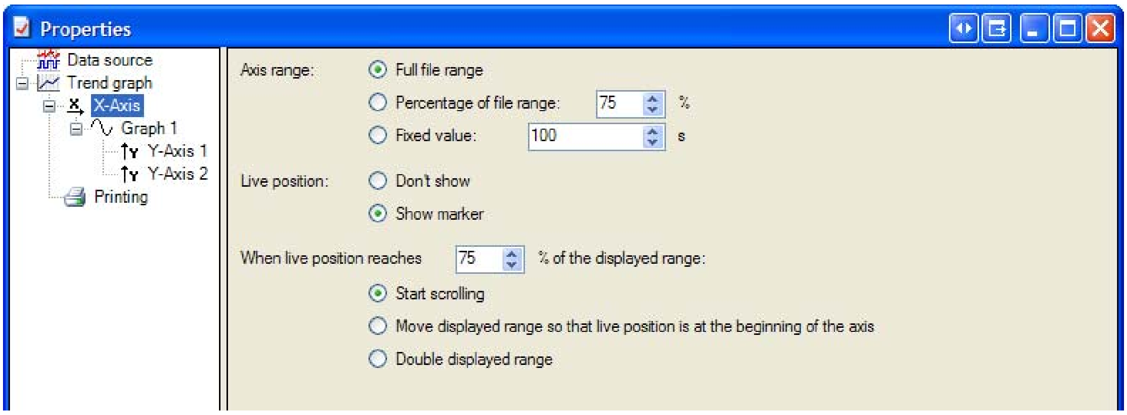
In the live section you can configure a live position signal. This signal comes from pda and it allows you to display a live positon marker on the trend graph. There are 2 modes for the signal:

- Absolute mode: The live position signal value determines the absolute position on the X-axis expressed in seconds for time-based signals and m (or the length unit from the file) for length-based signals.
- Relative mode: The live position signal value is expressed in a percentage of the complete file range. So its value has to go from 0% to 100%.



The screenshot above shows the offline trend graph in action. You can see the loaded dat file on the left. The tree contains the global infofields and all the signals with their infofields. You can doubleclick a signal or an infofield to add it to the trend graph. You can also drag & drop from the tree to the trend graph. The signal tree can also show groups and vectors if they are available in the dat file. Text channels are currently not supported. On the trend graph you can see the red live position marker.

The only difference between the normal trend graph and the offline trend graph are the properties of the X-axis.



Axis range:
● Full file range
○ Percentage of file range: 75 %
○ Fixed value: 100 s

Live position:
○ Don't show
● Show marker

When live position reaches 75 % of the displayed range:
● Start scrolling
○ Move displayed range so that live position is at the beginning of the axis
○ Double displayed range

You can setup the range of the X-axis. If you haven't configured a live position signal then the only mode that makes sense is the full file range mode. In this mode the X-axis will automatically show the complete signal data when a new dat file is loaded. If you use a live position signal then you can choose to only show a portion of the file range. This portion can be a percentage of the file range or a fixed value.

When the X-axis only shows a part of the file range then you can select how it will show more of the file’s data. First you decide when it should show more data. You do this by selecting what percentage of the visible range the live position should reach. When the live position reaches this percentage then there are 3 modes:

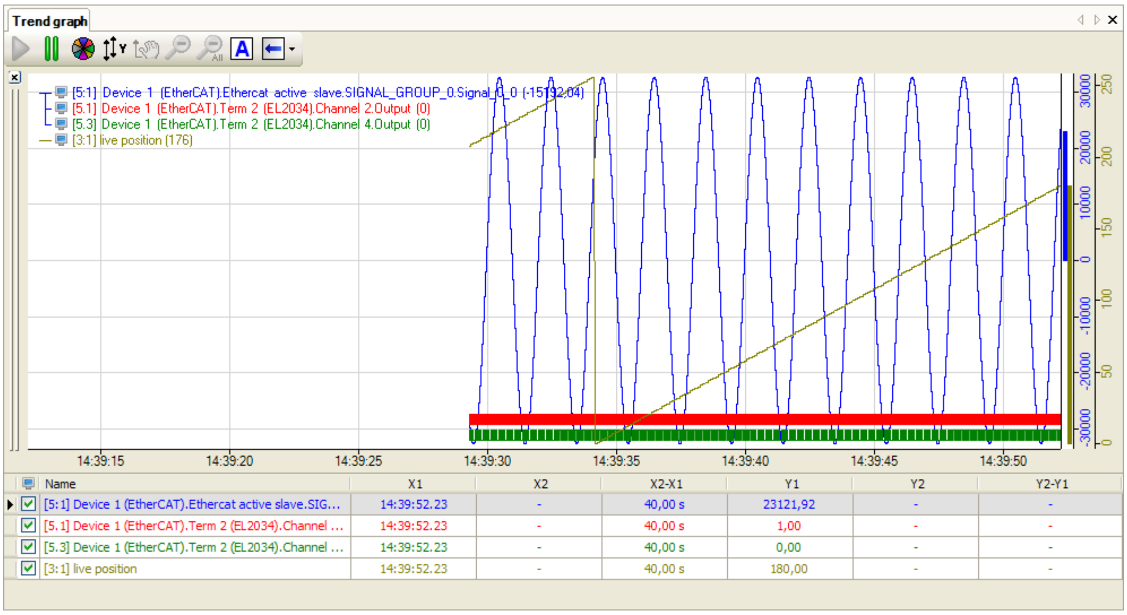
- The X-axis size stays the same and it just starts scrolling
- The X-axis size stays the same but its start position is set to the live position (so it jumps forwards)
- The X-axis size is doubled

The live position can be made visible via a marker.

4 Trend view changes

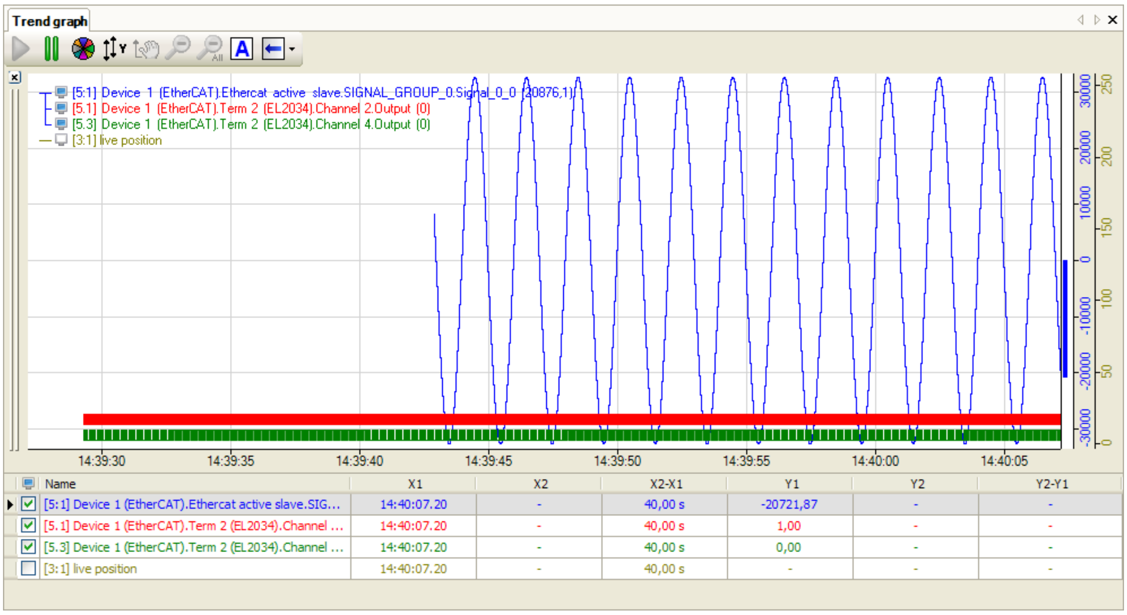
These changes have been made to the base trend graph so they are available in the normal trend graph, HD trend graph and offline trend graph.

