

imc STUDIO Monitor 2025

Manual

Doc. Rev.: 8.1 - 2025-03-11



Disclaimer of liability

The contents of this documentation have been carefully checked for consistency with the hardware and software systems described. Nevertheless, it is impossible to completely rule out inconsistencies, so that we decline to offer any guarantee of total conformity.

We reserve the right to make technical modifications of the systems.

Copyright

© 2025 imc Test & Measurement GmbH, Germany

This documentation is the intellectual property of imc Test & Measurement GmbH. imc Test & Measurement GmbH reserves all rights to this documentation. The applicable provisions are stipulated in the "imc Software License Agreement".

The software described in this document may only be used in accordance with the provisions of the "imc Software License Agreement".

Open Source Software Licenses

Some components of imc products use software which is licensed under the GNU General Public License (GPL). Details are available in the About dialog.

A list of the open source software licenses for the imc measurement devices is located on the imc STUDIO/imc WAVE/imc STUDIO Monitor installation medium in the folder "*Products\imc DEVICES\OSS*" or "*Products\imc DEVICEcore\OSS*" or "*Products\imc STUDIO\OSS*". If you wish to receive a copy of the GPL sources used, please contact our tech support.

Notes regarding this document

The manual for imc STUDIO Monitor is available both as an e-book and in PDF format. In the menu bar, click on the symbol  to open the help.

The program "[imc Help and Documentation](#)"  provides you with access to all formats and other documents such as the **device documentation** or the **technical data sheet**.

What is the best way to read the imc STUDIO Monitor documents?

Getting Started

Before installing the software, start by reading the document "*Getting Started*". This contains important tips for achieving problem-free installation, for updating the software, as well as for operation and integration of devices in measurement setups.

All information presented in the document "*Getting Started*" is also presented in the imc STUDIO Monitor manual.

Manual - complete documentation

The manual serves as reference material. This document describes how to operate the software and how to configure parameters for the devices. In some case, individual device groups have special functions. The documentation for these usually appear exclusively in the device manuals.

Training programs for introduction to the systems, and advanced workshops

We recommend that before you begin working with imc STUDIO Monitor you participate in an extensive training session. Such training will enable you to get started working efficiently much faster. Additionally, you will obtain valuable tips and information on how to use the software more effectively. More information is available on our homepage under "*Service & Training*" > "*imc ACADEMY*".

Special notes



Warning

Warnings contain information that must be observed to protect the user from harm or to prevent damage to property.



Note

Notes denote useful additional information on a particular topic.



Reference

A reference in this document is a reference in the text to another text passage.

Table of contents

1 General introduction	6
1.1 Before you start	6
1.2 Tech support	6
1.3 Service and maintenance	7
1.4 Legal notices	7
1.5 imc Software License Agreement	8
2 Chapter overview	11
3 Setting Up - Software	12
3.1 System requirements	13
3.2 Installation - Preparation	14
3.3 Installation - Step by step	18
3.4 Product configuration / Licensing	27
3.5 Start	27
3.6 Info / Version information	33
3.7 Information and tips	34
4 Documentation of devices - Help and Documentation	39
5 Monitor - First steps	40
6 imc STUDIO Monitor (general)	43
6.1 Experiments, projects and the database	43
6.2 Ribbon	54
6.3 Navigation pane and Quick access toolbar	68
6.4 Tool windows	69
6.5 User administration and Access rights	75
6.6 Views	84
6.7 Information and tips	90
6.8 Placeholders	95
7 Setup - Device (general)	115
7.1 Device overview	116
7.2 Ribbon	117
7.3 Tool Windows	129
7.4 Operation	132
7.5 Performing a measurement: Procedure	142
7.6 Metadata and Comments on the measurement	144
7.7 Information and tips	147
8 Setup pages - Configuring device	176
8.1 Documentation (experiment description)	177
8.2 Configuring devices	177
8.3 Configuring channels and variables	181
8.4 Additional pages	192
8.5 Storage options and directory structure	194

8.6 Storage media in the device	221
8.7 Information and tips	233
9 Data analysis and signal processing	237
9.1 Overview of the tools	237
9.2 imc Inline FAMOS	245
9.3 Inline Analysis	451
9.4 imc FAMOS dialog	475
9.5 Data Processing (general)	485
9.6 Bus Decoder - Decoding fieldbus log channels	487
9.7 Powertrain Monitoring - Machine and bearing diagnostics	492
9.8 imc WFT Assistant	516
10 Channel types, variable types, data types and data storage formats	523
10.1 Data types	523
10.2 Various variable types - Channel type	526
10.3 Data file format - Saving measured data and export/import formats	539
10.4 imc Data format	574
10.5 imc Format Converter	575
11 Panel - Data visualization and controls	588
11.1 Ribbon	589
11.2 Tool windows	594
11.3 Context menu	613
11.4 Design mode	616
11.5 Widgets - operation and properties	617
11.6 Curve window	647
11.7 Special Widgets	910
11.8 Pages	924
11.9 Variable linkage	939
11.10 Information and tips	943
12 Sequencer, events and commands	946
12.1 Sequence table	947
12.2 Event Dialog	948
12.3 Running and stopping	950
12.4 Creating a sequence of commands	951
12.5 Context menu	953
12.6 Information and tips	953
12.7 Command reference	954
13 Miscellaneous	1001
13.1 Last Changes	1001
Index	1009

1 General introduction

1.1 Before you start

Dear user.

1. The software you have obtained, as well as the associated manual are directed toward competent and instructed users. If you notice any discrepancies, we request that you contact our [tech support](#) ⁶.
2. Updates during software development can cause portions of the manual to become outdated. If you notice any discrepancies, we request that you contact our tech support.
3. Please contact our tech support if you find descriptions in the manual which you believe could be misunderstood and thereby lead to personal injury.
4. Read the enclosed [license agreement](#) ⁸. By using the software, you agree to the terms and conditions of the license agreement.



Note

Notes on the descriptions and the screenshots

- The help may also contain parts **shared imc software components**. These parts may differ from the rest of the help in terms of style and structure. All help files are equipped with a full text search functionality and have an index.
- The screen shots appearing in this documentation were created with a **variety of Windows versions** and their appearance may thus differ from that of your installed program.
- imc STUDIO works with [user groups and access rights](#) ⁷⁵. The **user group** has wide-reaching influence on the visibility and/or operability of menus, icons, etc. Note that in all imc STUDIO documents the administrator roll is assumed to be filled. All descriptions are therefore related to the full utilization of the user interface.
- The following descriptions and screenshots always pertain to the **view with the full scope of functions (Complete)**. Many of these functions are also found in the minimized view at another location.

1.2 Tech support

If you have problems or questions, please contact our tech support:

Phone: (Germany): **+49 30 467090-26**

E-Mail: hotline@imc-tm.de

Internet: <https://www.imc-tm.com/service-training/>

Tip for ensuring quick processing of your questions:

If you contact us **you would help us**, if you know the **serial number of your devices** and the **version info of the software**. This documentation should also be on hand.

- The device's serial number appears on the nameplate.
- The program version designation is available in the About-Dialog (click on the symbol  in the menu bar).

Product Improvement and change requests

Please help us to improve our documentation and products:

- Have you found any errors in the software, or would you suggest any changes?
- Would any change to the mechanical structure improve the operation of the device?
- Are there any terms or explanations in the manual or the technical data which are confusing?
- What amendments or enhancements would you suggest?

Our [tech support](#)  will be happy to receive your feedback.

1.3 Service and maintenance

Our service team is at your disposal for service and maintenance inquiries:

Phone: (Germany): **+49 30 629396-333** (Mon.-Fri.: 9.00 - 12.00 and 13.00 - 17.00)

E-Mail: service@imc-tm.de

Internet: <https://www.imc-tm.com/service>

Service and maintenance activities include, for example calibration and adjustment, service check, repairs.

1.4 Legal notices

Quality Management



imc Test & Measurement GmbH holds DIN EN ISO 9001 certification since May 1995 and DIN EN ISO 14001 certification since November 2023. You can download the CE Certification, current certificates and information about the imc quality system on our website:

<https://www.imc-tm.com/quality-assurance/>.

imc Warranty

Subject to the general terms and conditions of imc Test & Measurement GmbH.

Liability restrictions

All specifications and notes in this document are subject to applicable standards and regulations, and reflect the state of the art well as accumulated years of knowledge and experience. The contents of this document have been carefully checked for consistency with the hardware and the software systems described. Nevertheless, it is impossible to completely rule out inconsistencies, so that we decline to offer any guarantee of total conformity. We reserve the right to make technical modifications of the systems.

The manufacturer declines any liability for damage arising from:

- failure to comply with the provided documentation,
- inappropriate use of the equipment.

1.5 imc Software License Agreement

imc Test & Measurement GmbH
Voltastr. 5
13355 Berlin, Germany
Trade register: Berlin-Charlottenburg HRB 28778
Managing directors: Michael John Flaherty, Frank Mayer

imc Test & Measurement GmbH
Terms and Conditions
Governing the Use of imc Test & Measurement GmbH Software
As of: January 18th, 2024

§ 1 Objects of the Agreement

- (1) In addition to the "General Terms and Conditions Governing imc Test & Measurement GmbH Deliveries and Services to Customers", these terms and conditions apply to all contracts concluded with imc Test & Measurement GmbH (hereinafter referred to as "imc") which involve the transfer of rights of use to any software developed by imc (standard software, software created or adjusted specifically for the Customer, which is recorded on the machine-decodable data carriers such as data files, databases and database material, updates, upgrades, releases, etc., including corresponding documentation, information and materials, hereinafter referred to as "Software").
- (2) The Software is provided to the Customer as an executable object program on machine-decodable data carriers specified in the "Objects of the Agreement". The Software's product documentation is also supplied to the Customer either in print or on a machine-decodable data carrier. Unless otherwise expressly agreed in writing, the Customer is not issued the source code of the Software.

§ 2 Rights of Use, Scope

With regard to any transfer of rights of use to Software created by imc, the following provisions apply:

(1) Basic provisions

- a) The Customer is granted a non-exclusive and – subject to the terms and conditions governing the use of Software by third parties, resale and leasing – non-transferrable right of use to the Software for its own purposes. "Use" signifies running the programs and editing the data records.
- b) Until each due fee is paid in full, the Customer is entitled to use the Software solely on a revocable basis. If the Customer is in default with regard to the payment of fees, imc is entitled to revoke the use of the respective services for the duration of the default. The Customer is granted the permanent right to use copyright protected services, in particular the Software, only upon full payment of the agreed fee.
- c) The Customer agrees to undertake appropriate precautionary measures to prevent unauthorized access by third parties to the Software. The original data carriers and the data carries used to make copies as per the agreement, as well as the documentation, are to be stored in a secure location. Employees are to be notified that the production of copies beyond the scope of the agreement is not permitted.
- d) If the right of use is revoked or expires due to another reason, the Customer is obligated to return to imc the Software, the copies made by the Customer and the documentation. Provided that a physical return of the Software and the copies is not possible due to technical reasons, the Customer is obligated to delete such and confirm deletion to imc in writing.

(2) Reproduction

- a) The Customer is entitled to make copies of the Software only if copies are necessary to use the Software in accordance with the contract. The following are considered cases in which reproduction is necessary: installation of the Software from the original data carrier onto the hard disk drive of the hardware used, as well as loading the Software into the computer memory.
- b) The Customer is entitled to create a backup copy if such is necessary to safeguard future use. Copies may only be made for other purposes after prior written consent has been issued by imc.
- c) The Customer is not allowed to make any reproductions other than those expressly permitted under the provisions of this agreement.

(3) Use of the Software by Third Parties, Resale and Leasing

- a) The Software may be used for the purposes stipulated in this contract, in particular for the Customer's business operations. Access to the Software may also be provided to parties which rely on using the Software as instructed by the Customer. In particular, the Customer is entitled to operate the Software or allow the Software to be operated on data processing devices, which are located on the premises of and are directly owned by a third party company (outsourcing). The prohibition against multiple use remains unaffected.
- b) The Customer may permanently sell or give the Software to third parties provided that the Customer is granted permanent use of the Software. In the context of its period of use, the Customer may temporarily transfer the Software to third parties for a fee or free of charge. The prohibition against multiple use remains unaffected. The Customer is expressly notified that transfer to third parties is not permitted and use by third parties is technically not possible if an individual license must be acquired or an individual activation is required for third party usage, such as in the case of runtime licenses.
- c) With regard to the valid use of Software by a third party, the Customer is obliged to ensure that the third party acknowledges the provisions of this agreement governing the rights of use as binding for such third party. The Customer may not transfer Software and documentation to third parties if there are grounds to suspect that the third party may infringe upon the provisions of this agreement governing the rights of use, in particular with regard to the unauthorized production of copies.
- d) Subject to the provisions stipulated in § 4 Paragraphs 1 and 2 or a deviating express agreement in writing, the Customer may not use the Software while the Software is being used by a third party (prohibition against multiple use); in the event that the Software is transferred to the third party, the customer is obliged to surrender to imc all Software copies including, if applicable, all existing backup copies, or to destroy copies not surrendered.

(4) Decompilation

The reverse translation of the provided program code into other code forms (decompilation), disassembling and other forms of reverse engineering of the various production phases of the Software is not permitted. If interface information is required to achieve the interoperability of a separately created computer program, such may be requested from imc, or a third party to be named by imc, for a minor fee. Section 69 e of the German Copyright Act ("UrhG") remains unaffected by this provision.

(5) Changes by imc

If imc conducts adjustments, changes or enhances the Software on behalf and on account of the Customer, the Customer thus acquires the corresponding rights of use to the changes or enhancements of the Software to which he is entitled according to the stipulations of this agreement.

(6) Exceptional Usage Requests by the Customer

If the Customer requests to use the Software according to terms which deviate from the requirements stipulated in Paragraphs 2 through 5, this exceptional use of the Software must be agreed in writing by imc. In such an instance, the Customer agrees to provide imc with information about the desired scope of use, the pertinent field of application, etc. If imc subsequently grants a license covering the Customer's special intended use, the parties agree that a new license fee is owed by the Customer, which is independent of payments made by the Customer for the previously existing license.

§ 3 Copyright, Protection of the Software

- (1) The intellectual property, in particular the copyright as well as all industrial property rights and trade secrets, are retained by imc and are not transferred to the Customer. The Customer's ownership of the machine-decodable data carries and data processing units remains unaffected.
- (2) Copyright notices, serial numbers as well as designations and reservations of rights which serve as program identification or a protective right may not be removed or changed. The Customer is obliged to transfer the existing protective right notices to all copies. In particular, backup copies of the Software must be expressly designated as such.

§ 4 License Types, Multiple Use

- (1) In the case of a Single-User License, the Software may be activated and run on only one data processing unit. "Activation" refers to the process of transferring the license to the data processing unit.

If the technical specifications for the Software permit a second activation, then the Customer may additionally activate the Software on a second data processing unit. However, the Software may only run on one data processing unit at any one time, not on both simultaneously.

- (2) With a Network License, the Software may be run on as many data processing units as the amount of licenses obtained. In this case a central data processing unit acts as the license server for which the activation process is performed.
If the technical specifications for the Software permit a second activation, then the Customer may additionally activate and run the Software on as many data processing units as the amount of licenses obtained. However, these additional data processing units must be used by the same users who operate the Software via the license server.
- (3) Subject to the provisions in Paragraphs 1 and 2 or a deviating express agreement in writing regarding network use, multiple use of the Software is not permitted.
- (4) If the data processing unit is changed, the Customer is obliged to delete the Software from the hard disk drive of the previously used hardware.

§ 5 Software-Subscription

If the Software used is a software-subscription, the following additional restrictions apply:

- (1) The right of use is valid for a limited time period. The start and end of the time period are specified. After the end of the time period, the right of use is expired.
- (2) If the Customer wishes to continue using the Software after elapse of the specified time period, the subscription must be renewed.

§ 6 Trial Version

If the Software used is a free trial version, then the following additional limitations apply:

- (1) The trial version only entitles the user to test the Software. In particular, commercially productive utilization is not permitted.
- (2) The rights of use granted expire after the elapse of a period stated in the product description.

§ 7 License Key

- (1) Upon delivery of the Software the Customer receives a License Key. Using this License Key, the Customer is able to activate the Software purchased. By means of this License Key the Customer can also view his license status and order updates and upgrades.
- (2) The License Key is to be protected against access by third parties in order to prevent misuse. If, however, a third party gains unlawful access to the Key, the Customer is obliged to notify imc immediately via telephone, as well as in writing, so that the previous License Key may be suspended and a new one issued.

§ 8 Conclusion

- (1) The law of the Federal Republic of Germany shall apply under exclusion of private international law. The provisions of the UN Convention on Contracts for the International Sale of Goods (CISG) do not apply.
- (2) The place of performance for all obligations arising from this agreement is imc's registered seat. Insofar as the Customer is a merchant as defined by the German Commercial Code (HGB), a legal entity under public law, or a special asset under public law, the exclusive place of jurisdiction for all disputes directly or indirectly arising from the contractual relationship is agreed as imc's registered seat. The same applies to persons who have no general place of jurisdiction in Germany, as well as to persons who have moved their place of residence or usual whereabouts abroad since conclusion of the contract, or whose place of residence or usual whereabouts is unknown at the time the action is filed. In addition, imc is entitled to file suit at the statutory venue.
- (3) Oral side-agreements are not valid. Deviating or supplementary conditions as well as modifications of this contract, including this written requirement clause, are only valid if agreed in writing and expressly marked as a modification or supplement.
- (4) If certain provisions of this contract are inoperative or unfeasible, this does not prejudice other provisions of the contract. The contracting parties agree to contractually substitute an operable provision which approximates the commercial intention of the contract as closely as possible for any inoperative one.

2 Chapter overview

In this manual, you will find a detailed description of how to operate the imc STUDIO Monitor software.

Section	Synopsis
General introduction	Quality Management ^[7] , Tech support ^[6] , License agreement ^[8]
Setting Up ^[12]	Software setup operations • System requirements ^[13], Installation ^[14], Start ^[27]
 Monitor - First steps ^[40]	What steps are necessary in order to obtain measured data?
imc STUDIO Monitor (general) ^[43]	The imc STUDIO Monitor user interface and its operation • Project Menu ^[54], Creating an experiment ^[49] and "What is saved where?" ^[91]
 Setup - Device (general) ^[115]	The device configuration • Performing a measurement ^[142] and Devices supported (Device Overview) ^[116] • Metadata, additional columns ^[147]
 Setup pages - Configuring device ^[176]	The device configuration • Configuring device (Setup pages) ^[176] • Storage options ^[194] •
 Data analysis and signal processing ^[237]	Various possibilities for processing and analysis of measured data • during a running measurement on the PC: Inline FAMOS ^[245], Inline Analysis ^[451], Bus Decoder ^[487], Powertrain Monitoring ^[492] • subsequent analysis (post-processing): imc FAMOS ^[475]
 Channel types, variable types, data types and data storage formats ^[523]	In imc STUDIO Monitor, the various components work with variables, and some work with files as well. Variables can take a variety of formats and types. This chapter provides an overview of the various possibilities and applications. • Data types ^[523] • Various variable types - Channel type ^[526] such as User-defined variables ^[536] • Data file format - Saving measured data and export/import formats ^[539] such as Parameter set ^[546]
 Panel ^[588]	Display of measured data and operation via the user interface • Curve Window ^[647] and Widgets instruments ^[617] • Variable linkage ^[939] and Data Browser ^[597]
 Sequencer, events and commands ^[946]	Creating automated routines • Sequence table ^[947] and Creating a sequence of commands ^[951] • Command reference ^[954]

Tech support

If you have any open questions about our products, please contact our tech support.

Questions or problems? Contact our [Tech support](#) ^[6].

3 Setting Up - Software

This chapter describes the **first steps** in operating imc STUDIO Monitor and the **installation of additional imc products**.

imc STUDIO Monitor is an application which is constructed upon imc STUDIO. Many components of imc STUDIO are used. For this reason, some of the descriptions may make reference to imc STUDIO.

Which components are available depends on the product installation (order).

Chapter overview

Synopsis	Section
Startup operations for imc STUDIO - installation and product configuration prior to first use	• System requirements  13 • Installation / Uninstall  14 • Product Configuration / Licensing  27
The first start, important settings	• The first start  27 • Device connection / Network / Firewall  29
List of components used and their versions	• Version Information  33

3.1 System requirements

Supported operating systems

Windows 10*/11* (64 bit)

*released in conformance with the version of Windows 10/11 applicable at build date of imc software

Minimum requirements for the PC

4-core CPU 2 GHz ¹

8 GB RAM (recommended: 16 GB RAM) ¹

10 GB free hard disk space (recommended: SSD) ²

Display resolution: 1366 x 768 (recommended: 1920x1080)

1 Requirements and recommended minimum configurations for the applied PC will rise with increasing number of connected devices and the resulting overall system data rate. Another crucial factor will be the extend of used live analysis and visualization functions on the PC (in particular Data Processing, imc Inline FAMOS and active curve windows).

A smooth operation will also heavily rely on sufficient RAM memory resources on the PC: It must ensure to satisfy all vital functions without any outsourcing to slow external swap file (HDD/SSD) in order to guarantee sustainable performance.

2 The hard drive memory requirement increases if measured data are saved to the PC.

Other operating system components

The following components are installed with the imc STUDIO Monitor setup if they are not already present:

Component	Version	Folder in "System" directory
Microsoft .NET Framework	4.8	DotNetFx4.8
Microsoft VC 2015-2022	14.38.33135	Microsoft Visual C++ Redistributable\2022
Microsoft VC 2010	10.0.40219.1	Microsoft Visual C++ Redistributable\2010
Microsoft Build Tools 2015	14.0.23107.10	MSBuildTools2015

Supported measurement devices

Which devices you can use in imc STUDIO Monitor is described in the documentation on the "Setup" > "[Device Overview](#)" ¹¹⁶, as well as in the "Technical Data Sheet".

3.2 Installation - Preparation

Software requires a license

This means that the program may only be started **upon obtaining a license** (see [Product Configuration / Licensing](#) ).

Administrator rights required

For the purposes of installation and uninstallation, a user account with **administrator privileges for the PC is required**.

If you are logged on to the PC **without administrator rights, log yourself out** and log back on with an administration-level user account. If you don't possess the appropriate account type, you will need the support of your system administrator or IT department.

See also [Notes on Windows User Account Control](#) .

Restarting the computer in the process of installation

During the installation process, the installation program will prompt you to restart the PC.



Note

Restarting

After the restart, login to **the same user account** with which you had previously begun the installation.

Coexisting applications: imc STUDIO, imc STUDIO Monitor, imc WAVE, ...

Some imc programs are installed as an independent and specially adapted instance of imc STUDIO. They are based on imc STUDIO.

Unless otherwise stated, these programs are installed and used in parallel. As long as these instances are based on the same version of imc STUDIO (e.g. 2022 R1), they are all subordinated to the same program installation, which means they share resources.

For this reason, installation of the instances in this case must be performed in one single shared Setup procedure. Any attempt to perform installation in succession or subsequently will cause the already existing instances to be deleted.

This applies particularly to the joint and parallel installation of imc STUDIO Monitor, imc WAVE and imc STUDIO, which must be performed in a single step.

Do not install imc STUDIO, imc WAVE and imc STUDIO Monitor in succession, but always at the same time.

Update or parallel installation

The Setup utility checks whether any version of imc STUDIO (imc STUDIO Monitor) is currently installed on the computer. If so, it can be uninstalled by means of the Setup. An associated confirmation prompt is posted in that case. All user data such as the database remain intact.

The **two versions** can be **installed in parallel** provided that they have differing version numbers (e.g. 5.2 and 2023). The new version can be installed in the same folder (default case: "C:\Program Files\imc"). In this folder, a new subfolder with the new version number is created for imc STUDIO.

In both cases, you are able to **adopt** a variety of **settings** from the old versions. These include the project settings and views. By contrast, other settings such as the product configuration and the database path need to be set up from the beginning.

Adopting the settings with the help of the existing database (recommended)

When an existing database continues in use, any settings which can be retained will be retained. For a parallel installation, two data bases are needed. Create a copy so that the old imc STUDIO Monitor version can continue to work with the existing database. A confirmation prompt about making a copy is posted.

See: [Update with the help of the existing database](#) 

Adopting the settings without using the existing database

Some settings, e.g the views, can also be retained without the database. To do this, save and import the appropriate settings.

See: [Update without using the existing database](#) 

The subsequent procedure - an overview

Follow the installation program instructions (see: "[Installation step by step](#)" ).

- The installation process first checks whether the **required system software** is installed. If ones is missing, it will be installed automatically.
- After restarting the system, the selected products are installed.
- After the installation is concluded, it is possible to start imc LICENSE Manager directly to activate your license.
- Once the license has been activated, imc STUDIO Monitor is ready for use.

Uninstalling

Un-installation is performed via the menu item "*Windows-Settings*" > "*Apps & features*".



Reference

See also

- [Unattended Installation - Silent Installation](#) 
- [Installation - Installing projects](#) 
- [Recommended Virus Scanner Settings](#) 
- [Changing Languages and Installing Additional Languages](#) 

3.2.1 Update with the help of the existing database

When an existing database continues in use, any settings which can be retained will be retained. This includes Views, any Setup-columns created by the user, the user administration and all experiments.

If you wish to run multiple versions of imc STUDIO Monitor or maybe restore an older version at a later time, create a copy of the database.



Note

The database

It is impossible to use the imc STUDIO Monitor database by both versions in parallel.

- If the same path is selected in the new imc STUDIO Monitor version, the database is used automatically. When loading old experiments, a note is posted in the logbook that the **experiments belong to an older version. Once saved, they can no longer be loaded** with the old version.
- If the database structure has been altered, you will be notified of this. A **dialog for performing conversion** appears. This dialog can be used to convert the database and, if desired, copy it first. After the conversion, the entire **database can no longer be used with the old version**.



Note

Using new views

- Be aware that the new version provides new functions, such as new or enhanced Setup pages and new menu items.
- **Use of the new Views is recommended, in order for these new functions to be accessible! Please look under "What's new" whether there have been any changes in this regard.**
- User-made columns such as meta data columns are not automatically inserted into the pages. The configurations of these columns, however, is adopted from the old View. You can re-insert these columns at the desired position (by means of the Column Chooser).
- Multiple views are provided. Select a view and add whatever Setup pages are required or saved, and save the view under a new name.
- How to save Views is explained in the section "*imc STUDIO Monitor (general)*" > "[Views](#)"⁸⁴.

3.2.2 Update without using the existing database

In imc STUDIO, various settings were saved with the respective **project** and apply to all the experiments it contains. Among other things, these include the view settings: user-defined views, column configurations (e.g. metacolumns).

In order to **retain these settings** in the new version after having **performed the update** without using the existing database, export of the settings are necessary.

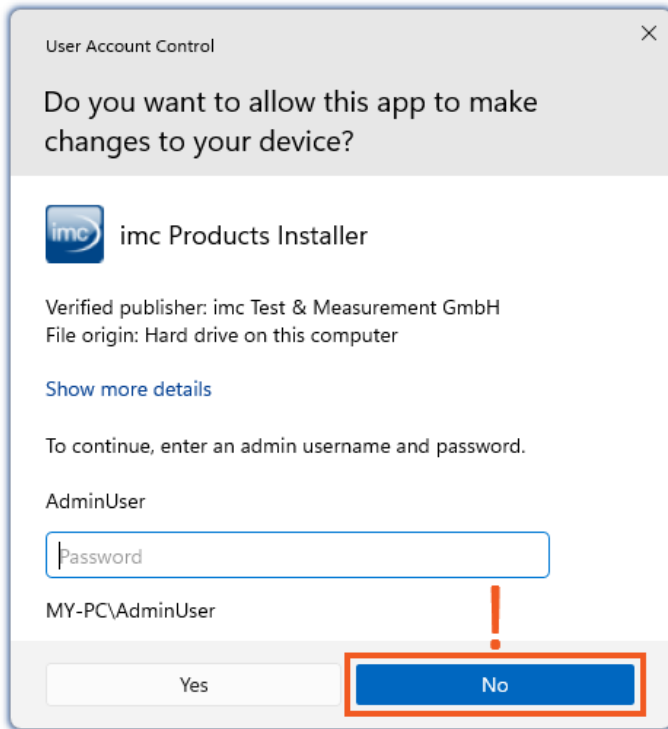
Please first check which settings you will need.

- If you wish to reuse **all of the project's settings**, [export the project settings](#)⁵⁰ (by means of the Projects-dialog: Menu ribbon: "*Project*" > "*Manage Project*").
- If you wish to reuse **only the viewing settings**, [export the views](#)⁸⁶.

3.2.3 Notes and troubleshooting

Notes on Windows User Account Control

Do not change the user account



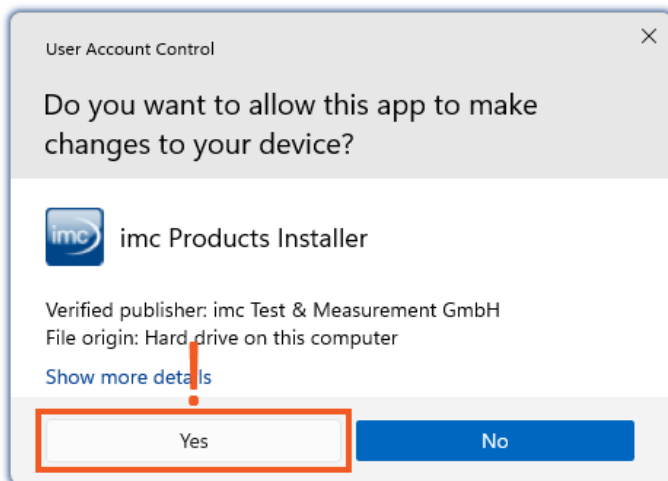
*Example of a password request.
Select "No".*

With Windows, installation can be started without administrator rights. In this case, the operating system requires selection of a user account and associated password, as shown in the example.



Do NOT continue with the installation in this way, otherwise it will be incorrect. **Select "No"**.

Confirm the prompt by the user account control



Select "Yes", to start the installation.

When you are logged in to a user account possessing administrator rights, you may receive a prompt by the Windows User Account Control (UAC) to allow changes. Select **"Yes"**.

Notes on the security software

Some virus scanners **prevent correct installation** of imc programs. We are currently aware of this affecting products from the companies McAfee and ESET. In principle, almost any virus scanner can have settings possibilities which prohibit necessary functions during installation.

For installation purposes, certain steps are necessary, such as:

- registration of programs for Autorun
- registration of programs as a service
- running of scripts from the TEMP-folder
- ...

The error profile may include various messages during installation. Or installed programs failing to launch.

In such cases, please contact your administrator to find out whether certain rules can be suspended for the duration of the installation. For any further questions, please contact our [tech support](#) .



Reference

See also

[Recommended Virus Scanner Settings](#) 

3.3 Installation - Step by step

Actual texts appearing in the user interface may differ from those shown in the screen shots, depending on the product configuration (path/version name).



Note

The following installation descriptions deal with several products and components. Not everything which you can activate can actually be used in imc STUDIO Monitor. For an exact listing of all of imc STUDIO Monitor's functions, see the Technical Specs sheet.

If you activate components which are not available in imc STUDIO Monitor, there won't be any negative effects on your program. They may even have some purpose if you use multiple products in parallel.

Download and Installation

The most up-to-date version can be downloaded from the imc website under "*Downloads*" > "*imc STUDIO*". Both a complete and a reduced installation package are offered. The reduced installation package is sufficient if you have only purchased imc STUDIO Monitor. Which products are included is indicated next to the Download button.

Start the download first, and next the file (e.g. "*Installer_imc_STUDIO_2024_R2.exe*").

The Installer unpacks the installation files. Subsequently, it independently starts the actual installation. Select an appropriate folder into which to unpack the files. The system's default suggestion is the Desktop.

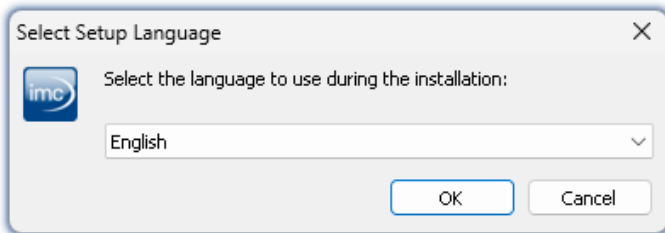


Hinweis

Installation files

The unpacked files can be deleted or saved once installation is completed. You can also use the unpacked files for installation at a later time. You do not necessarily need the Installations-exe file.

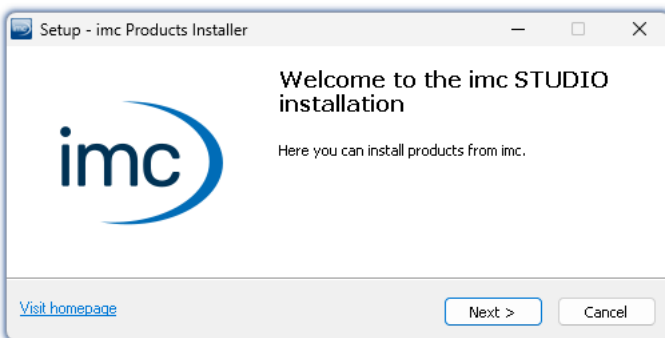
Installation language settings



Selecting the language during installation

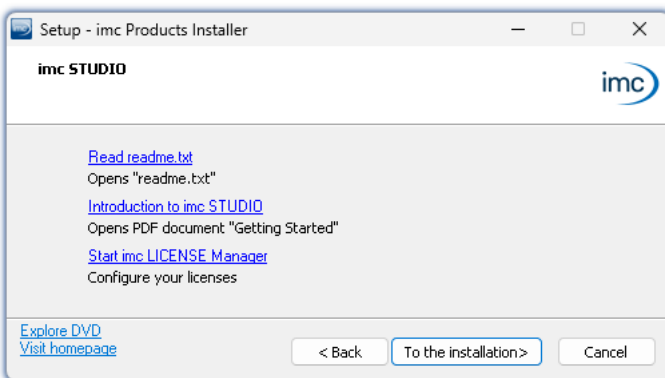
After starting installation, a window appears in which to select the installation language.

Performing installation



Installation setup welcome page

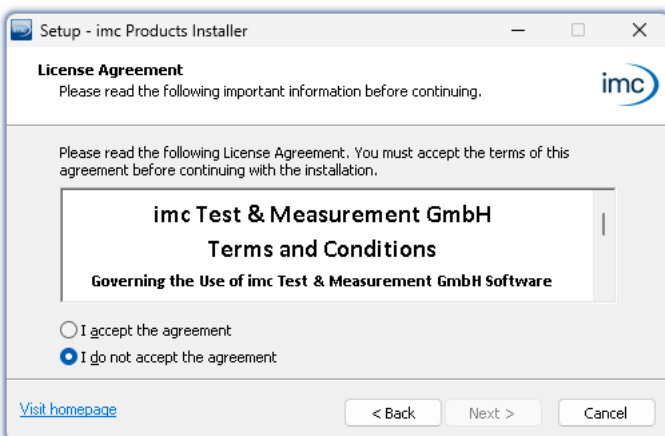
The installation setup then starts in the language selected.



Before starting installation

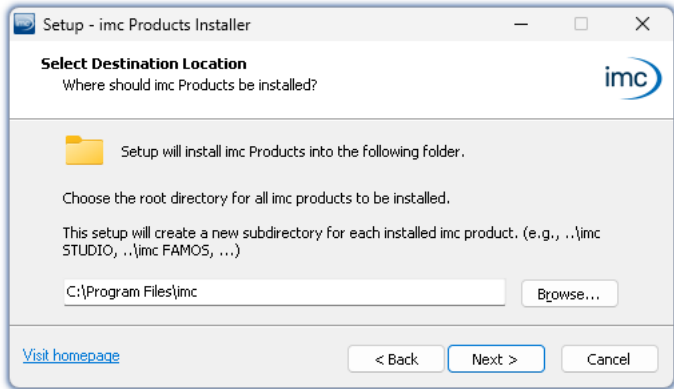
On the installation setup's next page, you are offered the following choices:

- open the "Readme"-file,
- open the "Getting Started"-document,
- separate installation of imc LICENSE Manager and
- display of the installation medium contents.



License agreement

You must accept the license agreement to continue the installation.



Entering the installation path

The next step is to select the installation folder. Under this folder, a separate subfolder will be created for each imc product (e.g. imc STUDIO 2024). It is recommended to end the associated pathname with "*imc*".

3.3.1 Select the products / Installation type

Now you can select which components should be installed. There are three types available:

- "Full functionality for 30-day demo" : hereafter called "Demo"
- "Typical functionality incl. imc STUDIO Professional" : hereafter called "Typical"
- "[User-defined](#)²³"

"User-defined" gives you the ability to affect all installation settings.

For the "Demo" and "Typical" installation types the installation settings are already configured. If you need further settings choose the "[user-defined](#)²³" installation type.

Installation type	Beschreibung
Demo	Use the installation type "Demo" in order to test the full scope of imc STUDIO and/or imc WAVE for 30 days. No other products which require a password will be installed. imc STUDIO Monitor is not installed.
Typical	Use the installation type "Typical" in order to install the Professional edition of imc STUDIO including all necessary components. No other products which require a password will be installed. imc WAVE and imc STUDIO Monitor are not installed.
User-defined	Use the installation type "User-defined" if you wish to configure the individual products . With this type, it is possible to install password-protected components . Further, you can install and configure additional imc products such as imc FAMOS . Note that some components can require a separate license.

Note

In order to install imc STUDIO Monitor, you must choose the installation type: "[User-defined](#)²³" or "Demo"!

Note

imc STUDIO Monitor reconfigure afterwards

Regardless of which variant of the installation is selected, when imc STUDIO Monitor is installed, all components are included. This way, after successful installation it is possible to adapt the imc STUDIO Monitor configuration at any time by means of the "[product configuration](#)²⁷".

Products	Demo	Typical	User-defined	Description
				● : included ○ : optional
imc LICENSE Manager	●	●	●	Managing the licensing (internal functions, does not appear in the list)
imc Shared Components	●	●	●	Providing common components of the imc products, such as the curve window (internal functions, does not appear in the list)
Required system components	●	●	●	System components require using of imc products.
imc SENSORS (1)			○	imc SENSORS is a database application for administering and editing sensor information. Applicable with devices of the firmware group A ^[116] (imc DEVICES).
imc STUDIO Developer (Demo) (2)	●		○	imc STUDIO is modular software platform handling all aspects of modern measurement technology from simple data acquisition to sophisticated test configurations.
imc STUDIO Professional (3)		●	○	
imc STUDIO any edition (3)			○	
imc STUDIO Monitor (3)			○	imc STUDIO Monitor makes it possible to connect with one or more measurement devices, particularly for the purpose of monitoring the instantaneous readings. Data can be viewed and edited live on multiple workstations. It is based on imc STUDIO and is installed as a separate instance of imc STUDIO.
imc WAVE (Demo) (2)	●		○	imc WAVE is a software package for NVH (Noise Vibration and Harshness) analysis. It is based on imc STUDIO and is installed as a separate instance of imc STUDIO. It can be equipped with multiple, separately licensed analyzers.
imc WAVE (3)			○	
Firmware and driver package imc DEVICES	●	●	○	Package of device drivers and firmware required by imc STUDIO and imc WAVE for devices of the firmware group A ^[116] (imc DEVICES - e.g. CRONOS family, C-SERIES, SPARTAN, BUSDAQ).
Firmware and driver package imc DEVICEScore	●	●	○	Package of device drivers and firmware required by imc STUDIO and imc WAVE for devices of the firmware group B ^[116] (imc DEVICEScore - e.g. ARGUSfit, EOS).
imc FAMOS Enterprise (Demo) (2)	●		○	imc FAMOS is a program for analyzing, evaluating and documenting measurement results.
imc FAMOS Reader (3)	●	●	○	
imc FAMOS any edition (3)			○	
imc Format Converter	●	●	○	Converts imc measurement data to other formats such as EXCEL and ASCII.
imc Documentation	●	●	○	Installs the documentation on the selected products. If the checkmark is removed, it is not possible to open the documentation via the software.

- 1: imc SENSORS requires a password for installation.
- 2: The associated Demo license must be activated after concluding installation by means of imc LICENSE Manager, otherwise it is not possible to start the product. Once the demo trial period has elapsed, use the [product configuration](#) ^[27] to convert the product to the license you have purchased.
- 3: The associated license must be activated after concluding installation by means of imc LICENSE Manager, otherwise it is not possible to start the product.

When you select "Demo" or "Typical", the start of the installation follows immediately (see: "[Starting the installation](#)" ^[26]). By contrast, if you select "User-defined", the installation is first [configured](#) ^[23].

3.3.2 User-defined

The installation variant can be configured in detail. Based on your selection of components desired, different installation steps are displayed.

Selecting components

Selecting User-defined installation

Once you have selected the installation type "*User-defined*", it is possible to select/de-select the desired products in the box at the bottom.

Activate imc STUDIO Monitor here!

imc STUDIO and imc WAVE, as well as imc SENSORS are not required for the purpose of operating imc STUDIO Monitor.

Required system components

Required system components (example)

In the next step, the system components which the products selected require are displayed, along with a note about which are already installed on your system. It is also possible to re-install any already installed components. Which components are missing/present depends on the system and its update status, so that the illustrations are intended as an example only.

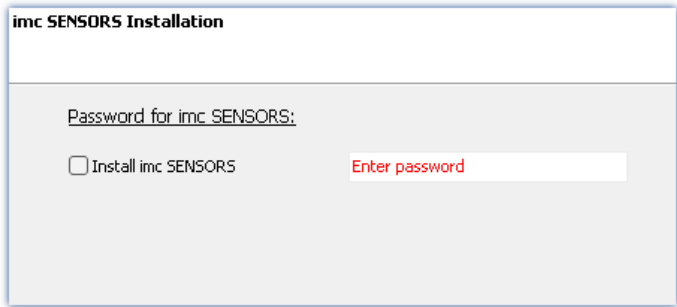
Configuring imc Shared Components

Configuring imc Shared Components

In this installation step, it is possible to install password-protected components of imc Shared Components.

Option/Component	Description
COM Products	The imc COM programming interface is a system integration tool
Class Counting Kit	Function library for imc COM products
Spectral Kit	Function library for imc COM products

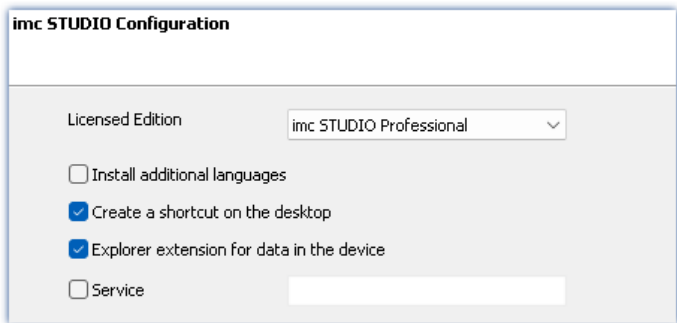
Password for imc SENSORS



Password prompt for imc SENSORS

If you have selected imc SENSORS, you are prompted to enter the password prior to installation. But if you do not wish to install imc SENSORS, de-select it here.

Configuring imc STUDIO



Configuring imc STUDIO

It is possible to make settings for your licensed product edition here. In consequence, later [product configuration](#) ²⁷ can be omitted.

Option/Component	Description
Licensed edition	Information on the editions is provided in the " <i>Technical data sheet</i> " for imc STUDIO.
Install additional languages	With this option, imc STUDIO Monitor is installed in all available languages. (This will substantially prolong the installation procedure). Please be aware that some functions may require the other languages. For instance, foreign-language parameter sets can only be imported if the associated language is installed. If this option is deactivated, English language is installed plus the language of the operating system, if available. If needed, all other available languages can be installed at a later time (see the section "Installing additional languages" ³⁵).
Create a shortcut on the desktop	Program starting links can be created on the Desktop.
Explorer extension for data in the device	Access to the device hard drive via the Windows-Explorer. for devices of the firmware group A ¹¹⁶ (imc DEVICES)
Service	Service support. Install this component only if requested by our tech support. for devices of the firmware group A ¹¹⁶ (imc DEVICES) and group B ¹¹⁶ (imc DEVICEcore)

Configuring imc FAMOS

imc FAMOS Edition
Which function scope do you wish to install?

Please select here which FAMOS edition you wish to install. Please select the edition purchased.

- ☐ Trial version (30 days for free)
- ☐ Reader (for free)
- ☐ FAMOS Standard
- ☒ FAMOS Professional
- ☐ FAMOS Enterprise
- ☐ FAMOS Runtime

Configuration of the imc FAMOS installation

In this step, you configure the imc FAMOS installation. Select the editions you wish to install. Note that all editions except the *Reader*-edition require a license.

Follow the Assistant's instructions and select your preferred language for the help files and example files like projects, sequences and dialogs. Select any optional components which you wish to install along with imc FAMOS. Additionally, a folder for sample files is needed.



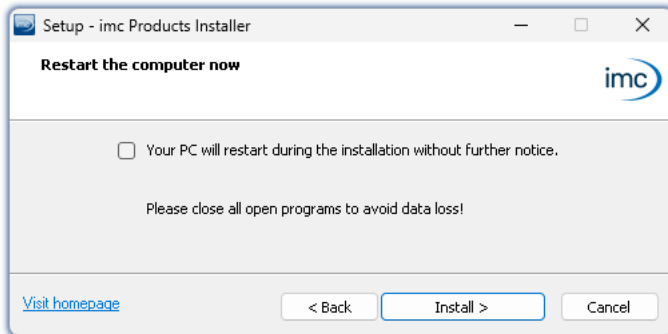
Reference

[imc FAMOS editions](#)

More detailed information on the imc FAMOS editions is provided in the imc FAMOS manual.

Once all products have been configured, the installation can start.

3.3.3 Starting the installation



Concluding the installation setup and starting installation

Before the installation, a **system reboot** is performed. For this reason, first **close any running programs** before continuing!

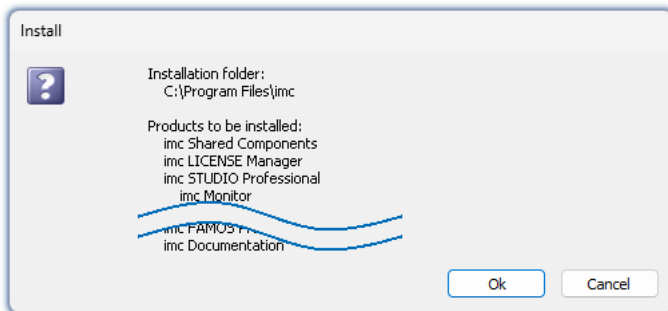
Confirm the reboot by means of the checkbox and then continue (button: "*Install*"). Initially, the required system components are installed. Subsequently, the PC reboots automatically.



Warning

Windows user account

After restarting, **be certain to log in with the same user account** with which you started the installation process. Using a different user account can cause failure of correct installation of the components.



List of components to install, following restart
(example)

After logging in, the actual installation of the products begins. You are presented with a list of which components are to be installed, which you can confirm by clicking on "*OK*".

The figure shows an example; the actual dialog may appear differently depending on your respective previous selections.

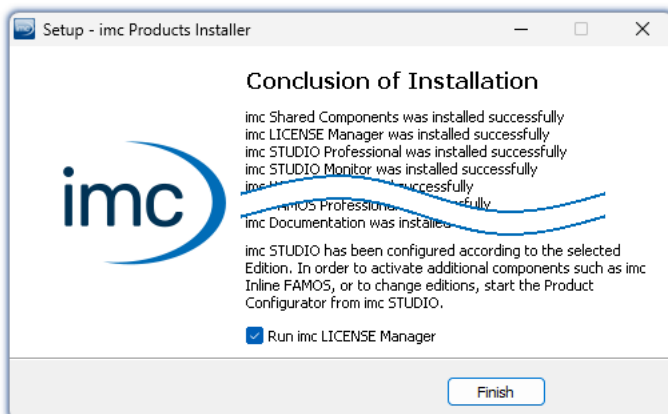
Next, installation of the individual imc products begins.



Note

Disabling the system start

While installation is running, the operating system is disabled from starting again, meaning that no other programs can be started.



Concluding installation

When installation is finished, the following message is posted.

Next, you can start imc LICENSE Manager in order to activate your license. It is also possible to start the imc LICENSE Manager later via the Start menu.

3.4 Product configuration / Licensing

Licensing (imc LICENSE Manager)

The software licenses are managed with "*imc LICENSE Manager*". After concluding installation, you can start imc LICENSE Manager directly in order to configure your license (e.g. by means of the Windows Start menu: group "*imc*" > "*imc LICENSE Manager*"). If imc STUDIO Monitor is started without an appropriate license having been configured, the product configuration program opens, from where it is also possible to start imc LICENSE Manager.

Follow the instructions of the imc LICENSE Manager. imc LICENSE Manager offers a separate documentation. Start imc LICENSE Manager and click on "*Help*".

Changing the product configuration

imc STUDIO Monitor is a preconfigured product. You can change the product configuration in order to adapt it to the license purchased. For instance, if you have purchased optional components. To do this, start imc STUDIO Monitor and click in the menu bar (top right) on the symbol . The product configuration assistant now opens. You can change the configuration to correspond to the licenses you have purchased. Follow the instructions and close the assistant.

After completing the configuration, you must close and restart imc STUDIO Monitor to apply the new settings.

3.5 Start

Start the software by means of the associated desktop icon or via the Start menu.



Further, as part of installation an **icon is set up on your desktop** - depending on your choice during [installation](#) ¹⁸.

If no icon was set up, then open the Windows Start menu. There you find the group "*imc*" and in it a link to start the product.

If [product configuration / licensing](#) ²⁷ is missing, the Product configuration starts automatic.

Splash screen

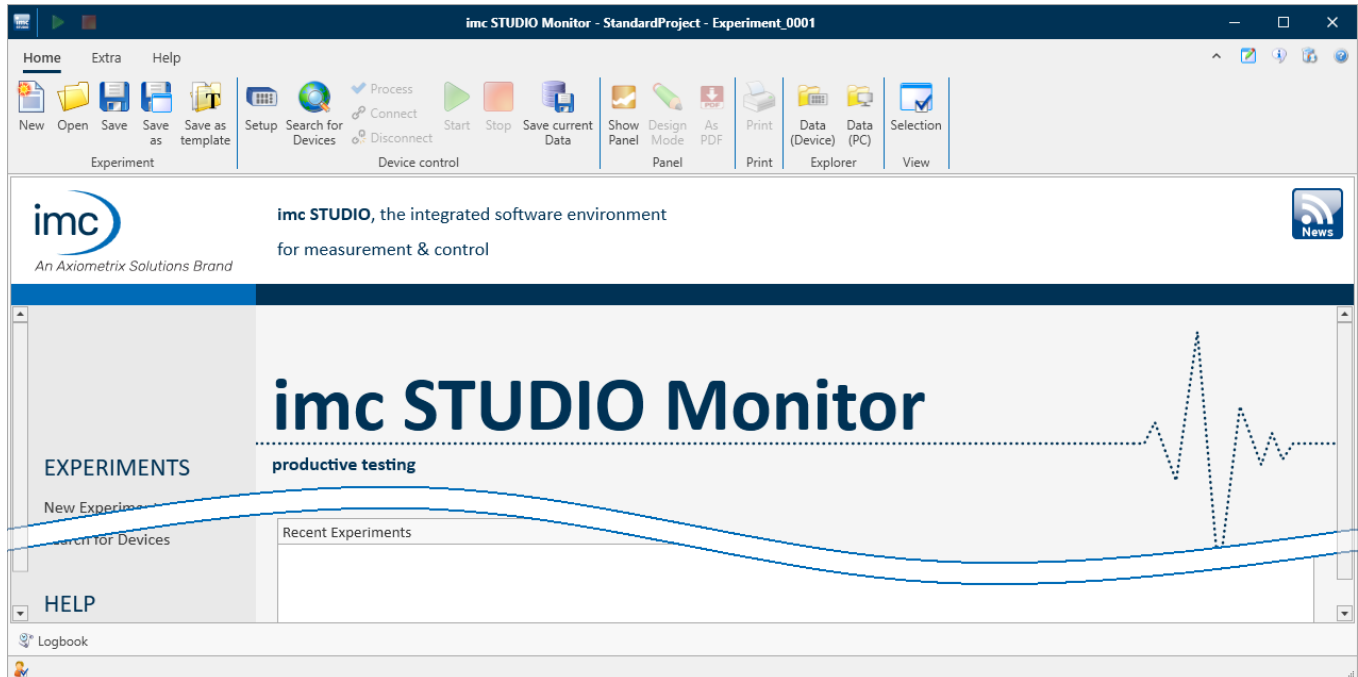


Next, a splash screen appears, indicating that software components are being loaded. Depending on the installation and the PC capacity, this could take some time.

Loading the software components after start

Homepage - Start page

By default the Homepage is shown when the software is started. The Homepage shows certain functions, depending on the particular product, as buttons. Click on the corresponding buttons, to select your desired functions. After the first installation, the "**Homepage**" may appear as shown in the example below:



Start page (example)

imc STUDIO starts with a simplified view

Along with installation of imc STUDIO Monitor, a number of **Views** are provided. Views in this sense are defined designs for the user interface's appearance.

imc STUDIO Monitor offers a wide variety of possibilities and functions, but the entire scope of these is not always needed for a single simple measurement.



Note

imc STUDIO Monitor starts with a simplified view

In order to make it easier to get familiar with the software, the program starts with a **simplified view** with limited choices: "**Standard**". All important functions for measurement and visualization of measured data are available. The menu ribbon is structured in such a way that proceeding from left to right accesses all of the [Main windows](#) ²⁹.

Changing from the simplified view to View: Complete

You can switch to a different view at any time, in order to obtain access to all of the existing functions. To do this, go first to the menu item "Extra" and in the drop-down list of views, select: "**Complete**".



Note

Notes on the descriptions and the screenshots

The following descriptions and screenshots always pertain to the view with the full scope of functions. Many of these functions are also found in the minimized view at another location.

3.5.1 Device Connection / Network / Firewall

To use an imc measurement device, imc STUDIO Monitor must set up a **connection via a network**. Details on the network settings are presented in the manual for the device software in the chapter "*Setting Up - Connect the device*". At this point in the instructions, it is assumed that the PC on which imc STUDIO Monitor is running has already been appropriately set up.

Firewall

The firewall used might **prevent** establishment of a **connection** between the operating software and the measurement device. If the software is unfamiliar to the firewall, then a corresponding prompt usually appears upon **accessing the network**. This happens, for example, after starting the software and upon first accessing the device (e.g. in a device search). Some firewall programs can also block access to system and hardware-components.

In all cases, **correct operation** is only possible if the **connection is not blocked**. The following programs are usually affected:

Program	Default path
imc STUDIO: imc.Studio.exe	C:\Program Files\imc\imc STUDIO...
imc DEVICES: imcDevices.exe	C:\Program Files (x86)\imc\imc DEVICES...
imc DEVICES Service: imcDevicesService.exe	C:\Program Files (x86)\imc\Shared

For operation on LAN, it is sufficient to **enable "Private networks"**. Enable this access type. You may wish to read the operating instructions for your firewall, and / or ask your administrator / IT-department.

Windows Defender Firewall: Unblocking is performed automatically upon installation.

3.5.2 Main Windows / Components



imc STUDIO provides the framework for various **components**. Components appear within imc STUDIO Monitor as **main windows**. A component can have one or more main windows.

The main windows appear in the Navigation pane.

On the left you see an example of the Navigation pane with the main windows **Homepage**, **Setup**, **Panel** (**Panel** is selected).

To jump to one of the main windows, click on the corresponding button in the Navigation pane.

[Navigation pane](#)⁶⁸: Select this link to go to further information on operation of the Navigation pane.

 **Note**

Notes on the use of the simplified view

- When you use the [simplified view](#)^[28], the Navigation pane is hidden. In this case, in order to go to the main windows Setup and Panel, use the menu ribbon.
- The other main windows can only be reached via the Navigation pane.
- If you need it, you can [show the Navigation pane](#)^[70], or change the [view](#)^[84].

Tool window

Most components have their own tool window (e.g. the Panel with the tool windows: Widgets and Data Browser). The tool windows are described in their respective sections.

 **Reference**

See also

[Tool window](#)^[69]: Select this link to go to further information on operating the tool windows and for a description of the tool windows belonging to the imc STUDIO Monitor frame.

3.5.3 Important settings

User and User groups

imc STUDIO Monitor has the ability to administer various **users** which are permitted to use the program in various ways. For details see the section "[User administration](#)"^[75].

Options

Before beginning to work with imc STUDIO Monitor, you can change essential settings, such as the **folder path where the experiments are saved**.

You can change the folder path in the [options](#)^[61] (under "*Project Management*" > "*HDD settings*"):

Ribbon	View
Extra > Options 	all

 **Note**

Reading and writing access rights are required

Please note that this setting will apply for every user of imc STUDIO Monitor. Each user must possess reading and writing access rights for the specified folder path.

3.5.4 Command line parameters

Opening an Experiment file

An [experiment file](#) ⁴³ (e.g. MyExperiment.imcstudio) can be opened directly in the Windows Explorer by double-clicking. This starts a new instance of imc STUDIO Monitor and opens the experiment file. The same procedure also applies for linkages to experiment files, for example files on the Desktop.

Experiments can be started with command line parameters. To do this, create a link to an experiment file and enter the desired parameter.

```
["imc STUDIO-Installation"] "Experiment-File" [/fullscreen] [/do[StartMeasurement]]
[/do[StartSequencer]]
```

["imc STUDIO-Installation"]	Open Experiment with a specific imc STUDIO installation (optional)
In order to open an experiment file with a particular installation of imc STUDIO (imc STUDIO Monitor), extend the link with the respective installation's name. For example: <pre>"C:\Program Files\imc\imc STUDIO 2023\imc.Monitor.exe" "imcDB://DB_Monitor\StandardProject\MyExperiment"</pre> The experiment file MyExperiment.imcstudio is started in the imc STUDIO installation (imc STUDIO Monitor).	

"Experiment file"	Experiment to be loaded
Mandatory entry. Opens the experiment entered with the imc STUDIO installation (imc STUDIO Monitor) which is set up as the default program under MS Windows. Mostly this is the last imc STUDIO version installed, or imc STUDIO Monitor/imc WAVE, if this component was installed along with it. One possibility is to specify the path in absolute terms: <pre>"C: \Users\Public\Documents\DB_Monitor\StandardProject\MyExperiment\config\MyExperiment.imc Studio"</pre> Another way is to specify it in reference to the name of the database and of the project (the appropriate database folder must be selected): <pre>"imcDB://DB_Monitor\StandardProject\MyExperiment"</pre>	

/fullscreen	Fullscreen mode (optional)
In order to start an experiment in fullscreen mode, the experiment file is opened with the parameter /fullscreen. <pre>"imcDB://DB_Monitor\StandardProject\MyExperiment" /fullscreen</pre> After opening the experiment file, the Panel is started in fullscreen mode, if the experiment contains at least one Panel page.	

/do[StartMeasurement]	Starting the measurement (optional)
In order to start the measurement automatically upon opening an experiment file, The experiment file is opened with the parameter /do[StartMeasurement]. <pre>"imcDB://DB_Monitor\StandardProject\MyExperiment" /do[StartMeasurement]</pre> or <pre>"imcDB://DB_Monitor\StandardProject\MyExperiment" /fullscreen /do[StartMeasurement]</pre>	

/do[StartSequencer]	Starting the Sequencer (optional)
In order to automatically start Sequencer upon opening an experiment file, the experiment file is opened with the parameter /do[StartSequencer]. <pre>"imcDB://DB_Monitor\StandardProject\MyExperiment" /do[StartSequencer]</pre> or <pre>"imcDB://DB_Monitor\StandardProject\MyExperiment" /fullscreen /do[StartSequencer]</pre>	

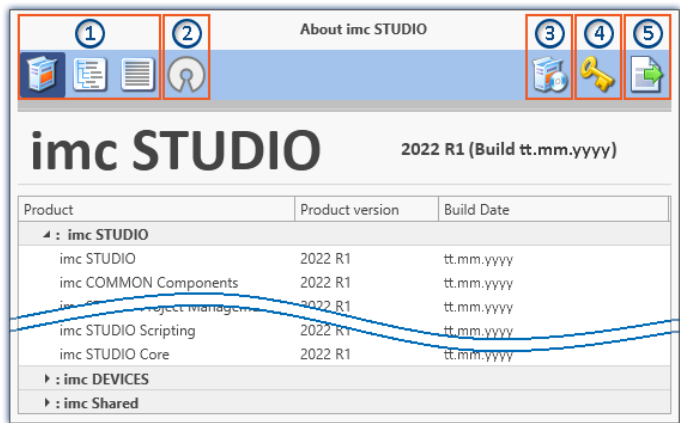
3.5.5 Error causes upon starting

Corrective actions for when the logbook opens with error messages directly upon starting imc STUDIO Monitor.

Message	Corrective action
Error while initializing the plug-in "imcDevices V2.x Adapter" ...	In the product configuration, the firmware imc DEVICES is activated. However, no appropriate firmware has been found or installed. By default all firmware products are activated. You can take the following corrective actions: - install the appropriate imc DEVICES-firmware. This is only necessary if you wish to use a device belonging to this firmware group. - deactivate the component "imc DEVICES V2.x Adapter" in the product configurator ²⁷.
The loaded Project-settings are from an older imc STUDIO version. If you save these settings, an older imc STUDIO version would not be able to read them correctly.	With every Version/Revision, new enhancements and other changes are introduced. In consequence of these changes, some saved projects may become no longer compatible with older versions. Make sure that no older version of imc STUDIO Monitor needs to access the database.
Could not connect to the Guardian process. Some functions (e.g. delete measurements) will not work.	The Guardian process is not running. Please start either the Guardian- or WatchDog-service.  For more info, see the section "Guardian" ⁹⁴

3.6 Info / Version information

imc STUDIO Monitor is an application for imc STUDIO. imc STUDIO is used as a base and consist of multiple components. In order to see which components are included in your package, click in the menu bar on the symbol . Then a pop-up window appears displaying the product names and details on the components:



Version information (example)

With the buttons on the left side ①, you can change the list's depth of detail and how it is sorted. Additionally, you will receive all copyright notes regarding "Open source software" (📄) ②.

With the buttons on the right side, you can start the assistant for the [product configuration](#) ²⁷ ③, start the [imc LICENSE Manager](#) ²⁷ ④ or export the product selection ⑤.

While performing the export, you can save the contents as a file in XML format. For inquiries to our tech support, it may be necessary to send it as an e-mail (see also section "[Tech support](#)" ⁶).

3.7 Information and tips

3.7.1 Changing languages and installing additional languages

Following installation, imc STUDIO Monitor is available in **English** and, if applicable, in the **language of the operating system**. Additional languages can be installed by selecting the option "**Install additional languages**" in the installation variant "[User-defined](#)".



Note

Observe operation system settings

- Please ensure that your operating system is set correctly to the display language. Some languages require modifications to be made accordingly. In particular, the "**System locale**" must be set correctly to correspond with the display language. Otherwise, with some languages there can be problems with the characters. Languages affected include Japanese, Chinese, Russian, ...

Changing the language

By default, the imc software will start in the same language as the Windows version installed. If this language is not supported, then the English program version appears.

The language can be changed, irrespective of the Windows version. Use the program "*imc Language Selector*". The program is found in the Start menu under the group "*imc*".



Note

Constraints

Only the imc software's texts are converted. Components which depend on the operating system's language setting are not affected.

Please only use one of the following two languages:

- Default: the operating system's language
- English

Any other selection may cause malfunctioning if the operating system and the imc programs use different languages.

Installing additional languages

If desired, all other supported languages can be installed later (no installation medium is required).

 **Note**

Which languages are be installed subsequently

In subsequent installation of languages, **all languages supported** are installed. There is **no selection** possibilities.

Step by step

- Administrator rights are required.
- Open the **installation folder** of imc STUDIO (e.g. "*C:\Program Files\imc\imc STUDIO...*"), e.g. with Windows Explorer or by means of the command line.
- There, open the folder "**Languages**"
- Run the file "**InstallLanguages.bat**".
- Wait until the script stops and posts the message "*Failures: 0*". This indicates that the installation has completed successfully.

 **Note**

Instruction notes

If installation was unsuccessful, the necessary user rights may be missing:

- If "*InstallLanguages.bat*" is called via the **command line**, run the **command line as Administrator** (context menu: "Run as Administrator")
- If "*InstallLanguages.bat*" is called via the **Windows Explorer**, run the **call as Administrator** (context menu: "Run as Administrator")

3.7.2 Recommended virus scanner settings

The presence of many channels can **burden the PC** if a virus protection program scans the data captured. This may cause a **jiggling display** and high demands on the processor.

We strongly recommend **removing imc STUDIO from the virus scan**. Most virus protection programs are able to classify individual programs as low-risk. To do this, please refer to the instructions for your virus scanner.

Low-risk	Default path
Add the programs imc.Studio.exe , imc.Monitor.exe and imc.WAVE.exe to the list of low-risk programs	C:\Program Files\imc\imc STUDIO...
Add the program imcDevices.exe to the list of low-risk programs	C:\Program Files (x86)\imc\imc DEVICES...
Add the database path along with all subfolders to the list of low-risk folders	C:\Users\Public\Documents\DB

Ensure that its data are not searched for viruses during reading or writing.

 **Reference**

Installation

See also the installation note: "[Notes on the security software](#)"

3.7.3 Enhanced installation

Installing projects

You can prepare projects which are automatically included with the installation. E.g., after the installation, special views and experiments will already be available.

- Create a project-export-file (".imcStudioExport") for the purpose. Make sure that the project is exported and not only the experiments (selection). Upon exporting, you can define what the file is to contain. E.g., only the project settings, or also the experiments.
- Save the file(s) on the installation medium in the following path:
..\Products\imc STUDIO\Projects

Any existing projects having the same name will be overwritten.

Produce an installation medium and run the installation as usual or install from the hard drive.

After the first launch of imc STUDIO, projects are imported into the database.



FAQ

Question	Answer
What happens if the project exists already?	All project settings will be overwritten.
What happens if the project has contents which do not exist in the project to be imported?	All content besides project settings remains intact. E.g. experiments, measured data, metadata, ...
What happens if the project has contents which already exist in the project to be reported?	These will be overwritten. As well, any experiments, measured data, metadata, ... are overwritten.
What happens if there are any other projects present?	These projects remain intact.
What all is imported from the project?	All project settings and any experiments, measured data, metadata, ... the projects contain

Adding your own personal programs

On the installation medium, in the folder "*Products*", create an extra folder for your own product. Save your installation file there. The Installer will now automatically detect the product and offer it for installation.

Optional: By means of an ini-file, you can cause your product to be the default selection in the Product-Selection list.

On the installation medium (on the same level as the folder "*Products*"), create an additional folder named "*Configuration*". There, create a file named "*Setup.ini*". Fill this file with the following text:

```
[SetupX]
SetupX1=<PRODUCT>
```

For <PRODUCT>, insert the name of the folder in which your installation file is located.

3.7.4 Unattended installation - Silent installation

For unattended installation ("*Silent installation*"), certain parameter files (.ini) are required. Create these files by means of a command line call.

```
Setup.exe /CREATEINIFILES="<TargetPatch>"
```

To do this, proceed as follows:

- Start the installer - e.g. "Installer_imc_STUDIO_2023_R2_2023-04-20.exe".
- The installer begins by unpacking the content of the installation medium. Select an appropriate target folder on your hard drive: e.g under "C:\DVD\".
- As soon as the unpacking process has concluded, configuration for the installation begins. Cancel this procedure!
- Start Windows' command prompt ("cmd", "command shell").
- Navigate to the appropriate file path and call the Setup with the following command line (in the example here, the folder is: "S:\DVD\"):

```
Setup.exe /CREATEINIFILES="C:\DVD\"
```

The frame setup is started and may be configured as desired. When the settings are finished being made, there is no restart, as there usually would be. Instead, at the specified target location, *.ini-files are created which have the following structure:

Main file

DVD/Configuration/Setup.ini

Product files

DVD/Products/\$Productname\$/Configuration/\$Filename\$.ini

z.B.

DVD/Products/imc STUDIO/Configuration/imc STUDIO.setup.ini

DVD/Products/imc FAMOS/Configuration/Setup_imcFamos.ini

The ini-files contain the settings which have been made.

Subsequently, the DVD folder can be copied to a data carrier and used for the unattended installation.

Calling the unattended installation

The installation process is called with command line parameters, where "/SILENT" causes the silent setup installation to start. The following is an example of a 64-bit system:

```
DVD/Setup.exe /Lang=en /DIR="C:\Program Files\imc" /SILENT
```

With the additional parameter "/UNINSTALL=ALL" all previous versions are removed.

If the switch is set to "NECESSARY" only the necessary version are de-installed. When setting the switch to "NONE", no version is removed.

In special cases a de-installation is needed, e.g. if you install the 5.2 R23 to a 5.2 R22. The 5.2 R22 needs to be de-installed.

**Note****Creating ini-files in a different folder**

You can also create the ini-files in a different folder. In this case, merge the two folders afterward (not recommended because prone to errors).

Please always create new ini-files each time

With any new version, it may occur that new keys are entered in the ini-files. For this reason, create new ini-files for any new version.

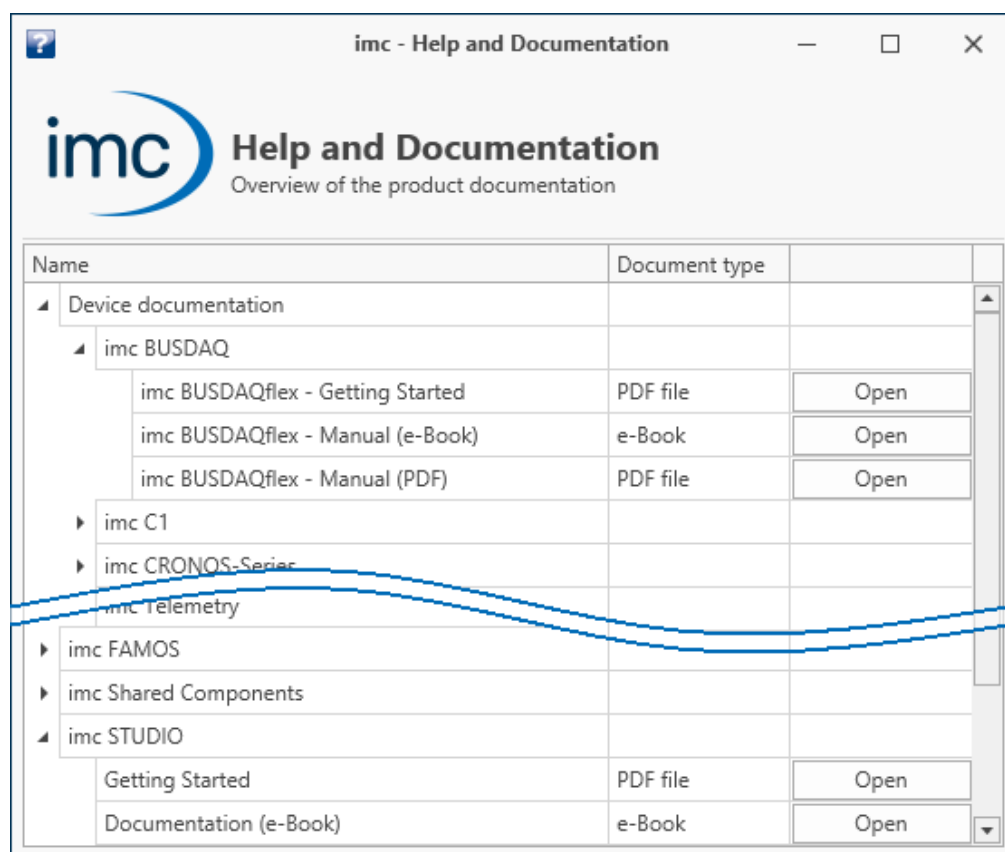
4 Documentation of devices - Help and Documentation

This manual is mainly about the device software. Separate documentation is available for each device model. Read this as well in order to obtain optimum utility from your device.

How to find the device manuals?

To do this, use the program "*imc Help and Documentation*" which helps to locate the documents. A few clicks of the mouse will open the desired document. This program is available

- via the Ribbon: "*Help*" > "*Additional Documents*",
- via the Startmenu under the group "*imc*" or
- on the installation medium ("*imc Help and Documentation.exe*")



"*imc Help and Documentation*" lists here the *imc C-SERIES* documents "*First Steps*" and "*Manual*" in both PDF and e-Book formats.



Note

Uninstalling the documentation

The documentation for older, already uninstalled software versions is not deleted automatically. Thus it may happen that multiple manual versions are listed. It is advisable to delete the old versions.

To delete documents manually, start the process of uninstalling the program: "*imc Help and Documentation*". A prompt appears to ask which documents to uninstall.

As long as not all of them are selected, only the selected documents are uninstalled. The program "*imc Help and Documentation*" is only included for uninstallation once the last documentation is uninstalled.

Thus, you will be able to uninstall old document versions without any difficulties.

5 Monitor - First steps

imc STUDIO Monitor makes it possible for teams to monitor the live data on multiple PCs. In doing so, each separate work station can create its own views and thus focus on particular monitoring tasks. During a running measurement, imc STUDIO Monitor can connect with one or more measurement devices, especially for the purposes of monitoring streaming measurement data, processing and saving them.

Configuration of the devices and start of the measurement are performed by separate device software: imc STUDIO or imc WAVE. imc STUDIO Monitor is exclusively for the purpose of configuring the device monitoring mechanism.

System requirements and limitations

Measurement device requirements

Configuration of multi-client monitoring of imc devices belonging to the groups:

- [firmware group A](#)  via firmware imc DEVICES

Accessing the device: at most, four imc STUDIO Monitor PCs can be directly connected with one device.

Reaction to changed device configuration: When the configuration is changed in imc STUDIO, the device must be re-selected in imc STUDIO Monitor.

Selecting a device

imc STUDIO Monitor can connect with imc devices as soon they were **prepared** or a **measurement is running**.

Selection of the measurement device is the first step.

- Open the **device configuration page**: "Setup".
- Go to the **tab**: "Devices" .

Here, all available devices are presented.

Note

Connecting with the device

The same conditions for a successful search apply as for imc STUDIO. If a device is not found, there are a variety of possible causes.

For more information on correct configuration of all network settings, see the imc STUDIO/imc WAVE manual: **Setting Up - Connect the device**.

Select your device:

- Click on the Checkbox-symbol (☐) in order **to select the device**.

Note

Firmware

In order to be able to connect with a device, the firmware version (imc DEVICES) must be installed which matches the device.

The firmware must not only be present on the imc STUDIO computer for configuring the device, but also on the imc STUDIO Monitor computer, in order for it to read from the device.

Configuring channels

Next, the channels to be used are configured:

- Go to the **tab**, e.g: "*Analog channels*"  (or "*Digital channels*" / "*Variables*").

Here, all available channels/variables are presented.

Most of the parameters shown here pertain only to the device control and in imc STUDIO Monitor they appear in gray font.

Relevant parameters appear under the tabs: "*Data transfer*"  and "*Curve properties*"  -> So here, configure:

- the "*transfer*" of data to the monitor PC,
- as well as the data storage,
- the "*Circular buffer time*" for the display, and
- the range specifications for the Widgets

Transfer to PC

Transfer of the channels to imc STUDIO Monitor can be activated on a per-channel basis. Thus it is possible to view individual channels without overburdening the network with all the channels.

For this purpose, imc STUDIO Monitor uses the channel-parameter: "*Transfer to PC*".

Name	Current value	Transfer to PC	Save (PC)
▼ Channel type: Analog inputs (Count=8)			
Channel_001	1014.2 mV	☐	☐
Channel_002	-26.552 mV	☒	☒
Channel_003	1.3734 mV	☒	☒
Channel_004	10.834 mV	☐	☐
Channel_005	4.8831 mV	☒	☒
Channel_006	3.9675 mV	☒	☒
Channel_007	9.1558 mV	☒	☒
Channel_008	-2.8993 mV	☐	☐

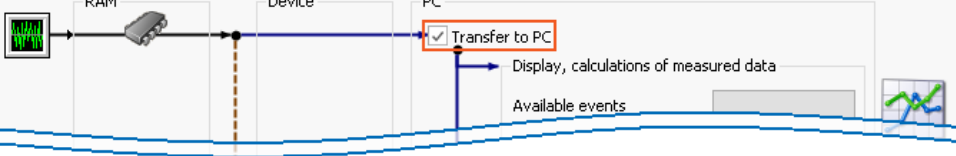
Channel definition

Range & Scaling

Filtering

Data transfer

Channel name Channel_002



Transfer to PC

In the example, channel_001 is not transferred while channel_002 records measurement data.



Note

Single value variables

Single value variables are always transferred.

Circular buffer time

All incoming data are kept in a circular buffer memory. The volume of this buffer memory can be set for each channel separately in the column "*Circular buffer time*".



Note

Circular buffer time

The circular buffer memory only applies to the display in the curve widow and does not affect storage of the measured data!

Beginning/concluding data recording

Start data recording by means of the Start-button ()

Stop data recording by means of the Stop-button ()

Recording of the data runs independently of the actual measurement. If the measurement is running on the device while recording of the data is running in imc STUDIO Monitor, the measured data are recorded. If, at a later time, measurement (on the device) is stopped, then consequently no measured data appear in imc STUDIO Monitor. Upon the next start of measurement, measured data immediately appear in imc STUDIO Monitor, as long as data recording continues there. In this case, the data of the last measurement are discarded.

Configuring display of measured data

You can observe the channels in the [curve window](#) ⁶⁴⁷, or use the other Widgets on the [Panel](#) ⁵⁸⁸.

Analysis and processing of measured data

Various tools for [subsequent processing and analysis](#) ²³⁷ of the measured data are available in conjunction with associated licenses.

Activating data storage

By default, [data storage](#) ²⁰⁰ is activated.

6 imc STUDIO Monitor (general)

imc STUDIO Monitor is an application which is constructed upon imc STUDIO. Many components of imc STUDIO are used. For this reason, some of the descriptions may make reference to imc STUDIO.

Which plug-ins are available depends on the product installation (order).

Chapter overview

Synopsis	Section
Data storage: How are data stored and loaded	• Experiments, Projects and the Database  43 • Ribbon "Project": Open / Save experiments  54 • What is saved where?  91
Navigation through the entire software	• Navigation Pane  68
Feedback from imc STUDIO Monitor - information, warnings and error reports	• Logbook  71
Limited operability due to disabled user privileges	• User administration and Access rights  75
The software user's interface is flexible. Saving and restoring of views	• Saving / Loading views  84
Support of process automation by means of replaceable text modules	• Placeholders  95

6.1 Experiments, projects and the database

This chapter describes the interdependencies among the "*database*", "*projects*", "*experiment templates*" and "*experiments*".

Experiment

The experiment includes a record of all settings needed for generating the measurement data as well as viewing and evaluating them. The actual measured data are saved with the "experiment" to which they belong.

You can create a variety of experiments to accomplish a corresponding variety of measurement tasks. imc STUDIO Monitor always works with exactly one experiment and all changes are saved to it.

Among others, the following settings are saved:

- the experiment file (filename extension: "*.imcStudio"),
- measurement files and metadata,
- various backup files and administration files

Among other things, the experiment file contains all the settings configured in the main windows and in the Setup-Assistants.

Each experiment has a unique name which is set when it is [saved](#)  49.

Measurement data

By default, the measured data are saved in the Experiment folder. They belong to the experiment. You specify the data structure in the Device-settings under "[Storage](#)"  180" (Setup page: "Devices" > "Storage").

Project

A project is primarily a collection of various experiments. The factory settings of imc STUDIO Monitor are configured in such a way that only one project is present and that you are aware of it to the minimum extent possible.

All of your experiments are saved in this project ("*StandardProject*").

Some options and configurations are not saved along with the experiment. For instance, they might be saved along with the project and apply to all experiments belonging to the project. In some cases, you can define what has to be saved where. E.g. when generating variables, you can change the validity scope and, for instance, save it not just for that experiment but for all experiments in the project.

Whatever is saved at the Project level (and thus not with the specific experiment) is correspondingly denoted at the locations affected. A project can include:

- one or more experiments
- one or more [experiment templates](#)^[52]
- project settings (e.g. views, user administration and project-events)

After the first start of imc STUDIO, a standard project is set up in which your experiments are saved.

