ibaPDA v8.1.0

New Features

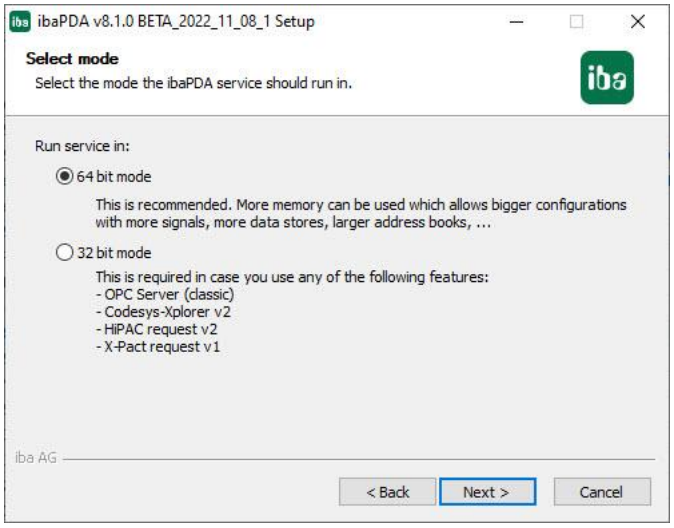
05.01.2023
iba AG

Table of contents

1 64-bit ibaPDA server

The ibaPDA service has always been a 32-bit process. This meant that only 2 GB of memory could be used by the process even when it ran on a 64-bit operating system. Now in ibaPDA v8.1.0 the service is also able to run as a 64-bit process on a 64-bit operating system.

When installing ibaPDA v8.1.0 you can choose whether you install the 32-bit or the 64-bit version of the service.



The default choice is 64-bit. A 64-bit process can use more than 2GB of memory. This allows ibaPDA to handle bigger configurations with lots of signals, data stores, large address books …

The 64-bit version does not support the following features:

- OPC Server (classic)
- Codesys-Xplorer v2
- HiPAC request v2
- X-Pact request v1

If your configuration uses one of these features then you should install the 32-bit version.

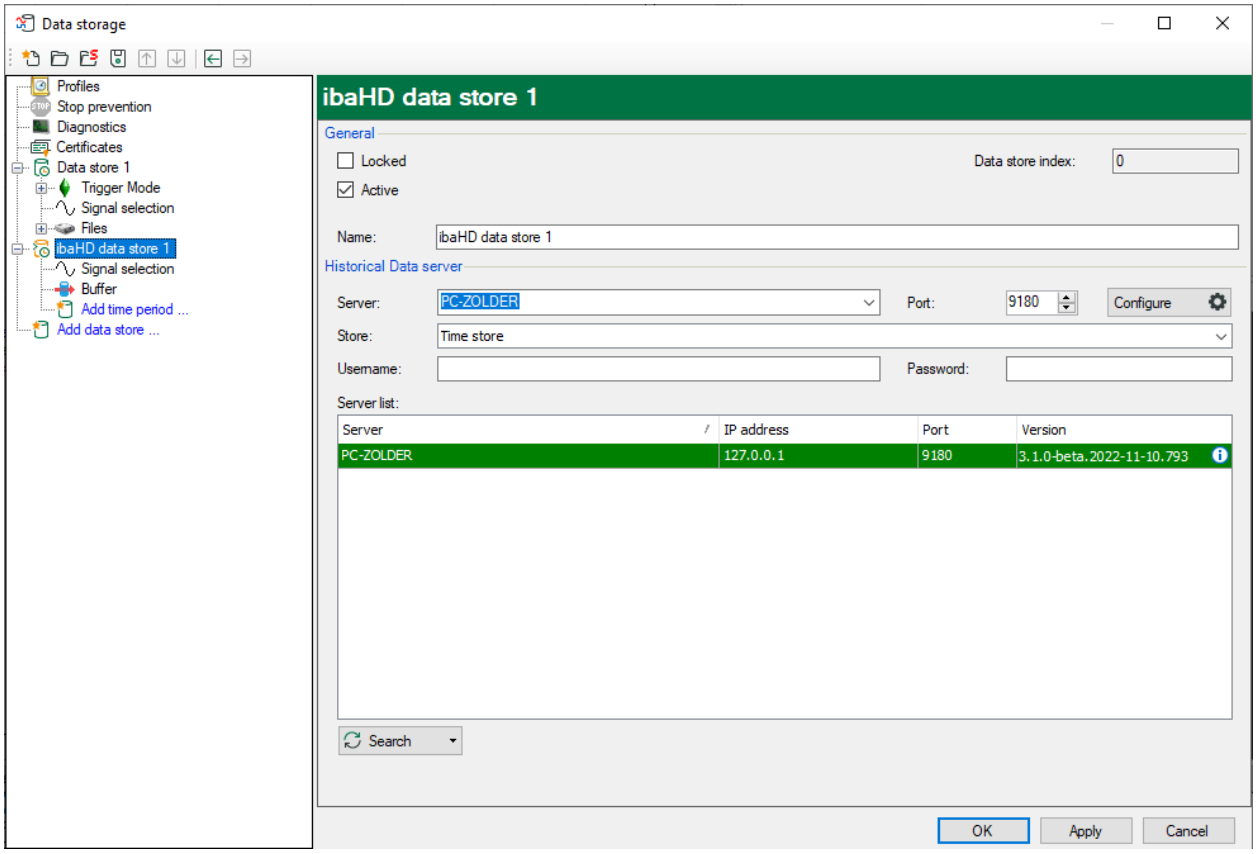
2 Support for ibaHD time periods

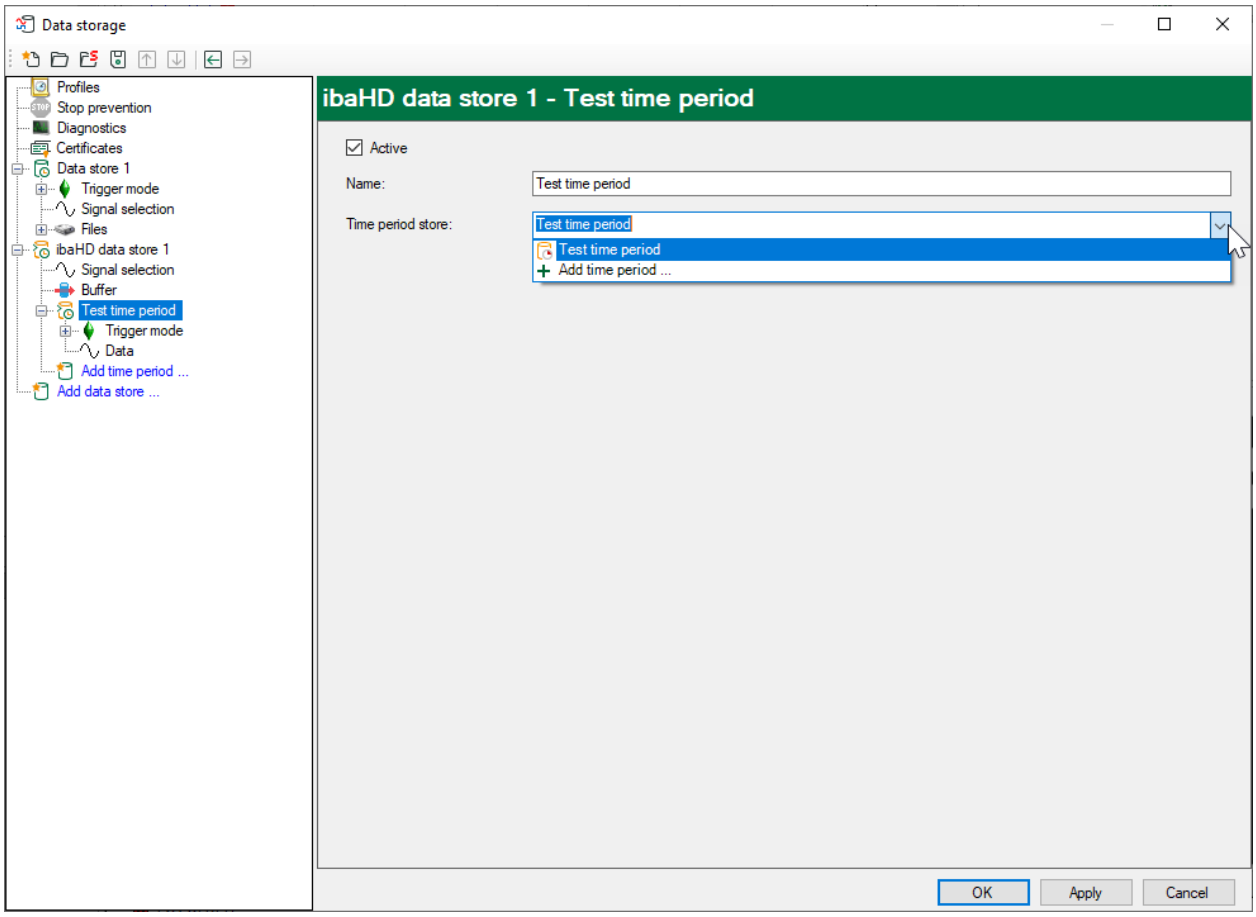
Time periods mark a range of time inside an ibaHD time store. A time period has a name. It can also have info fields. These info fields are used to store extra information about the time range like e.g., KPIs calculated over the time range. You can think of a time period as a dat file but with the signal data present in the ibaHD time store. A time period is stored in a database table. The info fields correspond to columns inside the database table.

For using ibaHD time periods an ibaHD-Server v3.1 or later is required.