4.1 Hide/show signals



[5:1] Device 1 (EtherCAT).Ethercat active slave.SIGNAL_GROUP_0.Signal_0_0 (-15192,04)
[5.1] Device 1 (EtherCAT).Term 2 (EL2034).Channel 2.Output (0)
[5.3] Device 1 (EtherCAT).Term 2 (EL2034).Channel 4.Output (0)
[3:1] live position (176)

	Name	X1	X2	X2-X1	Y1	Y2	Y2-Y1
☑	[5:1] Device 1 (EtherCAT).Ethercat active slave.SIG...	14:39:52.23	-	40,00 s	23121,92	-	-
☑	[5.1] Device 1 (EtherCAT).Term 2 (EL2034).Channel ...	14:39:52.23	-	40,00 s	1,00	-	-
☑	[5.3] Device 1 (EtherCAT).Term 2 (EL2034).Channel ...	14:39:52.23	-	40,00 s	0,00	-	-
☑	[3:1] live position	14:39:52.23	-	40,00 s	180,00	-	-



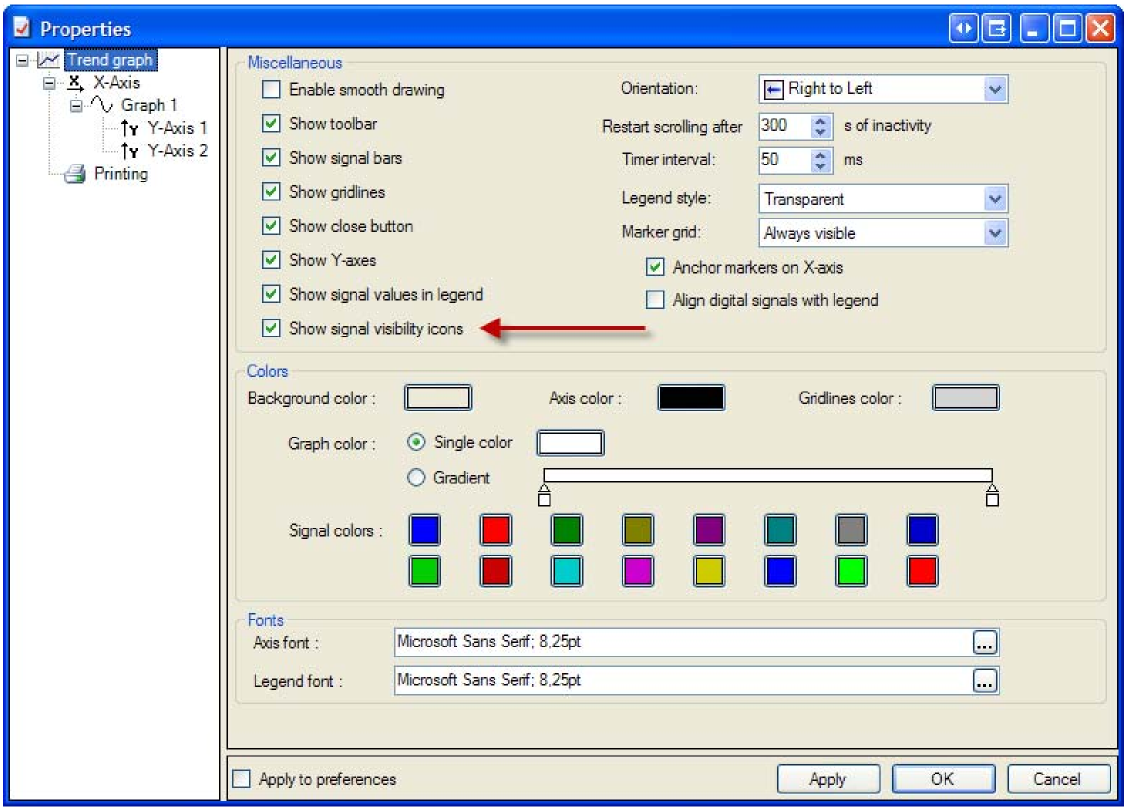
[5:1] Device 1 (EtherCAT).Ethercat active slave.SIGNAL_GROUP_0.Signal_0_0 (20876,1)
[5.1] Device 1 (EtherCAT).Term 2 (EL2034).Channel 2.Output (0)
[5.3] Device 1 (EtherCAT).Term 2 (EL2034).Channel 4.Output (0)
[3:1] live position

	Name	X1	X2	X2-X1	Y1	Y2	Y2-Y1
☑	[5:1] Device 1 (EtherCAT).Ethercat active slave.SIG...	14:40:07.20	-	40,00 s	-20721,87	-	-
☑	[5.1] Device 1 (EtherCAT).Term 2 (EL2034).Channel ...	14:40:07.20	-	40,00 s	1,00	-	-
☑	[5.3] Device 1 (EtherCAT).Term 2 (EL2034).Channel ...	14:40:07.20	-	40,00 s	0,00	-	-
□	[3:1] live position	14:40:07.20	-	40,00 s	-	-	-

The top screenshot shows a graph with 4 visible signals. The bottom screenshot shows the same 4 signals but now the 4th signal is hidden. A hidden signal is no longer visible in the graph. Its signal bar is empty and no current or marker values are shown in the legend and in the signal grid. The signal name is still visible in the legend and there is also a row for the signal in the signal grid.

There are 3 ways to hide/show a signal:

- Right click on a signal and use the context menu to hide or show a signal
- Click the checkbox in the signal grid
- Configure the trend graph to show the visibility icon in the legend and then click on the visibility icon.



Properties

- Trend graph
  - X-Axis
    - Graph 1
      - Y-Axis 1
      - Y-Axis 2
  - Printing

Miscellaneous

☐ Enable smooth drawing
☑ Show toolbar
☑ Show signal bars
☑ Show gridlines
☑ Show close button
☑ Show Y-axes
☑ Show signal values in legend
☑ Show signal visibility icons

Orientation: Right to Left
Restart scrolling after 300 s of inactivity
Timer interval: 50 ms
Legend style: Transparent
Marker grid: Always visible
☑ Anchor markers on X-axis
☐ Align digital signals with legend

Colors

Background color : ⬜ Axis color : ⬛ Gridlines color : ⬜

Graph color : ● Single color ⬜
○ Gradient

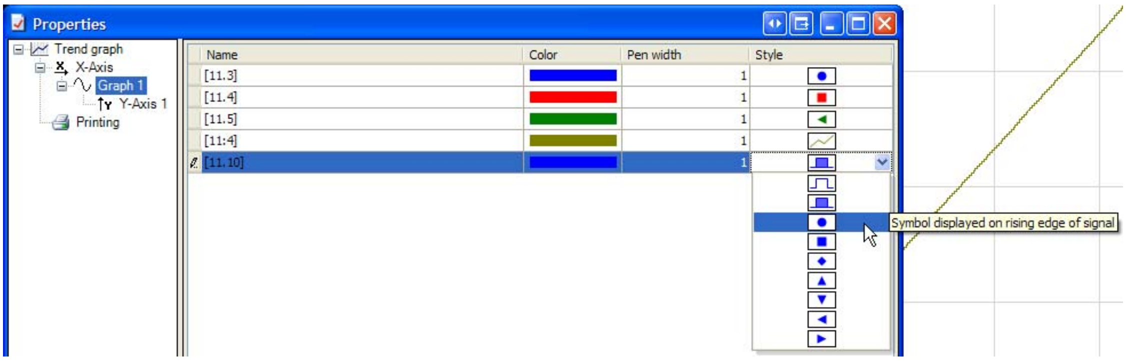
Signal colors : 🟦 🟥 🟩 🟫 🟪 🟦 ⬜ 🟦
🟩 🟥 🟦 🟪 🟨 🟦 🟩 🟥

Fonts

Axis font : Microsoft Sans Serif; 8,25pt ...
Legend font : Microsoft Sans Serif; 8,25pt ...