Project view - working with multiple projects

When you activate "*Project view*", there is a project-based tree diagram in the [Open- and Save-dialogs](#)^[48]. Here, you can create additional projects and load experiments from other projects.

 **Note**

Creating a project resets variables

When you create a new project, the status of the experiment currently open is saved temporarily and reloaded later after the new project has been set up. In consequence, the values of the variables are reset to the condition they would have if the experiment were re-loaded.

In the process, the event "*Experiment_Loaded*" is not triggered. If you need this to happen for your experiment, please re-load it manually. If you save your experiment with the new project ("*Save As*"), something similar happens. Here, too, the variables are reset, but additionally the event "*Experiment_Loaded*" is triggered.

You can activate the "*Project view*" in the options (under "*Project Management*" > "*General options*"):

Ribbon	View
Extra > Options 	All

Option	Description
	To create more projects, you must activate the Project view . This option is deactivated by default. (Information on Projects ^[44]).
	There is an extended view available of some dialogs ^[47] . Multiple projects can be created and used.

[Project view activated](#)^[64]

 Only activate the **Project view** if you wish to use and manage multiple experiments in separate projects.

 If this function is deactivated, some dialogs are displayed in **simplified form** (e.g. "*Save experiment*" and "*Open experiment*" are **missing Project selection**, for instance).

Database

The database is the data repository for imc STUDIO Monitor. Here, the projects and their associated experiments are saved. The database does not have its own settings and configurations.

The folder path for the **database can be freely selected** (in the Options under: "*Project Management*" > "*HDD Settings*").

Ribbon	View
Extra > Options 	All
Option	Description
Database folder 	Here you set where the "Database" is to be saved. (Information on "Data management" )  The database is not user-dependent. Note that every user who is set up in the system has writing and reading rights for this path.

Database conversion

If the database structure has been changed, you are subsequently notified. One way this can happen is as a result of updating to a new version.

A **conversion-dialog** appears. You can convert the database or have it copied beforehand. After the conversion, the entire **database can no longer be used with the older version**.

Databases are not downward compatible.

In the upper region, the reason why the database does not fit with the current version is noted. E.g., the database may be too new or too old. In the list below, all databases found in the folder selected are listed. The "Status"-bar contains information on the database.

The following approaches are available:

Approach	Description
Select an existing database	Select the appropriate database and click on the button "<i>Apply</i>". If it is necessary to convert the database, an additional dialog appears. Here, the user is prompted to decide whether to use the database under a new name. If you choose: no: the existing database is converted. It can no longer be used with the older version. yes (recommended): the database is copied. Enter a name for the new database. Now you have two databases. You will have a backup copy and can continue to use the old database with the old version.
Create a new database	Click on " <i>Create new</i> ". Enter an appropriate name for the database. It will be created in the folder selected (parallel to any already existing databases).
Change the database's folder	Click on the button "..." next to where the folder is indicated. Select an appropriate folder.  Here, select a folder where the database folder is to be created later. Not the database itself. E.g. the default folder: "<i>C:\Users\Public\Documents</i>". The database will be created in this folder, e.g. "<i>DB</i>".

Experiment Template

See: "[Experiment Templates](#)" 

Creating one's own files on an experiment - in the folder "Meta"

You can **create your own files on the experiment**, such as metadata, imc FAMOS-sequences, info-files. To do this, use the **folder: "Meta"** in the experiment path. If you save or export the experiment under a different name, all files from the folder "Meta" come along with it.

The folder "Meta" is now ignored in all components.

- The content is not recognized as a measurement although containing measured data. This means that there is no entry in the Data Browser.
- Interval saving does not delete the folder. When the number of intervals is limited, the folders of measured data are deleted in succession once the specified count has been reached. The folder "Meta" is ignored, even when the folder contains measured data.



Reference

See also

- Structure of and files in the database: "[Data Management](#)"⁹⁰
- [What is saved where?](#)⁹¹

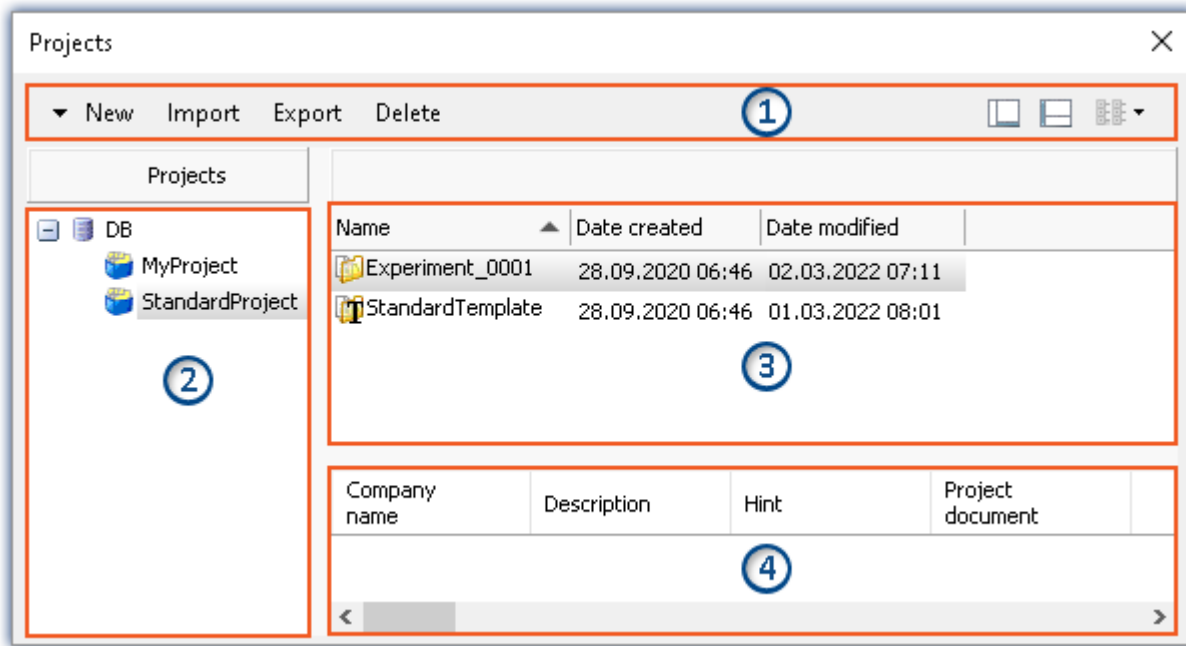
6.1.1 Dialogs: Project and Experiment

The Project Management dialogs are described below:

- with "Project view" and
- with "Experiment templates".

By default, these [options](#)⁶⁴ are deactivated, in consequence of which not all functions are available. Options-dependent functions are mentioned explicitly below.

The dialogs following the functions "Manage Project", "New Experiment", "Save As" and "Open" are structured similarly. The dialog is described below on the basis of "Project Management" as an example. Not all dialogs feature the complete scope of functionality.



Example: Manage project

The dialog is subdivided into four regions (from top to bottom):

1. Menu bar
2. List of all projects
3. List of all experiments and experiment templates in the selected project
4. Linked metadata information belonging to the selected experiment

By default, the regions 1 and 3 are displayed; the others can be activated if desired.

Region 1: Menu bar

Function	Description
New	Create new project: Creates a new project in the selected database (a database must be selected). Create new experiment template: Creates a new experiment template⁵² (a project must be selected).

Function	Description
Import  51	Imports elements (projects, experiments and or experiment templates) from a file in the selected entry. The file can contain multiple elements. You can select either all or individual elements for import. Additionally, the target name can be changed. Be sure to first select the target correctly. You can only import a project if the database is selected. You can only import experiments and experiment templates if a project is selected. You can only import projects if the Project view is activated.
Export  50	Exports the selected entries to a file. Whole projects and individual/multiple experiments and experiment templates can be exported to a file.  Manually created folders are not included with the export. The exception is a folder in the experiment path named: "Meta"  46.
Delete	Deletes the selected entries. Complete projects and individual experiments/experiment templates can be deleted. If you select delete an experiment with saved measurement data, a prompt appears for confirmation that you wish to delete the data.
 / 	Show/Hide Region 2 (list of all projects)
 / 	Show/Hide Region 4 (metadata information)

Regions 2 and 3: List of all projects / experiments / experiment templates

In these two regions, the projects/the experiment and experiment templates are listed. When you select a project, all elements of the project selected are displayed in the region at right.

Region 2 is only displayed if the project view is activated and the region is shown (via the menu button: ).

Region 2 only shows the experiment templates if these are activated.

Region 4: Linked metadata information

The metadata saved with the selected experiment are displayed in this region. When an experiment is saved, the metadata can be saved automatically along with it. In the [Options](#)  63 "Metadata" > "Experiment - Metadata" > "Setup page", you can select which source to use for the metadata.

Only shown if the region is shown (via the menu button: ).

6.1.2 Creating/Saving/Copying an experiment

Each experiment has an identifying name which is specified when it is created/saved. Each name may be used only once per project.

When you create an experiment (ribbon menu item: "Home" or "Project" > "New"), or when you save an experiment under a new name, you are asked for a name. In this case, enter a unique name.

If the "[Project view](#)"⁶⁴ is activated, you can additionally select the **target project**.

Action	Description
New Experiment	When you create a new experiment, it is generated from an experiment template (see "Experiment template"⁵²). • If experiment templates are displayed⁵², select an experiment template. • If they are not displayed, the preferred experiment template⁵³ is used automatically (at default: "StandardTemplate"). All changes since the last save are discarded if a new experiment is created.
Save experiment as	When you save an experiment under a new name, a new experiment having the current settings is created. • Measurement data from the existing experiment are not transferred along to the new experiment (exception: if the experiment was previously never saved; in that case a prompt appears to ask whether to include these measured data). • Files from the folder "Meta"⁴⁶ are copied and are available in the new experiment.
Copying an experiment to a different project	You can copy experiments to a different project by means either of Drag&Drop or of the context menu. This means that all files located in the Experiment-folder are included: Measured data, saved measurement settings, parameter set files or user-made folders of metadata.

6.1.3 Exporting and importing experiments and projects

Using the project- and experiment-dialogs, you can export/import complete projects along with associated experiments and measurement data. On the other side of scale, you can also export/import only the project settings, or individual experiments.

Exporting

The elements selected are exported to a file.

Ribbon	View
Start > Open (📁)	All
Start > Save as (💾)	All
Project > Manage Project (📁✎)	Complete
Project > Open (📁)	Complete
Project > Save as (💾)	Complete

Procedure: Select the elements you wish to export, e.g. two experiments. Click on "Export".
A dialog appears in which you can select what to package in the "imcStudioExport-file". Confirm your selection and select an appropriate destination.

Selection	Description
Project-settings	(Only when exporting projects) All projects selected are exported
Experiment-settings	Project-export: All experiments saved with the project are exported
	Experiment-export: All experiments selected are exported
Data	Among other items, the experiment's measurement data are included with the export (caution: the file can become very large)

 **Note**

Metadata

Manually created folders are not included with the export. The exception is a folder in the experiment path named: "Meta".

Importing

You can import previously exported projects or experiments by means of the project-/experiment-dialog.

Ribbon	View
Start > Open (📁)	All
Project > Manage Project (🔧)	Complete
Project > Open (📁)	Complete

Procedure: First select the location to which to import the content. Next, click on the "Import"-button and select the file desired.

The "imcStudioExport-file" may contain multiple elements. You can select either all or only individual elements for import. Additionally, the target name can be edited.

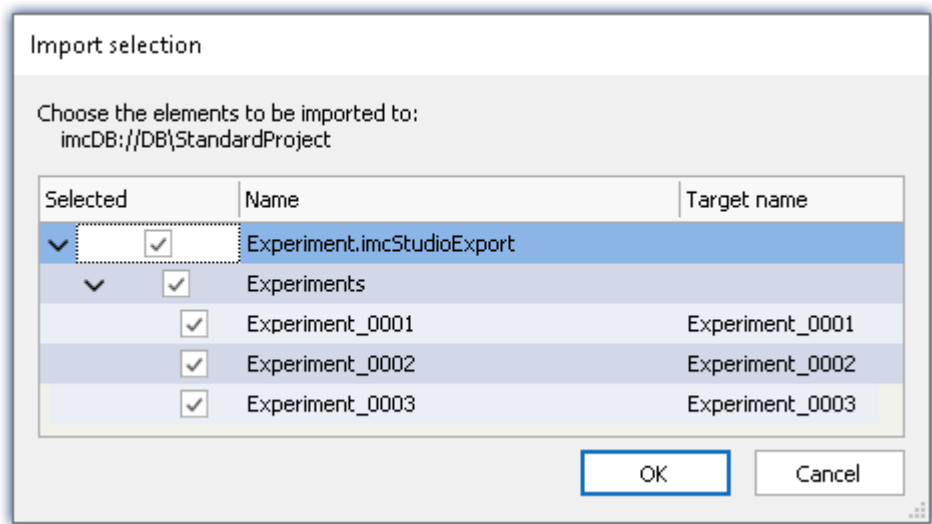
In the first dialog, select the main elements to import (which projects or which experiments). When importing projects, the individual projects are shown; when importing experiments, the various experiments are shown. In the column "Target name" you can edit the name. (Note: the dialog only appears when it contains more than one element, or when an element is renamed)

Be sure that you previously select the target correctly.

You can import a project if the database is selected.

Experiments, if a project is selected.

Notes: You can only import projects if the Project-view is activated.



Import multiple experiments from a single file

Confirm your selection.

Subsequently, another dialog appears (as for the Export procedure), where you can set which additional elements to import from the file.

Selection	Description
Project-settings	(Only when importing projects) All projects selected are imported
Experiment-settings	Project-import: All experiments saved with the project are imported Experiment-import: All experiments are imported from the file
Data	Among other items, the experiment's measurement data are included with the import

FAQ

What happens when importing existing elements?

You can import project-settings and experiment-settings without deleting the underlying elements. E.g., you can swap out the project. The underlying experiments remain intact.

Thus, the project can be modified at a developing PC and then imported to the test station.

Only elements having the same name are overwritten.

6.1.4 Experiment templates

When you create a new experiment, it is generated from an experiment template. The new experiment contains all properties of the selected template. In templates, all settings which are saved will be saved also in an experiment.

After the first installation or after creating a new project, the project contains an "empty" experiment template.

Making experiment templates visible

You can activate the "Experiment templates" in the options (under "Project Management" > "General options"):

Ribbon	View
Extra > Options ()	All
Option	Description
Show experiment template 	In order to be able to use experiment templates, you must previously make the experiment templates visible . This option is deactivated by default. (Information on Experiment templates )
	An extended view of certain dialogs  is available. Experiment templates can be created and used.
	Only activate this function if you wish to create multiple similar experiments.
	 If this function is deactivated, some dialogs are displayed in simplified form (e.g. "New experiment" and "Edit project" appear without experiment template selection).

Creating experiment templates

Ensure that experiment templates are visible. You can generate experiment templates from a variety of sources:

Source	Description
From present settings	The configuration currently set is used for the template. (see also " Save as template " )
From existing experiment	Select an experiment from which to generate the template.
From default settings	An empty template is created.
Click on the button: "Manage Projects"	
Menu ribbon	View
Project > Manage Project ()	Complete

- Select the project (if the projects are not displayed, click on the white background, in order to avoid selecting an experiment)

- In the dialog, click on the menu item: "New" > "Add experiment template"
- Select a source

The experiment template is created from the source and is available when [creating an experiment](#)  49.

Preferred experiment template

You can **mark** an experiment template **as preferred** (template context menu > "Set as preferred experiment template"). If the experiment templates are no longer displayed, the new preferred experiment template is automatically used when a new experiment is created.

6.2 Ribbon

6.2.1 Project menu

The "Project" menu is available in every plug-in.

Some of the dialogs offer an [extended view](#)^[64]. By default, the dialogs are displayed without "project view" and "experiment templates".

Project

Menu item	Description
 Manage Projects ^[47]	Managing projects and experiment templates
 Save Project	Saves the current project

Experiment

Menu item	Description
 New ^[49]	Creates a new experiment
 Open	Opens an existing experiment
 Save	Saves the current experiment
 Save as ^[49]	Saves the current experiment under a new name. Additionally, the experiment is reset. The variables are assigned the respective initial value (e.g. device variables = "0", or user-defined variables are assigned the respective initial value specified). The event "Experiment_Loaded" is triggered.
 Save as template	When you click on the button, a new template ^[52] is generated which the system then uses automatically (keyword " Preferred experiment template ^[53] "). (from the edition "imc STUDIO Professional")

Im-/Export

Menu item	Description
 Import / Export ^[55]	In this dialog you can import and export various components.

Current Measurement

Menu item	Description
 Save current Data	In addition to the optional automatic data saving, you can also perform targeted saving of the data currently in the PC (Current measurement) either during or subsequent to the measurement.
 Export current Data	See also Setup pages - Configuring device: "Storage options and directory structure" > "Controlling Data Storage" > " Targeted Data Saving, or Saving Subsequent to Measurement ^[217] "

 Note
Without imc STUDIO Project Management

In the product configuration, the component "**imc STUDIO Project Management**" can be deactivated. If "**imc STUDIO Project Management**" is deactivated, a number of limitations apply.

Among others, there is no longer any database. The dialogs for saving and opening experiments match the standardized "Save As"- or "Open"-dialogs. Experiments can be saved at any desired location. The saved measurement data are stored in the designated experiment folder.

The functions of the Project Management are listed in the technical data sheet.

The description of the Project- and Experiment-dialogs in the following pages pertain to imc STUDIO with activated Project Management.

Changes in the menu without imc STUDIO Project Management:**Project - without Project Management**

Menu item	Description
 Save Project	Saves the current project. In this case, all "Project"-settings are saved as "Application"-settings and apply for all experiments.

Experiment - without Project Management

Menu item	Description
 New experiment	Creates a new experiment. The storage location is first determined by means of "Save experiment/Save experiment as". The device currently selected remains selected. However, the device's configuration will be reset.
 Open experiment	Opens an existing experiment. The experiment can be saved anywhere in the file system.
 Save experiment	Saves the current experiment at the storage location previously set using "Save experiment as"; if no storage location had been determined yet, "Save experiment as" is run automatically.
 Save experiment as	Saves the current experiment under an arbitrary name at any location in the file system

6.2.1.1 Import / Export

In this dialog you can import and export various components.

Select the ribbon menu item *Project > Import / Export*.

Option	Description
Parameter set	Enables loading and exporting of values (data pool) and settings (device configuration). Select the desired settings/values. Details are provided in the documentation on the command: " Import parameters " and " Export parameters ".
Views, metadata columns, (sensors,) ...	Exports views, metacolumns, ... and sensors set up in imc STUDIO to a file. Or conversely, import views and metadata columns, ... from a file. You can adapt the imc STUDIO user interface, for instance by altering the Setup pages or the tool window. These settings can be saved by means of exporting. in this way, views can be transferred to other projects/applications/PCs.

Option	Description
Supplemental files	Supplemental files can be import and exported: • Characteristic curves and filter data for imc Online FAMOS / imc Inline FAMOS (*.dat) • imc Online FAMOS source code (*.ofa) • Synthesizer directory structures (*.dat) • Messaging configurations (e-mail, SMS, UDP, ...) (*.msg) Export: Shows the dialog to manage supplemental files¹²⁰⁾, in which you can export in the experiment existing supplemental files. See the description: "Setup" > "Ribbon" > "Configuration" > "Supplemental Files"¹²⁰⁾.
Setup table description	Imports an exported Setup table description.
Setup column descriptions	Imports an exported Setup column description to an existing table description.

6.2.2 Edit menu

The effects of some functions depend on the current main window.

Clipboard

Menu item	Description
 Cut	Removes the selection (the selected elements / the highlighted text) to the Clipboard.
 Copy	Copies the selection (the selected elements / the highlighted text) to the Clipboard.
 Paste	Inserts the Clipboard contents at the position highlighted.

Edit

Menu item	Description
 Undo	Reverses the effect of the last change to the current window.
 Redo	Restores the previously reversed change to the current window.
 Delete	Deletes the selection (e.g. the source text, the Panel-page, the Widgets selected or the selected Automation-element).
 Select all	Panel: Selects all of the page's Widgets.
 Aktualisieren	Setup: Metadata columns are updated. This affects file which, for instance, are accessed via a link (such as PDF-files). These are re-loaded if they have changed in the background.

Search

Menu item	Description
 Find	Find specific text in the current window.
 Find and Replace	Find and replace specific text in the current window.

Print

Menu item	Description
 Print	Panel: Opens the dialog for printing the Panel pages.
 Print Preview	Panel: Opens a print preview. Here you can see the possible appearance of printing results. This can be helpful when Widgets such as the curve window are used, which have their own color settings for the printout.

6.2.3 View menu

Views

Menu item	Description
 Save View  85	Saves current view Overwrites the used view in the project settings.
 Save View as  85	Saves current view in the project settings under a new name freely specified by the user, e.g. "Test Drive 1".
 View selection  84	Loads view from the project settings.
 Delete View  85	Deletes a saved view from the project settings.
 Restore  85	Restores views from the factory settings.

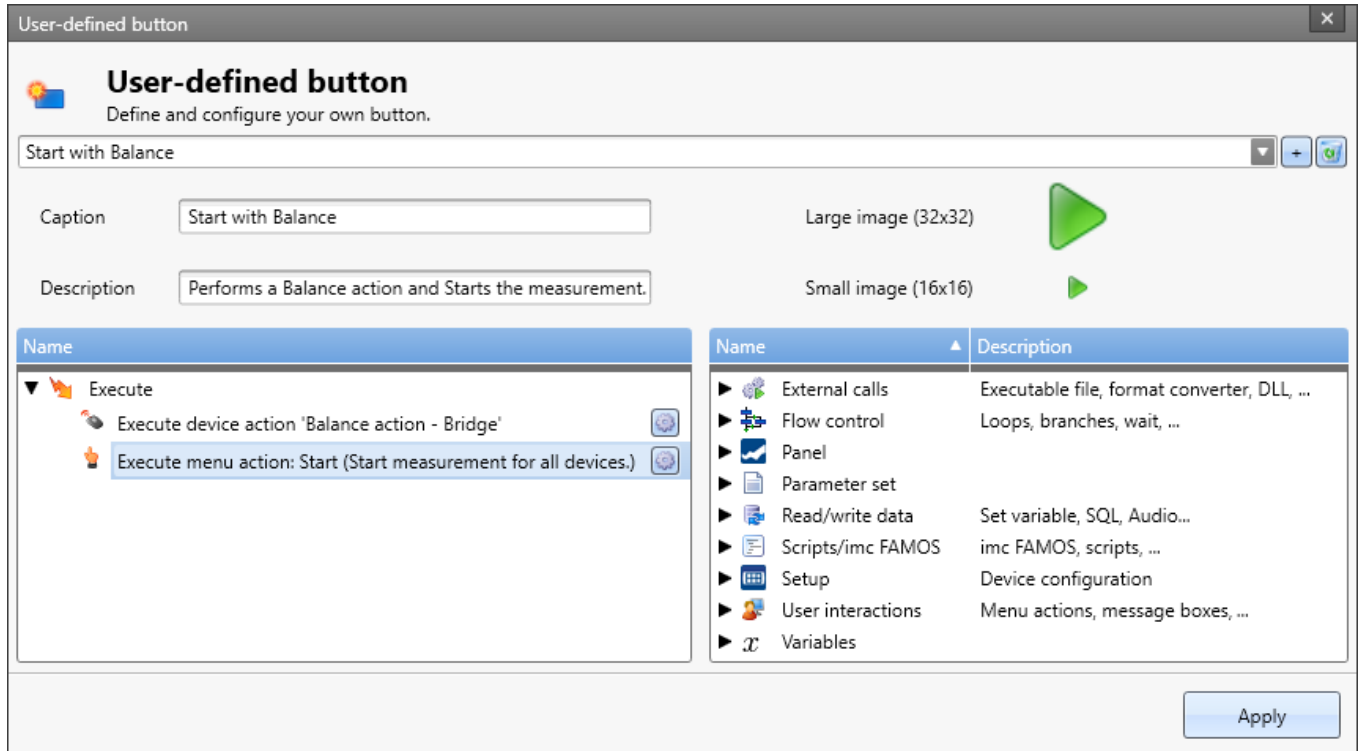
Layout

Menu item	Description
 Reset Window Arrangement	This function sets the currently open main window's GUI back to its last saved condition. This affects the window arrangements: • the position and size of the imc STUDIO window • the position and size of the tool window
 Customize Ribbon Menu  88	By means of this function the menu ribbon can be customized.
 User-defined buttons  59	Creates user-defined buttons which cause commands to be carried out.
 Choose Tool Windows  70	Opens the dialog: " <i>Tool Windows Chooser</i> " in order to show/hide the tool windows.

6.2.3.1 User-defined buttons

You can create your own personal buttons for the menu ribbon. Certain commands are associated with each button, which are run when the respective button is pressed.

The pop-down list contains all user-defined buttons available. The "Caption" of the respective button is used for identifying the button in this list. The list can be used to open existing buttons and to reconfigure/delete them.



*Example of a user-defined button:
this one performs a channel balance and then starts measurement*

Adding a new button

Click on the "+"-symbol to create a new button.

Configure the display in the button's menu ribbon:

- Caption: the displayed caption in the menu ribbon
- Description: the information in the button's tool-tip
- Image: the icon in the menu ribbon for the large and the small variants

Command configuration

Below, there are two areas:

- At left, the commands to be run
- At right, the available commands

Use the Drag&Drop technique to add commands to the list at left and configure the commands accordingly.

Deleting a command

If you wish to delete a command from the list, select it and click on the <Del>-key.

Adding a button to the menu ribbon

Click on the button "Apply". In consequence, the dialog for modifying the menu ribbon appears, but its scope is limited to the user-defined buttons. Add the button at the desired location in the menu ribbon. Information on this topic is presented in the chapter: "[Customize ribbon](#)"^[88].



Note

Saving the button's configuration

The **configuration of the buttons** is [saved](#)^[91] with the respective **project**. Thus, the button is available in all of the project's experiments.

The configuration is **only saved** whenever the **project is saved**.

The display in the button's menu ribbon is saved with the view.

6.2.4 Extra menu

The content of the menu "Extra" depends on what plug-ins are included.

Options

Menu item	Description
 Options ^[61]	General options

User Administration

Menu item	Description
 Login ^[80]	Logs in a user
 Logout ^[80]	Logs a user out
 Administration ^[76]	Sets up a user and assigns them to user groups
 User access rights ^[83]	Assigns rights to user groups
 Startup behavior ^[79]	For setting imc STUDIO's startup behavior

Sequencer

Menu item	Description
 Start (Sequencer)	Starts the Sequencer. For more info see the description of the plug-in Sequencer ^[946]
 Stop (Sequencer)	Stops the Sequencer. For more info see the description of the plug-in Sequencer ^[946]

6.2.4.1 Options

In this dialog you can make fundamental settings for the individual components.

Where are the options saved?

The individual options each have different storage locations. Some are saved with the project, others belong to the entire application (the imc STUDIO / imc STUDIO Monitor installation). A diskette-symbol indicates the storage location:

Scope	Description
 Application option	Options which apply for the imc STUDIO / imc STUDIO Monitor installation. They apply for all projects and experiments belonging to the selected database.
 Project option	Options applying to all experiments belonging to the current project.
 Experiment option	Options that apply to the current experiment.

Resetting options

By means of the selection list (at the bottom left), it is possible to reset either all options or individual pages.

Selection	Description
Reset	Resets the currently opened page to its last state . I.e., back to the settings which applied at the time the dialog was opened.
Reset All	Resets the options for all pages to their last state . I.e., back to the settings which applied at the time the dialog was opened.
Default	Resets the currently opened page to the factory settings . I.e., back to the settings which applied upon installation of the system.
Default All	Resets the options for all pages to factory settings . I.e., back to the settings which applied upon installation of the system.

Commands

E-mail options

Default values for the Mail-command. As long as no other settings are made in the corresponding command, these default values are used.

Please obtain the necessary settings values from your E-Mail provider.

Options - Mail options	Description
Mail server (SMTP)	Outgoing mail server of the E-Mail provider used.
Password	Password for authentication.
Port	SMTP server port used.
Postfix	This postfix will be appended to every e-mail.
Sender name	The name of the sender of the e-mail (your name).
Sender address	Sender address (your e-mail address).
User	User name for logging in.

Data Browser

General options

Options	Description
Grouping by categories	The variables are grouped in the Data Browser according to their respective categories (e.g. "Analog inputs", "User-defined variables"). This can be deactivated in order to sort the variables by their names. You can create your own groups by naming the variables according to the following syntax: "Group.Name". For instance, if there are two channels: MeasPoint1.Temperature and MeasPoint1.Voltage, these variables are automatically grouped in the group: "MeasPoint1".
Save the assignment of the measurement number	If a measurement number is assigned to a measurement, the assignment is saved with the experiment.

General options

Default dialog options

Here it is possible to specify default answers for dialog queries. If a value other than "Show dialog" is selected, the dialog is not shown and instead the selected answer is used.

Storage location measurement data

Options	Description
Custom base path for measurement data storage	The base directory path for storage of measured data represents the default filepath for experiments. In this folder, all measurements are saved (according to the structure specified in "Folder naming").  See also "Setup pages - Configuring device": "Storage options and directory structure" > "Controlling Data Storage" > "Customized storage location for measurement data"^[213].
Custom folder naming for measurement data storage	This specifies the structure below the base path. For instance, folders are set up here according to the settings for the directory structure ^[211].  See also "Setup pages - Configuring device": "Storage options and directory structure" > "Controlling Data Storage" > "Customized storage location for measurement data"^[213].

Logbook

Logbook

Options - Logbook	Description
Delete Logbook-files [days]	After the specified number of days elapse, the logbook files are deleted automatically. The deletion verification test is performed upon starting the program and at 0:00 (12 AM).
Default settings for the filters in the logbook tool window	Category of messages for which the filter presets are to be made.
Open logbook in response to a report of category:	For this category, the logbook is opened and attached.

Metadata

General Options

Options - Experiment - Metadata	Description
Export upon saving	Activates/deactivates automated export of the selected Setup page to the experiment. The export is performed whenever the experiment is saved. For example, in the project management dialogs (e.g. Open Experiment), the parameters saved for the experiment can be displayed (see: " Dialogs: Project and Experiment in Region 5 " ⁴⁸).
Setup page	Here you define what Setup page to export when saving the experiment.

Panel

General Options

Options - Options	Description
Multi-lingual text input	Enables textual input for computers with other language settings. For example, in this way, Panel pages can be pre-configured for different language settings. If the language set is configured for the respective text, it is displayed. Otherwise, the default language: "English" is displayed.

Navigation

Options - Options	Description
Always link new Widgets to Navigation bar	The navigation bar currently has no function.
Navigation through all Panel pages	The navigation bar currently has no function.
Post-processing mode as default	The navigation bar currently has no function.
Widgets navigation mode	The navigation bar currently has no function.

Options - View	Description
Datacut tools	The navigation bar currently has no function.
Post-processing tool	The navigation bar currently has no function.
Time controls	The navigation bar currently has no function.

Panel Widgets

Options - Widget Configuration	Description
Refresh rate of newly created Widgets	Specifies the refresh rate of Widgets which are newly created. If a Widget is created on a Panel page, it is assigned the refresh rate set here.

Project Management

General Options

Options - General	Description
Project view activated	To create more projects, you must activate the Project view. This option is deactivated by default. (Information on Projects^[44]). There is an extended view available of some dialogs^[47]. Multiple projects can be created and used. Only activate the Project view if you wish to use and manage multiple experiments in separate projects.  If this function is deactivated, some dialogs are displayed in simplified form (e.g. "Save experiment" and "Open experiment" are missing Project selection, for instance).
Show experiment template	In order to be able to use experiment templates, you must previously make the experiment templates visible. This option is deactivated by default. (Information on Experiment templates^[45]). An extended view of certain dialogs^[47] is available. Experiment templates can be created and used. Only activate this function if you wish to create multiple similar experiments.  If this function is deactivated, some dialogs are displayed in simplified form (e.g. "New experiment" and "Edit project" appear without experiment template selection).

HDD Settings

Options - General	Description
Database folder	Here you set where the "Database" is to be saved. (Information on "Data management"^[90])  The database is not user-dependent. Note that every user who is set up in the system has writing and reading rights for this path.

Sequencer

General options

Optionen - General	Description
Reduce logbook entries	Reduces the number of logbook entries produced by the Sequencer. The following messages will be suppressed and not be reported to the logbook window or the logbook file: Starting/Stopping command

Setup

General

Options - Common	Description
Data storage state after preparation	The state of the data storage after preparation (prepare or reconfiguration) of the devices has been performed. As before preparation: The state does not change in consequence of preparation. Data saving interrupted: After preparation, the state is set to: "Data storage suspend". Data storage must be resumed as soon as the incoming measured data need to be saved. Recording data: After preparation, the state is always set to: "Data storage running".  This option does not replace activation of storage for each channel. Instead, this option regulates the condition of "Suspend and Resume data storage".  Once the button has been pressed, preparation is only performed if any change has been made.
Parameter set export - Expanded mode	If set, you can control which columns will be exported upon parameterset export.
Options - First selected device	Description
Append device names to the channel names	If this option is activated, when a device is selected, the device name is always appended to the channel name, even when the first device is selected. If the option is deactivated, the device name is only appended as of the second device.  This setting only takes effect the next time a device is selected, unless any device was selected previously.

Traceability of data

Options - Traceability	Description
Traceability of data	Groups of Setup-parameters which are saved as metadata in the Channel Properties. They can be read out again at a later point in time. The channel properties make reconstruction of the measurement's configuration possible.
	See also: "Setup - Device (general)" > "Metadata and Comments on the measurement" > "Saving metadata with the channel - Traceability of data"

Variables

Datapool

Settings for memory use by the imc STUDIO Monitor Datapool.

Options - Speicher Optionen (PC)	Description
Split up pretrigger data into intervals	If the pretrigger's measured data are located in time across interval boundaries, it is possible to treat the measured data as follows: • deactivated: The pretrigger's data all are placed in the first interval (the interval in which the trigger fires) (<i>default setting</i>). • activated: The pretrigger's data are cut correctly. In this way, interval folders can be created subsequently.

Save/Export current Data

In addition to the optional automatic data saving, you can also perform targeted saving of the data currently in the PC (Current measurement) either during or subsequent to the measurement.



Reference

See also Setup pages - Configuring device: "Storage options and directory structure" > "Controlling Data Storage" > "[Targeted Data Saving, or Saving Subsequent to Measurement](#)".

Save current Data - Variable export

Define beforehand which file types are to be used for the export of the variables. The selection applies to:

- the command "[Export variable](#)" and
- the "[Export current data](#)" menu action

The following applies for the command "Export variable": This is only a preset which is set for the command when the variable is selected. You can still modify the file type in the command later. The system will not adopt any file type in the command configuration which the variable does not support.

User-defined variables

These settings are used automatically upon import, conversion, or when setting user-defined variables.

Options - Automatic prefix on variable name according to scope	Description
Scopes (Experiment, Persistent, Project, Sequencer, Temporary)	If true, all variables of the scope will receive a prefix to their name upon creation or conversion.

Measurement Administration

Options - Variables	Description
Access to stored measurements	If this option is activated, saved measurements in the Data Browser are displayed.

6.2.5 Help menu

Internet

Menu item	Description
 imc Website	Opens the imc website's Home page. Here, the user is provided with a quick way to contact imc , for example.

Activation

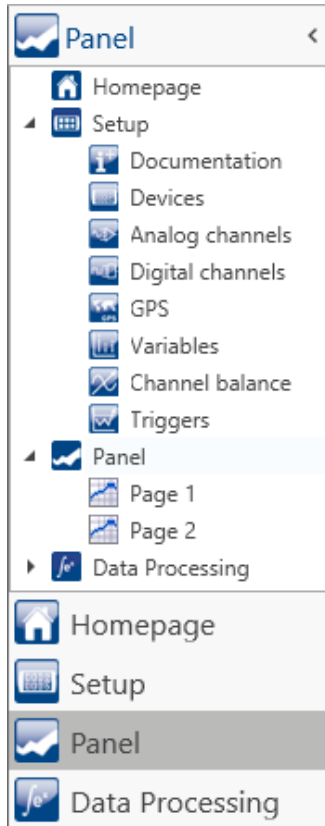
Menu item	Description
 Product Configuration	Opens the Product configurator. This allows you to modify the product configuration in order to adapt it to the license you purchased. See " Changing the product configuration " ²⁷
 imc LICENSE Manager	Opens the imc LICENSE Manager. The software licenses are managed by means of the imc LICENSE Manager. If appropriate, customize the licenses here.

Help

Menu item	Description
 Help	Opens the "Help" for imc STUDIO.
 Additional Documents	Opens the program " imc Help and Documentation " ³⁹ . Here you will find documents on the imc devices and other products.
 About	Here, all the important " version information " ³³ on your installation of imc STUDIO is presented.

6.3 Navigation pane and Quick access toolbar

Navigations pane



Maximized Navigation Pane
(example)

In the **Navigation pane**, the main windows of the installed plug-ins are displayed as a tree diagram. To open/change the main window, click on its corresponding entry.

The Navigation pane can be opened or closed, maximized and minimized.



- To **maximize/minimize** the Navigation pane, click in its top bar on the arrow symbol.
- To **open/close** the Navigation pane, click on the upper region of it below the arrow.

If the Navigation pane is maximized or opened, the main windows are additionally displayed as a tree diagram (see example shown).

Via the tree diagram, it is possible to jump to different main windows or directly to the pages containing the main windows.

You can hide the Navigation pane.

If you hide the Navigation pane, please note that your ability to navigate between the main windows. For this case, you should add the command: "*Browse workspace*" at suitable locations, for instance in the ribbon menu ("*User-defined buttons*"⁵⁹).

To show or hide the Navigation pane use the dialog: "*Tool windows chooser*"⁷⁰.

Quick access toolbar

The "Quick access toolbar" can be supplemented with various menu items. Which actions are available for this purpose doesn't depend on what tab is currently displayed in the menu ribbon.

Using the context menu, you can either add or remove menu actions.

- Adding: context menu of the menu item in the menu ribbon (note: the menu ribbon may not be minimized)
- Removing: context menu of the menu item in the toolbar

You may place the toolbar either above or below the menu ribbon.



Note

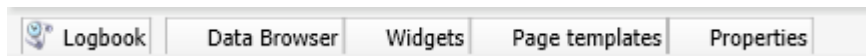
Saving the configuration

The **configuration of the "Quick access toolbar"** is **saved**⁹¹ with the respective **view**.

The configuration is **only saved** whenever the **project is saved**.

6.4 Tool windows

Tool windows contain special elements for operating/viewing a main window. (For information on operating the tool window: "[Operation](#)"⁶⁹)



Five tool windows

- The imc STUDIO frame's log book

- Four additional windows from the main window imc STUDIO Panel

One tool window belongs to the imc STUDIO frame and is always available:

- [Logbook](#)⁷¹

The *Logbook* is always displayed and can be minimized. The other main windows' tool windows (e.g. for the Panel) are displayed as soon as the respective main window is opened.



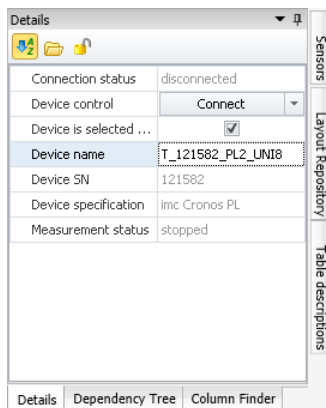
Note

Not all tool windows are visible

By default, all necessary tool windows are displayed. All others can be [displayed](#)⁶⁹, if you need them.

6.4.1 Operation

Tool windows contain special elements for operating and editing a main window.



Tool window (example)

Each main window has its own tool window, which is described in the documentation of the respective component. Tool windows can be moved and deleted.

By default, the tool windows are pinned to the main window (e.g. at the bottom or right margin).

The picture shows an opened tool window ("**Details**"). It is pinned and contains two additional tabs. If you click on one of these tabs, the corresponding tool window opens ("**Dependency Tree**" and "**Column Finder**").

Three other collapsed tool windows are seen at the right margin ("**Sensors**", "**Layout Repository**" and "**Table descriptions**").

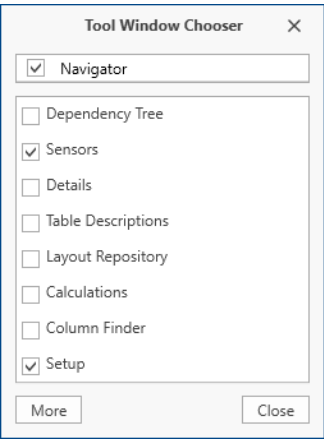
A tool window is opened when the mouse is swiped over it.

As long as a tool window is selected, or if the mouse is located over the tool window, the tool window remains open. Otherwise, it collapses to the side.

Showing/hiding the tool window

You can either show or hide tool windows. To do this, open the dialog: "Tool Window Chooser".

Ribbon	View
View > Choose Tool Windows (📁)	Complete
Extra > Choose Tool Windows (📁)	Standard

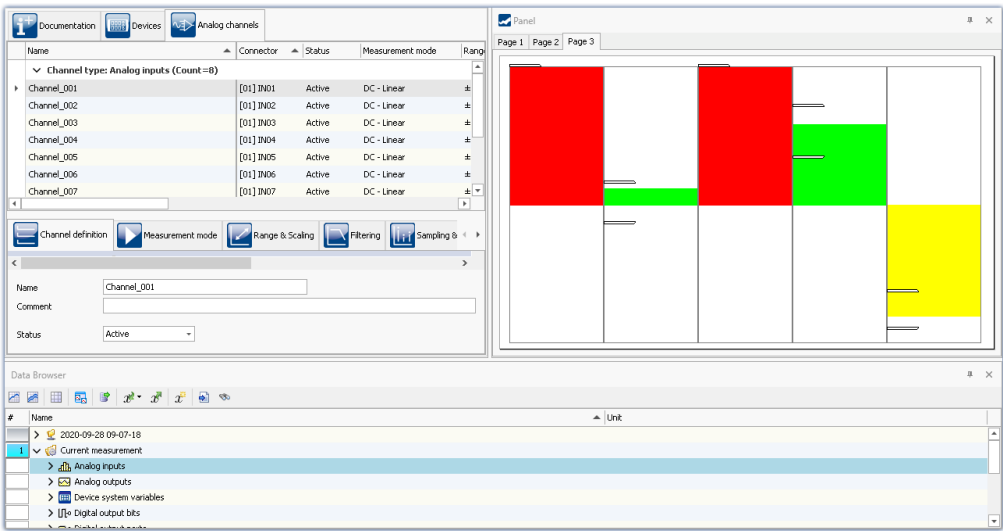


The dialog: "Tool Window Chooser" opens.

Set a checkmark in front of the tool windows which you wish to have displayed.

Selecting "More" provides you with comprehensive layouting capabilities for the view. You are able to place any tool window belonging to other components on the current page. For instance, you can display the Data Browser in the window of the Automation. Or you can show the Panel in the Setup.

Dialog: Tool windows chooser



Example: Panel and Data Browser are displayed together in the main window "Setup".

Other actions

Action	Description
Pinning the tool window	Tool windows can be pinned. Pinned tool windows remain open even when no longer selected. In order to pin a tool window, click on the thumbtack-button (📌).
Freely positioning the tool window	In order to freely position the pinned tool window, move the tool window's title bar using drag&drop to the desired position.
Docking the tool window	In order to dock the tool window, use the drag&drop technique by grabbing the tool window's by its title bar and moving it to the desired position. A Dock-symbol will appear at the locations which are possible.  In the center of the targeted window (main window or tool window), a cross appears. To dock the tool window, release the mouse button with the mouse over the desired position.  Dock at the top/bottom margin  Dock at the left/right margin  Insert as a tab in a different window (see example)

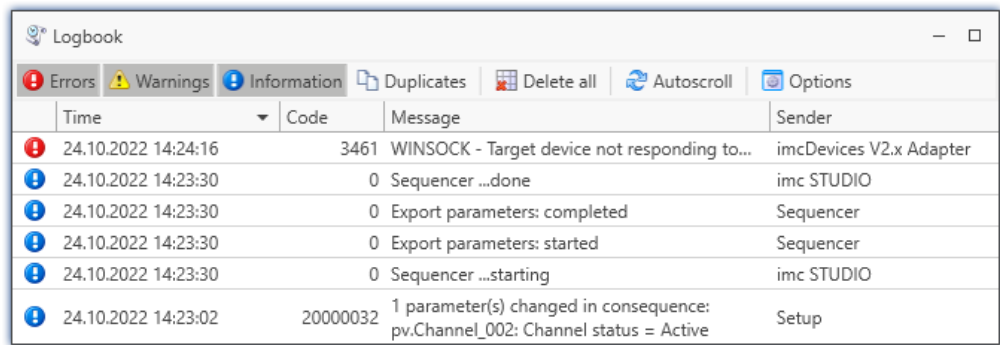
6.4.2 Logbook

In the tool window "Logbook", **messages of the categories:** "*Fatal*" (🛑), "*Error*" (🚫), "*Warning*" (⚠️) and "*Information*" (ℹ️) are entered. The logbook entries **indicate what problems and errors exist** and provide hints on where to find them and how to remedy them. Some actions performed are documented.

E.g., a note is entered in the logbook for every command performed:

- performed successfully ("*Information*") or
- otherwise "*Error*" or "*Warning*"

By default, the Logbook is opened when an entry any of the categories "*Fatal*", "*Error*" or "*Warning*" occurs. By default "*Information*" category notes are entered into the logbook without any further effect.



Example of logbook entries

Each logbook entry consists of:

Parameter	Description
Symbol for the category	Fatal (🛑), Error (🔴), Warning (⚠️) and Information (📘)
Time	Time the logbook entry occurred
Code	Logbook entry's error code
Message	Exact description of the logbook entry
Sender	The origin of the logbook entry

After a new imc STUDIO Monitor start, the logbook is empty. The logbook itself only displays messages which occurred since the last start. Older messages can be opened with the "Logbook-Viewer".

 **Note**

The logbook is saved

For purposes of tracing history, the logbook is saved daily:
Default path for Windows 10: **C:\ProgramData\imc\imc STUDIO\Applications_1\log** (example)

Buttons / Context menu	Description
Filtering the logbook	Filtering the Logbook according to categories. To do this, click on the Filter-button: "Errors", "Warnings" or "Information"  Errors  Warnings  Information Logbook filter Usually, all categories are selected.
	 Filtering also takes effect on new messages Note that the filter has an effect when a new message appears. For this reason, this message may not be displayed.
	 Default setting for the filters You can pre-configure the filter settings using the option ⁶³ ("Default settings for the filters in the logbook tool window"). Thus for example, you can have certain information entries hidden by default. Such entries can later be displayed again if needed without affecting the default setting.
Show search panel	Search the logbook for entries and text passages Located in the context menu of the title bar. An input appears. Enter the desired search keyword here. All messages are displayed which contain the text entered.
	 Search also takes effect on new messages Note that the search works like a filter and has an effect when a new message appears. For this reason, this message may not be displayed.

Buttons / Context menu	Description
 Duplicates	Hiding duplicates In certain cases, the system attempts to perform the same action multiple times in succession. In consequence, multiple duplicate reports are displayed in the logbook. In order to keep the logbook clear and concise, it is possible to group the duplicates. The first and the last messages are always displayed, thus indicating the points in time for the first and last occurrences. In front of the displayed text for the grouped message, a number appears in parentheses. This number indicates how many times the message occurred.
 Copy	Copying the selected message to the Clipboard Context menu "Copy": All information of the entry that can be seen in the logbook is now in the Clipboard.
 Copy with details	Copying the selected message with internal information to the Clipboard For error messages, the stack trace is added to the copied text. This helps our tech support to analyze errors reported to us. Context menu "Copy with details": All of the entry's information is then copied to the Clipboard.
 Delete all	Deleting the displayed messages  "Delete all" clears the display Be aware that this only purge the display, but the logbook on the hard drive remains unaffected. As well, it is still possible to view the deleted messages in the "Logbook-Viewer".
 Autoscroll	Activating/deactivating automatic scrolling When a new message is entered in the logbook, the display in the "Autoscroll" mode automatically jumps to the new entry. If you wish to investigate older messages, you can deactivate this mode. This happens automatically when a message is selected, or click on the "Autoscroll"-button. Re-activate the mode by: • pressing <Esc> • left-clicking the mouse over the selected entry while holding down the <Ctrl> key • clicking on the "Autoscroll"-button in the menu  New messages are not automatically displayed visibly Be aware that when the "Autoscroll" mode is deactivated, you will not be automatically notified when new messages appear.
 Logbook-Viewer	Starting the Logbook-Viewer and opening old logbooks By means of the "Logbook-Viewer", you can look at current, deleted and saved logbook entries. In addition to the familiar logbook-functions, it is possible to load saved logbooks.
 Options	Options for the Logbook From within the logbook, you can open the Options for the Logbook .
 Export	Logbook files are frequently needed by our technical support for error analysis purposes. In response to this button, all existing logbook files are zipped. They can then be sent to our technical support.

Useful options

Options - Logbook	Description
Delete Logbook-files [days]	After the specified number of days elapse, the logbook files are deleted automatically. The deletion verification test is performed upon starting the program and at 0:00 (12 AM).
Default settings for the filters in the logbook tool window	Category of messages for which the filter presets are to be made.
Open logbook in response to a report of category:	For this category, the logbook is opened and attached.

6.5 User administration and Access rights

In some cases, it is mandatory for each user to be able to identify themselves. This ensures that only authorized users can work with imc STUDIO Monitor.

 Note

- The user administration is available as of the product edition **imc STUDIO PRO**.
- By default, the User Administration is deactivated. In that state, each user has the rights belonging to the group "*imc Administrators*".

In imc STUDIO Monitor, a variety of actions are associated with particular user groups. Each user registered is assigned to a particular user group.

Affiliation with one of these user groups determines which imc STUDIO Monitor functions are available. For instance, a test engineer could configure experiments and hand them over to technicians to run. If the access rights are configured accordingly, then these users could perform the measurements and check and save the results. However, they would not be able to change the experiments.

Users' privileges are determined according to their respective group association. There are no special rights for an individual user.

The user group has extensive influence on the visibility and/or operability of menus, icons, etc..

The groups are ordered hierarchically (lower > higher):

	User is in the group " imc Standard Users "
	User is in the group " imc Advanced Users "
	User is in the group " imc Administrators "
	User is in the group " imc Developers " (only in the product edition: imc STUDIO DEV)

 Note

The user administration is saved with the project

- The User Administration settings are saved with the respective project.
- Changes take immediate effect, but are only saved when the project is saved.

Chapter overview

Synopsis	Section
The association of users to user groups is set in the " <i>User Administration</i> ".	User Administration ⁷⁶
Activating / deactivating the User Administration.	
Upon installation, default access rights are assigned. As a " <i>imc Administrators</i> ", you can make very detailed configurations.	Setting access rights for actions ⁸³
When the program starts, it determines to which user group the user logged into the PC belongs.	Startup behavior ⁷⁹
You can switch users after the program start.	Logging on, logging off, and switching users ⁸⁰

6.5.1 User administration

In this dialog you can activate/deactivate the user administration. You can set up users and assign them to the user groups. Thus, each user obtains rights corresponding to their respective group.

Open the dialog by selecting the ribbon menu item "Administration".

Ribbon	View
Extra > Administration (🔑)	Complete

Local user administration

Add User... Delete User... Change user...

Extra ▾

🔑 On

🔑 Off

User	.. imc user group
WERNER\Werner Schneider	.. imc Administrators
Bernd Müller@imc STUDIO	.. imc Administrators
Jens Schulze@imc STUDIO	.. imc Advanced Users
Werner Schneider@imc STUDIO	.. imc Standard Users

☒ Allow login of anonymous users

OK Cancel

User Administration

Assignment to imc user groups

! Note

Login is required in order to start the User Administration

The User Administration settings can only be changed if a user having the associated rights is logged in.

For this reason, a separate login process is necessary for starting the User Administration.

Activating / deactivating the User Administration

When the User Administration is (being) deactivated, every user has the rights associated with the group "imc Administrators".

- To activate the User Administration, click on the Activation-button (🔑)
- To deactivate the User Administration, click on the Deactivation-button (🔑)

User types

There are a variety of user-types:

Project-internal imc STUDIO users	Users who can be assigned independently of the particular PC or domain. The user can be protected with a password.
Local computer accounts or groups	Windows or Domain account or group. The group memberships set under Windows or in the domain are not used. The user is protected by the password used for login to Windows .
Active Directory domain accounts or groups	

Adding users

Adding a user of the type "Project-internal imc STUDIO user":

- click on the button "Add user"
- select the user type "Create an project-internal imc STUDIO user".

The following dialog appears:

Dialog for adding a user of the type
"imc STUDIO internal user"

- Enter a user name
- Enter a password and confirm it.
- When the entry is correct, click on "OK". The user is added.

The new user's identity is structured as follows:
UserName@imc STUDIO.



Note

Minimum length of user name and password

The minimum length of the user name and of the password can be determined using the options in the title bar.

Adding a user of the type Local computer or Active Directory domain accounts or groups:

- Click on the button "Add user"
- Select a user type "Add a Local Computer account or group" or "Add a domain account or group from Active Directory"
- The Windows dialog for finding an account or group appears; follow its instructions
- Once a correct entry has been made, the user is added.

The new user's identity is structured as follows: *UserName@Complete Computer Name*.

Deleting / removing a user

To delete a user from the list:

- select the user,
- click on the button "Delete User",
- confirm the prompt by clicking on "Yes" if you have selected the correct user for deletion.

Assigning a user to a user group / Changing the user group

The groups are ordered hierarchically (lower > higher):

	User is in the group "imc Standard Users"
	User is in the group "imc Advanced Users"
	User is in the group "imc Administrators"
	User is in the group "imc Developers" (only in the product edition: imc STUDIO DEV)

To assign a user to an user group

- open the drop-down list of the column "*imc User Group*" and
- select the desired group



Note

The system must always contain at least one user belonging to the group "*imc Administrators*" or higher.

Anonymous users

Users who are not entered in the User Administration are referred to as "*Anonymous Users*".

If Anonymous Users are permitted, any user can log in with an arbitrary name. This user receives the rights belonging to the user group "*imc Standard Users*".

If, for instance, the PC-user is not set up as a user, at the time that person starts imc STUDIO Monitor, they are logged in as an Anonymous (with [Startup behaviour](#)⁷⁹: "*Windows Session*").

Prohibiting logging in as an Anonymous User

To prohibit Anonymous Users from logging in,

- delete the checkmark next to "*Allow login of anonymous users*".

Export/Import

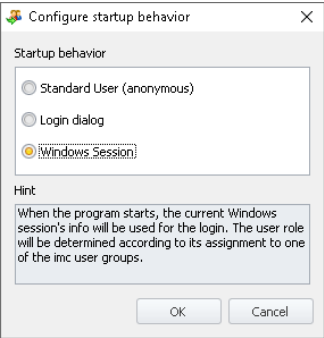
In order to be able to use the up-to-date user administration also in different projects or on other computers, you can export it and then import it at the intended destination.

- click on the button "*Extras*"
- select the desired action: Export or Import

6.5.2 Startup behavior

In this dialog, the startup behavior is set. Open the dialog by selecting the ribbon menu item *"Startup Behavior"*.

Ribbon	View
Extra > Startup Behavior (👤🔑)	Complete



Startup behavior

Startup Behavior	Description
Standard User (anonymous)	imc STUDIO Monitor always starts with the access rights of the group <i>"imc Standard Users"</i>
Login Dialog	When starting the program, the Login ⁸⁰ dialog is displayed.
Windows Session	The program uses the access rights of the user logged into the PC, in accordance with the affiliation with one of the user groups. That is the default setting after installation. For the assignment between the users and the user groups, see the section User administration ⁷⁶. If the User Administration is activated, any PC-users not belonging to any user group ⁷⁶ are assigned the rights of the group <i>"imc Standard Users"</i>.

6.5.3 Logging on, logging off and switching users

The user name serves to identify and authenticate a user. This ensures that only authorized users work with imc STUDIO Monitor.

With startup behaviour "[Login Dialog](#)⁷⁹" when imc STUDIO Monitor starts the Log On window appears. As well, after imc STUDIO Monitor has started, you can switch users

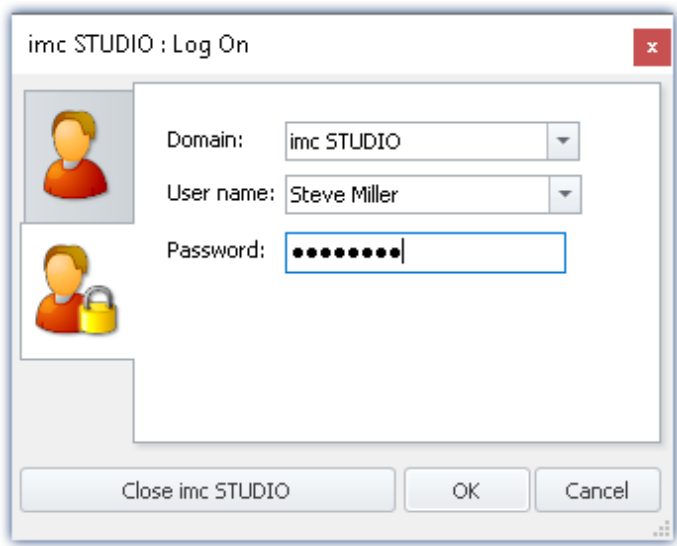
Ribbon	View
Extra > Login/Logout ( / )	Complete
Home > Login/Logout ( / )	Complete

If the **User Administration is deactivated**, logging on is always anonymous. Every user receives the rights belonging to the group "*imc Administrators*" for the duration of the session.

If the **User Administration is activated** the user can select:

Log on an anonymous user	Only possible if Anonymous Users are permitted (see " User Administration ⁷⁶ "). In this case, anonymous users have the rights belonging to the group " <i>imc Standard Users</i> ".
Log on as registered user	

Logging on a registered user



Login Dialog
Logging on a registered user

Authentication of the registered user upon logging on references:

• the domain,	Select the domain: "imc STUDIO", "Computer-name" or "Domain-name"
• the User name and	Select a registered user (all users belonging to the selected domain are listed)
• the associated password.	If required, enter the correct password

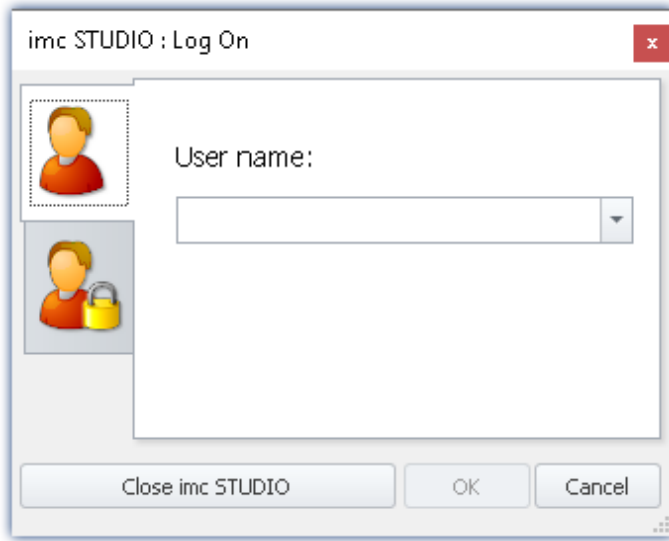
 **Note**

Windows-Password

- For logged-on users of the PC, imc STUDIO Monitor uses the associated Windows password.
- The password entered is never saved by imc STUDIO Monitor. The password is transmitted to Windows as hash-code. Windows verifies its validity and returns the result to imc STUDIO Monitor.

Logging on an anonymous user

Authentication of the anonymous user references the user name. However, any arbitrary name can be used.



*Login Dialog
Logging on an anonymous user*

Logging a user off

When the user currently logged on logs off, the Login dialog appears.

The user is logged off. Cancel is not possible.

Status bar

The Status bar at the lower left indicates which user is currently logged on. To the left of the name is the symbol representing the respective user group.

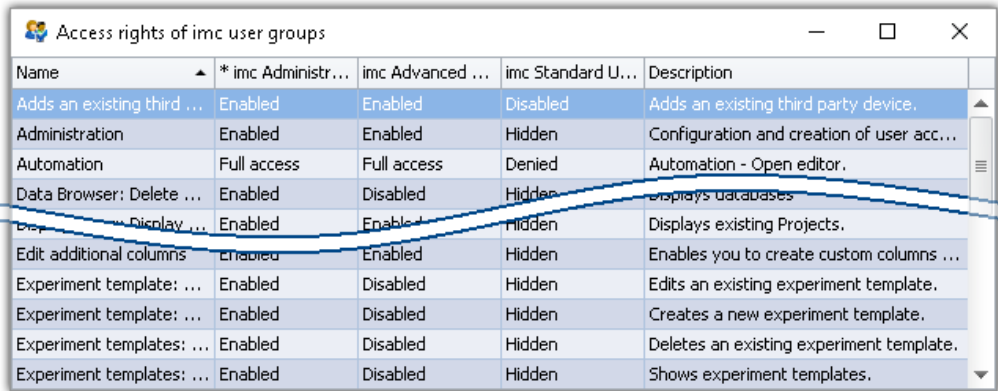


6.5.4 Setting access rights for actions

In the dialog "Access rights of imc user group" the rights belonging to the user groups are listed. The access rights can be edited according to the current requirements.

Open the dialog by selecting the ribbon menu item "Setting Access Rights".

Ribbon	View
Extra > Setting Access Rights (👤)	Complete



Dialog for setting access rights

Any user in a higher group can restrict the rights of a user in a lower group, or assign him higher rights up to his own level. However, one cannot change one's own rights or those of a member of a higher group.

The following rights levels exist:

Rights level	Description
Hidden	not visible (not accessible for operation)
Disabled	visible, but blocked (not accessible for operation)
Enabled	visible and accessible
Denied	
Write protected	visible and accessible for operation by not editable
Full access	no restrictions

For some entries, the rights may differ from the list above.

 **Note**

Rights: Saving an experiment

No changes can be saved unless the user who is logged in possesses experiment saving privileges. If a different experiment is loaded or if the program is closed while a user not having saving privileges is logged in, the changes will be discarded without any message.

 **Reference**

See also:

Panel in fullscreen mode - [Protection against changes](#) 

6.6 Views

The program's GUI is adaptable, so that for example the windows and columns can be positioned and displayed according to the user's desires. **The layout of the GUI is saved** in so-called "Views". It is possible to create separate views for various requirements.

The following settings are saved along with a view:

Region	Description
Window arrangement	The imc STUDIO Monitor window's and the tool window's position and size
Menus	Layout of the menu ribbon and the Quick access toolbar 68
Layout	• The last main window opened • The tool window's layout (e.g. what metadata columns are displayed in the Data Browser) • The arrangement and configuration of the Setup pages (e.g. arrangement of columns, such as metadata columns)

 **Note**

Views and projects

The relationship between saving the view and saving the project

The views are saved with their respective project. When a view is saved, it is necessary for the project to be saved. The system will notify the user accordingly.

Conversely, when a project is saved, the view is not automatically saved. Views are only saved whenever the action "Save View (as)" is intentionally selected. In order to avoid every little change to a view being saved whenever the project settings are saved, the views are ignored when specifically the project is saved.

Settings which apply globally to all views are saved with the Project

Please note that only the positions of individual elements is saved along with the views. The existence and configuration, for instance, of the Setup-columns (tables- and column descriptions) is saved along with the respective project. (see also "[What is saved where?](#)" 91)

Since the project is always saved whenever the view is saved, the current configuration of the columns is also included when saving.

Overview of settings saved with the project, which affect the views

- Setup table descriptions and column descriptions (these include, for example, additional columns, such as metadata-columns, parameter set columns, ...) (tool window: Table Descriptions)
- saved Setup complete layouts (tool window: Layout Repository)

 **Note**

Detecting changes to a view

If any major changes (columns added, menu changed, etc.) are made, the system explicitly recognizes this as a change to the view. When you exit the project by switching to a different project or by closing imc STUDIO Monitor, you are prompted to select whether to save the changes.

Minor changes, such as changed size of tool windows, are ignored.

Saving views

In order to save the current view, select the menu item "*Save View (as)*":

Ribbon	View
View > Save View / Save View as ( / )	Complete
Extra > Save View / Save View as ( / )	Standard

All settings belonging to the current view are saved. These include the items named above.



Note

The views are saved with the project

The views are saved with their respective project. When a view is saved, it is necessary for the project to be saved. The system will notify the user accordingly.

Loading views

In order to load a saved view, select the menu item "*View*" in the drop-down list of the respective view. Here, all views available to the project are displayed.

After the selection the view is loaded.

The record of an experiment includes the view in which the experiment was saved. When the experiment is loaded, the view also loads automatically.

Deleting views

To delete a saved view, select the menu item "*Delete View*".

Ribbon	View
View > Delete View ()	Complete
Extra > Delete View ()	Standard

In the dialog which then opens, select the view to delete from the drop-down list. Confirm your selection by clicking on "*Save project*".

Restoring views

Views are not write-protected. They can be restored to their original state. The factory settings come with a copy of all views created as part of the program installation.

To restore a view from the factory settings, select the menu item "*Restore*".

Ribbon	View
View > Restore ()	Complete
Extra > Restore ()	Standard

In the dialog which then opens, select the desired views. They are imported from the factory settings. Confirm your selection by clicking on "*OK*".



Reference

See also

- Export/import of [views](#) ⁸⁶
- Export/import of [Setup table descriptions](#) ¹⁷³
- Export/import of [Setup column descriptions](#) ¹⁷³
- [What is saved where?](#) ⁹¹

6.6.1 Exporting/importing views

When you export the view settings, a variety of project settings are stored in the export file.

Settings	Description
Saved views ⁸⁴	• table descriptions and column descriptions (these include, for example, additional columns, such as "metadata columns", "parameter set columns", ...) (tool window: "Table Descriptions") • saved complete layouts (tool window: "Layout Repository")
Sensors	• user-defined characteristic curves/sensors (tool window: "Sensors")

Export of views, metadata columns, sensors, ...

By means of the menu item "Import/Export", you can export the view settings:

Ribbon	View
Project > Import/Export	Complete
Extra > Import/Export	Standard

- Choose option "Export" and then the list item "**Export views, meta data columns, sensors, ...**"
- Choose an appropriate file name and path for the view settings file



Note

Save the view beforehand

Save the view beforehand if you have made any changes (see "[Views](#)" ⁸⁵).

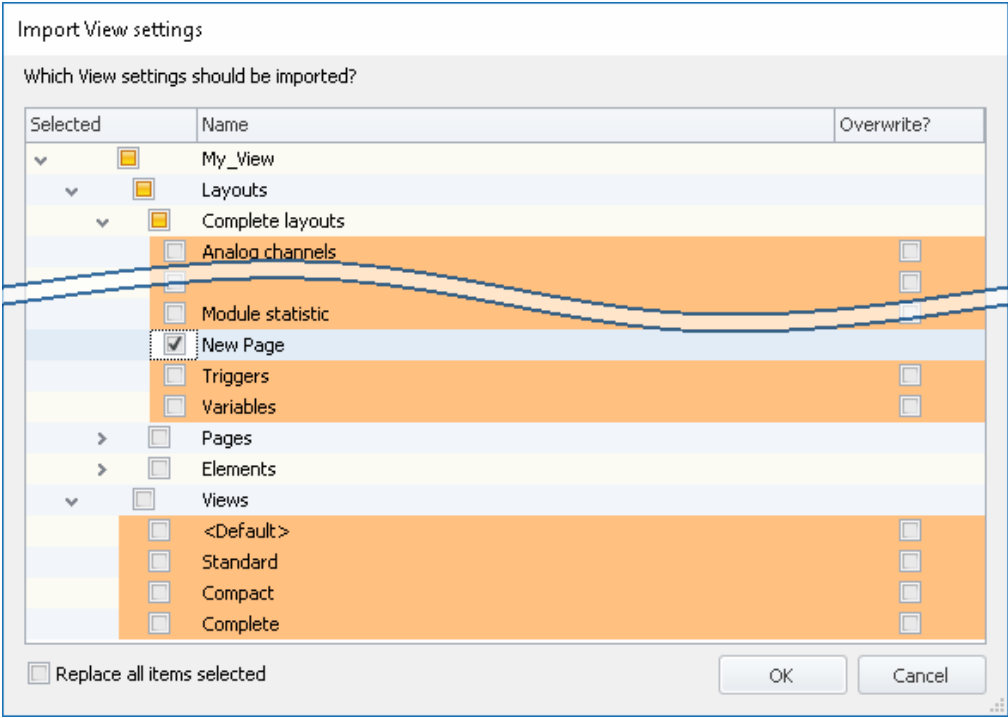
Import of views, metadata columns, ...

Using the menu item "Import/Export", you can **import the view settings to the current project**:

Ribbon	View
Project > Import/Export	Complete
Extra > Import/Export	Standard

- Choose option "Import" and then the list item "**Import views, meta data columns, ...**"
- In the file selection dialog, selected the saved view settings file

Thereafter you will see a dialog allowing you to import either all or individual view settings.



Entries having **colored shading already exist** in the current project. In this example, the complete layout "New Page" is not yet present there.

Checkbox	Description
Selected (left)	Check this box if you wish to import the element .
Overwrite? (right)	Check this box if the element in the project is to be overwritten without any confirmation prompt . If the box is not check-marked, a prompt appears to inquire the name for the element to be imported. (Unless "Replace all items selected" is check-marked.)
Replace all items selected	Corresponds to "Overwrite?" for all entries: Check this box if you want to replace all elements without any confirmation prompt .

You can select the following elements:

Element	Description
Layouts	All these elements belong to the Setup pages. Saved pages and saved designs, which can be displayed on the pages. For more information on the complete layouts, see the chapter: " Table Description and Complete Layout " ¹⁷⁴ .
Views	Here you find the imc STUDIO-views. The views ⁸⁴ apply for all components of imc STUDIO.

 **Note**

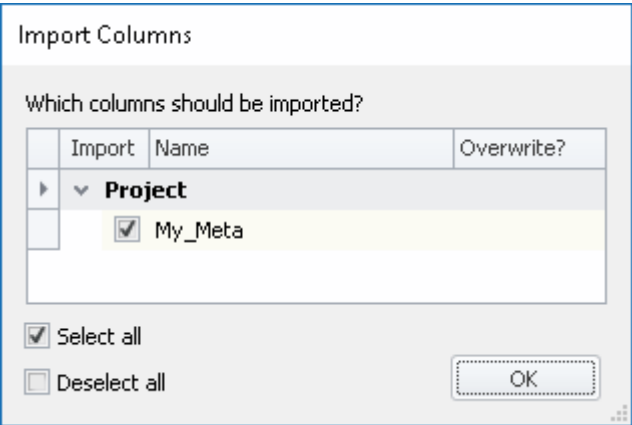
New functions can be overwritten

Be aware that when making replacements, new functions may be lost if you import old views/columns.

Importing defined additional columns

An additional prompt appears if

- discrepancies between existing **Setup columns** were detected or
- the file contains **additional columns**.



Select which columns to import and/or any to replace.

If the **column already exists**, a checkbox appears under "Overwrite?". If the box is not check-marked, the **column is automatically renamed**. You can edit the name afterward. For more information, see the chapter: "[Creating and configuring additional columns](#)"¹⁵¹.

 **Reference**

See also

- Export/import of [Setup table descriptions](#)¹⁷³
- Export/import of [Setup column descriptions](#)¹⁷³

6.6.2 Customize ribbon menu

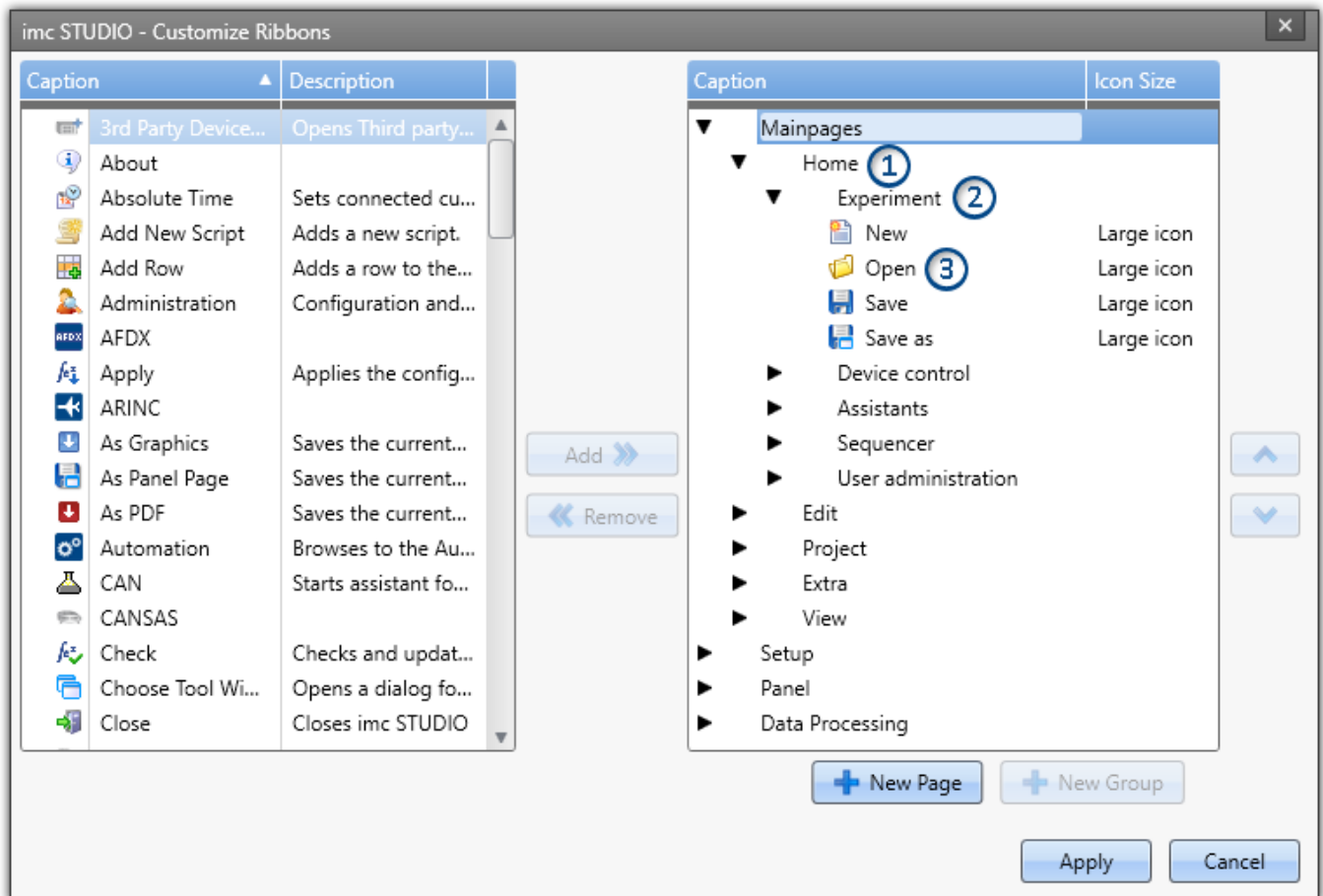
In order to provide quick access to the most important functions, you can customize the menu ribbon. The menu ribbon usually consists of multiple filecards (pages, panes) and multiple groups. The groups contain the menu actions. You can move or delete existing menu actions, or add new ones. Additionally, you are able to create your own custom groups and pages.

Each view has its own menu ribbon configuration. You can modify these configuration and save them along with the associated views.

In order to modify the menu ribbon, open the dialog: "*Customize Ribbon*".

Ribbon	View
View > Customize Ribbon Menu ()	Complete

If this button is not present in your view, the dialog can also be opened by clicking on the corresponding symbol () in the menu bar (top right).



In the left-side pane, you find all available menu actions. In the right-side pane, the menu actions belonging to the currently selected view are shown. The lists may also include actions which may not be visible in the user interface; e.g., actions only permitted for certain user roles/privileges, or actions which are only activated in conjunction with special product configurations.

Using the buttons: **+**, you can add new **pages** (1) or **groups** (2). To do this, first select the desired location in the right pane. Using the button: **➡**, add the **action** (3) selected in the left pane at below the current position.

Delete the currently selected action, page or group from the list using the **⬅** button.

The buttons **⬆** and **⬇** are used to change the action's position within the group. The **icon-size** in the menu ribbon can be modified in the column at the far right of the right pane, behind the respective action.



Reference

Personalized menu items

You have the ability to create your own personal menu items. See:

- a command sequence ("[User-defined buttons](#)"⁵⁹)

6.7 Information and tips

6.7.1 Data management

imc STUDIO uses a "Database" for saving projects. The database's path can be freely specified (see "[Experiments, Projects and the Database](#)"⁴⁵).

All "Projects", "Experiment templates", "Experiments" and **measured data** are saved to the database.

The following files are created for a sample project:

(Project: "MyProject", Experiment template: "MyTemplate", Experiment: "MyExp")

MyProject	Project Name	
+---config		
MyProject.imcAppSettings project.pcf	Project properties	
\---templates		
MyTemplate.imcStudioTemplate StandardTemplate.imcStudioTemplate	Experiment templates	
\---~MyProject		
\---2022-01-18.bak MyProject~001.imcAppSettings	Project backup files	
\---MyExp	Experiment Name	
+---2022-01-19 14-00-00		
129977989468984375.ms.lnk Chanel_01.raw	Saved measurements (the data structure is set in the device settings under Storage ¹⁸⁰) • *.raw: measured data • *.ms.lnk: link to the appropriate experiment setting	
+---2022-01-19 15-00-00		
129977989628330078.ms.lnk Chanel_01.raw		
+---config		
MyExp.imcStudio MyExp.Prepare_imcStudioArchiv MyExp.Prepare.disconnect	• *.imcStudio: experiment-file • *.Prepare*: Necessary in order to be able to re-connect with the running measurement • and additional internal files	
\---~MyExp		
\---2022-01-19.bak MyExp~001.imcStudio MyExp~002.imcStudio	Experiment's backup files	
\---Meta	Own files ⁴⁶ for the experiment, such as metadata, imc FAMOS-sequences, info-files	

 Note

Please note that not all files are listed here. For the internal administration of the database, several internal files are necessary.

6.7.2 What is saved where?

Every setting, option and parameter is saved. Some are saved with the loaded experiment, some with the project and others with the program. Below is a list of the important areas and where they are saved.

Legend:

Appl: Application

Seq: Sequencer

Tab.d: Table description

Proj: Project

Exp: Experiment

File: File system

Name	Description	Appl	Proj	Seq	Exp	View	Tab.d	File
Supplemental files	Files such as characteristic curves for device/ imc Inline FAMOS				Exp			
Selected device					Exp			
Known devices		Appl						
Connected sensors					Exp			
Default values	Setup pre-set values		Proj					
Repository (Automation)								File
Repository (Panel)								File
Sequencer	Sequence table in the Sequencer				Exp			
Sequencer-Events (Sequencer)	Events which apply to the loaded experiment			Seq				
Sequencer-Events (Experiment)	Events which apply to the loaded experiment				Exp			
Sequencer-Events (Project)	Events which apply to all of the project's experiments		Proj					
User Administration	Defined users		Proj					
Access rights	Privileges of user roles		Proj					
Show tabs (Panel-Fullscreen)	Visibility of the checkbox "Show tabs" in the Panel-Fullscreen		Proj					
Options (experiment)					Exp			
Options (project)			Proj					
Options (application)		Appl						
Views	window arrangements, menus, layouts, ...		Proj					
Menu ribbon (customizing)	Which button appears where					View		

Name	Description	Appl	Proj	Seq	Exp	View	Tab.d	File
Quick access toolbar	Which button appears where					View		
Column arrangement	What columns are displayed where					View		
User defined buttons (collection)	What buttons exist and their significance		Proj					
User defined buttons (customizing)	Which button appears where		Proj					
Table descriptions	Aggregation of the normal and additional columns		Proj					
Normal columns	What columns exist and their significance						Tab.b	
Additional columns	What columns exist and their significance						Tab.b	
Complete layouts	Completely designed pages		Proj					

6.7.3 Backup Copies – Data security mechanism for experiments

Each time the imc STUDIO experiment is saved, a backup copy of the previous "save-state" is also recorded. The following data storage depth arrangements are implemented:

- Per each calendar day, a maximum of the last 10 versions is saved.
- A maximum of 5 calendar days are saved.
- From the 3rd and previous calendar days, only the respective last version is saved.