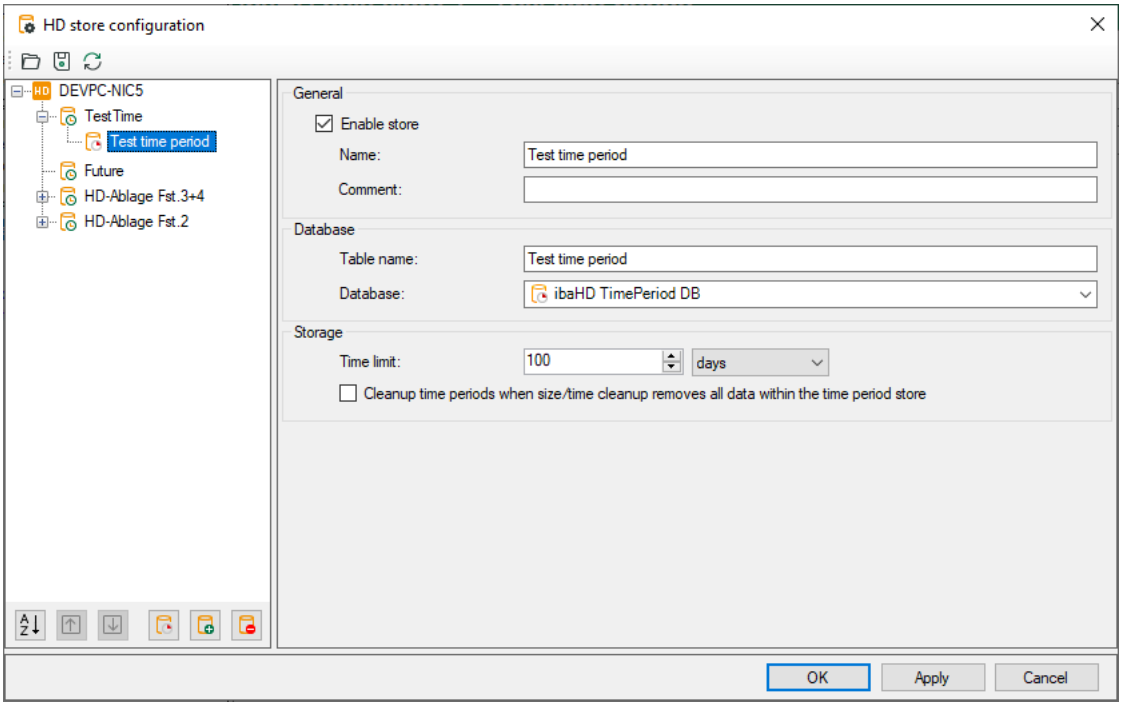
2.1 Data store

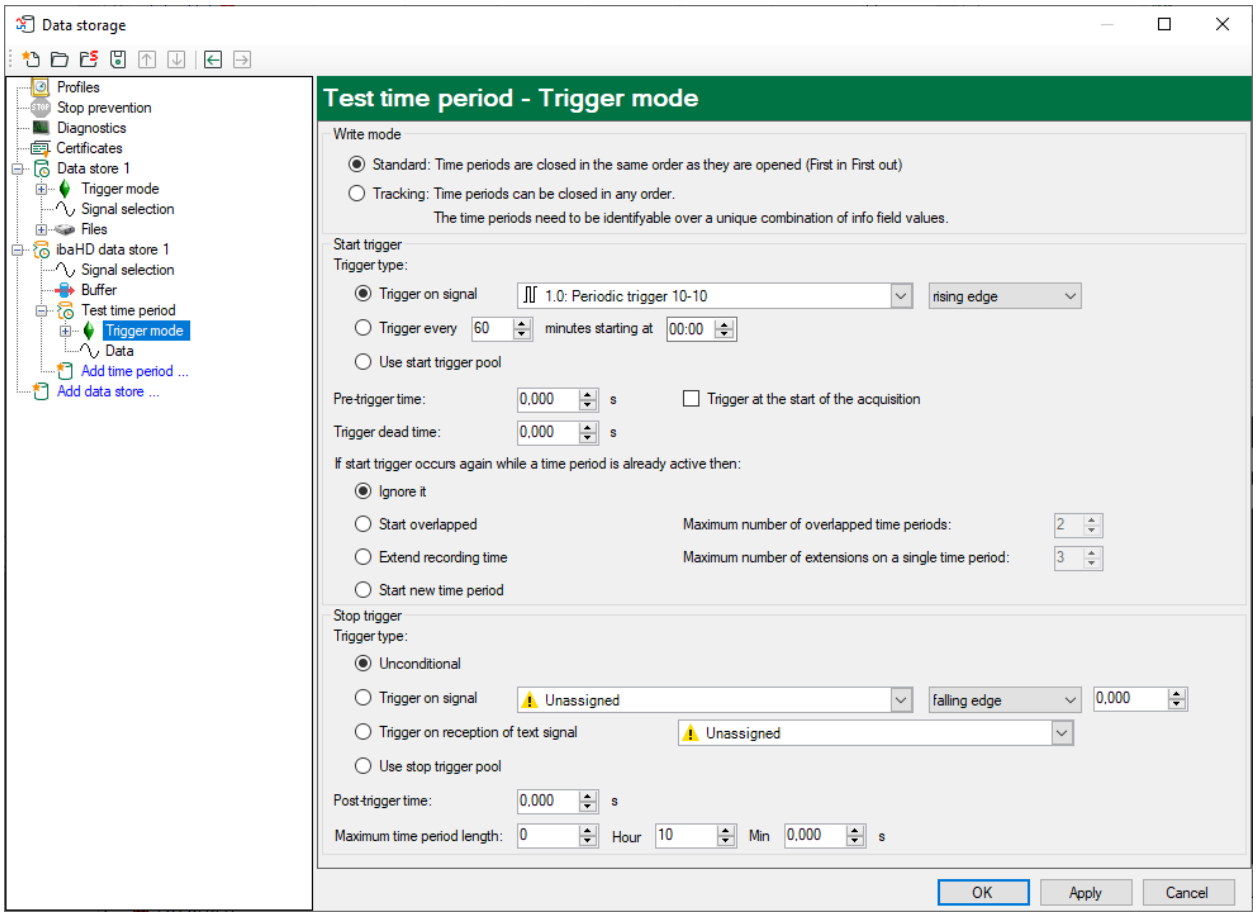
IbaPDA can be used to create time periods and to fill their info fields. This is configured in the data storage manager. Under an ibaHD time store you can add one or more time periods by clicking the *"Add time period …"* node.





On the time period node you have to select the time period store that you want to write to. You can also create a new time period store by selecting the *"Add time period …"* item. This will open the ibaHD store configuration dialog.





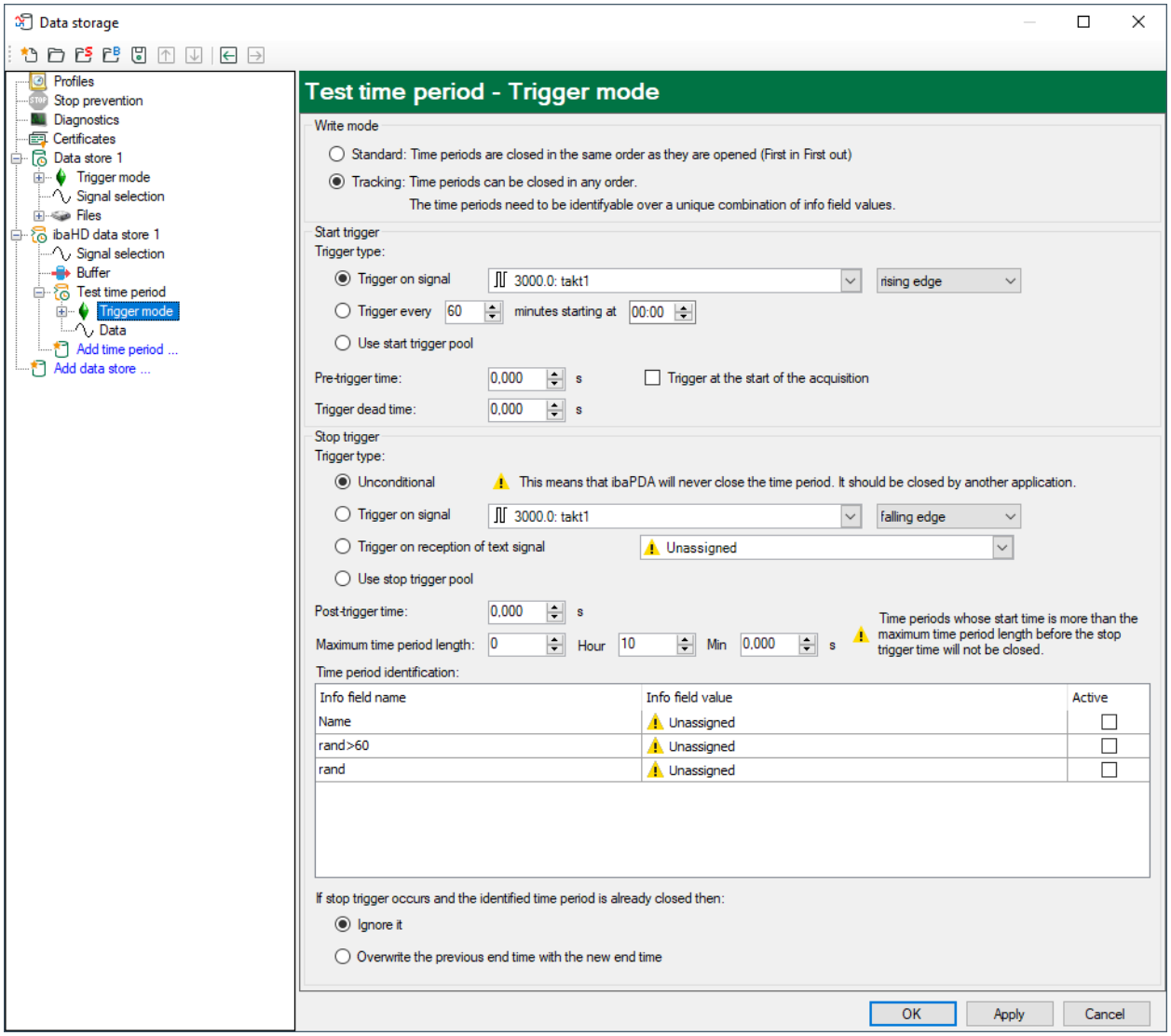
On the *"Trigger mode"* node you can configure the triggers to generate the time period. There are 2 modes possible: standard and tracking mode.

In standard mode you basically have the same options as for a standard timebased data store. You must define a start trigger. This can be a trigger on a signal, a period trigger or a trigger coming from the start trigger pool. You can define a pre-trigger time and a trigger dead time. The trigger dead time determines how long new start triggers will be ignored after the previous start trigger. You can also configure what ibaPDA should do when a new start trigger occurs while the current time period has not been stopped:

- Ignore it: This ignores the new start trigger and just keeps the current time period.
- Start overlapped: This starts an extra time period and does not close the current one. You can configure how many time periods can be open at the same time.
- Extend recording time: This will increase the maximum length of the current time period. It is increased with the configured maximum length in the stop trigger part. You can configure how many times the maximum length can be increased.
- Start new time period: This will close the current time period and start a new one.

In the stop trigger part, you must configure a maximum length of a time period. If there did not occur a stop trigger then the time period will be automatically closed when it reaches the maximum length. There are the same 4 options for the stop trigger as for the standard timebased data stores. You can trigger on a signal, on a text signal and on the stop trigger pool. You can also not define a stop trigger by choosing *"Unconditional"*. In this case the time period will always have the maximum length. A post-trigger time can also be configured.

In standard mode a stop trigger will always work on the oldest open time period. When the acquisition is stopped or a new data storage configuration is applied all open time periods will be automatically closed.

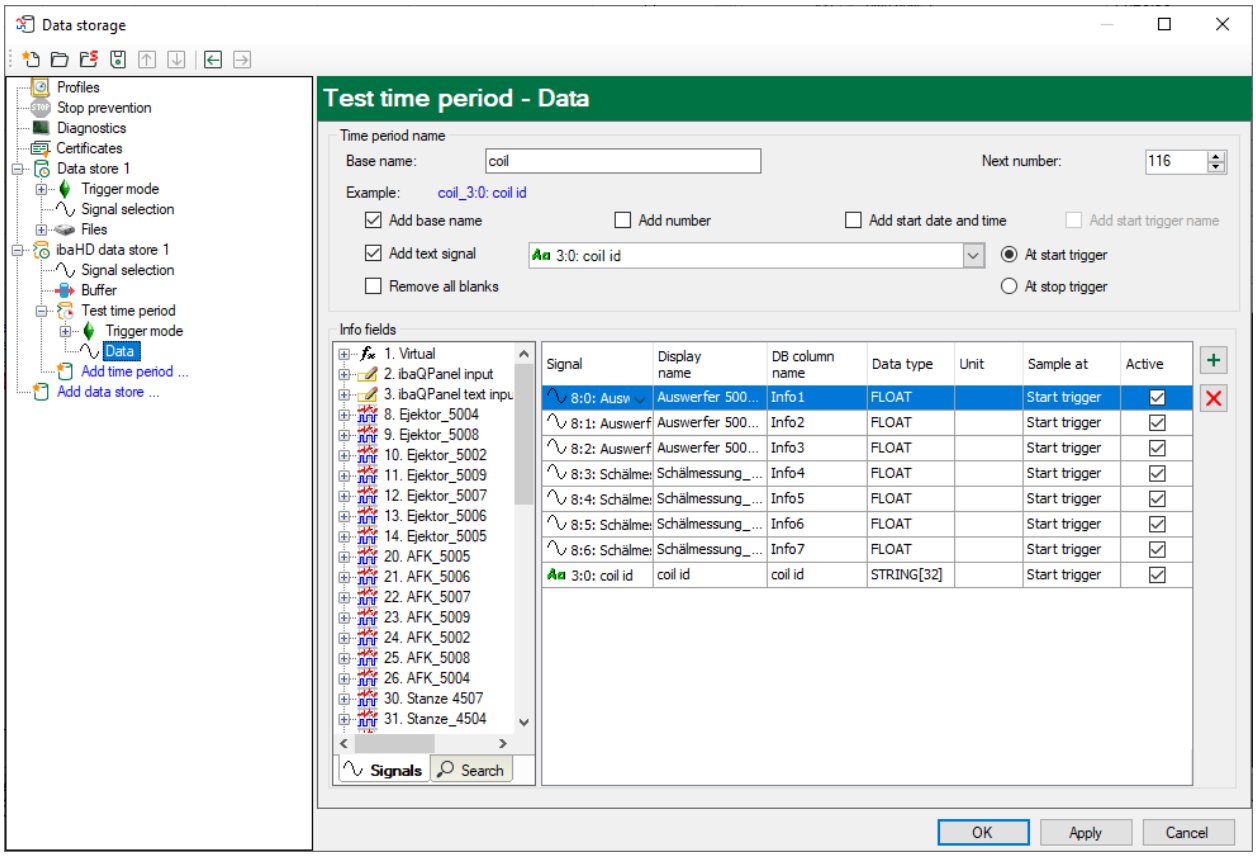


In tracking mode this is different. A stop trigger will not work on the oldest time period but on the time period that matches with the current id. The id of a time period is defined by one or more info fields. You can configure which info fields define the id in the time period identification table. By assigning a signal and checking active you add the info field to the id. The signals will be sampled at the stop trigger and ibaPDA will then query the time period store for time periods whose id info fields match the signal values. The query will only consider time periods whose start time is less than the maximum time period length before the stop trigger time. If ibaPDA finds multiple matching time periods then none of them will be closed. A warning will be generated in the event log. You can configure what ibaPDA does when a matching time period is already closed. You can choose between:

- Ignore it: the end time of the matching time period will not be changed.
- Overwrite the previous end time with the new end time: the end time of the matching time period will be changed.

In tracking mode no open time periods will be automatically closed when the acquisition is stopped or a new data storage configuration is applied. Time periods will only be closed by a

stop trigger. This also means that an “Unconditional” stop trigger means that another application should close the time period.



On the data node you can configure how time periods are named and which info fields should be written.