☐ Apply to preferences

4.2 Draw styles for signals



Properties

- Trend graph
  - X-Axis
    - Graph 1
      - Y-Axis 1
  - Printing

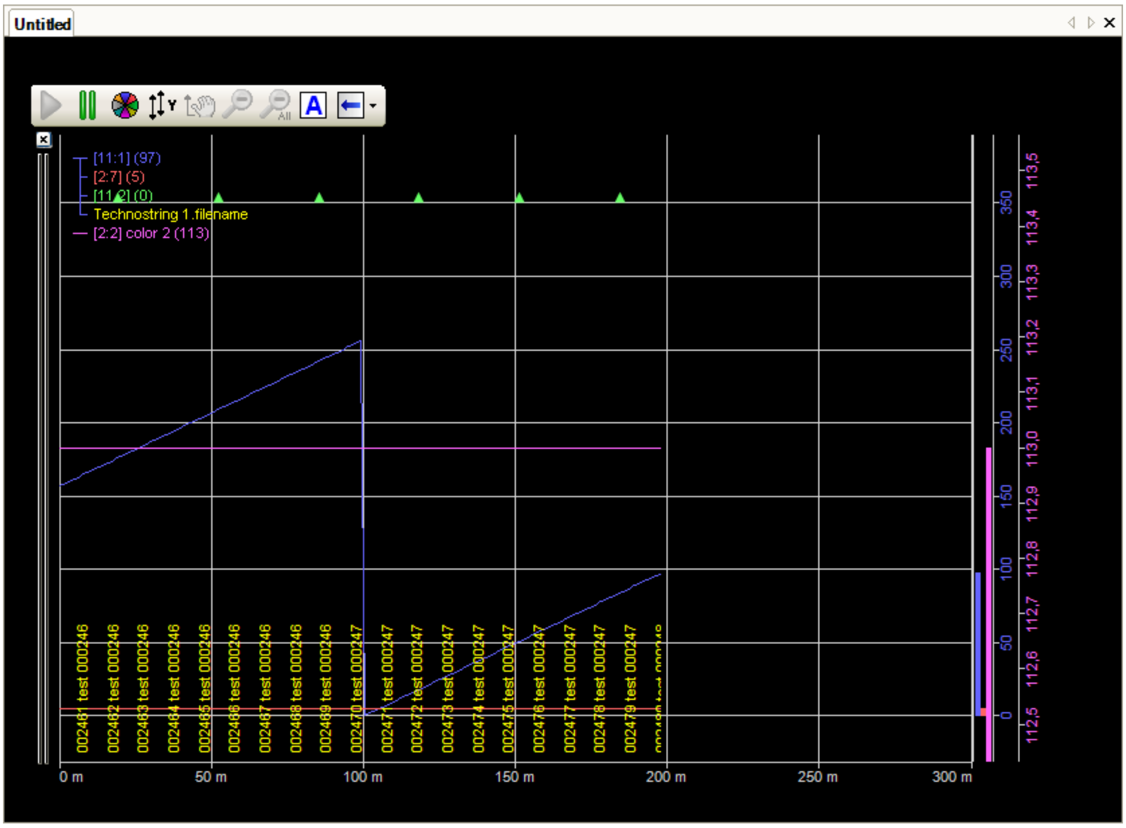
Name	Color	Pen width	Style
[11.3]	🟦	1	●
[11.4]	🟥	1	■
[11.5]	🟩	1	◀
[11:4]	🟫	1	
[11.10]	🟦	1	

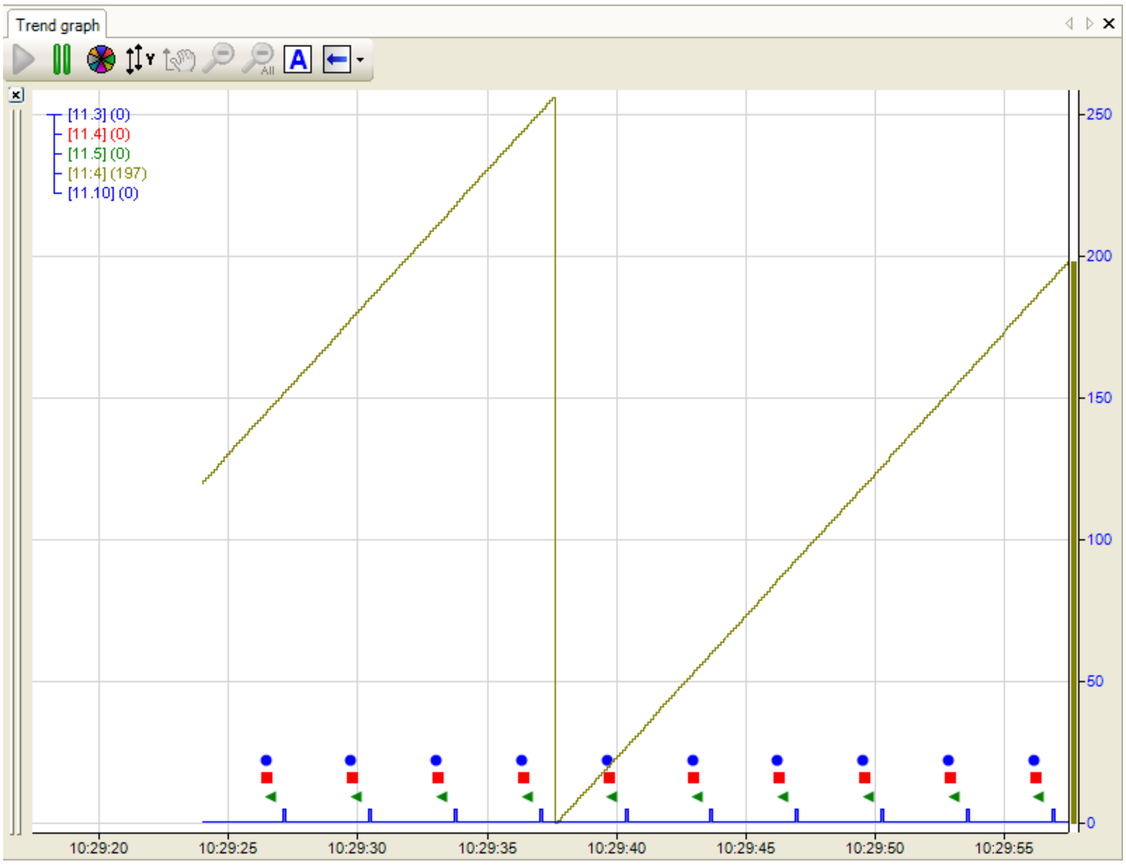
In previous versions you could select if a signal was drawn filled or non-filled. Now this is replaced with a drawing style property. For analog signals you can still decide between a non-filled line and a filled line. For digital signals there are more options. The filled and non-filled line still exist but you can now also decide to draw a symbol at each rising edge of the digital signal. There are 7 possible symbols:

- Circle

- Square
- Diamond
- Triangle pointing up
- Triangle pointing down
- Triangle pointing left
- Triangle pointing right

The following screen shots show these draw styles on a Q-Panel trend graph and on a normal trend graph.