Storage location - Backup-folder

The backup copies are located in a backup-folder within the Experiment-folder. There you will find a folder whose name has a prefixed tilde-character: "~<ExperimentName>". E.g. "~Experiment_001".



Note

The Backup-folder is hidden

This Backup-folder is hidden. However, it is possible to enter its path into the address bar of the Windows Explorer in order to open the folder. Or, you can have the hidden folders displayed.

The Backup-folder contains one associated sub-folder for each work day: e.g. 2018-05-17.bak, 2018-05-16.bak, ...

These each contain the most recent saved states of the experiment. Below is an example with three and four saved versions.

```
MyExperiment\config\~MyExperiment\2018-05-16.bak\  
  MyExperiment~001.imcStudio  
  MyExperiment~002.imcStudio  
  MyExperiment~003.imcStudio  
MyExperiment\config\~MyExperiment\2018-05-17.bak\  
  MyExperiment~001.imcStudio  
  MyExperiment~002.imcStudio  
  MyExperiment~003.imcStudio  
  MyExperiment~004.imcStudio
```

Restoring a backup version

To restore a saved version, import the .imcStudio-file via the [Project-dialog](#)⁴⁸. First copy the file, for instance, to the Desktop, and rename it in a convenient way, e.g. MyExperiment~004.imcStudio -> MyExperiment.imcStudio.



Note

Overwriting during import deletes the backups

Be aware that upon importing, the original experiment is overwritten if you give it exactly the same name. All additional backups are also deleted in consequence.

6.7.4 Guardian

Along with imc STUDIO Monitor, two system services are also installed: Guardian and WatchDog. These two services start upon starting the system.

WatchDog (GuardianWDSERVICE)

The WatchDog has the sole purpose starting the Guardian if it is not already running.

Guardian (GuardianService)

The Guardian handles a variety of tasks.

Task	Description
Preparing the database for imc STUDIO Monitor	The Data Browser displays the results of any measurements performed. When a measurement is run, the system informs imc STUDIO Monitor that a new measurement needs to be displayed. However, if any additional data sets are added at a later point in time, e.g. by means of a imc FAMOS-sequence, then the program needs to be notified of this. The Guardian monitors the folder and informs the program about every file which has been added. In consequence, these files are also available to you in the Data Browser.
Deleting elements from the database	The Guardian handles the deletion of elements from the database, e.g. measurements or whole experiments. For this purpose, imc STUDIO Monitor creates a ".deleteMe"-file in the respective folder. Subsequently the Guardian deletes the folder and its contents once the system is ready to do so. Example: Interval data saving with a limited measurement count is activated. In the measurement folders to be deleted, imc STUDIO Monitor creates a ".deleteMe"-file. The Guardian deletes the measurement as soon as the system give clearance to delete it, for instance, as soon as the measurement is no longer loaded.



Note

Unable to establish a connection with the Guardian process

imc STUDIO Monitor always monitors the service's accessibility so that you will be notified in case the Guardian is no longer running, or even if it had not started. If the Guardian is unable to respond for any reason, a corresponding message is posted in the logbook: *"Unable to establish a connection with the Guardian process."*

If the Guardian becomes accessible again at a later point in time, you receive an additional message: *"The connection with the Guardian process has been restored."*

Closing services

In some cases, the services are not desired and need to be closed. In this case, please first close the WatchDog-service and subsequently the Guardian-service. Be aware that the Guardian's tasks will no longer be performed; e.g. any interval limitation will no longer be applied.

Versionen

There is a separate Guardian and a separate WatchDog for the 64-bit version. In case of parallel installation with a Version 5 system, two services are running. The 64-bit variant has the suffix "V2".

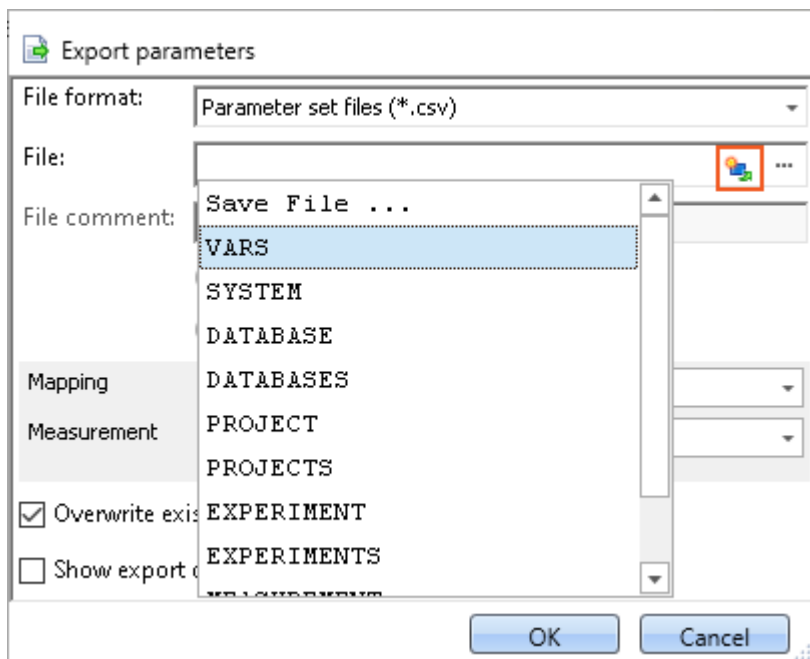
6.8 Placeholders

In imc STUDIO Monitor, you can use placeholders at various positions, which are **automatically recognized and resolved**.

With the help of placeholders, you can, for example, record a parameter set in the current experiment folder without knowing the path, or also display the variable's value in a text.

Placeholders can be used both in some [commands](#)^[954] as well as with [Widgets](#)^[617]. A list of all known placeholders is presented in the section "[list of placeholders](#)"^[96].

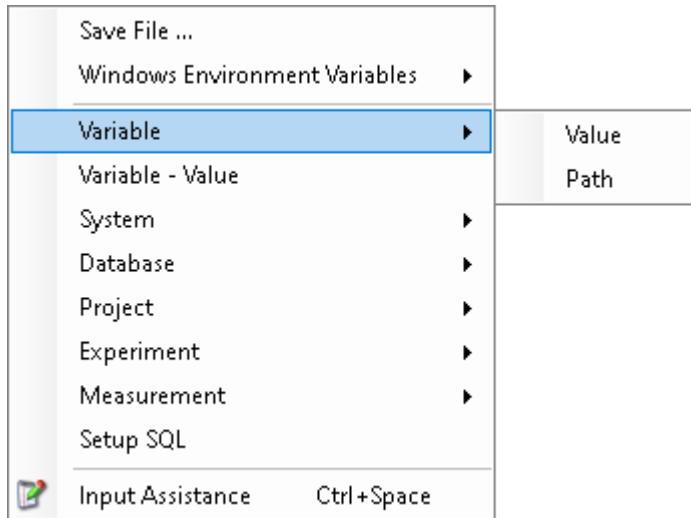
Placeholders are offered where the placeholder symbol () is displayed. Not all placeholders can be used at all positions. To obtain a list of available placeholders, click the left mouse button over the placeholder symbol ()



List of available placeholders for the command "Export parameters"

Along with clicking the mouse over the placeholder symbol, there are other possibilities in the input box for having the list of available placeholders displayed:

- by means of the key combination CTRL + Space
- by means of the context menu (right mouse button)
- in the context menu by means of the item "Input Assistance"



Input box context menu

There is also the possibility of entering the desired placeholder entirely manually; however, it is not possible to see whether the placeholder is supported at this location.

In the next sections, the following information is available:

- [Listing and description](#) ^[96] of known placeholders
- [Formatting](#) ^[111] of the placeholders
- [Environment variables of the OS](#) ^[114] supported by imc STUDIO Monitor

6.8.1 List of placeholders

The following placeholders are described in this section:

- [CONTROL](#) ^[97]
- [DATABASE/DATABASES](#) ^[98]
- [EXPERIMENT/EXPERIMENTS](#) ^[99]
- [MEASUREMENT](#) ^[102]
- [MEASUREMENTS](#) ^[103]
- [PAGE](#) ^[105]
- [PANEL](#) ^[105]
- [PROJECT/PROJECTS](#) ^[105]
- [SELCONTROL](#) ^[107]
- [SETUP](#) ^[108]
- [SYSTEM](#) ^[109]
- [VAR/VARS](#) ^[110]

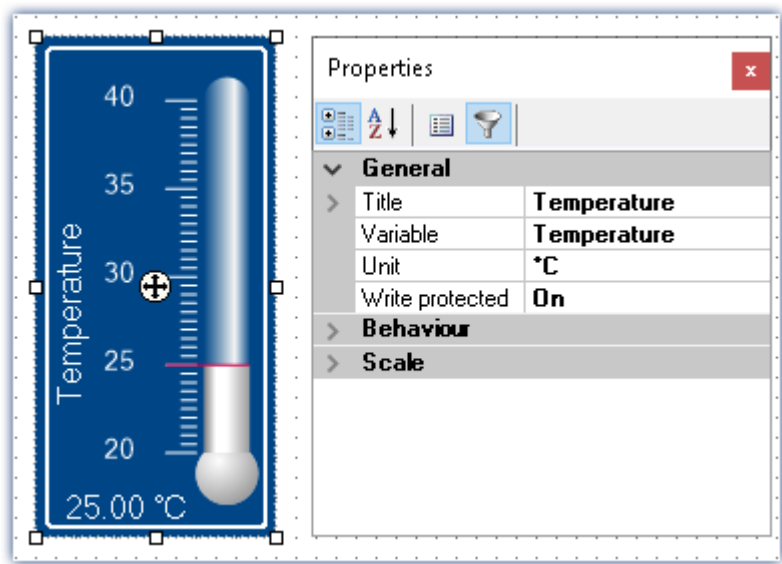
6.8.1.1 CONTROL

These placeholders are only available for Widgets.

Name	Context menu	Description
UNIT	Control - Unit	Unit of the current Widget, corresponding to the box Unit in the Properties-dialog. If there is no Unit, or if it is empty, the unit of the linked variable is adopted.
TITLE	Control - Title	Title of the current Widget, corresponding to the box Title in the Properties-dialog
VALUE	Control - Value	Current value of the variable, which is linked with the current Widget.

 Example

The following example illustrates what the three placeholders mean.



Properties of a Widget

In this example, the placeholders listed above have the following values:

- <CONTROL.UNIT> = °C
- <CONTROL.TITLE> = Temperature
- <CONTROL.VALUE> = 25

6.8.1.2 DATABASE/DATABASES

This placeholder is available everywhere where filepaths are needed, e.g. for export and import commands.

Name	Context menu	Description
NAME	Database - Name	Name of the database (= name of the database folder)
PATH	Database - Path	Path to the DB folder

Here, the entries are used which are entered in the *Global Options*.

Note

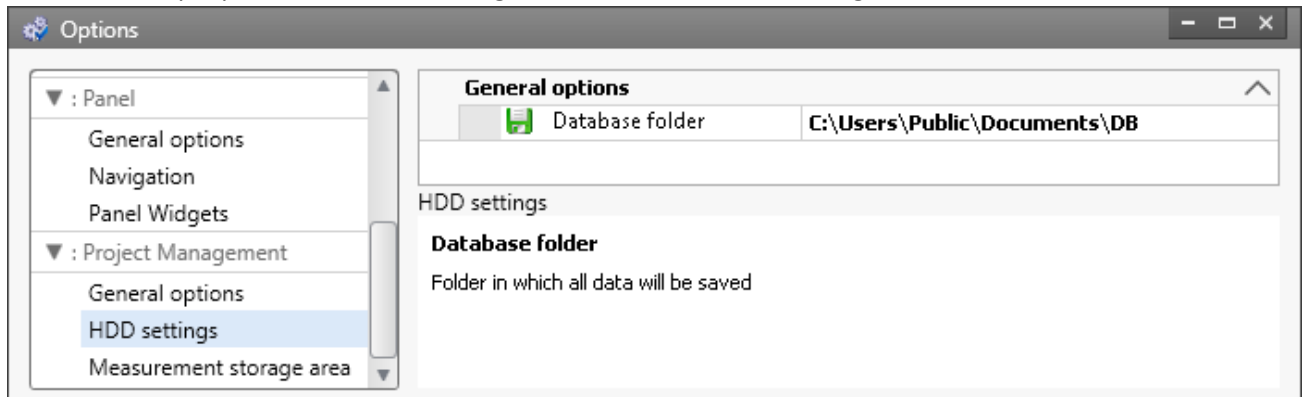
The placeholder *DATABASE* always pertains to the current project database. If you wish to use the placeholder with another database, the placeholder *DATABASES* must be used, e.g.:

<DATABASES["Another_Database"].PATH>

The workings of the two placeholders is illustrated by the following to examples.

Example 1

The first example pertains to the unchanged imc STUDIO default settings, here under Windows 10:



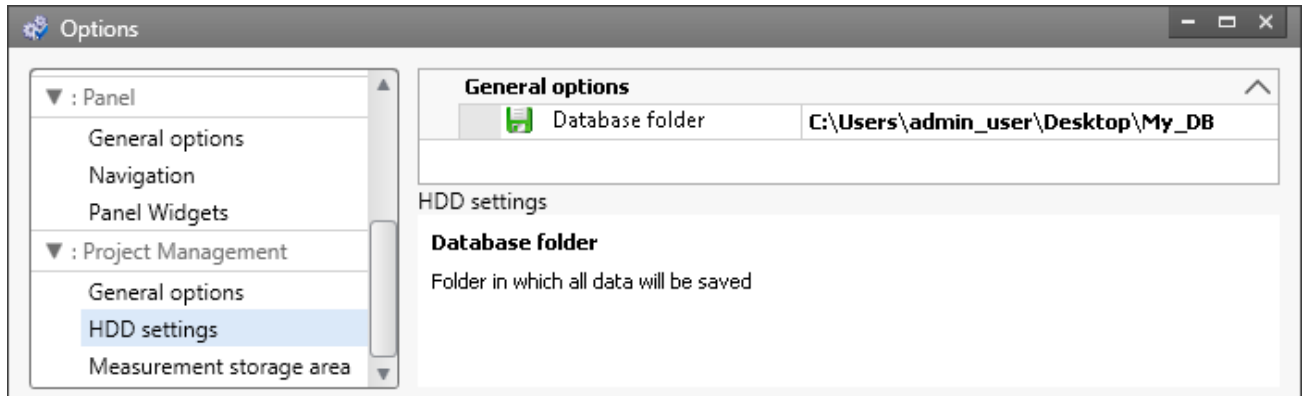
Default database path under Windows 10

- <DATABASE.NAME> = DB
- <DATABASE.PATH> = C:\Users\Public\Documents\DB



Example 2

The second example shows the resolved placeholders when the database path is changed:



Changed database path

- `<DATABASE.NAME>` = My_DB
- `<DATABASE.PATH>` = C:\Users\admin_user\Desktop\My_DB

To obtain the same values by using the placeholder DATABASES, the call must be structured as follows:

- `<DATABASES["My_DB"].NAME>` = My_DB
- `<DATABASES["My_DB"].PATH>` = C:\Users\admin_user\Desktop\My_DB

6.8.1.3 EXPERIMENT/EXPERIMENTS

Name	Context menu	Description
CFG	Experiment - Configuration path	Path to the configuration folder ("config" folder) of the experiment
NAME	Experiment - Name	Name of experiment
PATH	Experiment - Path	Path to the experiment's saving folder
SETTINGS	Experiment - Configuration file	Experiment file including path (*.imcStudio)



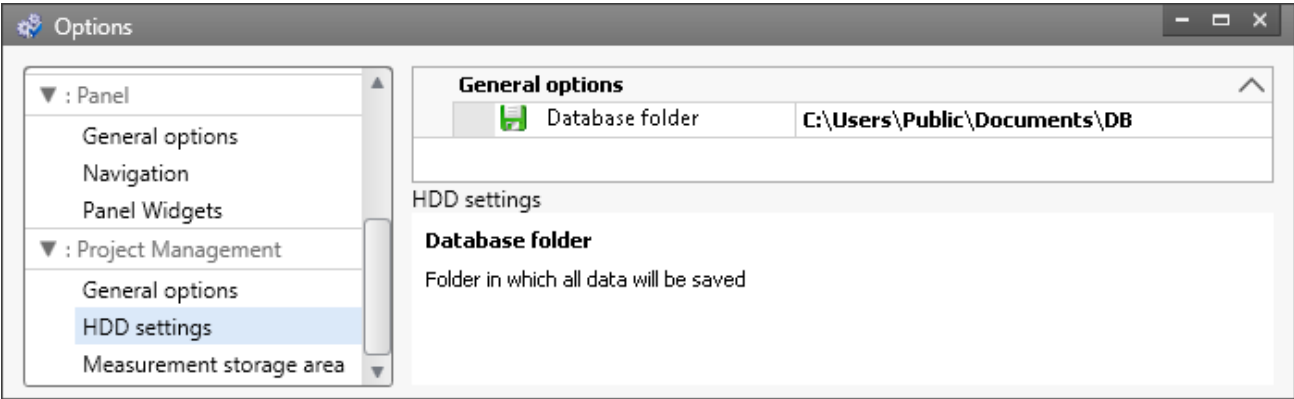
Note

The placeholder *EXPERIMENT* always pertains to the current experiment. If you wish to use the placeholder with another experiment, the placeholder *EXPERIMENTS* must be used, e.g.:

`<EXPERIMENTS["Another_Experiment"].PATH>`

 Example

The example is based on unchanged imc STUDIO default settings, here under Windows 10:



Default database path under Windows 10

The project's name is *StandardProject*, the experiment is called *My_Experiment*.
With these settings, the placeholders listed above have the following values:

Input	Result
<EXPERIMENT.CFG>	C:\Users\Public\Documents\DB\StandardProject\My_Experiment\config
<EXPERIMENT.NAME>	My_Experiment
<EXPERIMENT.PATH>	C:\Users\Public\Documents\DB\StandardProject\My_Experiment
<EXPERIMENT.SETTINGS>	C: \Users\Public\Documents\DB\StandardProject\My_Experiment\config\My_Experiment. imcStudio

To obtain the same values by using the placeholder "*EXPERIMENTS*", the call must be structured as follows:

Input	Result
<EXPERIMENTS["My_Experiment"].CFG>	C:\Users\Public\Documents\DB\StandardProject\My_Experiment\config
<EXPERIMENTS["My_Experiment"].NAME>	My_Experiment
<EXPERIMENTS["My_Experiment"].PATH>	C:\Users\Public\Documents\DB\StandardProject\My_Experiment
<EXPERIMENTS["My_Experiment"].SETTINGS>	C: \Users\Public\Documents\DB\StandardProject\My_Experiment\config\My_Experiment. imcStudio

 Note

Deviating System Behavior

Deviating system behavior in absence of the component: imc STUDIO Project Management:
If the component imc STUDIO Project Management is deactivated, no database is used to manage the experiments jointly. For this reason, the placeholder returns different results.

Name	Context menu	Description
CFG	-	<i>Not supported.</i>
NAME	-	<i>Not supported.</i>
PATH	Experiment - Path	Path to the storage directory of the configuration file ".imcStudio"/".imcExp"
SETTINGS	-	<i>Not supported.</i>

6.8.1.4 MEASUREMENT

With this placeholder, you can access properties and metadata belonging to a saved measurement, such as the directory path, or metadata saved along with the measurement.

Name	Context menu	Description
SQL	Measurement SQL	Access to data of a saved measurement

The placeholder reads properties and metadata belonging to the saved measurements. For this purpose, commands in the database language SQL are used. When the placeholder is entered, an assistant opens automatically, which handles the entering of the SQL command.

Columns

AlterationTime
CreationTime
Name
Path

Condition

And
Name Equals Measurement#1

SELECT 'Path' FROM [Measurement] WHERE [Name] = 'Measurement#1'

OK

MEASUREMENT.SQL-Assistant

In the assistant, there is a choice among various **columns**, which can be associated with a **condition**:

Parameter	Description
Columns	Here, all available properties and metadata are displayed. Select the desired property; multi-selection is possible using the CTRL-key. In the default case, where there is no additional export of data, you can select among the following properties: - the last time the measurement was modified (AlterationTime), - the time the measurement was created (CreationTime), - the measurement's directory path (Path) and - the measurement's name (Name). This always returns the fixed name. If any metadata with the measurement were saved, they are listed here.
Condition	Here you specify which measurement's selected properties and metadata you wish to see. For example, in order to always see the metadata belonging to measurement #1, proceed as follows: - Click on the  , which causes the condition to appear - Next, click on "AlterationTime" and select "Name". - Subsequently, select "Measurement#1" from the selection list behind "equals". If the condition is left empty, the properties/metadata of all measurements are displayed in succession (separated by ',').

 **Note**

Columns and Condition

In the **Columns**, select the **properties/metadata**, which you wish **to display**. In the **Condition**, specify **which measurement's properties/metadata** you wish to use.

 Example

The selection made in the image above returns the directory path of the measurement which is currently designated as #1 (in the Data Browser).

6.8.1.5 MEASUREMENTS

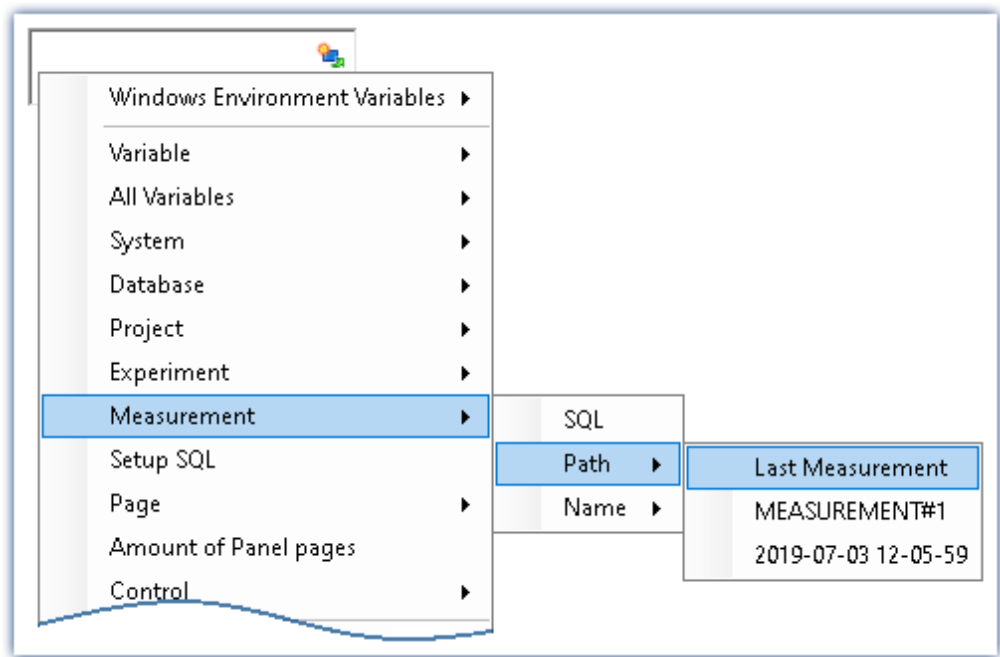
With this placeholder, you can get the path or name of a measurement; e.g. the folder path to the last saved measurement.

Name	Context menu	Description
PATH	Measurement - path	Returns a measurement's folder path
NAME	Measurement - name	Returns a measurement's name

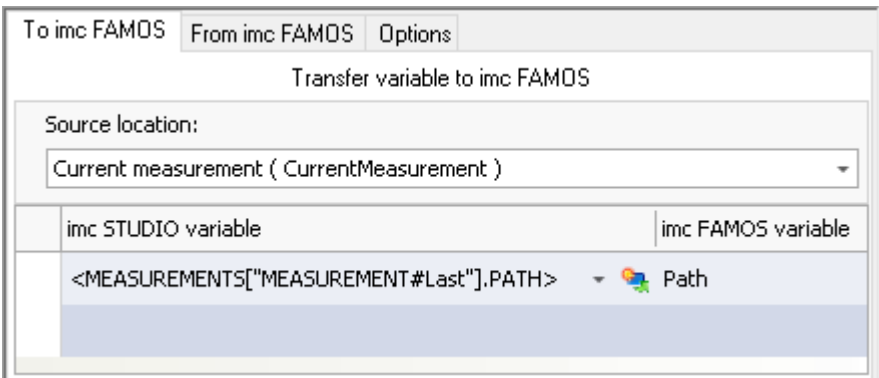
The following MEASUREMENTS-placeholders are available:

Placeholder	Description
<code><MEASUREMENTS["MEASUREMENT#LAST"].PATH></code>	Returns the data storage path to the measurement last saved.
<code><MEASUREMENTS["MEASUREMENT#LAST"].NAME></code>	Returns the name of the measurement last saved.
<code><MEASUREMENTS["MEASUREMENT#<Measurement number>"].PATH></code>	Returns the data storage path of the measurement having the respective measurement number; e.g., you can select a measurement in the Data Browser. For example, it can be assigned the number "1". You now have the ability to find the path to this measurement by means of the placeholder.
<code><MEASUREMENTS["MEASUREMENT#<Measurement number>"].NAME></code>	Returns the name of the measurement having the respective measurement number. For an example, see "PATH".
<code><MEASUREMENTS["<Measurement name>"].PATH></code>	Returns the data storage path of the measurement having the respective name.

Currently, you can't obtain the placeholder via the Placeholder-symbol, since there is no input support. Instead, use the context menu. In this menu, under "*Measurement*" there are various examples which you can select. Subsequently, you can modify the text according to your requirements.



Creating the placeholder via the context menu



Example: Transferring the data storage path to imc FAMOS

 Reference

The last measurement

More detailed information on the "last measurement" is presented in the chapter:
"Storage options and directory structure" > "[The last measurement](#)"

6.8.1.6 PAGE

This placeholder is only available for the Panel.

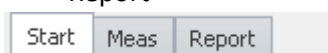
Name	Context menu	Description
NUMBER	Page - Number	Number of the current Panel page
VARS	Page - Variables	All variables which are linked with at least one Widget on the current page



Example

In the Panel, the following pages are generated:

- Start
- Meas
- Report



Existing pages in the Panel

For instance, if the placeholder is used on the page *Start*, then `<PAGE.NUMBER> = 1`, on the page *Report*, `<PAGE.NUMBER> = 3`.

If, for instance, a curve window is linked with *Channel_001* on the page *Start* and a standard meter is linked with the user-defined variable *Test*, then `<PAGE.VARS> = Channel_001,Test`.

6.8.1.7 PANEL

This placeholder is only available for the Panel.

Name	Context menu	Description
PAGECOUNT	Amount of Panel pages	Total number of pages in the Panel



Example

In the Panel, the following pages are generated:

- Start
- Meas
- Report

In which case, `<PANEL.PAGECOUNT> = 3`.

6.8.1.8 PROJECT/PROJECTS

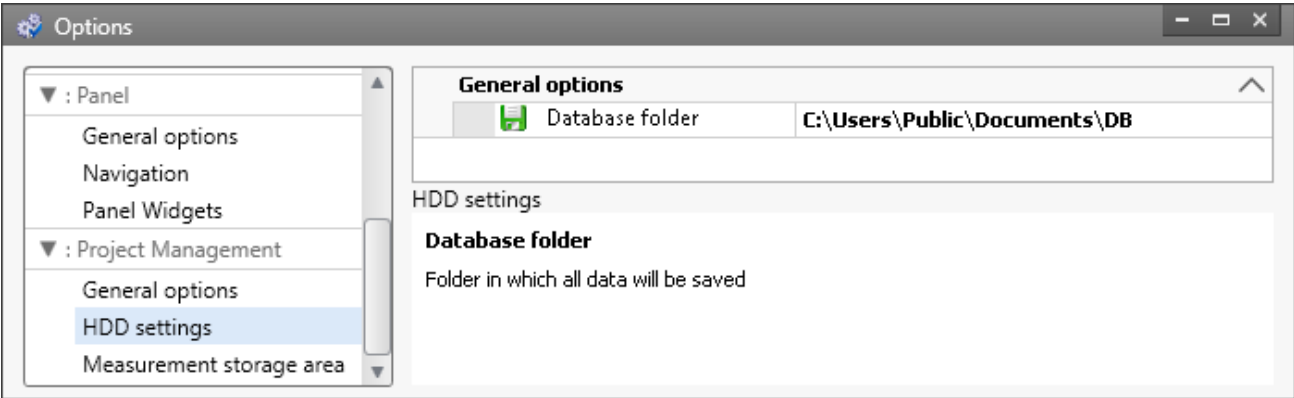
Name	Context menu	Description
CFG	Project - Configuration path	Path to the project's configuration folder (<i>config</i> folder)
NAME	Project - Name	Name of the project
PATH	Project - Path	Path to the project's saving folder
SETTINGS	Project - Configuration file	Settings for the project (*.imcAppSettings file)

 Note

The placeholder *PROJECT* always pertains to the current project. If you wish to use the placeholder with another experiment, the placeholder *PROJECTS* must be used, e.g.: `<PROJECTS["Another_Project"].PATH>`

 Example

This example is based on unchanged imc STUDIO, default settings, here under Windows 10:



Default database path under Windows 10

The project's name is *StandardProject*.
With these settings, the placeholders listed above have the following values:

Input	Result
<code><PROJECT.CFG></code>	C:\Users\Public\Documents\DB\StandardProject\config
<code><PROJECT.NAME></code>	StandardProject
<code><PROJECT.PATH></code>	C:\Users\Public\Documents\DB\StandardProject
<code><PROJECT.SETTINGS></code>	C: \Users\Public\Documents\DB\StandardProject\config\StandardProject.imcAppSettings

To obtain the same values using the placeholder *PROJECTS*, the call must be structured as follows:

Input	Result
<code><PROJECTS["StandardProject"].CFG></code>	C:\Users\Public\Documents\DB\StandardProject\config
<code><PROJECTS["StandardProject"].NAME></code>	StandardProject
<code><PROJECTS["StandardProject"].PATH></code>	C:\Users\Public\Documents\DB\StandardProject
<code><PROJECTS["StandardProject"].SETTINGS></code>	C: \Users\Public\Documents\DB\StandardProject\config\StandardProject.imcAppSettings

6.8.1.9 SELCONTROL

This placeholder is only available for the Panel.

Name	Context menu	Description
VARS	Variables of the selected control	List of all variables linked with the selected Widget

 Example

A curve window is linked with the following channels:

- Channel_001
- Channel_002
- Channel_003

In which case, <SELCONTROL.VARS> = Channel_001,Channel_002,Channel_003.

6.8.1.10 SETUP

By means of this placeholder, you can access data from the Setup.

Name	Context menu	Description
SQL	Setup SQL	Access to current data from the Setup

The placeholder reads current parameter values from the Setup. Therefor, commands expressed in the SQL database language are used. When entering the placeholder, a wizard is automatically opened, which handles making the entries for the SQL command:

Tables

Columns

Condition

Analog and digital channels
Analog channels
AppMod block parameters
Channels
Channels (Balance, Shunt c
Current time
Dependencies
Devices
Digital channels
Diskstart configurations
Experiment
Experiment options

Resulting sampling rate
Resulting sampling time
Sample count
Sampling & Filtering
Sampling rate
Sampling speed
Sampling time
Save (device)
Save (PC)
Saved events (PC)
Scaling factor
Scaling offset

And
Channel name Equals Chan

SELECT 'SampleRate' FROM [Channels] WHERE [eChannelName] = 'Channel_001'

OK

SETUP.SQL wizard

The wizard displays the following columns:

Parameter	Description
Tables	Here all table descriptions are listed, select the one desired. In order for the "SETUP"-placeholder to be able to get the current values from the Setup, it must query the table description. For more details on table descriptions and Setup pages, please observe the reference below.
Columns	Once your table description has been selected, all associated columns (parameters) which are available appear here. Select the parameter desired; it is also possible to select multiple parameters by using the CTRL-key.
Condition	Specify any condition here, for example a particular channel name, if desired. If the condition is left empty, all values belonging to the selected column, e.g. the sampling rates of all channels, are displayed.

 Example

The selection which was made in the picture above returns the sampling rate of the channel "Channel_001" as a text string, e.g. "100 Hz". If the condition which had been entered is deleted, the full amount of sampling rates would be returned as an enumeration, e.g. "100 Hz, 1 kHz, 100 Hz,100 Hz, 1 kHz, , ,100 Hz, ...". Elements which do not possess any sampling rate, return an empty element (" , ").

 Reference

Information on the relationship between table descriptions, complete layouts and Setup pages is provided in the section "[Table description and complete layout](#)".

Since metadata such as entries on the pages "*Documentation*", "*Annotation before measurement*" and "*Annotation after measurement*" belong to the Setup, these can also be displayed in the Panel or used in commands by this means.

Note

The placeholder reads by the SQL command **always the current value** from the Setup, i.e. the value is updated as soon as any change is made in the Setup.

6.8.1.11 SYSTEM

This placeholder enables the current date and the system's time to be used.

Name	Context menu	Description
DATE	System - Date - Standard	The operating system's current date. Format M/d/yyyy for English (USA) operating systems
	System - Date - Path suitable	The operating system's current date, format yyyy-MM-dd
TIME	System - Time - Standard	The operating system's current time, format hh:mm for English (USA) operating systems
	System - Time - Path suitable	The operating system's current time, format hh-mm-ss

Explanations of the formatting of the [date](#)^[112] and the [time](#)^[112] are provided in the section [Formatting](#)^[111].

Example

Supposing the time June 14th, 2011, 2:30 p.m. is set in the (English(USA)) operating system. Then the two placeholders return in their default formatting:

- <SYSTEM.DATE> = 6/14/2011
- <SYSTEM.TIME> = 2:30 PM

The remaining formatting options are described in the section [Formatting](#)^[111].

6.8.1.12 VAR/VARS

Name	Context menu	Description
COMMENT	Variable - Comment	Comment on the variable, e.g. the channel comment
FILE	Variable - File	File, in which the variable is saved
NAME	Variable - Name	Name of the variable
PATH	Variable - Path	Path in which the VARS.FILE is saved
PROPS	Variable - Properties	Several variable properties, e.g. category or metadata, which have been written into channel data.
UNIT	see YUNIT	see YUNIT
VALUE	Variable - Value	Current value of the variable, of a channel or a bit
XUNIT	Variable - X-unit	X - unit of the variable, e.g. s for seconds
YUNIT	Variable - Y-unit	Y - unit of the variable
YUNIT2	Variable - Y-unit2	Y-unit of the 2nd component (only relevant with complex data sets)
ZUNIT	Variable - Z-Unit	Z - unit of the variable

Note

The placeholder *VAR* always refers to the variable which is linked with the Widget. If you wish to access a different variable via the placeholder, the placeholder *VARS* is to be used, e.g.: `<VARS["myVariable"].PATH>`.



Example

A vibration in " μ eps" is measured over time (seconds). In imc Online FAMOS, an FFT of the input signal is performed.

The resulting virtual channel is saved together with the measurement channel on the PC.

Data storage settings:

- Path: C:\Users\Public\Documents\DB\StandardProject\My_Experiment
- Storage interval: End of measurement

The virtual channel's name in this example is "*myFFT*" and the comment is called "*Result of FFT*".

Thus, the placeholders listed above are resolved after the first measurement as follows:

Input	Result
<VARS["myFFT"].COMMENT>	Result of FFT
<VARS["myFFT"].FILE>	myFFT.RAW
<VARS["myFFT"].NAME>	myFFT
<VARS["myFFT"].PATH>	C:\Users\Public\Documents\DB\StandardProject\My_Experiment\2019-12-18 09-37-14
<VARS["myFFT"].UNIT>	μ eps
<VARS["myFFT"].XUNIT>	s
<VARS["myFFT"].YUNIT>	μ eps
<VARS["myFFT"].ZUNIT>	Hz

<VARS["myFFT"].VALUE> is resolved, however in this example (FFT-calculation), it does not return a sensible value. Notes on formatting the placeholder VALUE are found under "[Formatting - VALUE](#)".

6.8.2 Formatting

With some placeholders, you have the possibility of specifying the formatting desired. The various formatting options are described as examples.

Usually, manual entry of the formatting string is necessary. The placeholder "VALUE" offers you support in making entries.

A formatting is possible with the following placeholders:

- [PAGE.NUMBER/PANEL.PAGECOUNT](#)
- [SYSTEM.DATE](#)
- [SYSTEM.TIME](#)
- [VALUE](#)

6.8.2.1 PAGE.NUMBER/PANEL.PAGECOUNT

The placeholders PAGE.NUMBER and PANEL.PAGECOUNT are integers where the amount of digits can be set.

If, for instance, <PANEL.PAGECOUNT> has the value 3, then this value may be outputted as follows:

- <PANEL.PAGECOUNT("00")> = 03
- <PANEL.PAGECOUNT("000")> = 003

These formatting strings can be used for <PAGE.NUMBER> in the same way.

6.8.2.2 SYSTEM.DATE

In the formatting string for the date, the following components can be used:

- d or dd = current date
- M or MM = current month
- yy or yyyy = current year
- Separator e.g. '.', ',', '-', '_'

Note

To use the placeholder when specifying the path as a string, use the separator "-". This character is allowed as a special character in all languages in pathnames.

Example

With the example of the date July 14th, 2019 (operating system English (USA)), the use of the formatting string is illustrated.

- `<SYSTEM.DATE>` = 7/14/2019
- `<SYSTEM.DATE("yyyy-MM-dd")>` = 2019-07-14
- `<SYSTEM.DATE("yy-M-d")>` = 19-7-14
- `<SYSTEM.DATE("dd_MM_yy")>` = 14_07_18

6.8.2.3 SYSTEM.TIME

In the formatting string for the time, the following components can be used:

- h or hh = current hour; 12h
- H or HH = current hour; 24h
- m or mm = current minute
- s or ss = current seconds
- Separator e.g. '.', ',', '-', '_'

Note

To use the placeholder when specifying the path as a string, use the separator "-". This character is allowed as a special character in all languages in pathnames.

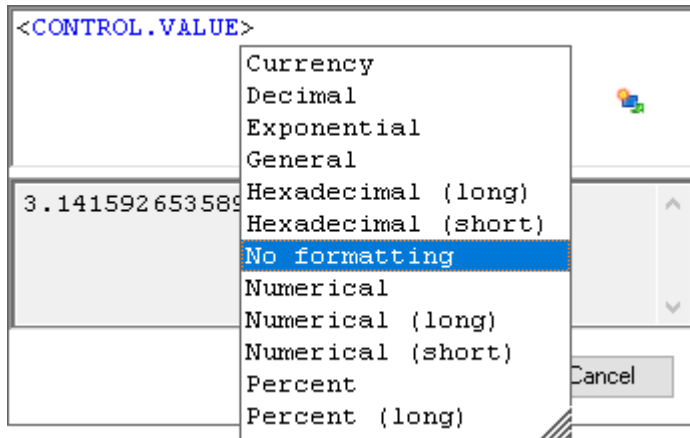
Example

With the example of the time 1:05:03 p.m. (operating system English (USA)), the use of the formatting string is illustrated.

- `<SYSTEM.TIME>` = 1:05
- `<SYSTEM.TIME("hh-mm-ss")>` = 01-05-03
- `<SYSTEM.TIME("h-m-s")>` = 1-5-3
- `<SYSTEM.TIME("HH_mm_ss")>` = 13_05_03

6.8.2.4 VALUE

Formatting for VALUE can either be entered by means of the input assistance (see figure below) or manually.



Input assistant for the formatting

The various formatting possibilities are explained in the following table on the basis of the variable $\pi=3.1415926535$. The amount of digits before and after the decimal specified here can be replaced with any desired integer amount.

Description	Input assistant	Manual	Output
No formatting	No formatting	<code><VARS.["pi"].VALUE></code>	3.14159
Fixed point, 2 decimal digits	Numeric	<code><VARS.["pi"].VALUE("f2")></code> <code><VARS.["pi"].VALUE("0.00")></code>	3.14
Fixed point, 1 decimal digit	Numeric Short	<code><VARS.["pi"].VALUE("0.0")></code>	3.1
Fixed point, 6 decimal digits	Numeric Long	<code><VARS.["pi"].VALUE("0.000000")></code>	3.141593
Floating point, exponential notation, 2 decimal digits	Exponential	<code><VARS.["pi"].VALUE("e2")></code>	3.14e+000
Fixed point, 2 decimal digits, always output sign		<code><VARS.["pi"].VALUE("#.##; -#.##")></code>	+3.14
Fixed point, 2 integer digits, 1 decimal digit, always output sign, no output of significant zeroes		<code><VARS.["pi"].VALUE("+00.0;-00.0")></code>	+3.14
Space before positive values, "-" before negative values		<code><VARS.["pi"].VALUE(" 0.00;-0.00")></code>	3.14
General	General	<code><VARS.["pi"].VALUE("g")></code>	3.1415926535
Fixed point, 4 decimal digits	Decimal	<code><VARS.["pi"].VALUE("0.0000")></code>	3.142
Hexadecimal	Hexadecimal Short	<code><VARS.["pi"].VALUE("x4")></code>	0003
	Hexadecimal Long	<code><VARS.["pi"].VALUE("x8")></code>	0003
Currency	Currency	<code><VARS.["pi"].VALUE("c2")></code>	3.14
Percent	Percent	<code><VARS.["pi"].VALUE("0.0%")></code>	314%
	Percent Long	<code><VARS.["pi"].VALUE("0.00%")></code>	314.16%

Instead of the decimal point comma can also be used. Then, the comma is displayed in the output, e.g.:

- `<VARS.["pi"].VALUE("0,00")> = 3,14`

With integer values, formatting is also possible. For instance, if a variable "*amount*" has the instantaneous value 4, then this value can be outputted also as 04, 004, etc.:

- `<VARS.["amount"].VALUE("00")> = 04`
- `<VARS.["amount"].VALUE("000")> = 004`

6.8.3 Environment variables of the OS

In imc STUDIO Monitor it is also possible to access the operating system's environment variables.

Tips on use:

- Set the variable to be resolved in angle brackets with percent symbols, as follows: `<%USERNAME%>`
- Observe capitalization. The variable must be spelled exactly as it is spelled by the operating system, e.g. *ProgramData*, *HOMEDRIVE* or *windir*.

7 Setup - Device (general)

Setup is the imc STUDIO Monitor component for uniform configuration and control of imc measurement devices.

Chapter overview

Synopsis	Section
Which devices are supported?	Device Overview 
Getting familiar with concepts concerning system operation	Operation 
What steps are necessary for performing a measurement? What actions are available?	Performing a measurement: Procedure  Ribbon 
Saving supplemental information on the measured data	Saving metadata with the channel 
Designing custom columns and user interfaces	Setup-Layout  Creating and configuring additional columns 

 **Reference**

Configuring the device and measurement

With the Setup pages , you can configure one or more measurement quickly and easily.

The extensive data saving, triggering and real-time functions are organized clearly for every device. The measurement parameters can be set individually for each channel. The measured data can be saved separately for each channel. Measurement channels can be monitored continually and only start recording upon the occurrence of specified events.

See: "[Setup pages - Configuring device](#)" 

 **Note**

If the documentation mentions a device, the comments apply similarly to other devices as well. All screenshots were created using the "*Complete*" view.

7.1 Device overview

Some of the capabilities discussed in this document only pertain to certain device models. The associated device groups are indicated at the respective locations. The groups are shown in the following table which can be used with imc STUDIO Monitor.

— not available • standard ○ optional
CRXT imc CRONOS-XT CRFX imc CRONOSflex CRC imc CRONOScompact

imc device	SPARTAN	BUSDAQ	BUSLOGflex	BUSDAQflex	SPARTAN-R SPARTAN-N	CRSL-N CRC-400	C1-N C-SERIE-N	C1-FD C-SERIE-FD	CRFX-400	CRFX-2000	CRC-2000G	CRC-400GP	CRFX-2000G	CRFX-2000GP	CRXT
Driver package	imc DEVICES														
Firmware group	A														
Device group	A4				A5				A6	A7					
SN ¹	13				14				16	19					
TCP/IP Interface [MBit/s]	100				100				100	1000					
Sampl.Rate ² [kHz]	400				400				2000 / 400 ³	2000 / 400 ³	2000 / 400 ³	2000	2000	2000	
STUDIO Monitor supported	●				●				●	●					
Connections ⁴	4				4				4	4					
Signal processing in the device															
Online FAMOS	○	○	—	○	○	●	●	●	●	●	●				
Preprocessing original channel	●	—	—	—	●	●	●	●	—	●	●	●	—	—	—
Preprocessing monitor channel	●	—	—	—	●	●	●	●	—	●	●	●	—	—	—
Data Storage															
CF	●				●				—	—					
Express Card	—				—				●	—					
CFast	—				—				—	●					
USB	—				—				●	●	●	●	●	—	
microSD	—				—				—	—					
Storage on network drive	●				●				●	●					
Internal hard disk	○	(○) ⁵	—	—	○	○	—	—	○	○	○				
Synchronization															
DCF	●				●				●	●					
IRIG-B	—	—	●	●	●				●	●					
GPS	●	●	—	(●) ⁶	●				●	●					
NTP	—	—	●	●	●				●	●					
PTP	—				—				—	—	●	—	●	●	
Phase offset correction	—	—	●	●	●				●	●					

1 Extend serial number range by four digits (three for imc EOS)

2 Max. aggregate sampling rate (see data sheet)

3 2000 via EtherCAT else 400

4 Number of imc STUDIO Monitor-connections or imc REMOTE (as of 14xxxx) connections

5 not available for imc BUSDAQ-2

6 not available for imc BUSDAQflex-2-S

7.2 Ribbon

The ribbon of "imc STUDIO Setup".

7.2.1 Control

Table filter

Menu item	Description
 Hide Passive Channels	No function

Device Control

Menu item	Description
 Process configuration 	The current configuration is inspected for errors and made available to all plug-ins. For this purpose, no connection to the device is required.
 Start measurement  (start monitoring)	Starts monitoring for the measurement devices selected.
 Stop measurement  (stop monitoring)	Stops monitoring for the measurement devices selected.
 Suspend data storage 	If data transfer to storage is activated, it is suspended for all devices.
 Resume data storage 	If data transfer to storage is activated, it is resumed for all devices.
 Search for devices	No function

7.2.1.1 Process configuration

The current configuration is inspected for errors and made available to all plug-ins. Connection to the devices is not required for this purpose.

 **Note**

There are cases where certain settings are possible to make during operation while not being actually supported by the respective device type. In such a case, a corresponding error message will be posted.

Ribbon	View
Home > Process ()	all
Setup-Control > Process ()	Complete

7.2.1.2 Start and Stop measurement

The measurement (monitoring) is started/stopped for all devices.

Note

- For starting and stopping the measurement, the PC needs to have established a connection with the device.

Ribbon	View
Home > Start ()	all
Setup-Control > Start ()	Complete
Home > Stop ()	all
Setup-Control > Stop ()	Complete

Measurement status (running, stopped)

The measurement's status is denoted e.g. by the stop-symbol (if imc STUDIO Monitor is connected with the device):

Icon	Description
	Measurement running (solid red)
	Measurement stopped (color depends on the Windows setting)

7.2.1.3 Suspend and Resume data storage

If the [saving of measured data is activated](#) ¹⁸⁸, then you are able to govern the data storage **during the running measurement**. You can **interrupt (suspend) data storage** and **resume it again** at a later time.

Ribbon	View
Home > Suspend data storage ()	all
Setup-Control > Suspend data storage ()	Complete
Home > Resume data storage ()	all
Setup-Control > Resume data storage ()	Complete

Effect

If you **suspend data storage**, the **files are closed** and appear in the Data Browser (as if the measurement had been concluded). Any **more data** which arrive or are calculated subsequently **are not saved**.

When you **resume data storage**, then **new files are created** in a **new measurement folder** (as if the measurement had just been started). Any data arriving before "resumption of data saving" are not saved.

This function is designed for **interrupting data saving for longer periods of time**. Please do **not use it to make "cuts" of data records** (concluding one measurement folder and immediately beginning a new one).

 **Note**

- Data storage **con only be controlled** if it was previously **activated**.
- This **suspension/resumption** of data storage **is not synchronized** across the devices! Thus, the respective **data excerpt boundary points** of different devices/channels **may differ by a few measurement points**.

Scope

These actions **apply to all devices and to the PC**. No separation is possible. Any additional components which return data to be saved (such as 3rd-party devices) are also affected.

 **Note**

Video-files are not affected

The **saving of Video files** is **not affected** by this function. Video files are always recorded.

 **Reference**

Data storage state after preparation

Using the option "Data storage state after preparation", you can control the function's state. E.g. it is possible to make data storage generally active following "Preparation".

See the option: "Setup" > "General" > "[Data storage state after preparation](#)"⁶⁵".

7.2.2 Configuration

 **Note**

In some dialogs (e.g. imc Online FAMOS), it is possible to make device-dependent settings. In that case, it is necessary to make a selection in the dialog of the device to which the settings pertain.

Assistants

Menu item	Description
 imc Inline FAMOS	This starts the imc Inline FAMOS dialog. For a detailed description, see the chapter: " imc Inline FAMOS " ²⁴⁵ .

Device configuration

Menu item	Description
 Default Values ¹²¹	These are preset values for the configurations of the devices and channel. These values are applied upon selection of devices and when creating new device variables/channels or Data Processing variables.
Supplemental Files ¹²⁰	Shows the dialog to manage supplemental files. Here you can export, import and view existing additional files.

7.2.2.1 Supplemental files

Supplemental files are **files saved in the device**. For instance, it is possible to **access** a characteristic curve supplemental file **with imc Online FAMOS**.

Additionally, you can also import a supplemental file **for a imc Inline FAMOS-task**. An imc Inline FAMOS-task is treated exactly the same as a device. Anything stated below about devices also applies to any imc Inline FAMOS-task.

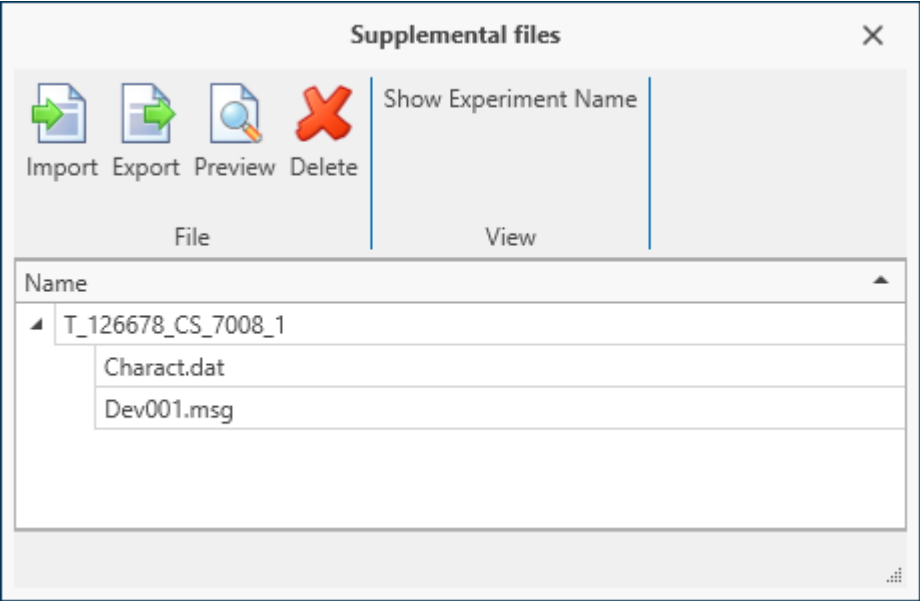
The following supplemental file type are available:

- Characteristic curves and filter data for imc Inline FAMOS (*.dat)

Ribbon	View
Setup-Configuration > Supplemental Files	Complete

The Supplemental files dialog

In this dialog, all existing supplemental files are listed according to their associated selected device. In the example below, there is a device having a characteristic curve file (Charact.dat) and a messaging configuration file (Dev001.msg).



Dialog: Supplemental Files (example)

By means of this dialog, supplemental files can be:

- deleted,
- displayed (Preview),
- exported,
- and new files can be imported (also possible by means of the menu ribbon item "Project" > "[Import / Export](#)")

Import

- After clicking on "import", **select the file desired** from the file selection dialog.
- If you use **multiple devices** in the experiment, then after selection of the file, a device selection dialog appears. Here, you can **select to which devices to import** the file.

Imported supplemental files are saved in the Experiment-file. When the **measurement is prepared**, the supplemental files are **copied to the device memory** (except imc Inline FAMOS-supplemental files). In this way, it is possible for imc Online FAMOS, for example, to access the file without needing to be connected with the PC.

Renaming

Click the mouse on the name of a file to be imported. A cursor appears, and you can rename the file.

Preview

Using the function "Preview", an **external program** is started, which opens the file selected.

Changes which were made by means of the external program **can not be applied** automatically in the existing supplemental file. If you wish to save the changes made, save the file as a copy in a different folder, and then import this new file.

7.2.2.2 Default values

Default values are pre-set values for the configurations (parameters) of the devices and channels (e.g. saving to the device). It is possible to specify either individual parameters or whole groups of parameters.

The default values are applied when the device is selected and when creating new device variables/channels as well as Data Processing-variables (e.g. for virtual channels and Fieldbus channels).

When does it make sense to alter the default values?

When the default settings of the devices and channels are not appropriate for most of your applications.

Ribbon	View
Setup-Configuration > Default Values 	Complete

In this dialog, all available default values are listed.

Example:

- For all channels, the channel saving on the PC should be activated. It is possible to later deactivate data saving for each channel separately, but after having been set up, each channel should start off being saved.
- The circular buffer for the curve window should always be set to "Undefined" after selection of the device.
- The interval storage should always be activated after selection of the device.

Note

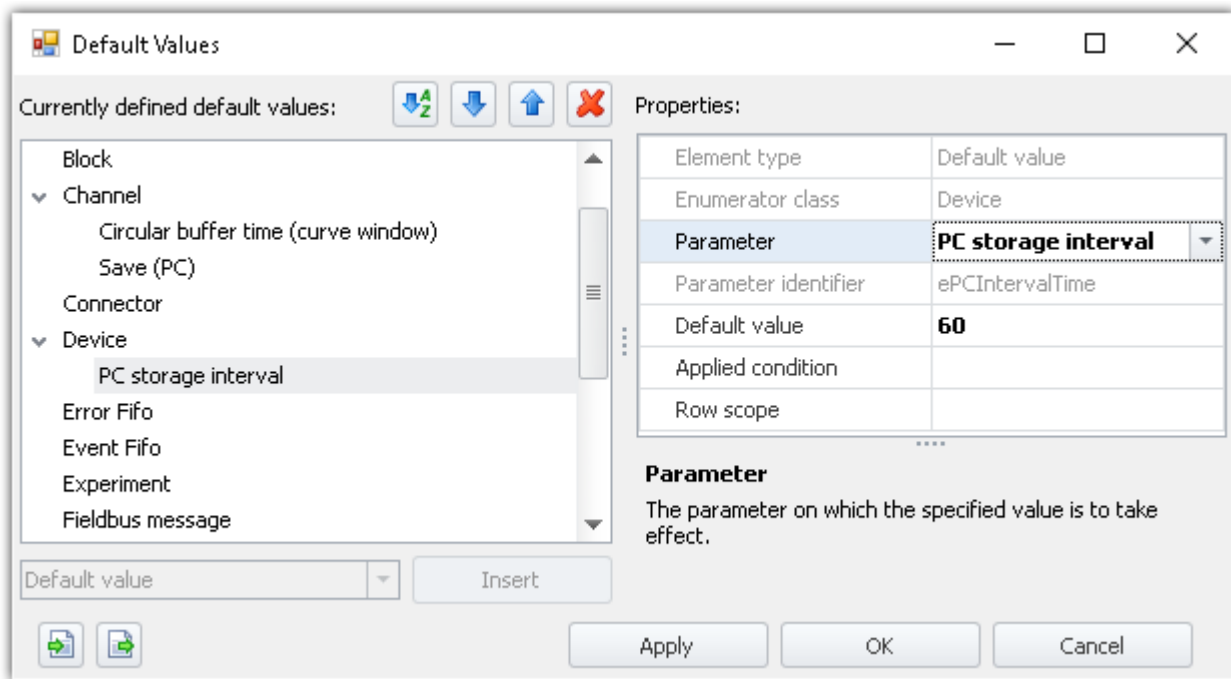
Ensure that these settings are set initially (e.g. upon device selection). Subsequently, every parameter can be modified.

Reference

Tip

To learn about a simple handling technique, observe the tip further below: "[Generating default values from the current settings](#)".

With the default settings, it is possible to set parameters in dependence on other parameters.



Example of default values: "PC storage interval" is set to "60 s"

Creating default values or groups of default values

In order to create "Default values" or "groups of default values", open the dialog: "Default values" (Ribbon "Setup-Configuration" > "Default Values").

- In the left-side portion of the dialog, select an appropriate Enumerator-class (application area of the parameters). A list of the classes and the areas they contain is presented in the chapter "[Enumerator-class](#)".
- In the drop-down selection list (lower left), select "Default value" or "Default value group".
- Click on "Insert" and the element is created at the location selected.
- In the "Properties" windows, you can set the target parameter, the default value to be set, and any other settings such as the applied condition, etc.

Properties

Property	Description
Element type	Makes the distinction between a default value and a group of default values. (write-protected)
Enumerator class	Specifies in which Enumerator-class the element belongs. (write-protected)
Parameter	The parameter to which the default value is to be applied. Select the parameter here.
Parameter identifier	The parameter's unique designation. (write-protected)
Default value	The value to which the specified parameter is to be set. E.g., when selecting a device or when creating virtual channels.
Applied condition	Here, you can define various logical conditions which allow default values to be set in certain cases. See also " Order and dependencies of the default values ".
Row scope	Determines to which rows the default value is to be applied. Specifies row numbers, separated by commas, or ranges by using the minus sign. The first row has the index: "1". Example: 1-4: The first four parameters found are set. Thus for instance, an amplifier's first four channels can be modified, and the last four by specifying "5-8". The following is also possible: 1,6-8. The order does not necessarily match the table sorting arrangement displayed. For instance, after the analog channel "[01] IN01", the associated monitor channel appears. For this reason, use a well-defined "<i>Applied condition</i>" in order to set clear boundaries. For example, among others the channel type should be included.
Group name	The group mane displayed. The name can be entered to be particular to selected languages. The respective name is then displayed in accordance with the particular language setting. If the (non-English) name is empty, then the English name is always used.

Order and dependencies of the default values - Sorting and deleting

To **sort** entries, use the "up/down" arrow-symbols ( / ) or move them by means of Drag&Drop.

To **delete** entries, use the "X" symbol ().

Sequential order

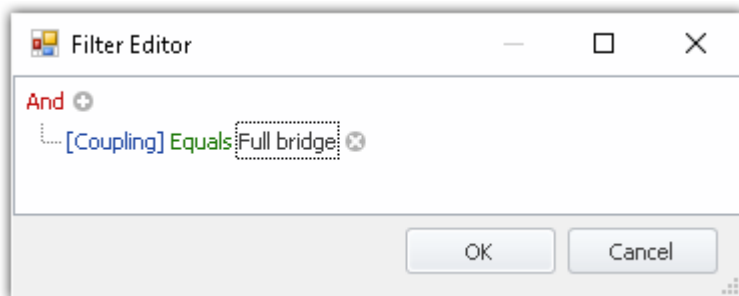
The order of the entries affects the results. The default values are set from top to bottom in succession. Since the values can be set to depend on other parameters, this is important to take into account.

Example:

The balance action is to be set to "Bridge" for all channels having the coupling "Half bridge". However, beforehand all full bridges must be set as half bridges.

Setting up dependencies

The dependency is defined by means of the property "**Applied condition**" using the Filter Editor. Here, you can enter various logical conditions. In the example mentioned above, that would be the following logical condition:



In order for the dependencies to be language independent, they are resolved in the list into the internal parameters. In this case: ([eCoupling] = 5).

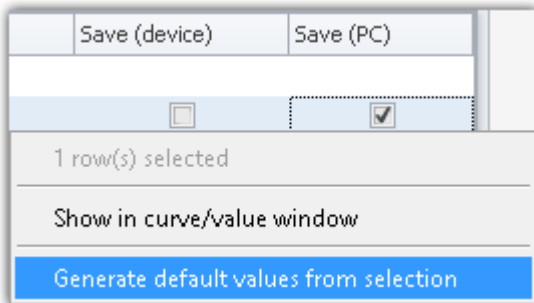
Dependencies can be defined either for each default value separately or for a whole group of default values. In this case, add a "Default value group" and add the respective parameters to the group. Define the application condition for the group. The individual default values do not require any application conditions of their own, but may also have separate conditions.

Generating default values from the current settings (tip)

It is often tedious and not so easy to set the default values correctly. Sometimes it is more reliable if the internal specifications are used to set the measurement range, for example.

In order to get familiar with the default values or to set up many parameters quickly, it is best to generate the default values from existing settings. By means of the channel table, you can define the selection as a default value.

To do this, open the context menu and select: "*Generate default values from selection*".



Context menu of a cell
Generate default values from selection

A dialog for setting the Default value group provides help in finding an appropriate condition. Select the condition for which the default value is to apply. The condition is entered in the generated group as the applicability condition.

Example:

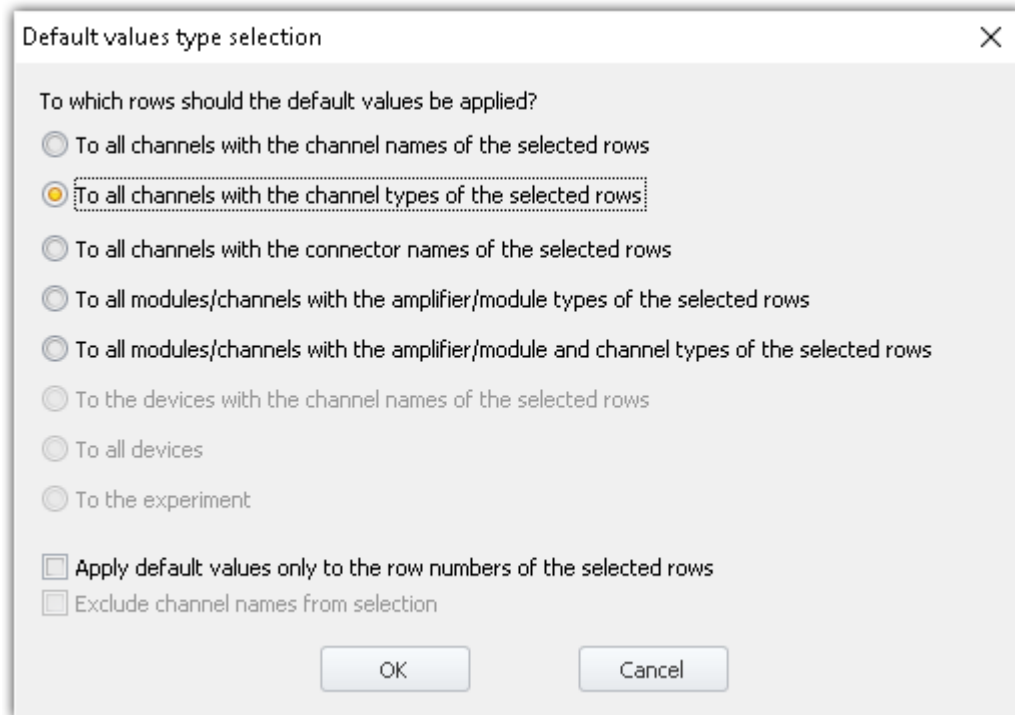
Make the following parameter settings for an analog channel:

Parameter	Value	Dialog (View: <i>Standard</i>)	Dialog (View: <i>Complete</i>)
Coupling	Half bridge	Measurement mode	Measurement mode
Input range	± 10 "mV/V"	Measurement mode	Range & Scaling
Sampling rate	10 kHz	Sampling & Filtering	Sampling & Preprocessing
Save data (PC)	activate	Data transfer	Data transfer

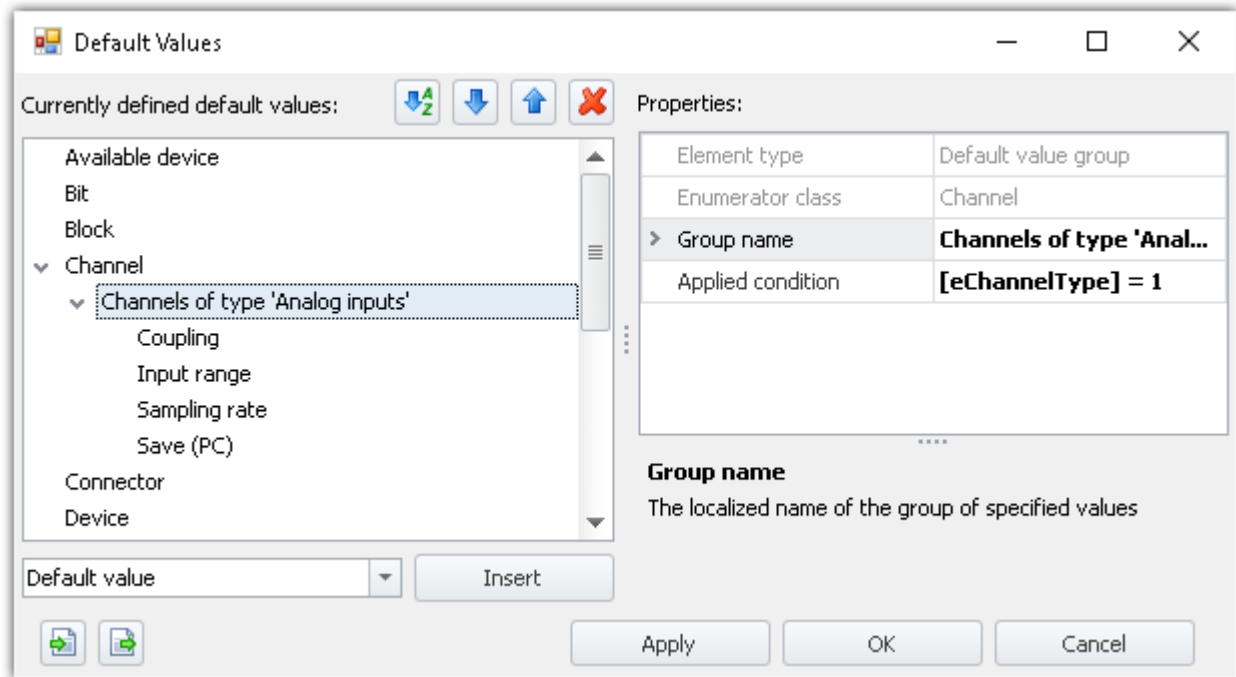
These settings are also found in the channel table in various columns. If you generate default values from a [combined column](#)¹⁵⁴ (one with multiple parameters), all of the column's parameter are set as the default value. In this case, either you have the desired parameter [displayed](#)¹⁴⁸ as a normal column and create the default value only from it, or you can later delete any default values not required.

Below, the second way is described:

- Select the (combined) columns: "Measurement mode", "Range & Scaling", "Sampling & Filtering" and "Save (PC)"
- Open the context menu and select: *"Generate default values from selection"*
- In the subsequent dialog for selecting the condition, select, for instance *"To all channels with the channel types of the selected rows"*. Thus, a group is generated with the applicability condition: *"Channels of type 'Analog inputs'"*



- Open the dialog "Default values" and delete from the group generated all parameters which are not to be set.



*Example: Setting default values for multiple parameters.
But only for the channels of type 'Analog inputs'*

- Upon the next device selection, the analog channels are modified accordingly.



Note

This example serves illustration purposes and is incomplete!

In any case, ensure whether every analog channel you use can really be used as a bridge amplifier. In that case, use the additional condition that only such channels are modified which can be set as a full bridge.

The pre-defined conditions

Conditions	Examples
To all channels with the channel names of the selected rows	"Channel name" equals "Channel_001"
To all channels with the channel types of the selected rows	"Channel type" equals "Analog inputs"
To all channels with the connector names of the selected rows	"Connector" equals "[01] IN01"
To all modules/channels with the amplifier/module types of the selected rows	"Module type" equals "Bridge, current and voltage amplifier and temperature measurement unit "UNI2-8""
To all modules/channels with the amplifier/module and channel types of the selected rows	"Module type" equals "Bridge, current and voltage amplifier and temperature measurement unit "UNI2-8"" AND "Channel type" equals "Analog inputs"
To the devices with the device names of the selected rows	"Device name" equals "imc_CS_7008_1"
To all devices	empty (thus, all devices are affected)
To the experiment	
Additional options	Description
Apply default values only to the row numbers of the selected rows	In the default values, the property: "Row scope" is set accordingly. E.g., the value "2" is entered if the second analog channel was selected.
Exclude channel names from selection	The channel name is not set as the default value even if it is selected.



Note

Saving the default values

The **configuration of the default values** is saved with the respective **project**. Thus, the default values are available in all the project's experiments.

These settings are only saved if the "**project**" is also saved.

Transferring default values to other PCs or installations (import/export)

In order to transfer the default values, use the Import/Export icons( / ).



Warning

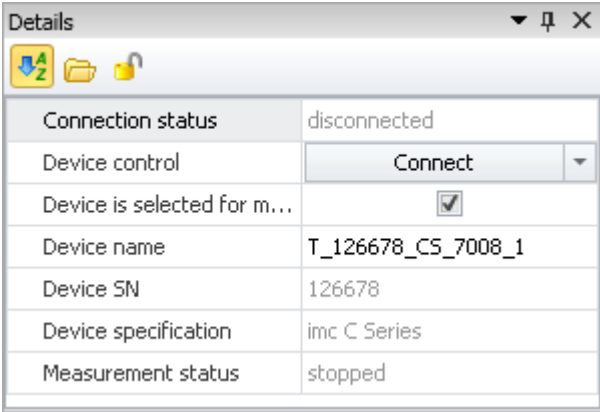
All existing entries will be deleted

If any default value already exist at the time of import, they will be deleted. After the import, only the imported default values are available.

7.3 Tool Windows

7.3.1 Details

The content of the "Details" window depends on which page is selected. As in the "[tabular display](#)¹³²", you can also [directly edit](#)¹⁴⁰ parameters in this tool window.



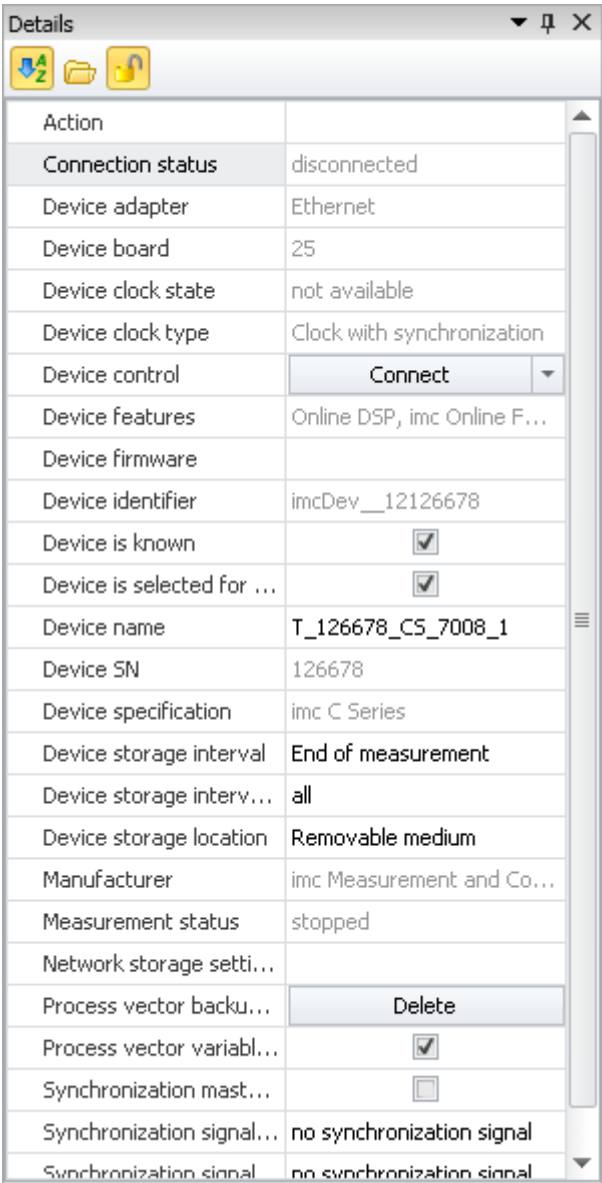
Tool window "Details" in the plug-in "imc STUDIO Setup" (example)

Hidden parameters

By default, the Details window sows a selected set of parameters which can also be seen in the tabular display and the dialogs. By clicking on the **padlock symbol** (🔒), all other parameters are also displayed which belong to the selections in the tabular display ("hidden parameters").

Example

When you click on the padlock symbol, you see the much more extensive list of all device parameters. The image below shows only an excerpt:



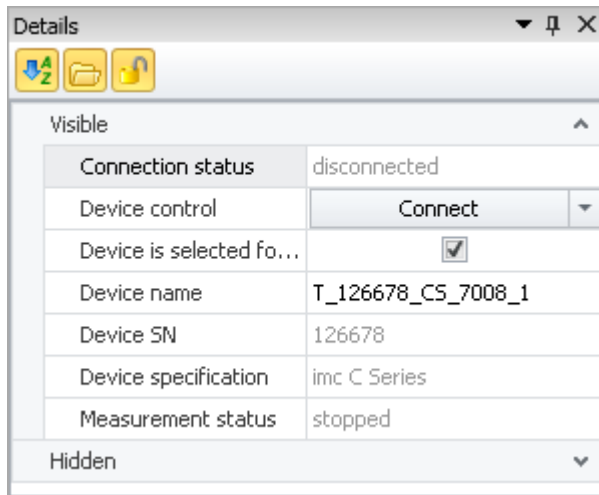
"Details" tool window for a selected device with display of the hidden parameters

In this display, you can also set parameters which are not available either in the tabular display or in the dialogs.

Grouping / sorting parameters

The parameters in the "Details" window can also be displayed in groups. To do this, click on the grouping symbol (). The parameters are then divided into the following groups:

- Visible
- Hidden



"Details" tool window in grouped display

The groups can be opened or closed individually (symbols:  ).

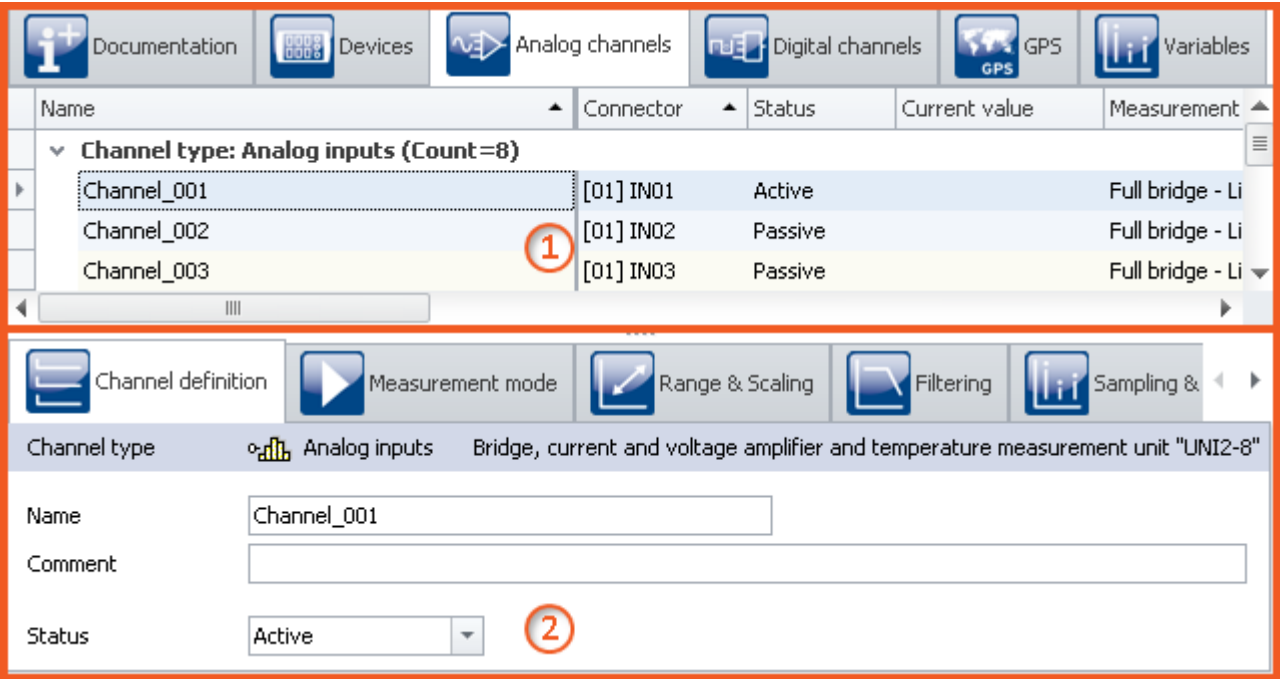
To sort the parameters alphabetically, click on the sorting symbol (). Click again to change the sorting order.

7.3.2 Layout repository

Complete layouts: Here you will find a list of all available Setup pages. In order to display a page, use Drag&Drop to move it to the tab-bar (or use the context menu - see "[Additional Pages](#)" ¹⁹²).

7.4 Operation

The Setup main window is subdivided into several predefined pages. Some pages are divided in two windows stacked vertically, as shown in the example below:



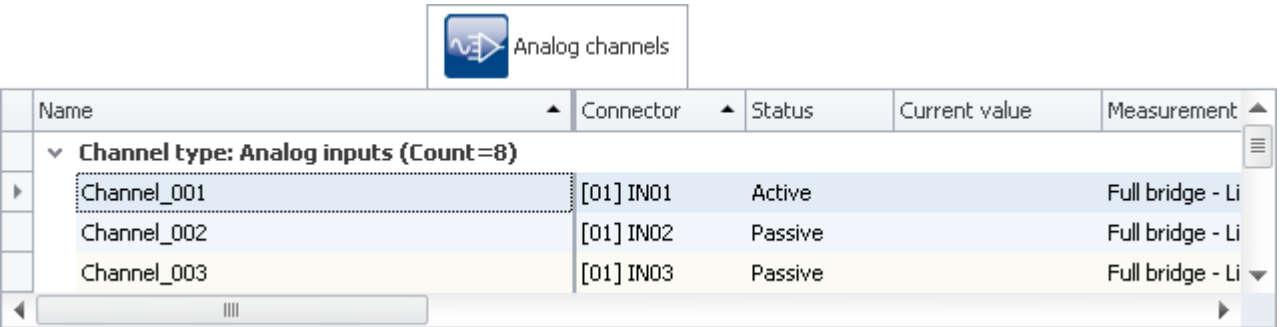
Tabular display and dialog
(example)

Region	Description
1: Tabular display ¹³²	The top window shows a tabular (rows/columns/cells). In the table, e.g. all known devices or channels/variables are listed. It offers a quick overview of parameters and can be filled with many columns.
2: Dialogs ¹³⁹	The lower window shows special dialogs ¹³⁹ for the most important parameters.
Assistenten ¹⁴¹	The Assistants are an aid in configuring particular device components such as Fieldbusses or imc Online FAMOS.

Editing: You can modify the parameters by means of [in-place editing](#) ¹³⁵ in the table, as well as [in the dialog](#) ¹⁴⁰. To edit the height of the window, use the mouse to drag the dashed separator line.

7.4.1 Tabular display

In the tabular display, many parameters of the selected page are displayed in a segmented table.



Example of tabular "Analog channels"

You can [edit](#)^[135] both individual parameters and sets of multiple parameters (see [Selection](#)^[133]) directly in the table cells. The availability and selection of settings depends on the page opened and on the selection in the table.

Other parameters: In principle, all parameters can be configured using the tabular. For this purpose, you can use the [Column Chooser](#)^[136] to include the desired parameters in the table. Also, you can configure complex columns using the function "[Additional columns](#)"^[151].