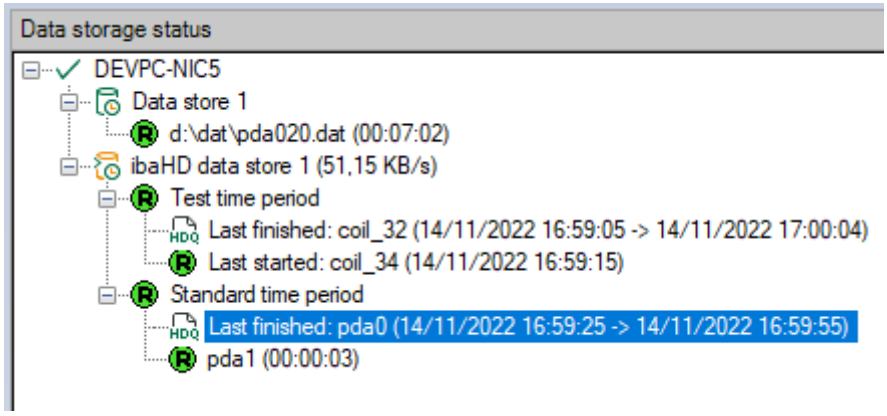
The time period name has the same options the file name in a standard time-based data store. You can configure a base name and then add a number, start date and time, start trigger name and a text signal value.

In the info fields table, you can add info fields by dragging signals from the signal tree to the table. You can also use the plus button to add a line. For each info field you must define:

- Signal: The signal whose value will be written to the info field.
- Display name: The display name of the info field. There are no limitations to the display name. By default, the signal name is used.
- DB column name: The name of the database column that will contain this info field. There might be some restrictions on the length and on which characters can be used in the column name depending on the database type used by the time period. By default, a normalized version of the signal name is used.
- Data type: The data type used for the database column. By default, the data type of the signal is used.
- Unit: This is the unit of the info field. It is always equal to the unit of the signal. It is read-only.
- Sample at: You can decide when the value will be taken, at the start or at the stop trigger.

- Active: If an info field is active then ibaPDA will write a value to it. If it is not active then ibaPDA will not touch the database column corresponding with the info field.

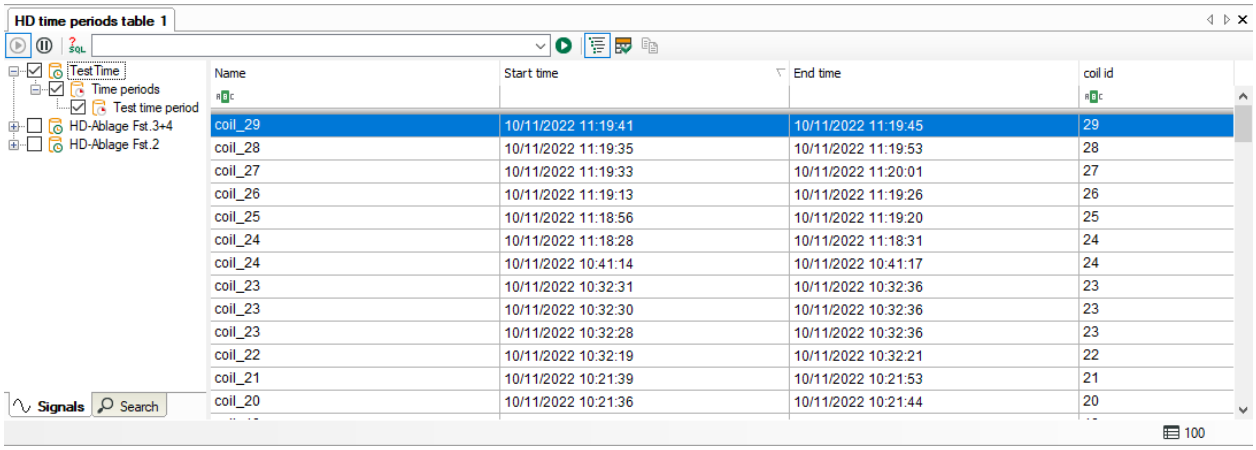
The table shows the complete list of info fields that are available in the database. IbaPDA does not have to write them all.



In the data storage status window you can see the current state of the time periods. In the screenshot the “Test time period” store uses tracking mode. It shows both the last finished time period and the last start time period. The “Standard time period” store uses standard mode. It is now recording time period “pda1” and it shows the last finished one “pda0”. For each time period it shows the name and start time. If the end time is available then it also shows the end time. You can double-click a time period node to open the corresponding time range of the ibaHD timebased store in ibaAnalyzer.

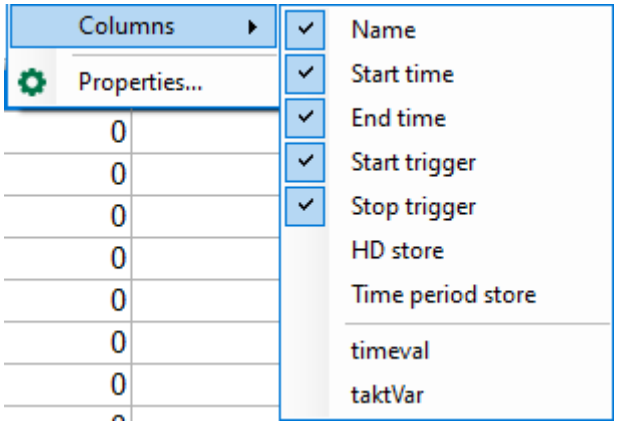
2.2 Time period table

The time period table allows you to show the different time periods from multiple time period stores within one ibaHD server. It is very similar to the event table. A time period table can be added via the button .

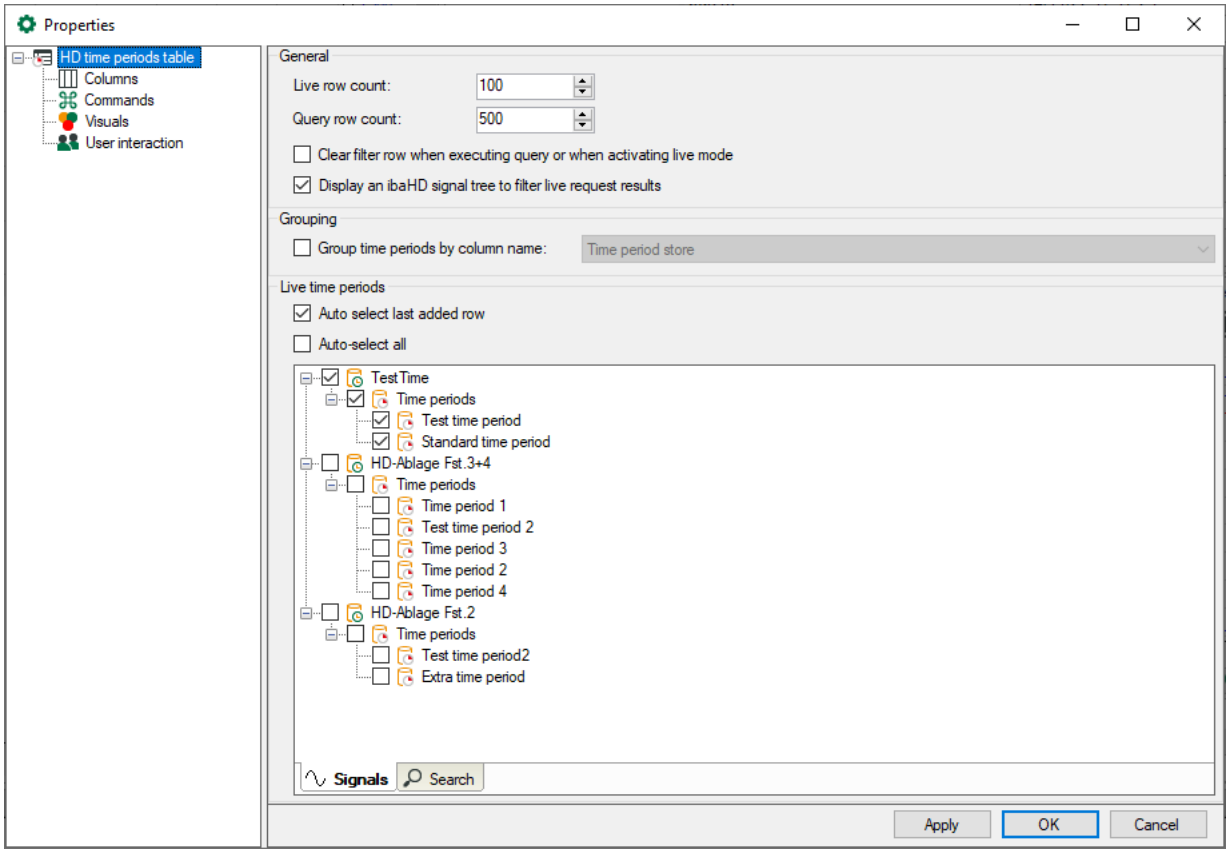


You can select which time period stores to show in three different ways:

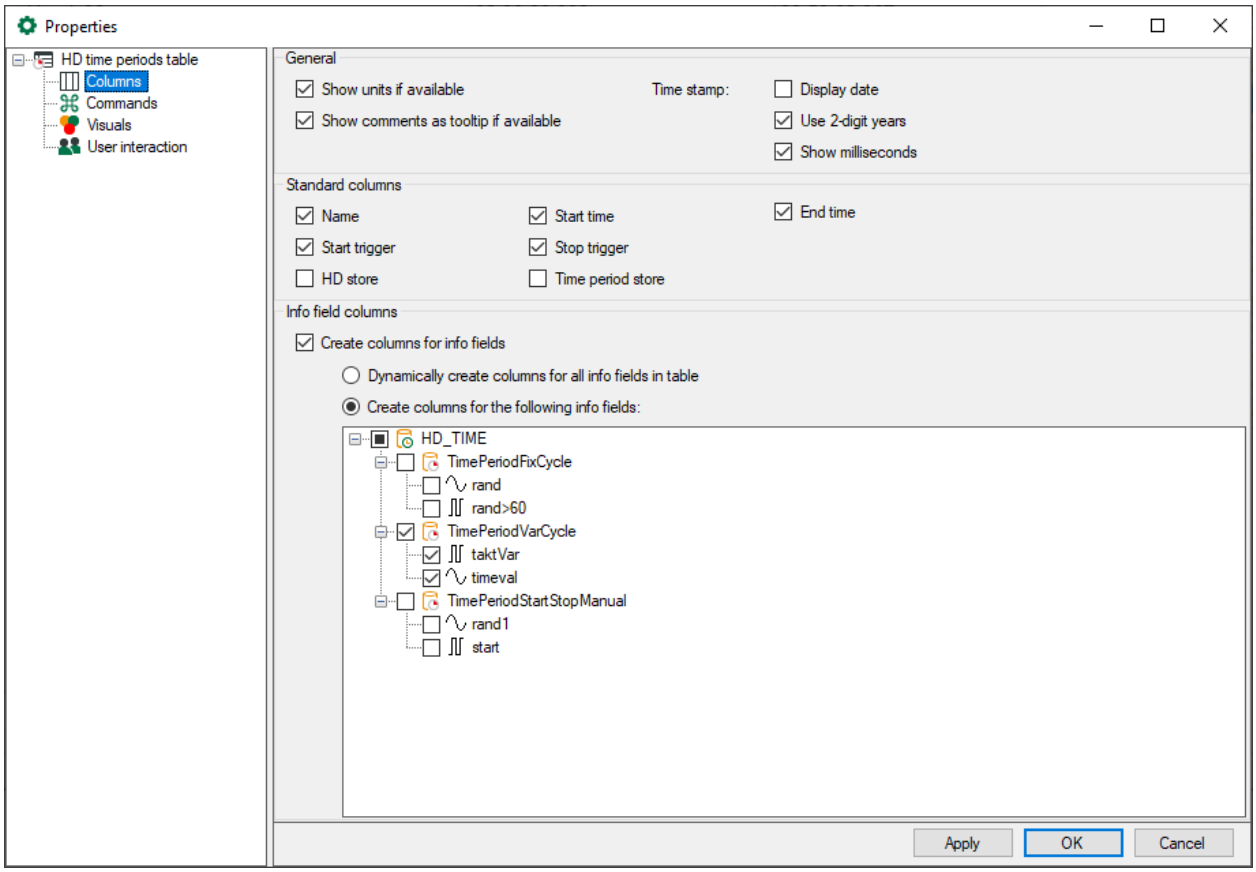
1. Use the toolbar button to show the tree. In the tree you can check the time period stores you want to see.
2. Drag a time period store from the main ibaHD signal tree and drop it on the table.
3. Open the properties of the time period table via right-click and select the time period stores there.



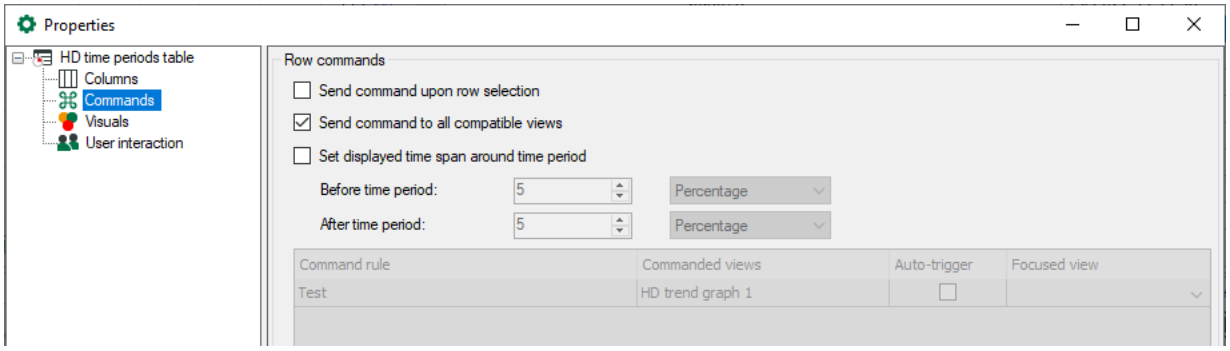
You can right-click the table to show the context menu. In the menu you can select the visible columns. The list contains 2 parts: the standard columns at the top and the info field columns at the bottom. The list of info field columns can be configured in the properties. You can open the properties via the context menu.