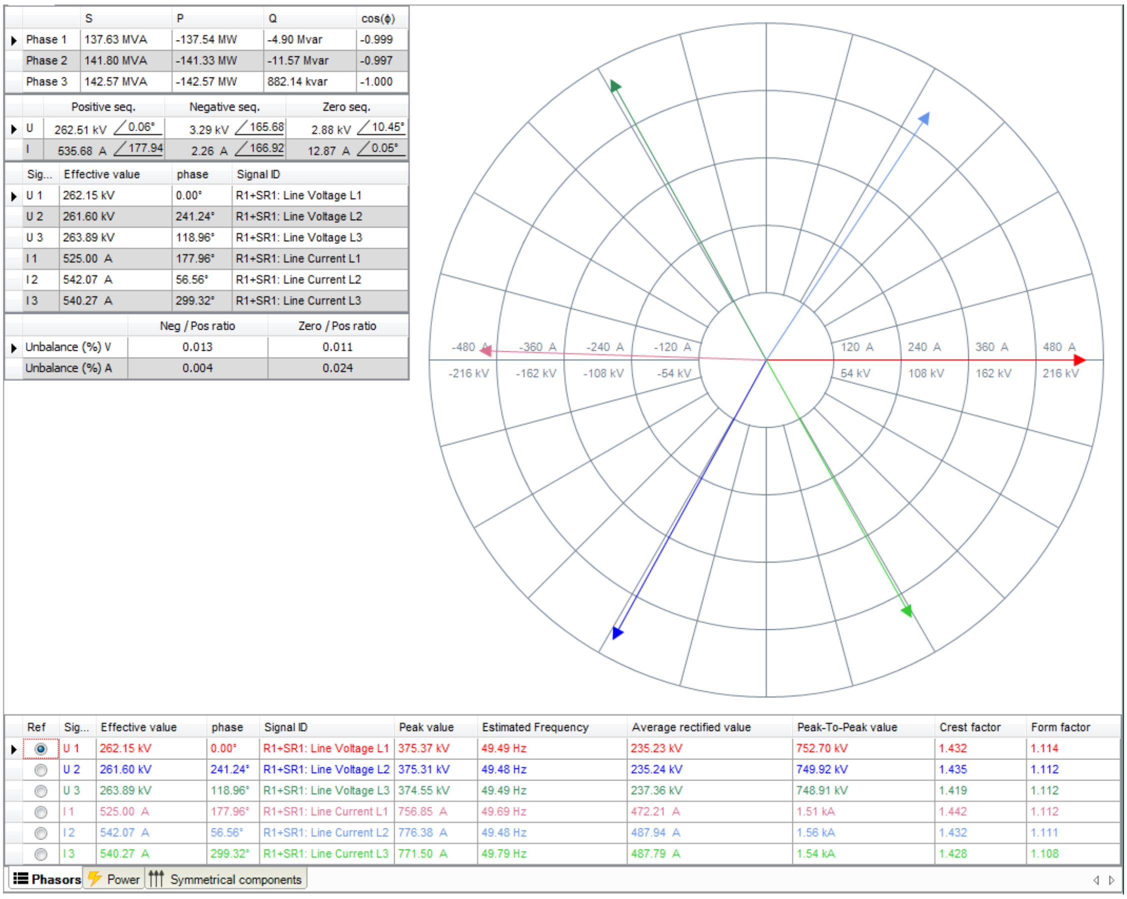




The digital symbol draw styles are not supported on the HD trend graph. This is because on the higher levels of a digital signal there is not enough information available to determine if there is a rising or a falling edge.

5 Phasor view

See separate document: TechSpec_Phasor_View.doc



	S	P	Q	cos(ϕ)
Phase 1	137.63 MVA	-137.54 MW	-4.90 Mvar	-0.999
Phase 2	141.80 MVA	-141.33 MW	-11.57 Mvar	-0.997
Phase 3	142.57 MVA	-142.57 MW	882.14 kvar	-1.000

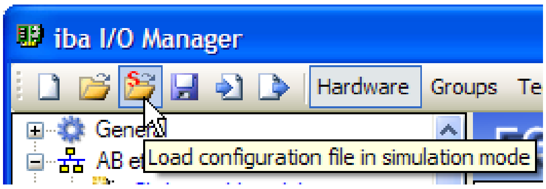
	Positive seq.	Negative seq.	Zero seq.
U	262.51 kV ∠0.06°	3.29 kV ∠165.68	2.88 kV ∠10.45°
I	535.68 A ∠177.94	2.26 A ∠166.92	12.87 A ∠0.05°

Sig...	Effective value	phase	Signal ID
U 1	262.15 kV	0.00°	R1+SR1: Line Voltage L1
U 2	261.60 kV	241.24°	R1+SR1: Line Voltage L2
U 3	263.89 kV	118.96°	R1+SR1: Line Voltage L3
I 1	525.00 A	177.96°	R1+SR1: Line Current L1
I 2	542.07 A	56.56°	R1+SR1: Line Current L2
I 3	540.27 A	299.32°	R1+SR1: Line Current L3

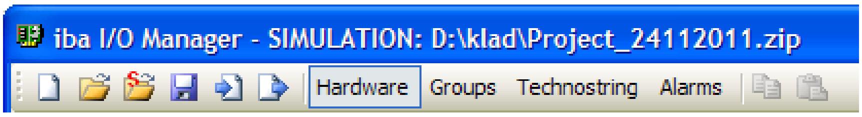
	Neg / Pos ratio	Zero / Pos ratio
Unbalance (%) V	0.013	0.011
Unbalance (%) A	0.004	0.024

Ref	Sig...	Effective value	phase	Signal ID	Peak value	Estimated Frequency	Average rectified value	Peak-To-Peak value	Crest factor	Form factor
●	U 1	262.15 kV	0.00°	R1+SR1: Line Voltage L1	375.37 kV	49.49 Hz	235.23 kV	752.70 kV	1.432	1.114
○	U 2	261.60 kV	241.24°	R1+SR1: Line Voltage L2	375.31 kV	49.48 Hz	235.24 kV	749.92 kV	1.435	1.112
○	U 3	263.89 kV	118.96°	R1+SR1: Line Voltage L3	374.55 kV	49.49 Hz	237.36 kV	748.91 kV	1.419	1.112
○	I 1	525.00 A	177.96°	R1+SR1: Line Current L1	756.85 A	49.69 Hz	472.21 A	1.51 kA	1.442	1.112
○	I 2	542.07 A	56.56°	R1+SR1: Line Current L2	776.38 A	49.48 Hz	487.94 A	1.56 kA	1.432	1.111
○	I 3	540.27 A	299.32°	R1+SR1: Line Current L3	771.50 A	49.79 Hz	487.79 A	1.54 kA	1.428	1.108

6 Simulation mode



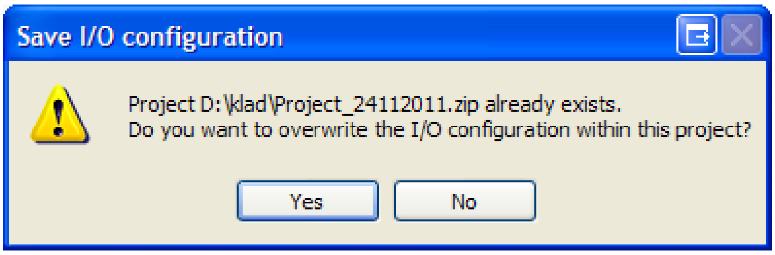
There is a new button in the I/O manager that allows you to load an I/O configuration in simulation mode. Simulation mode means that the configuration is not changed to match the local hardware. This is the same functionality as the button that used to be on the unmapped interface.