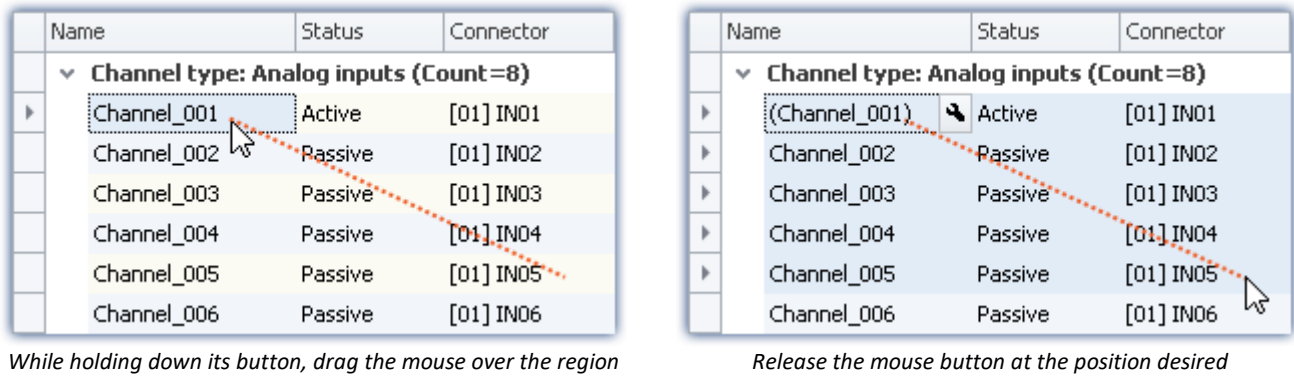
7.4.1.1 Selection

In the tabular display, you can edit both individual parameters as well as sets of multiple parameters **directly in table cell**.

To do this, first select the row(s) which contain the parameters.

Selection with the mouse

To select contiguous cells, you only need to drag open a rectangle over the desired region with the mouse.

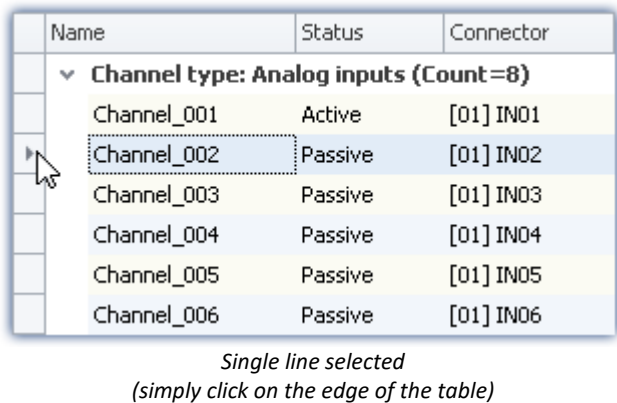


Selection with mouse and keyboard

To do this, use the left margin column, as shown in the first image.

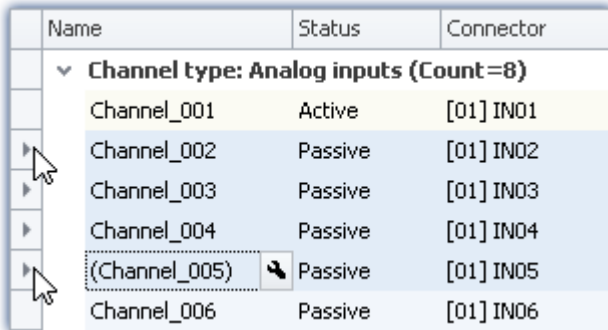
Selecting individual rows

Click on the left edge of the desired entry:



Selecting multiple, consecutive rows

Click on the first row which you wish to select. Then hold down the **<Shift>** key while clicking on the last row you wish to include in the selection. This selects all the rows from the first through the last.



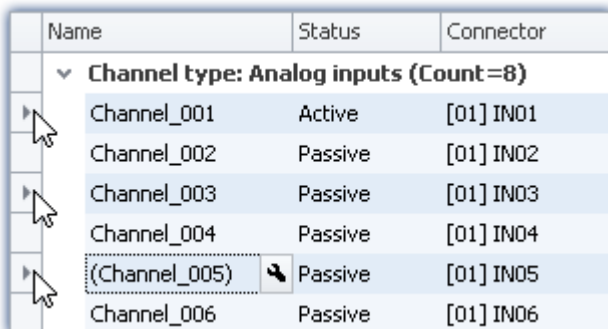
Name	Status	Connector
▼ Channel type: Analog inputs (Count=8)		
Channel_001	Active	[01] IN01
Channel_002	Passive	[01] IN02
Channel_003	Passive	[01] IN03
Channel_004	Passive	[01] IN04
(Channel_005)	Passive	[01] IN05
Channel_006	Passive	[01] IN06

*Selecting multiple consecutive lines
(hold down <Shift>-key)*

You can also perform this procedure repeatedly (in multiple steps). To access more entries, you can use the scroll bar to move through the table. This means that you can also select lines in the table which are so far apart that they are not visible at the same time.

Selecting multiple rows separately

Click on the first row you wish to select. Then hold the **<Ctrl>** - key down and click on the next row which you wish to add to the selection. Each row you click on while holding down the **<Ctrl>** - key will be added to the selection.



Name	Status	Connector
▼ Channel type: Analog inputs (Count=8)		
Channel_001	Active	[01] IN01
Channel_002	Passive	[01] IN02
Channel_003	Passive	[01] IN03
Channel_004	Passive	[01] IN04
(Channel_005)	Passive	[01] IN05
Channel_006	Passive	[01] IN06

*Selecting multiple lines separately
(hold down <Ctrl>-key)*

You can repeat this procedure (in multiple steps). To access more entries, you can use the scroll bar to move through the table. This means that you can also select rows in the table which are so far apart that they are not visible at the same time.

Cancel selection

To cancel a multiple selection, click on the left side of a single entry in the table

7.4.1.2 In-place editing

All parameters for which there is a column in the table can be edited in place (directly in the table). Note that there are [dialogs](#)¹³⁹ for selected parameters.

Editing single parameters

Name	Status	Connector
▼ Channel type: Analog inputs (Count=8)		
Channel_001	Active	[01] IN01
Channel_002	Active	[01] IN02
Channel_003	Passive	[01] IN03

Editing single parameters:
click on the row

In order to edit individual parameters, click directly in the associated cell of the table. It is not necessary to select the line beforehand. After clicking, a control appears with which you can edit the parameters. The control's type depends on the particular parameter concerned. In the image above, you see a drop-down list for selecting the desired channel status (Active/Passive).

Editing multiple parameters

Name	Status	Connector
▼ Channel type: Analog inputs (Count=8)		
Channel_001	Active	[01] IN01
Channel_002	Active	[01] IN02
Channel_003	(Active)	[01] IN03

Channel status

Active

Active

Passive

Control for multiple selection
(example)

Select the desired row as explained in the previous chapter ([Selection](#)¹³³). Next, click on one of the rows where you wish to edit parameters. A control appears in which the corresponding parameter can be edited. If all of the parameter values are the same, the value is indicated; otherwise, an inequality - symbol (≠) is displayed:

Name	Status	Connector
▼ Channel type: Analog inputs (Count=8)		
Channel_001	Active	[01] IN01
Channel_002	Passive	[01] IN02
Channel_003	(Passive)	[01] IN03

Channel status

≠

Active

Passive

Different parameters in a multiple selection procedure
(example)

7.4.1.3 Configuring columns

By default, the various pages of **Setup** show a certain selection of columns (parameters). You can also configure the selection and order of the columns according to your own wishes (see "[Showing and moving columns](#)"¹⁴⁸). The column configuration is saved/opened with the view configured.



Note

Saving / restoring a configuration

In order to save a column layout, use the functions belonging to the View menu (see the manual imc STUDIO Monitor (general) > chapter "[Views](#)"⁸⁴).

7.4.1.4 Sorting and grouping



Note

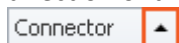
Saving the sorting and grouping arrangements

The **sorting/grouping arrangements** are **saved with the views**. The views are **saved along with the project**. This means that any views are available for all of the project's experiments.

- See: "[What is saved where?](#)"⁹¹
- See: "[Views](#)"⁸⁴

Sorting

Clicking the mouse over a column header causes the table to be sorted according to that column's content. The direction of the arrow at the right margin of the box indicates the sorting direction:



Reversing the direction: Each click of the mouse toggles the sorting direction to the respective opposite direction.

Undoing the sorting: Open the column header's context menu and select "*Clear Sorting*".

Sorting according to multiple columns: Activate the sorting according to an additional column by clicking the mouse while holding down the <SHIFT>-key.

Name ▲	Connector	Status ▲
▼ Channel type: Analog inputs (Count=8)		
Temp_1	[01] IN01	Active
Temp_3	[01] IN03	Active
Channel_004	[01] IN04	Passive
Channel_005	[01] IN05	Passive
Channel_006	[01] IN06	Passive
Channel_007	[01] IN07	Passive
Channel_008	[01] IN08	Passive
Temp_2	[01] IN02	Passive

The table is sorted according to the parameter "Status" and next by "Name".

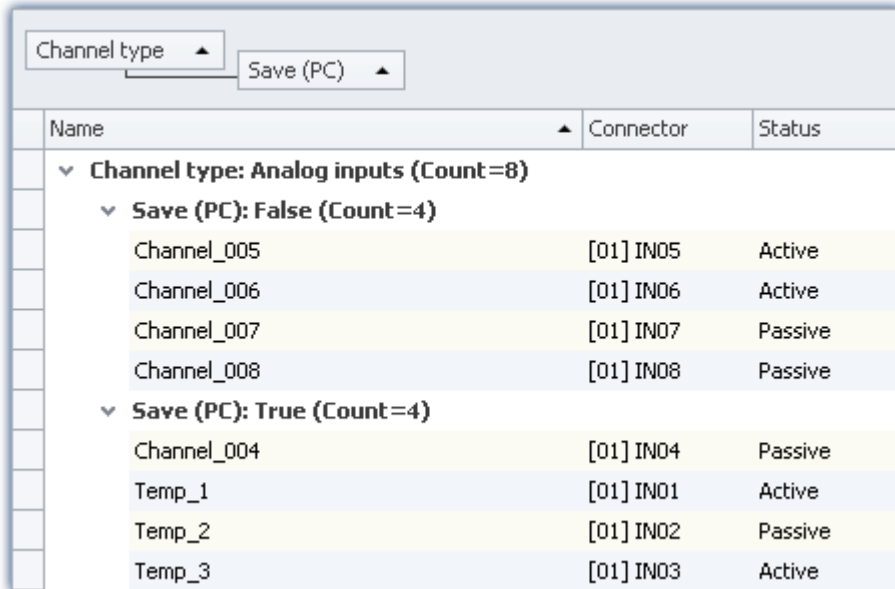
Active channels are thus always at the top.

All active and passive channels are sorted alphabetically among themselves.

Grouping

By default, most tables are grouped according to one of the parameters. For instance, the channel tables are grouped according to the parameter: "Channel type". You can, if desired, cancel the grouping or modify it.

To do this, open the context menu in the upper left corner of the table and select the item "Group By Box". The box is then displayed above the table. Here, you can add table columns using Drag&Drop.



The screenshot shows a table with a header row: Name, Connector, Status. Above the table, there are two dropdown menus: 'Channel type' and 'Save (PC)'. The table content is grouped as follows:

Name	Connector	Status
Channel type: Analog inputs (Count=8)		
Save (PC): False (Count=4)		
Channel_005	[01] IN05	Active
Channel_006	[01] IN06	Active
Channel_007	[01] IN07	Passive
Channel_008	[01] IN08	Passive
Save (PC): True (Count=4)		
Channel_004	[01] IN04	Passive
Temp_1	[01] IN01	Active
Temp_2	[01] IN02	Passive
Temp_3	[01] IN03	Active

Example of grouping according to the channel type and next by the saving status.

Disadvantage: Any columns according to which groups were formed can no longer be displayed in the table. If this is required however, then create an additional column of the type : "[Combined column](#)". This workaround lets you show the parameter after all. (Creating/editing of the columns is only possible with at minimum the [imc STUDIO PRO](#) product versions.)

7.4.1.5 Filtering columns



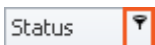
Note

Saving the filter settings

The **filter setting is saved with the views**. The views are **saved along with the project**. This means that any views are available for all of the project's experiments.

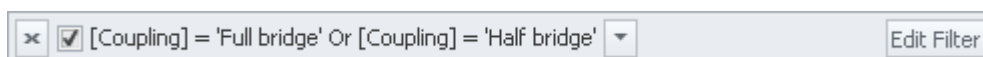
- See: "[What is saved where?](#)"
- See: "[Views](#)"

To select a filter you use the header of the column to which the filter is to be applied. When you move the mouse over the column header, the filter-symbol appears at right.



Clicking the mouse over the filter-symbol causes a selection menu to appear. Here you can select a value from among the current settings, which is to be the only one still displayed. Or, you can define your own filter ("Custom").

Once you have defined a filter, a filter bar appears on the bottom.



Filter-bar

Here, the filter currently set is shown. In this example, all columns which use either "Full bridge" or "Half bridge" coupling are used.

Removing filters: You can remove a filter by clicking on the "X", or by using the filter-symbol in the table header: select "All".

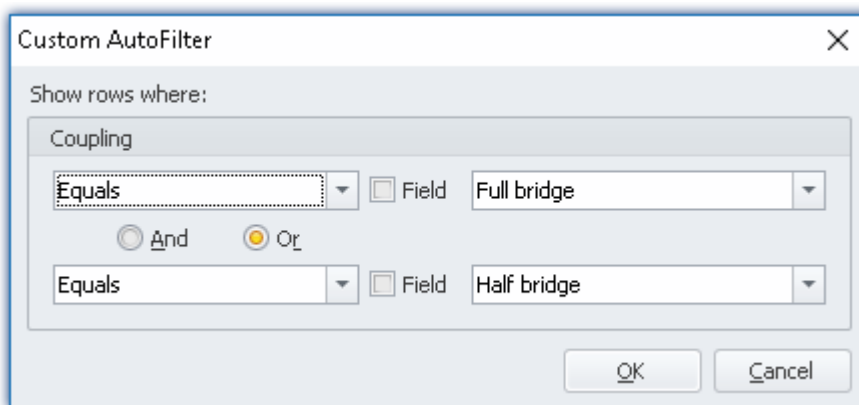
Activating/deactivating: Using the checkbox in the bar, you can activate/deactivate the filter without removing it.

Changing filters: Clicking the mouse over the Drop-Down symbol causes a list to appear, which offers the most recently used filters for selection. Here, you can select a different filter.

User-defined (custom) filters

If you can't find the appropriate value by means of the filter-symbol, or if you require different conditions, you can implement "Custom" entries.

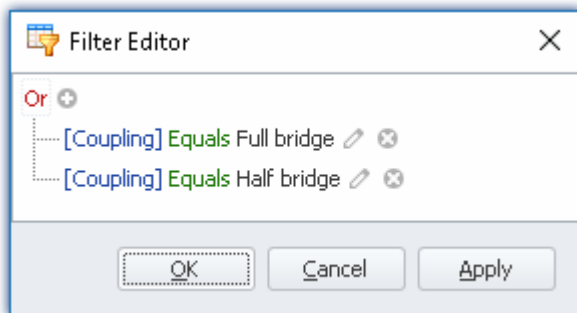
The following dialog appears. In this dialog, configure the filter as desired and then click on "OK". Note that the filter only works with exact values/texts.



Custom filter

Editing filters

In order to define complex filters, click in the Filter-bar on "Edit filter". The following dialog window then opens:



General Filter-dialog

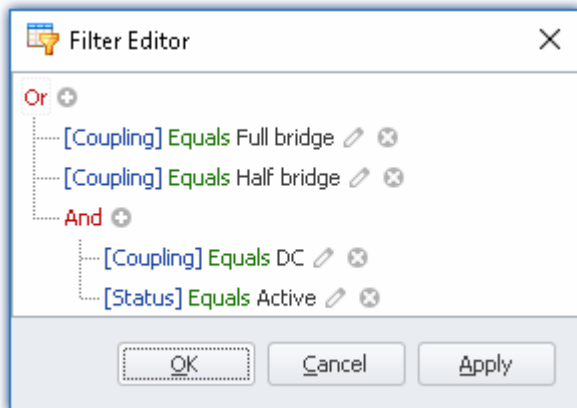
Using this dialog window, you are able to formulate complex combinations of filter conditions.

Changing the group's operator: When you click on the operator which is already set for the group, a list of operators then appears, from which you can select the appropriate one: "And", "Or", ...

Adding a condition: By clicking on , you can add any desired number of filter conditions to the operator group. In doing so, be sure to follow the pattern of placing the parameter (e.g. coupling) at the first position in the condition statement, then the condition itself in the second position (e.g. "equals" or "Does not equal", ...), and finally at the third position the "value".

Deleting: Clicking on  lets you delete filter conditions.

Adding a group with an additional operator: Click on the operator set for the group. In the list which then appears, select "Add group". The new group can have a different operator type and can in turn contain any kind of conditions/groups.



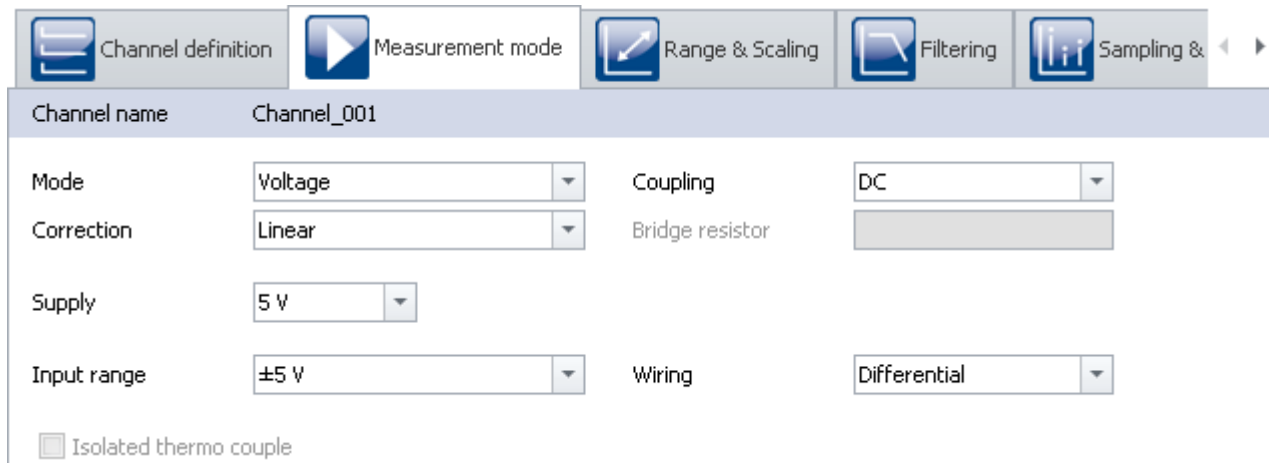
Filter setting with two groups

*Coupling = "Full bridge" OR coupling = "Half bridge" OR
(coupling = "DC" AND Status = "Active")*

Dependency on other parameters: Instead of entering a value according to which to filter, you can also select another parameter. This means you are able to add dependency relationships to other settings. To do this, click on the pencil-symbol or on the box-symbol next to the value to change.

7.4.2 Dialogs

In this section, the most important parameters of special dialogs are offered.



Example of "Measurement mode" dialog

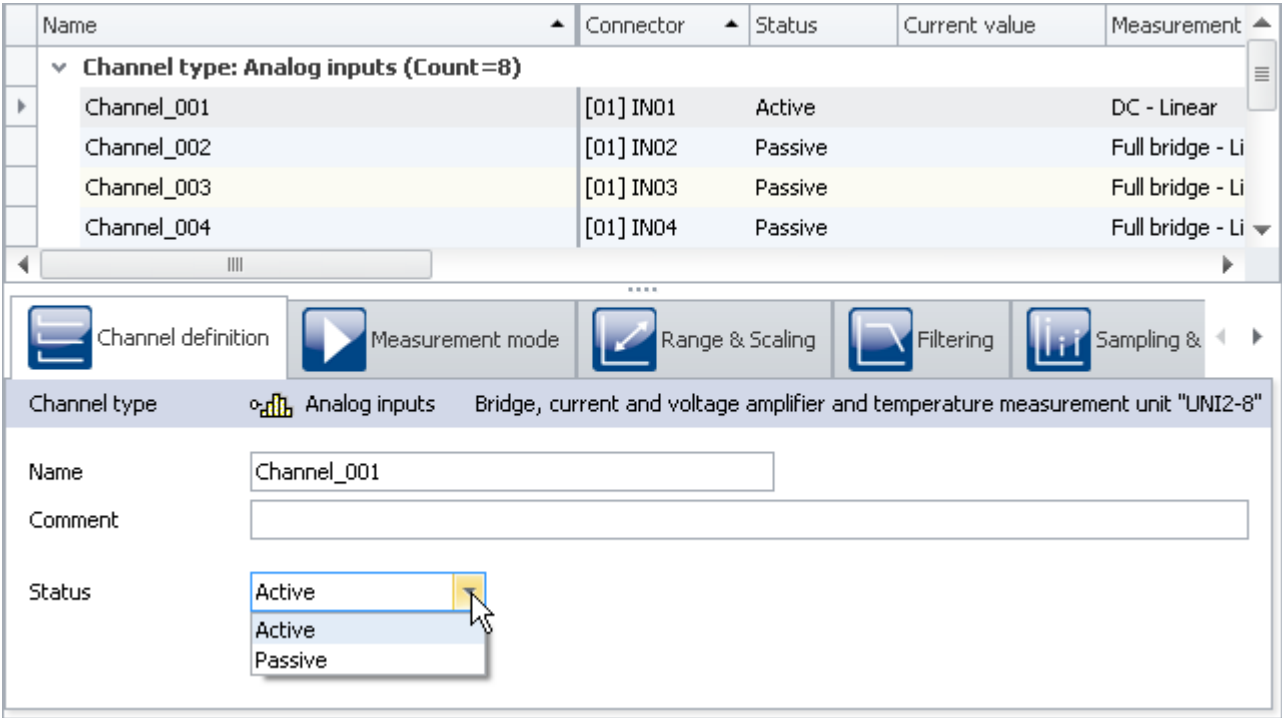
Selected parameters are presented here, distributed over multiple dialogs. The dialogs are designed in a clearly structured manner. As with the tabular display, you can also [directly edit](#) ¹⁴⁰ parameters in the dialogs.

Availability and selection of the dialogs and the settings depends on what page is open and on the selection in the table.

7.4.2.1 Direct editing

You can directly edit (change) all parameters displayed in the dialogs.

Editing individual parameters



Editing individual parameters (example)

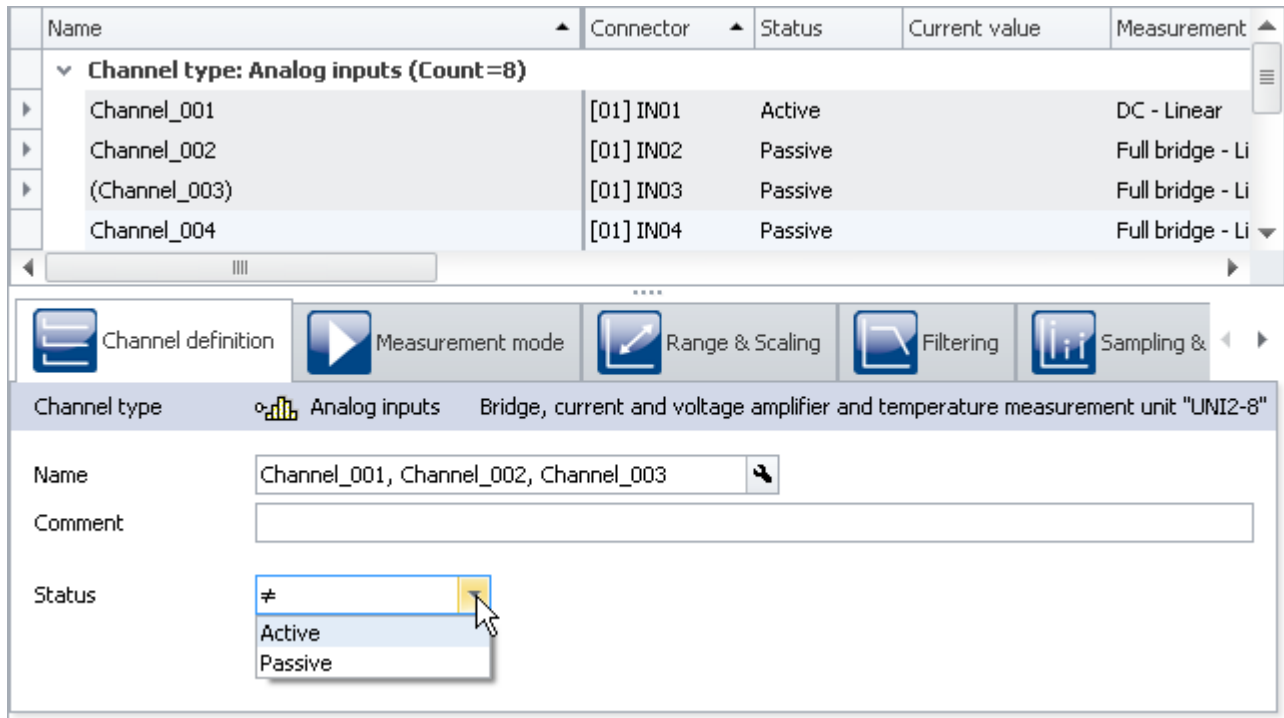
To edit a parameter, first select the corresponding table cell in the tabular display (see [Selection](#)¹³³⁾). The dialogs associated with the selection are indicated. Open the dialog in which the parameter to be changed is found.

Click on the control with which the parameter can be edited (changed). The control's type depends on the parameter. In the image above, you see a drop-down selection list for setting the channel status (active/passive).

Editing multiple parameters

When a parameter in a dialog is edited, then this change causes changes to all rows in the tabular display selected. If all parameters have the same value, this is displayed, otherwise an "unequal" symbol is shown ($\neq$).

In the image shown below as an example, you see three selected channels and the dialog *channel definition*. If you select the **status active** in the drop-down dialog, the status for all three channels is set to *active*. The results are shown immediately in the channel table.



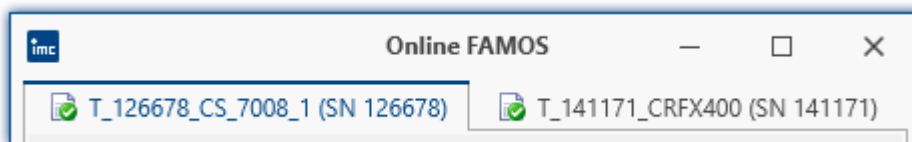
Non-matching parameters in a multiple selection
(example)

7.4.3 Assistants

The Assistants are an aid in configuring particular device components such as Fieldbusses or imc Online FAMOS. The methods of operating and configuring the respective Assistants are presented in the associated chapters.

All Assistants are shown in a dialog frame by means of which you can select the device affected. If only one device is involved in the experiment, there is nothing for you to do; that device is automatically selected. But if there are multiple devices in the list, then use the tabs in the upper region of the dialog to select the device you wish to configure.

For each device a separate tab appears.



The symbols appearing in front of the tabs provide an indication of any name conflicts (red X-symbol). A green check-mark indicates that there is no name conflict.

7.5 Performing a measurement: Procedure

This chapter describes how the measurement procedure (the experiment) is controlled. Many functions can be accessed via the Ribbon **Home** and all via the Ribbon **Setup** .

Action	Description
Starting measurement	Starts measurement for one device via the device software imc STUDIO or imc WAVE.
Starting imc STUDIO Monitor	
Searching for and selecting devices	Before you can create an experiment, you must select one or more devices in the device table. All available devices are listed.
Firmware Update	The software only works with devices having the right firmware. Upon the connection/preparation, the versions are compared. If they don't match, then appropriate firmware must be installed. The device software imc STUDIO or imc WAVE determines the firmware necessary.
Configure devices 	Each measurement setup can require a variety measurement device settings. Configure your device and each measurement channel according to their respective requirements. For this purpose, various software Assistants (wizards) are provided.
 Process configuration 	Current configuration is inspected for errors and made available to all the plug-ins. No connection with the device is required for this.
Connect	The imc STUDIO Monitor itself with all measurement devices selected (generally via LAN).
 Start measurement  (start monitoring)	Starts monitoring for the measurement devices selected.
 Stop measurement  (stop monitoring)	Stops monitoring for the measurement devices selected.

7.5.1 Device Search - known and selected

Term	Description
Search for devices	In a device search, the system searches the network for all suitable devices. The " device table "  lists all devices found.
Device is known	" <i>Device is known</i> " means that the device is already available for selection as soon as the software starts. The device's setup is familiar to the software and an experiment can created without needing to connect with the device.
Selected	" <i>Selected</i> " means that a device is to be used for the current experiment.

Device selection

To be selected for an experiment, a device must be "selected".

	Selected	Device name	SN	Device specification
▶	☒	T_124835_C1_1_LEMO_ET	124835	imc C1-1 LEMO
	☐	T_130039_busDAQ_X	130039	busDAQ-X
	☐	T_130311_SPARTAN_U32_CAN	130311	imc SPARTAN

By clicking on the desired devices "Selected" box, the device becomes ready for the experiment. You can also select multiple devices for your experiment.

You are not yet connected with the device!

If your device was not previously known, it is made *known* the first time the it is selected.

Status display

After selecting the device, the **status** of the "**Connection**" and of the "**Measurement**" is indicated in corresponding columns:

Devices		Analog channels		Digital channels		Variables		Triggers	
Selected	Device name	SN	Device specification	Connection status	Measurement status	Device control			
▶ ☒	T_126678_CS...	126678	imc C Series	disconnected	stopped	Connect	▼		

Status indication in the columns
"Connection status" and "Measurement status"

The same information is also available in the "Details" toolbox ("Measurement status", "Connection status"):

Details	
Connection status	disconnected
Device control	Connect ▼
Device is selected for me...	☒
Device name	T_126678_CS_7008_1
Device SN	126678
Device specification	imc C Series
Measurement status	stopped

Status display in the "Details" window

7.6 Metadata and Comments on the measurement

7.6.1 Saving metadata with the channel - Traceability of data

Metadata can be saved directly in the channel file. No separate file is created. This means the relevant information is saved **in all channels** and thus **in all channel files** on the PC hard drive. This applies to saving both **on the PC** and **in the device** (for devices of the [firmware group A](#)^[116]). With Diskstart, the meta information which was available at the time when the Diskstart was created is saved with the channel.

Column selection

In the Options "Setup" > "[Traceability of data](#)"^[66], you have the choice of which groups of parameters to save.

Ribbon	View
Extra > Options ()	all

Option	Description
Traceability of data ^[66]	Groups of Setup-parameters which are saved as metadata in the Channel Properties. They can be read out again at a later point in time. The channel properties make reconstruction of the measurement's configuration possible.

Various groups of Setup-parameters are available for selection, which can also be saved. The tooltip for each of these selections contains an exact list of all parameters. Here is an overview:

Groups	Description
Basic information	Device serial number, channel number, module serial number
Hardware information	Among others connector, channel type, module number, module type, amplifier calibration date
Channel settings	Among others measurement mode, coupling, offset, factor, bridge parameters, inc.-parameters, filter parameters
Balance information	Balance date, balance compensation 1/2, balance time
Metadata	All user-defined metadata columns ^[160] , which had been created for the channel  Channels can record any data which can be interpreted as either text or a number. Something which cannot be saved with the channels is embedded files (images/PDFs). For instance, when a file is saves as "Link" or as "Link + in experiment"^[161], the path to the file is saved as metadata in the channel. If this is not the case, the parameter remains empty in the channel.

 Notes**Saved parameters - adapted to the channel type**

Only parameters relevant to the channel type are saved. Analog channels contain different parameters than incremental counter channels.

Changes during measurement

During the **start of the measurement**, the metadata for the channel are specified. For this reason, any changes made to the metadata **during the measurement are not saved to the file**.

This also pertains to "Interval saving", for instance, and in the case of "*Suspend/resume data storage*".

No storage of port metadata

Port metadata is not saved in a measurement file. Reason: The port is not saved as a separate file, only the bits.

Reading out metadata

In order to read out the metadata saved with the channel, proceed as follows:

- **Widgets:** The placeholder `<VARS [ "myChannel" ] . PROPS [ "myMetadata" ] >` enables querying of the metadata saved with the channel.
- **imc FAMOS:** Use the command `UserPropText?` (for texts) or `UserPropValue?` (for numbers).
- **Data Browser:** In the Data Browser, the saved metadata can be added as a column. To do this, open the context menu in the Data Browser and select "*Add metadata columns*". In the list which then appears, select the "*column ID*" of the desired metadata item. The column appears and the metadata are displayed in the corresponding column of the channel. The columns of the Data Browser are part of the view. Save the [view](#) ⁸⁵ before closing so that the settings remain in force upon next starting imc STUDIO Monitor.

For the purpose of parameter selection, the the metadata's language-independent column ID is required.

 FAQ**Question: How do I find out the metadata's column ID?**

Answer: Open a saved channel containing metadata in the free-floating curve window (double-click on the channel in the Data Browser).

Next, select the menu item "*Configuration*" > "*More Channels*". Jump to the tab "*Channel info*".

Here you will find the available saved metadata with their respective column IDs under the heading "*User-defined properties*".



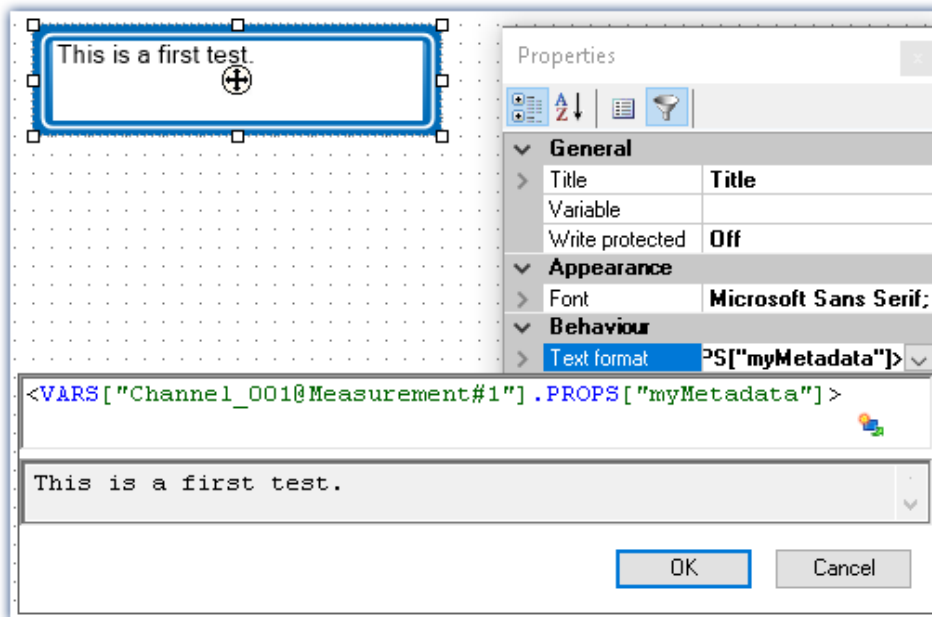
Example

Getting a channel's "myMetadata"

Suppose "myMetadata" from the group "User-defined Metadata-columns" is saved with the channel. Additionally, at least one measurement has been performed under this configuration. Now you can get the saved metadata item as follows:

- In the **Widget**: create a textbox, e.g. one from the group "Automotive". Open the properties for the text box. For the property "Text format", enter the following text (if it is not possible to create the box from scratch using the input support, then amend the text):

```
<VARS ["Channel_001@Measurement#1"] . PROPS ["myMetadata"] >
```



Querying the parameter "myMetadata" of the measurement having the symbolic measurement number#1

- Via **imc FAMOS**: Transfer the channel using the command "[Run imc FAMOS sequence](#)". Enter as the imc FAMOS command `commentStart = UserPropText?(Channel_001, "myMetadata")`. In order to always use the channel belonging to the measurement #1, in the transfer dialog make the setting "To imc FAMOS Source=Measurement#1".
- In the **Data Browser**: Open the context menu in the Data Browser and select "Add metadata columns". Select "myMetadata" from the list. The column now appears in the Data Browser. The contents appear in the associated channel column.

7.7 Information and tips

7.7.1 Enumeration classes

In an Enumeration class, the associations to "Object-groups" (e.g. analog channels, Display variables, devices, ...) are defined.

It is important to make an appropriate selection for the [Default Values](#) ¹²¹ for example, in order to have the correct targets for the defaults.

Enumerator class	Internal designation	Description
Experiment	eExperiment	Experiment metadata
Device	eDevice	Device settings
Module	eModule	Amplifier parameters
Bit	eSubChannel	DI-/DO-bits (incl. Mon-DI-Bits), virtual bits, Ethernet bits, Field-bus bits (MFBDIO), pv-variables (of DI-ports, analog-, incremental counter- or Fieldbus channels)
Channel	eChannel	Analog channels, DIO-ports, other pv-variables, virtual channels, Display variables, ...
Trigger	eTrigger	Trigger settings (trigger name, trigger linkage, event number, trigger, trigger count)
Parameter	eParameter	AppMod block parameter (only the parameter value can be set)

7.7.2 Setup-layout

This section describes the Setup-interface's designing options.

Chapter overview

Synopsis	Section
Showing and arranging existing columns	Showing and moving columns ¹⁴⁸
Personally defined columns	Creating and configuring additional columns ¹⁵¹
Preventing dialogs from closing until after an entry has been made	Using Mandatory Boxes ¹⁶⁹
Transferring columns to other PCs or to other projects	Export/import of column/table descriptions ¹⁷³

7.7.2.1 Showing and moving columns

By default, the various Setup pages show a certain selection of columns (parameters). You can also **configure the selection and order** of the columns according to your own wishes. The column configuration is saved/opened with the view configured.



Note

Saving the column selection

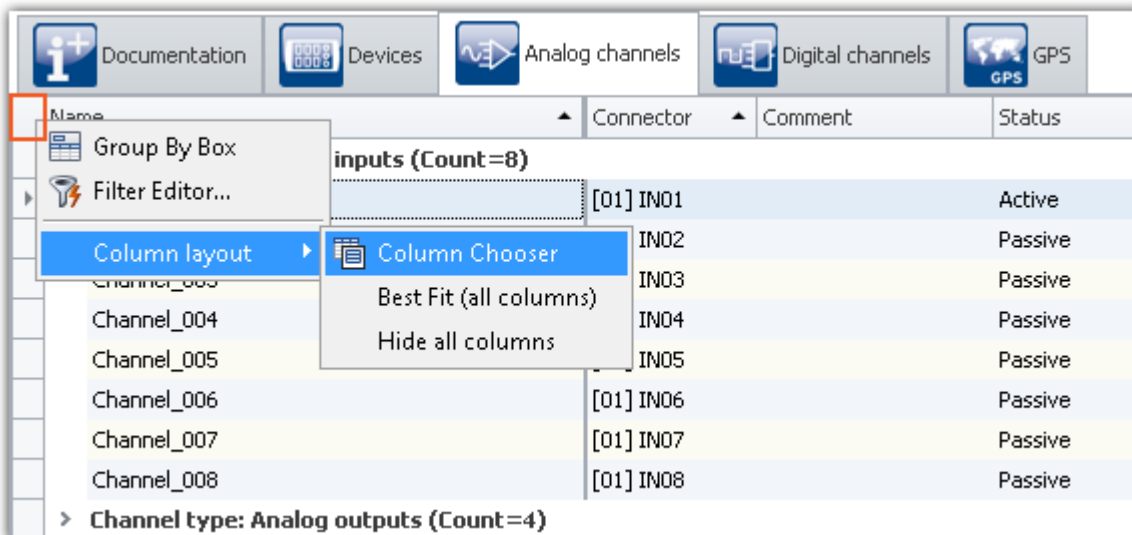
The **position and size of the column** is **saved with the views**. The views are **saved along with the project**. This means that any views are available for all of the project's experiments.

- See: "[What is saved where?](#)"⁹¹
- See: "[Views](#)"⁸⁴

The pages are constructed in a variety of styles. The various design elements have different methods for configuration.

Display in a "Table": e.g. the pages "Devices", "Analog channels", ...

Opening "Column Chooser"

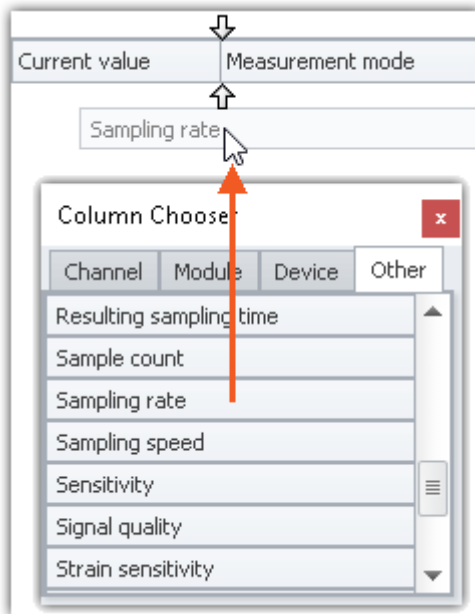


Opening the table's context menu

In order to insert columns, open the "Column Chooser" dialog. To do this, open the context menu in the upper left corner of the table. Select "Column Chooser".

Inserting columns

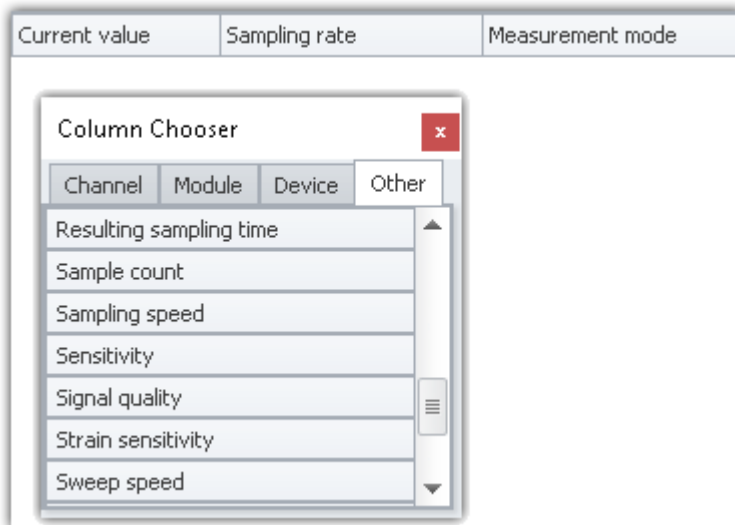
The dialog "Column Chooser" contains a list of all columns which can be inserted. In order to insert a column, **drag it from the list** to the column's header, as shown in the image below.



The position of the arrows (before the column has been placed) **indicates** whether the column is to be inserted. In this example, the column "Sampling rate" is inserted to the right next to "Current value".

Dragging a column

Once you have placed the column, it is inserted at the position chosen.



The column positioned

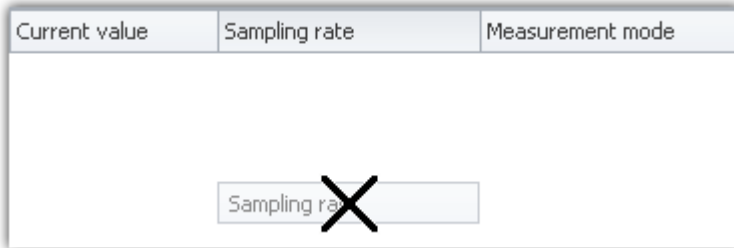
Moving columns (changing position)

To change a column's position, **use Drag&Drop to move it** to the title bar at the desired position.

Deleting columns

In order to delete a column, use one of the two following possibilities:

- Open the "Column Chooser" dialog and drag the column to the dialog window
- Drag the column to an empty area next to the table as shown below:



*Deleting a column by
Drag&Drop*

The "Column Chooser" dialog does not need to be open.

Display in a "Detail-Table": e.g. the pages "Documentation", "Experiment", ...

Opening "Column Chooser"

In order to insert columns, open the dialog "Column Chooser". To do this, open the **context menu of an existing column title** or of any empty area. Select the item "Column Chooser".

Inserting a column

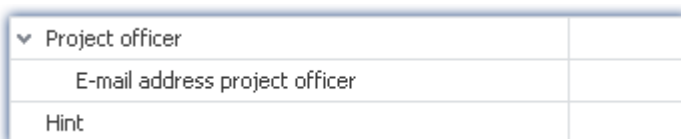
The dialog "Column Chooser" contains a list of all columns which can be inserted. In order to insert a column, **drag it from the list** into the existing table.

The **mouse cursor** indicates where the column will be inserted.

As a branch of an existing group

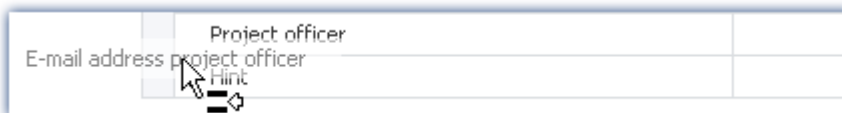


Insertion as a branch in an existing column

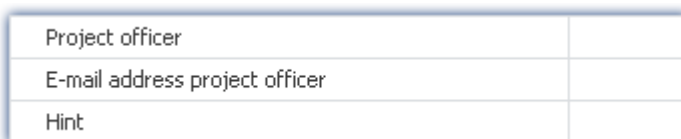


The next higher level column becomes a grouping element

Between existing columns

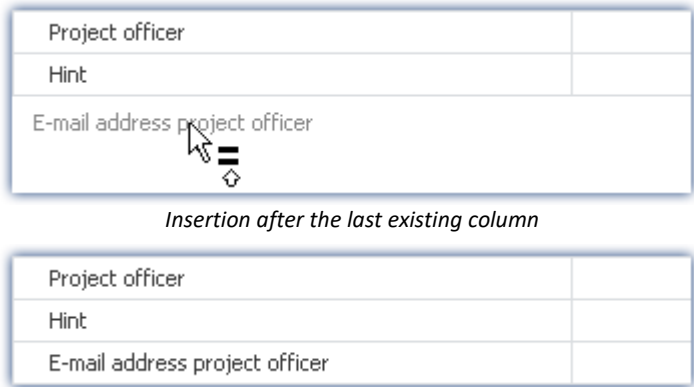


*Insertion between two existing columns
Be aware that the cursor needs to be rather far to the left*



The column is inserted in between

After the last column



Insertion after the last existing column

The column is inserted at the end

Moving columns (changing position)

First open the dialog "Column Chooser". In order to change the position of a column, **move the title** by means of Drag&Drop to the **desired position** (in the same manner as inserting the column).

Deleting columns

First open the dialog "Column Chooser". To delete a column, **move the title** by means of Drag&Drop to the **dialog**.

 **Note**

Saving / restoring a configuration

In order to save a column layout, use the functions belonging to the View menu (see the manual imc STUDIO Monitor (general) > chapter "[Views](#)"⁸⁴).

7.7.2.2 Creating and configuring additional columns

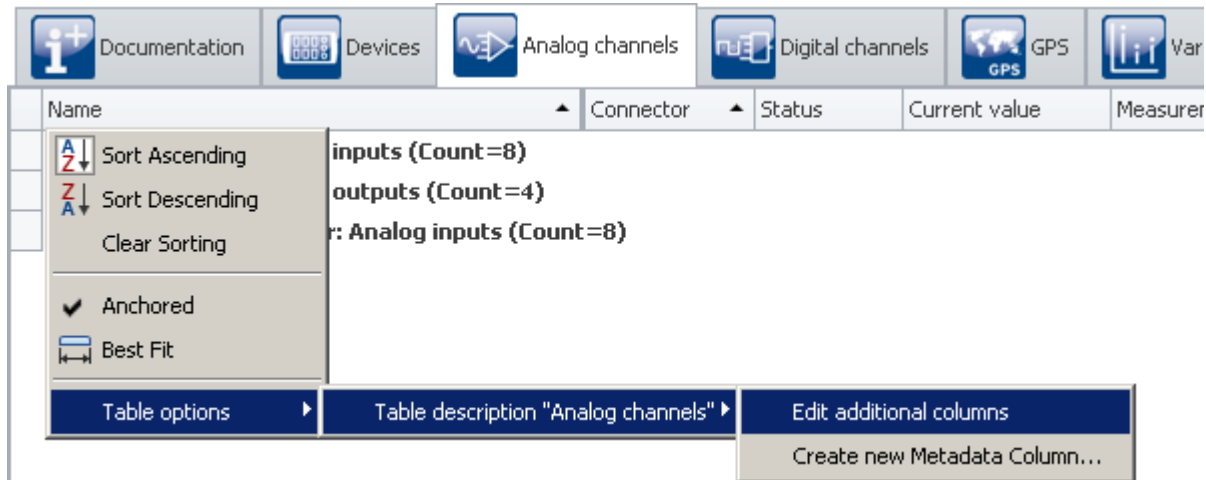
"Additional columns" contains controls for **improving and adapting** the **operation and documentation** of your measurement results in a wide variety of ways.

Type	Description
Combined column ¹⁵⁴	Allows combination, operation and display of multiple parameters in a single column.
Metadata column ¹⁶⁰	Allows display of any desired information and documents (also from third-party) applications.
Parameter set column ¹⁶²	Fixed parameter sets (combinations of settings) can be saved in a column and set with a single click of a button.
Parameter set import column ¹⁶⁷	Pre-defined, comprehensive parameter set files (*.csv) can be loaded by means of the drop-down list.

Additional columns are defined in two steps. **First select the type** of extra column and set certain basic properties such as the title and designation. Then for the "Combined column" and the "Parameter set column", select **the corresponding columns** and for the "Metadata column" select **the data type** or data source (file).

Creating the columns with the Collection editor

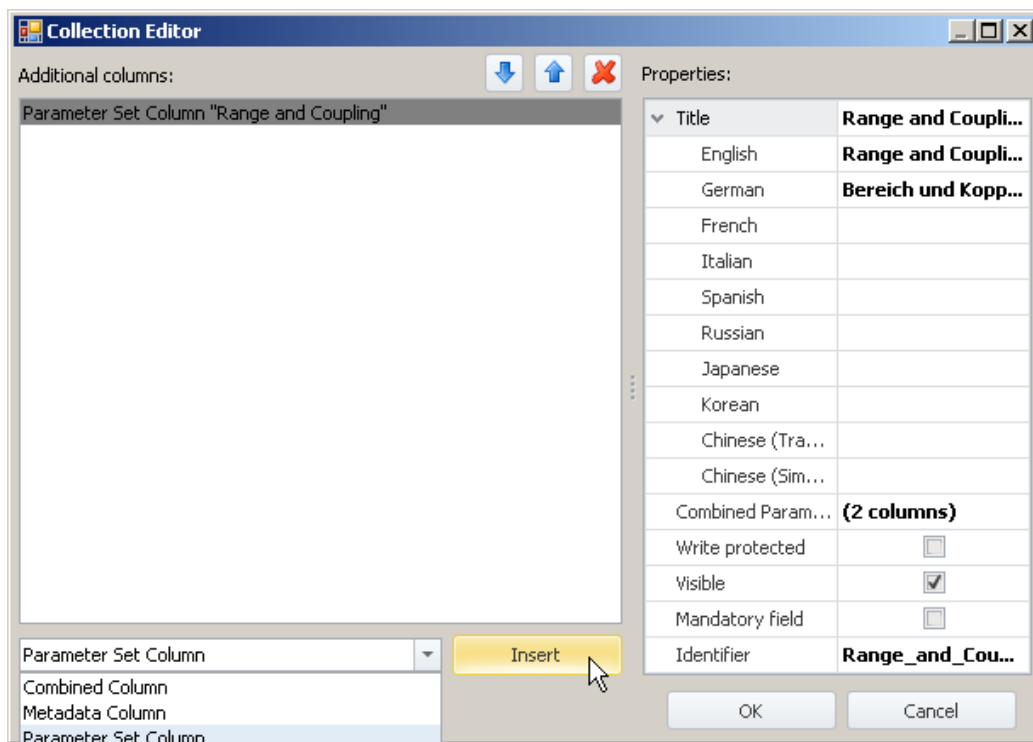
To create additional columns, open the **context menu of the column headers** and select "Table options" > "Table description "<Name>" " > "Edit additional columns".



*"Edit additional columns" in the context menu of the column headers
Here also: Context menu of the table header*

This function is available on every Setup page (Documentation (here without table), Devices, Channels, ...).

The **Collection Editor** opens:



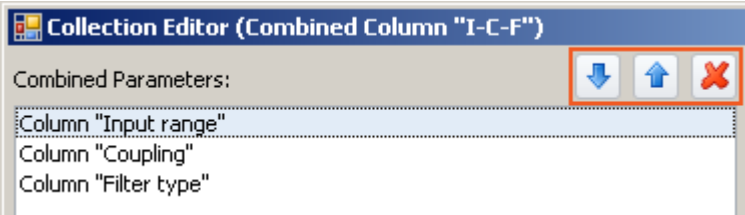
Collection Editor (example)

- In the drop-down list (at bottom left), **select the type of the additional column**.
- Click on "Insert" and **the column is adopted into the list**.
- In the "Properties" window, set the column's **appearance / functionality**.

For a detailed example, see the chapter "[Combined column](#)"¹⁵⁴ or "[Parameter set column](#)"¹⁶².

Sorting/deleting columns and parameters in the Collection Editor

To **sort entries**, use the "Up/Down"-arrow icons ( / ). To delete entries, use the "X" icon ().



Collection Editor (example shows sorting)
List of parameters of a "combined column"

For a detailed example, see the chapter "[Combined column](#)"¹⁵⁴."

Common properties

Some properties are shared by all Additional columns:

Properties	Description
Title	The column title as it appears in the table. The title can be set separately for selected languages. Depending on the language setting, the corresponding title is displayed. If the title is empty, the English version is used.
	Special characters and spaces may be used. A comma in a column name should be avoided.
Write-protected	The column content cannot be edited in the column.
Visible	If you select this option, the column is defined but not visible. This makes sense for instance when you pre-define different columns and later display / select them individually.
Identifier	This identifier is used within the program to uniquely distinguish the column. By default, the title is used for it. If this identifier is already assigned, a distinguishing suffix is automatically appended ("_1", "_2", ...).
	Special characters (except "_") and spaces may not be used.

 **Note**

Saving the columns

The **Additional column's configuration is saved** in the respective **table description**. The table descriptions are **saved along with the project**. This means that any additional columns defined are available for all of the project's experiments.

- See: "[What is saved where?](#)"⁹¹"
- See: "[Views](#)"⁸⁴"

7.7.2.2.1 Combined column

The **Combined column** joins multiple parameters in one column.

This can make sense, for instance when a measurement is often performed and a certain set of parameters must be set before every measurement. You can then set the parameters in this column and no longer need to open a series of various columns/index cards.

Name	Status	I-C-F	Connector
Channel type: Analog inputs (Count=8)			
Channel_001	Active	±5 V; DC; AAF	[01] IN01
Channel_002	Passive		
Channel_003	Passive		
Channel_004	Passive		
Channel_005	Passive		
Channel_006	Passive		
Channel_007	Passive		
Channel_008	Passive	±1000 "mV/V";	
Channel type: Analog outputs (Count=4)			

Combined column
Example of column: I-C-F

Configuring

In order to configure the parameters, click on them in the table cell (it is also possible to select multiple cells at once; see the chapter *Tabular Display* > [Selection](#) ¹³³). The controls for the parameters are displayed and at the edge you see the wrench icon ():

Name	Status	I-C-F	Connector
Channel type: Analog inputs (Count=8)			
Channel_001	Active	±5 V; DC; AAF	[01] IN01
Channel_002	Active	±5 V; DC; AAF	[01] IN02
Channel_003	Active	±5 V; DC; AAF	[01] IN03
Channel_004	Active	(±5 V; DC; AAF)	[01] IN04
Channel_005	Passive		
Channel_006	Passive		
Channel_007	Passive		
Channel_008	Passive		
Channel type: Analog outputs (Count=4)			

Combined column with multiple selected cells
Example of column: I-C-F

If you change the coupling in the image above from "DC" to "Half bridge", for instance, the change applies to all selected cells. If the cells all had different couplings before opening the combined column, then the inequality symbol () is shown in the box for the coupling.

To close the combined column, click again on the wrench icon ()

Properties

For shared properties, see: [Additional Columns](#) ¹⁵³.

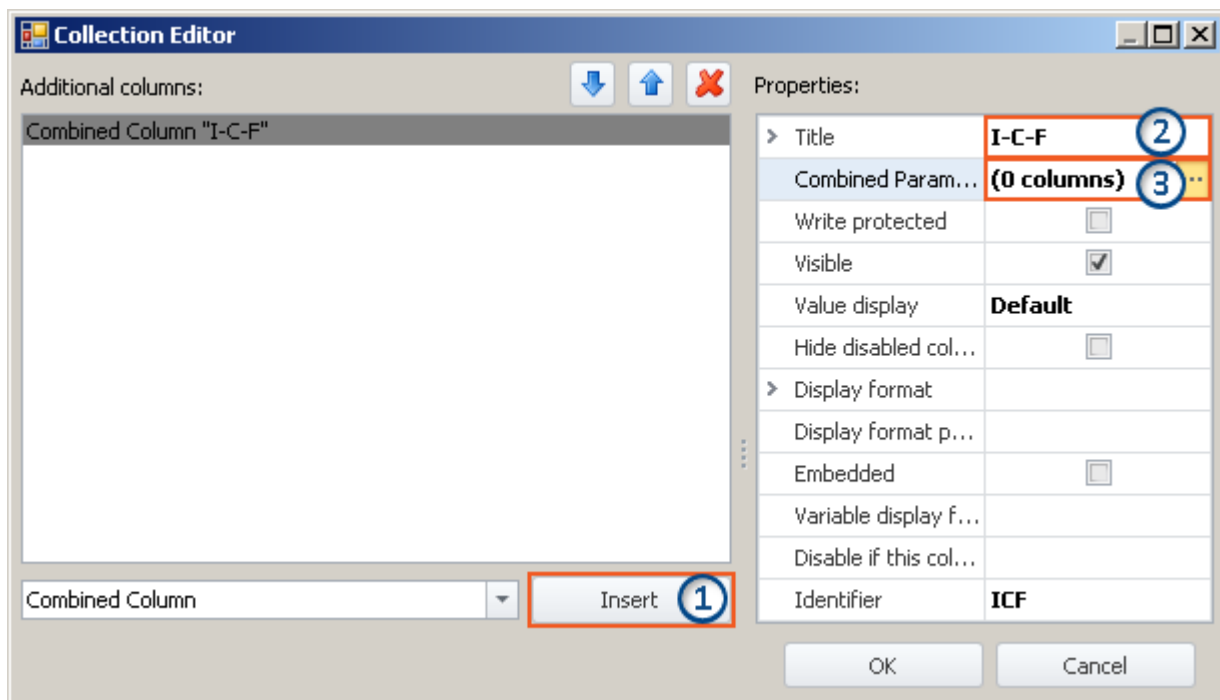
Type	Description
Combined Parameters	List of the parameters which are to be set by means of the column.

Type	Description
Value display	Display format in the cell. The values belonging to the combined columns are always listed in sequence in a row. By means of the following options, the format can be modified. • Default: Displays only the value of the combined columns e.g. Channel_001; Active • SimpleFormat: Selected parameters can be shown/hidden. Separators or additional texts can be freely defined. • DependentFormat: Selected parameters can be shown/hidden according to the status of one parameter. Separators or additional texts can be freely defined. • ShortCaption: Displays the short form of the parameter name before each value e.g. Name: Channel_001; Status: Active • LongCaption: Displays the long form of the parameter name before each value e.g. Channel name: Channel_001; Channel status: Active
Hide disabled columns from caption	In the cell, the space for deactivated columns is omitted. For instance, in a column with the parameters: Filter type + Filter cut-off frequency 1 + Filter cut-off frequency 2 and parameter values: Low pass, 20 kHz with Hide <i>on</i>: Low pass; 20 kHz with Hide <i>off</i>: Low pass; 20 kHz; or for the parameter value: AAF with Hide <i>on</i>: AAF with Hide <i>off</i>: AAF; ;
Display format	By means of the display format, you can set which parameters are visible in the cell. Example with four parameters: 0: Filter type 1: Filter characteristic 2: Filter cut-off frequency 1 3: Filter cut-off frequency 2 Display format: {0} - {2} - {3} Result: e.g. Band pass - 10 Hz - 1 kHz You can also use other texts and separators: Display format: Filter: {0} > {2} > {3} Result: e.g. Filter: Band pass > 10 Hz > 1 kHz Only usable with the value display type: SimpleFormat.

Type	Description						
Display format parameter and Variable display format	A variable display format is defined in accordance with a parameter's value. Example: Column: Sampling & Filtering. Here, the measurement duration is only displayed if it is not set to <i>Undefined</i>. The column has seven parameters. 0: Sampling rate 1: Duration 2: Filter characteristic 3: Filter type 4: ... Display format parameter: Duration <table><tr><th>Variable display format</th><th>Properties</th></tr><tr><td>0: "{0} - {3}"</td><td>ParameterValue: 0 PrintoutFormat: {0} - {3}</td></tr><tr><td>: "{0} - {3} - {1}"</td><td>ParameterValue: PrintoutFormat: {0} - {3} - {1}</td></tr></table> Parameter = 0 (Duration = <i>Undefined</i>) Show the 0th and 3rd value with separator "-": e.g. 100 Hz - Low pass Parameter = 10 s (Duration = <i>10 s</i>) Show the 0th and 3rd value with separator "-": e.g. 100 Hz - Low pass - 10 s In this way, other display formats can also be defined. Only usable with the value display: DependentFormat.	Variable display format	Properties	0: "{0} - {3}"	ParameterValue: 0 PrintoutFormat: {0} - {3}	: "{0} - {3} - {1}"	ParameterValue: PrintoutFormat: {0} - {3} - {1}
Variable display format	Properties						
0: "{0} - {3}"	ParameterValue: 0 PrintoutFormat: {0} - {3}						
: "{0} - {3} - {1}"	ParameterValue: PrintoutFormat: {0} - {3} - {1}						
Embedded	The parameters can be edited directly in the cell, instead of in the separate window. An example is presented on the page: Triggers, the combined column: <i>Properties</i>. For this function, it is necessary to alter the row height in order to make all parameters visible (only available as of imc STUDIO Developer).						
Disable if this column is empty	If the selected column is empty (e.g. an empty metadata-column), it is not possible to view or edit the parameters belonging to the combined column. In order to delete the view, click on the <Backspace> key.						

Example of a Combined column

Start the Collection editor as explained in the previous chapter [Additional columns](#) ¹⁵¹.



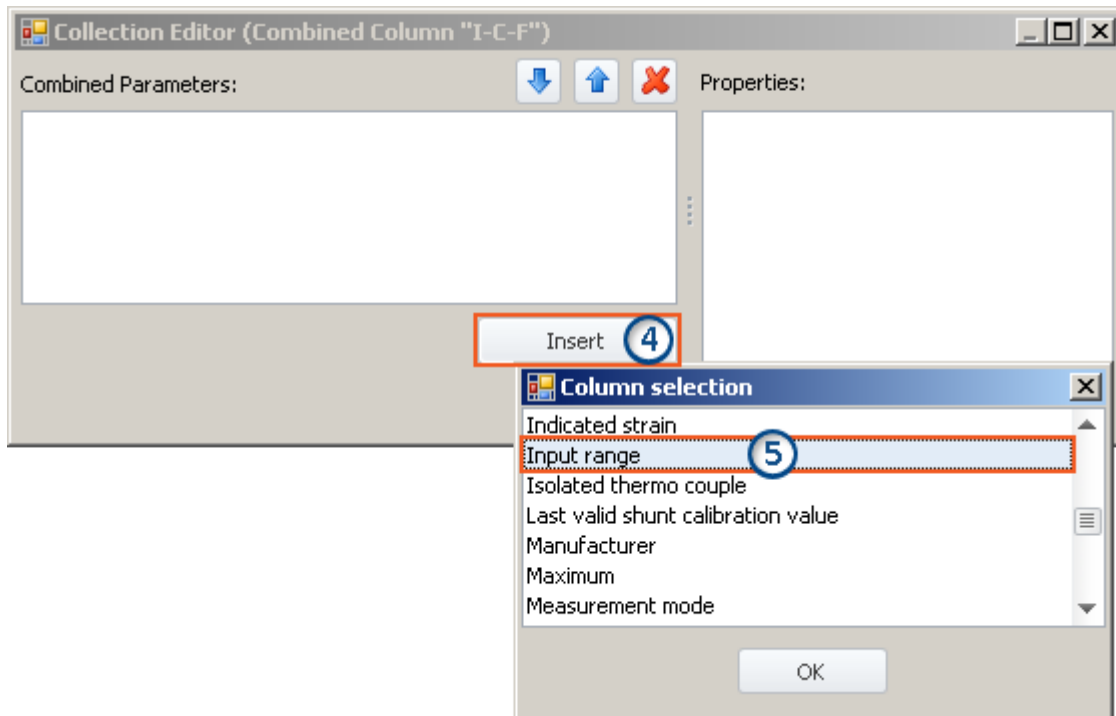
Creating a combined column
(example)

- 1) In the drop-down list, select the entry **Combined Column** and click on **Insert**. This creates a nameless (title) combined column.
- 2) Enter the desired title in the input box **Title**, e.g. "I-C-F" and hit the <Enter>-key (see image above). This entry is automatically used as a unique **Identifier**. Note that special characters (except "_") and spaces can not be used as identifiers. Remove all "-" in this example.

Adding parameters (columns)

- 3) Click in the row **Combined Parameters** in the Properties window and click on the button with the three dots ().

This opens an **additional window belonging to the Collection editor**. Here you set which columns/parameters are to be combined.



Collection editor, second window, adding column "Input range"

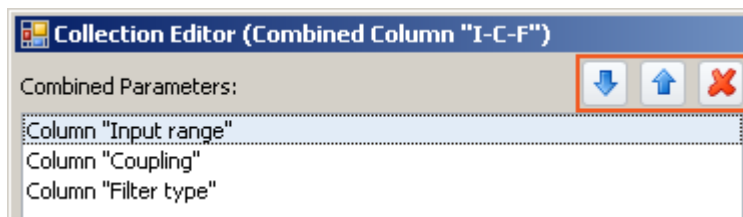
4) Click on **Insert**: An additional window with a list of all available parameters opens.

5) Select from the list the desired parameter(s) (for this example: **Input range**, **Coupling** and **Filter type**).

Sort

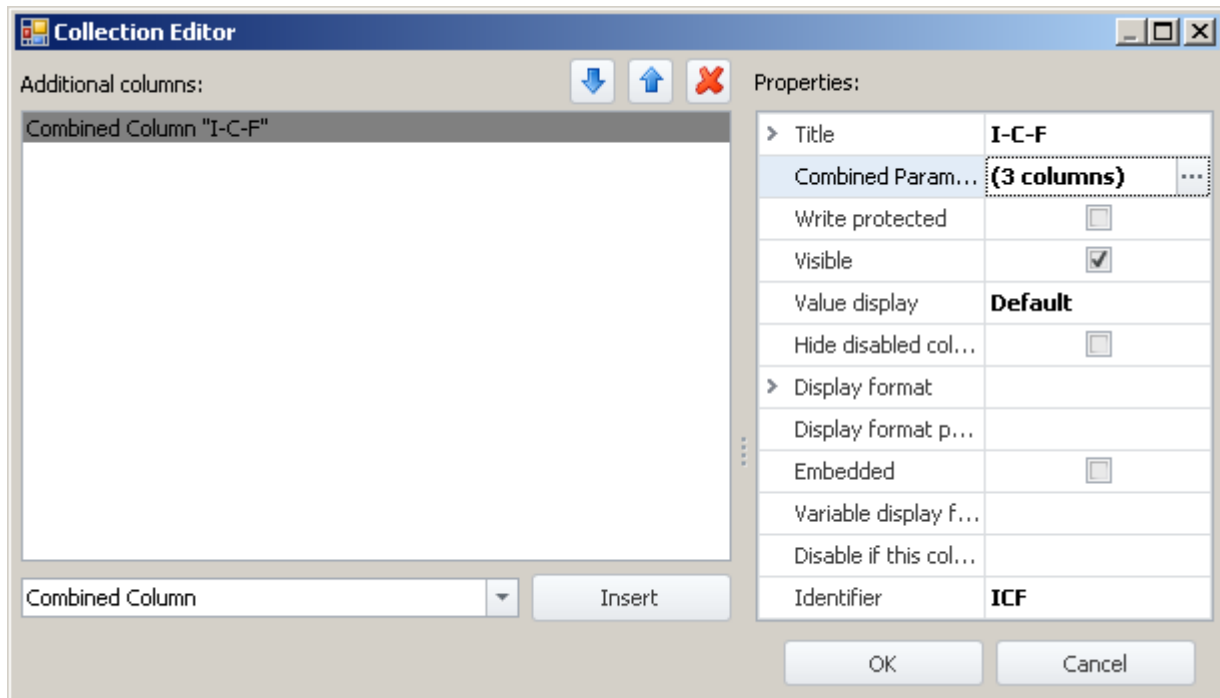
To sort entries, use the "Up/Down"-arrow icons ( / ).

Sort the order of "Input range", "Coupling", "Filter type", as shown below:



*List of the combined parameters
(after sorting)*

Close the second window by clicking **OK**. You then see the first window with the configured column "I-C-F" and the note indicating that "(3 columns)" have been combined:



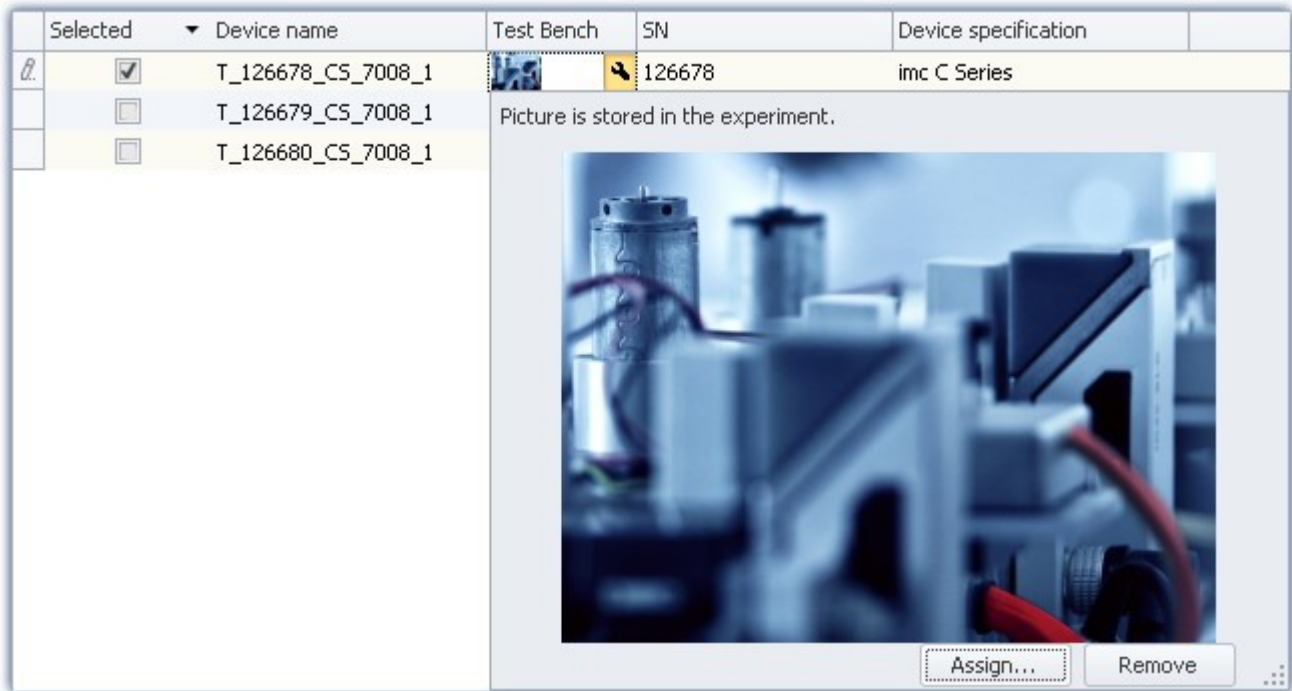
Collection editor, first window, combined column "I-C-F" (3 columns)

Close the window by clicking **OK**. You now have a new column in the channel list, which shows the three parameters together and with the help of which you can configure them together.

7.7.2.2 Metadata column

In a **Metadata column**, you can display any desired information or documents. They can be pictures (e.g. a picture of how to place the test object into the test rig), or a display of the date and time, or descriptive texts, or anything similar.

The metadata can be entered/edited at any time, for instance, even after a measurement. The natural use of metadata is for test/inspection documentation. The metadata content is saved along with the experiment, except for the types "Picture" and "Document". For these types, you can choose whether the file contents are saved with the experiment or only as a link.



Example of a metadata column

Metadata types

The following types are available:

Type	Description
Single-line text	Shows a single-line text in the column. To enter the text, click on the column.
Multiline text	Shows a multi-line text in the column. To read or edit the complete text, click on the cell.
Picture	Shows a picture (graphic) from a file. In the cell, the picture is minimized as a <i>Preview</i> or the associated icon is displayed as a <i>Placeholder</i> . In order to see the picture in full scale, click on the cell (see picture above)
Document	Shows a document of any desired type. In the cell, the associated icon is displayed as a <i>Placeholder</i> . Some document types, such as PDF and some graphics, can be viewed directly when you click in the cell. In order to open the document with the associated default program, click on the button: <i>Open</i> .
Text from list	Shows a selected text from a list of texts. To select a different text from the list, click on the cell. You create the list when making the column definitions (Property: "Text list").
Date	Shows a control having Date format. To enter the date, click on the cell. Click on the wrench icon () to select a date.

Type	Description
Time	Shows a control having Time format (Hour:Minute:Second). To enter the time, click on the cell.
Logical value	Shows a logical value in the form of a checkbox (☒ / ☐). This lets you symbolize any binary state "true/false" ("on/off", "Error/OK"). To change the state, click on the checkbox.
Directory	Shows a link to a directory. Click on the wrench icon () in order to start the folder selection dialog.

Properties

For shared properties, see: [Additional Columns](#)  153.

Type	Description
Text list	List of the texts available for selection. To edit the list, click on the button with the three dots (). Only usable for the type: Text from list.
Show placeholder	If <i>on</i>: instead of the preview, the icon corresponding to the file type is displayed. If <i>off</i>: a preview of the content is displayed • Type Picture: minimizes the picture • Type Multiline text: the first row Only usable for the types: Picture, Multiline text and Document (here, always <i>on</i>).
Storage type	Determines how the external file is integrated/linked. • In the experiment: The file is incorporated into the experiment. The column thus has no link to the original file. If the original file is altered, the change is ignored. This ensures, however, that the original file is always visible in the experiment. • Link: Only the link to the file is saved in the column. Please ensure that the file is retained. • Link + in experiment: The file is incorporated in the Experiment-file and the link exists in addition. As long as the file exists, it can be opened via the link. As soon as the file is no longer present, the incorporated file is used. If the file is saved in the experiment, the Experiment-file is correspondingly larger. If possible, the link should be used for large files. Only usable for the types: Picture and Document.
Default value	Value which is displayed before any value has been entered. Only usable for the types: Single-line text.

Type	Description
Entry format	Input format for a numerical input. • 0: Mandatory • #: Optional • Decimal separator: decimal period "." (please don't use decimal comma ",") When the entry format is empty, any arbitrary text can be entered. If the entry format is used, the only possible entry is a number conforming to the specified format. Only usable for the type: Single-line text. Example: 000.0## 001.2 123.4 123456.789 Example: 0.00 1.23 123.45 0.12
Mandatory box	Mandatory if the Setup page is embedded in a dialog. The dialog can only be closed once the entry has been made. In the following dialogs, mandatory box are evaluated: • Command: "Show Panel page as dialog" • Command: "Display Setup dialog" An example is presented in the chapter: Using Mandatory Boxes .
Show most recently use values	When active, a list of the most recently entered values can be opened in the column. Each entry in the list can be selected and used. Only usable for the type: Single-line text.
Most recently use values	For viewing and editing the list of most recently used values.

7.7.2.2.3 Parameter set column

The **Parameter set column** offers the ability to join multiple settings with permanently set values to a **parameter set**. Then, if the parameter sets are pre-defined, the end user must only select the parameter set corresponding to the desired measurement. This saves the user the need to know the entire measurement hardware and its complete settings options.

Creating a column

- Create a parameter set column (Title, Identifier).
- Set which columns (parameters) are joined together in the parameter set.
- This concludes the definition of the parameter set column.

Creating/saving parameters

- Next, make the appropriate individual parameter settings.
- Then click in the Parameter set column and select "- New parameter set -".

Name	Activate	Status	Save (device)	Save (PC)
▼ Channel type: Analog inputs (Count=8)				
Channel_001	Active, without storage	Active	☐	☐
Channel_002		Passive	☐	☐
Channel_003	Active, storage PC	Active	☐	☒
Channel_004	Active, without storage Active, storage PC - New parameter set -	Active	☒	☒
Channel_005	Active, without storage	Passive	☐	☐
Channel_006	Active, storage PC	Passive	☐	☐
Channel_007		Passive	☐	☐
Channel_008		Passive	☐	☐

Example: Parameter set column with the parameters: Channel status, Save (device) and Save (PC).

- Assign a name for the parameter set.

In the parameter, all column settings are saved which are associated with the parameter set column.

In the example above, if a new parameter set belonging to the selected column is created, the following settings are saved:

- Channel status: Active
- Save (device): true
- Save (PC): true

For this column, an appropriate name can be assigned: e.g. *Active, storage PC and device*.



Note

Saving the parameter

The **configuration of the parameters** is **saved** in the respective **supplemental column**. The column itself is saved with the project. Thus, the parameters defined are available in all the project's experiments. The parameters **are only saved** when the **project is saved**.

Selecting a parameter set

When you select a parameter set from the list, all columns adopt the settings saved which are associated with the Parameter set column.

	Name ▲	Activate	Status	Save (device)	Save (PC)
	▼ Channel type: Analog inputs (Count=8)				
	Channel_001	Active, without storage	Active	☐	☐
	Channel_002		Passive	☐	☐
	Channel_003	Active, storage PC	Active	☐	☒
	Channel_004	Active, storage PC and device	Active	☒	☒
	Channel_005		Passive	☐	☐
	Channel_006	Active, without storage	Passive	☐	☐
	Channel_007	Active, storage PC	Passive	☐	☐
	Channel_008	Active, storage PC and device	Passive	☐	☐
		- New parameter set -	Passive	☐	☐

In the example, if the parameter set *Active, storage PC and device* is selected for Channel_005, the following settings are adopted:

- Channel status: Active
- Save (device): true
- Save (PC): true

These settings are adopted because they were previously so defined (see: **Creating/saving parameters**)

As long as no further columns are linked with the Parameter set column, all other parameters remain unchanged.

Renaming/deleting a parameter set

To rename or delete a parameter set, open the selection list in a cell and then open the context menu of the particular item.

Next, select *Rename parameter set* or *Delete parameter set*.

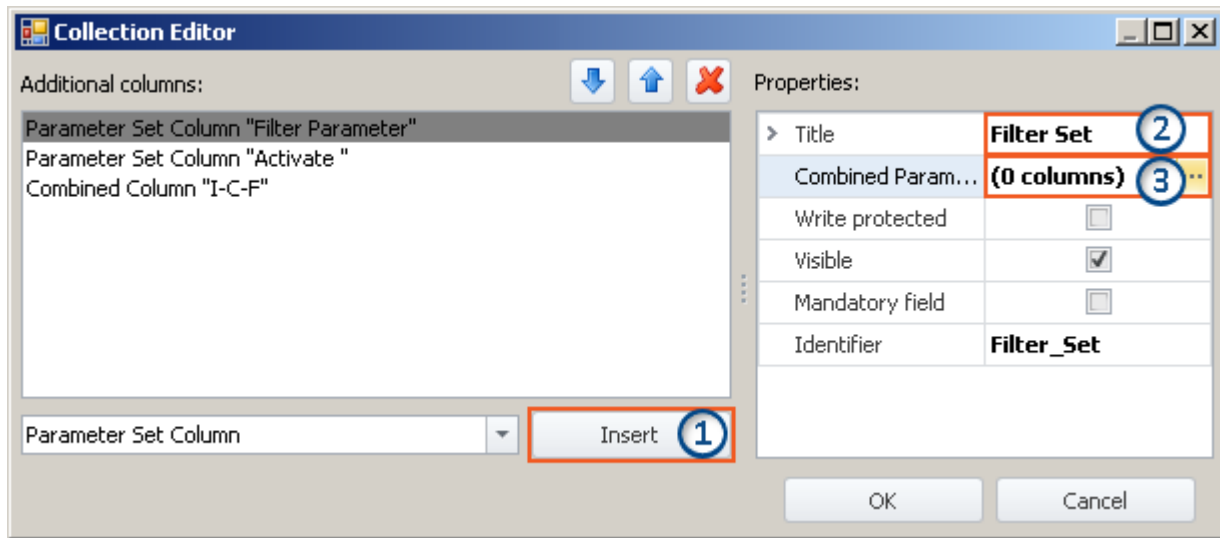
Properties

For shared properties, see: [Additional columns](#) ¹⁵³.