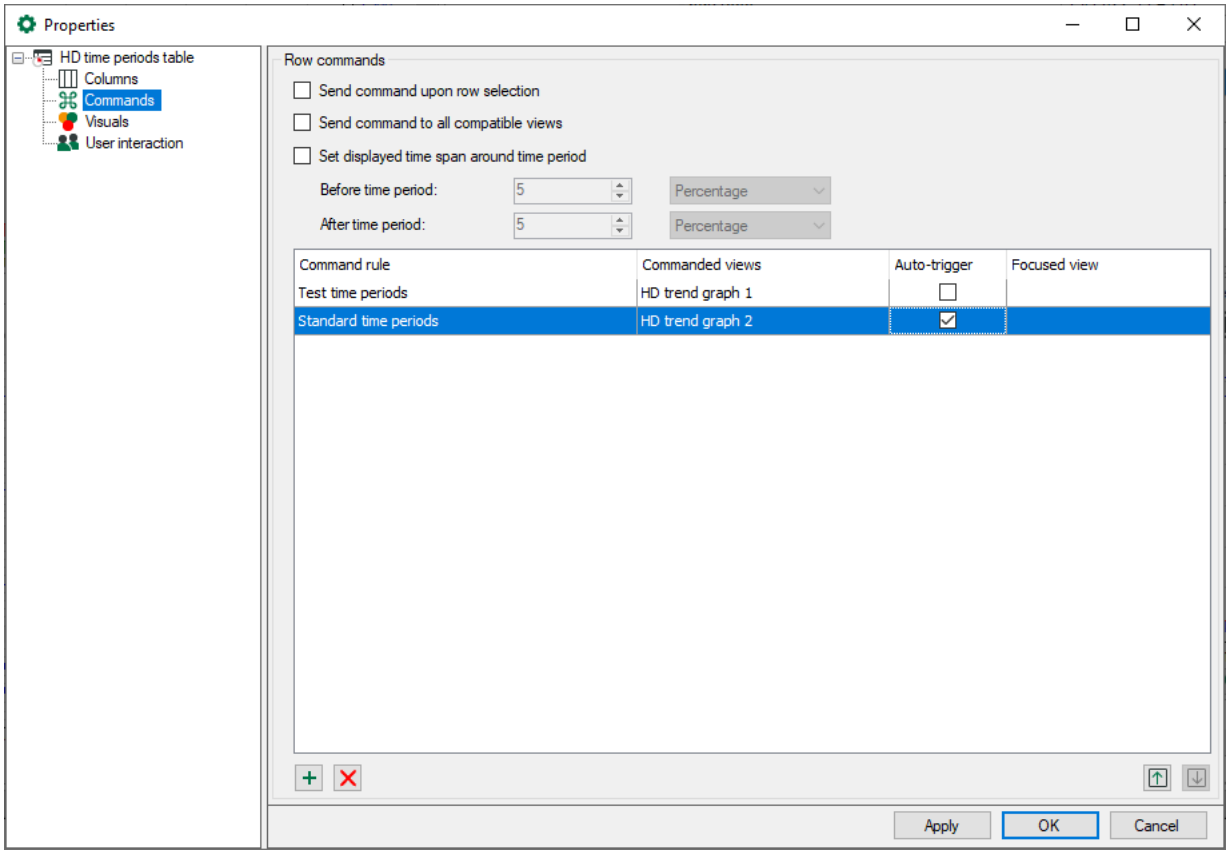
On the first page of the properties you can configure how many rows to show in live mode and in query mode. You can choose to show the ibaHD time periods tree next to the table or not. You can select one column to group the rows by. In the “Live time periods” part you can select which time period stores to show time periods from. You can also configure if the last added row should be automatically selected.



On the columns page you can configure which columns are shown and how data is formatted inside the columns. The standard columns section contains all columns that each time period has. By clicking the checkboxes you can show or hide them. In the info field columns section you can configure which info field columns will be available. You can explicitly select the columns or you can let the table automatically add columns for all info fields. The selected columns can then be shown or hidden via the table context menu. If info fields of different time period stores have the same database column name then they will be merged into a single column in the time period table.

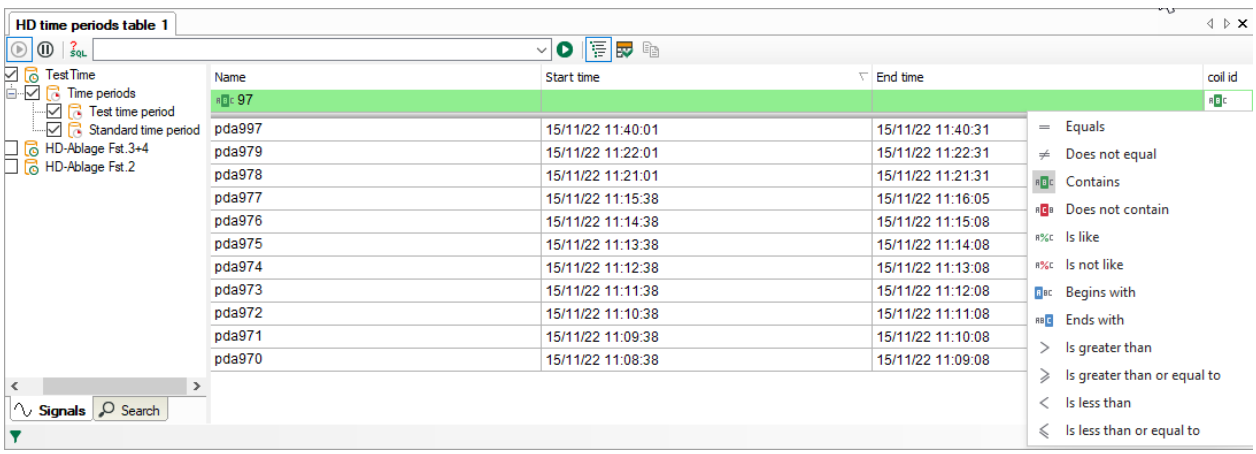


On the commands page you can configure if the time period table should control other views (e.g. ibaHD trend graphs). By default the “Send command to all compatible views” option is enabled. It means that when you double-click a row in the time period table that all ibaHD trend graphs will display the time range that corresponds with the time period of the selected row. If you enable “Send command upon row selection” then you just have to select a row to execute the command instead of double-clicking the row. The “Set displayed time span around time period” option allows you to add some extra time before and/or after the time period when it is displayed in an ibaHD trend graph. When you uncheck the “Send command to all compatible views” then you can configure a custom list of commands.



A command has the following properties:

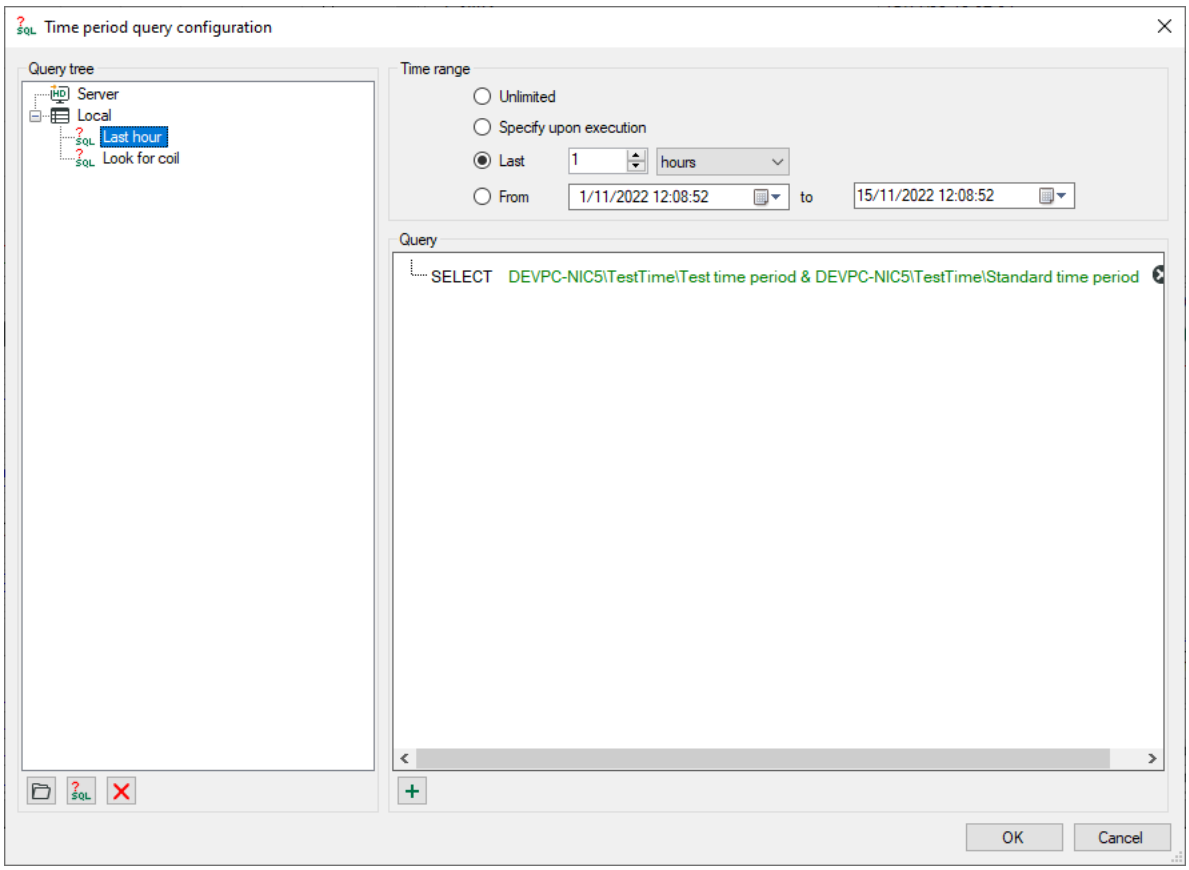
- Rule: The rule determines to which time period the command applies. You can select time period stores for it or you can set certain conditions on columns.
- Commanded views: Here you select the views that will jump to the selected time period. In the example each rule controls a different trend graph.
- Auto-trigger: If auto-trigger is enabled then the command will fire when a row is added to the table that matches the rule. If auto-trigger is not enabled then you have to select or double-click the row depending on the “Send command upon row selection” option in order to fire the command.
- Focused view: Here you can select if you want to focus another view after the command fires.



The top row in the time period table is the so-called filter row. It allows you to filter the displayed rows. You can enter a value and select an operator by clicking the icon on the left. The filter can be cleared again by removing the value.

In live mode new rows are added when new time periods are created. Rows are updated when the end time or some other info fields are updated. When you click the pause button the table is no longer updated. In pause mode you can select multiple rows and copy them to the clipboard via CTRL+C or the copy button on the toolbar.

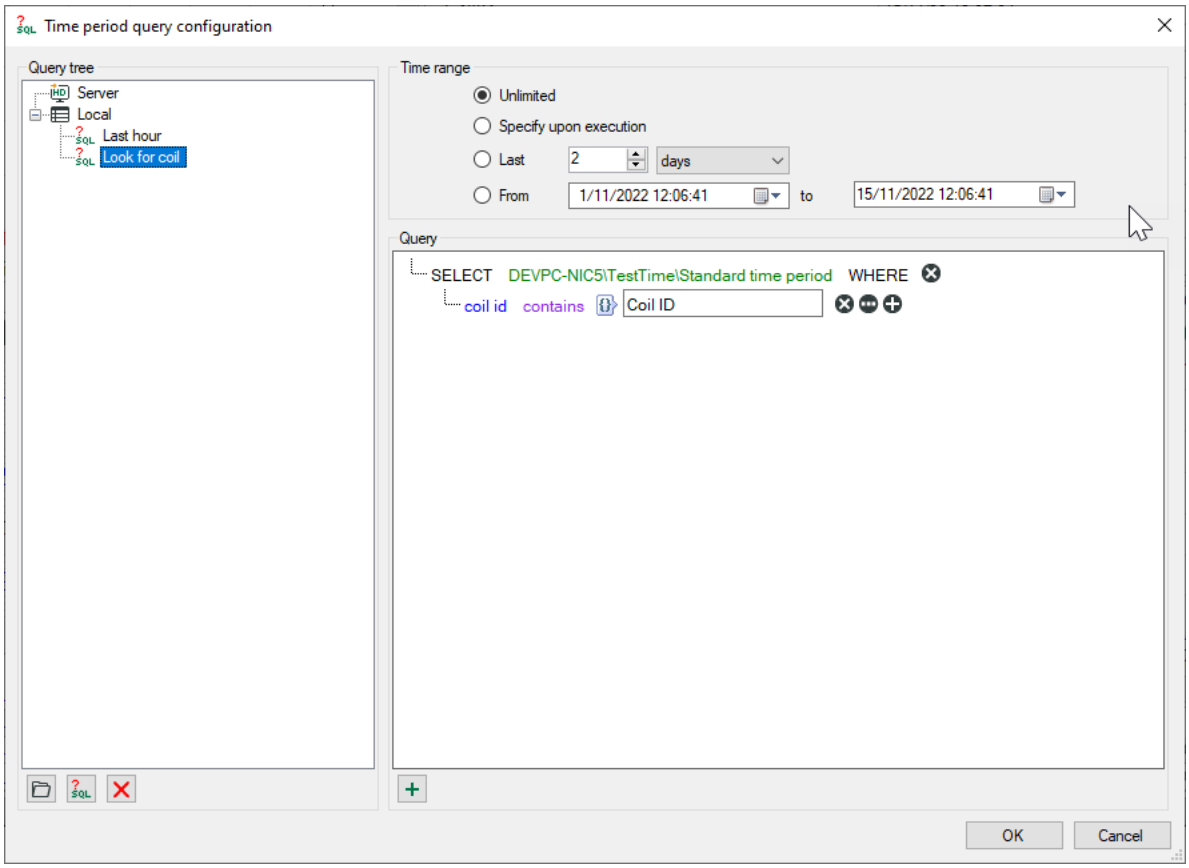
Next to the live mode there is also a query mode. Use the button to edit the queries.