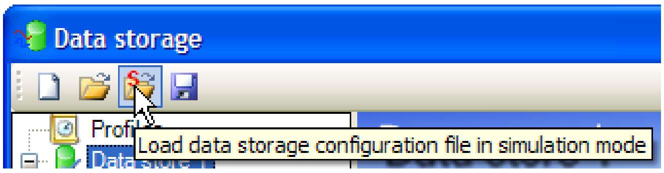
You can load either an I/O configuration file (.io) or a project file (.zip). After you load it in simulation mode the title bar of the I/O manager window changes and shows you simulation and the full path of the loaded file. You can also drag and drop a .io or a .zip file from windows explorer into the tree to open the configuration in simulation mode.

The simulation mode can be used by the support department to diagnose customer problems. It can also be used to prepare an I/O configuration when the hardware is not available or the dongle licenses are not available. In simulation mode you can right-click on the general node to add interfaces. You can also add modules that you don’t have a license for.

The save functionality has been extended to save the I/O configuration inside of a project zip file. In the save dialog you just have to type in a filename with the .zip extension. If the file doesn’t exist then ibaPDA will create a new project zip file that contains the saved I/O configuration (CurrentIoConfig.io) and the exported I/O configuration (IoConfig.txt). If the zip file already existed then a messagebox will appear asking you if you want to update the I/O configuration within the existing project.



Project D:\klad\Project_24112011.zip already exists.
Do you want to overwrite the I/O configuration within this project?

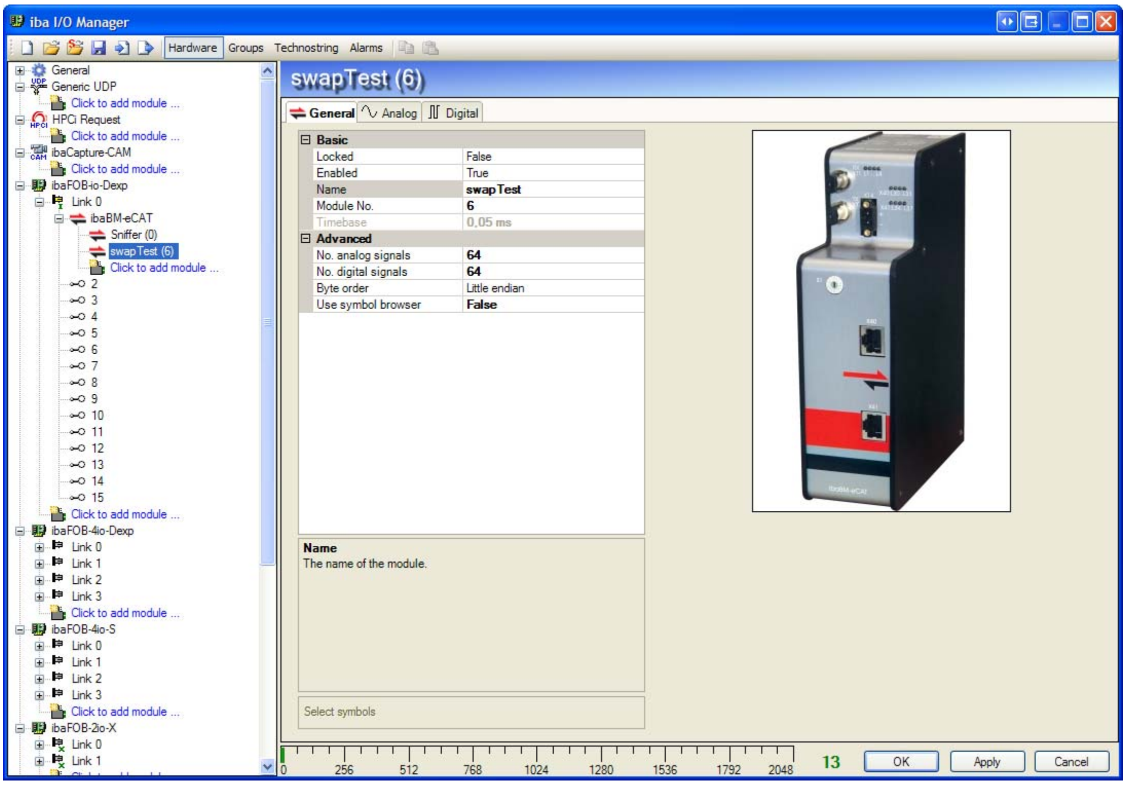


The data storage manager has also been extended with a simulation mode. In simulation mode you can load a project zip file or drag and drop it into the tree. The I/O configuration of the project will be loaded and the data store configuration will be loaded. This way you can configure the data storage without applying the I/O configuration. This can also be used by the support department to see how the data stores are configured on a customer’s system.

Similar to the I/O manager save functionality the data storage manager can save the data storage configuration in a project zip file.

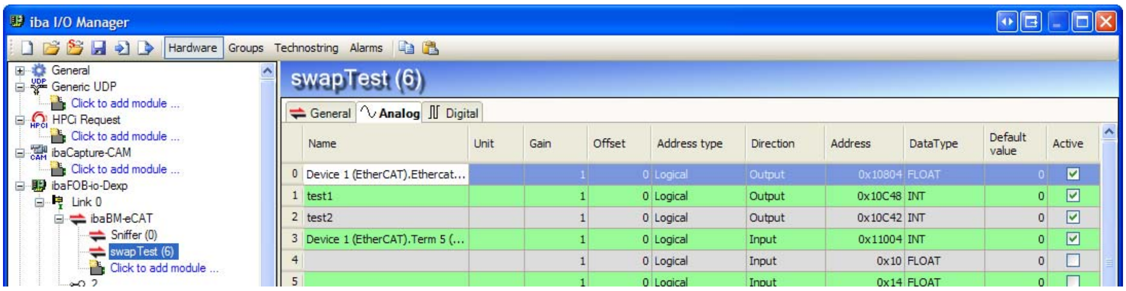
7 ibaBM-eCAT changes

The ibaBM-eCAT module has a new property called “Use symbol browser”. It allows you to switch between symbol mode and raw address mode. In symbol mode you use the symbol browser to select the signals and the address information is retrieved from the addressbook. In raw address mode you have to specify all the address information yourself. The address information consists of address type, direction, address, bit number and or data type. In the previous versions you could only switch between the two modes by setting or clearing the symbol path on the ibaBM-eCAT device. The following screenshot shows a module in raw address mode.



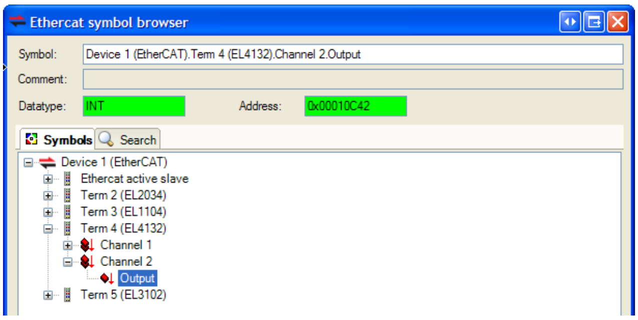
Basic	
Locked	False
Enabled	True
Name	**swap Test**
Module No.	**6**
Timebase	**0,05 ms**
Advanced	
No. analog signals	**64**
No. digital signals	**64**
Byte order	Little endian
Use symbol browser	**False**

Name
The name of the module.