Type	Description
Combined Parameters	List of the parameters which can be set via the column.
Mandatory box	Mandatory when the Setup page is embedded on a Panel page as a dialog. It is only possible to close the dialog once the entry has been made. An example is presented in the description of the command: Show Panel page as a dialog ⁹⁷⁵ .

Example of a Parameter set column

Start the Collection editor as explained in the chapter [Additional columns](#) ¹⁵¹.



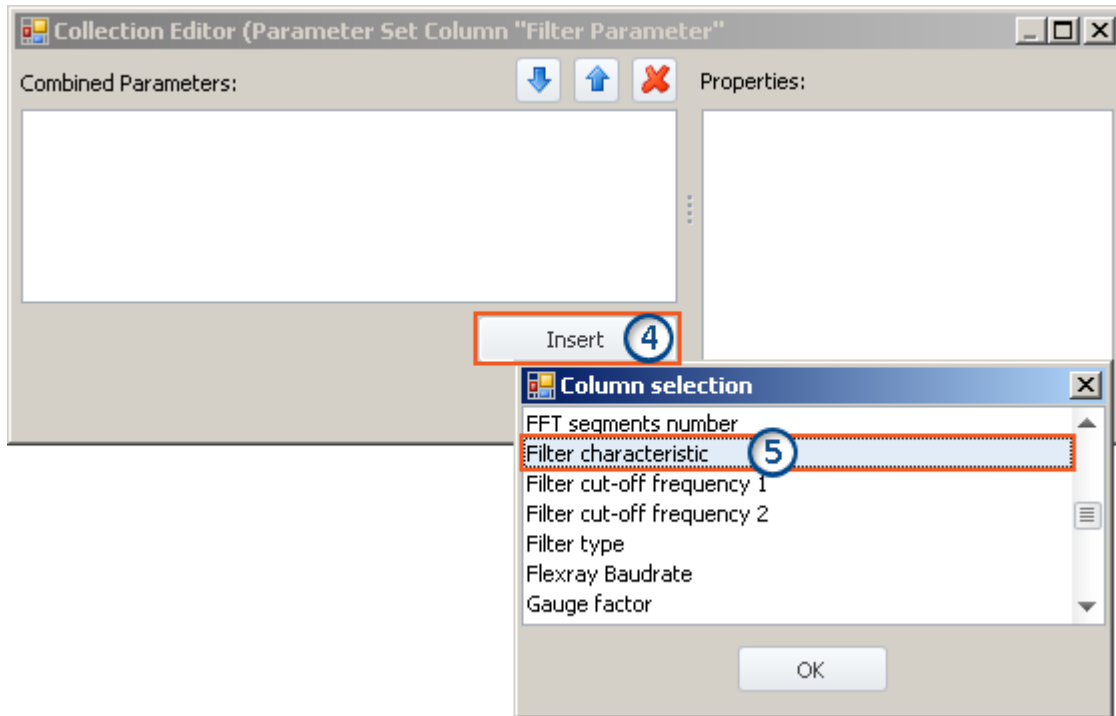
Creating a parameter set column
(example)

- 1) In the drop-down list, select the entry **Parameter Set** and click on **Insert**. This creates a nameless (title) parameter set column.
- 2) Enter the desired title in the input box **Title**, e.g. "Filter Parameter" and hit the **<Enter>**-key (see image above). This entry is automatically used as a unique **Identifier**. Note that special characters (except "_") and spaces should not be used in identifiers. Delete all " " in this example. Instead you can use: "_".

Adding parameters (columns)

3) Click in the row **Combined Parameters** in the Properties window and then on the button with the three dots(...).

This opens a **further window belonging to the Collection editor**. Here you set which columns/parameters are to be combined.



Collection editor, second window, adding column "Filter characteristic"

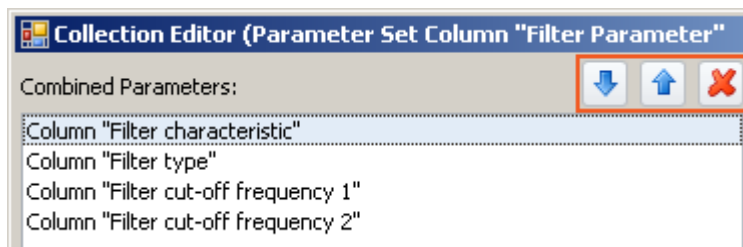
4) Click on **Insert**: An additional window with a list of all available parameters opens.

5) From the list, select the desired parameter(s) (in this example: **Filter characteristic**, **Filter cut-off frequency 1**, **Filter cut-off frequency 2** and **Filter type**).

Sort

To sort entries, use the "Up/Down"-arrow icons ( / ).

Sort the order of "Filter characteristic", "Filter cut-off frequency 1", "Filter cut-off frequency 2", "Filter type", as shown below:



*List of the combined parameters
(after sorting)*

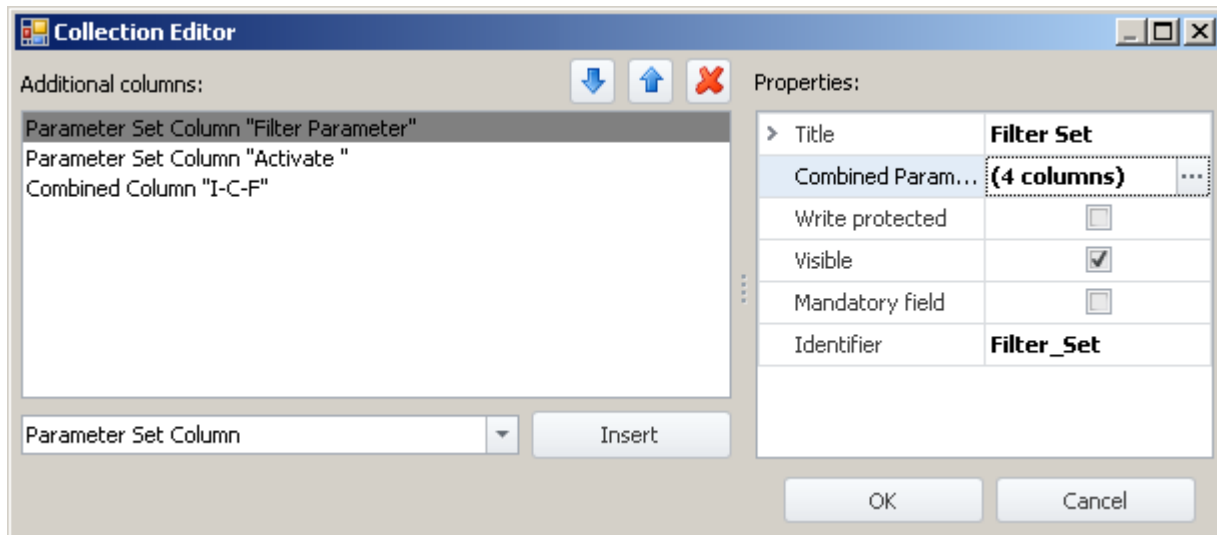
**Note****Why the sorting is important**

The sorting determines the order in which the parameters are set.

For instance, the column: *Filter cut-off frequency 2* can only be assigned a value if the column is "active". In other words, if the column is required for the filter type used, as is the case with Bandpass, for example.

In consequence, it is necessary to set the parameter values in the order which their respective dependencies requires. The filter type must be set before the cut-off frequency.

Close the second window by clicking **OK**. You then see the first window with the configured column "**Filter Parameter**" and the note indicating that "(4 columns)" have been combined:



Collection editor, first window, parameter set column "Filter Parameter" (4 columns)

Close the window by clicking **OK**. You now have a new column in the channel list, by means of which you can save and load the parameters of the four columns.

7.7.2.2.4 Parameter set import column

By using the "*Parameter Set Import Column*", it is possible to quickly load a variety of settings. Selecting particular entries automatically selects the associated saved parameter set (*.csv).

Operation technique, and distinction from the Parameter Set Column:

The [Parameter Set Column](#)^[162] is bound to its associated columns. By contrast, the **Parameter Set Import Column** operates like the command: [Parameter set import](#)^[559]. A parameter set is imported and all settings belonging to it are made.

Parameters which are not in the file retain their values.

Example of a Parameter Set Import Column

At a certain test station, two engine types are routinely tested. For each type, there are 3 test procedures, having varying designations (metadata) and channel settings.

For each of these six "measurement states", one parameter set (*.csv) is generated. The following is a sample file:

Company name	Test object number	Test part number	Test station photo	Hint
My company	X1	X1_123	.\Picture1.png	speed test 1
Channel name	Save (PC)	Sampling time	Duration	
Channel_001	True	1 ms	40 s	

These files are saved in folders as follows:

```
Engine types
+---X1
|   speed test 1.csv
|   +---speed test 1.data
|   |   \---DevSetup
|   |       Picture1.png
|   speed test 2.csv
|   +---speed test 2.data
|   |   \---DevSetup
|   |       Picture2.png
|   low.csv
|   \---low.data
|   |   \---DevSetup
|   |       Picture3.png
+---X2
|   speed test 3.csv
|   speed test 5.csv
|   low 3.csv
|   \---low 3.data
|   |   \---DevSetup
|   |       Picture4.png
```

The Engine Types X1 and X2 are folders in which the csv-files are located. Along with the csv-file, there can be a folder of the same name with the extension "*.data" containing additional files. These files can be used as linked metadata (e.g. pictures).

On the Setup-page, two columns are set up to conform to this folder structure.

- First column: First folder level: In this example, the engine type can be selected here (folder name: "X1" or "X2")
- Last column: File level: In this example, the test procedure can be selected here (name of the existing csv-files)

Engine type	X1
Test run	speed test 1
	Low
	speed test 1
	speed test 2

If an entry is selected in the first column, the content of the second column adapts according to what csv-files are present.

If an entry in the second column is selected, the csv-file is imported.

Properties

For shared properties, see: [Additional Columns](#)  153.

Type	Description
ImportRoot	Path to the first folder level
Column captions	Optional list: Designations of the column titles • No specification: columns have the default name: Parameter Set Level 1,2,3,... • Specification provided: Columns are assigned the pre-defined name. See example: <i>Engine type</i> and <i>Test run</i> To edit the list, click on the button with the three dots ().

Note

- Multiple subfolders are also possible. For each additional subfolder, a separate column is created. However, a csv-file is imported only to the lowest level.
- Parameter Set Import uses for **Mapping: Import mapping by name**. When channel parameters are imported, the names of the channels must match the names in the csv-file.
- Parameter Set Import performs importation across Table Descriptions. This means that the csv-file can contain parameters for a variety of different Setup pages. In this example, there are parameters for the pages:
 - Documentation (table description: Experiment)
 - Analog channels (table description: Analog Channels, Digital Channels and Channels, ...)
- The column type is only present on the Setup pages: *Documentation* or *Experiment*.

7.7.2.3 Using mandatory boxes

When Setup pages are called in dialogs, it is possible to prevent the dialogs from being closed. In this case, it is only possible to close them once the important "mandatory entries" have been made.

Only metadata columns can be defined as so-called "Mandatory boxes".

It is not possible to use this function to verify any other entries (for instance, any entries made via Widgets or entries of device/channel-parameters). The verification of the validity of entries is based on what type and what format you had previously selected for the metadata column. So, for the type *"Folder"*, for example, there is a check of whether the character \ is included, unless a user-defined format was used.

A mandatory box exclusively affects the behavior of the following functions:

- the command "Show Panel page as dialog"
- the command "Display Setup dialog"

It has no effect on the normal Setup pages.

To illustrate this function and its use, the following examples are presented.

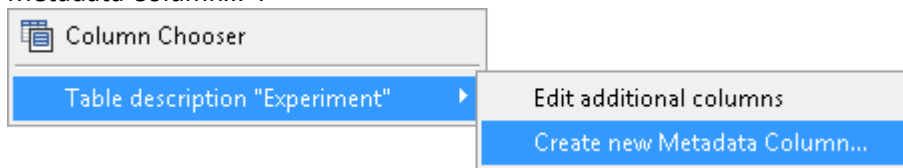
Example: Creating a Panel page as a dialog with a mandatory box

In this example, a new metadata column *"User"* is added to the already existing Setup page *"Documentation"* and declared as a mandatory box. Next, a Panel page is created on which the Setup page *"Documentation"* is displayed. This Panel page is called as a dialog, in such a way that the dialog can only be closed by clicking "OK" if there is an entry in the mandatory box.

Note: The page *"Documentation"* displays the columns of the table description named *"Experiment"*.

Open the page *"Documentation"* in the **Setup**

Open the context menu over the white area and select *"Table description "Experiment" > "Create new Metadata Column..."*.



Context menu of the *"Documentation"* page
(Table description: *"Experiment"*)

A dialog opens, which is named *"Collection editor"*. Click on the button *"Insert"*. A column with no title initially opens.

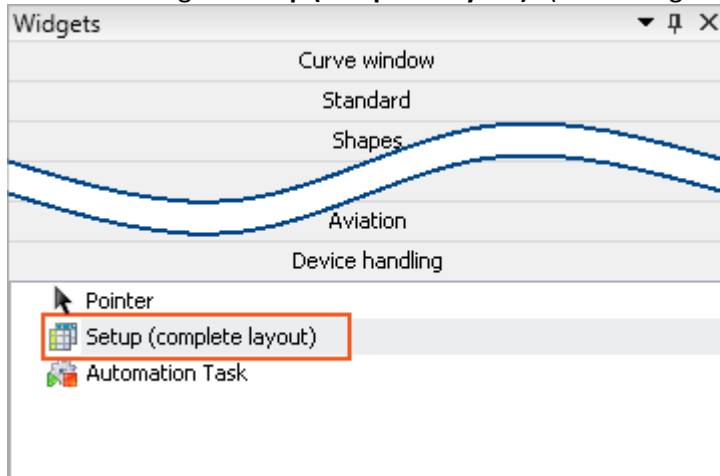
Enter the following **properties**:

- Title: User
- Type: Single-line text (unchanged)
- Mandatory box: ☒

All other settings remain intact. Close the dialog with *"OK"*. The new column is now present on the page *"Documentation"*.

Next, go to **Panel**.

Insert the Widget "**Setup (complete layout)**" (from the group "*Device handling*") into the Panel page.



Device handling - Widget "Setup (complete layout)"

Open the Widget's properties and select the **Layout**: "*Documentation*".

Note

If the new column is not visible, perform the following steps:

To display it, open the context menu in the Widget and select the menu item "Column Chooser". There, select the "User" (Tab: "Rows") column and use Drag&Drop to move it to the Widget.

Jump to **Sequencer** and there, add the command "*Show Panel page as dialog*".

Configure the command:

- Panel page to be displayed: Select the previously configured page
- Display dialog buttons: Yes
- OK-button: Active: if all entries are valid

All other settings remain intact. Close the configuration by clicking "OK"

Next, **start** the **Sequencer**. The dialog opens and only closes again when "*User*" is assigned a **valid value** (here: text).

Example: Defining the existing column "**Test engineer**" as a mandatory box

On the Setup page "*Documentation*", you find the metadata column "*Test engineer*". We will define it as a mandatory box.

Note: The page "*Documentation*" displays the columns of the table description named "*Experiment*".

Note

It is not possible to edit existing columns with the Standard version of imc STUDIO!

In order to access existing metadata columns, open the properties of the table description "Experiment":

- Jump to the Setup
- If necessary, start by turning on the visibility of the tool window "Table Descriptions" (ribbon: "View" > "Choose Tool Windows"). The tool window is not available in the Standard version of imc STUDIO.
- Open the context menu of the table description "Experiment" and select "Properties".

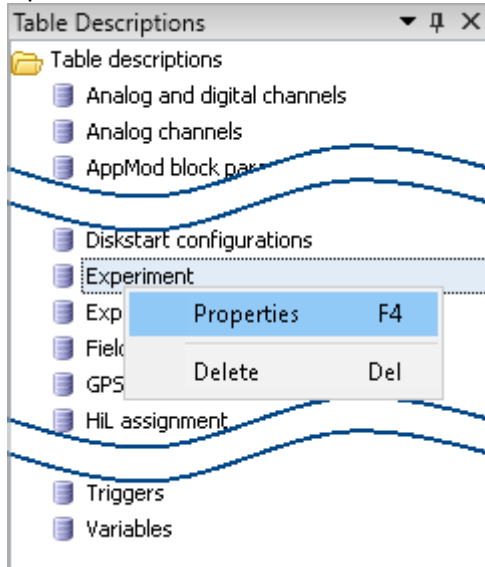
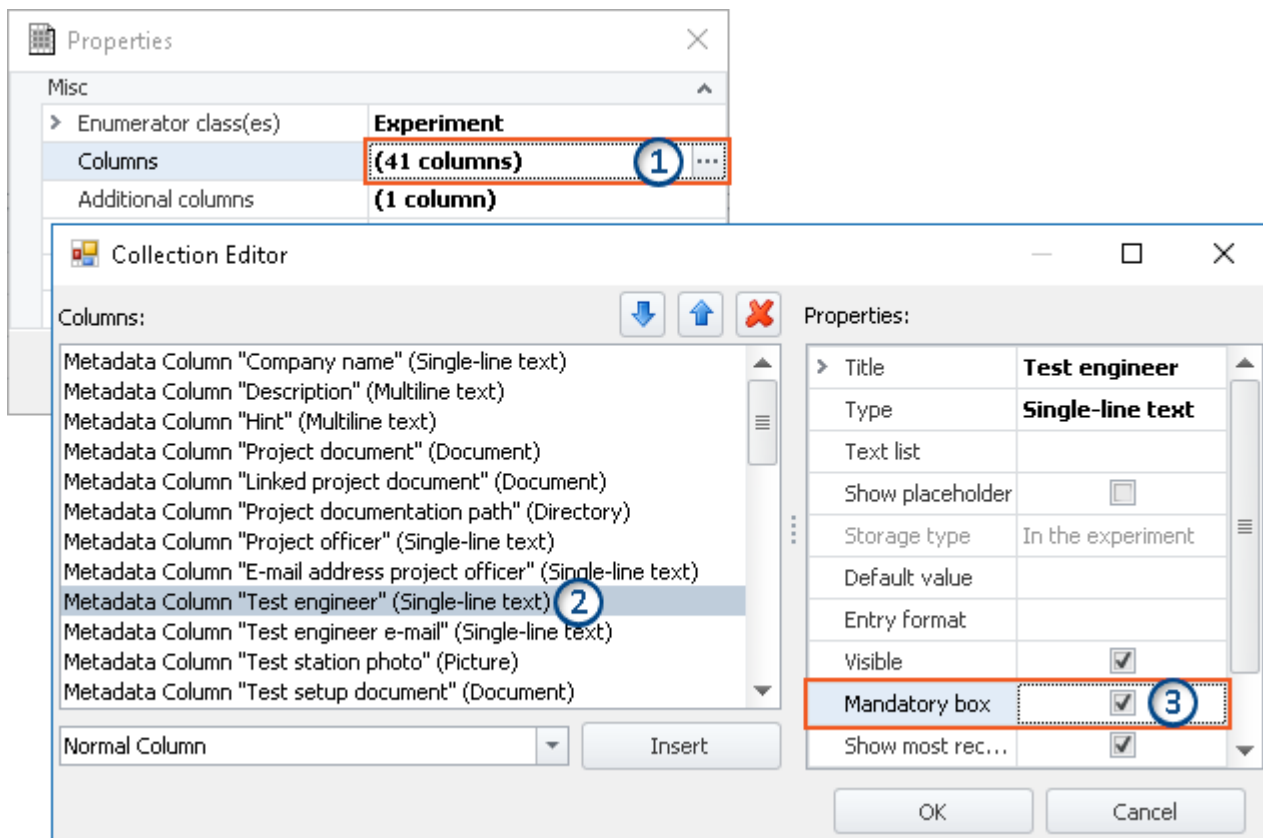


Table Descriptions tool window

Open the list of existing columns:

- In the Properties dialog, left-click the mouse over the column named "Columns" (1).
- Open the list by clicking on the button (...). A dialog named "Collection Editor" appears, which contains the properties of the individual metadata columns.



Properties and list editor for the table Experiment

Define the column "Test engineer" as a mandatory box:

- In the listing editor, select the column "Test engineer" (2)
- Put a check mark in the box for "Mandatory box" (3)
- Close the dialog with "OK".

Next, display the page "Documentation" in a dialog. The dialog only closes again when "Test engineer" is assigned a **valid value** (here: text).

7.7.2.4 Export/import of column/table descriptions

You are able to export/import complete table descriptions and individual column descriptions in a targeted way. For instance, you could transfer defined additional columns from one table description to another one. Or, you can exchange table descriptions without needing to import the entire view.

Exporting column/table descriptions

Using the menu item "Import/Export", you can **save the column and table description**:

Ribbon	View
Project > Import/Export ()	Complete
Extra > Import/Export ()	Standard

- Select the option "Export", and in the list, select the entry "Export of Setup table description" or "Export of Setup column descriptions"
- From the list offered, select the desired elements (a table description or one or more column descriptions belonging to a table description)
- Select an appropriate file name and path for the export file



Note

Save the view beforehand

Save the view beforehand if you have made any changes (see "[Views](#)" ).

Import of column/table descriptions

Using the menu item "Import/Export", you can **import the column - and table descriptions**:

Ribbon	View
Project > Import/Export ()	Complete
Extra > Import/Export ()	Standard

Importing a Setup table description

- Select the option "Import" and from the list, the entry "Import of Setup table description"
- In the file selection dialog, select the saved file

The table description is imported along with all columns it contains. If any table description having the same "ID" exists, a prompt appears. Here, you can either replace the existing description, or rename the "ID" of the new table description.

Please note that the "ID" is only the internal ID. The displayed name (e.g. in the tool window: "Table description") is not changed. If you have renamed the "ID", it is now possible that two table descriptions having the same displayed name, but different IDs, exist. This situation is not recommended.

**Note****New functions can be overwritten**

Please be aware that when making replacements, new functions may be lost if you import old views/columns.

Importing Setup column descriptions

- Select the option "*Import*" and from the list, the entry "*Import of Setup column descriptions*"
- In the file selection dialog, select the saved file
- From the list offered, select the target table description

All columns are imported to the selected table description. Table descriptions are not offered if they have columns with the same "IDs".

**Reference****See also**

Exporting/importing [Views](#) 

7.7.2.5 Configuring and designing the pages**7.7.2.5.1 Table description and Complete layout**

Terms	Description
Table description	Collection of normal and additional columns
Column description	Configuration of a column in the table description
Complete Layout	Finished designed Setup pages. The page is filled with elements which are linked with table descriptions or columns belonging to table descriptions.
View	Arrangements of windows, menus, layouts, column arrangements, ...

The "*Table description*" **contains a variety of columns**. Based on these columns, the individual Setup pages are created. To do this, **the desired columns** belonging to one or more table descriptions are **selected and ordered**. The **new page** created can be **saved** as a "*Complete layout*" (as of imc STUDIO Professional) and thus made available e.g. for the metadata export.

In the "*Complete Layout*", only the information regarding the "visual" appearance is saved:

- which columns are displayed, and
- how and where they are displayed.

The columns' **values** are linked to the "*Table description*". This ensures that the values in all Setup pages are updated simultaneously.

**Example**

The column "*Channel name*" is used both on the page "*Analog channels*" and on the page "*Channel balance*". Now, if you change the channel name on one of these pages, it is automatically updated on the other page, since it affects the same column which simply appears on two different pages.

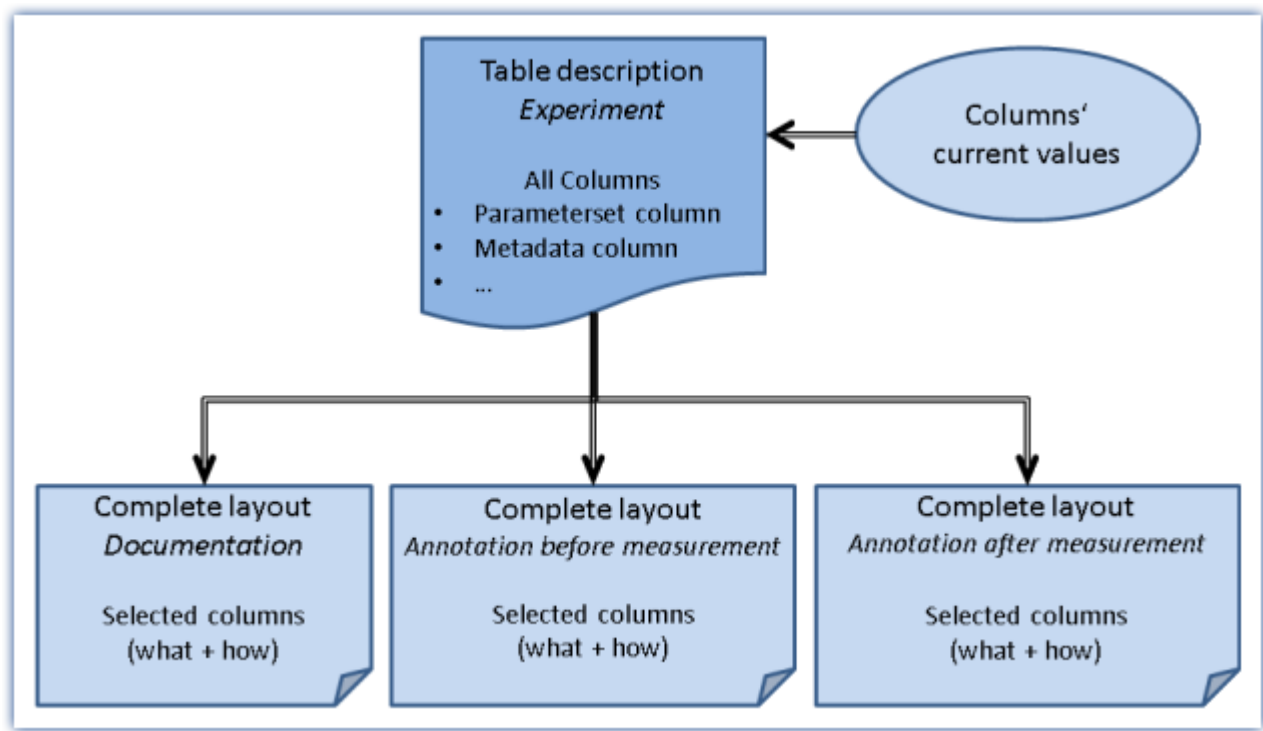


Table description and complete layout

8 Setup pages - Configuring device

Each measurement setup can have a variety of requirements in terms of measurement device settings. Configure the device and each measurement channel according to its requirements. For this purpose, various software Assistants (wizards) are provided.

The following chapters describe the individual parameters of all pages (Setup pages) for configuring the measurement device.



Note

Parameters not visible

The subsequent chapters describe the complete device configuration procedure. Many of the parameters can not be changed in Monitor, or are not visible.

Page	Description
 Documentation ¹⁷⁷	On this page, you can supply the experiment with descriptions and metadata .
 Devices ¹⁷⁷	Here, select the device for the measurement. Configure the data storage ¹⁸⁰ , among other settings.
 Analog channels ¹⁸¹	Here, configure the channel-specific parameters. Among others, those for the analog inputs and virtual channels .
 Digital channels ¹⁸¹	Here, configure the channel-specific parameters. Among others, those for the digital inputs/outputs and incremental counter channels .
 GPS ¹⁸¹	Here, configure the channel-specific parameters relating to the GPS-channels .
 Variables ¹⁸¹	Here, configure the channel-specific parameters relating to the Display variables and process vector variables .
 Storage options and directory structure ¹⁹⁴	Saving measured data on the PC and in the device
Storage media in the device ²²¹	This section describes how to handle the storage media of the imc measurement devices and how to use them with imc STUDIO Monitor.

8.1 Documentation (experiment description)

On the page "Documentation", you can supply the experiment with descriptions and metadata.

Project	
Company name	
Hint	
Project document	
Linked project document	
Project documentation ...	
Project officer	
E-mail address project ...	
Test engineer	
Test engineer	
Test engineer e-mail	
Test setup	

The page is subdivided into two areas: the **Table** below and a **Description**.

In the table, you can enter information pertinent to the experiment.

The table can be filled with multiple columns (see [Configuring columns](#) ¹⁵⁰). By opening the **Column Chooser**, you can additionally move columns, group them together, or delete them.

In the box **Description**, you can provide your experiment with up to multiple lines of explicit description.

8.2 Configuring devices

On the page **Devices**, you can set all device-specific parameters.

Selected	Device name	SN	Device specification	Connection status
☒	T_126678_CS_7008_1	126678	imc C Series	disconnected
☐	T_126680_CS_7008_1	126680	imc C Series	
☐	T_140578_CRC_400	141578	imc CRONOScompact	
☐	T_141692_CS_7008_1	141692	imc C Serie	

Below the table are additional tabs: Storage, Synchronization, Timestart, and Measurement options.

"Devices" page in the plug-in "imc STUDIO Setup"

The page is subdivided into two areas: the "Device table" and the "Dialogs".

In the Devices table, all known devices are listed. It offers quick access to many settings. In the lower area, you can select various dialogs. These offer clearly-presented access to the properties.

The availability and selection of the settings depends on the device type, which determines what settings options the parameters can have.

The following dialogs may be present:

Page	Description
 Storage ¹⁸⁰	Settings for saving measured data in the device and on the PC.
 Synchronization	No function
 Timed start	No function
 Measurement options	No function
 Reference	See also additional info on:
• Operation and configuration of the tabular display ¹³² and the dialogs ¹³⁹ • Showing and moving columns ¹⁴⁸ • Creating and configuring additional columns ¹⁵¹	

8.2.1 Device table

In this area, the [known](#) ¹⁴² **devices** are displayed in a structured table, as seen in the example image below.

 Documentation

 **Devices**

 Analog channels

 Digital channels

 GPS

 Variables

Selected	Device name	SN	Device specification	Connection status
☐	T_123020_CRPL_2_DIO	123020	imc Cronos PL	
☐	T_123794_CRPL_2_S	123794	imc Cronos PL	
☐	T_120640_C1	120640	imc C1	
☐	T_120688_C1	120688	imc C1	
☐	T_130039_busDAQ_X	130039	busDAQ-X	
☐	T_121582_PL2_UNI8	121582	imc Cronos PL	

Device-Table (example)

To be able to **use a device** for the current experiment, you must first [find and select](#) ¹⁴² it. To **adapt the devices' settings**, select one or more devices. After that, you either [edit the table cells](#) ¹³⁵ directly or select the respective [dialog](#) ¹³⁹ for the desired parameters.

Parameter	Description
	<i>long name</i> <i>short name</i> <i>column ID</i>
Selected	Device is selected for measurement Selected Dev_SelfForMeasurement

Clicking on the "Selected" box which corresponds to the desired device makes that device available for the experiment. You can also select multiple devices for your experiment.

See "[Device Search - Known and Selected](#)" ¹⁴²

Parameter	Description		
	<i>long name</i>	<i>short name</i>	<i>column ID</i>
Device name	<i>Device name</i>	<i>Name</i>	<i>eDeviceNickname</i>
The displayed name of the device. You can modify the name if desired. The device's name must be unique; i.e. no two devices may have the same name. Maximum number of characters: 24. In order to be able to modify the name, first establish a "<i>Connection</i>" with the device. After having made the change, perform the "<i>Prepare</i>", so that the change is saved in the device. The device's name is also saved with the experiment. As soon as you load an experiment with a different device name and prepare it, the changed name is overwritten.			
SN	<i>Device SN</i>	<i>SN</i>	<i>eDeviceSN</i>
Unique identification number for the imc devices.			
Device specification	<i>Device specification</i>	<i>Specification</i>	<i>eDeviceProduct</i>
imc device type. Reference this name, for example, to obtain the associated device documentation.			
Connection status	<i>Connection status</i>	<i>Connection</i>	<i>eConnectionStatus</i>
Indicates the current connection status: " <i>connected</i> " or " <i>disconnected</i> ".			
Measurement status	<i>Measurement status</i>	<i>Measurement</i>	<i>eMeasurementStatus</i>
Indicates the current status of the measurement: " <i>stopped</i> ", " <i>running</i> " or " <i>reconfigured</i> ".			
Device control	<i>Device control</i>	<i>Device control</i>	<i>eDeviceControlAction</i>
You can select an action via the selection list. By clicking the mouse over the button, you can cause the device to run the action. In contrast to menu items, this action only applies to the device selected. Thus you can stop a device during measurement, modify it in some way, and restart it, without affecting the other devices. The only other effect is that from that point in time, measured data are saved in a new folder.			

8.2.2 Storage

 Reference

This chapter contains a brief introduction to saving channel signal data. More detailed descriptions of all functions are presented in the section: "[Storage options and directory structure](#)"^[194].

 Storage

 Synchronization

 Timestart

 Measurement options

Experiment options (applies to all devices) - Device

☐ Save trigger events in individual files

Device specific options - Device

Storage location

Removable medium

Network storage settings

Storage interval

1 h

Storage interval count

4

Experiment options (applies to all devices) - PC

☐ Save trigger events in individual files (PC)

Storage location

C:\Users\Public\Documents\DB\StandardProject\Experiment_0001

Storage interval

End of measurement

Storage interval count

all

"Storage" dialog

This dialog set s the options for saving the measured data to the device and to the PC. You can decide how to name folders and where to place them. You can also set at what intervals to save the measured data.

Parameter	Description
Save trigger events in individual files	 See "Save trigger events in individual files"^[212]  In order to display the measured data associated with the events, add the column "Event time"^[598] manually in the Data Browser.
Storage interval / Storage interval count	 See "Saving interval"^[202]  When interval storage is active, all folders are deleted except for the specified number of intervals!

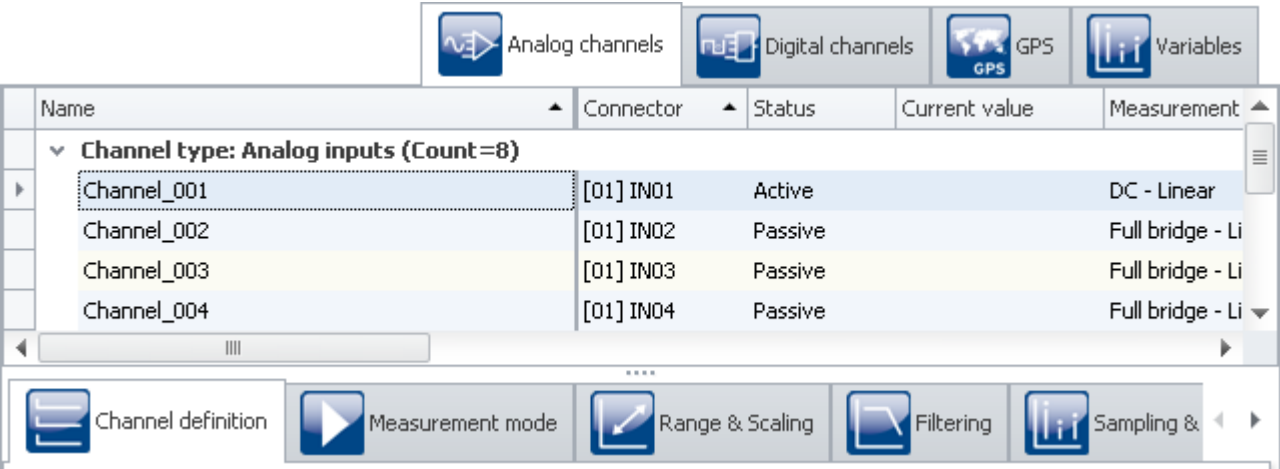
© 2025 imc Test & Measurement GmbH

imc STUDIO Monitor 2025 - Manual, Doc. Rev.: 8.1 - 2025-03-11

Page 180

8.3 Configuring channels and variables

On the pages **Analog channels**, **Digital channels**, **Variables** and **GPS** you can set all channel-specific parameters.



The four pages are subdivided into two areas: the *Channel table* and the *dialogs*.

In the channel table, all channels are displayed which are present in the [selected measurement devices](#)^[142]. It offers quick access to many settings. In the lower region, you can select various dialogs. These offer clearly-presented access to the properties.

The availability and selection of the dialogs and of the settings depends on the channel type, i.e. on the amplifier hardware installed in the device. Also depending on the channel type, there are different applicable settings options for the parameters.

The following dialogs may be present:

Page	Description
 Channel definition ^[185]	Name and Status of the channel.
 Sampling & Preprocessing ^[186]	Configuration of the sampling and the measurement duration , of the data type and the pre-processing , such as mean value, RMS-value, or maximum/minimum.
 Data transfer ^[188]	Configuration of the channel-specific data saving on the PC or the device. Circular buffer data storage and event count in the curve window and the data storage.
 Histogram / Rainflow ^[189]	Refresh rate and data saving rate for histograms and other matrices.
 Curve properties ^[189]	Pre-defined settings for how the channel is to be displayed. Defined channel-color for the curve window. Specified range for the display elements (Widgets).

 **Note**

Additional parameters

Not all parameters can be found in the dialogs. For special applications, additional settings are also required. In the chapter "[Additional Parameters](#)"^[191], you will find description s of the most important additional parameters.

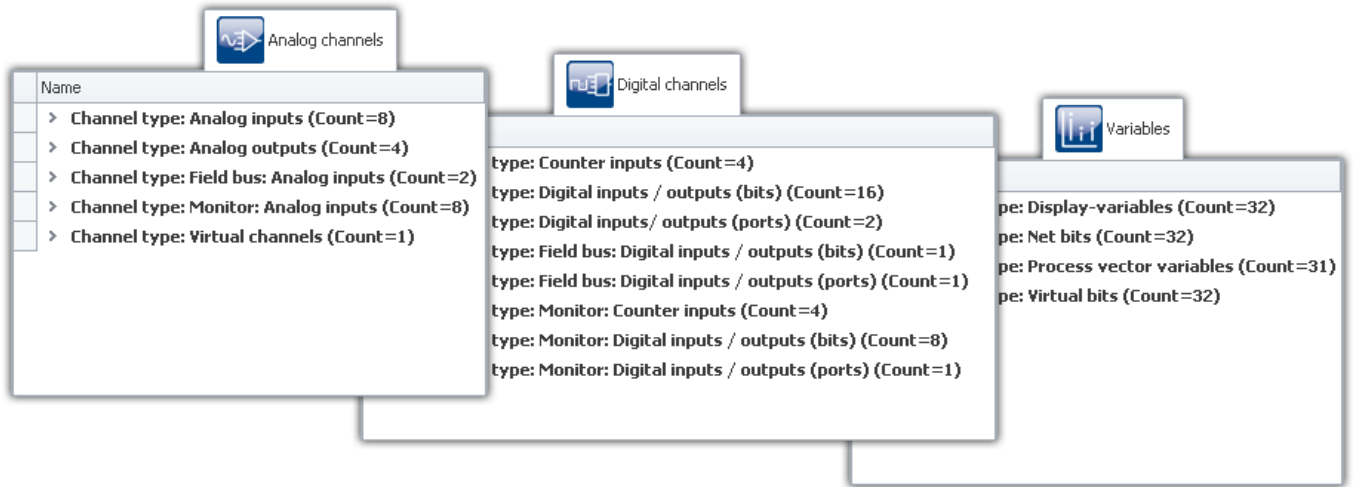
 **Reference**

See also additional info on:

- Operation and configuration of the [tabular display](#)^[132] and the [dialogs](#)^[139]
- [Showing and moving columns](#)^[148]
- [Creating and configuring additional columns](#)^[151]

8.3.1 Channel table

In this area, the **channels** belonging to the selected devices are displayed in a structured table, as shown below:



Channel table

In the channel table, all the channels of all devices appear in the list, along with their configurations. The list is arranged in groups, each of which is represented by a branch (or: node) in the list. The group branches can be expanded or collapsed, thus determining whether or not the entries for the channels in the branch are displayed.

To select channels, first open a branch (▶). Then select one or more channels. Subsequently, you can either [edit table cells](#)^[135] directly, or open the [dialog](#)^[139] for the desired parameter.

All the settings controls in the dialogs take effect on the channels which are selected in the table.

The channels are grouped as follows:

Analog channels	Description
Analog outputs	Analog outputs
Analog inputs	Analog data acquisition channels
Fieldbus: Analog inputs	CAN-interface required (CAN, LIN, etc.)
Virtual channels	Channels created with imc Online FAMOS
Monitor: <xyz> ^[527]	Digital copies of the input channels which have the same name. With monitor channels it is possible to show slowly-sampled signals recorded before the trigger event.
Digital channels	Description
Digital inputs / outputs (bits and ports)	Depending on your device hardware configuration.
Incremental counter inputs	e.g. ENC4 or INK4 required
Fieldbus: Digital inputs / outputs (bits and ports)	CAN-interface required (CAN, LIN, etc.)
Monitor: <xyz> ^[527]	Digital copies of the input channels which have the same name. With monitor channels it is possible to show slowly-sampled signals recorded before the trigger event.
GPS	Description
GPS-Channels ^[533] (internal "Fieldbus: Analog inputs")	GPS-data as time-domain channels.

Variables	Description
Display variables	The optionally available hand-held display unit interacts with the device via display variables. In effect, Display variables are virtual bits - they can be used for controlling the display of measurement data or for directly influencing a measurement process. These variables can also be used as temporary storage for a number in imc Online FAMOS or for the purpose of display/operation on a imc STUDIO Panel page.
Ethernet bits  527	Up to 32 ethernet bits are available for Ethernet devices. Events can be assigned to ethernet bits and used as triggers in an Ethernet.
Process vector variables  531	The " <i>Process vector</i> " is a collection of single-value variables (process vector variables or pv-variables). For all measurement channels of the device, these single-value variables are already pre-defined. In parallel to the already existing device-pv-variables, you can create your own pv-variables.
Virtual bits	The virtual bits behave like a 32-bit register, they can be polled and/ or set. This makes it possible to keep a record of states occurring during measurement and, for example, to use the info in the trigger machine.

Maximum channel count per device in firmware group A (imc DEVICES)

Maximum channel count per device									
Active channels within a systems...		512		Active channels of the current configuration: Total number of analog, digital, fieldbus and virtual channels, as well as monitor channels, if any.					
...of which active analog channels		198 ⁽¹⁾		Active analog channels of the current configuration (sum of primary channels + monitor channels)					
Fieldbus channels		1000		Number of defined channels (active and passive); Currently activated channels are limited by the total number of activated channels (512).					
Process vector variables		800		Single value variables, each containing the last measured values. A process vector variable is automatically created for each channel.					
		without monitor channels				with monitor channels			
Channel type	determined by	limit (active+passive)		activated	total activated	limit (active+passive)		activated	total activated
Analog channels	system-expansion	Channel	240	198 ⁽¹⁾	512	Channel	240	198 ⁽¹⁾	512
						Monitor	240		
Incremental counter	system-expansion	Channel	16	16		Channel	16	16	
						Monitor	16	16	
DIO/DAC-Ports	system-expansion	Port	16	16		Port	16	16	
						Monitor	16	16	
Fieldbus channels	flexible	Channel	1000	512		Channel	1000	512	
						Monitor			
Virtual channels (OFA)	flexible	-	-	512			-	-	

Occupancy for ports (examples):

- one DO module (e.g. DO-16) occupies 1 port
- one DI8-DO8-ENC4-DAC4 module occupies 3 ports
- one DAC module (e.g. DAC-8 or DAC-4) occupies 1 port



Monitor-ports: DI-ports (respectively channels) have monitor-ports, DO/DAC-ports in contrary do not have monitor-ports

(1): 128 with imc CRONOSflex (CRFX) and imc CRONOS-XT (CRXT), incl. output channels of type DAC-8 and DIO-Ports of type DI / DO, incl. 18 channels per CRFX/WFT-2 input

Maximum channel count per device in firmware group B (imc DEVICecore)

Maximum channel count per device		
Active channels within a systems...	1000	Active channels of the current configuration: Total number of analog, fieldbus and virtual channels, as well as monitor channels, if any.
...of which active analog channels	1000	Active analog channels of the current configuration (sum of primary channels + monitor channels)
For fieldbus log channels	any number of channels	Log channels: non-decoded CAN traffic ("dump")

8.3.2 Channel definition

Channel definition

Channel type

Analog inputs

Bridge, current and voltage amplifier and temperature measurement unit "UNI2-8"

Calibration date 01.06.2016

Name

Channel_001

Comment

Status

Active

Curve color

auto

Sensor

Read sensor information

Parameter	Description		
	<i>long name</i>	<i>short name</i>	<i>column ID</i>
Name	<i>Channel name</i>	<i>Name</i>	<i>eChannelName</i>
The channel's Name must be unique, i.e. no two names may have the same name. Maximum number of characters: 65. Three more characters may be added for the sake of any pv-variables.			
Comment	<i>Channel comment</i>	<i>Comment</i>	<i>eChannelComment</i>
The Comment contains a text for explanation concerning the channel. The comment is saved along with the channel's settings in the experiment, and can be displayed in the Panel, for example.			
Status	<i>Channel status</i>	<i>Status</i>	<i>eStatus</i>
The Status determines whether the particular channel is activated for the measurement/the experiment. • Active: Activates the channel • Passive: The channel is not recorded. However, the settings are kept.			
Color	<i>Color</i>	<i>Curve color</i>	<i>eCurveColor</i>
Pre-defined setting for the channel-color in the curve window.			

Calibrating the amplifiers

At delivery, the amplifiers are freshly calibrated. Routine calibration ensures reliable measurement. Along with the information on the amplifier type, the calibration date is also displayed.

If you have any questions about your measurement amplifier's calibration, please contact our tech support.

8.3.3 Sampling & Preprocessing

Sampling & Preprocessing

Channel nameChannel_001

Sampling

Sampling rate100 Hz

Sampling time10 ms

Data type16-bit integer

Sample countUndefined

DurationUndefined

X-axis

Preprocessing

FunctionNone

Points1

Resulting sampling rate100 Hz

Resulting sampling time10 ms

Sampling & Preprocessing

The availability and selection of settings depends on the channel type.

Parameter	Description		
	long name	short name	column ID
Sample count	Sample count		SampleCount
	The amount of measured values per measurement. Select <i>undefined</i> (or enter "0") for unlimited measurement duration.		
	When the Sample count is entered, the resulting measurement duration is corrected accordingly. The measurement duration is given by: Sample count * Sampling time.		
Duration	Duration		eDuration
	The duration of the measurement. Select <i>undefined</i> (or enter "0") for unlimited measurement duration.		
	When the measurement duration is entered, the resulting sample count is corrected accordingly. The sample count is given by Duration/Sampling time.		

© 2025 imc Test & Measurement GmbH

imc STUDIO Monitor 2025 - Manual, Doc. Rev.: 8.1 - 2025-03-11

Page 186

Settings for fieldbus channels

The data of these channels are captured via a special Fieldbus interface. The necessary definitions are made by use of the Fieldbus Assistant. Fieldbus channels are also included in the trigger machine.

Channel type	Description
Fieldbus: Analog inputs	The Fieldbus channels carry data acquired by sensors which are connected to a fieldbus such as a CAN-Bus.
Fieldbus: Digital inputs / outputs	The Fieldbus channels carry digital data acquired by sensors which are connected to a fieldbus.

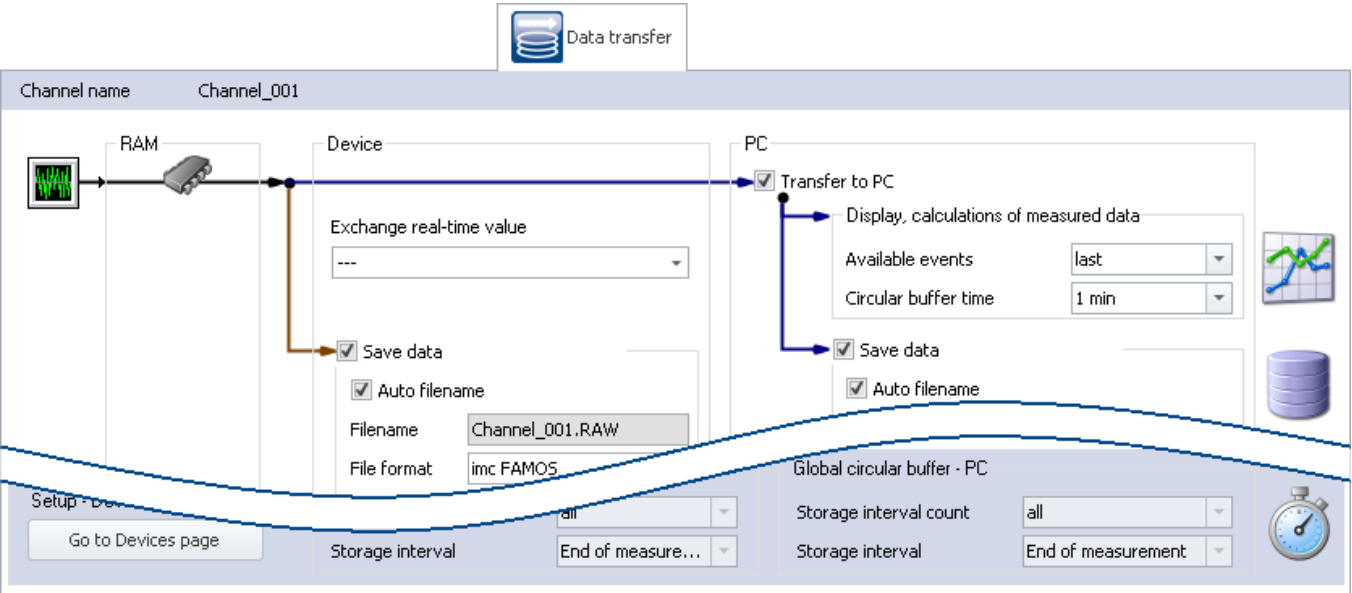
If Fieldbus channels are selected in the Channel table (assuming the device is configured correspondingly), there are additional parameters to set:

Parameter	Description		
	<i>long name</i>	<i>short name</i>	<i>column ID</i>
X-axis	<i>X-axis</i>		<i>eXFormatVariable</i>
	• <i>Sampling rate</i>: The setting Sampling time is related to the settings Duration or Sample count and Sampling rate/time as previously discussed. • <i>Time stamp</i>: Data acquisition is handled by external, asynchronous sampling within the fieldbus. The corresponding time coordinate for the data point is passed as a time stamp along with the data for subsequent processing. In this case, the parameters Sample count and Sampling rate/time are irrelevant.		
Duration	<i>Duration</i>		<i>eDuration</i>
	If the parameter X-axis is set to <i>Time stamp</i> , the duration of the measurement depends only on the setting in Duration .		
Assignment	<i>Assignment</i>		<i>eAllocation</i>
	The data can additionally be assigned to a Display variable (only for analog Fieldbus inputs).		

8.3.4 Data transfer

Reference

This chapter contains a brief introduction to saving channel signal data. More detailed descriptions of all functions are presented in the section: "[Storage options and directory structure](#)"¹⁹⁴.



Transfer to PC

For transfer of the measured data to the PC, there are two regions of the dialog:

Region	Description
Display, calculations of measured data	Temporary storage of measured data. E.g. for display of a Panel page and for subsequent processing.
Save data	Saving of measured data in the experiment folder. The data structure is set in the device settings under Storage¹⁸⁰. Per interval or measurement, one file is created per channel, which contains the respective measurement files

Each of the regions is set separately.

Reference

See also

- Storage options: [Settings for the channels](#)¹⁹⁶
- Available events and Circular buffer time: see "[Events and circular buffer](#)"²⁰¹

8.3.5 Histogram / Rainflow

Only visible for certain virtual channels generated in imc Online FAMOS using appropriate functions.

Here you can set defaults. These will affect the display and storage of the data of the respective virtual channel:
e.g. Histogram or Rainflow.

Histogram / Rainflow

Channel nameHisto1

Histogram display update interval10 s

Histogram storage interval5 min

Parameter	Description		
	long name	short name	column ID
Histogram display update interval	Histogram display update interval		eHistogramUpdateInterval
Time interval for updating data which are derived from counting functions			
Histogram storage interval	Histogram storage interval		eHistogramSaveInterval
Time interval for saving data which are derived from counting functions			

8.3.6 Curve properties

Curve properties

Channel nameChannel_001

RangeAuto

Min-10

Max10

Color☐ auto

Parameter	Description														
	<i>long name</i>	<i>short name</i>	<i>column ID</i>												
Range	<i>y-axis range</i>	<i>Range</i>	<i>eCurveYAxisOption</i>												
Pre-defined setting for the range of display elements (Widgets). The setting is applied to the (display) range if: • "auto" is set in the curve window or • the "<i>Range</i>" for the other Widgets is set as "<i>From the variable</i>". The following possibilities are available: <table><tr><th>Selection</th><th>Function</th><th>Wirkt auf Widgets</th></tr><tr><td>Auto</td><td>Max and Min values match the measurement range set for the channel</td><td> • scale-Widgets </td></tr><tr><td>Input range</td><td>Max- and Min-values match the measurement range set for the channel</td><td> • scale-Widgets • curve window (rezoom) </td></tr><tr><td>User-defined (min, max)</td><td>Max- and Min-values can be set in the parameters further below</td><td> • scale-Widgets • curve window (rezoom) </td></tr></table>				Selection	Function	Wirkt auf Widgets	Auto	Max and Min values match the measurement range set for the channel	• scale-Widgets	Input range	Max- and Min-values match the measurement range set for the channel	• scale-Widgets • curve window (rezoom)	User-defined (min, max)	Max- and Min-values can be set in the parameters further below	• scale-Widgets • curve window (rezoom)
Selection	Function	Wirkt auf Widgets													
Auto	Max and Min values match the measurement range set for the channel	• scale-Widgets													
Input range	Max- and Min-values match the measurement range set for the channel	• scale-Widgets • curve window (rezoom)													
User-defined (min, max)	Max- and Min-values can be set in the parameters further below	• scale-Widgets • curve window (rezoom)													
Min	<i>y-axis min</i>	<i>Min</i>	<i>eCurveYAxisMin</i>												
If "<i>y-axis range</i>" is set to "<i>User-defined</i>": The transmitted Minimum-value for the display range.															
Max	<i>y-axis max</i>	<i>Max</i>	<i>eCurveYAxisMax</i>												
If "<i>Y-Axis option</i>" is set to "<i>User-defined</i>": The transmitted Maximum-value for the display range.															
Color	<i>Curve color</i>	<i>Color</i>	<i>eCurveColor</i>												
Pre-defined setting for the channel-color in the curve window.															

Setting the expected signal value range in addition to the measurement range

For some sensors, it is sometimes desirable not only to define the measurement range, for example ± 10 kN or $-270..500$ °C, but also an expected value range, e.g. $0..9$ kN or $20..80$ °C. The expected value range is usually only a subset of the measurement range. The intention would be to display only values within that range in order to be able to directly concentrate attention on the relevant portion.

Whenever values outside of this range are measured, they are to be considered overmodulation. This provides a quick indication that something is not right at the moment, even if there is no actual overmodulation from an electrical standpoint.

Specify whether the measurement range ("Auto"/"Input range") or other Min- and Max-values ("User-defined") transfer to the channel. The Widgets apply these presets if no other settings ("From Variable") have been made.

In many cases, no modification is necessary any longer, or the range can be governed by means of the parameters for all Widgets which display the channel.

Instead of that resulting range, it is also possible to specify a separate, different range for all Widgets.

Level indicators

"Current value"-column in the Setup, as well as the Widget "Level indicator"

The level indicator displays also use the ranged defined here. When values outside of this range are measured, the level indicators indicate overmodulation even when there is none from an electrical standpoint.

8.3.7 Additional parameters

Additional parameters are available, which are either already displayed in the channel table, or which you can add manually using the "Column chooser" ¹⁴⁸.

Category: Channel

Parameter	Description																														
	long name	short name	column ID																												
Connector	Connector		ePlugInName																												
The number returns a fixed identification of the variable. E.g. "[03] IN04". It is composed of: [i]: the module's address (mainly only for physical inputs and outputs) - xyz: abbreviation for the channel type (IN = analog channel, MIN: monitor: analog channel, ...) - j: consecutive numbering within the respective module or entire channel type, if no address available Examples: [03] IN04: analog input, fourth input at the address "03" V005: Virtual channel with the number 5 [00] DO01Bit05: Digital outputs (bit) fifth bit at Port: "01", at address: "00" [02] MCIN02: Monitor: incremental counter-inputs, second input at the address "02" PV007: Process vector variable with the number 7																															
Enumerated channel number	See "Channel number"																														
Channel number	Enumerated channel number	Channel number	eEnumeratedChannelNumber																												
Corresponds to the column "<i>Connector</i>", with the exception of the module address. The module address is not included in the display and so the count is performed across all modules, rather than each module separately. The column: "Connector" does not correspond to the consecutive numbering on the front panel for some devices (e.g. imc SPARTAN and imc CRC). Example: Above: "Connector"; below: "Enumerated channel number" <table><tr><td>[01] IN01</td><td>[01] IN02</td><td>[01] IN03</td><td>...</td><td>[01] IN07</td><td>[01] IN08</td><td>[02] IN01</td><td>[02] IN02</td><td>[02] IN03</td><td>...</td><td>[02] IN07</td><td>[02] IN08</td><td>[03] IN01</td><td>[03] IN02</td></tr><tr><td>IN001</td><td>IN002</td><td>IN003</td><td>...</td><td>IN007</td><td>IN008</td><td>IN009</td><td>IN010</td><td>IN011</td><td>....</td><td>IN015</td><td>IN016</td><td>IN017</td><td>IN018</td></tr></table>				[01] IN01	[01] IN02	[01] IN03	...	[01] IN07	[01] IN08	[02] IN01	[02] IN02	[02] IN03	...	[02] IN07	[02] IN08	[03] IN01	[03] IN02	IN001	IN002	IN003	...	IN007	IN008	IN009	IN010	IN011		IN015	IN016	IN017	IN018
[01] IN01	[01] IN02	[01] IN03	...	[01] IN07	[01] IN08	[02] IN01	[02] IN02	[02] IN03	...	[02] IN07	[02] IN08	[03] IN01	[03] IN02																		
IN001	IN002	IN003	...	IN007	IN008	IN009	IN010	IN011		IN015	IN016	IN017	IN018																		

Additional parameters are available, which are either already displayed in the channel table, or which you can add manually using the "[Column chooser](#)"¹⁴⁸.

Category: Other

Parameter	Description		
	long name	short name	column ID
Current value	Current value		CurrentValueDisplay
Shows the current reading of the following devices/channel types: • analog inputs, incremental counter inputs, field bus channels, FunctionSimulator channels, Application module channels, GPS channels • pv-variables, display variables • ethernet bits, virtual bits If no value is displayed, the measurement has not been prepared/started. For channels having an associated pv-variable (e.g. analog inputs or incremental counter inputs), the following applies: • The current value is displayed as soon as the action "<i>Prepare</i>" is performed (so for instance, also before and after the measurement). • If the value is displayed in gray font, channel settings have been changed since the "<i>Prepare</i>" procedure. The other channel types display the current reading as long as the channel is returning measurement data. If the measurement is not running, the last value is retained and displayed in gray font. This is to clearly indicate which value is still being updated and which might be obsolete. Level indicator for channels having a measurement range (except incremental counter inputs) Color coding indicates in which portion of the measurement range the value is located (green/red shading).			

8.4 Additional pages

By default, only a selection of the available Setup pages is displayed. It is possible to have additional pre-configured pages shown.

Displaying a page

To add a page to the user interface,

- open the context menu over a page-tab or over the empty space to the right of one.
- Click on the corresponding page entry in the list under "*Insert complete layout*".

In consequence, the page is inserted to the right of the selected locations.

 **Note**

Saving views

Changes to the view, such as: "Insert Setup page", must be saved with the view in order to be visible the next time the program is started.

See the manual "*imc STUDIO Monitor (general)*" > "[Views](#)"⁸⁴.

8.4.1 Summary

This page provides a quick overview of all active channels, the data storage status, and of the circular buffer time. Additionally, the page can be filled with any arbitrary choice of channel parameters, as accustomed.

Documentation

Devices

Analog channels

Digital channels

Summary

>

Channel name	Connector	Channel status	Current value	Save (device)	Save (PC)	Circular buffer time (curve window)	Channel type
DIN002	[01] DI02	Active		☐	☒	1 min	Digital inputs / outputs (ports)
Channel_001	[01] IN01	Active		☐	☒	1 min	Analog inputs
Channel_002	[01] IN02	Active		☐	☒	1 min	Analog inputs
Channel_003	[01] IN03	Active		☐	☒	1 min	Analog inputs
Channel_004	[01] IN04	Active		☐	☒	1 min	Analog inputs
Channel_005	[01] IN05	Active		☐	☒	1 min	Analog inputs
Channel_006	[01] IN06	Active		☐	☒	1 min	Analog inputs
Channel_007	[01] IN07	Active		☐	☒	1 min	Analog inputs
Channel_008	[01] IN08	Active		☐	☒	1 min	Analog inputs
Mon_Chanel_001	[01] MIN01	Active		☐	☐	1 h	Monitor: Analog inputs
Mon_Chanel_002	[01] MIN02	Active		☐	☐	1 h	Monitor: Analog inputs
Mon_Chanel_003	[01] MIN03	Active		☐	☐	1 h	Monitor: Analog inputs
Mon_Chanel_004	[01] MIN04	Active		☐	☐	1 h	Monitor: Analog inputs
Counter_001	[02] CIN01	Active		☐	☒	1 min	Incremental counter inputs
Counter_002	[02] CIN02	Active		☐	☒	1 min	Incremental counter inputs

Summary - all active channels

By means of the grouping box at the top left, it is possible to use Drag&Drop to apply grouping to individual columns. Thus for instance, it would be possible to group channels according to whether or not their data are to be saved.

Save (PC)				
Channel name	Connector	Channel status	Circular buffer time (curve window)	Channel type
▼ Save (PC): False (Count=4)				
Mon_Chanel_001	[01] MIN01	Active	1 h	Monitor: Analog inputs
Mon_Chanel_002	[01] MIN02	Active	1 h	Monitor: Analog inputs
Mon_Chanel_003	[01] MIN03	Active	1 h	Monitor: Analog inputs
Mon_Chanel_004	[01] MIN04	Active	1 h	Monitor: Analog inputs
▼ Save (PC): True (Count=11)				
DIN002	[01] DI02	Active	1 min	Digital inputs / outputs (ports)
Channel_001	[01] IN01	Active	1 min	Analog inputs
Channel_002	[01] IN02	Active	1 min	Analog inputs
Channel_003	[01] IN03	Active	1 min	Analog inputs
Channel_004	[01] IN04	Active	1 min	Analog inputs
Channel_005	[01] IN05	Active	1 min	Analog inputs
Channel_006	[01] IN06	Active	1 min	Analog inputs
Channel_007	[01] IN07	Active	1 min	Analog inputs
Channel_008	[01] IN08	Active	1 min	Analog inputs
Counter_001	[02] CIN01	Active	1 min	Incremental counter inputs
Counter_002	[02] CIN02	Active	1 min	Incremental counter inputs

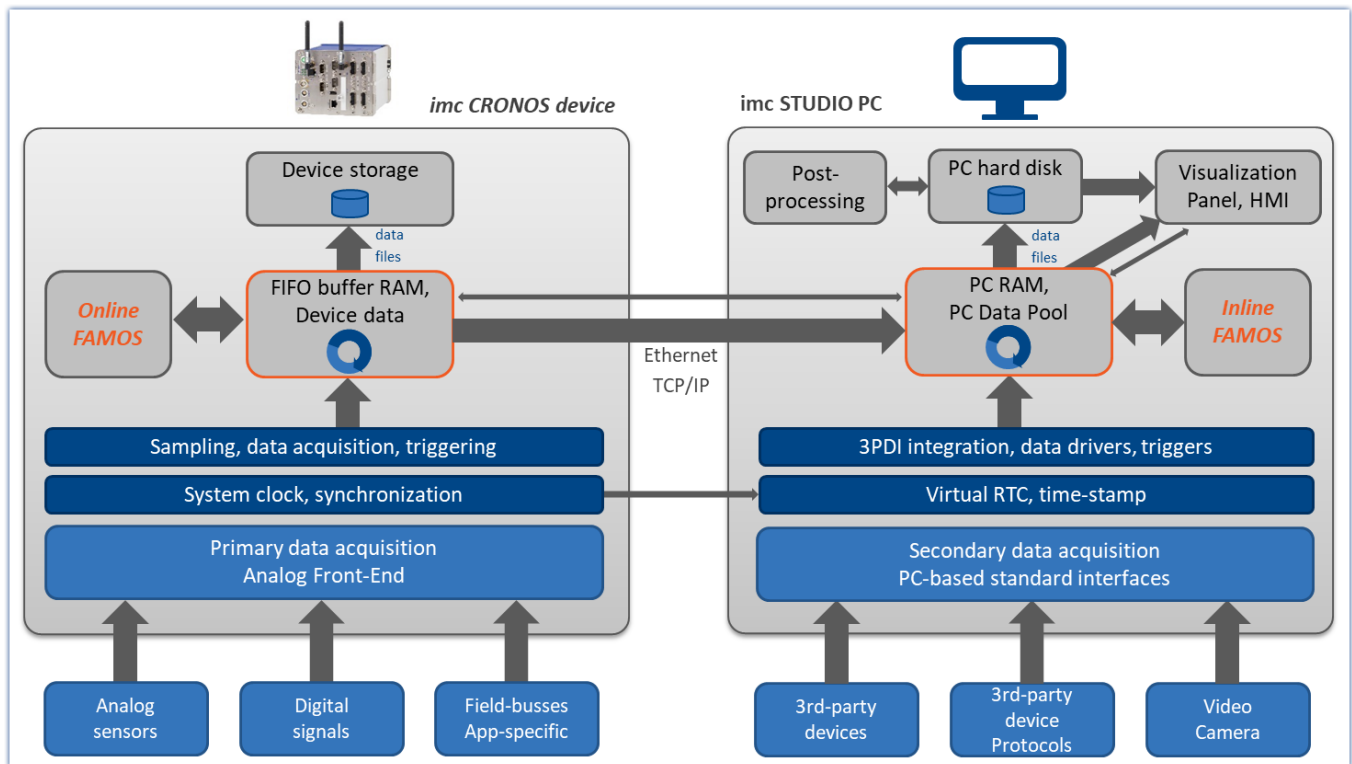
Summary - grouped according to data saving

8.5 Storage options and directory structure

This chapter describes the different options for saving measurement data with imc STUDIO Monitor.

The data stream

Once the measurement starts, the measurement device generates a data stream. The goal in every measurement is to process this data stream and to observe and or save it.



Simplified representation of the data stream - illustrated by the example of an imc CRONOS device

The data stream is first written to FIFO [RAM buffer memory](#) ^[205] in the device.

This RAM buffer is read by a signal processor ("DSP").

- If you use an imc Online FAMOS program in the experiment, the data stream is read and processed at this point of the data stream.
- For closed-loop control applications, it can make sense to interrupt the data stream at this point already (deselect "[Transfer to the PC](#)" ^[196]).

After this step, the data can either be saved by the device itself, or transmitted to the control-PC and saved if desired. Whether or not data are transferred to the PC and whether they can be saved can be set for each channel separately.



Note

Display without Data Storage

Data storage isn't necessary in such settings as system setup or trial readings.

Data saving		Description
Saving by the device ^[198]	By default, data saving is deactivated	Only configurable via imc STUDIO or imc WAVE.
Saving and displaying data on the PC ^[200]	By default, data saving is activated	Here, the data stream is transferred to the controlling PC via the connection (mostly LAN) ("Transfer to the PC" ^[196]). The data can be displayed using the Panel pages. Additionally, the transferred measurement data can be saved on the PC. • By default, the data for the display are saved on the PC in a circular buffer. By using the circular buffer memory, it is possible to temporarily store the data in the higher speed ("rapid access") working memory (RAM). • For data storage purposes, circular buffer memory is not used.

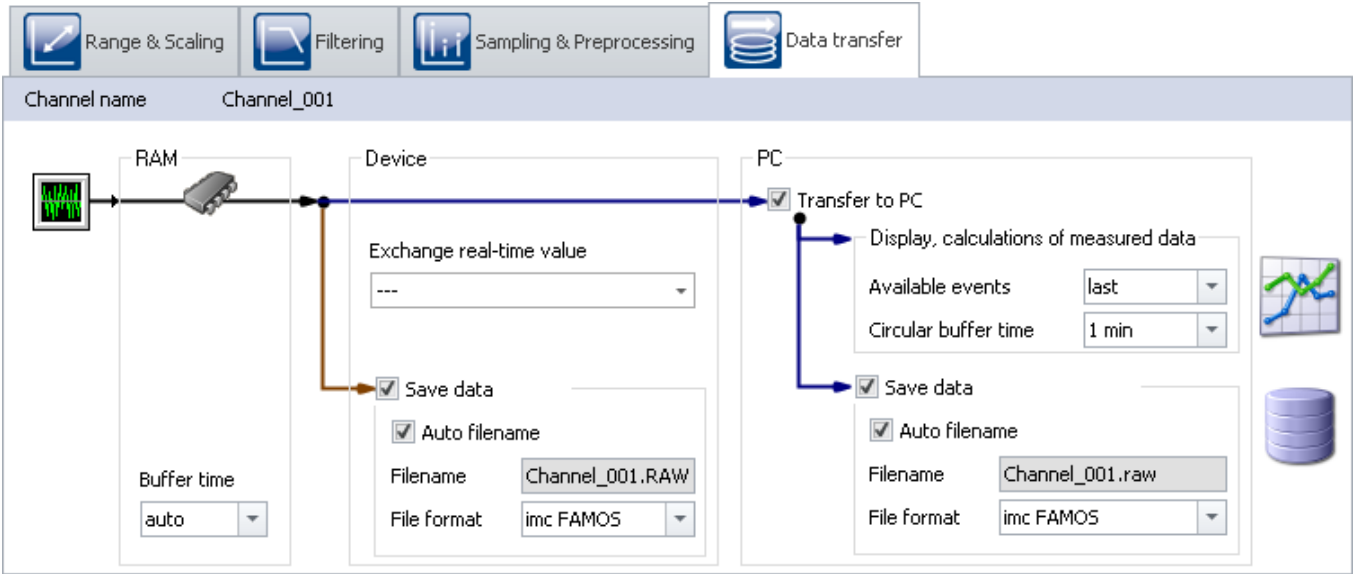
Data saving by the PC and by the device

There are a variety of data storage settings. Appearing below are links to descriptions of the most important parameters:

Channel-specific parameters	Device-parameters
File-Name ^[196] and File-Format ^[197]	Interval storage ^[202]
Circular buffer memory ^[197]	Save trigger events in individual files ^[212]
Available events ^[197]	Storage location on the device ^[198]
Transfer to PC ^[196]	

8.5.1 Storage options - Settings for the channels

Setting the storage options for the channels is performed on the Setup page: "Analog/digital channels" in the dialog: "Data transfer".



Accumulating measured data are temporarily stored in the device until they are transferred to the PC for purposes of display, analysis or storage.

imc STUDIO Monitor offers the following options located under the dialog: "Data transfer":

- 1. Only online processing (no transfer to or display on the PC)
- 2. Saving on internal storage medium (device drive)
- 3. Display and calculation of the data on the PC (with or without Circular buffer¹)
- 4. Saving on the PC's hard drive

Items 2 through 4 above can be combined in any way desired. All storage options can be set for each channel separately.

¹ Circular buffer: the data are saved on the PC's memory for the specified amount of time and then deleted (FIFO).

Parameter	Description
Transfer to PC	When the box is checked, the data are transmitted to the PC. This is a prerequisite for displaying , calculating and storing data on the PC. • Only online processing: When the box is not checked, the data are not transferred to the PC. The data are only available for calculations in imc Online FAMOS or for storage in the device. No display of the measured data in the curve window or in the Data Browser.
Save data	• Saving on the PC²⁰⁰: Saving of the measured data on the PC hard drive according to the folder structure²¹¹ set.
Auto filename	If the box is checkmarked, imc STUDIO Monitor will create a file name: • PC and Device: The channel name will be used as file name

Parameter	Description
Circular buffer time on the PC (for display/calculation)	Here you set how long the maximum time segment is. When using this option, the required memory is known from the beginning of the measurement. Deactivating circular buffering: If you enter "0", the circular buffer is deactivated and all data are saved.  • Recommendation: For the measured data (display and calculations), use an activated circular memory buffer. This transfers the data to the faster-working RAM memory. • The circular buffer duration set is also valid for segmented channels such as the FFT. After the time period set, the count of segments no longer increases, since the first segments are deleted instead.  See "Events and circular buffer"²⁰¹¹
Available events (Display)	With this setting, you determine whether all events (triggered events) can be displayed, or only the last one.  • all: As soon as the setting is set to "all", the channel is an event-based data set. E.g., it is possible in this way to display all events in one curve window. • last: Only the last event is available for display or calculations (recommended).  See "Events and circular buffer"²⁰¹¹
File format	Format of the saved measured data. By default, the measured data are saved in the file format imc FAMOS.

8.5.2 Storage in the device

Only configurable via imc STUDIO or imc WAVE.

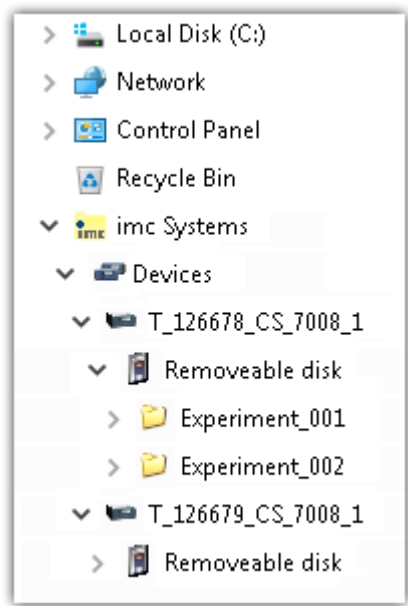
Data transfer - Access via Windows Explorer

For devices belonging to the [firmware group A](#)^[116] (imc DEVICES):

The menu item "*Data (Device)*" () causes the Windows-Explorer to start while indicating the device selected.

Ribbon	View
Extra > Data (Device) ()	Complete
Home > Data (Device) ()	Standard

Access via "*imc Systems*" - an Explorer shell extension



While installing the operating software, if the option "[Extension for Windows-Explorer](#)^[24]" is activated, you are able to copy, display or delete the files of measured data saved within the device (e.g. on the removable storage medium). The method of doing this is the familiar one under Windows.

This function is independent of the device software. As well, selection of the devices in the tree diagram is independent of the device list in the operating software.

In order to get access to your device's storage medium, you must first add it to the tree diagram (see "[imc Systems - Adding a device](#)^[227]"). Subsequently, you are able to navigate to the desired data in the storage medium and thus to work with them.

8.5.2.1 Supplemental files: DirClosed and imcSyslog

In addition to files of measured data, the measurement system also creates two system files.

DirClosed

On the device hard drive, the file "*DirClosed*" is created as soon as a measurement data folder is completely concluded. The file has no content. If this file exists, it is possible to access the measured data. For imc LINK or any personally created transfer programs, reliable copying and deleting are ensured.

imcSyslog

 **Note**

The following description applies to devices belonging to the [firmware group A](#)^[116] (imc DEVICES).

While a device is operating without being connected to any PC, the device is not able to report any errors. So if for instance a data overflow occurs during online processing or when saving the data, this information must be saved in the device.

Such information is saved in the file "*Syslog*", which is located in the respective measurement data folder. At this time, only important information regarding cases of data overflow are recorded. By this means, the measured data can be reconstructed except for the lost samples.

Name of the file

DeviceXXXXXX.syslog, "XXXXXX" standing for the device's serial number, e.g. *Device120345.syslog*.

Structure of the file when functioning normally

```
imcSyslog Version 1.0
imc DEVICES 2.13R8 (19.8.2020)
Nothing to report!
```

Structure of the file in case of data overflow

```
imcSyslog Version 1.0
2020-09-21 11:08:06.929190 M#:Data overflow on disk! F#:V006_G01.raw E#:1 FE#:0 O#:206 L#:1253
2020-09-21 11:08:08.147645 M#:Data overflow on disk! F#:V006_G01.raw E#:1 FE#:0 O#:1459 L#:1800
2020-09-21 11:08:11.029787 M#:Data overflow on disk! F#:V006_G01.raw E#:1 FE#:0 O#:3259 L#:1113
2020-09-21 11:08:16.071192 M#:Data overflow on disk! F#:V006_G01.raw E#:1 FE#:0 O#:6918 L#:1304
2020-09-21 11:08:22.906673 M#:Data overflow on disk! F#:V006_G01.raw E#:1 FE#:0 O#:10111 L#:1054
2020-09-21 11:08:24.630697 M#:Data overflow on disk! F#:V006_G01.raw E#:1 FE#:0 O#:11186 L#:1100
Log closed!
```

The structure is designed so that the information can be both read and also processed by a repair program.

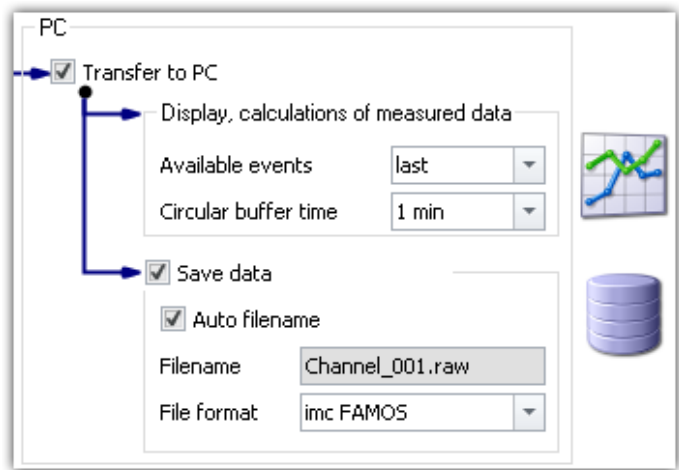
	2020-09-21 11:08:06.929190	M#:	Data overflow on disk!	F#:	V006_G01.raw	E#:	1	FE#:	0	O#:	206	L#:	1253
	1	2	3	4	5	6	7	8	9	10	11	12	13
	Description					Example							
1	Date and time of the event					e.g. 2020-09-21 11:08:06.929190							
2	Identifier for the message text					M#							
3	Report as an English-language text					e.g.: Data overflow on disk!							
4	Identifier of the filename					F#							
5	Name of the file in which the overflow occurred					e.g. V006_G01.raw							
6	Identifier of the absolute event number					E#							
7	Absolute number of the event in which the data overflow occurred					e.g. 1 for the first trigger release on this channel since the measurement start							
8	Identifier of the relative event number					FE#							
9	Relative number of the event, in reference to the events in the file affected					e.g. 0 for first event in this file; 1 for second event in the file...							
10	Identifier for the offset					O#							
11	Offset of the overflow within the file in 16-bit words					e.g.: 206. -1 -> The overflow is located between the file for the last interval and this one							
12	Identifier for the length of the data overflow					L#							
13	Length of the data overflow in 16-bit words					e.g.: 1253							

8.5.3 Transfer and storage on the PC

Setting the storage options for the channels is performed on the Setup page: *"Analog/digital channels"* in the dialog: *"Data transfer"*.

Transfer to PC

The measured data are transferred to the PC if the option *"Transfer to PC"* is activated. During measurement, you can also have the data displayed on the PC while being saved on the hard drive.



Display and calculations of measured data

The measured data available for display on Panel pages and for calculations.

Save data

To activate data storage on the PC, set a check in the box for *"Save data"*.

Storage location

The **measurement files** are saved in a defined folder structure **together** with the **experiment**. There are a variety of available options for modifying the [folder structure](#)^[211]. Additionally, it is possible to **personally adapt how and where data are saved** using the [options](#)^[213].

After conclusion of the measurement, display the data thus saved e.g on Panel pages, or with imc FAMOS.

Note

Make sure, that there is enough free memory on the selected drive for the expected measurement data.

Access via Windows Explorer

The menu item *"Data (PC)"* () causes the Windows-Explorer to start while indicating the memory location of the measured data.