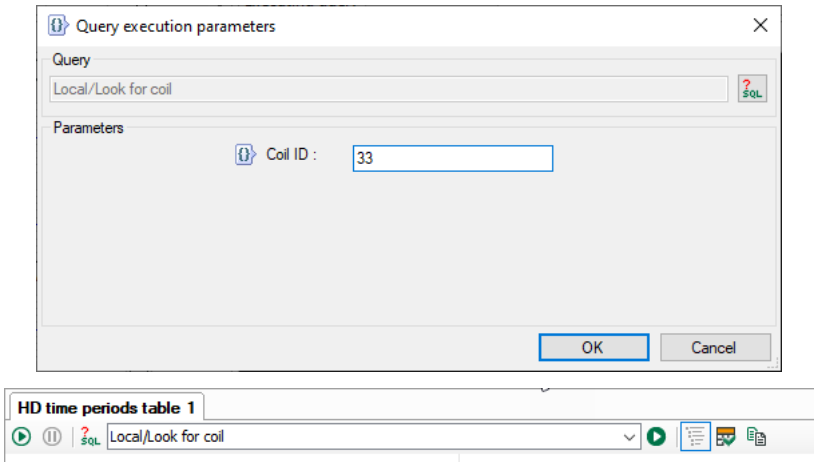
You can add queries in the tree on the left. The queries can be local or server queries. Server queries are stored on the ibaHD server. So all connected clients can use these same server queries. Local queries are stored in the layout. Other connected clients won't see these.

A query consists of a time range and the query statement. There are 4 options for the time range:

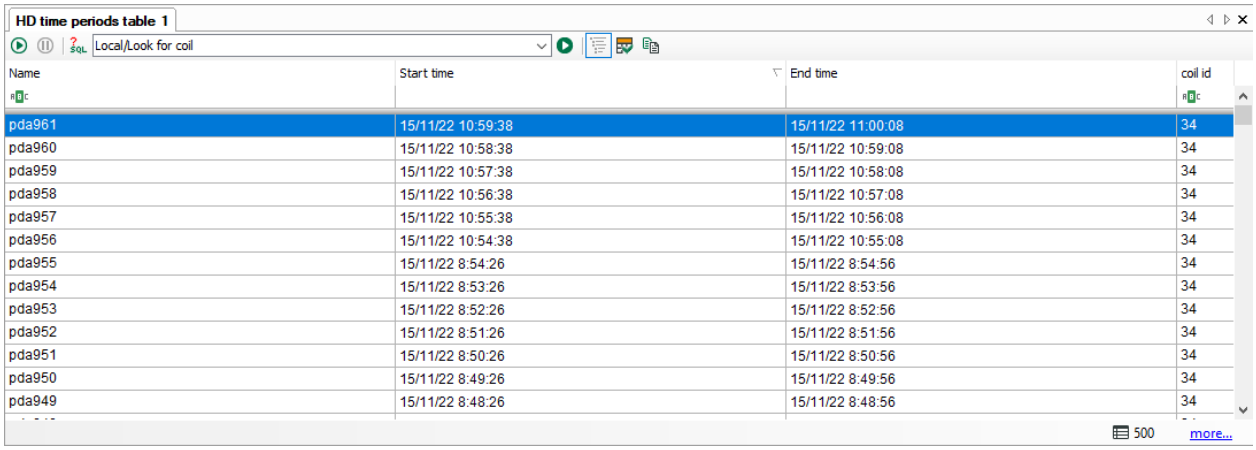
- Unlimited: The complete available time range of the time period store(s) is queried.
- Specify upon execution: A dialog will appear when the user executes the query. In the dialog the user can specify the time range he wants to use.
- Last: This can be used to look for time periods in the last day, hour, …
- From to: This is a fixed time range.



The query statement begins with a SELECT followed by the list of time period stores you want to query. This can then be followed by WHERE conditions. In the screenshot a WHERE condition is added with a parameter. If a parameter is defined then the user will be able to enter the value for the parameter when he executes the query like in the screenshot below.



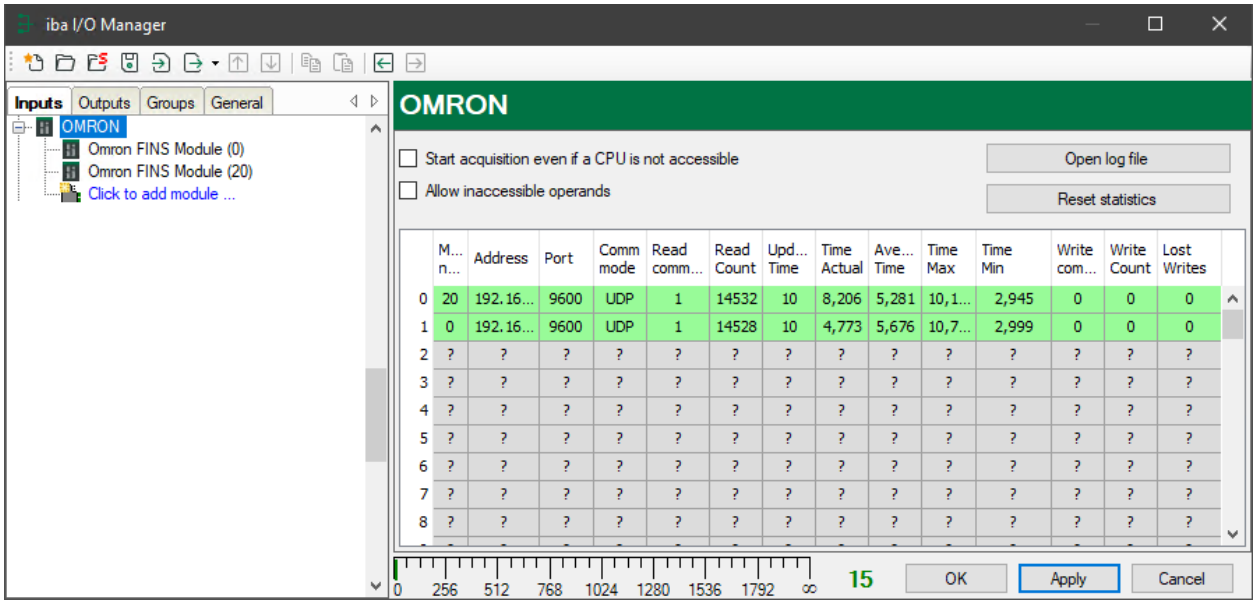
A query is executed by selecting the query in the toolbar and then pushing the button. After executing a query the table is in query mode.



The number of rows the query returned is shown in the bottom right. If the number of rows was more than the configured maximum then a *"more…"* link will be shown. You can click it to show extra rows. You can return to live mode by clicking the ▶ button.

3 OMRON-Xplorer

The OMRON-Xplorer interface in ibaPDA is used to measure data from OMRON controls using the FINS protocol. It is an Xplorer interface which means that the data is cyclically read by ibaPDA instead of being sent by the PLC.



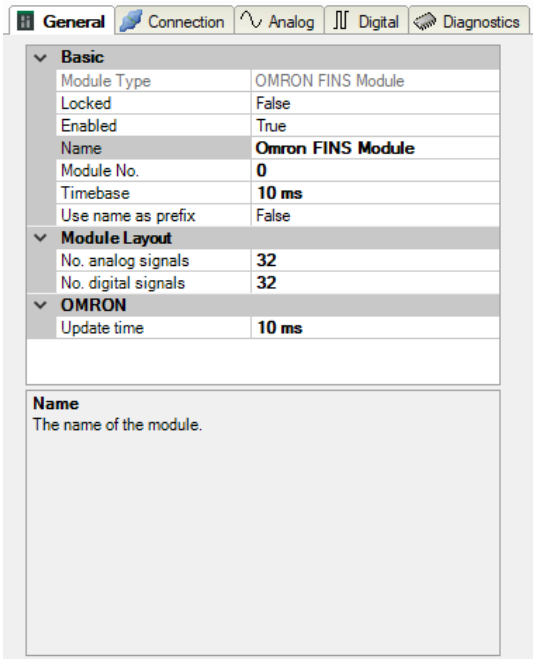
The OMRON-Xplorer interface shows a table of the available connections. Per OMRON-Xplorer license you get 16 connections. A maximum of 240 connections is allowed. This means that maximum 15 licenses can be used. Each connection corresponds to a row in the table. The row is green when the connection is ok and data is being read. The row is orange when the connection is ok but the data is coming slower than the configured update time. The row is red when the connection could not be established. The row is grey when there is no connection configured. The response time is the time it takes to read and write the data for a connection. The table shows the actual, average, minimum and maximum values of the response time. The update time is the time between 2 read operations. The number of read and write commands needed for the configured signals are shown in the Read commands resp. Write commands column. You can use the “Reset counters” button to clear the counters for all connections. Clicking “Open log file” opens the most recent log file related to OMRON-Xplorer connections.

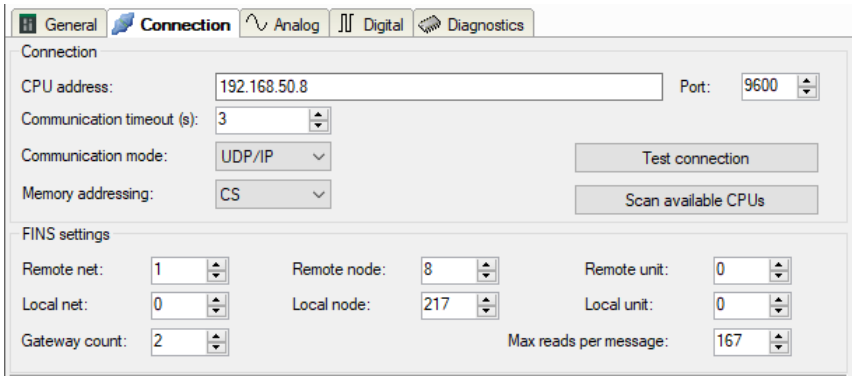
On the interface you can also decide how to handle some error conditions:

- When an OMRON Control is not accessible during the start of the acquisition then you can choose if the acquisition starts without this control or if the acquisition is not started. When the acquisition is started without the OMRON Control then ibaPDA will periodically (every 10s) try to connect to the control during the acquisition. As long as the control is disconnected the values will remain at zero.

- When ibaPDA tries to access an operand that is not available when validating the I/O configuration, the OMRON Control will return an error. If the option “Allow inaccessible operands” is enabled then ibaPDA will ignore this signal and start the acquisition without this signal. If the option is not enabled then the acquisition will not start.