	Name	Unit	Gain	Offset	Address type	Direction	Address	DataType	Default value	Active
0	Device 1 (EtherCAT).Ethercat...		1	0	Logical	Output	0x10804	FLOAT	0	☑
1	test1		1	0	Logical	Output	0x10C48	INT	0	☑
2	test2		1	0	Logical	Output	0x10C42	INT	0	☑
3	Device 1 (EtherCAT).Term 5 (...		1	0	Logical	Input	0x11004	INT	0	☑
4			1	0	Logical	Input	0x10	FLOAT	0	☐
5			1	0	Logical	Input	0x14	FLOAT	0	☐

The symbol browser has been extended with an address field. It shows you the logical address of the selected symbol. In symbol mode there is a tooltip when you hover over the symbol column. This tooltip now also contains all the address information for the signal under the mouse.

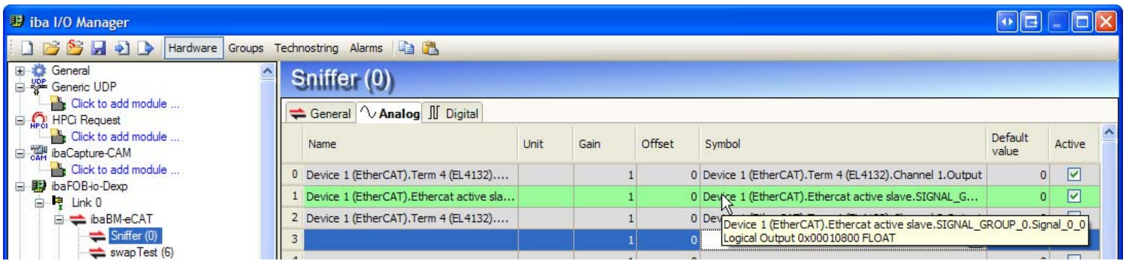


Symbol: Device 1 (EtherCAT).Term 4 (EL4132).Channel 2.Output

Comment:

Datatype: INT

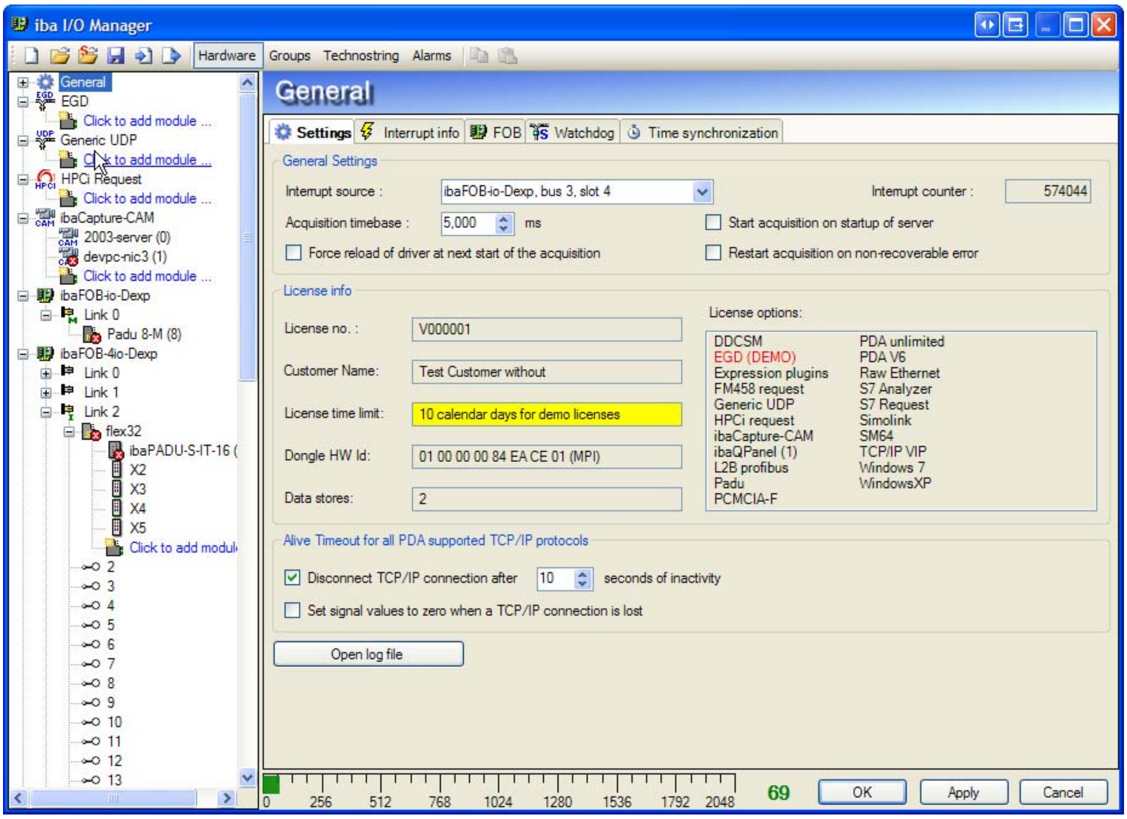
Address: 0x00010C42



	Name	Unit	Gain	Offset	Symbol	Default value	Active
0	Device 1 (EtherCAT).Term 4 (EL4132)....		1	0	Device 1 (EtherCAT).Term 4 (EL4132).Channel 1.Output	0	☑
1	Device 1 (EtherCAT).Ethercat active sla...		1	0	Device 1 (EtherCAT).Ethercat active slave.SIGNAL_G...	0	☑
2	Device 1 (EtherCAT).Term 4 (EL4132)....		1	0	Dev		
3			1	0			

Device 1 (EtherCAT).Ethercat active slave.SIGNAL_GROUP_0.Signal_0_0
Logical Output 0x00010800 FLOAT

8 Demo licenses



General

General Settings

Interrupt source : ibaFOB-io-Dexp, bus 3, slot 4

Interrupt counter : 574044

Acquisition timebase : 5,000 ms

□ Start acquisition on startup of server

□ Force reload of driver at next start of the acquisition

□ Restart acquisition on non-recoverable error

License info

License no. : V000001

Customer Name: Test Customer without

License time limit: 10 calendar days for demo licenses

Dongle HW Id: 01 00 00 00 84 EA CE 01 (MPI)

Data stores: 2

License options:

DDCSM	PDA unlimited
EGD (DEMO)	PDA V6
Expression plugins	Raw Ethernet
FM458 request	S7 Analyzer
Generic UDP	S7 Request
HPCi request	Simolink
ibaCapture-CAM	SM64
ibaQPanel (1)	TCP/IP VIP
L2B profibus	Windows 7
Padu	WindowsXP
PCMCIA-F	

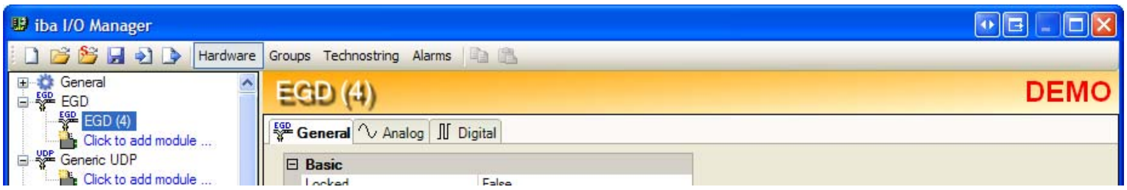
Alive Timeout for all PDA supported TCP/IP protocols

☑ Disconnect TCP/IP connection after 10 seconds of inactivity

□ Set signal values to zero when a TCP/IP connection is lost

Open log file

IbaPDA uses a new dongle library that supports both MPI and SmarxOS dongles. The dongle type is shown after the dongle hardware ID. The new dongle library also supports up to 8 demo licenses. In the license info in the I/O manager you can see the days remaining for the demo licenses. In the license options you can recognize demo licenses by their red color and the text (DEMO) appended to them. This same information is also shown in the license info of the server status program.