Ribbon	View
Extra > Data (PC) ()	Complete
Home > Data (PC) ()	Standard

 **Reference**

See also

- Information on the options for display and storage of the measured data is presented in the channel: "[Storage Options - Settings for the Channels](#)".
- Information on display of the stored measurement data is presented in the chapter: "[Panel](#)".
- [Storage media in the device](#)

8.5.4 Events and circular buffer

Events

Events result from interruptions in the time plot of measured data, during which the measurement actually continues to run, but no measurement data are recorded. This can occur either intentionally, due to multiple triggering for example, or unintentionally due to a data overflow.

In order to be able to display multiple events together in one data set, a data set containing multiple Events is generated, a so-called "[Event-based data set](#)".

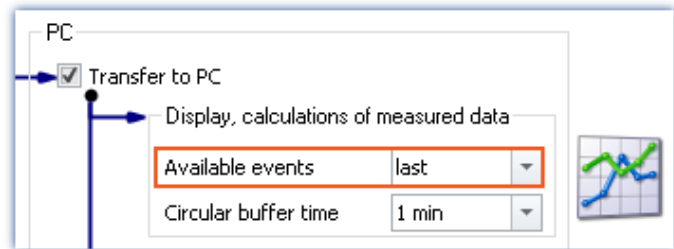
Events in data storage

All events are always saved.

- As soon as multiple events can occur (activated multi-triggering), the channel becomes an **event-based** data set.
- If multi-triggering is not activated or no triggering at all (e.g. connected with the BaseTrigger), then an event-based data set will usually not be generated (the exception is a data overflow).

Special case: For storage of the measured data on the PC, each event can be saved to a separate file. (see Device settings > "[Storage](#)" > "[Save trigger events in individual files](#)")

Events in visualization and for calculations



- It is only possible to set the amount if the respective channel is not set for circular buffer memory, which means it is set to "*Unlimited*".
- Note on analysis using imc FAMOS: If possible, use the measured data saved on the hard drive for analysis purposes.

Parameter: Available events

Parameter	Description
Available events (Display)	With this setting, you determine whether all events (triggered events) can be displayed, or only the last one. all: As soon as the setting is set to "all", the channel is an event-based data set. E.g., it is possible in this way to display all events in one curve window. last: Only the last event is available for display or calculations (recommended).

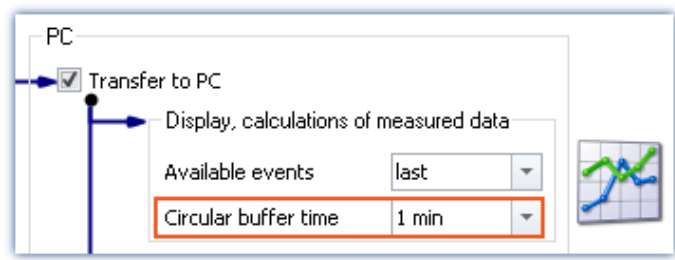
Circular buffer

Circular buffering in data storage

For data saving purposes, there is **no circular buffer**.

Circular buffering in visualization and for calculations

By default, the **data for the display** are saved on the PC in a **circular buffer**. By using the circular buffer memory, it is possible to temporarily store the data in the higher speed ("rapid access") working memory (RAM).



The screenshot shows a software window with a 'PC' label and a 'Transfer to PC' checkbox that is checked. Below this, there is a section for 'Display, calculations of measured data'. It contains two dropdown menus: 'Available events' set to 'last' and 'Circular buffer time' set to '1 min'. The 'Circular buffer time' dropdown is highlighted with a red rectangle. To the right of the dropdowns is a small line graph icon.

- Recommendation: For the measured data (display and calculations), use an activated circular memory buffer. This transfers the data to the faster-working RAM memory.
- The circular buffer duration set is also valid for segmented channels such as the FFT. After the time period set, the count of segments no longer increases, since the first segments are deleted instead.

Parameter: Circular buffer time

Parameter	Description
Circular buffer time on the PC (for display/calculation)	Here you set how long the maximum time segment is. When using this option, the required memory is known from the beginning of the measurement. Deactivating circular buffering: If you enter "0", the circular buffer is deactivated and all data are saved.

 **Warning**

Deactivated circular buffer

This setting is absolutely not suitable for long-term measurements

Recommendation: For the measured data (display and calculations), use an activated circular memory buffer. This transfers the data to the faster-working RAM memory.

8.5.5 Saving interval

Saving interval - What is it (for)?

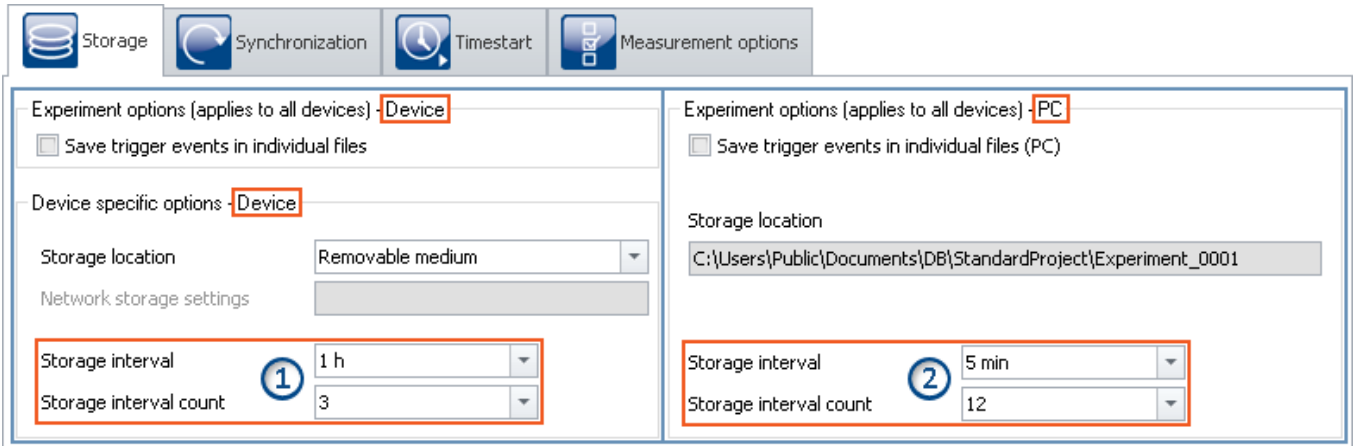
Upon the elapse of a certain time interval, a running data set is closed and a new file created.

- A number of saving intervals is specified. This number times the duration of the interval equals the maximum memory depth. When this number of intervals has been exceeded, the recordings of the oldest respective intervals are deleted.
- The setting is valid for all of a device's channels.

Suppose that there is a power outage during a running measurement; there is no guarantee that the files are closed properly. Therefore the last recorded experiment can not be evaluated. By setting a saving interval, you determine the maximum possible data loss due to a system failure.

Activating interval data saving

To activate interval data saving, open the dialog: "Storage" on the Setup page: "Devices".



Interval saving can be set for data storage on both the devices and on the PC separately.

- On the left side, you see the settings for the data storage on the devices: The interval saving can be set **for each device** separately ①.
- On the right side, you see the settings for storage on the PC: the storage interval set **applies to all the devices** ②.
- The interval storage setting **not for each channel separately!**

 **Note**

- Activating the interval storage does not activate the storage for each channel separately.
- [Activate additional data storage](#) ¹⁹⁶ on the PC or in the device for the channels desired!

Storage interval / Storage interval count

Parameter	Description
Storage interval / Storage interval count	With these two parameters, you can limit the amount of data and number of files. In this way, it is possible to prevent complete exhaustion of available memory even during long-term measurements. Storage interval: After this much time elapses, a new file is created for the duration of the next saving interval. Storage interval count: Save the last x intervals sets the desired number of intervals. Once this number is reached, the oldest interval is deleted. Multiplying the count and the interval determines which measurement duration is available after the measurement has been run. Example: Storage interval = 5 min, Storage interval count = 12. The measurement duration is set to 24 h. This ensures that after one day (24 h) at least the last 60 min of the measurement are available in intervals of a maximum of 5 min.

**Note****Amount of data saving intervals**

Once the measurement is concluded, you usually obtain **one data saving interval more** than your settings specified. This ensures that at minimum the desired amount of data are available at the end of the measurement.

- Example:

Measurement duration: 60 s; with **intervals** of 10 s; and an **interval count** of 3 -> result in 30 seconds of expected data.

- Measurement start: 00:05

- The intervals are allocated as follows:

00:05 to 00:10 <- deleted

00:10 to 00:20 <- deleted

00:20 to 00:30 <- deleted

00:30 to 00:40

00:40 to 00:50

00:50 to 01:00

01:00 to 01:05

In order for at least 30 seconds to be available after the measurement ends, the last 4 intervals must remain saved. In this example, that means 35 seconds of measured data. If the oldest interval were deleted, then there would only be 25 seconds.

- Incomplete intervals can be avoided by means of a **Timed start**. By this means, start measurement exactly at an **interval boundary**.

For technical reasons, with some data types, one extra interval is created. This mostly affects such data types as matrices, histograms or TimeStampASCII (TSA) data.

During the measurement the PC usually contains **two extra saving intervals**, since it is not yet possible to access the current interval. As a result, in this case at least the desired data are always available for display/analysis.

**Warning****Measured data will be deleted**

- When interval storage is active, all **folders are deleted except for the specified number** of intervals!
- If you **prepare or start** a measurement, **the number of existing measurement folders is reduced** to the number of storage intervals.
This also applies if the number of intervals is decreased.
- Only measurements which are not loaded are automatically deleted in this way. Any measurements which are loaded are ignored. Thus, any measurements which are currently being viewed or analyzed on the Panel are not automatically deleted.
In consequence, it may happen that more saved measurements remain in existence than were specified. The measurements will only be automatically deleted if the particular measurement has been closed.
- The measurements can be deleted manually at any time via the context menu of the Data Browser.



FAQ

Why is the first interval shorter?

Answer: To achieve a standardization of interval folders on the PC and on the device, the first interval is cut at the next interval.

Example:

- The measurement starts at 8:55:03, the saving interval has been set to 10 minutes.
- The first interval will be closed at the next whole-number 10 minute interval, that is 9:00:00. From here it will be in steps of 10 minutes: 13:10:00, 13:20:00 etc.

8.5.6 RAM time duration - Outage of the network connection

Loss of the network connection between the PC and the device often causes measured data to be lost.

**Note****Saving measured data in the device**

Remedy: Save an extra copy of the measured data in the device. In this way, the measured data can be compared and reconstructed.

The RAM buffer duration is a time interval reserved for each channel, for the duration of which the data are stored in the measurement device's RAM. The longer this time interval is, the less likely data overflow is to occur if the PC's connection to the measurement device is interrupted.

For example, if a faulty connection to the PC prevents the data from being transferred to the PC, data overflow will occur after elapse of the RAM buffer duration set. A long duration is also necessary if you intend to switch the internal data storage medium during the course of a measurement (Hot-Plug).

**Example**

Suppose the active channels have a RAM buffer duration of 10 s. If the PC doesn't retrieve any data from the Ethernet-interface for 10 s, the buffer overflows, i.e. data to be transferred to the PC are lost. The message "*Data overflow*" is posted.

Behavior upon outage of the network connection

At regular intervals, imc STUDIO Monitor attempts to restore the connection with the device. If this is not possible, a message is posted accordingly. Once the connection has been re-established, data acquisition on the individual channels resumes. This happens on each channel separately and can thus lead to differing results, for instance, if the RAM-buffer sizes are different.

Result	Possible causes
Data storage resumes seamlessly from the previously saved data. The data are retrieved from the RAM; there is no loss of data .	For example, if the RAM buffer is sufficiently large that the data can be retained in the device.
A new "event" is generated (event-based data sets). The RAM-buffer data are not retrieved. The measurement data acquired during interruption of the network connection are missing and are not saved. In accordance with the applicable settings, either all events or only the last event is displayed ^[197] (the option does not apply to data storage).	For instance, if a gap in the data is detected. A data overflow, because the RAM-buffer duration was exceeded.
A new measurement folder is created and has the extension ".002". The measurement data acquired during interruption of the network connection are missing and are not saved.	E.g if the saved data can not be opened or if the data type does not permit events.

 [Reference](#)

[See also](#)

- ["Behavior upon Data Overflow"](#)^[236]

8.5.6.1 RAM time duration for imc DEVICEcore devices

For devices belonging to the [firmware group B](#)^[116].

The RAM memory available is distributed among the active channels. Thus, in an emergency the maximum available memory is at the system's disposal.

How much memory space is available for the buffer depends on a variety of factors, such as the size of the imc Online FAMOS source texts and other internal files. Thus it is not possible to specify exactly.

The possible amount of RAM buffer per channel depends on the channel count, the respective sampling rates, the aggregate sampling rate and the data types, among other things.

 **Note**

[RAM-buffer time and synchronization](#)

RAM-buffer time settings have **no effect** on synchronization of imc DEVICEcore-devices. These settings only need to be considered when using [imc DEVICES-devices](#)^[207].

 [Reference](#)

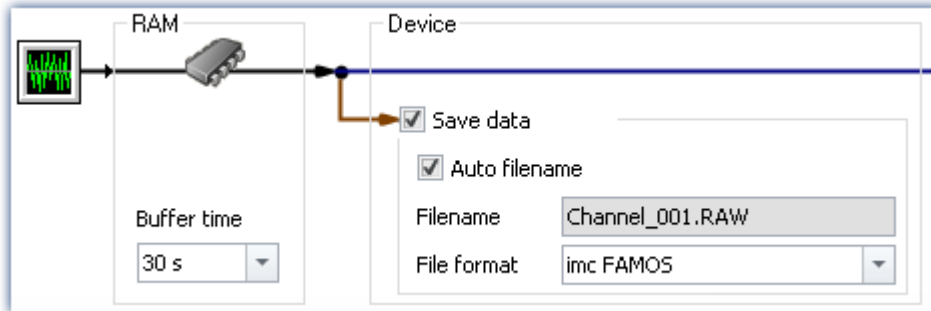
[Time-stamped data](#)

Also observe the notes on the RAM buffer duration for channels with time-stamped (e.g. some fieldbus channels), in the section: "[Time-stamped channels - Setting the RAM-buffer duration](#)"^[525].

8.5.6.2 RAM time duration for imc DEVICES devices

For devices belonging to the [firmware group A](#) ¹¹⁶.

Setting the RAM options for the channels is performed on the Setup page: "Analog/digital channels" in the dialog: "Data transfer".



A channel's RAM buffer volume is computed by multiplying sampling rate x buffer duration. For instance, a channel sampled at 5 kHz for the recommended RAM buffer duration requires 10 s $5000 \times 10 = 50,000$ Samples. When preparing for measurement, the system checks among other things whether there is enough free RAM available for the requirements of all channels.

If **error messages are posted during the preparation process**, then reduce the number of active channels or the buffer duration of individual channels, such as those which are not saved to the PC.



Reference

Time-stamped data

Also observe the notes on the RAM buffer duration for channels with time-stamped (e.g. some fieldbus channels), in the section: "[Time-stamped channels - Setting the RAM-buffer duration](#)" ⁵²⁵.

Automatic adjustment of the RAM-buffer

When **multiple high-sampling-rate channels** are used, the **RAM-buffer** in the device may eventually become **insufficient**. When the buffer size is fixed, the following error will often occur: "*Insufficient memory in device! Please observe the RAM buffer time of the channels and the number of triggers.*".

By means of the **setting "auto"**, the RAM buffer time is **dynamically adjusted** between 2 seconds and 10 seconds.

If one of the calculated **buffer times falls below 2 s**, the **error message** stated above also appears.

"auto" is the default setting. If the buffer is set to a **fixed value**, it is **no longer adjusted automatically**.

RAM-buffer time and synchronization

Absolutely synchronized recording of multiple channels, e.g. for computing power from a current signal and its associated voltage signal, is ensured in one of the three following ways:

1. Select for the channels to be mathematically combined the **same sampling interval** and the **same RAM buffer time**. Assign these channels to the same trigger.
2. Avoid multiple triggers and prepare the measurement before starting it, even if the configuration has not been changed. To force a repeat preparation, select the Ribbon *Setup-Control > Reconfigure*.
3. Select a **RAM buffer time** resulting in a **buffer volume less than 65,536**. Example: 20 kHz sampling rate: $65,536 / 20,000 = 3.3$ s. You can use the keyboard to enter this value in the input box next to the suggested values.

If none of these options was implemented, there may be a time offset between the channels. The cause for this is as follows:

There are $2^{16} = 65,536$ addresses available for a channel's RAM buffer. This means that for a RAM buffer volume of up to 65,536 samples, each sample can be addressed individually. If a greater value results from sampling rate x buffer time, the RAM buffer is organized in blocks.



Example

A sampling rate of 10 kHz x 10 s buffer time results in 100,000 > 65,536 samples. Here, the RAM buffer must be organized in 50,000 blocks of 2 samples apiece. This results in a tolerance in the addressing of magnitude block size-1, in this example, 1 sample.

In rare cases, this tolerance can cause a time offset between two channels. The possible time offset is then in the range between 0 and (block size-1) samples.

8.5.7 Overview storage and display options

Analog, digital, or other equidistantly sampled or time-stamped channels (e.g data-reduced channels)

0/1 = deactivated / activated | --- = Option is ignored

For display and calculation (with or without storage of measured data)

Multiple triggering?	Circular buffer	Circular buffer Time	Available events
No	unlimited		---
	activated	any (3)	---
Yes	unlimited		Last / All (2)
	activated	any (3, 4)	Last (1)

Storage of measured data on the PC and on the device

Multiple triggering?			Saved events	Storage interval	Storage interval count
No			---	any (3)	1..n or all
Yes			All	any (3)	1..n or all

- (1) For channels with multi-triggering, circular buffering pertains to the last trigger-event. The circular buffer memory does NOT extend across multiple trigger-events! Only the last event is displayed.
- (2) The default setting is: Display of the "last" event (also for channels with an unlimited trigger-event count). By this means, overflow of memory during long measurements is to be avoided. Regardless of this setting, for the display in the curve window, you can set the displayed events to "All/last/some".
- (3) The circular buffer length and the number of saving intervals can take 'any' value in the framework of the storage options for the PC used. The minimum storage size is 2 data samples.
- (4) The buffer memory size is limited by the channel's trigger length. Since the amount of data on a trigger is limited, more memory space would be useless.

FFT or similar virtual channels

0/1 = deactivated / activated | --- = Option is ignored

SL = Segment length (length of the FFT or similar)

For display and calculation (with or without storage of measured data)

Multiple triggering?	Circular buffer	Circular buffer Time	Available events
No	unlimited		---
	activated	n*SL	---
Yes	unlimited		Last / All (2)
	activated	n*SL	Last (1)

Storage of measured data on the PC and on the device

Multiple triggering?		Saved events	Storage interval	Storage interval count
No		---	any (3)	1..n or all
Yes		All	any (3)	1..n or all

- (1) For channels with multi-triggering, circular buffering pertains to the last trigger-event. The circular buffer memory does NOT extend across multiple trigger-events! Only the last event is displayed.
- (2) The default setting is: Display of the "last" event (also for channels with an unlimited trigger-event count). By this means, overflow of memory during long measurements is to be avoided. Regardless of this setting, for the display in the curve window, you can set the displayed events to "All/last/some".
- (3) The circular buffer length and the number of saving intervals can take 'any' value in the framework of the storage options for the PC used. The minimum storage size is 2 data samples.

Matrix or histograms

0/1 = deactivated / activated | --- = Option is ignored

For display and calculation (With or without storage of measured data)

Multiple triggering?	Circular buffer	Circular buffer Time	Available events
No	unlimited		---
	activated	---	---
Yes	unlimited		Last / All (2, 5)
	activated	---	Last (1, 5)

Storage of measured data on the PC and on the device

Multiple triggering?		Saved events	Storage interval	Storage interval count
No		---	any (3)	1..n or all
Yes		Alle (5)	any (3)	1..n or all

- (1) For channels with multi-triggering, circular buffering pertains to the last trigger-event. The circular buffer memory does NOT extend across multiple trigger-events! Only the last event is displayed.
- (2) The default setting is: Display of the "last" event (also for channels with an unlimited trigger-event count). By this means, overflow of memory during long measurements is to be avoided. Regardless of this setting, for the display in the curve window, you can set the displayed events to "All/last/some".
- (3) The circular buffer length and the number of saving intervals can take 'any' value in the framework of the storage options for the PC used. The minimum storage size is 2 data samples.
- (5) The calculations of matrices and histograms are based only on the data measured for the current trigger event.

8.5.8 Data storage and directory structure

By default, the **folders** are named along with their **date and time**.

Timestamp - date time
\2021-06-12 17-01-30
\Channel_001.raw
\Channel_002.raw
\Channel_003.raw
\---.raw
\2021-06-16 14-01-30
\---.raw
\Starttime
\Filename.raw

The folder name is formed from:

- the start time, or
- the new interval's start time, or
- the point in time when data saving resumes after an interruption.

By means of the Data Browser, you are able to change a measurement's name at a later time. The folder name automatically changes accordingly.

Modifying the data storage and folder structure

The folder structure can be modified using the following options:

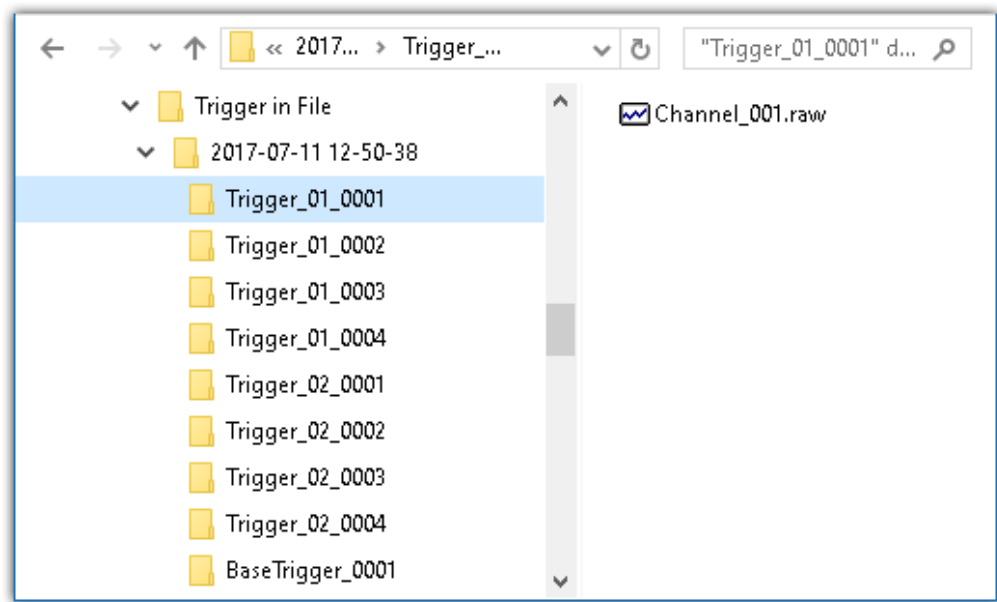
Options		Description
Save trigger events in individual files ²¹²⁾	Setup-page: " <i>Devices</i> ", on the dialog: " <i>Data transfer</i> "	Each trigger firing can be saved to its own separate file. For this purpose, additional subfolders are created for each trigger firing.
Customized storage location for measurement data ²¹³⁾	Options: " <i>General options</i> " > " <i>Storage location</i> "	On the PC-side, you can personally define your own storage location and folder structure . For the folder designations, you can also use metadata .

8.5.8.1 Save trigger events in individual files

The option: "[Save trigger events in individual files](#)"¹⁸⁰ lets you set the system's data saving behavior with triggered Events:

Not activated	Activated
Every triggered event is saved to the same file.	Each occurring trigger event is saved to a new file.
All events are always saved. As soon as multiple events can occur (activated multi-triggering), the channel becomes an event-based data set (see " Data transfer > Available events " ²⁰¹¹).	• For each trigger event, a separate folder is generated, in which the associated triggered channel data are saved. • Already while the measurement is still running, it is possible to access data on events which have already been completed.
\2021-06-12 17-01-30 \Channel_001.raw \Channel_002.raw \Channel_003.raw \2021-06-16 14-01-30 \---.raw \Starttime (measurement number) \Filename.raw	\2021-06-12 17-01-30 \BaseTrigger_0001\Channel_001.raw \Trigger_01_0001\Channel_002.raw \Trigger_01_0002\Channel_002.raw --- \Trigger_02_0001\Channel_003.raw \2021-06-16 14-01-30 \BaseTrigger_0001\---.raw \Starttime (measurement number) \TriggerName_TriggerNumber\Filename.raw

With this setting active, the folder structure for the measured data may appear as shown below:



 **Note**

Selection in Data Browser

In order to display the measured data associated with the events, add the column "[Event time](#)"⁵⁹⁸ manually in the Data Browser.

 **Warning**

- It is necessary to be very careful when using this option. Inconvenient selection of trigger conditions may cause the hard drive to quickly be filled with trigger folders.
- The time which the system needs to access data increases along with the number of folders.
 - For this reason, regarding data storage in the device, when **1000 trigger folders** accumulate, a **new measurement data folder** is created. This does not apply to storage on the PC.

8.5.8.2 Customized storage location for measurement data

When data storage is activated, the measurement results are normally saved to the database in the Experiment-folder.

According to the particular data storage settings, a [Directory Structure](#)^[211] is established in the Experiment-folder.

It is possible for you to define your own data storage structure. The settings for this purpose are available at options: "General options" > "[Storage location measurement data](#)"^[62].

Even if the measurement folder is changed within an experiment (e.g. by means of placeholders), all measurements saved are listed in the Data Browser. On the hard drive, the measurements are located at a variety of positions. In the Data Browser, they are all displayed together.

 **Warning**

- When you change these settings, some data storage settings may no longer be used. The settings can be used in conjunction with placeholders, but don't necessarily need to be.
- You define the structure's appearance. This may mean that the measurement results are always saved to the same folder. In order to avoid data being overwritten, a post-fix is automatically appended.
- Ensure that you possess reading and writing privileges for the data storage drive.

Definition of terms

Term	Description
Base path for measurement data storage	The base directory path for storage of measured data represents the default filepath for experiments. In this folder, all measurements are saved (according to the structure specified in "Folder naming").
Folder naming for measurement data storage	This specifies the structure below the base path. For instance, folders are set up here according to the settings for the directory structure ^[211] .
Preview	The preview provides an example of where the measurement would be saved according to the current settings. Be aware that the preview is for demonstration purposes only and may not reflect certain settings for the Verzeichnisstruktur.

Custom base path for measurement data storage

Default setting: `<EXPERIMENT.PATH>`

- Returns the Experiment-path. All measurement results are saved with the experiment.

If you specify a different folder, all measurement results are saved to this new folder and no longer according to the Experiment-path.

Examples - Custom base path



Example 1

`C:\Measurement\<EXPERIMENT.NAME>\Results`

- Measurement results for the experiment: *Experiment_0001* are saved as follows, for example:

`C:\Measurement\Experiment_0001\Results\2013-01-01 08-00-00`

`C:\Measurement\Experiment_0001\Results\2013-01-01 09-00-00`

`C:\Measurement\Experiment_0001\Results\2013-01-01 10-00-00`



Example 2

`C:\Measurement\<EXPERIMENT.NAME>`

- Measurement results for the experiment: *Experiment_0002* are saved as follows, for example:

`C:\Measurement\Experiment_0002\2013-01-02 08-00-00`

`C:\Measurement\Experiment_0002\2013-01-02 09-00-00`

`C:\Measurement\Experiment_0002\2013-01-02 10-00-00`



Example 3

`<EXPERIMENT.PATH>\Results`

In a case where the database was moved to the following path:

`c:\DB\`

- Measurement results for the experiment: *Experiment_0003* are saved as follows, for example:

`C:\DB\StandardProject\Experiment_0003\Results\2022-01-03 08-00-00`

`C:\DB\StandardProject\Experiment_0003\Results\2022-01-03 09-00-00`

`C:\DB\StandardProject\Experiment_0003\Results\2022-01-03 10-00-00`

Custom folder naming for measurement data storage

Default setting: `\<STORAGE.FOLDERNAME>\`

- Returns the experiment's data saving settings

If you specify a different folder naming, all measurement results will be save according to the specified folder structure.

Useful placeholders for the folder naming:

- `<STORAGE.FOLDERNAME>`
Generates a folder from the data storage settings (e.g. *2022-01-01 08-00-00*)
The result can change during a measurement, for instance when interval saving is activated. This placeholders ensures that each measurement result is assigned its own folder.
- `<STORAGE.MEASUREMENT>`
Returns the data and time of the measurement start (e.g. *2022-01-01 08-00-00*). The result remains unchanged until the end of the measurement. By means of this placeholder, it is possible for instance to assign each measurement its own permanent folder.
- `<VARS.VALUE>`
Returns a variable's value. The placeholder may serve as its own counter or as a metadatum representing the module serial number for the path.
- `<SETUP.SQL>`
Returns the value of a cell in a Setup-page. This could be uses as, for instance, a metadatum representing the module serial number for the path.

Examples - Custom folder naming

In all examples, no custom folder naming for measurement data storage is used. This means that the Experiment-folder is used.

Additionally, the database for all examples was moved to the following path: `c:\DB\`



Example 1

`\Results\<STORAGE.FOLDERNAME>\`

- Measurement results for the experiment: *Experiment_0004* are saved as follows, for example:
`C:\DB\StandardProject\Experiment_0004\Results\2022-01-01 08-00-00`
`C:\DB\StandardProject\Experiment_0004\Results\2022-01-01 09-00-00`
`C:\DB\StandardProject\Experiment_0004\Results\2022-01-01 10-00-00`
- In the Data Browser, the results are displayed as follows:
Results.2022-01-01 08-00-00
Results.2022-01-01 09-00-00
Results.2022-01-01 10-00-00



Example 2

`<STORAGE.MEASUREMENT>\<STORAGE.FOLDERNAME>\`

Interval saving (1 min) is activated. After every three intervals the measurement is stopped.

- Measurement results for the experiment: *Experiment_0005* are saved as follows, for example:
`C:\DB\StandardProject\Experiment_0005\2022-01-02 08-00-00\2022-01-02 08-00-00`
`C:\DB\StandardProject\Experiment_0005\2022-01-02 08-00-00\2022-01-02 08-01-00`
`C:\DB\StandardProject\Experiment_0005\2022-01-02 08-00-00\2022-01-02 08-02-00`
`C:\DB\StandardProject\Experiment_0005\2022-01-05 08-00-00\2022-01-05 08-00-00`
`C:\DB\StandardProject\Experiment_0005\2022-01-05 08-00-00\2022-01-05 08-01-00`
`C:\DB\StandardProject\Experiment_0005\2022-01-05 08-00-00\2022-01-05 08-02-00`
`C:\DB\StandardProject\Experiment_0005\2022-01-10 08-00-00\2022-01-10 08-00-00`
`C:\DB\StandardProject\Experiment_0005\2022-01-10 08-00-00\2022-01-10 08-01-00`
`C:\DB\StandardProject\Experiment_0005\2022-01-10 08-00-00\2022-01-10 08-02-00`
- In the Data Browser the results are displayed as follows:
2022-01-02 08-00-00.2022-01-02 08-00-00
2022-01-02 08-00-00.2022-01-02 08-01-00
2022-01-02 08-00-00.2022-01-02 08-02-00
2022-01-05 08-00-00.2022-01-05 08-00-00
2022-01-05 08-00-00.2022-01-05 08-01-00
2022-01-05 08-00-00.2022-01-05 08-02-00
2022-01-10 08-00-00.2022-01-10 08-00-00
2022-01-10 08-00-00.2022-01-10 08-01-00
2022-01-10 08-00-00.2022-01-10 08-02-00



Example 3

```
\Object <SETUP.SQL("SELECT TestPartNo FROM Project")>\Measurement
<VARS["Measurement_No"].VALUE("0")>\
```

The *SETUP*-placeholder returns the content of the column *Test part number*.

The *VARS*-placeholder returns the content of the variable *Measurement_No*.

Attention: The user must ensure that the Setup column is filled and that the variable has a value, which is raised automatically once per measurement.

- Measurement results for the experiment: *Experiment_0006* are saved as follows, for example:

```
C:\DB\StandardProject\Experiment_0006\Object 12\Measurement 1
C:\DB\StandardProject\Experiment_0006\Object 12\Measurement 2
C:\DB\StandardProject\Experiment_0006\Object 12\Measurement 3
C:\DB\StandardProject\Experiment_0006\Object 13\Measurement 1
C:\DB\StandardProject\Experiment_0006\Object 14\Measurement 1
```

- In the Data Browser, the results are displayed as follows:

```
Object 12.Measurement 1
Object 12.Measurement 2
Object 12.Measurement 3
Object 13.Measurement 1
Object 14.Measurement 1
```



Note

Point in time for determining target folder

The target folder is only determined once data which are to be saved arrive at the PC. Up until this point in time, it is possible to control the folder path. E.g. by means of the Setup pages which are called up before the measurement. When columns of this Setup page are used to set the path (via `<SETUP.SQL>`), the new values are observed.

Using the menu action: "*Suspend/Resume data storage*", data storage is interrupted/resumed. Upon the start of the resumed storage, the target folder is also determined again from the beginning.

This means you can change the target folder during the measurement.



FAQ

Why not use a `<SYSTEM.*>` placeholder?

Answer: The `<STORAGE.*>`-placeholders should be used here instead of the `<SYSTEM.*>`-placeholder. The `<STORAGE.*>`-placeholders are resolved once and then apply to all devices. The `<SYSTEM.*>`-placeholders are resolved anew each time. This means that they are resolved again for each device and in consequence that may lead to a different folder for each device.

The results of previous measurements are deleted although their origin is a different experiment

Answer: If you have activated interval-saving, it doesn't matter where the measurement results located in the target folder come from.

If the specified amount has been reached, older measurement results are deleted.

This can also happen when different PCs use the same measurement data folder and the same experiment name.

8.5.9 Controlling data storage

8.5.9.1 Targeted data saving, or saving subsequent to measurement

In addition to the optional automatic data saving, you can also perform targeted saving of the data currently in the PC (Current measurement) either during or subsequent to the measurement.

Targeted saving of the current measurement is only possible until a new measurement has been prepared or started!

In order to save the measured data in this way, select the ribbon item "*Save current data / Export current data*".

Ribbon	View
Project > Save current Data ()	Complete
Home > Save current Data ()	Standard

Ribbon	View
Project > Export current Data ()	Complete

Note

No single values can be saved in this way. These include, for instance, Display variables.

Be aware that with this function, only such variables are saved, which are found in the Data Browser under **Current measurement**. For these variables, a **circular buffer memory** is usually activated, so that not all results since the measurement start are available.

For device-variables, therefore, the [Data Transfer Settings](#)¹⁸⁸ are applicable. The storage type: "*Display, calculation of measured data*" is used.

Save current data

Using "*Save current data*", the measured data are saved in the database. If additionally the saving of measured data was activated, the results of data saving and of targeted saving are located in parallel next to each other.

When the button is clicked, a dialog appears in which the folder name can be entered. The folder is created and whatever measurement results are already available at the time are saved.

The **file format** used for saving is ".dat". Conversion of the measured data to other formats is possible by means of "*Export current data*" or by means of the imc Format Converter.

Saved measurement data are secure

Any measured data which are already saved cannot be overwritten by the menu action "*Save current data*". If a repeat attempt is made to perform "Save" to an existing file, the system refuses to perform this action. It is also not possible to perform in conjunction with measured data belonging to a continually saved measurement.

Export current data

When the button is clicked, a folder selection dialog appears, in which the target folder can be selected. The measurement results available up to that point in time are saved there.

The files are exported in the configured file formats: "*Options*" > "*Save current data*" > "[Variable export](#)"⁶⁶.

Options

In the [Options](#)⁶⁶, it is possible to configure how the data saving works.

- **Folder name / Path**
Here, a folder can be specified or a path entered, in which to save the variables. It is a preferable choice to use placeholders such as `SYSTEM.TIME`, so that the path can vary. Otherwise, the results will always be overwritten. In this way, you can also integrate the metadata into the folder path.
Path: You can pre-define the data storage path in which the data storage dialog is to launch. If the folder doesn't exist, it is created temporarily. But if you don't save the files in it, the folder is deleted once the dialog is closed. However this only happens when it had been created by this method and if it is really empty.
- **Show dialog**
The dialog for entering the folder name / path can be deactivated. The condition for this is: the option "Folder name / Path" has been filled in.
- **Save to one file per file format (Save all in a single file)**
If this option is activated, all variables are saved to a single (multi-channel) file. The file name can be specified. The file format for the file does not need to be specified. However, if no file name is specified, all variables are automatically saved to the file `data.dat`. Furthermore, for each file format a separate file is created.
If the option is not activated, each variable is saved in a separate file.
- **Overwrite existing files without confirmation prompt**
(For export only) - If this option is activated, files having the same name in the destination folder will be overwritten without a confirmation prompt. This option is useful for automated routines.
- **Mapping instruction**
If desired, you can provide you own personal mapping instruction according to which to determine the channel type selection.

8.5.9.2 Suspend and resume data storage

If the [saving of measured data is activated](#)¹⁸⁸, then you are able to govern the data storage **during the running measurement**. You can **interrupt (suspend) data storage** and **resume it again** at a later time.

Ribbon	View
Home > Suspend data storage (🟡🟡)	all
Setup-Control > Suspend data storage (🟡🟡)	Complete
Home > Resume data storage (🟡🔴)	all
Setup-Control > Resume data storage (🟡🔴)	Complete

Effect

If you **suspend data storage**, the **files are closed** and appear in the Data Browser (as if the measurement had been concluded). Any **more data** which arrive or are calculated subsequently **are not saved**.

When you **resume data storage**, then **new files are created** in a **new measurement folder** (as if the measurement had just been started). Any data arriving before "resumption of data saving" are not saved.

This function is designed for **interrupting data saving for longer periods of time**. Please do **not use it to make "cuts" of data records** (concluding one measurement folder and immediately beginning a new one).

 Note

- Data storage **can only be controlled** if it was previously **activated**.
- This **suspension/resumption** of data storage **is not synchronized** across the devices! Thus, the respective **data excerpt boundary points** of different devices/channels **may differ by a few measurement points**.

Scope

These actions **apply to all devices and to the PC**. No separation is possible. Any additional components which return data to be saved (such as 3rd-party devices) are also affected.

 Note

Video-files are not affected

The **saving of Video files** is **not affected** by this function. Video files are always recorded.

 Reference

Data storage state after preparation

Using the option "Data storage state after preparation", you can control the function's state. E.g. it is possible to make data storage generally active following "Preparation".

See the option: "Setup" > "General" > "[Data storage state after preparation](#)"  65".

8.5.10 Information and tips

8.5.10.1 The last measurement

In many cases, various reports are composed and analysis procedures as well as other actions are performed based on the data from the last measurement conducted. Many of these actions are performed automatically. In order for this to be possible, these actions must be able to recognize the "Last measurement".

Toward this end, there are various approaches.

1. You can select the measurement in the Data Browser, which assigns a measurement number to it (this happens automatically). There are many places in which you can access a measurement by this measurement number. This makes automated accessing of all measurements possible, including of the last measurement.
2. In various commands and actions, you are able to select whether you wish to access the "Last measurement". Internally, a record is stored of which measurement was last (recommended).

What data becomes the "last measurement"?

- A measurement folder closed by, for instance:
 - conclusion of a measurement
 - conclusion of an interval
 - saving of measurement data concluded due to interruption.
- Data saving by means of the menu item "*Save current Data*": The system considers this a measurement and it can become the "Last Measurement".

Notes

Each measurement folder represents a measurement

The last measurement is always only "**one**" folder. Thus for example, when interval data saving is activated, each interval is a separate measurement. It is always the last interval concluded which is considered the last measurement.

How long is the info stored?

The path to the measurement remains known to the system even when the **experiment is re-loaded**. When the **last measurement is deleted**, the measurement performed right before it becomes the "Last Measurement".

There are a variety of ways to access the last measurement:

- [Placeholders](#)¹⁰³: e.g. <MEASUREMENTS["MEASUREMENT#LAST"].PATH>
- "[imc FAMOS-command](#)⁹⁵⁵" (data source)
- "[Set Measurement Number](#)⁹⁷³"-command

Special cases:

- Measurement data folders created by different methods: In the Data Browser, only such folders are considered the "Last Measurement" if they were genuinely created by running a measurement. Whenever folders are created in a different way, e.g. imc FAMOS-events or by means of Scripting, etc., they are not treated as the "Last Measurement".
- Subsequent adding of files: When imc FAMOS-results are saved to an already existing measurement, this does not count as a "new" "Last Measurement".

8.5.10.2 Incomplete measurement files

Measurement files being written during the measurement process are not complete and can neither be deleted or copied. If either of the options "*Saving interval*" or "*Save trigger events in individual files*" is used, it is possible to access the data even during a running measurement. However, this only applies to data from completed events. The current interval or trigger event is subject to the same limitation.

For storage in device: Once all files have been concluded, the system creates an additional file "[DirClosed](#)¹⁹⁸". As soon as it is present, you have reliable access to the measured data.

8.6 Storage media in the device

This section describes how to handle the storage media of the imc measurement devices and how to use them with imc STUDIO Monitor.

For saving measured data, some devices support an internal storage medium. Depending on the device type, the storage medium could be, for example, a memory card or a hard drive (see "[Device overview](#)"¹¹⁶). Card-Bus interface are not recommended.

The storage media are exclusively for data acquisition with imc STUDIO Monitor.

Storage media with verified performance can be purchased as accessories from imc. Hard drives are ordered with the device and can only be installed subsequently by imc.

 **Note**

Manufacturer and Age of the storage medium

- imc has no way to affect the quality of the removable storage media provided by the various manufacturers.
- Storage media which come with newly purchased devices have been inspected in the framework of quality assurance and have passed the relevant tests.
- We expressly declare that the use of removable storage media is at the user's own risk.
- imc and its resellers are only liable within the framework of the guarantee and only to the extent of providing a substitute.
- imc expressly declines any liability for any damages resulting from loss of data.

For devices of firmware groups A and B (see "[Device overview](#)"¹¹⁶), a distinction is made between how the storage media are to be handled.

- [For devices of the firmware group B \(imc DEVICEcore\)](#)²²¹
- [For devices of the firmware group A \(imc DEVICES\)](#)²²⁴

8.6.1 For devices of the firmware group B (imc DEVICEcore)

Swapping the storage medium

The microSD card may only be changed when the **device is switched off**.

8.6.1.1 Storage media

Storage media	Description
Internal hard disk	Applicable to devices having a hard drive (see " Device overview " ¹¹⁶). • The hard drive has been formatted with the file system "Ext2". Re-formatting is not possible. • The hard drive is permanently installed. • The specified operating temperature range of the media is relevant.
microSD	Concerning devices with microSD slot (see " Device overview " ¹¹⁶). • Only microSD memory cards verified by imc should be used, as these have been specially qualified by us for high data rate of 5 MS/s. • The specified operating temperature range of the media is relevant.

8.6.1.2 Data transfer

The storage medium can be accessed **directly via Windows Explorer**. Alternatively, the storage medium can be inserted into a **card reader** on the PC (suitable for large amounts of data due to faster transfer).

 **Warnung**

Do **not use force** to insert or remove the device storage medium.

Access via Windows Explorer

The menu item "*Data (Device)*" () causes the Windows-Explorer to start while indicating the device selected.

Ribbon	View
Extra > Data (Device) ()	Complete
Home > Data (Device) ()	Standard

Using this menu item, a separate Windows-Explorer is opened for each device, providing direct access to the internal memory.

For the purpose of accessing the tool, it is necessary to log in. The factory default settings are:

- User: "*imc*"
- Password: device serial number

 **Note**

Manual entry

The device is called by reference to either its IP address or its internal name. The device's name is constructed as follows: "*imcDev*" + SerialNumber; e.g.
\\imcDev4120110
\\10.0.12.70

8.6.1.3 Memory cards - File system and formatting

Memory cards (microSD) with the file system FAT32 are supported. It is recommended that a memory card be [formatted](#)²²³ before use.

 **Note**

Routine formatting protects the memory card

Routine formatting is recommended
Take every opportunity to format the storage medium. **Recommendation:** at least every **six months**.
In this way, any **damaged storage medium** can be detected and repaired if possible. A damaged file system may cause **data loss**. Or the **measurement system may fail to start** correctly.
In order to avoid data loss, any data still needed should first be saved!

Using a data storage medium in different devices
There are no known limitations. But it is recommended to always format the medium whenever transferring it in order to avoid data loss.

 **Notes**

General restrictions applicable to file systems

Please observe the general restrictions regarding the respective file systems.

8.6.1.3.1 Formatting



Warning

Please back up the data first

Formatting causes all data on the storage medium to be deleted. Before performing the formatting, ensure that all data have been saved on a different storage medium.

The memory card (microSD) **can not be formatted in the device**.

For this purpose, please use an appropriate tool. Such a tool is provided on the installation medium for imc STUDIO Monitor in the folder "Tools\RemoveableDriveFormatter":

- "rufus-4.1p.exe"

This program can be used for the purpose of formatting the data volume.

Drive Properties

Device
NO_LABEL (E:) [256 GB]

Boot selection
Non bootable ☒ SELECT

Partition scheme
MBR

Target system
BIOS or UEFI

▼ Show advanced drive properties

Format Options

Volume label
256 GB

File system
Large FAT32

Cluster size
16 kilobytes

▼ Show advanced format options

Status

READY

START CLOSE

Settings required for formatting by means of Rufus 4.1



Note

Only **one(!)** partition may be created. Multiple partitions may cause the measuring device not to recognize the storage medium.

Make the **following settings** for the memory card:

- Boot selection: "Not bootable"
- Partition scheme: "MBR"
- File system: "Large FAT32"
- Cluster size: "16 kilobytes"

8.6.1.4 Known issues and limitations

Known issues and limitations	Description
File system becoming gradually slower	As the count of folders increases, so does the time required by the system to access the data. In consequence, the data saving procedure becomes slower and data loss becomes possible. For this reason, creating more than 1000 folders is to be avoided.
Errors in accessing the storage medium	Errors can have the following causes, among others: • The data rate is too high, the storage medium can't keep up and data overflow results. • The storage medium is full. If the PC is connected to the measuring device, imc STUDIO Monitor documents the error with a message in the Logbook.

8.6.2 For devices of the firmware group A (imc DEVICES)

Swapping the storage medium

Pressing the button signalizes to the system that you intend to remove the storage medium. Once this is done, the device stops access to the storage medium. If you were to remove the drive without prior announcement, it could produce defective clusters. If the storage medium is removed while a measurement is in progress, the data records are not completed. Therefore, always proceed as follows when swapping the storage medium:

1. **Important!** Before removing the storage medium from the measurement device, first announce the procedure to the system by pushing the button, in order to **avoid damage** to the storage medium.
2. Once the LED blinks, remove the storage medium.
3. Insert the new storage medium. Devices indicate by a short flash that the new drive has been successfully recognized.

Hot-Plug (exchanging the storage medium during a measurement)

It's possible to exchange the storage medium during a running measurement. This makes it possible to carry out a measurement without a PC practically without any limitations. It is only necessary to check the amount of memory available using imc Online FAMOS. To do this, use the function `DiskFreeSpace` belonging to the group "System". You can set an LED, for instance, or a digital output or a beeper to be activated when less than the minimum amount remains. One convenient solution would be to have a readout of the remaining space outputted by a display variable, which would indicate by a display on the device how the remaining memory decreases.

While swapping the storage medium during a running measurement, the data are stored in the measurement device's internal memory. If you complete the process within the specified RAM buffer duration, this is certain to work without any loss of data (see section "[RAM buffer time](#)"²⁰⁵). Note that not only the time for the swap must be buffered, but that the buffered data must also be transferred to the new disk once the swap has been completed.

Swapping the storage medium

1. **Important!** Before removing the storage medium from the measurement device, first announce the procedure to the system by pushing the button, in order to **avoid data loss and damage** to the storage medium. The LED will **shine continuously** in green.
2. Once the device is ready for removal of the storage medium, the LED **blinks**.
3. Remove the already full storage medium.
4. No announcement is necessary for inserting a storage medium.

8.6.2.1 Storage media

Storage media	Description
CF Cards (Compact Flash)	For devices of group A4 and A5 ^[116]: The device group exclusively uses CF cards for storage medium.
USB Storage Medium	Concerning devices with USB (see "Device overview" ^[116]). Memory sticks or external hard drives can be connected at this terminal. • Do not use multiple storage media at the same time! Devices belonging to group A6 ^[116] have two USB terminals and a slot for the ExpressCard. However, the device can use only one data storage medium. The system determines which one it is upon activation, and there is no fixed order of precedence among the media. For this reason, delete any which you do not wish to use for the measurement before switching the device on. • With USB, the Hot-Plug ^[224] functionality is available. Make sure that sufficient time is available for swapping the storage medium. How much time is required for de-registering and re-registering with the system depends on the particular storage medium and on the number of channels. As an orientation value, we recommend at least 30 s, even for simple configurations!  Do not use USB hard drives with external power supply Please do not use any USB hard drive which has an external power supply. Such a drive may not be connected at the imc USB-port. Otherwise, when the measurement device is powered down, the imc USB-port's current limiting mechanism may be destroyed.
ExpressCard	Applicable to devices having ExpressCard slot (see "Device overview" ^[116]). • Remove any connected USB storage medium! Devices belonging to group A6 ^[116] have two USB terminals and a slot for the ExpressCard. However, the device can use only one data storage medium. The system determines which one it is upon activation, and there is no fixed order of precedence among the media. For this reason, delete any which you do not wish to use for the measurement before switching the device on. • With ExpressCards, the Hot-Plug ^[224] functionality is available.
CFAST	Concerning devices with CFAST slot (see "Device overview" ^[116]). • Remove a connected USB storage medium! The device can use only one data storage medium. The system determines which one it is upon activation, and there is no fixed order of precedence among the media. For this reason, delete any which you do not wish to use for the measurement before switching the device on. • With CFAST cards, the Hotplug ^[224] functionality is available.
SSD	Applicable to devices having a hard drive (see "Device overview" ^[116]). • With SSD hard drives, Hot-Plug ^[224] is not possible! If the SSD is used in a frame for removable data carrier, it can be exchanged while the device is deactivated. • SSD hard drives appear in the device software as a hard drive and can be read out via the Explorer-shell ^[226]. • Due to the formatting, the content of the SSD in the PC is not displayed when the SSD is connected directly in the PC. SSD hard drives can only be formatted in the device ^[229]. • In addition to the SSD, a CF/CFAST-card can be inserted in the measurement device and used alternatively.

8.6.2.2 Data transfer

The internal storage medium can be accessed **directly via Windows Explorer**. Alternatively, the storage medium can be inserted into a **card reader** on the PC (suitable for large amounts of data due to faster transfer).

 **Warning**

- Do **not use force** to insert or remove the device storage medium.
- During a **running measurement** having a high sampling rate, you should **never** try to **access the storage medium in the device** using the Windows Explorer shell. Otherwise, this additional burden could cause a data overflow.

 **Note**

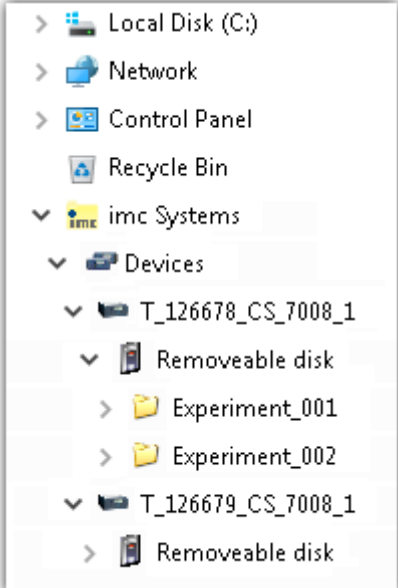
A tip on **interval saving**: Supposing the system's power supply suffers an outage during a measurement, it would not be possible to guarantee that the data file on the storage medium is terminated properly. This may lead to a failure to record the last measurement taken. Saving at intervals is a way to reduce this risk (see section "[Saving Interval](#)"^[202]).

Access via Windows Explorer

The menu item "*Data (Device)*" () causes the Windows-Explorer to start while indicating the device selected.

Ribbon	View
Extra > Data (Device) ()	Complete
Home > Data (Device) ()	Standard

Access via "*imc Systems*" - an Explorer shell extension



While installing the operating software, if the option "[Extension for Windows-Explorer](#)"^[24] is activated, you are able to copy, display or delete the files of measured data saved within the device (e.g. on the removable storage medium). The method of doing this is the familiar one under Windows.

This function is independent of the device software. As well, selection of the devices in the tree diagram is independent of the device list in the operating software.

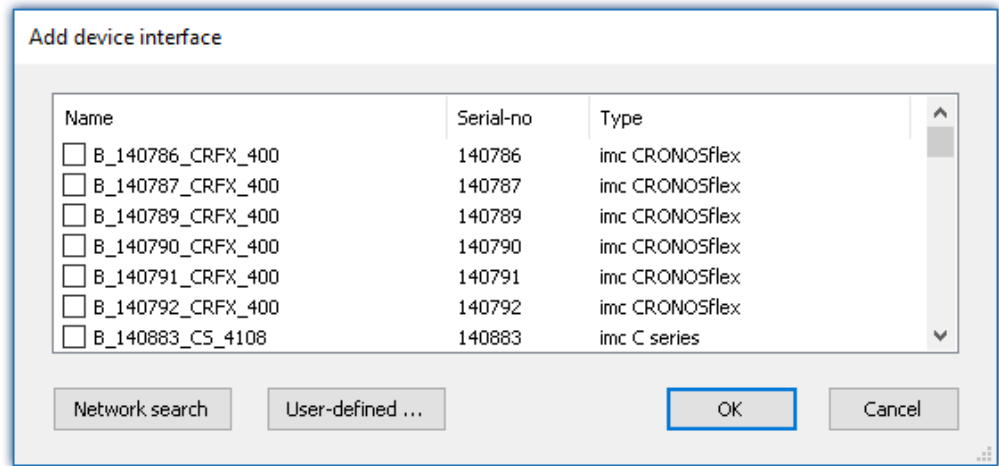
In order to get access to your device's storage medium, you must first add it to the tree diagram (see "[imc Systems - Adding a device](#)"^[227]). Subsequently, you are able to navigate to the desired data in the storage medium and thus to work with them.

8.6.2.2.1 imc Systems - Adding a device (Add)

Even if you have already been connected with the device by means of the imc STUDIO Monitor software, it is still not listed in the Explorer. It's possible to measure with one device while copying data from another.

- Click on "Devices" under "imc Systems" to highlight that entry.
- Open the context menu over the "Devices"-area and select "Add".

The "Add device interface" dialog appears:



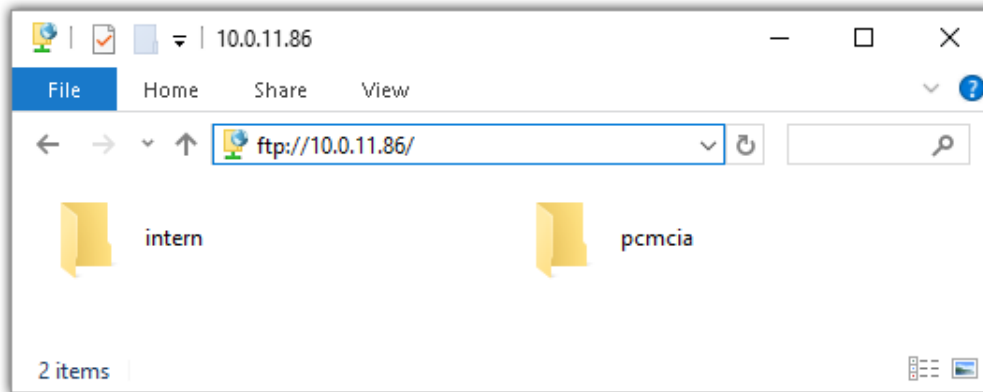
Add Device interface

Search for devices	Description
Network search	"Network search" causes the system to search the network for any suitable devices. How long this will take depends on how many devices are connected and on the network type. Ultimately, the devices found are listed. Select your measurement device and confirm your selection with "OK". The measurement device is then available.
User-defined	In a structured network (network with routers, Internet, ...), imc devices could not be integrated by means of a network search. With the knowledge of the IP address or of the domain name (DNS name), it is now possible to integrate a device into the list.

8.6.2.2.2 FTP access

It is also possible to access the device's internal data via FTP, as well as to transfer data. Other goals are to change the configuration of devices via FTP and to restart the device for measurement with the altered configuration. Application areas include test drives, where there is no way to connect the devices directly with the device software. The Diskstart/Autostart capabilities are applied and enhanced. For this purpose, the device is usually configured for autostart. Upon activation, the configuration is loaded and the measurement, as well as data transmission, starts automatically.

Start the explorer and enter "*ftp://*" and the IP address of the device:



Note

- In general it is a read only operation. If you intend to delete files via FTP, you have to add "*imc@*" between "*ftp://*" and the IP address:
Example: <ftp://imc@10.0.10.219>
- Furthermore, a password can be assigned to protect access via FTP. This password is entered into the device properties.

Warning

The following limitations apply when accessing the storage media in a device via an FTP-client:

- The device can't delete folders, accessed by a FTP-client.
- It is not possible to replace the storage medium during measurement (Hot-Plug).

8.6.2.3 File system and formatting

Storage media with the file systems FAT32 and FAT16 (maximum 2 GB) are supported. It is recommended that a memory card be [formatted](#)^[229] and possibly partitioned before use.



Note

Routine formatting protects the memory card

Routine formatting is recommended

Take every opportunity to format the storage medium. **Recommendation:** at least every **six months**.

In this way, any **damaged storage medium** can be detected and repaired if possible. A damaged file system may cause **data loss**. Or the **measurement system may fail to start** correctly.

In order to avoid data loss, any data still needed should first be saved!

Using a data storage medium in different devices

There are no known limitations. But it is recommended to always format the medium whenever transferring it in order to avoid data loss.

Additional notes

- To select the appropriate file system for the respective application, observe the notes on the data rate and on "[Avoidance of Data Overflow](#)"^[229].
- No limitations regarding the currently available storage medium volumes are known.
- The maximum filesize is 2 GB. In case a signal would exceed that limit, use [interval saving](#)^[202].



Notes

General restrictions applicable to file systems

Please observe the general restrictions regarding the respective file systems.

8.6.2.3.1 Formatting

The formatting can be performed directly in the PC's hard drive by the Windows operating system, or in the device using the Explorer shell.



Note

Recommendation

- **imc recommends formatting in the device:** In comparison to formatting by Windows, this provides higher data writing rates for high-speed channels.
- Only **one(!)** partition may be created. Multiple partitions may cause the measuring device not to recognize the storage medium.



Warning

Please back up the data first

Formatting causes all data on the storage medium to be deleted. Before performing the formatting, ensure that all data have been saved on a different storage medium.

Note

Cluster size - Avoidance of Data Overflow

The size and number of assignment units (clusters) and thus also the [file system](#)^[229] used have a substantial effect on the storage medium's speed! Small clusters can dramatically reduce the speed! If high data rates are required, it is normally recommended to have a size of 8 kB/cluster.

The optimum size of the clusters must be determined for each storage medium separately. For all of them, the following applies:

- **Few channels having a high data rate**

If a few channels having a high data rate are being recorded, then **large clusters** on the data carrier provide better advantage. Formatting with FAT32 on the PC and drive sizes < 8 GB creates disadvantageously small clusters, which in conjunction with the full aggregate sampling rate can lead to a data overflow.

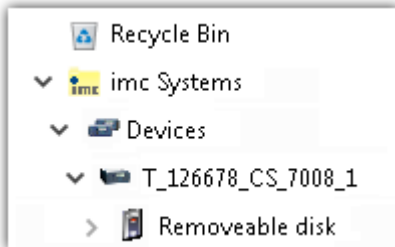
Whenever using cards of up to 8 GB, always use the formatting by the device.

In the device, cards larger than 512 MB are formatted with 8 kByte clusters and cards larger than 4 GB are formatted with 16 kByte clusters. Cards of up to 1 GB can alternatively be formatted by the PC with FAT16. With cards of 16 GB onward, there is no difference whether the formatting is done in the PC or in the device.

- **Very many channels with a low data rate**

If hundreds of channels having a low data rate (e.g. CAN channels) are saved, the exact opposite is true. Here, **small clusters** are an advantage. This means that drives with up to 8 GB should in such cases be formatted in the PC with FAT32.

Formatting in the device (Recommended)



For **formatting in the device**, navigate via the Explorer shell "[imc Systems](#)"^[226] to the desired device.

There, open the properties of the drive: context menu > "*Properties*" (not via the Navigation pane in the Explorer).

Go to the Property dialog under the tab: "*Tools*".

Start the formatting by clicking on "*Format now!*".

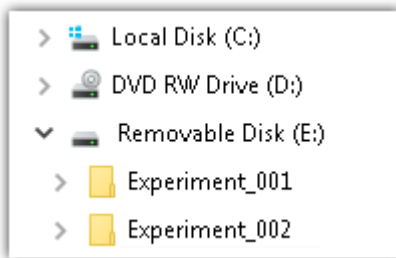
The device performs the formatting according to the following rule:

Drive size	Cluster size	File system
<= 512 MB	2 kB	FAT16
<= 4 GB	8 kB	FAT32
> 4 GB	16 kB	FAT32

Note

Formatting the storage medium is not allowed if an experiment whose data are to be saved internally has just been prepared.

Formatting using the Windows-Explorer



To perform **formatting of a storage medium via the [Windows-Explorer](#)**²²⁸, navigate to the desired storage medium. Run the formatting by means of the context menu, for example.

Select one of the following two file systems: "FAT32" or "FAT" ("FAT16").

The file system "FAT32" is designed for media **larger** than 32 MB. Under no circumstances can smaller media can be formatted to "FAT32". With "FAT32", Windows generates 4 kByte clusters when the drive size is up to 8 GB, which is inconvenient for high-speed writing rates.

SSD hard drives are inherently formatted with Ext2 and for that reason **can not be formatted directly in the PC**, but only in the [device](#)²²⁹.

However the Ext2 format offers these advantages:

- Mistaken duplicate occupation of individual clusters is not possible.
- Integration into the operating system is accomplished much more quickly than with FAT32.
- Higher writing output than with FAT32.

8.6.2.4 Known issues and limitations

Known issues and limitations	Description
If the memory card can not be read under Windows	The memory cards must first be partitioned (formatted) under Windows. Windows automatically generates the correct partitioning information. Subsequently, the memory card should be formatted again in the device. When in doubt, please contact our tech support ⁶ .
The system won't recognize the data storage drive	• Answer 1: Check the file system: The device supports FAT32/FAT16 ²²⁹. • Answer 2: If there are two storage media simultaneously plugged into your device, only one media will be detected (e.g. USB and CFast). Only the first plugged media will be detected.
File system becoming gradually slower	As the count of folders increases, so does the time required by the system to access the data. In consequence, the data saving procedure becomes slower and data loss becomes possible. For this reason, creating more than 1000 folders is to be avoided.
Errors in accessing the storage medium	Errors can have the following causes, among others: • The data rate is too high, the storage medium can't keep up and data overflow results. • The storage medium is full. The device signalizes any error by flashing this LED. Its further responses depend on whether or not the device is connected to the PC. • If no PC is connected, for instance in cases involving automatic self-start capability, the button lights continuously. At the end of an experiment, always check for this if measurements are taken without PC aid. • If the PC is connected to the measuring device, imc STUDIO Monitor documents the error with an message in the Logbook and switches the LED off. Any one-time data overflow only shows up in the Logbook, since the LED is reset afterwards. If data overflow occurs repeatedly, The LED is activated again, the PC records the message again, and as a result the LED blinks intermittently.
Data overflow due to improper cluster size	• With a storage medium formatted by Windows to FAT32 ²²⁹, data overflow can occur if a high aggregate sampling rate is generated by a few high-speed channels. • With a storage medium formatted in the device ²²⁹, data overflow can occur if a high aggregate sampling rate is generated by very many low-speed channels.

8.7 Information and tips

8.7.1 Data recording with sampling or with time stamp

Data recording with sampling

Equidistant sampling is performed according to the following algorithm: each time the sampling interval elapses, one new value is recorded.

- e.g. with an analog input: the instantaneous reading is recorded
- e.g. with a Fieldbus: the most recent reading transmitted on the fieldbus is recorded

The files are saved compactly **without any time stamp**, as a pure series of readings. The individual values' time references are derivable from the parameters "*Measurement start time*", "*Sampling interval*" and "*Count of preceding values*".



Note

When to use equidistant sampling

- If the sensor transmits at a more or less regular rate, then sampling is the most compact means of storing data and the quickest way of viewing it.
- If the is going to be processed with imc Online FAMOS. For many processing calculations, such as all digital filtering, FFTs, etc., it is important to obtain equidistantly sampled data.

Data recording with time stamp

To each reading, a time stamp is assigned, which reflects the point in time when the reading was measured. This time stamp occupies 6 Bytes. Thus, an individual time-stamped sample occupies 6 Bytes more memory than a sample having a fixed sampling rate.



Note

When does it make sense to capture time-stamped readings?

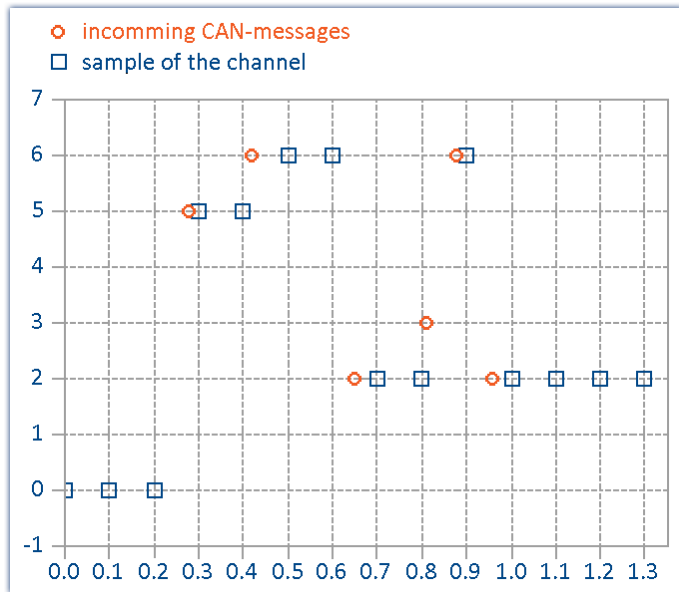
- When you intend to record the exact moment in time when a reading is captured.
- When the sensor is transmitting values at extremely irregular intervals, so that equidistant sampling would unnecessarily increase the data volume.

Disadvantages of time-stamp data acquisition

- It is not possible to subject the data to processing calculations, such as by using imc Online FAMOS.
- If a channel sends high volume of data, the memory requirements must be taken into account. In particular, the device's circular buffer memory requires correspondingly more memory for the same buffer duration than equidistant measurement.

8.7.1.1 Capturing fieldbus input data

Fieldbus messages arrive at more or less irregular intervals, i.e. they cannot be captured at a constant sampling rate. Nonetheless, in many cases it is desirable to record data at regular time intervals. In fact, many fieldbus sensors do attempt to send their messages at regular intervals. Even if they do not succeed at doing this, it is often acceptable to assume this to be the case.



The last reading captured via the Fiedbus is always recorded

This method has important consequences:

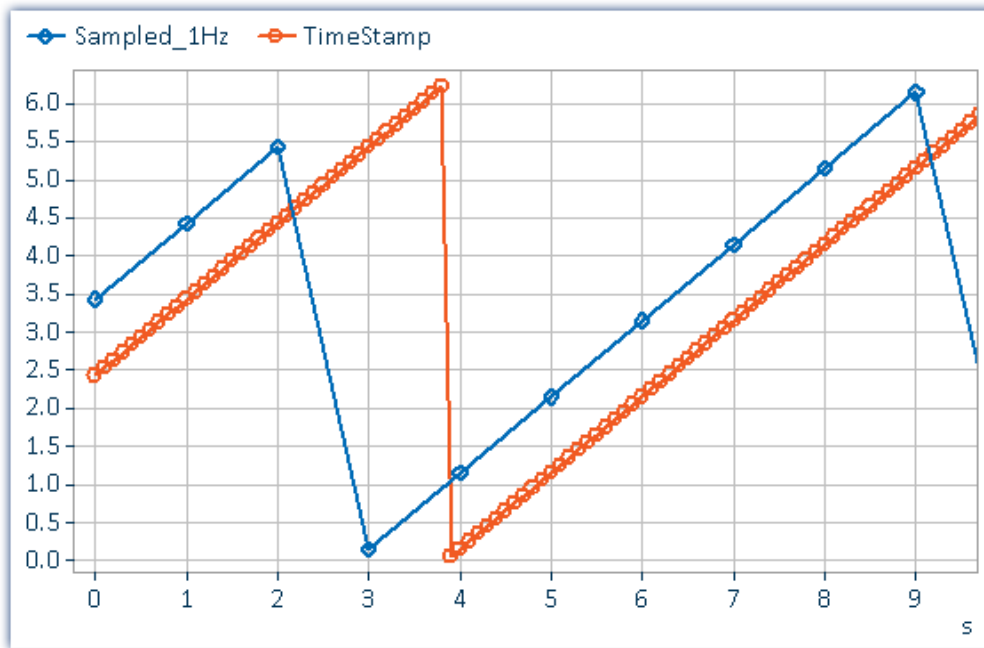
- As long as no messages have been received, the fieldbus channel is filled with zeroes. This can only occur at the beginning of an untriggered measurement before the first fieldbus messages have been received. When a measurement is stopped and restarted without being re-initialized, there are usually previous messages which are being used. In general, zeroes are not created when messages are being sent every 200 ms or less. If the zeroes at the beginning are not desired, we recommend performing a triggered measurement. When you start a triggered measurement, the system is armed and waits until the specified trigger condition (e.g. message value $\neq 0$) occurs.
- If the sampling rate is set significantly more rapid than the intervals at which the sensor transmits, you obtain unnecessarily many data, where successive data points have the same value, thus wasting memory and computation resources. If the sampling rate is significantly slower than the intervals at which the sensor transmits data, then the sparse sampling loses data values.
- It is recommended to select as far as possible a sampling interval matching the messages sending interval.
- If the sensor no longer transmits, the last value received is entered in the channel as the current sampling value.

Notes regarding the display in the curve window

When comparing time-stamped data with equidistantly sampled results from the same fieldbus channel, the equidistantly sampled data appear to be associated with earlier sampling times than the time-stamped data.

This would seem to contradict the previously described functioning principle of the system. What is actually happening is that the equidistant channel adopts the value measured at the end of the sampling interval, but places it at the start of that interval.

In the example below, a channel's messages are recorded with a time stamp (red: TimeStamp). The messages are sent every 100 ms. The same channel is additionally sampled at a fixed sampling rate 1 Hz. As the diagram illustrates, both channels begin at 0 s. The first value of the 1 Hz channel corresponds to the last message in the first sampling interval. The channel thus appears to be running ahead of schedule:



Why is the value placed at the beginning of the interval and not at its end?

With equidistant sampling there is always one sampling interval's worth of uncertainty in the time. For technical reasons, the physically captured value is delayed by the analog input filter, the digitalization, and the process of transfer to the fieldbus, so it is better to apply at least approximate compensation.

8.7.2 Behavior upon data overflow

If data overflow occurs when retrieving the measured data, a gap in the data will result. How the system handles this depends on the data storage settings.

With interval data saving

If a data overflow occurs when the measurement data are retrieved, resulting in a data gap larger than the sampling interval, a new file is created rather than an empty file for the missing data.

With and without interval data saving

If data overflow occurs **within an interval**, or in the **absence of interval data saving**, there are multiple approaches:

Data storage	Behavior
Data storage in the device	No gap in the measured data will be generated. The new samples continue to be appended to the existing measured data, no matter how many samples are lost. For this reason, they are assigned an incorrect time stamp (the samples have a time offset). As soon as a new file is started, the time stamp is again accurate (e.g. interval boundary).  For devices belonging to the firmware group A¹¹⁶ (imc DEVICES): In the "SysLog"-file¹⁹⁸, it is exactly specified when a data overflow occurred and how many data points are missing. Thus, the file can be corrected.
Data storage on the PC	The channel becomes an event-based data set. The new samples are saved in a new event. They thus have a correct time stamp. All data from before and after the overflow are present. The time stamp is correct.

 Reference	See also "Behavior upon outage of the network connection"²⁰⁶ (RAM-buffer)

9 Data analysis and signal processing

This chapter describes the various data analysis and signal processing possibilities available.

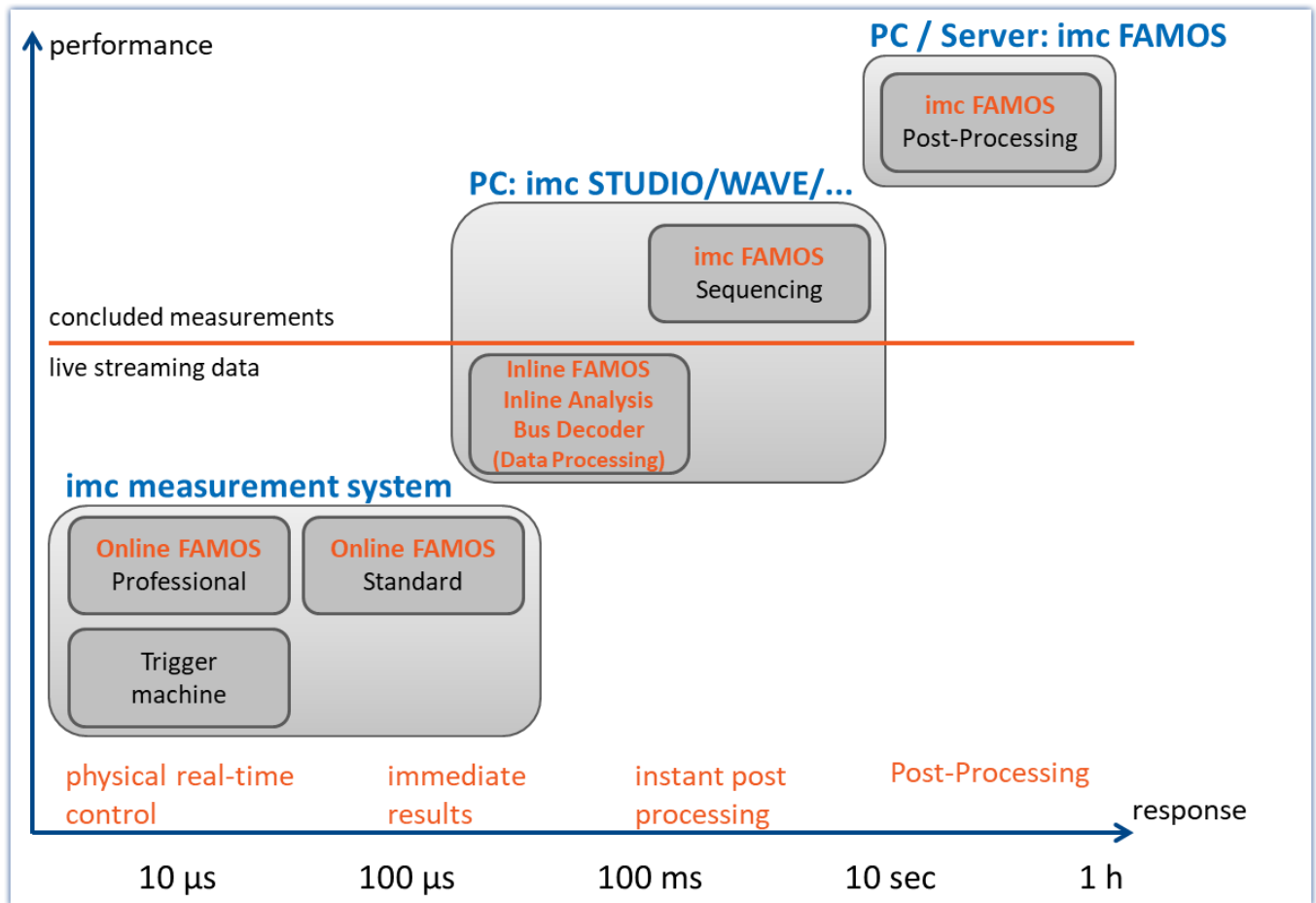
9.1 Overview of the tools

Tool	Description
imc Online FAMOS	Not available
imc Inline FAMOS  245	imc Inline FAMOS enables calculations to be performed on data streams from the measurement currently running. The calculations are performed on the PC, taking advantage of the PC's processing power. By contrast, with imc Online FAMOS, the calculations are performed by the device. A number of pre-defined functions are available for calculation purposes. Most of the imc Online FAMOS functions are available in the same way and with the same syntax as in imc Inline FAMOS. There are a few additional functions exclusively in imc Inline FAMOS.
Inline Analysis  451	Inline Analysis enables calculations to be performed on data streams from the measurement currently running. The calculations are performed on the PC, taking advantage of the PC's processing power. It is possible to compute a derivative or calculated channel from a input channel. For this purpose, a variety of functions packages are available.
imc FAMOS  475	From within imc STUDIO it is possible to transfer variable currently being measured to imc FAMOS and to process them with a imc FAMOS sequence there. imc FAMOS processes entire data sets. When you load a measurement to imc FAMOS, you load the complete measurement. You have equal access to the first and last values of a measurement. And you can also write imc FAMOS-sequences consisting of various loops and conditions for processing data. In imc Inline FAMOS, data are processed already while being captured. This means that the raw data are no longer in memory. The current raw data are used to generate calculation results. For all applications having online functionality requirements, it is not adequate to wait until the end of the measurement.
Bus Decoder  487	This component allows either all or individual measurement channels belonging to a log-channel to be decoded/extracted. A log-channel can be a logged Fieldbus communication ("Logfile"). This provides more flexibility and dynamic capability for deciding on targeted extraction of individual channels from the compressed logfile. The decoding is performed on the basis of the data streams of the measurement currently running on the PC. This utilizes the PC's computation resources.
Powertrain Monitoring  492	The component imc STUDIO Powertrain Monitoring has been developed in cooperation with the company GfM (Gesellschaft für Maschinendiagnose mbH) for power train diagnostics.

When should which tool be used?