The General tab of an OMRON-Xplorer module looks as follows:





In the **Connection** tab, all parameters to establish a proper connection to the OMRON Control have to be configured:

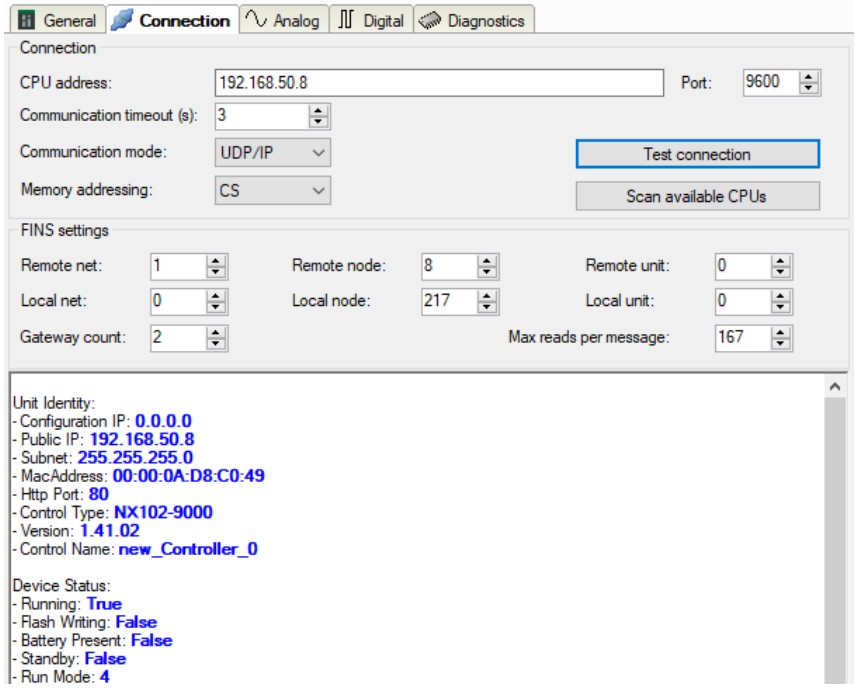
- **CPU address**: The IP address of the OMRON Control at which the network interface to be used is located at.
- **Port**: The standard port for the FINS protocol is 9600. Change this only if you configured something else in the Sysmac Studio
- **Communication Timeout**: the amount of time after which a connection attempt will be aborted.
- **Communication mode**: Use either TCP/IP or UDP/IP to connect to your OMRON Control. Depends on the OMRON Control used and the network setup.
- **Memory addressing**: The memory addressing mode (CS or CV) depends on the OMRON Control used. Usually, only control types of starting with “CV” use the old CV scheme, while “CS” or “NX” controls use the CS scheme.

In the FINS, you can adjust some additional parameters that are needed for the FINS protocol. These parameters have to be set according to the settings selected in the Sysmac Studio.

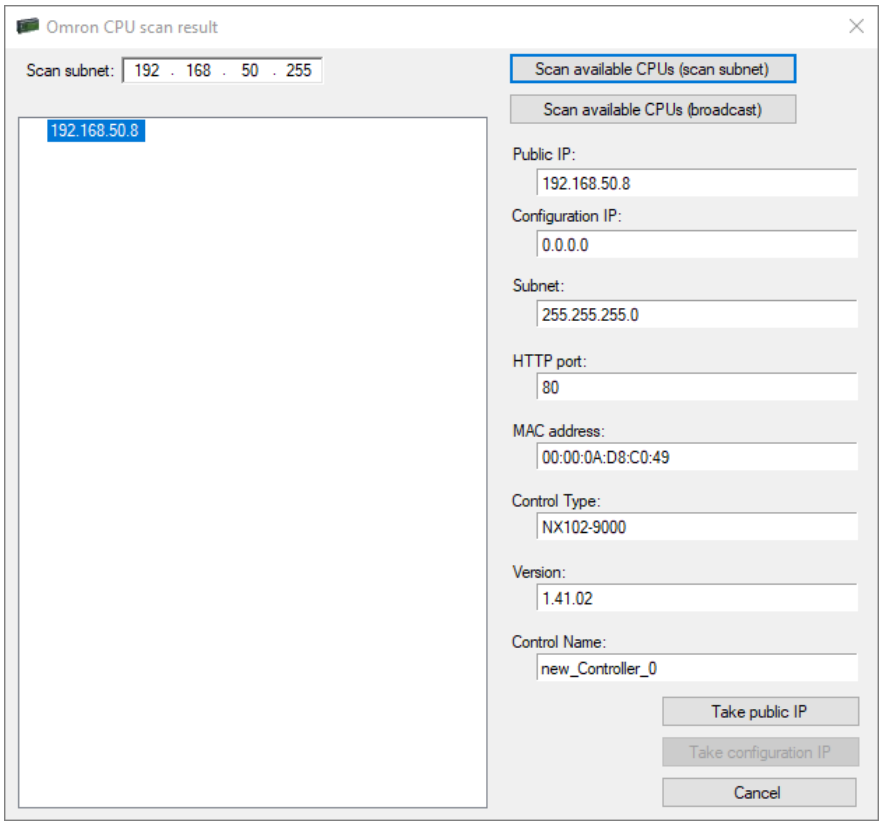
- **Remote net**: usually set to 1.

- **Remote node**: standard is the last value of the IP address, can be changed in Sysmac studio. In TCP mode, this value is automatically set during communication initialization.
- **Remote unit**: usually set to 0.
- **Local net**: usually set to 0.
- **Local node**: can be selected freely, except when some dedicated routing is configured in Sysmac Studio. In TCP mode, this value is automatically set during communication initialization.
- **Local unit**: usually set to 0.
- **Gateway count**: Standard values are 2 or 7, depending on the network layout and the values configured in Sysmac studio.
- **Max reads per message**: Standard value is 167, except when DevieNet connections are configured, where only 89 elements per read are allowed.

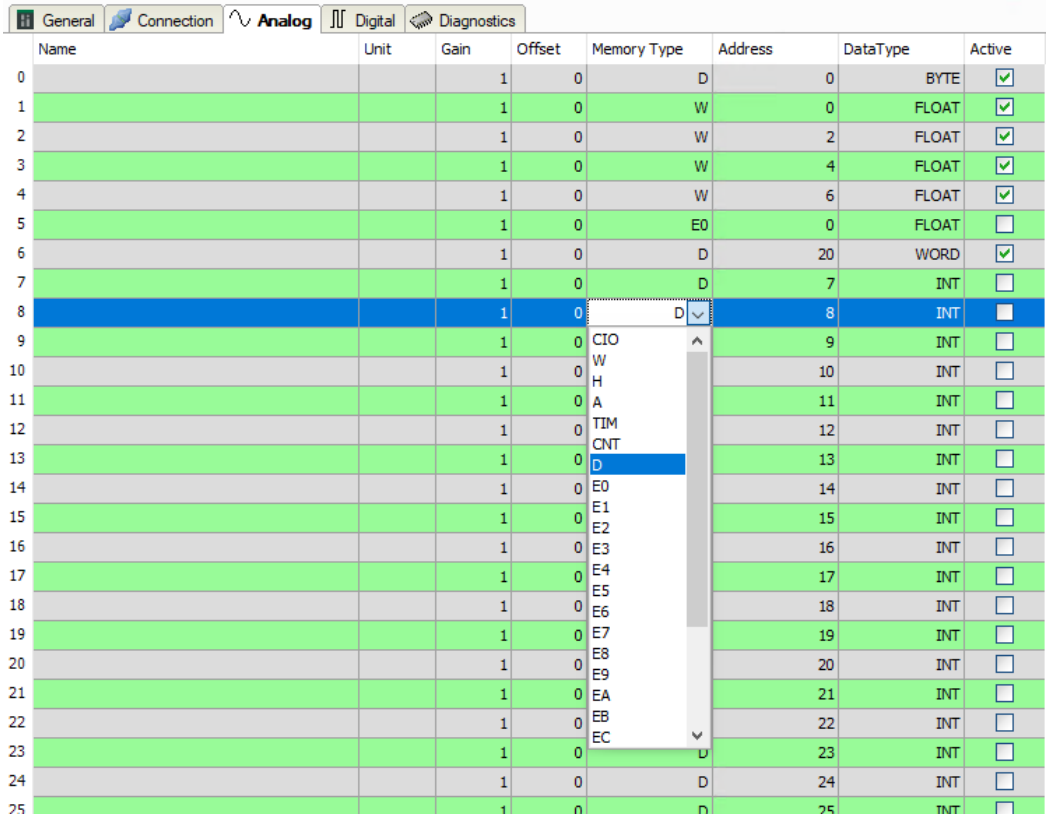
When clicking the **Test Connection** button, ibaPDA will try to establish a connection to the OMRON Control and show some diagnostic data about the target control in the output window.



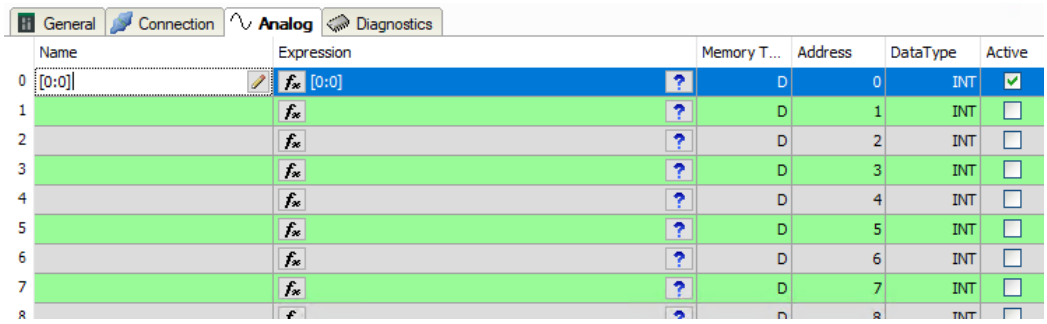
When clicking on **Scan available CPUs**, a dialog is shown to perform a network scan.



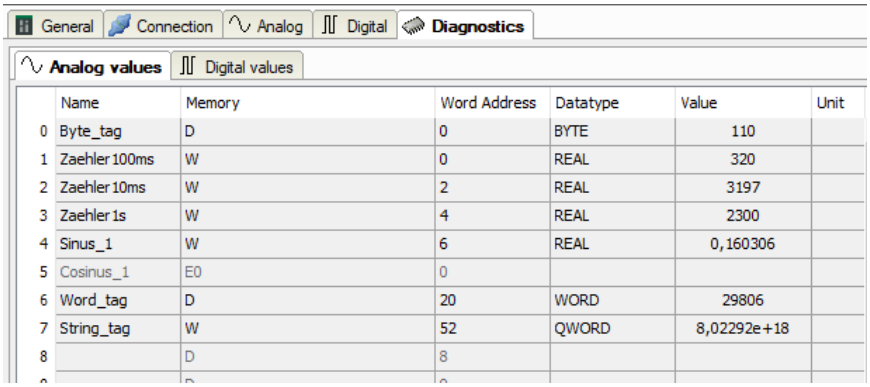
This might be done either per broadcast, or per subnet scan. Broadcast is usually limited to the subnet the PDA PC is currently loated in, it won't cross subnet boundaries. If another subnet is neede, a dedicated subnet scan can be performed. Any control found will be shown in the lefthand tree. The data transmitted per control is shown on the right side. Here the IP of the control can be taken to be used for the configured module.



The above figure shows an example of the Analog tab of an OMRON module. The memory type, address and data type have to be set according to the mapping configured in Sysmac Studio. The address is a word address, as it is usual for OMRON controls, and one address cannot be shared by two smaller types. That means that for data transfer as well as the mapping configuration in Sysmac Studio, SINT, BYTE, WORD, INT use one word, DWORD, FLOAT and DINT use two words, DOUBLE and LINT use four words. ibaPDA handles the Sysmac datatypes USINT, UINT and UDINT as BYTE, WORD and DWORD. ULINT is currently not supported. For digital values, the relevant bit number has to be configured, too.

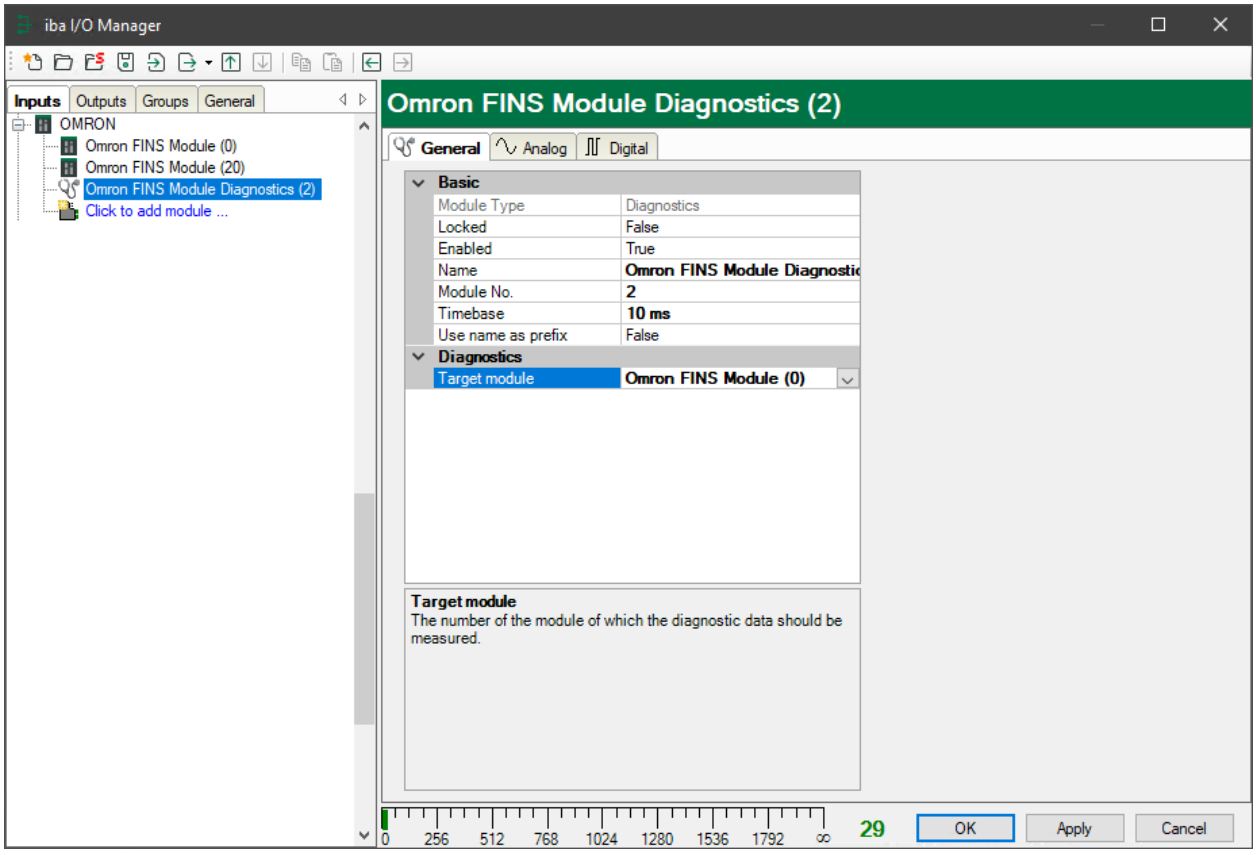


Outputs have a similar configuration, with the additional Expression to set. Digital values are not supported for outputs.