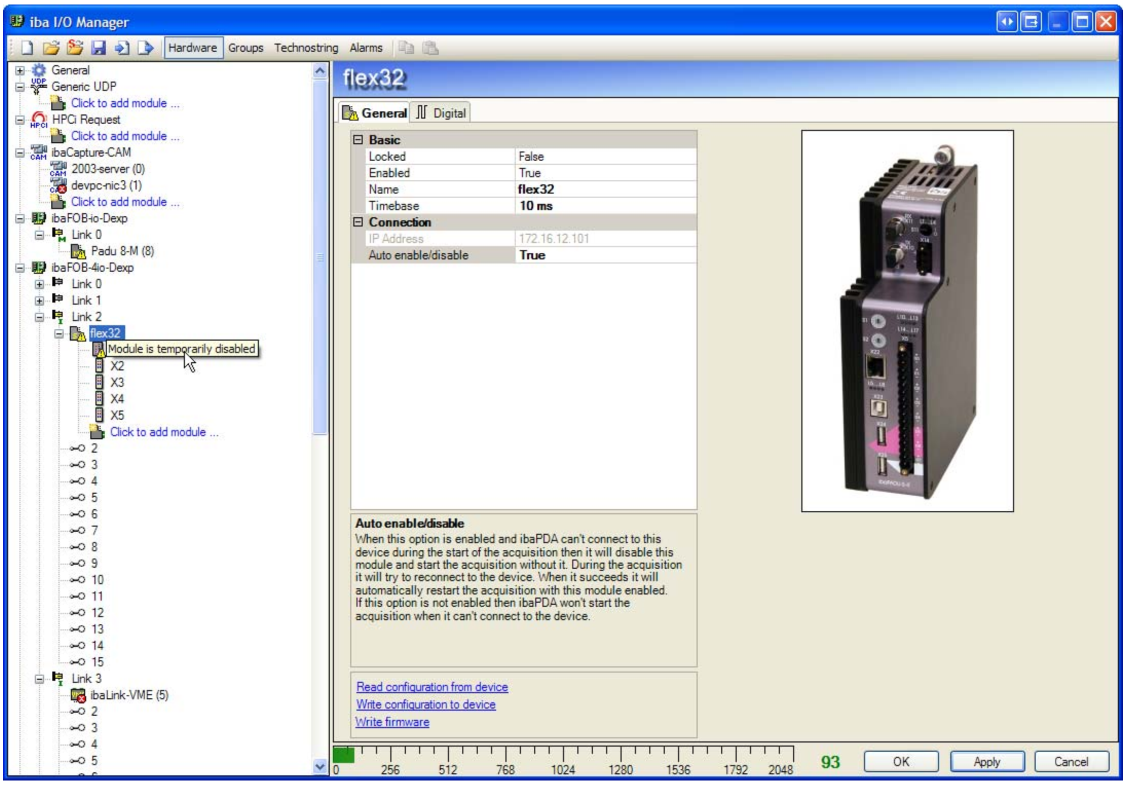


Basic

Locked False

The I/O manger header changes color and shows DEMO when you select an interface or module that you only have a demo license for.

9 Auto enable/disable of flex- and M-modules



Module is temporarily disabled

flex32

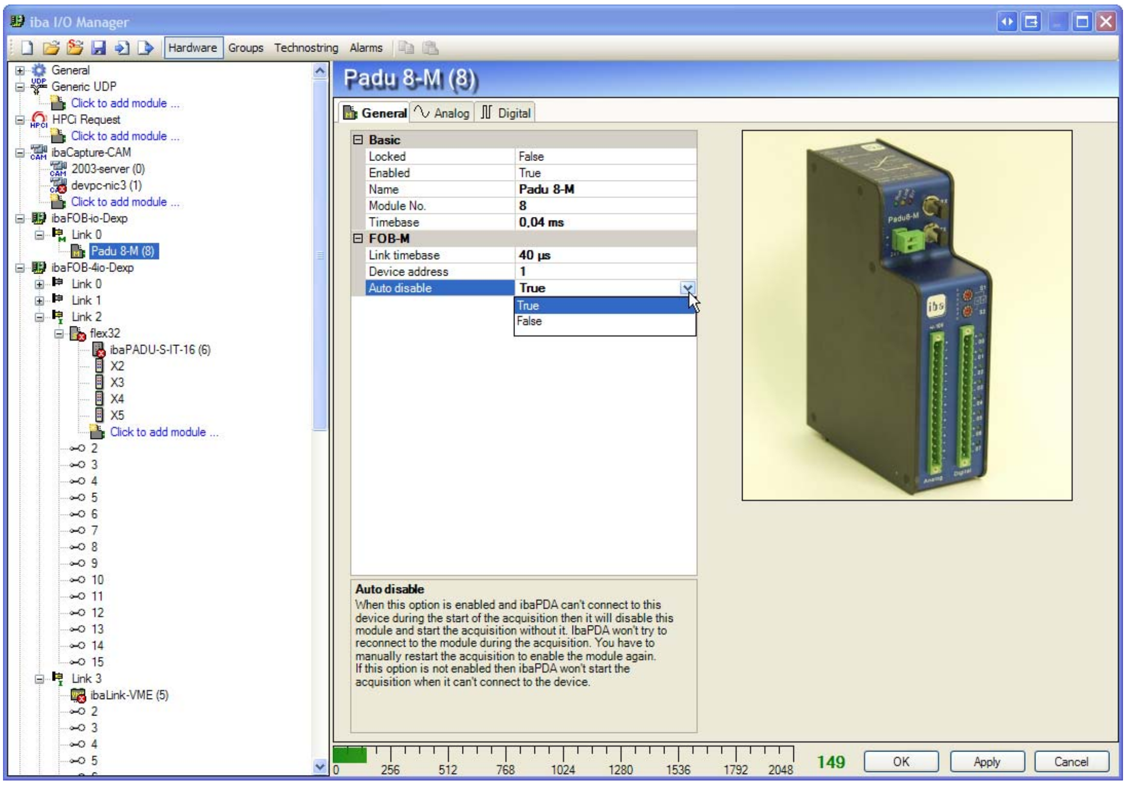
Basic	
Locked	False
Enabled	True
Name	flex32
Timebase	10 ms
Connection	
IP Address	172.16.12.101
Auto enable/disable	True

Auto enable/disable
When this option is enabled and ibaPDA can't connect to this device during the start of the acquisition then it will disable this module and start the acquisition without it. During the acquisition it will try to reconnect to the device. When it succeeds it will automatically restart the acquisition with this module enabled. If this option is not enabled then ibaPDA won't start the acquisition when it can't connect to the device.

Read configuration from device
Write configuration to device
Write firmware

In previous versions of ibaPDA you couldn’t start the acquisition if a flex module like ibaPADU-S-IT, ibaBM-DDCS, ibaBM-eCAT, … or a FOB-M module like Padu8-M, Padu16-M or Padu8-ICP wasn’t connected to the system. You had to manually disable the module before you could start the acquisition.