Application fields:

- Real-time analysis and brief reaction time: imc Online FAMOS
- Complex scope of functions (calculation-intensive): imc Inline FAMOS
- Calculation-intensive subsequent analysis (post-processing): imc FAMOS



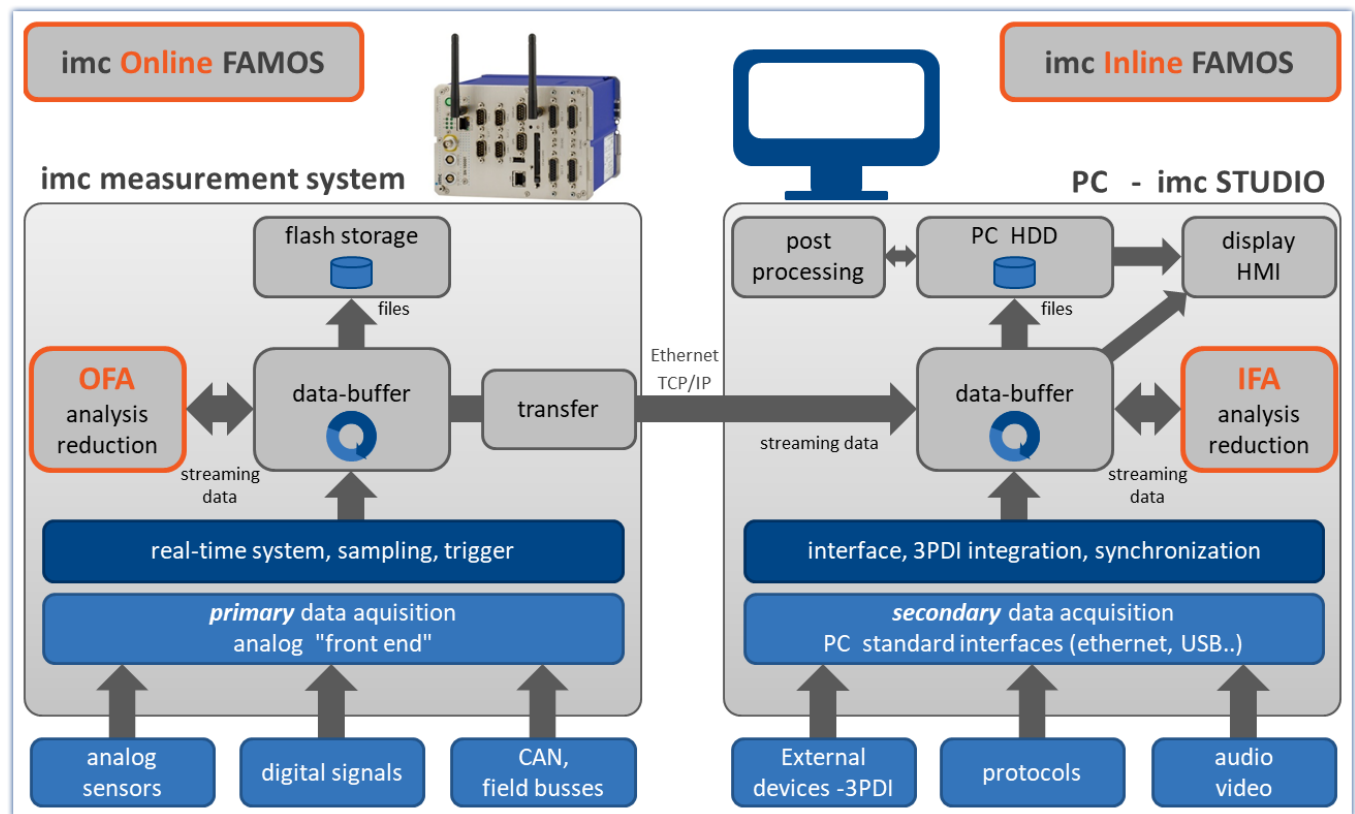
9.1.1 Comparing imc Online FAMOS with imc Inline FAMOS

imc Online FAMOS	imc Inline FAMOS
Device-based, classical real-time analysis	PC-based analysis of live streaming data diametrically different from imc FAMOS (post-processing of completed data sets)
Processing occurs where the data are captured, inside of the measurement device • no calculations across multiple devices possible • stand-alone capability	Processing occurs on the PC, and not where the data are captured. • Calculations across multiple devices possible (applies to all channels which are captured as of measurement start (BaseTrigger)), optionally also of 3rd-party devices (via 3PDI) • not stand-alone capable • correspondingly reduced real-time reaction • conversely: use of the powerful and scalable PC-platform
Resolution of the calculations and results: 4 byte	Resolution of the calculations and results: 8 byte
Commonalities: • Live-analysis: immediate visual feedback • Processing of live data streams: running, not concluded, measurements (not post-processing) • Unified syntax, same scope of functions • Application of calculation operations to combinations of multiple channels assigned to the same trigger	

Professional-Version

In imc Inline FAMOS, there are no version grades as with imc Online FAMOS (imc Online FAMOS / imc Online FAMOS Professional).

Data flow schematic



Data flow schematic

Comparison measurement

An example is presented in the section "[Performance comparisons](#)" ²⁴².

Scope of Functions

Most imc Online FAMOS functions are also available in imc Inline FAMOS in the same way and with the same syntax. Additionally, there are some functions available only in imc Inline FAMOS.

Among others, the following function groups are not provided:

- all closed-loop control functions and many system functions
- all CAN-functions and all ECU-functions
- ReadyForPowerOff, SyncOverload
- OnSyncTask (Synchronous Tasks), OnPowerOff
- IntegralP/IntegralP2

In the Functions Reference, it is noted if a particular function is available only in imc Online FAMOS or only in imc Inline FAMOS.

What kind of tasks can be accomplished?

Any task which could be performed by an electrical circuit can be handled. A circuit doesn't include loops or conditions, which are inappropriate to the streaming data we are dealing with. The data streams can only be processed and subjected to operations jointly with other data streams. The data streams can be transferred to the PC, to the storage medium, to DACs and to digital output bits.

If, instead of digital processing by imc Inline FAMOS, we put a complex analog circuit (e.g. filter, ...) in front of the device' inputs and digitalize the outputs of such a system, the channels recorded directly correspond to the virtual channels. They also possess all properties of a normal channel. Furthermore, if we consider logic operations and filtering (mean value, etc.) to be the main imc Inline FAMOS functions, then this is comparable to an analog circuit. Thus, imc Inline FAMOS can be viewed as a substitute for a freely configurable analog pre-processing circuit.

What can imc Inline FAMOS not do?

- It is impossible to formulate any processes which do **not** utilize **continuously streaming data**.
- imc Inline FAMOS works asynchronously. It processes data according to the formula that you have entered as quickly as possible. If the formula that you specify is somewhat long or involved, however, it may lead to delays. It is impossible to state a fixed time interval within which the loop can be processed.
- Fieldbus channels recorded in **time stamping mode, can't** be proceeded!
- **Unsigned 32-bit integer values** (32-bit UINT) are not calculated correctly.
 - Workaround: The channel must be set up in the CAN Assistant as an unsigned 31-bit integer. 32-bit UNIT channels are mostly created when a DBC file is imported. If such a channel is processed any further in imc Inline FAMOS, the bit count must be set to 31 bits following import of the DBC file.

9.1.1.1 Performance comparisons

A test comparing the performance of imc Online FAMOS with that of imc Online FAMOS is conducted using familiar hardware.

Test devices and configuration

Hardware used

Computer:

- WINDOWS 10
- Intel(R) Xeon(R) CPU E3-1270 V2 @ 3.50GHz 3.50GHz
- RAM: 8 GB
- Normal magnetic hard disk drive (not SSD)

Device:

- imc CRONOS compact 400 with two amplifiers of model UNI2-8

Test – configuration:

- Channel_001: 100kHz
- imc Online FAMOS: 2x FFT and 5x filtering, second order (bandpass, band-stop, high-pass, low-pass)
- RAM-buffer duration of all channels: 1s

```
; initialization prior to first measurement
```

```
OnInitAll
```

```
sample = 0
```

```
End
```

```
OnTriggerMeasure(BaseTrigger)
```

```
FFT_001 = fft(Channel_001, 1, 1024)
```

```
FFT_002 = fft(Channel_001, 1, 1024)
```

```
Res_BP = FiltBP(Channel_001, 0, 0, 2, 100.0, 1000.0)
```

```
Res_BS = FiltBS(Channel_001, 0, 0, 2, 100.0, 1000.0)
```

```
Res_HP = FiltHP(Channel_001, 0, 0, 2, 100.0)
```

```
Res_LP = FiltLP(Channel_001, 0, 0, 2, 100.0)
```

```
Res_LP_ = FiltLP(Channel_001, 0, 0, 2, 100.0)
```

```
sample = GetSampleCount(Channel_001)
```

```
GetSample = Channel_001 * 0 + sample
```

```
End
```

Test procedure and results

imc Online FAMOS:

Handling a 100kHz channel, the imc Online FAMOS test program is at the edge of its capacities. The virtual channel "GetSample" continuously shows the limit value. In the curve window, the distance between the analog channel "Channel_001" and the calculated channel continually increases. As soon as the RAM-buffer of 1 second is no longer sufficient, data overflow occurs. This results in a **loss of data**.

Endurance test (= 24h) capability is just possible in this example with 2xFFT + 4 filters.

imc Inline FAMOS:

Endurance test capability still prevailed with 25xFFT + 50 filters, which is more than 12 times the performance of imc Online FAMOS. Here, the qualification for endurance test capability is considered to be simultaneous display of the analog and the virtual channel in the curve window.

As well, significantly more substantial calculations are possible in imc Inline FAMOS. However, depending on the performance of the computer's respective hardware configuration, the curve window will **eventually no longer display the data in real time**.

Since the data storage management is handled by Windows, this moment in time is not predictable. With a 24-fold load (50xFFT + 100 filters) by comparison with the OFA experiment, this hardware experienced a **data overflow in the curve window** after 1.5 hours. However, there was no resulting loss of data since the data can be retrieved after the end of the measurement. When true data loss would occur in the computer is not possible to determine.

This result applies exclusively for the hardware and software settings specified!

Conclusion

The dynamic data storage management provided by WINDOWS enables multiple times the computation power of the fixed RAM-buffer structure in the device. This is scalable in accordance with the processor speed and PC's RAM size. imc Inline FAMOS is thus future-proof, able to keep pace with technological progress without requiring replacement of the actual measurement hardware.

9.1.2 imc Inline FAMOS - System requirements and limitations

The limitations of imc Online FAMOS do not apply to imc Inline FAMOS!

The amount and scope of the tasks which can be performed depends on the computational demands of the functions used and on the power of the PC used. Since all calculations are performed by the PC, the PC's performance determines the possible scope.

Minimum PC requirements ¹	Recommended PC configuration ²
The minimum requirements for imc STUDIO apply here.	The recommended configuration of imc STUDIO applies here. Additional recommendation: Processor with at least 2+n processor core, where n is the number of Tasks.

¹ A system with the minimum requirements is not sufficient for intensive calculations.

² To ensure smooth performance of calculations, storage and visualization of the measured data, one processor core per Data Processing-Task should be available. Ideally 2+n; where n is the number of Tasks.

Which channels can be combined in calculation operations?

1. Channels belonging to one device

Here, the same prerequisites apply as for imc Online FAMOS. Channels can be combined together when they are subject to the same trigger. E.g. Start/Stop-trigger (BaseTrigger) or a trigger x.

2. Channels belonging to different devices

Channels which are assigned to the Start/Stop trigger (BaseTrigger) can be combined in calculations across multiple-devices.

Calculations on combinations of triggered channels from different kinds of devices will not work because:

- Trigger x may be defined differently on differing devices
- Device triggers do not propagate and synchronize globally, i.e.: a device "A" channel can not be triggered by a device "B" trigger

Process vector and other device variables

All device variables, including process vector and Display variables, network and virtual bits can be processed and set like in imc Online FAMOS.

Process vector variables can not be set up in imc Inline FAMOS. Process vector variables are device variables explicitly created for a specific device, e.g. via imc Online FAMOS, via the plug-in Setup or via Automation.

Process vector variables can be saved and restored exclusively on the device.

Refresh/writing rate of the device variables

The reading and writing access to the devices is limited.

9.2 imc Inline FAMOS

imc Inline FAMOS

Processing and analysis of measured data during a running measurement

imc Inline FAMOS is a functions package for Data Processing.

imc Inline FAMOS enables calculations to be performed on data streams from the measurement currently running. The calculations are performed on the PC, taking advantage of the PC's processing power. By contrast, with imc Online FAMOS, the calculations are performed by the device.

Scope of functions

A number of pre-defined functions are available for calculation purposes. Most of the imc Online FAMOS functions are available in the same way and with the same syntax as in imc Inline FAMOS. There are a few additional functions exclusively in imc Inline FAMOS.

Cross-device calculations

In contrast to imc Online FAMOS, imc Inline FAMOS provides the ability to apply calculation operations to channels belonging to different devices, if the channels are assigned to "*BaseTrigger*" (measurement Start/Stop).

Displaying results:

The results generated are treated as device variables/channels. They are configured on the Setup pages (e.g. Storage) and can be displayed on Panel pages.

Tasks:

Multiple independent, complete calculation sequences (**Tasks**) can be processed in parallel. The maximum possible scope/amount of these tasks depends on the computational resources required by the functions used, in conjunction with the computational resources available to the PC used.

9.2.1 Overview

imc Inline FAMOS works with streams of binary data.

Take a simple example where the difference between two channels is computed:

Difference = Input1 - Input2

If data from both **Input1** and from **Input2** are present, then the **Difference** is computed and assigned to the virtual channel **Difference**.

When measurement is started, the device performs the following procedure:

Repeat while measurement continues

Wait for **Input1** data

Wait for **Input2** data

Retrieve last value of **Input1** from memory

Retrieve last value of **Input2** from memory

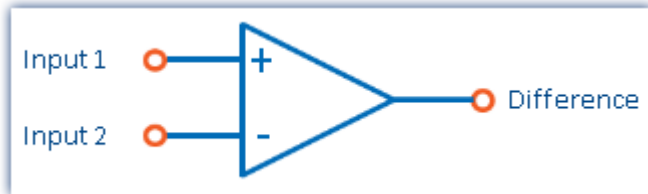
Compute difference

Assign computed difference to virtual channel: **Difference**

The virtual channel only contains values if the channels on which its calculations are based contains measured values. This means that only after the input channel is triggered is it possible for the virtual channel to be calculated. When the input channel's signal is concluded, the virtual channel is also concluded.

In the diagram of the internal process above, you see a "Repeat while..." loop. In the equation defined, $\text{Difference} = \text{Input1} - \text{Input2}$, however, there is no loop. In spite of this, a loop is running internally in order for the term **Difference** to be calculated while the input channels are returning input channels.

Calculation of the difference can be compared to the following simplified electrical circuit diagram:



Schematic circuit diagram of a calculated difference

Output of an analog output

We now consider assigning the difference to an analog output channel (DAC channel):

$$\text{DAC1} = \text{Input1} - \text{Input2}$$

The following steps are executed internally:

<i>Repeat while measurement continues</i>	
Wait for Input1 data	
Wait for Input2 data	
Retrieve last value of Input1 from memory	
Retrieve last value of Input2 from memory	
Compute difference	
Assign computed difference to DAC1	

The procedures for calculating a virtual channel and for generating a DAC output are the same. Thus, the DAC can be referred to as a virtual channel identified by its permanent name (a system name). The DAC is not saved but outputted directly to the respective output terminal. The output to digital output bits proceeds in the same way.

9.2.1.1 What happens when imc Online FAMOS overflows?

General notes on data overflow in software

imc Inline FAMOS operates without synchronization to the data capture. This means that imc Inline FAMOS works through its program in an uninterrupted loop.

imc Inline FAMOS working without overflow

If the data rate is low enough, and there are no new samples available for calculations, then there is nothing for imc Inline FAMOS to do.

If the online calculation is so complex that new samples accumulate by the time the calculation is done, then imc Inline FAMOS enters the new data into the next calculation iteration. If the average number of new samples remains equal, there is no overload.

imc Inline FAMOS overflows

So what happens if the data rate is so high that the number of new samples increases after every iteration? In such a case, imc Inline FAMOS falls behind in performing the calculations. The amount of data increases and with it the time required for calculation.

You will notice this overflow if you display the input data and the virtual channels in a curve window. The virtual channels' values are still pending calculation and are not yet visible, while the input data is already on display. In consequence, the gap between the input data and the virtual data grows at the end of the channel graph.

9.2.1.2 Source text with Control commands

Conventional **source text without control commands** mainly provided the ability to define virtual channels and subject them to calculations. An imc Inline FAMOS program is executed line-by-line; all instructions entered are carried out in the order they appear in the program.

If the **Control Commands** are activated, the ability to use branching instructions is provided. Program branching can be achieved using conditions and case differentiation, for instance: "if condition is fulfilled, carry out one operation, otherwise carry out other operation". Thus, different procedures can be defined in imc Inline FAMOS to depend on values of variables.

Also, operations can be directly assigned to various states within the course of a measurement (e.g. at the start, end, or during the measurement). An example shall illustrate the difference between "with" and "without" control commands:



Example

```
OnTriggerStart( BaseTrigger )      ; to be carried out at start of measurement
    Reset = 1                      ; Reset is a digital output
End

OnTriggerMeasure( BaseTrigger )    ; to be carried out during measurement
    If Temperature > 25             ; branch, querying temp. channel signal
        Reset = 1
    Else
        Reset = 0
    End
End
```

At the start of the measurement, the digital output bit *Reset* is activated. During the measurement, *Reset* will be set depending on the Temperature.

More information on syntax is presented in the section: "[Variables and Syntax with Control Commands](#)".

9.2.1.2.1 Transforming to imc Online FAMOS with/without Control commands

By default, imc Inline FAMOS starts without control commands.

Activate the Control Commands via the [context menu](#) ²⁵⁶ in the Editor:

Menu item	Description
 With control commands	Deactivates/activates the Control Commands.

Upon making this switch, control commands are automatically added to the source text in the Editor. To prevent making this switch inadvertently, it must first be confirmed in order to take effect.

The control commands added automatically are:

Control command	Description
OnInitAll	initialization prior to first measurement, or for imc Inline FAMOS: after starting the Task
OnAlways	for code segment performed constantly
OnTriggerStart	for code segment performed once at the start of a measurement
OnTriggerEnd	for code segment performed once at the end of a measurement
OnTriggerMeasure	for code segment performed for duration of measurement

If operations were already in the editor when the switch is made, these operations are automatically assigned to the control commands [OnAlways](#) and [OnTriggerMeasure](#). Usually, no revisions are necessary.

Deactivating Control Commands

Deactivation of the control commands is also effected by means of the menu item stated above.

Only such operations are directly transferred which can also be used "*without control commands*", while all other operations still appear, but commented out. After such a transformation, alteration of the source text is usually also necessary, and will sometimes cause significant reduction of functionality.

As a rule, it is inadvisable to port a program created with control commands to "*without control commands*".

9.2.1.2.2 Additional groups in the function list

When you activate the Control Commands, then among others, the following three additional groups appear in the function list: "*Controls*", "*Control functions*" and "*Comparison operators*".

The "*Controls*" group lists all control commands supported. This comprises control commands for conditions and case differentiation (e.g. [IF](#)), on the one hand, and on the other hand control commands concerning various measurement process states, e.g. [OnTriggerStart](#) (for operations performed at the start of a measurement).

The "*Control functions*" group contains special functions; such as timer functions.

The "*Comparison operators*" group contains comparison operators such as [>](#), [<](#), [<>](#) and [=](#). As well, the logic operators [AND](#), [OR](#) and [NOT](#) are provided here.

 Reference	Function Reference
The functions are described in the chapter " imc Inline FAMOS function reference " ²⁷⁹ .	

9.2.2 Operation

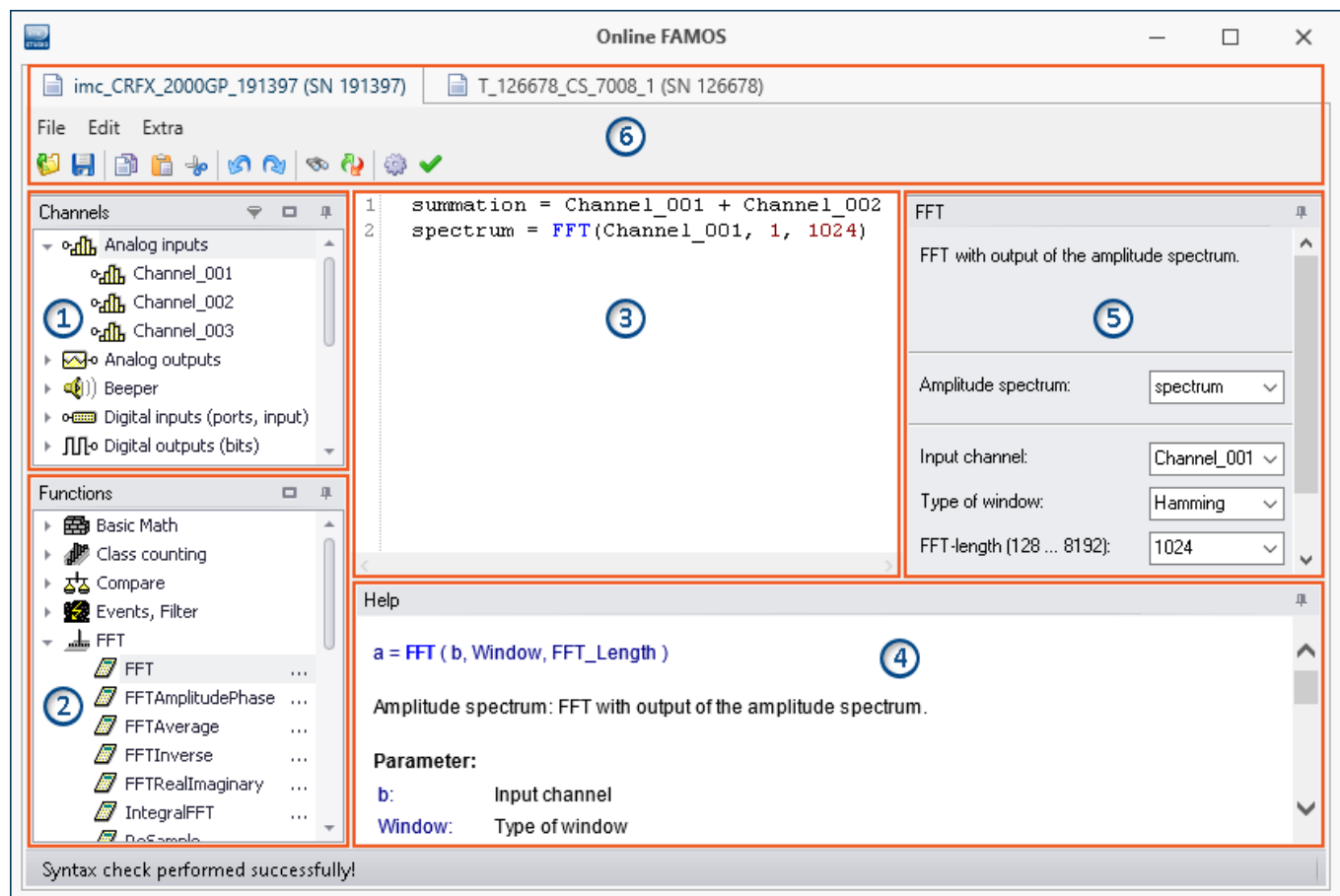
9.2.2.1 Opening the editor

imc Inline FAMOS

Dialog: Select "imc Inline FAMOS" from the Ribbon "*Home*" (or "*Setup-Configuration*") to start imc Inline FAMOS. The imc Inline FAMOS Editor is displayed in a dialog.

Embedded: Go to the Data Processing via the Navigation pane. There is a page tab (Task: ) for the respective function package imc Inline FAMOS . If it is not yet selected, click the left mouse button on the Task name. The imc Inline FAMOS Editor is displayed in the main window.

9.2.2.2 User interface



The window is subdivided into six regions:

1. [Variables list](#) ²⁵¹
2. [Function list](#) ²⁵¹
3. [Editor](#) ²⁵² (text box) for the calculation operations
4. [Help](#) ²⁵²
5. [Formula assistant](#) ²⁵²
6. [Menu and actions](#) ²⁵⁵ (only in imc Online FAMOS)

Region 1: Variables List

Here is the list of all available variables supported (imc Inline FAMOS: among others device variables and user-defined variables).

You can group the variables according to the variable type. With or without grouping, a symbol indicating the respective variable type appears before each name. Variables which were created in the current Task have a green symbol.

Icon	Description	Icon	Description
	active analog input channels and incremental counter channels		Fieldbus: Analog inputs
	Digital inputs (ports)		Fieldbus: Digital inputs (bits)
	Digital outputs (ports)		Fieldbus: Digital inputs (ports)
	calculated virtual channels		housing LED's
	calculated local channels		Virtual bits
	calculated local single value variables		Ethernet bits
	Process vector variables 		Display variables
	Digital inputs (bits) (of all DIO-ports set for "Bit-input")		Beeper (miniature speaker)
	Digital outputs (bits) (of all DIO-ports)		Trigger which starts an analog channel. Only those triggers are listed to which at least one channel is assigned.
	Analog outputs (DAC)		Local text variables
	Event channel		

Region 2: Functions List

Here is a list of all available mathematics functions and characters. With the help of the functions, you can perform calculations on the channels and variables. As a result, you obtain virtual and local channels or local single value variables.

The mathematical functions are always updated instantaneously. To learn about their workings, see the Help-box or the Functions Reference.

Region 3: Editor for the calculation operations

In the text box, enter the calculation operations. You may insert carriage returns and spaces as desired. However, any instruction must be coded in its entirety on one line!

Adding a variable or function to the editor

There are multiple ways to add an element to the editor:

- by **double-clicking** on the element
- by applying the **Drag&Drop** technique to the editor
- by calling the **Formula Assistant**: using the Formula Assistant, it is easy to set function parameters and enter the functions into the editor.
- by **input** with **auto-completion** support

The "first suggestion" from the list is applied by means of the Tab-key. When the ENTER-key is pressed for the "first suggestion", a line break is inserted. When navigating through the list of suggestions, either TAB or ENTER can be used to apply the suggestion.

Functions are entered into the editor along with parenthesis.

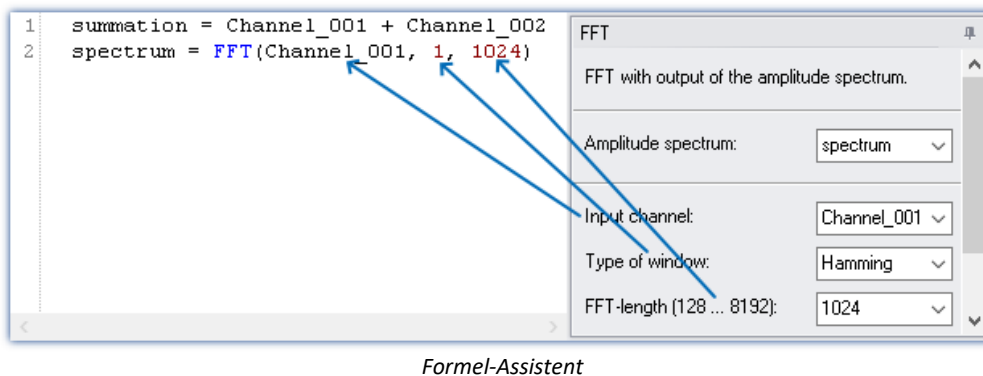
Region 4: Help

Here you find an explicit description of the functions and of the variables' properties.

To open the help text for a variable or a function, left click the mouse to select the desired item. The editor remains unchanged.

Region 5: Formula-Assistant

The Formula-Assistant provides support in parameterizing the functions. The Assistant shows the function selected in the editor.



Formel-Assistent

What parameters are needed depends on the math function selected. For instance, for the Rainflow-function, there are 10 parameters to set. Procedures, in contrast to functions, don't have result values.

For the result, a variable name can be entered manually or an output channel can be selected from the pop-down list.

Changes to the parameters in the Formula-Assistant are immediately adopted in the editor and vice versa.

9.2.2.3 Brief introduction

The Formula Assistant helps you in selecting and parameterizing a mathematical function.

Clicking on a function in the Function List calls a help window displaying the associated help text. You can magnify the help window if desired.

Enter the function along with its parameters in the Editor. You may wish to use the Formula Assistant for this purpose. To assign an operation to a virtual channel, use an equals sign. The name for the virtual channel can be freely specified.



Example

```
summation = Channel_001 + Channel_002
difference = Channel_003 - 5
DAC_01 = difference
```

In the example, we have assigned a sum and a difference to two virtual channels as well as outputting channels on a DAC channel.



Note

Note on analog outputs (DAC)

The analog outputs of the measurement device have an output voltage range of -10 V...10 V. You can therefore only assign values lying within this range! A scaling in the plug-in Setup for the DAC channels is taken into account.

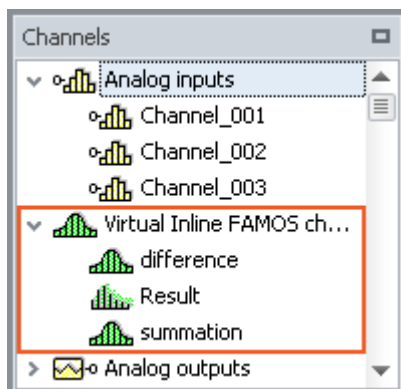
Adding a variable or function

- You can use Drag&Drop to take names from the Variables list into the Editor. Or, alternatively, double-click on the name to insert it in the editor.
- Analogously, you can move functions from the Function List to the editor using Drag&Drop. Just double-clicking on a function's entry in the Function List move the function into the editor.

Definitions for each individual function are made on its own **separate line**.

Check / Syntax check |

Click on the button to perform a "Syntax check". If an error occurs, a notification will appear in the status bar! The faulty portion will appear highlighted in the Editor.



If the syntax check concludes without an error, all newly defined virtual channels and variables will be adopted in the Channels list.

Optional: Saving the source text

Extra saving of the source text is not necessary, since the entries are saved with the experiment. However, it is possible to create an extra copy of the source text. This file can be stored externally, but is not used in the experiment. The source text file can be re-imported at a later time or used in a different experiment.

To save the source text, open the Editor's context menu and select "Save source text" ().

Applying the imc Inline FAMOS-configuration

If the Syntax-Check concludes without finding an error, the Task will still not run. As soon as the configuration has been **applied**, the Task starts and calculations are performed. To do this, click on the button *Apply* ().

In the table of channels in the Setup, the newly defined channels appear as additional **virtual channels**.

9.2.2.4 Menu

The menu is not included in imc Inline FAMOS. All functions can be accessed via the "[context menu](#)²⁵⁶". The description of the Data Processing menu ribbon is presented here: "*Data Processing*" > "[Ribbon](#)⁴⁸⁶".

File - Menu

Menu item	Hot key	Description
 Load source text	(Shift + F2)	Loads a text file.
 Save source text	(Shift + F3)	The contents of the editor are saved in a text file. This does not replace the Save to Configuration command!

Edit - Menu

Menu item	Hot key	Description
 Undo	(Ctrl + Z)	Reverses the last change in the Editor. This function can be used multiple times in succession.
 Redo	(Ctrl + Y)	Restores the previously reversed change. This function can be used multiple times in succession.
 Undo all		Reverses any changes which have been made to the source text since the Editor was called.
 Find	(Strg + F)	Finds texts in the imc Online FAMOS editor.
 Find and Replace	(Strg + H)	Finds and replaces texts in the imc Online FAMOS editor.
 Copy	(Ctrl + C)	Copies the highlighted region in the Editor to the Clipboard.
 Paste	(Ctrl + V)	Inserts the contents of the Clipboard at the highlighted location.
 Cut	(Ctrl + X)	Extracts the highlighted region of the Editor and moves it to the Clipboard.
 Delete	(Ctrl + Del)	This deletes the selected area from the editor.
 Properties ²⁷²	(F5)	Opens the Properties window. There, the properties of virtual channels can be edited subsequently.
 Syntax check	(F6)	The source text is checked for errors. If an error occurs, a notification appears in the status bar! The faulty spot is highlighted in the Editor. If the Syntax-Check concludes without any error located, the status bar reads: "Syntax-Check performed successfully!". All newly defined virtual channels and variables are adopted into the Variables list.

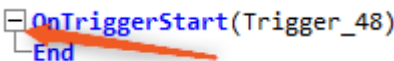
Extras - Menu

Menu item	Description
Supplemental files	See " Supplemental files " ^[277] - Menu item is only available for devices of firmware group B ^[116] .
 imc Online FAMOS with control commands	Activate control commands ^[247] .
 imc Online FAMOS without control commands	Deactivate control commands ^[247] .

9.2.2.5 Context menu

Editor

Right-clicking in the editor calls the following context menu:

Menu item	Hot key	Description
 Load source text	(Shift + F2)	Loads a previously saved source text file and thus overwrites the current configuration.
 Save source text	(Shift + F3)	Saves the source text and/or the properties at a specific location.
 Copy	(Ctrl + C)	Copies the highlighted region in the Editor to the Clipboard.
 Paste	(Ctrl + V)	Inserts the contents of the Clipboard at the highlighted location.
 Cut	(Ctrl + X)	Extracts the highlighted region of the Editor and moves it to the Clipboard.
 Undo	(Ctrl + Z)	Reverses the last change in the Editor. This function can be used multiple times in succession.
 Redo	(Ctrl + Y)	Restores the previously reversed change. This function can be used multiple times in succession.
 Syntax check	(F6)	The source text is checked for errors. If an error occurs, a notification appears in the status bar! The faulty spot is highlighted in the Editor. If the Syntax-Check concludes without any error located, the status bar reads: "Syntax-Check performed successfully!". All newly defined virtual channels and variables are adopted into the Variables list.
 With control commands	-	Deactivates/activates the Control Commands ^[247] .
 Properties	(F5)	Opens the Properties window. There, the properties of virtual channels can be edited subsequently (see Virtual channels ^[258]).
 Show line numbers	-	Turns the display of line numbers in the Editor on/off.
 Show folding	-	Turns on/off the code-folding in the Editor for the control commands, used to group logically associated source text sections. 

Menu item	Hot key	Description
 Collapse foldings	(Ctrl + J)	All folds are folded.
 Unfold foldings	(Ctrl+Shift + J)	All folds are folded out.
 Autocomplete	(F7)	The automatic text predicting function provides you with suggestions of possible terms while you are typing. You can use the Tab-key to adopt the current selection in the Editor. • The "first suggestion" from the list is applied by means of the Tab-key. • When the ENTER-key is pressed for the "first suggestion", a line break is inserted. • When navigating through the list of suggestions, either TAB or ENTER can be used to apply the suggestion.

Variables List

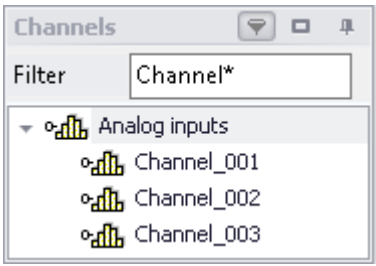
Right-clicking in the Variables List calls the following context menu:

Menu item	Description
 Tree view	The variables are displayed in groups by channel type.
 List view	All variables are displayed in a list. The channel types are not indicated explicitly. The type is denoted by the symbol in front of the name.
 Expand tree	Expands the groups in the Tree view.
 Collapse tree	Collapses the groups in the Tree view.
 Sort by types	Sorts by channel type in the List view.
 Sort by names	Sorts by name in the List view.

Shows a filter bar.



Filter



Filtering the channel list with Wilcards

Function List

Right-clicking in the Function List calls the following context menu:

Menu item	Description
 Tree view	All functions are displayed in alphabetical order by function group.
 List view	All functions are displayed as an alphabetically ordered list.
 Expand tree	Expands the function groups in the Tree view.
 Collapse tree	Collapses the function groups in the Tree view.

Help

Right-clicking in the Help box the following context menu:

Menu item	Description
 Copy all	Copies the complete text from the current help text to the Clipboard.
 Copy example	Copies any existing examples from the current help text to the Clipboard.

9.2.3 Variables and syntax

9.2.3.1 Virtual channels and local variables (channels and single value)

Defining virtual channels

Virtual channels are defined in the editor.

A channel assignment consists of

- the virtual channel's freely selected name,
- the equals sign, and
- an arithmetic expression, which may, for example, include an already existing channel.



Example

```
Sum = Channel_001 + Channel_002
CorrectedOffset = Channel_001 + 5
Stretch = 2 * Channel_003
```

You can also use parentheses and minus signs:

```
Complicated = -( 3 + 4 * ( Channel_001 + 1 ) )
```

Function arguments must be written in parentheses:

```
SquareRoot = sqrt ( Channel_001 )
```

Formulas which are not permitted:

Incorrect	Explanation
a = DAC1	This is not allowed because an analog output (DAC) can only be written to but not read from.

Local channels and single value variables

If you need intermediate results to compute a virtual channel, it is possible to keep them local by prefixing an **underline** to their names. This applies both to single values and to channels.



Access to local variables is twice as fast as normal virtual channels or single values. This increases the computation power. The disadvantage is that these variables are then no longer visible to the PC.



Local variables appear in the Variables List with a trash bin symbol when they were created but no longer used. For example, if they became obsolete in the process of developing the source text but were forgotten rather than deleted.

Creating local channels

A virtual channel remains local if the variable's name begins with an underline "_".



Example

```
LocalChannel = Channel_001 * 3
Virtual1 = LocalChannel + 1
Virtual2 = LocalChannel + 2
Virtual3 = LocalChannel + 3
```

Creating local single value variables

Local single value variables are assigned once and can later be used in subsequent lines. A single value is an isolated number without any further properties. Single values only exist locally and do not require an underline: "a = 1" or "a = 1".



Example

```
constant = 3 + 4 * sqrt ( 5 )
Virtual1 = Channel_001 * constant
Virtual2 = Channel_002 * constant
Virtual3 = Channel_003 * constant
```

9.2.3.2 Polling digital Inputs

Digital inputs can be polled by imc Online FAMOS. However, they are not polled at a fixed sampling rate, but rather as fast as the signal analysis platform can process the internal loop. The digital inputs are thus not a data stream, but are in the nature of single values.

A data stream can only be created in conjunction with a channel.



Example

```
BitStream = Channel_001 * 0 + DIO01_Bit11
```

Virtual bits can also be polled:

```
Constant1 = Virt_Bit_04 ; SW only locally possible
```

The return value of a digital bit is 0 for "LOW" and 1 for "HIGH".

9.2.3.3 Setting system outputs

The system outputs must be defined in a single statement. The left side of the assignment contains the system output name. The following system outputs are possible (as long as they are displayed in the Variables List):

- Analog outputs (DAC)
- Digital outputs
- Beeper
- LED
- Virtual bit and Ethernet bits
- Display variables
- Triggers

Analog outputs (DAC)

The analog outputs of the measurement device have an output voltage range of -10V...10V. You can therefore only assign values lying within this range! A scaling in the plug-in Setup for the DAC channels is taken into account.

Definition:

DAC1 = 5	You can assign constants or data streams to a DAC. With data streams, the most recent valid value appears at the DAC.
DAC2 = Channel_001 / 2	

Not allowed:

Virtuell = DAC3	appearance on the right side of an equation
DAC1 = 1 DAC1 = 2	multiple assignment
DAC4 = 125	exceeding the value range

Digital outputs

For digital outputs, the assignment may take either of the values zero or one. 0 for LOW and 1 for HIGH. Internally, any assignment not equal to zero is set to 1. Only exactly zero equals 0.

Definition:

DIO02_Bit01 = 1 DIO02_Bit02 = 0 DIO02_Bit03 = STRI(Channel_001, -5, 5)	If you use mathematics functions, ensure that the resulting return value is either exactly 0 or 1. The Schmitt trigger function is particularly suited for generating digital signals from analog signals.

Not allowed:

Virtuell = DIO02_Bit04	appearance on the right side of an equation
DIO02_Bit01 = 1 DIO02_Bit01 = 0	multiple assignment

Beeper

The beeper is to be considered a digital output which is directly connected with the beeper. Only a tone of fixed pitch can be emitted.

Definition:

BEEP1 = 1	Assigning 1 (TRUE) to the beeper turns it on and assigning 0 (FALSE) to it turns it off.
BEEP1 = 0	
BEEP1 = STRI(Channel_001, -5, 5)	

LED

Some imc measurement devices have LEDs on the housing, which can be operated as a form of digital output. An LED can either shine or be off. (See also in the device operating manual "*LED-flashing during measurement*")

Definition:

LED1 = 1	Assigning 1 (TRUE) to the LED turns it on and assigning 0 (FALSE) to it turns it off.
LED1 = 0	
LED1 = STRI(Channel_001, -5, 5)	

Virtual bits and Ethernet bits

Virtual bits and Ethernet bits are set in the same way as digital outputs:

Definition:

Virt_Bit_01 = 1	Assigning 1 (TRUE) or 0 (FALSE).
Virt_Bit_01 = 0	
Virt_Bit_01 = STRI(Channel_001, -5, 5)	

Display variables

Display variables are set in the same way as digital outputs. However, they can take on a wider (4Byte) range of values.

Definition:

DisplayVar_01 = 1	Assigning a number.
DisplayVar_01 = 123.456	
DisplayVar_01 = STRI(Channel_001, -5, 5)	

Triggers

A trigger can be released not only in response to events as defined in the plug-in Setup, but also by imc Inline FAMOS.

As soon as an assignment of 1 is made to an armed trigger, this trigger is released.

This does not change the arming of the trigger. A trigger can initiate recording of data. imc Inline FAMOS releases the trigger just as if a different event had released the trigger.

Triggers with the combination "*Passive*" or "*Immediate*" cannot be released using this method.

Definition:

Trigger_01 = 1	By assignment of a 1 (TRUE), the trigger is released.
Trigger_01 = 0	
Trigger_01 = STRI(Channel_001, -5, 5)	

9.2.3.4 Syntax: channel name

Usually the channel name can be used as variable name directly.



Example

```
; Channel name "My_channel"
Res= FFT( My_channel, 2, 1024)
```

If a channel name begins with a number or contains special characters such as \?+! or a space, the channel name must be set into **curly braces**.



Example

```
; Channel name "123 My channel 100% display"
Res= FFT( {123 My channel 100% display}, 2, 1024)
```

9.2.3.5 Syntax: comment

Single line comment

A complete line or a part of a line can be turned into a comment with a **semicolon**.



Example

```
; Computation begins here
Sum = Channel_001 + Channel_002 ; Computes the sum
; Computation ends here
```

Multiline comments / Block comments

Several lines can be commented using `(* .....*)`.



Example

```
(* the following lines are commented
Sum1 = Channel_001 + Channel_002 ; computes the sum
Sum2 = Channel_003 + Channel_004 ; computes the sum
Sum3 = Channel_005 + Channel_006 ; computes the sum
this is the end of the commented block *)
```

9.2.3.6 Syntax: multiple channels in a formula

Time Base for Channels in a Formula

If multiple channels are jointly subjected to a function, they must all have the same time base. Since the calculation functions process all a channel's data points one-by-one, the following parameters must be the same for all channels in a formula:

- Sampling Time
- Trigger Affiliation
- Pretrigger
- Measurement Duration

Processing a function on two channels having different sampling rates would require the function to use interpolation. This would vastly increase the demands on the system.

The data must be simultaneous. This means that the channels must be started by the same trigger and with the same pre-trigger duration.

 Note

If it is impossible to avoid having differing sampling intervals for the input channels, there are various functions to use to compensate. Two good possibilities are [ReSample](#) and [Mean](#).

9.2.4 Variables and syntax with Control commands

The following descriptions apply to **imc Inline FAMOS with Control Commands**.

In order to be able to use the advanced functions and options offered by the control commands, activate them. More information is presented in the section: "[Source Text with Control Commands](#)²⁴⁷".

9.2.4.1 Creating a variable

With activated Control Commands, it is additionally possible to generate single values, process vector variables, as well as local and global data arrays. The values each occupy 4 Byte or 8 Byte and can be created in either Integer or Float format.

They are created in the `OnInitAll` block (see: [Types of Variables](#)^[270]).



Note

Notes on precision

- **int**: a pure numerical value (disregarding factor and offset) with 32-bit precision
- **float**: a scaled numerical value (factor and offset are applied)
imc Inline FAMOS: with 52-bit precision

Double-initialization

Avoid double-initialization in the `OnInitAll`-block. However if this is necessary, the first statement defines the type; e.g.

```
OnInitAll
  int pv.x = 0
  ...
  pv.x = 5
End
```

In any subsequent lines, any differing type will not be applied.



Example 1

Process vector

```
OnInitAll
; int creates a variable in Integer format:
int pv.NewEntryA = 0

; without int, the variable is in Float format:
pv.NewEntryB = 0
float pv.NewEntryC = 0
End
```

 Example 2

```
; Executed once at the start
OnInitAll
  v[2] ; local array
  vs      = VectorStatic( BaseTrigger, 4)
  int VarInt  = 1
  VarFloat   = 0.0
  int pv.Var1 = 0
  pv.Var2    = 0
  VChanReal  = SingleValueChannel( BaseTrigger, 1000 )
End

; run upon start of the measurement
OnTriggerStart(BaseTrigger)
  v[1] = pv.Channel_001 ; Channel values upon start
  v[2] = pv.Channel_002

  vs[1] = 0
  vs[2] = 0
  vs[3] = 0
  vs[4] = 0

  VarInt   = 0
  VarFloat = 0.0
  pv.Var1  = 0
  pv.Var2  = 0
End

; Executed during a running measurement
OnTriggerMeasure(BaseTrigger)
  ;...
  if Virt_Bit01
    Virt_Bit01 = 0
    VChanReal  = pv.Channel_001
  end
End

; Executed at the end of every measurement
OnTriggerEnd(BaseTrigger)
  DisplayVar_01 = pv.Channel_001 - v[1] ; difference end - start
  DisplayVar_02 = pv.Channel_002 - v[2] ; difference end - start
End
```

9.2.4.2 Comparison operators

Comparison operators are used for queries within condition statements. Here, we will describe only the `>` operator (Greater), and the other comparison operators `>=`, `=`, `<=`, `<` and `<>` are used in an analogous manner. Comparison operators return results of the BOOL data type, as required for condition statements. Single values and channels can also be used as operands.

Example: "Greater than"-operator

Compares the two operands to check if the first is greater than the second.

```
IsGreater = a > b
```

IsGreater: Result	A: 1st operand
	B: 2nd operand
IsGreater = 1, if A is greater than B.	
IsGreater = 0, if A is less than or equal to B.	



Example

```

OnInitAll
    Value = 0
    VrtBit_01 = 0
End
OnTriggerMeasure( BaseTrigger )
    Value = CurrentValue( Channel_001, 0, 0.0 )
    If Value > 5
        VrtBit_01 = 1
    End
End

```

Combinations of different conditions can be implemented using the operators **AND** (logical And) or **OR** (logical Or); for instance,

```

If VrtBit_01 > 0 AND VrtBit_02 = 0
or
VrtBit01 = DisplayVar_01 < -10 OR DisplayVar_01 > 10

```

9.2.4.3 Defining virtual channels with conditions

- **Deactivated Control Commands:** Virtual channels are always filled with values (according to the clock rate of the parameter channels, etc.).
- **Activated Control Commands:** Virtual channels are only filled with values if they are defined by the currently operable branch of the condition. If the condition in the example below is met (the virtual bit `VrtBit_01` is set), then the virtual channel `VrtChannel_001` is filled with values, otherwise it remains empty:

```

If VrtBit_01 > 0
    VrtChannel_001 = Channel_001 + 10
End

```

If the condition's default branch also defined the virtual channel, then the virtual channel is always filled with values. But then, the virtual channel's particular values depend on the operative branch of the condition, for example,

```

If VrtBit_01 > 0
    VrtChannel_001 = Channel_001 + 10
Else
    VrtChannel_001 = Channel_001 + 20
End

```

If a virtual channel is used in a formula (or if a virtual channel is queried), the virtual channel must be defined in every program branch which leads to the formula or query in question. The following example is only permitted if the default branch of the condition is called. Otherwise, the virtual channel `VrtChannel_001` wouldn't be defined for `VrtBit_01 = 0`.

```

If VrtBit_01 > 0
    VrtChannel_001 = Channel_001 + 10
Else
    VrtChannel_001 = Channel_001 + 20
End
VrtChannel_002 = 2*VrtChannel_001 + Channel_001

```

9.2.4.4 Conditions, case-differentiation and CAN-transmission with channels

If conditions, case-differentiation or CAN-transmission with channels is called in `OnTriggerMeasure`, then the respective current (or most recent) value in the channel is used for the query or transmission. For this purpose, imc Inline FAMOS automatically inserts a `CurrentValue`-function before the querying expression, and then the resulting current value of the channel is retrieved. The following examples illustrate channels subjected to conditions. The behavior of channels in case-differentiation and CAN-transmissions is analogous.

If a physical input channel is used in at least one query (e.g. `If Channel_001 > 0`), the current value of the input channel is noted at the start of `OnTriggerMeasure`. Whenever the physical channel is queried, the single value noted stands in for it:

Coded in imc Inline FAMOS	imc Inline FAMOS automatically transforms this to
<pre>LED_01 = 1 If Channel_001 > 0 If Channel_001 < 0</pre>	<pre>LED_01 = 1 _cv = CurrentValue(Channel_001, 0, 0.0) If _cv > 0 If _cv < 0</pre>

If an expression is queried, e.g. `Channel_001 - 1 > 0`, the current value of the expression `Channel_001 - 1` is automatically generated by imc Inline FAMOS and placed before the query, and the single value generated is then queried:

Coded in imc Inline FAMOS	imc Inline FAMOS automatically transforms this to
<pre>LED_01 = 1 If Channel_001 + 1 > 0 If Channel_001 - 1 < 0</pre>	<pre>LED_01 = 1 _cv = CurrentValue(Channel_001 + 1, 0, 0.0) If _cv > 0 _cv = CurrentValue(Channel_001 - 1, 0, 0.0) If _cv < 0</pre>

If virtual channels are queried, e.g. `If VrtChan_001 > 0`, then the current value in the channel is determined and noted after each definition (with activated control commands, virtual channels can be defined at various points in the source text). If the virtual channel is queried, the query retrieves the noted current value of the virtual channel rather than the channel itself:

Coded in imc Inline FAMOS	imc Inline FAMOS automatically transforms this to
<pre>If VrtBit_01 > 0 VrtChannel_001= Ch01+3 Else VrtChannel_001= Ch01+5 End LED_01 = 1 If VrtChannel_001 > 0 If VrtChannel_001 < 0</pre>	<pre>If VrtBit_01 > 0 VrtChannel_001 = Channel_001 + 3 _cv = CurrentValue(VrtChannel_001, 0, 0.0) Else VrtChannel_001 = Channel_001 + 5 _cv = CurrentValue(VrtChannel_001, 0, 0.0) End LED_01 = 1 If _cv > 0 If _cv < 0</pre>

 Note

Note on CurrentValue-funktion

The `CurrentValue`-functions automatically generated by imc Inline FAMOS reflect the option "last value". However, a high sampling rate can lead to situations where many sample values from a channel are still being processed. If "last value" is the valid option for the `CurrentValue` function, this can lead to values being ignored.

To avoid this, there are other options which can be set for the `CurrentValue` function (e.g. use of maximum value, if multiple values are available for the channel). But then, the `CurrentValue`-function, along with the corresponding setting for the option, must be written into the source text; it isn't added automatically.

**Note****Boolean variables from files**

Boolean (True/False) variables can be queried in **If** conditions without comparison:

```
Switch_A = Channel_01 > 5
If Switch_A
...

```

imc Inline FAMOS allows this, because the compiler makes the result `Switch_A` a boolean variable by comparing `Channel_01 > 5`.

When **importing variables from files**, however, the compiler cannot always clearly recognize them as boolean variables.

Earlier versions of imc Inline FAMOS accepted this, but this could lead to incorrect queries. The following construct leads to an error message in current imc Inline FAMOS versions:

```
Bools= VectorFromFile("Bools.dat")
Switch_B= Bools[2]
...
If Switch_B ; -> Error message
...

```

Therefore **If** queries in current imc Inline FAMOS programs must always be written with comparison, e.g. :

```
Bools= VectorFromFile("Bools.dat")
Switch_B= Bools[2]
If Switch_B = 1 ; -> OK
...

```

This change may cause old experiments that worked with earlier imc STUDIO versions to generate error messages in current versions. In this case the imc Inline FAMOS code must be changed according to the last example.

9.2.4.5 Example - Simplified motor simulation

The use of various control commands is illustrated (in imc Online FAMOS) below in reference to a simplified motor simulation. The voltage and current signals from a motor during run-up, normal operation and wind-down are simulated (for 5 s each). The current state of the simulation can be frozen using a Pause-bit.

```
-----
; In the OnInitAll block, initializations are performed prior to the
; first measurement. All single value variables used in the formulas
; must be initialized here.
-----
OnInitAll
    Status      = 0          ; Initialization of single value variables
    Counter     = 0
    Data        = 0
    PauseOld    = 0
    PauseNew    = 0
End

-----
; The block OnAlways is run constantly. The Pause-function is
; implemented here. If you click on the Pause bit, the timer is stopped.
; If the Pause-bit is currently deactivated, clicking on it starts the
; periodic timer. After 10ms, the timer ticks at 10ms intervals.
-----
OnAlways
    ; PauseNew is the 1st virtual bit
    if PauseNew <> 0 AND PauseOld = 0          ; Pause-bit is clicked on
        RecordText ( "Pause!!!" )             ; output text
        StopTimer ( 1 )                       ; stop timer
    end
    if PauseNew = 0 AND PauseOld <> 0          ; Pause-bit is clicked off
        RecordText ( "Pause finished!!!" )     ; output text
        StartTimerPeriodic ( 1, 0.01 , 0.01 ) ; start periodic timer
    end
    PauseOld = PauseNew
End
```

```

;-----
; The OnTriggerStart block is executed at the start of
; the measurement. When the measurement starts, a periodic timer is
; set up. After 10ms, the timer ticks at 10ms intervals.
;-----
OnTriggerStart ( BaseTrigger )
    LED_01 = 1 ; switch on LED_01
    StartTimerPeriodic ( 1, 0.01 , 0.01 ) ; start timer
End

;-----
; The OnTriggerEnd block is executed at the end of a measurement.
; Here, the timer is stopped and LED_01 switched off.
;-----
OnTriggerEnd ( BaseTrigger )
    LED_01 = 0 ; switch off LED_01
    StopTimer ( 1 ) ; stop timer with the ID 1
End

;-----
; The OnTriggerMeasure block runs constantly during measurement.
; The virtual channel Voltage is created under the control of a timer,
; and the virtual channel Current is computed from the channel Voltage.
;-----
OnTriggerMeasure (BaseTrigger )
    Voltage = channel_001 * 0 + Data
    _x = Voltage + sin ( sawtooth ( Voltage, 0, 1, 10000 ) )
    rpm = upper ( 0, filt1p ( _x, 0, 0, 3, 1 ) * 60 + _x * 20 )
    _y = diff ( _x )
    Current = 10*filt1p ( _y, 0, 0, 2, 2 )
    if Voltage > 5 ; queries virtual channel Voltage. If the
        ; current value is greater than 5,
        LED_05 = 1 ; switch on LED_05
        LED_06 = 0 ; switch off LED_06
    else ; otherwise (current value in channel Voltage <= 5)
        LED_05 = 0
        LED_06 = 1
    end
End

;-----
; The OnTimer block is run upon every click of Timer 1. The periodic
; timer, Timer 1, was set up either in OnTriggerStart or OnAlways
; using the function StartTimerPeriodic. In the OnTimer block,
; the numerical values for the virtual channel Voltage are computed.
;-----
OnTimer ( 1 ) ; OnTimer-Block for Timer 1
    switch Status ; case differentiation based on the
        ; single value Status
        case 0 ; case: Status = 0
            Data = Data + 0.02 ; compute the data
            Counter = Counter + 1 ; increment counter
            if Counter > 500 ; query counter
                Counter = 0 ; re- initialize counter
                Status = 1 ; set Status
            end ; finish condition
        end ; finish case-differentiation
        case 1 ; case: Status = 1
            Counter = Counter + 1
            if Counter > 500
                Counter = 0
                Status = 2
            end
        end
        case 2 ; case: Status = 2
            Data = Data - 0.02
            Counter = Counter + 1
            if Counter > 500
                Counter = 0
                Status = 0
            end
        end
        default ; default treatment for Status,
            ; if Status is not equal to 0, 1 or 2
        end
    end
End

```

9.2.5 Types of variables

The following is a list of the usual variable types which you can define in imc Inline FAMOS. Special types of variable such as CAN-structures, controller-structures, ... are not mentioned here. These are presented in the respective chapters.

Virtual channels ()

Description	Example	Syntax	Definition
Virtual channel, (Datenfeld equidistant)	<code>Virt = Channel_001</code>	various functions	Without control command Otherwise: OnTriggerMeasure
Data array global	<code>vs = VectorStatic (BaseTrigger, 10)</code>	Array = <code>VectorStatic</code> (Trigger#, Size)	OnInitAll
Data array dynamic	<code>vsChan = SingleValueChannel (BaseTrigger, 1000)</code>	Array = <code>SingleValueChannel</code> (Trigger#, DataRate) various additional functions	OnInitAll
Vector series dynamic	<code>vChan = VectorChannel (BaseTrigger, 100, 5)</code>	Array = <code>VectorChannel</code> (Trigger#, DataRate, Size)	OnInitAll
Vector series equidistant	<code>Virt_fft = FFT (Channel_001, 0, 1024)</code>	various functions	Without control command Otherwise: OnTriggerMeasure

Local channels ()

Description	Example	Syntax	Definition
Local channel	<code>_Local_Virt = Channel_001</code>	various functions	Without control command Otherwise: OnTriggerMeasure
Local vector, Data array local	<code>v[10] int v[10]</code>	Array[Size] various additional functions	OnInitAll

Local single value variables ()

Description	Example	Syntax	Definition
Single value as Float	<code>Lokal_Var1 = 4 Lokal_Var2 = 0xA5 ; hex</code>	Variable = 0	Without control command Otherwise: OnInitAll
Single value as Integer	<code>int Var1= 1 int Var4= 0xA5 ; hex</code>	<code>int</code> Variable = 0	OnInitAll
Single value as Float	<code>Var2= 0.0 float Var3= 0 Var4= 0xA5 ; hex</code>	VariableA = 0 <code>float</code> VariableB = 0	OnInitAll

pv-variables (single value;) (only in imc Online FAMOS Professional)

Description	Example	Syntax	Definition
Single value as Integer	<code>int pv.Var1= 1</code> <code>int pv.Var4= 0xFFFF ; hex</code>	<code>int pv.Variable = 0</code>	OnInitAll
Single value as Float	<code>float pv.Var2= 0.0</code> <code>pv.Var3= 0</code> <code>pv.Var4= 0xFFFF ; hex</code>	<code>pv.VariableA = 0</code> <code>float pv.VariableB = 0</code>	OnInitAll

Local texts ()

Description	Example	Syntax	Definition
Text	<code>Text = "Hello"</code>	various functions	

9.2.5.1 Local vectors

Local Vectors in OnInitAll-Block

Local vectors can be created using the function `VectorFromFile` and now also in the `OnInitAll` block:

```
OnInitAll
  Vector = VectorFromFile( "Vector_01.DAT" )
  ; or
  int sv1[VectorLength]
  ; or
  sv2[VectorLength]
End
```

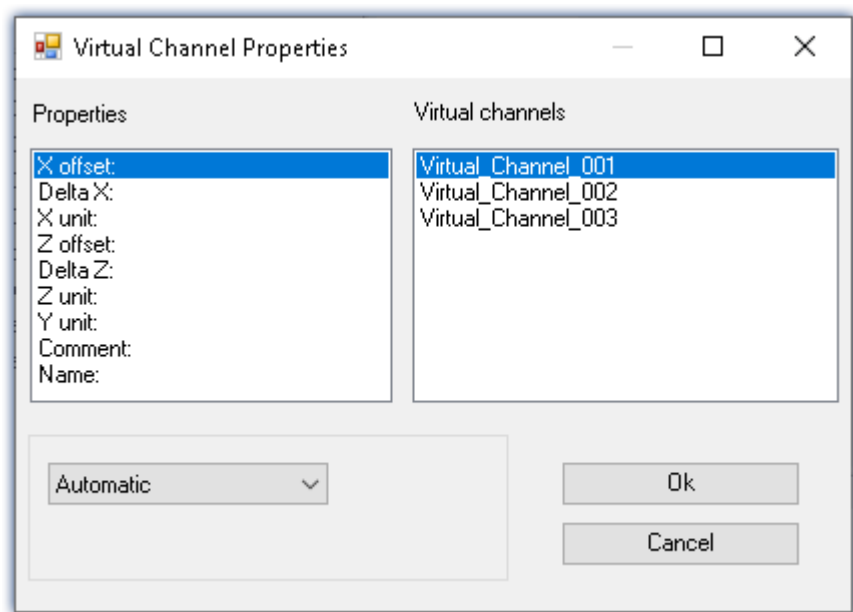
- The vector elements are initialized at 0.
- The vector length of `sv` must be selected appropriately. Make sure to use only possible vector indexes, especially for variable vector indexes.
`sv[1]...sv[vector length]`
- Local vectors can be used in synchronous tasks, too.
- Elements of local vectors can be used on the left or right side of an expression:
`Virt_Bit01 = sv[1]`
or
`if sv[2] > 0`
or
`sv[1] = sv[1] + 1`
- The elements of local vectors can be addressed with variable indexes, e.g.:
`sv[Index + 2] = sv[Index + 1]`
- Functions which return streaming vectors can be assigned to static vectors:
`sv2 = FFT(...)`
- The function `GetSampleCount` of group Control functions returns for local vectors the length of the vector. This is useful, if the a local vector has been created with the function `VectorFromFile`. Thus the possible vector indexes for this vector can be checked.

9.2.6 Properties of virtual channels

Already defined virtual channels can subsequently be assigned certain properties. A virtual channel's name, comment and units can be set. If any virtual channels are available as vectors, it is additionally possible to set special vector properties.

To do this, open the dialog: "Virtual Channel Properties" via the [context menu](#)^[256] in the Editor, or via the menu item "Edit" > "Properties" (via the menu item applies only to imc Online FAMOS):

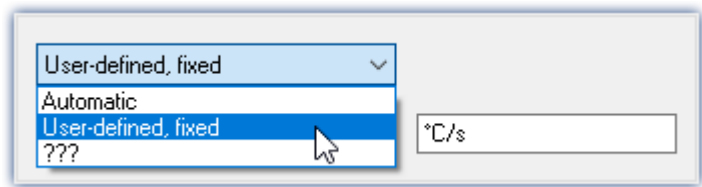
Menu item	Description
 Properties	Opens the Properties window. There, the properties of virtual channels can be edited subsequently (see " Virtual channels " ^[258]).



The virtual channels appear on the right side of the window. Multiple selection is possible for assigning the same properties to several channels.

The various properties are displayed on the left side of the list. You can define one property at a time for the selected virtual channel(s) on the right.

The options for the selected property are made in the drop-down list boxes at the bottom of the window. The default setting for all properties is *automatic*. To define a property select *User-defined* in the list box. Then, a text box will appear in which you can enter the value desired.



The properties of virtual channels can only be set in this dialog. If you wish to change properties which were already set or to make settings for new virtual channels, call the dialog again.

Property	Description
X-offset	The offset in the x-direction can be set to a fixed value for purposes of display in the curve window. Otherwise, the value is automatically determined. This property can only be set for vectors .

Property	Description
Delta-X	The distance between two sample values in the x-direction can be set to a fixed value for purposes of display in the curve window. Otherwise, the value is automatically determined. This property can only be set for vectors .
Defining X-axis units	For the display of a virtual channel in the curve window, the units of the X-axis can either be determined automatically or given a fixed setting by the user. It is best to specify only SI-units. Do not specify any orders of magnitude such as milli- and kilo- (exception: kg). The same guidelines for units are valid as for defining the units of input channels. For channels whose X-axis is the time-axis, it is recommended to leave the X-axis unit on automatic. In that case, the unit displayed will be "s" for seconds.
Z-offset	The z-offset is the z-value given for the "front" curve in a 3D curve window. The z-offset can either be determined automatically or given a fixed setting by the user. This property can only be set for vectors !
Delta-Z	Delta-Z is the distance between two adjacent curves in the curve window. This property can only be set for vectors !
Comment	Comments can be added to virtual channels in the same way as with normal input channels. You can use any characters in this field - you can also leave it blank.
Name	Instead of the virtual channel's name which you used when defining it, you can specify a different name here. This saves you the work of renaming all names in the entire source text, for example for the purpose of the display in the curve window. It also makes sense to insert a shorter name in the formulas while stating the whole name here. The names of all channels must be unique within the system. The same limitations apply for constructing a name as for the input channels.

9.2.7 Calculation examples

9.2.7.1 Determining the RMS of a mixed signal's alternating component

The root-mean-square (RMS) of the signals alternating components is to be determined; the signal contains a direct as well as an alternating component.

For example, it is possible to determine the RMS of a noise signal in an amplifier which has an offset voltage. Another use would be for determining the noise component in a supply current.

Problem background:

The signal $u(t)$ consists of direct and alternating components and can be represented as $u(t) = u_{-}(t) + u_0$ where u_0 is the signal's direct component.

If one tries to find the alternating component's RMS by using the equation

$$U_{-} = \sqrt{\frac{1}{T} \int_{t-T}^T [u(t) - \text{Meanvalue}(u(t))]^2 dt}$$

directly, T being the time over which the data are collected, an error message will be generated when the source text is compiled. The reason is that the RMS, a single value, is to be subtracted from $u(t)$, which is a vectored data stream.; in other words, the data types don't match.

Solution:

If the initial equation $u(t) = u_{\sim}(t) + u_0$ is incorporated into the definition of the RMS, and it is noted that the mean of the alternating quantity is eliminated, it follows that the sum of the RMS components' squares $U^2 = U_{\sim}^2 + U_0^2$ equals the square of their sum. This makes it possible to determine $U_{\sim}$ as the difference of two data sets of matching type.

**Example**

Letting Ch_01 designate the complete signal, the RMS of the alternating component can be computed as

```
Rms_noise= Sqrt(mean(Ch_01*Ch_01, 1000,1000) - mean(Ch_01, 1000, 1000)^2)
```

9.2.7.2 Determining one or more frequencies' contributions to a signal

Purpose:

The 150 Hz contribution to a non-sinusoid current of mains frequency is to be displayed as a curve plotted over time. A digital output could additionally be set in response to this contribution's exceeding a threshold value.

Problem background:

If the FFT function is applied directly to a measurement, the resulting frequency spacing is given by

$$\Delta f = \frac{f_T}{N}$$

where f_T is the sampling rate and N is the number of data points used, which must be expressed as a power of two (e.g., 1024 = 2¹⁰). If, for instance, the sampling rate 1 kHz and N=1024 are specified, the frequency spacing is 0.9765625 Hz. This means that it is impossible to determine the 150 Hz line exactly.

Solution:

One could extract exactly 1000 values from the continuous stream of measurement data. This excerpt can afterwards be extended to 1024 points by resampling. The resulting, resampled data set then has a sampling frequency of 1.024 kHz, and the 150th spectral line is located exactly at 150 Hz.

**Example**

Declare `Current` as the measurement channel to be analyzed, which is sampled at a rate of 1 kHz and whose 150 Hz spectral line is to be charted over time. One appropriate source text would be the following:

```
_I_Channel=VectorizeAndSkip(Current ,1000 , 0 )  
_I_Res=ReSample(_I_Channel ,1024 )  
Spec = FFT(_I_Res, 0, 1024)  
I_150 = VValueAtXValue(Spec, 150)
```

With the function `VectorizeAndSkip`, 1000 samples are extracted from the continuous data stream designated by the variable "Current". A data "skip" is not performed in this instance, as indicated by the zero as the last parameter in the function call.

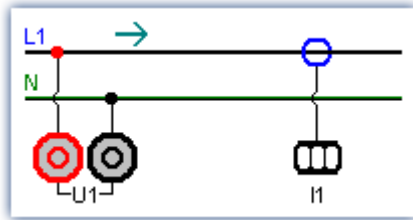
The assigned name, `_I_Channel`, begins with an underline, identifying it as an internal variable. This isn't strictly necessary, but it helps to conserve memory.

Next, the `ReSample` function is employed to produce 1024 samples. Now the FFT calculation can be performed. The function `VValueAtXValue` makes it possible to extract the Y-value at a specified X-value. In this case, the spectral line #150 is at 150 Hz, since the frequency spacing is 1 Hz, and it is designated by the variable `I_150`. All available functions such as triggering in response to threshold crossing, can be applied to this variable.

9.2.7.3 Power measurements

Here follows the description for the Single-phase, Two-phase- and Three-phase-power measurement.

9.2.7.3.1 Single-phase measurement power1



Single-phase power measurement

Effective value (root-mean-square)

$$y = \sqrt{\frac{1}{T} \int_T x^2 dt}$$

The effective value is the square root of the mean-square of the input signal.

Instantaneous power

$$p = u \cdot i$$

The operation Instantaneous Power supplies the product of two respective sampling values.

Active power

$$P = \frac{1}{T} \int_T (u \cdot i) dt$$

The active power is the average of the instantaneous powers during an averaging interval. This describes the actual power recorded by the user.

Apparent power

$$P_S = U \cdot I$$

The apparent power is the product of the root-mean-squares for current and voltage. The root-mean-squares are calculated according to the algorithm described under the operation effective value (root-mean-square).

Reactive Power

$$Q = \sqrt{S^2 - P^2}$$

The reactive power is the geometric difference between apparent and active power. Apparent and active power are calculated according to the above algorithm. The reactive power is the portion of the apparent power not recorded by the user.

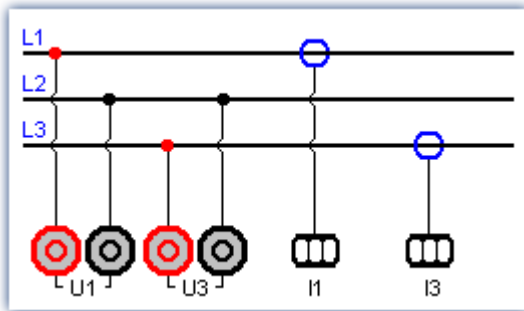
Power Factor

$$\cos(\varphi) = \frac{P}{S}$$

The power factor is the ratio between active power and apparent power. Active power and apparent power are calculated according to the above algorithm. The power factor corresponds to the cosine of the phase shift between current and voltage.

9.2.7.3.2 Two-phase measurement Power2 (ARON)

The power can be determined from only two phase voltages and two conductor currents, assuming that the zero-wire of the three-phase circuit has no current. For that balanced load the third quantity is thus clearly defined. This type of power measurement will be referred to below as an ARON measuring circuit. The ARON measuring circuit is often the only way to measure the power on a user whose star point is not accessible.



ARON measuring circuit

In the triangular circuit, the connection of the measurement unit results analogous to the illustrated star circuit.

Instantaneous power

The operation instantaneous power (ARON) supplies the sum of the products of two respective sampling values.

$$p = u_1 \cdot i_1 + u_3 \cdot i_3$$

Active power

The active power is the average of the instantaneous powers during an averaging interval.

$$P = \frac{1}{T} \int (u_1 \cdot i_1 + u_3 \cdot i_3) dt$$

This describes the actual power recorded by the user.

Apparent power

The apparent power results from the sum of the products of the effective values (root-mean-squares) for the currents and voltages during the individual phases.

$$S = \frac{\sqrt{3}}{2} \int (U_{rms1} \cdot I_{rms1} + U_{rms3} \cdot I_{rms3}) dt$$

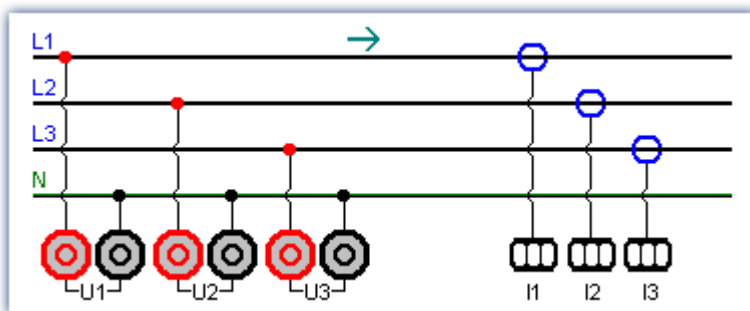
It is required that the symmetry of the supply circuit (three equal voltages with phase shifts of 120°) be stable. The root-mean-squares are calculated according to the algorithm under the operation effective value (root-mean-square).

Reactive power and Power Factor

See [Single-phase power measurement](#) ²⁷⁵

9.2.7.3.3 Three-phase measurement with N: Power3

Measurements on the three-phase circuit **with** N result according to the following diagram. For measurements on the three-phase circuit **without** N use [ARON measuring circuit](#) ²⁷⁵.



with N (Y-connection)

Instantaneous power

The operation instantaneous power supplies the sum of the products of two respective sampling values.

$$p = u_1 \cdot i_1 + u_2 \cdot i_2 + u_3 \cdot i_3$$

Active power

The active power is the average of the instantaneous powers during an averaging time.

$$P = \frac{1}{T} \int (u_1 \cdot i_1 + u_2 \cdot i_2 + u_3 \cdot i_3) dt$$

This describes the actual power recorded by the user.

Apparent power

The apparent power results from the sum of the products of the effective values (root-mean-squares) for the currents and voltages during the individual phases.

$$S = U_{rms1} \cdot I_{rms1} + U_{rms2} \cdot I_{rms2} + U_{rms3} \cdot I_{rms3}$$

The root-mean-squares are calculated according to the algorithm under the operation effective value (root-mean-square).

Reactive power and Power Factor

See [Single-phase power measurement](#)  275

9.2.8 Information and tips

9.2.8.1 Transferring a configuration

The possible configurations depend on how the device is equipped:

When imc Online FAMOS configurations are copied to other devices, their behavior may well be altered. This also applies when copying the imc Online FAMOS configuration to imc Inline FAMOS. For instance, an assignment `LED_01 = Greater(Channel_001, 5)` will cause LED_01 to be switched on, if the device has LEDs, otherwise a virtual channel by that name will be created.

The same goes for beepers, DAC-outputs and DIO-bits.

9.2.8.2 Virtual channels and the trigger machine

Virtual channels don't appear along with the channel selections in the Trigger dialog. Thus, a virtual channel cannot be triggered directly. To trigger the value of such a computed channel, the pertinent event must be detected in imc Inline FAMOS, e.g. using the function `Greater`. The event sets a virtual bit, which in turn can be used by the trigger machine:

```
Virtual_Bit01 = Greater(Temp_difference, 5)
```

9.2.8.3 Supplemental files

Supplemental file such as characteristic curves for functions, must be imported to the experiment.



Reference

On this section, see the description for: "Setup" > "Ribbon" > "[Supplemental files](#)"  120.

9.2.8.4 Comments for class-counting functions

To apply class-counting functions, a number of parameters need to be specified. However, the data presented in a curve window are not sufficient to determine all of these parameters. For this reason, the parameters for class-counting functions, in particular, are automatically inserted in the comment for the class-counting result. To save space, code-words corresponding to the various parameter settings options are used. The comment is produced when a syntax-check is successfully performed. The class-counting function names all begin with "CI", e.g. CIRainflow.

With class-counting functions which return a matrix and a residue as their results, the automatic comment is produced only for the virtual channel containing the matrix.

If you write your own comment for a virtual channel which results from a class-counting function (Properties dialog in imc Inline FAMOS), the comment will be displayed in the curve window. The comment which would otherwise be made automatically is then no longer available.

List of possible code-words and their meanings for class-counting function comments:

Code-word	Definition
OFA:	Comment intro for any class-counting function
RowMin	Lower limit of bin range (for rows)
RowMax	Upper limit of bin range (for rows)
ColMin	Lower limit of bin range (for columns)
ColMax	Upper limit of bin range (for columns)
Hyst	Hysteresis width
RowCI	Axis arrangement: X-axis is target class
RowAmplitude	Axis arrangement: X-axis is amplitude
RowCIStart	Axis arrangement: X-axis is start class
RowMeanValue	Axis arrangement: X-axis is mean value
UnitCI	Result unit: classes
UnitInput	Result unit: as input channel
EndCLOpen	Open-ended outer bins
EndCIClosed	Closed outer bins
OptAlgBasic	Calculating method: base algorithm
OptAlgClor	Calculating method: with Clormann-Seeger correction
RowUnitStr	Unit (row-wise); in the comment: <Unit>
ColUnitStr	Unit (column-wise); in the comment: <Unit>
Level	Reference level
RowRevs	Axis arrangement: rpm in X-axis
RowNotRevs	Axis arrangement: rpm in Z-axis
Func	Function name of class-counting function

After the code words RowMin, RowMax, ColMin and ColMax, the corresponding value is stated, e.g. RowMax 1.00000E+2.

After the code words RowUnitStr and ColUnitStr, the unit is stated. The unit text is framed by "greater than" and "less than" symbols, e.g. RowUnitStr <Unit>.

9.2.8.5 Nesting depth, maximum stack

For conditions, etc., the maximum nesting depth is 90.

If (...)

 If (...)

 If (...) ...

With formulas, it is also possible to keep up to 90 expression elements.

For example: Result = a+b comprises 4 elements: a, b, + and "Result"

Note that there is **no** improvement of performance if the entire expression is written in one line. Instead it only reduces clarity!

9.2.9 imc Inline FAMOS function reference

This document contains the reference for the imc Online FAMOS and imc Inline FAMOS functions.

The functions are sorted in alphabetical order.

Functions for imc Online FAMOS or imc Inline FAMOS with control commands are listed in one group.

9.2.9.1 Operators and symbols

(+) Addition

Addition: Basic arithmetic

Sum = Summand1 + Summand2

If one of the two summands is a real number, the result is also a real number. In the addition of two integers, the result is an integer.

No handling mechanism is provided against potential data overflow; the user is required to take precautions for this possibility.

(-) Subtraction

Subtraction: Basic Arithmetic

Difference = Subtrahend - Minuend

If the subtrahend or the minuend is a real number, the result is a real number. In the subtraction of one integer from another integer, the result is an integer.

No handling mechanism is provided against potential data overflow; the user is required to take precautions for this possibility.

(*) Multiplication

Multiplication: Basic Arithmetic

Product = Factor1 * Factor2

If either of the two factors is a real number, the result is a real number. In the multiplication of two integers, the result is an integer.

No handling mechanism is provided against potential data overflow; the user is required to take precautions for this possibility.

(/) Division

Division: Basic Arithmetic

Quotient = Numerator / Denominator

With division, the result is always a real number.

For division between integers, please use the operator `iDiv` .

(^) Power

Power: Base raised to exponent;

Result = Base ^ Exponent

0^0 is defined to be 1.

(%) Modulus

Modulus: Remainder from division of two variables

Result = Numerator % Denominator



Example

```
Result = 10 % 3 ; Result = 1
```

If either the numerator or the denominator is a real number, the result is a real number.

When using the Modulo operator on two integers, the result is an integer. See also: [iDiv](#) .

(=) Equal sign (Assignment)

Equal sign: The "=" (equal sign) is used for making assignments to local and virtual channels.

Character: "=" (equal sign)



Example

```
_LocChan = Data + 1  
VirtChan = _LocChan * 2
```

(=) Equal operator (Comparison)

"Equals"-operator. Compares the two operands for equality.

Prerequisite:

Only usable with control commands!

IsEqual = A = B

IsEqual: Result

A: 1st operand

B: 2nd operand

IsEqual = 1, if A = B.

IsEqual = 0, if A is not equal to B.



Example

```
OnTriggerMeasure( BaseTrigger )  
  Value = CurrentValue( Channel_001, 0, 0.0 )  
  If Value = 1  
    VirtChannel = Channel_001 + 10  
  Else  
    VirtChannel = Channel_001 + 5  
  End  
End
```

(<>) Inequality operator

Inequality-operator. Compares the two operands for inequality.

Prerequisite:

Only usable with control commands!

IsUnequal = A <> B

IsUnequal: Result

A: 1st operand

B: 2nd operand

IsUnequal = 1, if A is not equal to B.

IsUnequal = 0, if A = B.

(<) Less than operator

"Less than"-operator. Compares the two operators to check whether the first is less than the second.

Prerequisite:

Only usable with control commands!

IsLess = A < B

IsLess: Result

A: 1st operand

B: 2nd operand

IsLess = 1, if A is less than B.

IsLess = 0, if A is greater than or equal to B.



Example

```
OnTriggerMeasure( BaseTrigger )
  Value = CurrentValue( Channel_001, 0, 0.0 )
  If Value < 1
    VirtChannel = Channel_001 + 5
  Else
    VirtChannel = Channel_001 + 10
  End
End
```

(<=) Less than or equal operator

"Less than or equal to"-operator. Compares the two operands to check if the first is either less than or equal to the second.

Prerequisite:

Only usable with control commands!

IsLessEqual = A <= B

IsLessEqual: Result

A: 1st operand

B: 2nd operand

IsGreater = 1, if A is less than or equal to B.

IsGreater = 0, if A is greater than B.

(>) Greater than operator

"Greater than"-operator. Compares the two operands to check if the first is greater than the second.

Prerequisite:

Only usable with control commands!

IsGreater = A > B

IsGreater: Result

A: 1st operand

B: 2nd operand

IsGreater = 1, if A is greater than B.

IsGreater = 0, if A is less than or equal to B.

(>=) Greater than or equal operator

"Greater than or equal to"-operator. Compares the two operands to check if the first is either greater than or equal to the second.

Prerequisite:

Only usable with control commands!

IsGreaterEqual = A >= B

IsGreaterEqual: Result

A: 1st operand

B: 2nd operand

IsGreaterEqual = 1, if A is greater than or equal to B.

IsGreaterEqual = 0, if A is less than B.

() Opening closing bracket

Opening parenthesis: The parentheses "(" and ")" are used in nested expressions and with functions.

Character: "(" (opening parenthesis)



Example

```
VirtChannel1 = 2 * ( Data + 1 )
VirtChannel2 = Max( Data, 5, 10 )
```

(;) Semicolon

Semicolon: A semicolon ";" is used to precede remark comments. Any information to the right of a semicolon is ignored by the program.

Character: ";" (Semicolon)



Example

```
VirtChan = Data + 1 ; first virtual channel
```


9.2.9.2 Constants

Prerequisite:	
Only available in: imc Online FAMOS!	
imc Online FAMOS has several predefined constants at its disposal. These constants can be used as single values. No new value can be assigned to them, however, and they are not registered in the variable list.	
Constants	Description
e	Euler number e
InDegr	constant for converting radians to degrees
InRad	constant for converting degrees to radians
PI	circle constant PI
PI2	2 * circle constant PI

9.2.9.3 Functions (alphabetic)

9.2.9.3.1 A

ABCRating

ABC-weighting: Conducts either an A-, B- or C-frequency evaluation of a signal according to DIN IEC 651. In addition, subsequent time-rating (moving RMS-value with exponential averaging) and resampling can be performed.