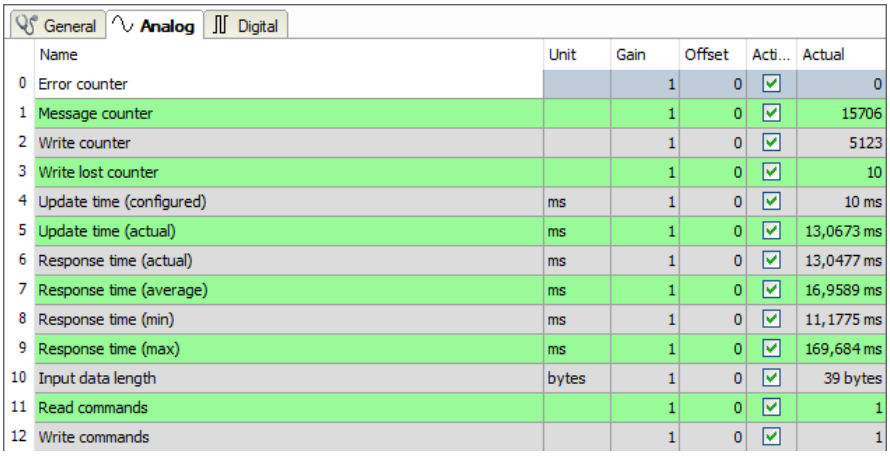


	Name	Memory	Word Address	Datatype	Value	Unit
0	Byte_tag	D	0	BYTE	110	
1	Zaehler100ms	W	0	REAL	320	
2	Zaehler10ms	W	2	REAL	3197	
3	Zaehler1s	W	4	REAL	2300	
4	Sinus_1	W	6	REAL	0,160306	
5	Cosinus_1	E0	0			
6	Word_tag	D	20	WORD	29806	
7	String_tag	W	52	QWORD	8,02292e+18	
8		D	8			

The current values of the requested topics can be monitored in the **Diagnostics** tab. There a further distinction is made between **Analog values** and **Digital values**.



A diagnostic module can be added to record additional information about the Control connection.

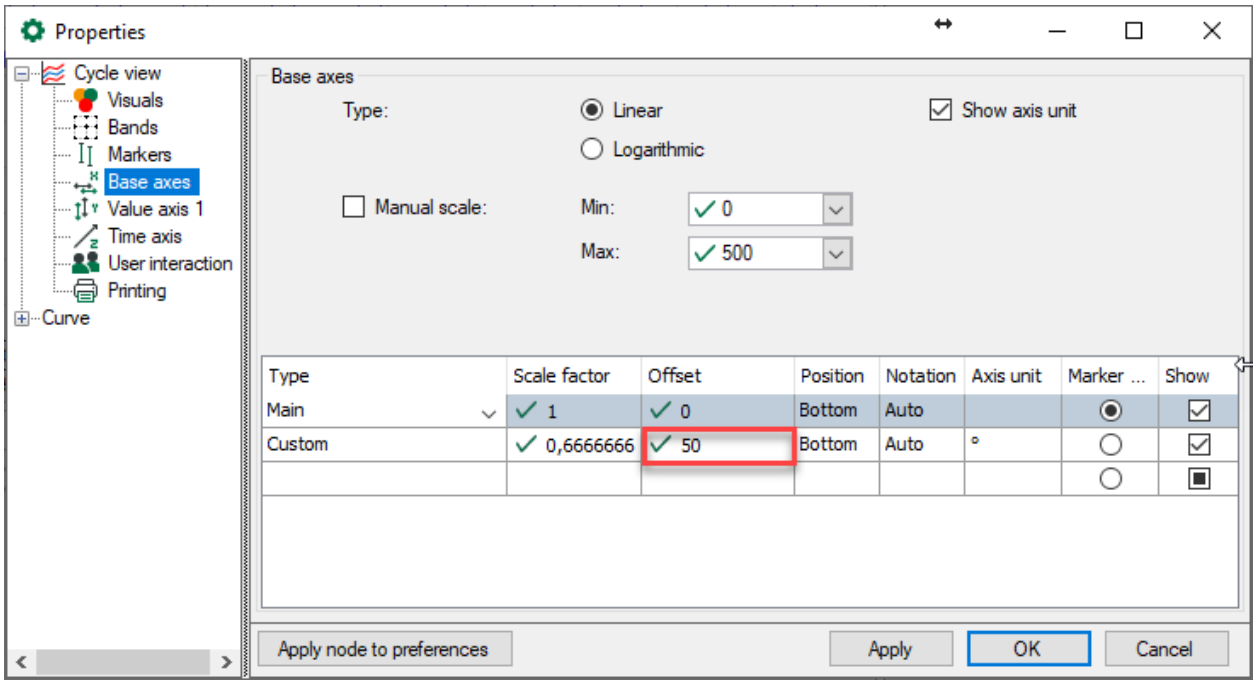


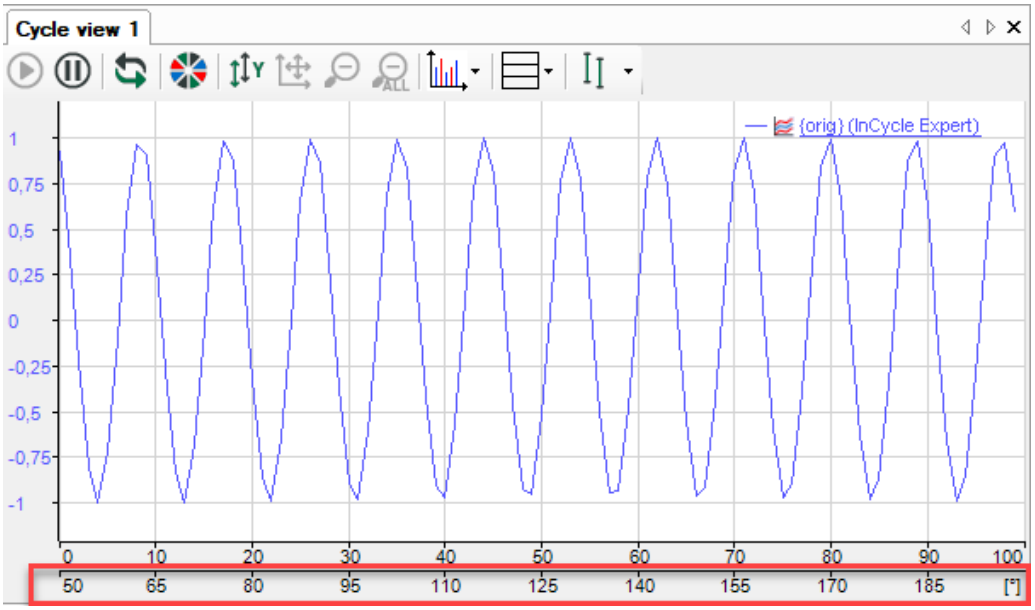
	Name	Unit	Gain	Offset	Acti...	Actual
0	Error counter		1	0	☑	0
1	Message counter		1	0	☑	15706
2	Write counter		1	0	☑	5123
3	Write lost counter		1	0	☑	10
4	Update time (configured)	ms	1	0	☑	10 ms
5	Update time (actual)	ms	1	0	☑	13,0673 ms
6	Response time (actual)	ms	1	0	☑	13,0477 ms
7	Response time (average)	ms	1	0	☑	16,9589 ms
8	Response time (min)	ms	1	0	☑	11,1775 ms
9	Response time (max)	ms	1	0	☑	169,684 ms
10	Input data length	bytes	1	0	☑	39 bytes
11	Read commands		1	0	☑	1
12	Write commands		1	0	☑	1

4 FFT/Cycle view

4.1 Offset for custom base axis

A custom base axis can be configured with an offset. For example, one uses an InCycle Expert module with 100 samples per cycle, with start trigger at 50° and stop trigger at 200°. To display a range of 150° we use a scale factor of 0.66666. Because we want the scale to start at 50°, we configure an offset of 50. The resulting range of the custom base axis is [50°..200°[.

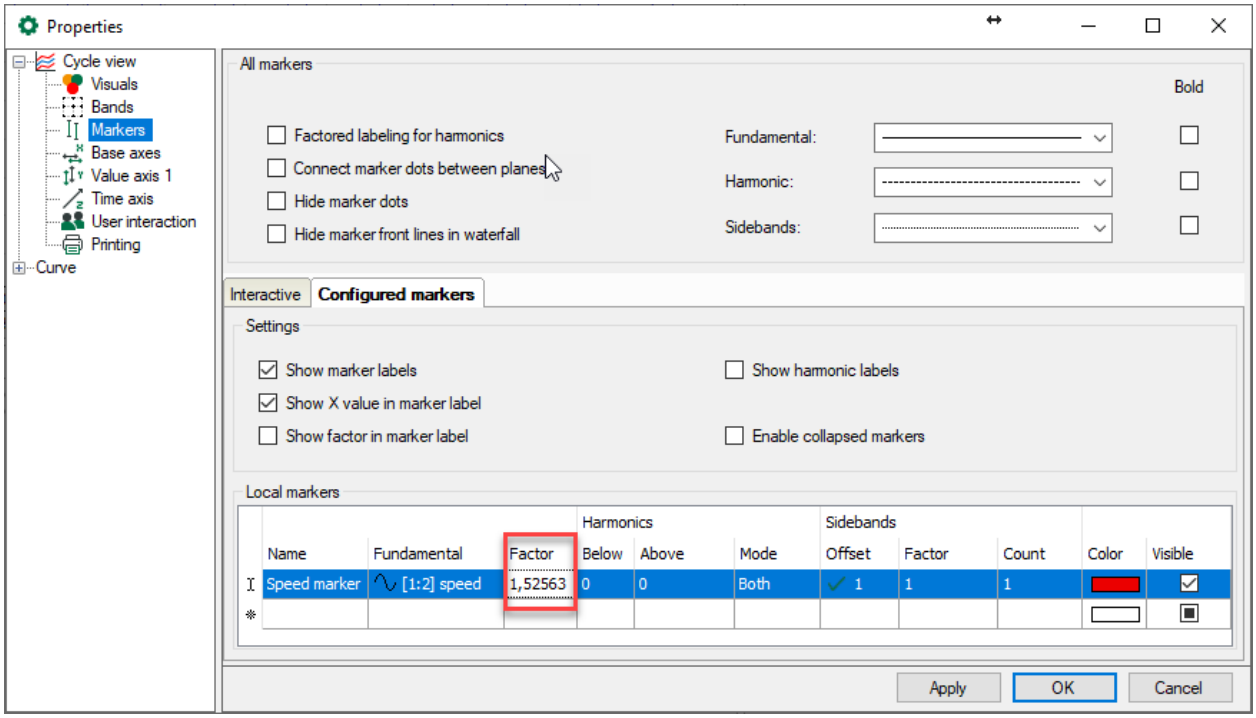




Note it is also possible to configure a dynamic offset i.e., an offset depending on a signal value.