Now the flex modules have a property called “Auto enable/disable”. When this property is set ibaPDA will automatically disable the module when it can’t connect to the device during the start of the acquisition. When the acquisition is running it will periodically try to connect to the device. When it succeeds it will automatically restart the acquisition with this module enabled.



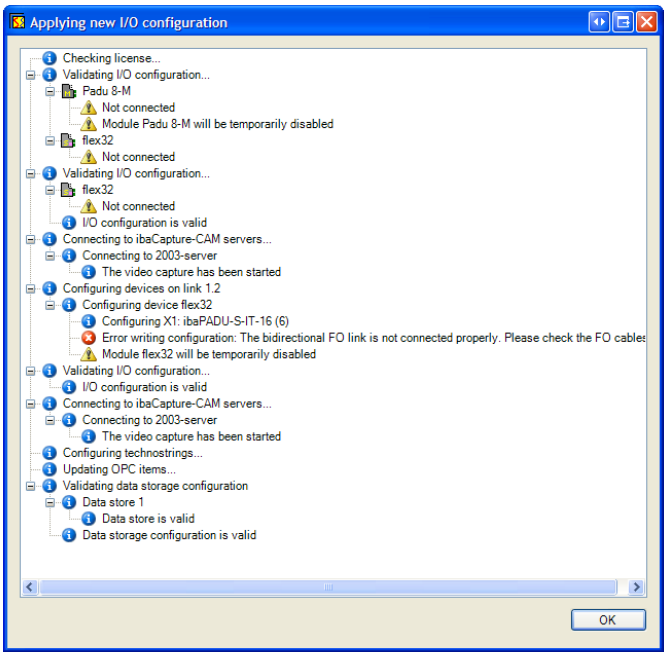
Padu 8-M (8)

Basic	
Locked	False
Enabled	True
Name	**Padu 8-M**
Module No.	**8**
Timebase	**0.04 ms**
FOB-M	
Link timebase	**40 µs**
Device address	**1**
Auto disable	**True**

True
False

Auto disable
When this option is enabled and ibaPDA can't connect to this device during the start of the acquisition then it will disable this module and start the acquisition without it. IbaPDA won't try to reconnect to the module during the acquisition. You have to manually restart the acquisition to enable the module again.
If this option is not enabled then ibaPDA won't start the acquisition when it can't connect to the device.

The FOB-M modules have a similar property called “Auto disable”. This will automatically disable a FOB-M device when it isn’t connected when the acquisition is started. IbaPda will **NOT** try to connect to it during the acquisition. You have to manually restart the acquisition when the device is reconnected.

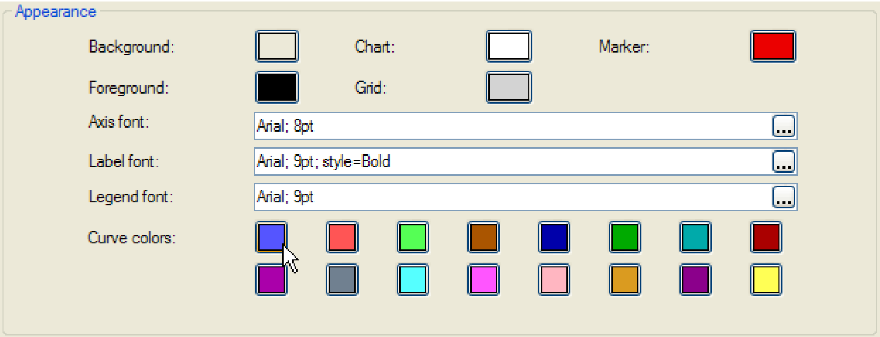


```
Checking license...
Validating I/O configuration...
    Padu 8-M
        Not connected
        Module Padu 8-M will be temporarily disabled
    flex32
        Not connected
Validating I/O configuration...
    flex32
        Not connected
    I/O configuration is valid
Connecting to ibaCapture-CAM servers...
    Connecting to 2003-server
        The video capture has been started
Configuring devices on link 1.2
    Configuring device flex32
        Configuring X1: ibaPADU-S-IT-16 (6)
        Error writing configuration: The bidirectional FO link is not connected properly. Please check the FO cables
        Module flex32 will be temporarily disabled
Validating I/O configuration...
    I/O configuration is valid
Connecting to ibaCapture-CAM servers...
    Connecting to 2003-server
        The video capture has been started
Configuring technostrings...
Updating OPC items...
Validating data storage configuration
    Data store 1
        Data store is valid
    Data storage configuration is valid
```

The screenshot shows the validation form with a Padu8-M module that is not connected and an ibaPADU-S-IT module that is not connected. You can see that the first time the Padu8-M module is disabled. Then the validation is restarted and you see that pda can’t connect to the ibaPADU-S-IT so it is disabled as well. The validation is then started for the third time and now everything is ok and the acquisition is started.

10 Standardized color pickers

The different color pickers in the different views within ibaPDA and ibaQPanel have been standardized. There are now 2 color pickers: one is a button and the other one is an editor in a grid. They both show the same color picker form. This form is the standard Windows color picker form.



Appearance

Background: ⬜ Chart: ⬜ Marker: 🟥

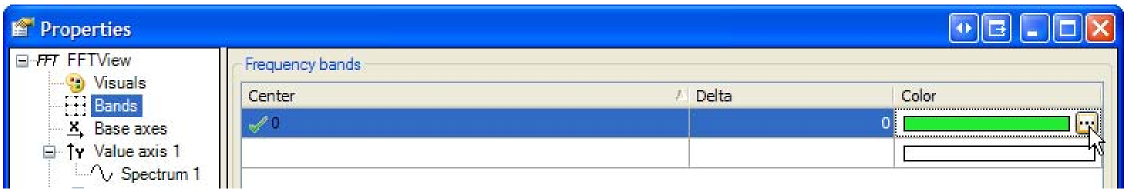
Foreground: ⬛ Grid: ⬜

Axis font: Arial; 8pt

Label font: Arial; 9pt; style=Bold

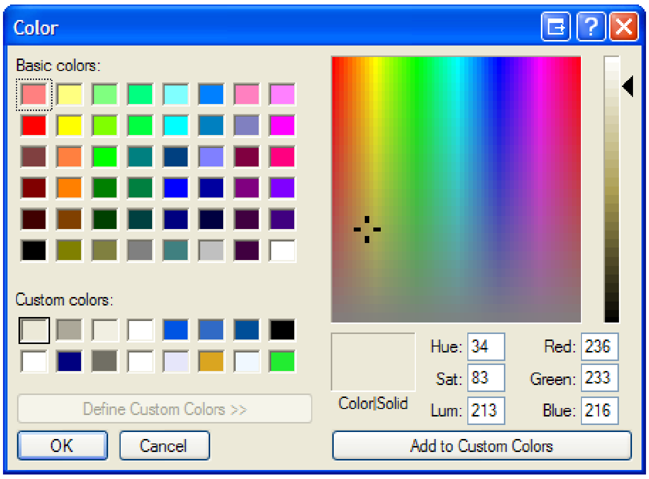
Legend font: Arial; 9pt

Curve colors:



Frequency bands

Center	Delta	Color
0	0	🟩



Hue: 34 Sat: 83 Lum: 213

Red: 236 Green: 233 Blue: 216

There are 16 possible custom colors in this form. You can change them via the “Add to Custom Colors” button. These 16 colors are saved so that they can be reused in all controls.