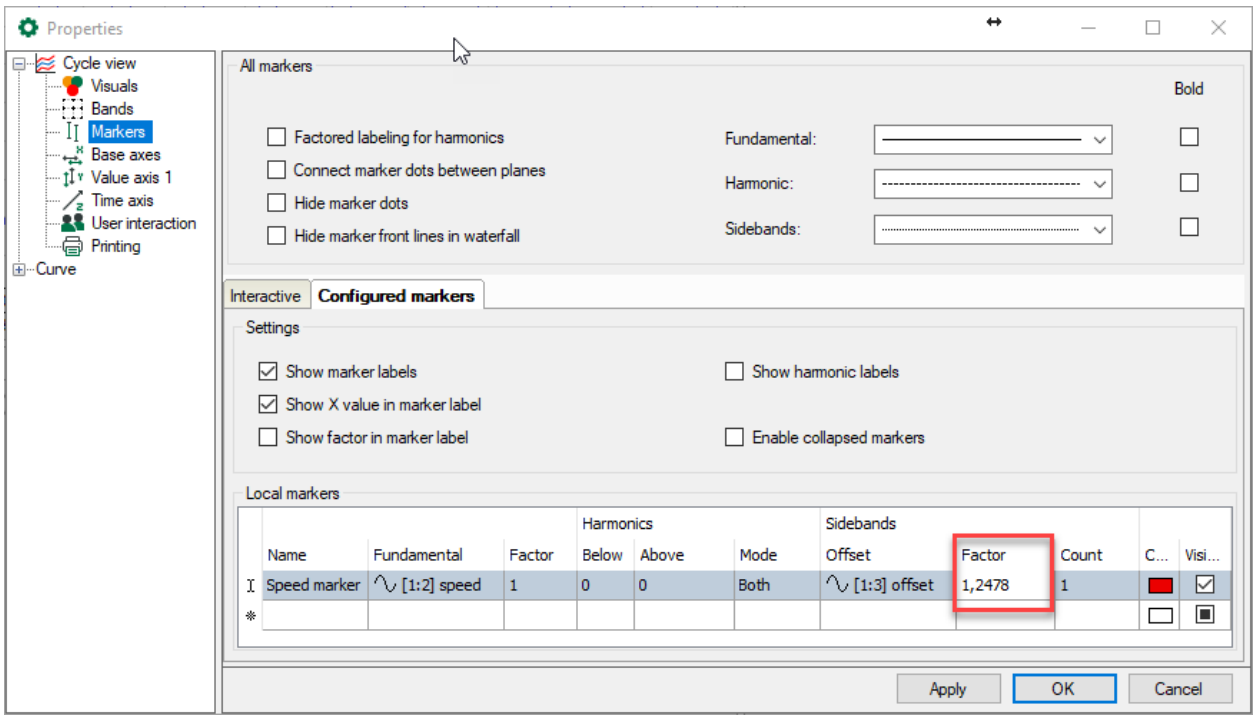
4.2 All markers are draggable

One can drag all configured markers. This also applies when the position of the marker depends on a signal value. The already existing *factor* variable is updated during dragging.



4.3 Draggable sidebands

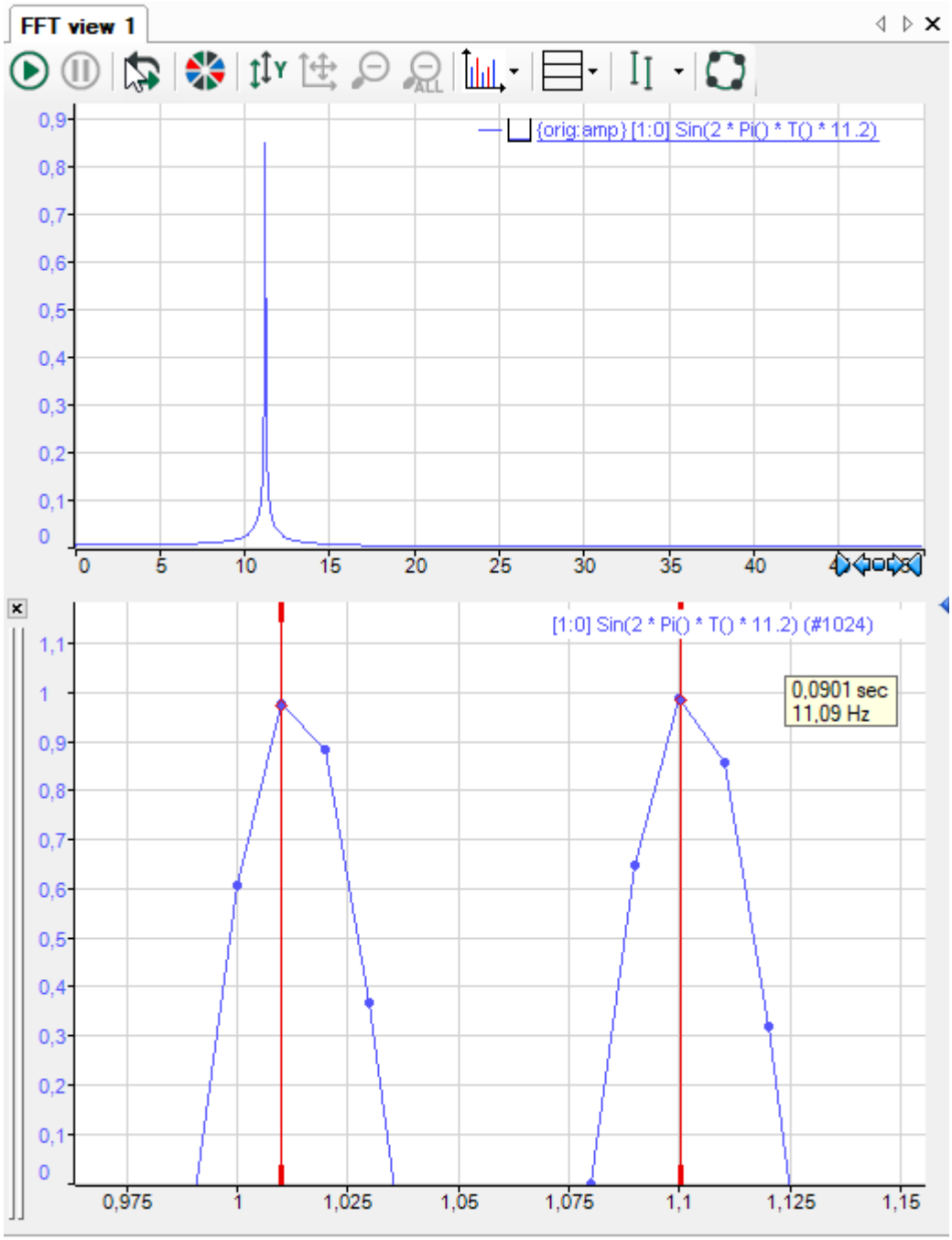
One can drag sideband markers with the mouse to increase/decrease the sideband offset. In previous versions, this was only possible for markers with a static offset. In the current version, this dragging is also possible if the offset depends on a signal value. Therefore, a *factor* variable was introduced. This factor is updated during the dragging.



4.4 Two markers in the time graph

In the Time Graph of the FFT view, there are 2 markers now. You can visualize these via the context menu of the graph. When they are visualized, a special legend is shown on the top right of the graph. Two values are displayed in this legend:

- The absolute time difference between the markers
- The inverse of this time difference

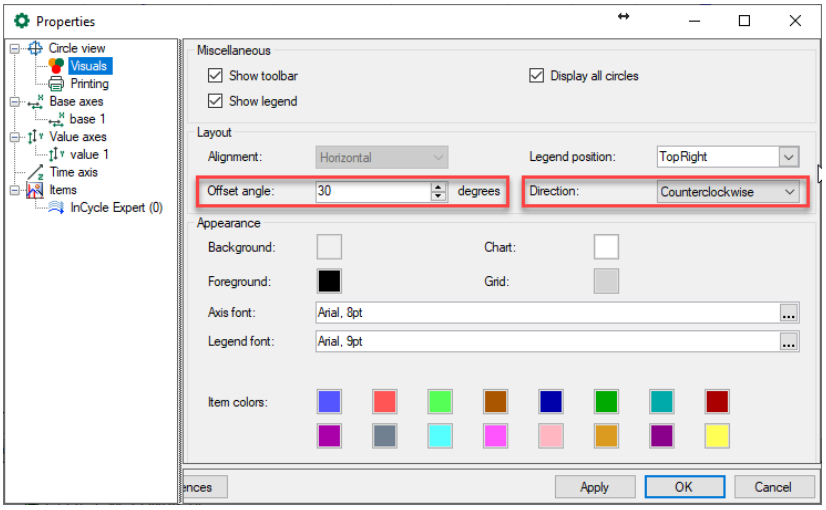


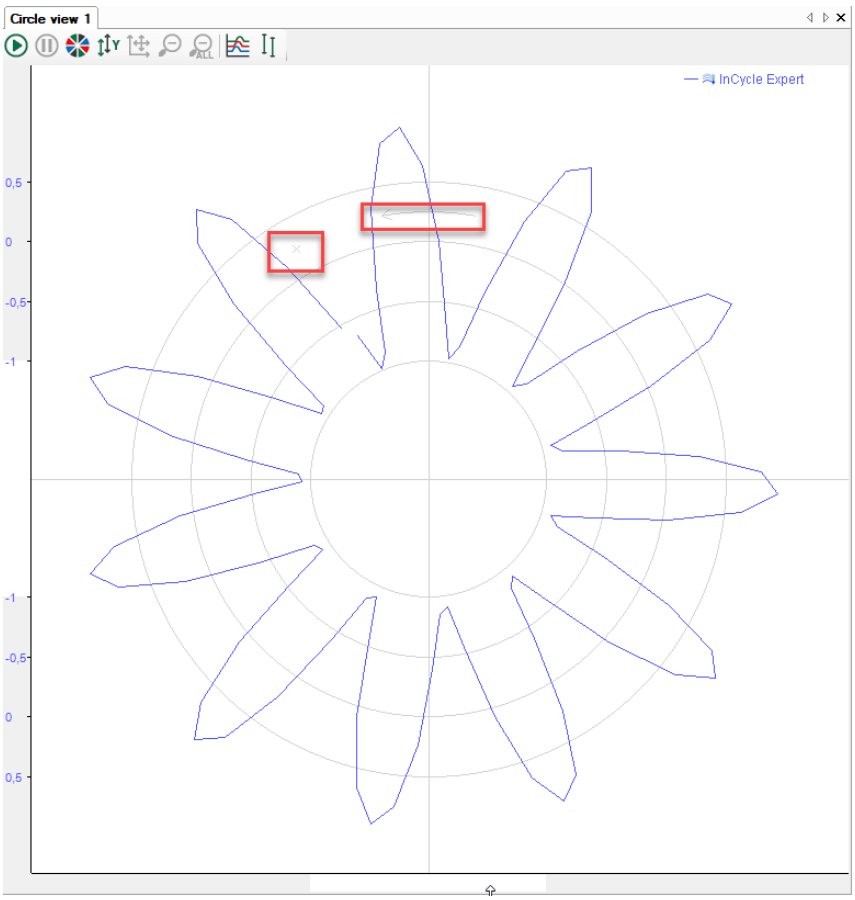
5 Circle view

5.1 Configurable offset angle and direction

In previous versions, the first data point of the cycle was always painted on top. Now, it is possible to configure an offset angle. One can configure this angle in the properties of the view (see first picture). Alternatively, one can rotate the cycle by dragging the grey cross (see second picture).

In previous versions, the data was always painted counterclockwise. In this version, the painting direction can be configured in the properties (see first picture). The painting direction is indicated with a grey arrow (see second picture).

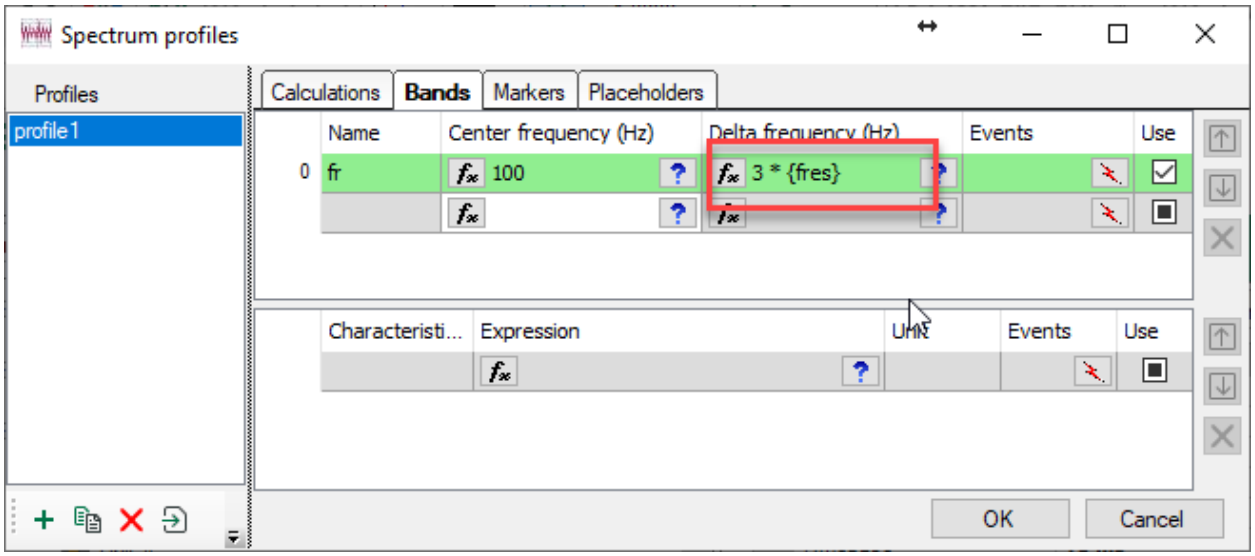




6 InSpectra Expert

6.1 Resolution placeholder

The frequency resolution of a spectrum is depending on the sampling rate and the number of samples per FFT. For generic profiles that can work for different sampling rates, it is useful to define the delta frequency of a band depending on the resolution, to make sure one has a certain number of lines inside the band. Therefore, the placeholder *fres* was introduced (= the frequency resolution):



6.2 Enable calculation signal

The functionality of the enable calculations signal changed. The buffering now immediately starts after a rise of this signal.