ABCSignal = ABCRating(Signal, Frequency weighting, TimeConstant, Reduction[, Mode])

ABCSignal: Rated signal	TimeConstant: Time constant for averaging
Signal: Signal to be evaluated	In imc Online FAMOS:
Frequency weighting: The signal's frequency rating	≥ 0.0: Time constant is defined by the user in s e.g. 0.125 for FAST-weighting, 1.0 for SLOW-weighting
1: A-rating	In imc Inline FAMOS:
2: B-rating	-1: Fast (0.125s)
3: C-rating	-2: Slow (1s)
	-3: Pulse
	-4: Peak
	-5: RMS in interval
	-6: RMS from start
	≥ 0.0: Time constant is defined by the user in s
	Reduction: Reduction depending on the value of "Mode"
	Mode: [Optional] Interpretation of the parameter "Reduction"
	1: The parameter "Reduction" is the reduction factor expressed in value count
	2: The parameter "Reduction" is the sampling interval of the result

The optional "Mode" parameter applies only to imc Online FAMOS. It does not exist for imc Inline FAMOS and is interpreted as 1.

When the value 1 is specified for "Mode", the "Reduction"-value is interpreted as the reduction factor (RF). The function `ABCRating` reduces the count of sample values by the reduction factor. Additionally, the data rate is reduced by the factor "Reduction".

Reduction = 1: After each input value, a result value is calculated.

Reduction = 2: After every other input value, a result value is calculated.

Reduction = n: After every n-th input value, a result value is calculated.

The reduction factor must be an integer > 0.

When the value 2 is specified for "Mode", the "Reduction"-value is interpreted as the target sampling interval of the result channel. The target sampling interval must be the result of dividing the input channel's sampling interval by an integer.

Example: For an input channel sampling interval of 100 ms (0.1), the target sampling interval could be 500 ms (0.5), for example. This would represent a reduction factor (RF) of 5.

The 3th parameter "Mode" is optional. In the absence of any specification, the default value 1 is applied.

For a time constant = 0, no subsequent time weighting is performed. The reduction factor must in this case be exactly 1.

The A-rating corresponds to IEC 61672-1, 1st edition, 2002-05, Class1 and DIN IEC 651, 1981, class 0.



Example 1

```
SignalA = ABCRating ( Signal, 1, 0.125, 1000 )
```

The signal is subjected to A-rating. The frequency-weighted signal is then time-weighted with a time constant of 0.125 s (FAST) and resampled with a factor of 1000. The signal originally has a sampling frequency of 20 kHz, the A-rated result has a sampling frequency of only 20 Hz.



Example 2

with optional parameter

```
SignalB = ABCRating ( Signal, 2, 1.0, 0.05, 2 )
```

The signal is subjected to B-rating. The frequency-weighted signal is then time-weighted with a time constant of 1.0 s (SLOW) and resampled. The signal originally has a sampling frequency of 20 kHz, the B-rated result has a sampling frequency of only 20 Hz.

Abs

Absolute value of input channel

a = `Abs( b )`

a: Results channel

b: Input channel

General information about the Accu* functions

The functions determine result over the entire measurement duration.

The result is a virtual channel or a single value. If the result is a virtual channel, this virtual channel is filled with exactly one result value at the end of the measurement.

It is possible to perform calculations on the results of the Accu-functions. Toward this end, be aware that these results of the Accu-functions are only available once the measurement is finished. For this reason, the results of the Accu-functions can not be taken into account in virtual channels which are calculated during the measurement. However, it is also possible to generate new virtual channels from the results of the Accu-functions, which are only available at the end of the measurement.



Example

Example of including results of the Accu-functions in a calculation

Determining the input signal's minimum and maximum values and sample count over the entire measurement duration. The results are subjected to calculations afterwards.

```
AccuMin1 = AccuMin( signal )
AccuMax1 = AccuMax( signal )
AccuLength1 = AccuLength( signal )
Res1 = (AccuMax1 - AccuMin1) * 10
Res2 = AccuLength1 / 2
```



Example

Example of an incorrect application

Determining an input channel's count of samples throughout the entire measurement duration.

The result of the AccuLength-function is determined at the end of the measurement. This result is generated too late to be used in generating calculated virtual channels during the measurement.

The virtual channel Res returns 0 as its result.

```
AccuLength1 = AccuLength( signal )
Res = signal*0 + AccuLength1
```

AccuLength

Count of samples over the entire measurement

Result = AccuLength(Signal)

Result: Results channel

Signal: Input signal

Remarks:

The function determines the count of samples in the input signal over the entire measurement.

The result is a virtual channel or a single value. If the result is a virtual channel, this virtual channel is filled with exactly one result value at the end of the measurement.



Example

Determining the input signal's minimum and maximum values and sample count over the entire measurement duration.

"*signal*" is the input channel.

```
AccuMin1 = AccuMin( signal )
AccuMax1 = AccuMax( signal )
AccuLength1 = AccuLength( signal )
```

AccuMax

Maximum value of entire measurement

Result = `AccuMax`(`Signal`)

Result: Results channel

Signal: Input signal

Remarks:

The function determines the maximum value of the input signal over the entire measurement duration.

The result is a virtual channel or a single value. If the result is a virtual channel, this virtual channel is filled with exactly one result value at the end of the measurement.



Example

Determines the minimum, maximum and mean value of the input signal over the entire measurement duration.

"*signal*" is the input channel.

```
AccuMin1 = AccuMin( signal )
```

```
AccuMax1 = AccuMax( signal )
```

```
AccuMean1 = AccuMean( signal )
```

AccuMean

Mean value over entire measurement

Result = `AccuMean`(`Signal`)

Result: Results channel

Signal: Input signal

Remarks:

The function determines the mean value of the input signal over the entire measurement duration.

The result is a virtual channel or a single value. If the result is a virtual channel, this virtual channel is filled with exactly one result value at the end of the measurement.



Example

Determines the minimum, maximum and mean value of the input signal over the entire measurement duration.

"*signal*" is the input channel.

```
AccuMin1 = AccuMin( signal )
```

```
AccuMax1 = AccuMax( signal )
```

```
AccuMean1 = AccuMean( signal )
```

AccuMin

Minimum value of entire measurement

Result = `AccuMin`(`Signal`)

Result: Results channel

Signal: Input signal

Remarks:

The function determines the minimum value of the input signal over the entire measurement.

The result is a virtual channel or a single value. If the result is a virtual channel, this virtual channel is filled with exactly one result value at the end of the measurement.



Example

Determines the minimum, maximum and mean value of the input signal over the entire measurement duration.

"*signal*" is the input channel.

```
AccuMin1 = AccuMin( signal )
```

```
AccuMax1 = AccuMax( signal )
```

```
AccuMean1 = AccuMean( signal )
```

AccuRMS

RMS-value over the entire measurement

Result = `AccuRMS`(`Signal`)

Result: Results channel

Signal: Input signal

Remarks:

The function determines the RMS-value of the input signal over the entire measurement duration.

The result is a virtual channel or a single value. If the result is a virtual channel, this virtual channel is filled with exactly one result value at the end of the measurement.



Example

Determines the minimum, maximum and RMS-value of the input signal over the entire measurement duration.

"*signal*" is the input channel.

```
AccuMin1 = AccuMin( signal )
```

```
AccuMax1 = AccuMax( signal )
```

```
AccuRMS1 = AccuRMS( signal )
```

AccuStDev

Standard deviation over the entire measurement

Result = `AccuStDev`(`Signal`)

Result: Results channel

Signal: Input signal

Remarks:

The function determines the standard deviation of the input signal over the entire measurement duration.

The result is a virtual channel or a single value. If the result is a virtual channel, this virtual channel is filled with exactly one result value at the end of the measurement.



Example

Determines the minimum, maximum and standard deviation of the input signal over the entire measurement duration.

"*signal*" is the input channel.

```
AccuMin1 = AccuMin( signal )
```

```
AccuMax1 = AccuMax( signal )
```

```
AccuStDev1 = AccuStDev( signal )
```

Acos

Inverse cosine of input channel

a = `Acos`(`b`)

a: Result; in radians

b: Input channel

AND

Logical AND operator applied to A and B.

Prerequisite:

Only usable with control commands!

LogicalAnd = A AND B

LogicalAnd: Result

A: 1st operand

B: 2nd operand

LogicalAND = 1, if A is nonzero and B is nonzero.

LogicalAND = 0, if A is nonzero and B = 0.

LogicalAND = 0, if A = 0 and B is nonzero.

LogicalAND = 0, if A = 0 and B = 0.

The And-operator may only be used on operands of type BOOL. A result of type BOOL is returned by the operators <, <=, >, >=, =, <>, AND, OR and NOT.



Example

```
OnTriggerMeasure( BaseTrigger )
  Value1 = CurrentValue( Channel_001, 0, 0.0 )
  Value2 = CurrentValue( Channel_002, 0, 0.0 )
  If Value1 > 0 AND Value2 > 0
    VirtChannel_001 = Channel_001 + 10
  Else
    VirtChannel_001 = Channel_001 + 5
  End
End
```

Asin

Inverse sine of input channel

a = Asin(b)

a: Result; in radians

b: Input channel

Atan2

Inverse tangent of b/c

a = Atan2(b, c)

a: Result

b: Numerator

c: Denominator

9.2.9.3.2 B

BitAnd

Bit wise AND connection of b and c

a = **BitAnd**(**b**, **c**)



Example

```
lownibble = BitAnd( DIO_Port01, 0x0f )
```

Extracts the DIO-Port's bottom four bits.

BitNot

Negation, bit-wise

a = **BitNot**(**b** [,Dataformat])

a: Result

b: Argument

Dataformat: Assumed (integer) data format (only in imc Inline FAMOS)

Allowed data types

-32: 32 bit signed

-16: 16 bit signed

-8: 8 bit signed

1: 1 Bit (digital)

Bitwise inversion of the parameter's value.

A logical inversion is performed for each bit. The result bit is 0 if the input bit is 1, else 1. Negative integers are represented as a two's complement.

For imc Inline FAMOS applies: Each value is converted to the specified integer data format. When converting to the data type "*Digital*", all values which do not equal 0 are regarded as 1. The result is identical to that of the logical NOT-operator.



Example

[imc Online FAMOS](#)

Bit-wise Negation of DIO_Port_01.

```
Res = BitNot( DIO_Port_01 )
```



Example

[imc Inline FAMOS](#)

Inversion of the digital bit 'Virt_Bit01'.

```
Res1 = BitNot( Virt_Bit01, 1 )
```

All bits of the 2-Byte wide physical channel 'Signal_01' are inverted.

```
Res2 = BitNot( Signal_01, -16 )
```

All bits of the 4-Byte wide CAN channel 'CAN_01' are inverted.

```
Res3 = BitNot( CAN_01, -32 )
```

BitOr

Bit wise OR connection of b and c

a = **BitOr**(**b**, **c**)

BitXor

Exclusive OR, bit-wise: Bit-wise exclusive OR operation on b and c

a = **BitXor**(**b**, **c**)

9.2.9.3.3 C

Charact

Characteristic: The input channel is corrected using a characteristic curve. For each original value, a result is calculated from the characteristic in the file.

a = Charact(b, "Filename")

"Filename": characteristic in file

Values which lie between two characteristic points are linearly interpolated.

It is possible to calculate characteristic curves from **equidistant** data sets and from **XY**-data sets in imc FAMOS format. The X-coordinates of an XY-characteristic curve must be monotonically increasing. If they are not strictly monotonically increasing, then for one x-value there are multiple possible y-values; this means that the characteristic curve has a vertical portion at this location. If the input channel has such a value, then one of the associated y-values is arbitrarily selected as the result.

Driving cycle: This function can be used to output a waveform to a analog output or fieldbus.

Load the file

In imc STUDIO, the file of the characteristic curve is permanently integrated in the experiment.

- Import this file using the ribbon "Setup-Configuration" > "Supplemental files".



Note

imc Inline FAMOS: The characteristic curve is not limited.

ClHistogram

Classification histogram: amplitude time-of-stay procedure

a = ClHistogram(b, MinValue, MaxValue, Total, ResUnit, EndClass)

a: Results

b: Input channel

MinValue: Lower limit of range

MaxValue: Upper limit of range

Total: Number of classes; ≥ 1

ResUnit: Result units

0: classes

1: like input channel

EndClass: End classes are

0: closed

1: open

The classification results are not complete until the end of measurement and therefore cannot be used in further computations.

This data classification procedure is based on DIN 45667 (German standard).

- For each class, the sum of the times during which the signal lies within the respective class boundaries is averaged separately.

In effect, this function counts the number of samples which occur in each class during a measurement.

- The time-of-stay in a class is computed by dividing the no. of samples.



Example

```
Histo1 = ClHistogram( Chan01, -10.0, 10.0, 32, 0, 0 )
```

- The input channel is divided into 32 classes ranging from -10 to 10, with closed end classes.
- The results are displayed as classes 0 to 31.

ClLevelCrossing

Level crossing histogram: Class counting with the level crossing method.

a = ClLevelCrossing(b, MinValue, MaxValue, Total, Reference, Hysteresis, ResUnit, EndClasses)

a: Results	ResUnit: Result units
b: Input channel	0: classes
MinValue: Lower limit of range	1: like input channel
MaxValue: Upper limit of range	EndClass: End classes are
Total: Number of classes; ≥ 1	0: closed
Reference: Reference level	1: open
Hysteresis: hysteresis for peak identification; ≥ 0.0 specified in percent of class width	

The number of class level crossings are being counted, thus the result is divided into the number of classes + 1.

- If closed end classes are specified, the end classes of the result will necessarily be empty.
- With open end classes, level crossings through the upper and lower classes are counted.

The class counting results are not complete until the end of measurement and therefore cannot be used in further computations.

 Example

Histo1 = ClLevelCrossing(Chan01, -10.0, 10.0, 64, 0.0, 10.0, 0, 0)

- The input channel is divided into 64 classes ranging from -10 to 10. Level crossings relative to 0.0 are counted with 10 percent hysteresis relative to the class width.
- The end classes are closed. The results are displayed as classes 0 to 64

ClMarkov

Data classification using Markov procedure

a = ClMarkov(b, MinValue, MaxValue, Total, Hysteresis, Axes, ResType, ResUnits, EndClass, Option)

a: Results

b: Input channel

MinValue: Lower limit of range

MaxValue: Upper limit of range

Total: Number of classes; ≥ 1

Hysteresis: Hysteresis for peak identification; ≥ 0.0

Axes: Axes configuration, x-axis is

0: target class or amplitude

1: starting class or mean value

ResType: Result matrix is of type

0: target/starting class

1: amplitude/mean

ResUnits: Result units

0: classes

1: like input channel

EndClass: End classes are

0: closed

1: open

Option: Algorithm versions

0: main algorithm

The classification results are not complete until the end of measurement and therefore cannot be used in further computations.

Class-counting according to the Markov procedure counts all consecutive class transitions between local signal extrema, e.g. from Maximum 1 to Minimum 1, from Minimum 1 to Maximum 2, ...

The hysteresis suppresses noise and minor (i.e. unimportant) cycles. A new peak or valley is recognized only when the distance to the previous one is larger than the specified hysteresis width.

The result matrix can use the starting/target class or amplitude/mean as dimensions.



Example

```
Matrix1 = ClMarkov(Chan01, -10.0, 10.0, 64, 0.15, 0, 0, 0, 1, 0)
```

- The input channel is divided into 64 classes ranging from -10 to 10 with open end classes.
- The results are stored in a starting / target class matrix with classes from 0 to 63.
- The hysteresis width is approx. one-half the class width.

CloseSaveInterval

With this function, a new storage folder is set up on the internal storage medium. The folder is closed along with all data captured up to this time.

All subsequent measurement data are then saved in the new folder.

Prerequisite:

Only available in: imc Online FAMOS!

Only usable with control commands!

CloseSaveInterval()

If this function occurs within a pre-set saving interval, then the saving interval in progress is terminated, a new folder is created and filled with measured data until the actual end of the interval is reached. Subsequently, interval storage resumes according to the usual procedure.

The function can be used when data storage in the device is activated. At each call, an additional folder is created. For this, a command is triggered whose execution can be delayed depending on the configuration.

Constant calling of the function may cause the creation of an uncontrolled amount of folders.



Example

```
OnTriggerMeasure( BaseTrigger )  
  If Virt_Bit01 > 0  
    CloseSaveIntervall()  
    Virt_Bit01 = 0  
  End  
End
```

ClRainFlow

Data classification using rainflow procedure

a = ClRainFlow(b, MinValue, MaxValue, Total, Hysteresis, Axes, ResType, ResUnits, EndClass, Option)

a: Results

b: Input channel

MinValue: Lower limit of range

MaxValue: Upper limit of range

Total: Number of classes; ≥ 1

Hysteresis: Hysteresis for peak identification; ≥ 0.0

Axes: Axes configuration, x-axis is

0: target class or amplitude

1: starting class or mean value

ResType: Result matrix is of type

0: target/starting class

1: amplitude/mean

ResUnits: Result units

0: classes

1: like input channel

EndClass: End classes are

0: closed

1: open

Option: Algorithm versions

0: main algorithm

1: with Clormann-Seeger correction

The classification results are not complete until the end of measurement and therefore cannot be used in further computations.

The rainflow procedure counts the cycles in a measured signal. Each cycle consists of a peak-valley pair (e.g. maximum and minimum).

The hysteresis suppresses noise and minor (i.e. unimportant) cycles. A new peak or valley is recognized only when the distance to the previous one is larger than the specified hysteresis width.

The result matrix can use the starting/target class or amplitude/mean as dimensions.



Example

```
Matrix1 = ClRainFlow( Chan01, -10.0, 10.0, 64, 0.15, 0, 0, 0, 1, 0 )
```

- The input channel is divided into 64 classes ranging from -10 to 10 with open end classes.
- The results are stored in a starting / target class matrix with classes from 0 to 63.
- The hysteresis width is approx. one-half the class width.

ClRainFlowRes

Rainflow with residues: data classification using rainflow procedure, output of remainder

ClRainFlowRes(*a*, *Remainder*, *b*, *MinValue*, *MaxValue*, *Total*, *Hysteresis*, *Axes*, *ResType*, *ResUnits*, *EndClass*, *Option*)

a: result matrix

Remainder: remainder

b: input channel

MinValue: lower limit of range

MaxValue: upper limit of range

Total: number of classes; ≥ 1

Hysteresis: hysteresis for peak identification; ≥ 0.0

Axes: axes configuration, x-axis is

0: target class or amplitude

1: starting class or mean value

ResType: Result matrix is of type

0: target/starting class

1: amplitude/mean

ResUnits: result units

0: classes

1: like input channel

EndClass: end classes are

0: closed

1: open

Option: algorithm versions

0: main algorithm

1: with Clormann-Seeger correction

The classification results are not complete until the end of measurement and therefore cannot be used in further computations. The remainder is not output until the end of the measurement.

The rainflow procedure counts the cycles in a measured signal. Each cycle consists of a peak-valley pair (e.g. maximum and minimum).

The hysteresis suppresses noise and minor (i.e. unimportant) cycles. A new peak or valley is recognized only when the distance to the previous one is larger than the specified hysteresis width.

The result matrix can use the starting / target class or amplitude / mean as dimensions.



Example

```
ClRainflowRes( Matrix1, Res1, Chan01, -10.0, 10.0, 64, 0.15, 0, 0, 0, 1, 0)
```

- The input channel is divided into 64 classes ranging from -10 to 10 with open end classes.
- The results are stored in a starting target class matrix with classes from 0 to 63. The remaining partial cycles are also displayed.
- The hysteresis width is approx. one-half the class width.

ClRainFlowTM

Rainflow with TrueMax filter: Rainflow class counting with TrueMax filter and hysteresis in percent of the class width.

a = ClRainFlowTM(b, MinValue, MaxValue, Total, Hysteresis, Axes, ResType, ResUnits, EndClass, Option)

a: Results	ResUnits: Result units
b: Input channel	0: classes
MinValue: Lower limit of range	1: like input channel
MaxValue: Upper limit of range	EndClass: End classes are
Total: Number of classes; ≥ 1	0: closed
Hysteresis: Hysteresis for peak identification; ≥ 0.0 Specify as a percentage of the class width.	1: open
Axes: Axes configuration, x-axis is	Option: Algorithm versions
0: target class or amplitude	0: main algorithm
1: starting class or mean value	1: with Clormann-Seeger correction
ResType: Result matrix is of type	
0: target/starting class	
1: amplitude/mean	

The class counting results are not complete until the end of measurement and therefore cannot be used in further computations.

The rainflow procedure counts the cycles in a measured signal. Each cycle consists of a peak-valley pair (e.g. maximum and minimum).

The hysteresis suppresses noise and minor (i.e. unimportant) cycles. A new peak or valley is recognized only when the distance to the previous one is larger than the specified hysteresis width.

The dimension of the result matrix can be the starting/target class or the amplitude/mean.

Because the extreme points of a signal are not usually accurately sampled, the TrueMax filter attempts to interpolate them more accurately.



Example

`Matrix1 = ClRainFlowTM( Chan01, -10.0, 10.0, 64, 50.0, 0, 0, 0, 1, 0)`

- The input channel is divided into 64 classes ranging from -10 to 10 with open end classes.
- The results are stored in a starting/target class matrix with classes from 0 to 63.
- The hysteresis width is one-half the class width.

ClRainFlowTMRes

RainflowTM with Residue: Data classification using rainflow procedure, output of remainder. With TrueMax filter and hysteresis in percent of the class width.

ClRainFlowTMRes(a, Remainder, b, MinValue, MaxValue, Total, Hysteresis, Axes, ResType, ResUnits, EndClass, Algorithm)

a: Result matrix

Remainder: Remainder

b: Input channel

MinValue: Lower limit of range

MaxValue: Upper limit of range

Total: Number of classes; ≥ 1

Hysteresis: Hysteresis for peak identification; ≥ 0.0
in % of class width

Axes: Axes configuration, x-axis is

0: target class or amplitude

1: starting class or mean value

ResType: Result matrix is of type

0: target/starting class

1: amplitude/mean

ResUnits: Result units

0: classes

1: like input channel

EndClass: End classes are

0: closed

1: open

Algorithm: use

0: main algorithm

1: main algorithm with Clormann-Seeger correction

The class counting results are not complete until the end of measurement and therefore cannot be used in further computations. The remainder is not outputted until the end of the measurement.

The rainflow procedure counts the cycles in a measured signal. Each cycle consists of a peak-valley pair (e.g. maximum and minimum).

The hysteresis suppresses noise and minor (i.e. unimportant) cycles. A new peak or valley is recognized only when the distance to the previous one is larger than the specified hysteresis width.

The dimensions of the result matrix can be the starting/target class or amplitude/mean.

Because the extreme points of a signal are not usually accurately sampled, the TrueMax filter attempts to interpolate them more accurately.



Example

```
ClRainflowTMRes(Matrix1, Res1, Chan01, -10.0, 10.0, 64, 50.0, 0, 0, 0, 1, 0)
```

- The input channel is divided into 64 classes ranging from -10 to 10 with open end classes.
- The results are stored in a starting/target class matrix with classes from 0 to 63. The remaining partial cycles are also displayed.
- The hysteresis width is one-half the class width.

ClRangePairCount

Classification histogram: Using the range pair counting procedure.

a = ClRangePairCount(b, MinValue, MaxValue, Total, Hyst, ResUnit)

a: Results

b: Input channel

MinValue: Lower limit of range

MaxValue: Upper limit of range

Total: Number of classes; ≥ 2

Hyst: Hysteresis; in physical units, ≥ 0

ResUnit: Unit of the Result

0: classes

1: like input channel

The class width is calculated according to the following equation:

$$\text{Classwidth} = (\text{MaxValue} - \text{MinValue}) / (\text{Total} + 1)$$

In other words, with a range from -10 V to 10 V, it is necessary to create 19 classes in order to achieve a class width of 1 V. The lowest, 'zero-th class' would represent a range from 0 V to 1 V, and since nothing is ever counted in that range, it is omitted and the remaining total is 19 instead of 20 classes.

Only the magnitude is counted. So all classes would range from 1 V to 20 V.

The class counting results are not complete until the end of measurement and therefore cannot be used in further computations. This data classification procedure is based on DIN 45667 (German standard): the sum of the range-pairs is counted and allocated in classes.



Example

```
Histo1 = ClRangePairCount( Chan01, -10.0, 10.0, 19, 0.1, 0 )
```

- The input channel is divided into 19 classes ranging from 1 to 20.
- The results are displayed as Classes 1 to 19.

ClRevolutionsHistogram

Revolutions histogram: Counts the revolutions into the classes of the input channel.

a = ClRevolutionsHistogram(b, c, MinValue, MaxValue, Total, ResUnit, EndClass)

a: Result

b: Input channel

c: speed in rpm

MinValue: Lower limit of range

MaxValue: Upper limit of range

Total: Number of classes; ≥ 1

ResUnit: Result units

0: classes

1: like input channel

EndClass: End classes are

0: closed

1: open

The class counting results are not complete until the end of measurement and therefore cannot be used in further computations.

CIRevolutionsMatrix

Two-channel revolutions matrix: counts the revolutions in a matrix.

a = **CIRevolutionsMatrix**(**b**, **c**, **MinValue1**, **MaxValue1**, **Total1**, **MinValue2**, **MaxValue2**, **Total2**, **Axes**, **ResUnits**, **EndClasses**)

a : Results	Axes : Axis order, speed is
b : first input channel	0: Z axis
c : Speed in rpm (second input channel)	1: X axis
MinValue1 : Lower limit of range	ResUnits : Result units
MaxValue1 : Upper limit of range	0: classes
Total1 : Number of classes; ≥ 1	1: like input channel
MinValue2 : Lower limit of range	EndClasses : End classes are
MaxValue2 : Upper limit of range	0: closed
Total2 : Number of classes; ≥ 1	1: open

The class counting results are not complete until the end of measurement and therefore cannot be used in further computations.

CITrueMax

Extreme value approximation: Approximates extreme values using interpolation.

a = **CITrueMax**(**b**)

CITwoChannelHistogram

Two channel histogram: Amplitude time-at-level procedure with two input channels and result matrix.

a = **CITwoChannelHistogram**(**b**, **c**, **MinValue1**, **MaxValue1**, **Total1**, **MinValue2**, **MaxValue2**, **Total2**, **ResUnit**, **EndClass**)

a : Results	Total2 : Number of classes; ≥ 1
b : first input channel	ResUnit : Result units
c : second input channel	0: classes
MinValue1 : Lower limit of range 1	1: like input channel
MaxValue1 : Upper limit of range 1	EndClass : End classes are
Total1 : Number of classes; ≥ 1	0: closed
MinValue2 : Lower limit of range 2	1: open
MaxValue2 : Upper limit of range 2	

The class counting results are not complete until the end of measurement and therefore cannot be used in further computations.

CodeRange

Returns specifiable values in response to a signal's value falling inside certain ranges.

a = **CodeRange**(**b**, **Delay**, "Filename")

a: Result

Delay: delay time

b: Input channel

"Filename": coefficient vector file

The coefficients must be supplied in a group of three in the following manner (Minimum, Maximum, Code value). If the input channel lies within one of the specified ranges, the result is the code value, otherwise the result is zero.

If a switching time is specified, the output of the result values' data stream is delayed by this switching time. The result is only then a code value, if the input value stays within the corresponding range for at least the duration of the switching time, otherwise it is zero.

If the resulting data stream and other channels are to be used together as parameters of functions, the other channels must also be supplied with a delay (**Function DelayLine**).



Example

```
Transmission = GearRatio (Nin, Nout, 1, 1, 10, 0 )
Gears = CodeRange( Transmission, 0.1, "CoGear.dat")
DelayedLoad = DelayLine( Stress, 0.1 )
Gear1Load = SamplesGate( DelayedLoad, Gears, 1 )
```

In the channel Gear1Load, all loading measured while the vehicle was in first gear is recorded.

Cos

Cosine of input channel

a = **Cos**(**b**)

a: Result

b: Input channel; in radians

CreateVChannel

By means of this function, measurement channels which are represented by only a single value can be provided with a time track. In this way, it is possible to trace how a value changes over time.

To this end, a virtual channel (*real numbers*) is generated with the specified value. This channel has the properties of the input channel.

a = **CreateVChannel**(**b**, **Value**)

a: Virtual channel

Value: single value, which is supplied to the virtual channel

b: input channel

The value supplied must be a single value. It can be a number or also, for instance, a virtual bit or a process vector element.



Example

```
Virt1 = CreateVChannel( Channel_001, 2.5 )
Virt2 = CreateVChannel( Channel_001, Virt_Bit01 )
Virt3 = CreateVChannel( Channel_001, pv.X123 )
```

CreateVChannelInt

By means of this function, measurement channels which are represented by only a single value can be provided with a time track. In this way, it is possible to trace how a value changes over time.

To this end, a virtual channel (*integers*) is generated with the specified value. This channel has the properties of the input channel.

a = CreateVChannelInt(b, Value)

a: Virtual channel

Value: single value, which is supplied to the virtual channel

b: input channel

The value supplied must be a single value. It can be a number or also, for instance, a virtual bit or a process vector element.



Example

```
VirtInt1 = CreateVChannelInt( Channel_001, 2 )
VirtInt2 = CreateVChannelInt( Channel_001, Virt_Bit01 )
VirtInt3 = CreateVChannelInt( Channel_001, pv.X123 )
```

CrossCorrelation

Cross-correlation between a test channel and a reference channel.

Prerequisite:

Only available in: imc Inline FAMOS!

Result = CrossCorrelation(Reference, Test, BlockLength)

Result: Results of the cross correlation

Test: test channel

Reference: Reference channel

BlockLength: Maximum delay (128 ... 131072)

The cross-correlation function (CCF) indicates how similar two data sets are for different shifts in the x direction. The cross-correlation function returns values between -1 and +1.

A value of 1 at the location t signifies that shift the test channel in the x-direction by t from the reference channel achieves the maximum correlation between the two channels. The two signals then have the same curve plot. A value of -1 signifies that the two signals are equal opposites (where one signal is positive, the other is negative by the same magnitude). A value of 0 signifies that the two signals are entirely uncorrelated at this shift. All values between -1 and +1 are possible.

Using the cross-correlation function, it is possible to determine whether one signal is present (hidden) within another, and additionally, by what delay one signal appears in another channel. For this purpose, the maximum of the cross-correlation function is of particular interest. Its location indicates the size of the delay by which the signal appears within the other channel. Its magnitude indicates how similar the signal having the delay is to the other.

In order to be able to interpret the delay between signals correctly, it is important to specify the parameters of the function [CrossCorrelation](#) in the correct order. The first parameter represents the reference channel containing the original signal (having no delay). The second parameter represents the test channel which contains a signal having a delay (and sometimes also interference) compared to the original. The x-coordinate of the cross-correlation function's maximum value directly indicates the shift between the test channel from the reference channel.

During calculation of the cross-correlation, the input channel is factorized into vectors. The length of these vectors represents the maximum delay between the test channel and the reference channel which can be determined by using the function `CrossCorrelation`.

Another effect of factorizing the input channel is that the data generated are considered periodic; i.e. the vectors are imagined to extend in both directions (left and right) in a periodic pattern. Thus, if one vector (window) contains one single pulse, the signal is interpreted as representing a chain of multiple pulses. One particular consequence is that when the cross-correlation function returns a large shift of 0.9 periods, this has the same meaning as a small shift in the negative direction, namely by -0.1 periods.

The x-scaling of the function `CrossCorrelation` is that of the two channels specified. Both channels should have the same x-scaling, otherwise the results may not have any sensible interpretation.

The function `CrossCorrelation` is normalized to the product of the RMS (root-mean square) of the two channels specified. Thus, the channel generated has no y-unit.



Example

```
Res = CrossCorrelation( RefChannel, TestChannel, 1024 )
```

The function `CrossCorrelation` is applied to a reference channel and a test channel (which may feature interference and a delay).

For example, if the channels have a sampling rate of 100 Hz, then due to the window width of 1024 samples, maximum delays of 10 s can be detected.

CurrentValue

Current value: A single value is isolated from out of the given channel.

Prerequisite:

Only usable with control commands!

SingleValue = `CurrentValue`(Channel, Option, Init)

SingleValue: Result	Option: Which value?	Init: Initialization value
Channel: Channel from which the value is to be taken	0: Last value	
	1: Minimum	
	2: Maximum	
	3: Mean value	

The return value is the initialization value, until an actual value from the channel is obtained. The parameter "Option" is important if, due to a high sampling rate for the channel, there are multiple values which were not processed yet. In that case, the parameter "Option" decides how the return value is derived from these values. The selection "last value" refers to the current value, the data point most recently measured on the channel.



Example

```
OnTriggerMeasure( BaseTrigger )
  Value = CurrentValue( Channel_001, 2, 0.0 )
  If Value > 100
    VrtChannel_001 = Channel_001*0 + 100
  Else
    VrtChannel_001 = Channel_001*0
  End
End
```

9.2.9.3.4 D

dB

Converts input channel to decibel

a = dB(b)

The calculation is performed according to the formula: $a = 20 * \log(\text{abs}(b))$

General information about the DelayBuffer functions

Prerequisite:

Only usable with control commands!

To use the `DelayBuffer` function, a `DelayBuffer` structure in the `OnInitAll` segment must be defined by means of the function `DelayBuffer`.

DelayBuffer

Initialization of the delay buffer: Sets up a delay line while specifying the maximum delay.

a = DelayBuffer(MaxSize)

a: The result is a DelayBuffer structure

MaxSize: Maximum buffer size, maximum size of the delay line

A signal can be delayed by N clock pulses. A buffer for measurement values is created having a defined length. In a processing step the current measured value is appended to the end of the buffer and the oldest measured value is retrieved from the start of the buffer. The buffer's length and the pulse rate of the processing steps determines the effective delay (see example).

The buffer's size is set to the specified maximum size. The buffer is completely pre-initialized with the default value 0.0. The actual delay can be set at any time using the function `DelayBuffer.SetSize`. The buffer also be completely refilled at any time with the specified value using the function `DelayBuffer.Fill`.

The most recently accumulated measured value are always held in the buffer, whose size is MaxSize. It is necessary to take account of the delay line's warmup time. This is because it must first be filled with a sufficient number of measured values before properly delayed values can be read out.

The function `DelayBuffer` must be called in the `OnInitAll` segment. In the process, a DelayBuffer structure is created and initialized.

**Example 1**

```

OnInitAll
    Dlb1 = DelayBuffer( 100 )
    Value = 0
    DelayValue = 0
End

OnSyncTask( 0.01 )
    ...
    ; delay of the output values by 1s (100*0.01s):
    DelayValue = Dlb1.Next( Value )
End

; Alternatively to the synchronized task, it is also possible to call
; the function .Next() within the timer:
OnTriggerStart( BaseTrigger )
    StartTimerPeriodic( 1, 0.1, 0.0 )
End

OnTimer( 1 )
    ...
    ; delay of the output values by 10s (100*0.1s):
    DelayValue = Dlb1.Next( Value )
End

```

**Example 2****Delay measuring channel**

A channel sampled at 1 kHz is delayed by 200 ms.

```

OnInitAll
    Dlb1 = DelayBuffer( 200 )
    DelayValue = 0
End

OnSyncTask( 0.001 )
    ; delay of the output values by 200 ms (200*0.001 s):
    DelayValue = Dlb1.Next( pv.Channel_001 )
End

OnTriggerMeasure( BaseTrigger )
    ; Create a virtual channel
    Channel_Delay = Channel_001 * 0 + DelayValue
End

```


DelayBuffer.Fill

Fill buffer: The entire buffer is filled with the value specified.

DelayBuffer.Fill(Value)

Value: The value with which the entire buffer is filled

All values ever previously stored in the buffer are overwritten.



Example

```
OnInitAll
    Dlb1 = DelayBuffer( 100 )
    Value = 0
    DelayValue = 0
End

OnSyncTask( 0.01 )
    If Virt_Bit01 > 0
        Dlb1.Fill( -1000.0 )
        Virt_Bit01 = 0
    End
    ...
    ; delay if the output value by 1s (100*0.01s):
    DelayValue = Dlb1.Next( Value )
End

; Alternatively to the synchronized task, it is also possible to call
; the function .Next() within the timer:
OnTriggerStart( BaseTrigger )
    StartTimerPeriodic( 1, 0.1, 0.0 )
End

OnTimer( 1 )
    ...
    ; delay of the output values by 10s (100*0.1s):
    DelayValue = Dlb1.Next( Value )
End
```

DelayBuffer.Next

Perform processing step: The new measured value (b) is appended to the end of the buffer and the oldest measurement value at the beginning of the buffer is returned.

a = DelayBuffer.Next(b)

a: The result is the delayed value, the oldest measurement value at the start of the buffer

b: The new measured value to be appended at the end of the buffer

The warmup time must be taken into account. If a delay of N measured values is set, then the very first N calls of this function only return the pre-initialized value.



Example

```
OnInitAll
    Dbl1 = DelayBuffer( 100 )
    Value = 0
    DelayValue = 0
End

OnSyncTask( 0.01 )
    ...
    ; delays the output value by 1s (100*0.01s)
    DelayValue = Dbl1.Next( Value )
End

; alternatively to the synchronous task, it is also possible to call
; the function .Next() within the timer:
OnTriggerStart( BaseTrigger )
    StartTimerPeriodic( 1, 0.1, 0.0 )
End

OnTimer( 1 )
    ...
    ; delay of the output value by 10s (100*0.1s)
    DelayValue = Dbl1.Next( Value )
End
```

DelayBuffer.SetSize

Set buffer size: Changing the buffer's size

DelayBuffer.SetSize(Size)

Size: New number of values in the buffer, new size of delay line

The size specified determines the delay. The signal is delayed by this many steps. The size can be changed at any time. A value of 0 means no delay. The maximum size specified in `DelayBuffer` may not be exceeded, however.

If the size is reduced, then stored values are skipped the next time `DelayBuffer.Next` is called. If the size is increased, then an older value which is still saved is fetched from the buffer the next time `DelayBuffer.Next` is called.



Example

```
OnInitAll
    Dlb1 = DelayBuffer( 100 )
    Value = 0
    DelayValue = 0
End

OnSyncTask( 0.01 )
    If Virt_Bit01 > 0
        Dlb1.SetSize( 50 )
        Virt_Bit01 = 0
    End
    ...
    ; delay of the output values by 1s (100*0.01s):
    DelayValue = Dlb1.Next( Value )
End

; Alternatively to the synchronized task, it is also possible to call
; the function .Next() within the timer:
OnTriggerStart( BaseTrigger )
    StartTimerPeriodic( 1, 0.1, 0.0 )
End

OnTimer( 1 )
    ...
    ; delay of the output values by 10s (100*0.1s):
    DelayValue = Dlb1.Next( Value )
End
```

DelayLine

Delay stream: The data stream is delayed by the specified time.

a = **DelayLine**(**b**, **Delay**)

a: result

Delay: delay time

b: input channel

The result value data stream is delayed, a value from the input channel only appears at the output after a delay interval.

If other functions used produce delayed results, and the channel in question is to be combined with others in computations, then it is necessary to amend the channel with the same delay time that the others have.



Example

```
Gears = CodeRange (Transmission, 0.1, "CoGear.dat" )
DelayedLoad = DelayLine( Load, 0.1 )
Gear1Load = SamplesGate( DelayedLoad, Gears, 1 )
```

- In the channel Gear1Load, all loading which occurs while the vehicle is in first gear is recorded.
- Since the function `CodeRange` delays its result by 0.1 seconds, the Loading-channel must also be delayed by 0.1s.

DFilt

Digital filter: sends input channel through a digital filter

a = **DFilt**(**b**, "Filename")

a: Result

"Filename": Coefficient vector file

b: Input channel

The file of coefficients must be saved in imc FAMOS format.

The coefficients come in two different formats:

1. Product sum format

$$y[t] = b_0 / a_0 * x[t] + \dots + b_n / a_0 * x[t-n] - a_1 / a_0 * y[t-1] - \dots - a_n / a_0 * y[t-n]$$

The a_n terms must lead, followed by the b_n terms.

The coefficient vector must include the same number of denominator and numerator terms. Set any non-appearing terms to zero.

The denominator coefficients are to be specified first and the first value of the denominator coefficients a_0 must be equal to 1. All higher order denominator coefficients $a[n]$ are assumed equal to zero.

2. Biquad format

The filter consists of 2nd order elements connected in "series". Each of these filters is defined as a biquad term of the form

$$y[t] = b_0 * x[t] + b_1 * x[t-1] + b_2 * x[t-2] + a_1 * y[t-1] + a_2 * y[t-2]$$

The coefficient vector must be specified in the following form:

$$0 \ a_2 \ a_1 \ b_2 \ b_1 \ b_0 \ \dots \ a_2 \ a_1 \ b_2 \ b_1 \ b_0$$

The 0 in the first position designates the biquad format.



Example

Suppose the filter coefficients for a Butterworth 2nd-order low pass filter with a cutoff frequency of 100 Hz are given. The filter is to process a channel sampled at 5 kHz. The file of coefficients contains the following numbers:

```
0
-0.950212 a2
1.94894 a1
0.000317864 b2
0.000635728 b1
0.000317864 b0
```

$$y[t] = b0 * x[t] + b1 * x[t-1] + b2 * x[t-2] + a1 * y[t-1] + a2 * y[t-2]$$

$$y[t] = 0.000317864 * x[t] + 0.000635728 * x[t-1] + 0.000317864 * x[t-2] + 1.94894 * y[t-1] + -0.950212 * y[t-2]$$

The file of coefficients must be saved in imc FAMOS format. If you have an ASCII table, load that file into imc FAMOS first and save it as *.dat file.



Note

- **DFilt** is initialized in imc Online FAMOS and imc Inline FAMOS as if the first measured value has already existed for a long time. With "quiet" signals, then, such as temperatures, transients are barely noticed. The drawback becomes clear, however, if a disturbance pulse is the first value and causes the filter to oscillate strongly.
- The **DFilt** function in the analysis software imc FAMOS initializes the filter with zeroes. For this reason, the two varieties respond differently during filter transient times.

Diff

Differentiation of argument

a = Diff(b)

a: Results

b: Input channel

The result values are the difference of the last and next to last input value divided by the sampling rate.

For the first input value a zero is returned.

DiskFreeSpace

Returns the free disk space on internal storage medium.

Prerequisite:

Only available in: imc Online FAMOS!

a = DiskFreeSpace(b)

a: Free disk space in kBytes (1 kByte = 1024 Bytes).

b: Reserved parameter (set to 0)

Zero will be returned in case of failure or absent storage medium.

DiskFreeTime

Free time: Estimate of time remaining for recording measurement data until internal storage medium is full.

Prerequisite:

Only available in: imc Online FAMOS!

a = DiskFreeTime(b)

a: Remaining free time in s

b: Reserved parameter (set to 0)

In general, the remaining free time cannot be determined exactly. The accuracy of the approximation made by this function depends on several factors. If you doubt the accuracy of this function, try using the function [GetDiskFreeSpace](#) which returns a more reliable estimate.

The approximation assumes that all triggers have been released and that data from these events is being recorded continuously to the hard drive.

Note the following:

- This function makes use of the function [DiskFreeSpace](#). Consequently, its maximum resolution is 32 kByte.
- If new triggers are released, the hard drive space is used up more quickly.
- This function cannot make accurate predictions when data is not streaming at a constant sampling rate. In these cases the following mean data rates can be used as approximations:
 - Transitional Recording: 50% reduction
 - DIO Port reduced: 20 bytes/s
 - CAN data with time stamps: 50 bytes/s
 - Alarm channels: 1 byte/s
 - Histograms, matrices: not possible (space required one time only at trigger event)
 - Residues: not possible (space required one time only at the end of the measurement)

If clusters on the hard drive are defective, this also reduces the remaining time from the function estimate.

If you expect large deviations, the return value of the function has to be corrected!

For example, you could multiply the function result with an appropriate factor to account for uncertainties.



Example

Note that the result is a single value and cannot be treated as a channel.

```
Display_01 = GetDiskFreeTime (0) * 0.9 / 60 ;remaining time in min, 10% uncertainty
Duration_Channel = Channel_001 * 0 + GetDiskFreeTime ( 0 )
; returns time duration continuously
```

DiskRunDir

Current folder number on the storage medium

Prerequisite:
Only available in: imc Online FAMOS!
Function only for imc DEVICES; in imc STUDIO Monitor it is not used.

a = DiskRunDir(b)

a: Current folder number	b: Reserved parameter (set to 0)

This function gets the number of the current folder on the device's drive when "Continuous numbering" is active.

When "Continuous numbering" is activated for data storage, then upon starting measurement, or with interval data saving, the folder number is incremented by 1.

"0" will be returned in case of failure or absent storage medium.

DisplaySetButton

Function for setting a display button

Prerequisite:
Only available in: imc Online FAMOS!

DisplaySetButton(Page, Button)

Page: Page number	Button: Button index.
The enumeration of the page number starts at 1. This corresponds to the enumeration in the function DisplaySetPage .	The enumeration of the button number starts at 0. This corresponds to the preset enumeration that one obtains by using the option 'Quick Edit' in the display editor.

 **Note**

If a display button is linked to a function like balance of channels or jump calibration, such a procedure can be executed with [DisplaySetButton](#) by imc Online FAMOS also.

DisplaySetPage

Function to show a new Displaypage

Prerequisite:
Only available in: imc Online FAMOS!

DisplaySetPage(Page)

Page: Pagenumber

9.2.9.3.5 E

General information about the ECU functions

Prerequisite:

Only available in: imc Online FAMOS!

Only usable with control commands!

ECU-functions are only available if the measurement device has a CAN-MultiIO.

When the function is called several times in succession, the commands are carried out separately one at a time. For this purpose, there is a small buffer which is sufficient for a few commands.

The following ECU functions are evaluated with [OnECUCmdReturn_ECU_001](#). There you will find a comprehensive example.

ECUReadTroubleCodes_ECU_*

The function runs the ECU-command 'ReadDiagnosticTroubleCodesByStatus'.

ECUReadTroubleCodes_ECU_*

When the function [ECUReadTroubleCodes_ECU_001](#) is called, only one command is activated, and then the online configuration is processed further.

The activated command is only completed when [OnECUCmdReturn_ECU_001](#) returns the value ECU command 18h.

ECUStartSession_ECU_*

The function runs the ECU-command 'StartDiagnosticSession'.

ECUStartSession_ECU_*

When the function [ECUStartSession_ECU_001](#) is called, only one command is activated, and then the online configuration is processed further.

The activated command is only completed when [OnECUCmdReturn_ECU_001](#) returns the value ECU command 10h.

ECUStopSession_ECU_*

The function runs the ECU-command 'StopDiagnosticSession'.

ECUStopSession_ECU_*

When the function [ECUStopSession_ECU_001](#) is called, only one command is activated, and then the online configuration is processed further.

The activated command is only completed when [OnECUCmdReturn_ECU_001](#) returns the value ECU command 20h.

ECUClearDiagInformation_ECU_*

The function runs the ECU-command '[ClearDiagnosticInformation](#)'.

ECUClearDiagInformation_ECU_001()

When the function [ECUClearDiagInformation_ECU_001](#) is called, only one command is activated, and then the online configuration is processed further.

The activated command is only completed when [OnECUCmdReturn_S1](#) returns the value ECU command 4h.

When the function is called several times in succession, the commands are carried out separately one at a time. For this purpose, there is a small buffer which is sufficient for a few commands.



Example

```
OnTriggerMeasure( BaseTrigger )
  If Virt_Bit01 > 0
    ECUClearDiagInformation_ECU_001( )
    Virt_Bit01 = 0
  End
End

OnECUCmdReturn_S1( Return, ECUCmd, CmdID )
  ECUCmd1 = BitAnd( ECUCmd, 0x000FFFFF )
  ECUIndex1 = BitAnd( ECUCmd, 0x00FF0000 )
  NodeIndex1 = BitAnd( ECUCmd, 0x0F000000 )
  If Return <> 0
    If ECUCmd1 = 0x4
      RecordText( "Error occurred in command 0x4" )
    Else
      RecordText( "General error occurred" )
    End
  End
End
End
```

OnECUCmdReturn_ECU_*

Return-treatment: Evaluation of the command-return values

OnECUCmdReturn_ECU_001(Return, ECUCmd, CmdID)

Return: return value; possible options:

- = 0: performed successfully
- = 1: timeout error occurred
- > 1: protocol-specific error occurred, referred to in the manual under the pertinent protocol

ECUCmd: ID of the ECU command, see below on list of ECU commands with associated IDs

CmdID: ID or address of the ECU command (e.g. for routines), else 0

The function parameters Return, ECUCmd and CmdID are treated as local single value variables in the [OnECUCmdReturn_ECU_001](#) block, and may only be used within this command. These three variables are filled when the command is carried out and can be inquired in the [OnECUCmdReturn_ECU_001](#) block. They are single values.

With compound commands, the parameter ECUCmd returns the ID of the last command carried out. For instance, for the compound command ECU-Start with security access, the parameter ECUCmd returns the security access ID and not the ID of the ECU-Start command.



Example

```
OnTriggerMeasure( BaseTrigger )
  If VirtBit_01 > 0
    ECUCmdReadTroubleCodes_ECU_001( )
    VirtBit_01 = 0
  End
  If VirtBit_02 > 0
    ECUCmdStartSession_ECU_001( )
    VirtBit_02 = 0
  End
  If VirtBit_03 > 0
    ECUCmdStopSession_ECU_001( )
    VirtBit_03 = 0
  End
End
OnECUCmdReturn_ECU_001( Return, ECUCmd, CmdID )
  If Return <> 0
    If ECUCmd = 0x18
      RecordText( "Error occurred in command..." )
    End
  End
End
End
```

KWP 2000 commands set in the CAN-Assistant, IDs hexadecimal:

StartDiagnosticSession: 10 86	ReadMemoryByAddress: 23
SecurityAccess: 27	WriteDataByLocalID: 3b
LogInKey: 00 00	WriteDataByCommonID: 2e
StopDiagnosticSession: 20	WriteMemoryByAddress: 3d
TesterPresent: 3e	StartRoutineByLocalID: 31
ReadDiagnosticTroubleCodesByStatus: 18 00 ff 00	StartRoutineByAddress: 38
ReadDataByLocalID: 21	RequestRoutineResultsByLocalID: 33
ReadDataByCommonID: 22	RequestRoutineResultsByAddress: 3a

Equal

Compares for equality

a = Equal(b, c)

a: Result

b: 1. argument

c: 2. argument

a = 1 if the arguments are equal, otherwise 0.

ExpoRMS

Moving RMS: Moving root-mean-square with exponential averaging and reduction.

a = **ExpoRMS**(**b**, **tau**, **Reduction** [, **Mode**])

a: Result

b: Input channel or input value

tau: Time constant for exponential weighting

In imc Online FAMOS:

≥ 0.0: Time constant is defined by the user in s

In imc Inline FAMOS:

-1: Fast (0.125s)

-2: Slow (1s)

-3: Pulse

-4: Peak

-5: RMS in interval

-6: RMS from start

≥ 0.0: Time constant is defined by the user in s

Reduction: Reduction depending on the value of "Mode"

Mode: [Optional] Interpretation of the parameter "Reduction"

1: The parameter "Reduction" is the reduction factor expressed in value count

2: The parameter "Reduction" is the sampling interval of the result

When the value 1 is specified for "Mode", the "Reduction"-value is interpreted as the reduction factor (RF). The **ExpoRMS**-function calculates the moving RMS-value with exponential averaging. Additionally, the data rate is reduced by the factor "Reduction".

Reduction = 1: After each input value, a result value is calculated.

Reduction = 2: After every other input value, a result value is calculated.

Reduction = n: After every n-th input value, a result value is calculated.

When the value 2 is specified for "Mode", the "Reduction"-value is interpreted as the target sampling interval of the result channel. The target sampling interval must be the result of dividing the input channel's sampling interval by an integer.

Example: For an input channel sampling interval of 100 ms (0.1), the target sampling interval could be 500 ms (0.5), for example. This would represent a reduction factor (RF) of 5.

The 4th parameter "Mode" is optional. In the absence of any specification, the default value 1 is applied.

In order to obtain useful results, the function requires a time base. When a channel is set as the 1st parameter, it is provided as the channel's sampling interval. When a single value (e.g. a process vector variable) is set as the 1st parameter, one can obtain a time reference by using timers (or on the device, in synchronous imc Online FAMOS tasks). When using single values as the input value (1st parameter), then for "Mode" = 1 the only possible setting for the reduction factor is 1. "Mode" = 2 is not supported with single values.



Example 1

```
Res = ExpoRMS( Signal_01, 0.125, 1000 )
```

Returns the moving RMS-value for every thousandth input value.

**Example 2****with optional parameter**

```
Res = ExpoRMS( Signal_01, 0.125, 0.1, 2 )
```

For a channel sampled at 10 kHz, the moving RMS-value is returned the moving RMS-value is returned.
The result has a sampling rate of 10 Hz.

9.2.9.3.6 F

FFT

Magnitude spectrum: FFT with output of the magnitude spectrum

a = FFT(b, Window, FFT_length)

a : Result; Magnitude spectrum	Window : Type of window	FFT_length : FFT-length
b : Input channel	0: Rectangle	In imc Online FAMOS: 128 ... 8192
	1: Hamming	In imc Inline FAMOS: 128 ... 131072
	2: Hanning	
	3: Blackman	
	4: Blackman-Harris	
	5: Flat-Top	

The FFT length may only take one of the following values:
In imc Online FAMOS: 128, 256, 512, 1024, 2048, 4096, 8192.
In imc Inline FAMOS: 128, 256, 512, 1024, 2048, 4096, 8192, 16384, 32768, 65536, 131072.
The length of the result vector is determined as 1+(FFT_Length/2).

 Example

Res = FFT(Signal_01, 3, 1024)
Calvculation of an FFT with Blackman window and an FFT-length of 1024. The result is a streaming vector of length 513.

FFTAplitudePhase

Magnitude-phase spectrum: FFT with output of magnitude and phase

FFTAplitudePhase(Amplitude, Phase, b, Window, FFT_length)

Amplitude: Result 1; Amplitude (magnitude) spectrum	Window: Type of window	FFT_length: FFT-length
Phase: Result 2; Phase spectrum	0: Rectangle	In imc Online FAMOS:
b: Input channel	1: Hamming	128 ... 8192
	2: Hanning	In imc Inline FAMOS:
	3: Blackman	128 ... 131072
	4: Blackman-Harris	
	5: Flat-Top	

The results of the procedure are the two first parameters: Magnitude and Phase. The FFT length may only take one of the following values:

- In imc Online FAMOS: 128, 256, 512, 1024, 2048, 4096, 8192.
- In imc Inline FAMOS: 128, 256, 512, 1024, 2048, 4096, 8192, 16384, 32768, 65536, 131072.

The length of the result vectors is determined as 1+(FFT_Length/2).

 **Note**

Peak values or RMS-values

The function FFTAplitudePhase calculates the peak values when applied in imc Online FAMOS, but in imc Inline FAMOS it calculates the RMS-values.

For this reason, different calculation results are usually obtained.

 **Example**

```
FFTAplitudePhase( Amplitude, Phase, Signal_01, 1, 2048 )
```

Calculation of an FFT with Hamming window and an FFT length of 2048. The results are streaming vectors of length 1025.

FFTAverage

Overlapped, resampled and/or averaged amplitude spectrum: FFT with overlapped or skipped windows, arbitrary window size and averaging across a certain number of spectra or from the beginning.

Spectra = FFTAverage(Data, Width, Window, Overlap, Average, Count)

Spectra: Result; Averaged FFTs	Overlap: stated in percent	Average: averaging type
Data: Input channel, time-domain data	-400: every 5th FFT	0: none
	-100: every 2nd FFT	1: arithmetic mean
Width: window width	0: every FFT	2: maximum
In imc Online FAMOS:	33: 33.33% overlap	3: minimum
100 ... 8192	50: 50% overlap	4: average from beginning
In imc Inline FAMOS:	66: 66.66% overlap	Count: number of data averaged
100 ... 131072	75: 75% overlap	
Window: Type of window	90: 90% overlap	
0: rectangle		
1: Hamming		
2: Hanning		
3: Blackman		
4: Blackman-Harris		
5: Flat-Top		

If the count is given as 10, then at first, 10 time data spectra are taken and averaged. The average spectrum is returned. All averages are carried out with the spectrum's magnitude.

The window width does not need to be a power of 2, in which case the data automatically gets interpolated. It is thus possible, for example, for a sampling rate of 5 kHz and 1000 points window width, to obtain spectral lines at exact multiples of 5 Hz.

- If the overlap is > 0, then the time windows overlap by the amount specified. However, in that case the computation time increases greatly!
- If the overlap is negative, not all time data are considered when forming the spectra.

The result values are to be interpreted as RMS values.

It is possible to calculate a Peak-Hold spectrum from the beginning of the measurement in combination with the function `VmaxV`⁴³²; see Example 2.



Example

```
Res = FFTAverage( Signal_Vib, 1000, 3, 0, 1, 100 )
```

Calculation of averaged FFTs with a Blackman window and window widths of 1000. Every FFT is factored in and the arithmetic mean of 100 spectra is taken.



Example 2

Peak-Hold spectrum

```
Res = FFTAverage( Signal_Vib, 1024, 0, 0, 1, 10 )
ResPeakHold = VMaxV( Res, 2, 1 )
```

Calculation of averaged FFTs with a rectangle window and window widths of 1024. Every FFT is factored in and the arithmetic mean of 10 spectra is taken. Calculation of the Peak-Hold spectrum from the beginning of the measurement.

FFTInverse

Inverse FFT: Inverse FFT from complex data

a = **FFTInverse**(**RealPart**, **ImaginaryPart**)

a : Result, Output time signal	RealPart : Real part of complex spectra
	ImaginaryPart : Imaginary part of complex spectra

The internal FFT function works with up to 4096 (actually complex) points.
A special technique deconstructs 8192 real (time-domain) points so that the spectrum can be computed by means of the 4 k internal FFT. The 4 k internal FFT contains a 4 k Real part and a 4 k Imaginary part.
However, the technique can not be used on complex data in the frequency domain. The data are already complex, for which reason it is not possible to transfer 8 k complex data.



Example

```
FFTRealImaginary( Reall, Imagl, Signal_01, 3, 1024 )  
Res = FFTInverse( Reall, Imagl )
```

Calculation of a complexe FFT with Real- and Imaginary parts. Subsequently, determination of the time signal from the Real- and Imaginary parts.

FFTRealImaginary

Complex FFT: FFT with real- and imaginary part output

FFTRealImaginary(**RealPart**, **ImaginaryPart**, **b**, **Window**, **FFT_length**)

RealPart : Result 1; Real part of the spectrum (returned vector)	Window : Type of window	FFT_length : FFT-length
ImaginaryPart : Result 2; Imaginary part of the spectrum (returned vector)	0: Rectangle	In imc Online FAMOS:
b : Input channel	1: Hamming	128 ... 8192
	2: Hanning	In imc Inline FAMOS:
	3: Blackman	128 ... 131072
	4: Blackman-Harris	
	5: Flat-Top	

The FFT length may only take one of the following values:
In imc Online FAMOS: 128, 256, 512, 1024, 2048, 4096, 8192.
In imc Inline FAMOS: 128, 256, 512, 1024, 2048, 4096, 8192, 16384, 32768, 65536, 131072.
The length of the result vectors is determined as 1+(FFT_Length/2).



Example

```
FFTRealImag( Reall, Imagl, Signal_01, 5, 8192 )
```

Calculation of a complexen FFT with Flat-Top window and an FFT length of 8192. The results are streaming vectors of vector length 4097.

General information about the filters

The filter functions `FiltBP`, `FiltBS`, `FiltHP` and `FiltLP` are like their analog examples. It is recommended to set the parameters in a way, the analog circuit would be chosen. Parameters like order and cutoff frequency could create filters, that couldn't be build as a real analog circuit. However, usually extreme settings don't end in practicable results and stress the online processor unnecessary.

Due to numerical problems, the ratio of the sampling frequency to the cutoff frequency with filters may not exceed a certain amount.

The following rule applies:

$fsample/fg < (Order)th\ Root(1000000)$; where $fsample$ = sampling frequency and fg = cutoff frequency

FiltBP

Processes channel data with a band pass filter

Result = `FiltBP`(Data, Characteristic, Para, Order, CutoffFreqLow, CutoffFreqUp)

Result: Filtering result	Para: Reserved
Data: Channel to be filtered	Order: Filter order
Characteristic: Filter characteristic	Bessel: 1..40
0: Butterworth	otherwise: 1..100
1: Bessel	CutoffFreqLow: Lower cutoff frequency (Hz)
2: Chebychev	CutoffFreqUp: Upper cutoff frequency (Hz)
3: Critical damping	

FiltBS

Processes channel data with a band stop filter

Result = `FiltBS`(Data, Characteristic, Para, Order, CutoffFreqLow, CutoffFreqUp)

Result: Filtering result	Para: Reserved
Data: Channel to be filtered	Order: Filter order
Characteristic: Filter characteristic	Bessel: 1..40
0: Butterworth	otherwise: 1..100
1: Bessel	CutoffFreqLow: Lower cutoff frequency (Hz)
2: Chebychev	CutoffFreqUp: Upper cutoff frequency (Hz)
3: Critical damping	

FiltHP

Performs high pass filtering of channel data

Result = FiltHP(Data, Characteristic, Para, Order, CutoffFreq)

Result: Filtering result	Para: Reserved
Data: Channel to be filtered	Order: Filter order
Characteristic: Filter characteristic	Bessel: 1..20
0: Butterworth	Chebyshev: 1..50
1: Bessel	otherwise: 1..100
2: Chebyshev	CutoffFreq: Cutoff frequency (Hz)
3: Critical damping	

FiltLP

Performs low-pass filtering of channel data

Result = FiltLP(Data, Characteristic, Para, Order, CutoffFreq)

Result: Filtering result	Para: Reserved
Data: Channel to be filtered	Order: Filter order
Characteristic: Filter characteristic	Bessel: 1..20
0: Butterworth	Chebyshev: 1..50
1: Bessel	otherwise: 1..100
2: Chebyshev	CutoffFreq: Cutoff frequency (Hz)
3: Critical damping	

Floor

Truncates the argument to the next smallest or equal integer.

a = Floor(b)

This function returns the **next lower integer** for a real number input, i.e. it rounds down to the next lower integer.



The function returns

Input channel value	Result value
1,2	1
1,9	1
-1,2	-2
-1,9	-2

9.2.9.3.7 G

GearRatio

Transmission ratio: A gear system's transmission ratio is calculated on the basis of the engine rpm and the live axle rpm.

a = **GearRatio**(**Ne**, **Na**, **MinNe**, **MinNa**, **Ft**, **Code**)

a : result	MinNe : minimum input rpm
Ne : input rpm, engine	MinNa : minimum output rpm
Na : output rpm, live axle	Ft : low-pass cutoff frequency
	Code : code value, output less than minimum

The two rpm values are filtered with a low-pass and the transmission ratio Ne to Na is computed. Additionally, the rpm values are checked against their respective minima.

If the output rpm value falls short of MinNa or both of the minima are fallen short of, the output is a specified result value.

If only the input rpm value falls short of the minimum, the output is zero.



Example

```
Transmission = GearRatio( Nin, Nout, 1, 1, 10, 0 )
Gears = CodeRange( Transmission, 0.1, "CoGear.dat" )
DelayedLoad = DelayLine( Last, 0.1 )
Gear1Load = SamplesGate( DelayedLoad, Gears, 1 )
```

In the channel Gear1Load, all loading measured while the vehicle was in first gear is recorded.

GetDateTime

The function determines the current date and time.

Prerequisite:

Only usable with control commands!

GetDateTime(**SecondsDecimal**, **Seconds**, **Minute**, **Hour**, **Day**, **Month**, **Year**)

SecondsDecimal : current fraction of seconds in decimal notation, 0.0 ... 0.99	Day : current day, 1 ... 31
Seconds : current seconds, 0 ... 59	Month : current month, 1 ... 12
Minute : current minutes, 0 ... 59	Year : current year, e.g. 2009
Hour : current hours, 0 ... 23	

The decimal fraction of seconds is precise to within one 64th of a second.

The function parameters may be local single value variables, display variables or pv-variables.

Channels are not permitted as parameters. If any parameter is not required, a zero can be entered in its place.



Example

```
OnInitAll
    FindDate = 0
    FindTime = 0
    pv.Minute = 0
    pv.Hour = 0
    Time1 = ""
End

OnTriggerMeasure( BaseTrigger )
    If FindDate > 0
        GetDateTime( 0, 0, 0, 0, DisplayVar_Day, DisplayVar_Month, DisplayVar_Year )
        FindDate = 0
    End

    If FindTime > 0
        GetDateTime( 0, 0, pv.Minute, pv.Hour, 0, 0, 0 )
        FindTime = 0
        ; output as text "Time: ss:mm"
        Time1 = "Time: "
        If pv.Hour < 10
            Time1 = TextAdd( Time1, "0" )
        End
        Time1 = TextAdd( Time1, TextFormatI( pv.Hour ) )
        Time1 = TextAdd( Time1, ":" )
        If pv.Minute < 10
            Time1 = TextAdd( Time1, "0" )
        End
        Time1 = TextAdd( Time1, TextFormatI( pv.Minute ) )
        RecordText( Time1 )
    End
End
```



Example

```
OnInitAll
    SecondsDp = 0
    Seconds = 0
    Minute = 0
    Hour = 0
End

OnSyncTask( 0.1 )
    If Virt_Bit01 > 0
        GetDateTime( SecondsDp, Seconds, Minute, Hour, 0, 0, 0 )
        Virt_Bit01 = 0
    End
End
```

GetDuration

A channel's measurement duration: Finds the duration of measurement for a physical channel.

MeasurementDuration = **GetDuration**(**InputChannel**)

MeasurementDuration: Measurement duration for the physical channel in s. **InputChannel**: Physical channel

The measurement value can be determined for all physical channels.

In particular for analog inputs, incremental counter inputs, Fieldbus inputs, digital Fieldbus ports and digital ports.

If no measurement duration is defined, the value 0 is returned.



Example

```
; measurement duration of physical channel Channel_001 in s
_MeasDuration_01 = GetDuration( Channel_001 )
```

GetHistoValue

Value from a histogram: Extracts the value from the specified index of a histogram.

HistogramValue = **GetHistoValue**(**Histogram**, **Index**)

HistogramValue: Histogram value at specified index **Index**: Index of histogram position; 0, 1, ..., number of classes -1
Histogram: Histogram from which the value at the specified index location is to be extracted



Example

```
OnTriggerMeasure( BaseTrigger )
  Histo = ClHistogram( Channel_001, -10.0, 10.0, 32, 1, 1 )
  ; 1st histogram value, histogram index: 0
  Histo_0 = Channel_001*0 + GetHistoValue( Histo, 0 )
  ; Last histogram value, histogram index: 31
  Histo_1 = Channel_001*0 + GetHistoValue( Histo, 31 )
End
```

GetHistoValue2

Value from a histogram: Extracts the value from the specified column index (X_Index) and row index (Y_Index) of a histogram.

HistogramValue = **GetHistoValue2**(Histogram, X_Index, Y_Index)

HistogramValue: Histogram value at specified indices

Histogram: Histogram from which the value at the specified index locations is to be extracted.

X_Index: Column index in the histogram; 0, 1, ..., number of x-classes -1

Y_Index: Row index in the histogram; 0, 1, ..., number of y-classes -1

The histogram must be given in the form of a matrix of dimensions (number of x-classes) * (number of y-classes).

The histogram's X-index denotes the index in the X-direction in the curve window (histogram column index), the histogram's Y-index denotes the index in the Y-direction in the curve window (histogram row index).



Example

```
OnTriggerMeasure( BaseTrigger )
  Histo = ClRainFlow( Channel_001, -10.0, 10.0, 32, 0, 0, 0, 1, 1, 0 )
  ; 1st histogram value, histogram index: 0, column index: 0, row index: 0
  Histo_0 = Channel_001*0 + GetHistoValue2( Histo, 0, 0 )
  ; 69th histogram value, histo index: 68, column index: 4, row index: 2
  ; Histogram index = row index * 32 + column index = 2 * 32 + 4 = 68
  Histo_68 = Channel_001*0 + GetHistoValue2( Histo, 4, 2 )
End
```

GetLastError

Determines the error number of the last error. Determines the error number of the last error depends on what filter is configured.

GetLastError(ErrorNumber, ErrorOrigin, Filter)

ErrorNumber: Result. Error number of last error. The result values are ≤ 0 , e.g. -5613.

ErrorOrigin: Result. Denotes where error occurred. The result values are ≥ 0 and ≤ 11 .

- 0: No error
- 1: Error in Fieldbus system, Slot 1
- 2: Error in Fieldbus system, Slot 2
- 3: Error in Fieldbus system, Slot 3
- 4: Error in Fieldbus system, Slot 4
- 5: Error in Fieldbus system, Slot 5
- 6: Error in Fieldbus system, Slot 6
- 7: Error in Fieldbus system, Slot 7
- 8: Error in Fieldbus system, Slot 8
- 9: Error in the online-system
- 10: Error on the internal drive
- 11: Error in data capture

Filter: Which errors are accounted for?

- 1: All errors
- 0: All errors in the Fieldbus system
- 1: Errors in the Fieldbus system, Slot 1
- 2: Errors in the Fieldbus system, Slot 2
- 3: Errors in the Fieldbus system, Slot 3
- 4: Errors in the Fieldbus system, Slot 4
- 5: Errors in the Fieldbus system, Slot 5
- 6: Errors in the Fieldbus system, Slot 6
- 7: Errors in the Fieldbus system, Slot 7
- 8: Errors in the Fieldbus system, Slot 8
- 9: Errors in the online-system
- 10: Errors on the internal drive
- 11: Error in data capture

Each time the function is called, the last occurring error number and the error origin are found, in accordance with the specified filter. If no error occurred, the function returns a value of 0 as the result. The result values can be outputted in a single value variable (see example) or in a device variable (e.g. DisplayVar_01). Instead of a return variable for the error origin, the value 0 is also allowed. Then the error origin is not determined.

Any occurring error number is overwritten the next time the function is called. For purposes of analysis, having an additional variable can be useful (see example).

The function can only be applied in imc Online FAMOS with control commands and dependent upon the device. As a fundamental rule, the function `GetLastError()` may only be called once per device with a particular parameter. If the function `GetLastError( ..., -1 )` is called, there may not be any further calls of the function `GetLastError()`. If the function `GetLastError( ..., 0 )` is called, then only the functions `GetLastError( ..., 9 )` and `GetLastError( ..., 10 )` may additionally be called.

To avoid burdening the device's computational power, it is recommended to call the function `GetLastError()` with a controlling timer (see example).

Excerpt of possible error numbers

CAN-Bus	Description
-5100	Overflow in an int. data repository of the CAN-Bus system, e.g. data red. of channels needed.
-5101	Aggregate sampl. frequ. of the CAN channels is too high, e.g. longer sampl. intervals needed.
-5102	At least one CAN-message has been lost.
-5103	Access to the CAN-Bus has been interrupted due to too many bus errors, e.g. wrong Baud rate.
-5106	Failure to send message during the starting phase of the synchronization.
-5107	Measurement values for a synchr. channel were discarded, e.g. the sampling rate is too low.
-5108	For a synchr. channel, suppl. readings were provided (bus error, or sampl. rate too high).
-5109	A CANSAS module has failed to execute a command, e.g. due to data transmission error.
-5110	No acknowledgement received from a CANSAS module!
-5111	Other bus nodes have been detected. Unable to convert the baud rate!
-5112	The modules are not suitable to the modules connected to the bus. Unable to convert Baud rates!
-5113	After conversion of the Baud rate, a module was no longer found!
-5114	A CANSAS unit's firmware is not up-to-date.
-5115	A received norm display number does not correspond to the configuration. The value is ignored.
-5116	A CAN-channel's bits lie outside of the message received. No value is generated.
-5117	Error in loading a SeedKey-library.
-5118	Unable to find a SeedKey-function.

EtherCAT	Description
-5600	Internal error of imc CRONOSflex DAQ-system.
-5601	Overflow in an internal hardware data volume belonging to the imc CRONOSflex system.
-5602	The aggregate sampling frequency of the channels is too high.
-5603	The number of events (trigger inputs) to calculate is too high.
-5604	The max. channel count has been exceeded. Reduce number of act. channels/Mon. channels.
-5605	Error of imc CRONOSflex system in setting the synchronization.
-5606	Error in synchronizing the CRFX-modules connected to the imc CRONOSflex system bus.
-5607	Synchronization error, VCXO outside of capture range.
-5610	Communication error 1 in the imc CRONOSflex system. Please re-initialize the device.
-5611	Communication error 2 in the imc CRONOSflex system. Please re-initialize the device.
-5612	Communication error 3 in the imc CRONOSflex system. Please re-initialize the device.
-5613	Communication between imc CRONOSflex system and CRFX module failed.
-5614	Communication between imc CRONOSflex system and CRFX module back in order.
-5615	Overload of imc CRONOSflex system. Reduce data load and re-initialize device.
-5616	Error in event calculation in the imc CRONOSflex system.
-5617	A CRFX-module connected to the imc CRONOSflex system is in an invalid state.
-5618	Unable to process data transmitted via the imc CRONOSflex system bus.
-5619	Capture of meas. data by CRFX-modules o the CRONOSflex system bus no longer synchronized.
-5620	Interference on the imc CRONOSflex system bus. Please check the cabling.



Example

; If an error occurred, the error number and slot of the respective last fieldbus
; error are outputted as a virtual channel

```

OnInitAll
    DisplayVar_32 = 0
    int LastError = 0
    int Slot = 0
End

OnTriggerStart( BaseTrigger )
    StartTimerPeriodic( 1, 0.1, 0 )
End

OnTriggerMeasure( BaseTrigger )
    V_Error = CreateVChannelInt( Channel_001, DisplayVar_31 )
    V_Slot = CreateVChannelInt( Channel_001, DisplayVar_32 )
End

OnTimer( 1 )
    GetLastError( LastError, Slot, 0 )
    If LastError = -5613 or LastError = -5618
        DisplayVar_31 = LastError
        DisplayVar_32 = Slot
    End
End

```

GetSampleCount

Sample counting: Determines the number of values in a channel, which were not yet processed.

Samplenumber = **GetSampleCount**(**Channel**)

Samplenumber: Result

Channel: Input channel

If the parameter channel contains no values which apply, the function returns the value 0.



Example

```
OnInitAll
    SC = 0
End
OnTriggerMeasure( BaseTrigger )
    SC = GetSampleCount( Channel_001 )
    If SC > 1
        RecordText( "Proceeding more than 1 sample..." )
    End
End
```

GetSamplingTime

Channel sampling interval: Determines the sampling interval for a physical channel.

Samplinginterval = **GetSamplingTime**(**Inputchannel**)

Samplinginterval: Sampling interval of the physical channel

Inputchannel: Physical channel

The sampling interval can be found for any physical channel.

This means for any analog, incremental counter inputs, fieldbus inputs, as well as digital and digital fieldbus ports.



Example

```
_Samplinginterval_01 = GetSamplingTime( Channel_001 )
```

Greater

Checks whether the first argument is greater than the second argument

a = **Greater**(**b**, **c**)

a = 1 if b is greater than c, otherwise 0.



Example

```
LED_01 = Greater( Channel_001, 8 )
```

The LED lights up whenever the signal value exceeds 8.

GreaterEqual

Checks whether the first argument is greater than or equal to the second argument

a = **GreaterEqual**(**b**, **c**)

a = 1 if b is greater than or equal to c, otherwise 0.

9.2.9.3.8 H

HighLowRatio

High low ratio: The pulse duty ratio of the signal in a window, with reduction.

a = **HighLowRatio**(**b**, **Reduction** [, **Mode**])

a: Result

b: Input channel or input value

Reduction: Reduction depending on the value of "Mode"

Mode: [Optional] Interpretation of the parameter "Reduction"

1: The parameter "Reduction" is the reduction factor expressed in value count

2: The parameter "Reduction" is the sampling interval of the result

When the value 1 is specified for "Mode", the "Reduction"-value is interpreted as the reduction factor (RF). The function calculates the ratio of the count of non-zero values to the count of values of zero, for completed pulses. A complete pulse is a sequence of three edges, an edge being a transition from zero to a non-zero value or vice versa.

Additionally, the data rate is reduced by the factor "Reduction".

Reduction = 1: After each input value, a result value is calculated.

Reduction = 2: After every other input value, a result value is calculated.

Reduction = n: After every n-th input value, a result value is calculated.

Analog signals should be pre-processed (Schmitt-Trigger function ([STri](#)₄₀₈)); see examples).

When the value 2 is specified for "Mode", the "Reduction"-value is interpreted as the target sampling interval of the result channel. The target sampling interval must be the result of dividing the input channel's sampling interval by an integer.

Example: For an input channel sampling interval of 100 ms (0.1), the target sampling interval could be 500 ms (0.5), for example. This would represent a reduction factor (RF) of 5.

The 3rd parameter "Mode" is optional. In the absence of any specification, the default value 1 is applied.

In order to obtain useful results, the function requires a time base. When a channel is set as the 1st parameter, it is provided as the channel's sampling interval. When a single value (e.g. a process vector variable) is set as the 1st parameter, one can obtain a time reference by using timers (or on the device, in synchronous imc Online FAMOS tasks). When using single values as the input value (1st parameter), then for "Mode" = 1 the only possible setting for the reduction factor is 1. "Mode" = 2 is not supported with single values.



Example 1

```
STri1 = STri( Signal_01, 0.5, 4.5 )
Res = HighLowRatio( STri1, 100 )
```

Pre-processing with the Schmitt-Trigger function. Subsequently, the Pulse-Pause-ratio of the pre-processed signal is determined. Only every hundredth result value is outputted.

**Example 2****with optional parameter**

```
STri1 = Stri( Signal_01, 0.5, 4.5 )  
Res = HighLowRatio( STri1, 0.1, 2 )
```

A channel sampled at 1 kHz is pre-processed with the Schmitt-Trigger-function. Subsequently, the Pulse-Pause-ratio of the pre-processed signal is determined. The result has a sampling rate of 10 Hz.

Hyst

Hysteresis: Applies a hysteresis filter to the input channel

a = Hyst(b, Width)

a: Result

Width: Hysteresis width

b: Input channel

If the slope of the input channel changes sign, the output channel follows only if the difference between the last two samples is larger than the hysteresis width. Otherwise the output channel stays constant.

If the slope does not change sign, the result is equal to the input channel. This filters out all vibrations with amplitudes smaller than the hysteresis width.

9.2.9.3.9 I-J

iDiv

Integer division: Division of two integer values. The result is an integer value. Decimal places are omitted.

Result = Numerator iDiv Denominator

Both the numerator and denominator must be of data type Integer, numbers in Real-format are not permitted.

Single value variables of type Integer may only be defined with control commands.



Example

```
Quotient1 = 14 iDiv 4 ; Quotient1 = 3
Quotient2 = 14 / 4 ; Quotient2 = 3.5
```

Integral

a = Integral(b)

The result values are the sum of all current input samples multiplied by the sampling rate.

Integral2

Integral with reset to 0. The return values are the respective sums of all input values accumulated since the last reset to 0 or since the start of the measurement, multiplied by the sampling interval.

Result = Integral2(Channel, Reset)

Result: Integral

Reset: If this value is not 0, the integral is reset to 0

Channel: Input channel

In contrast to the function `Integral`, the function `Integral2` offers the option to reset the integral to 0 during measurement. Otherwise, the behavior of the two functions is the same.



Example

```
Result = Integral2( Channel_001, VrtBit_01 )
```

IntegralFFT

Integral over amplitude spectrum: Integration in the amplitude spectrum's frequency range.

IntFFT = IntegralFFT(Vector)

IntFFT: Result of integration

Vector: Amplitude spectrum Magnitude spectrum

Integration is performed by multiplying the amplitude spectrum with an appropriate weighting function.

For a frequency of 0 Hz, the result of integration is 0 and thus meaningless.



Example

```
vFFT = FFT( InputChannel, 0, 1024 )
IntFFT = IntegralFFT( vFFT )
```

Calculation of an FFT with rectangular window and an FFT-length of 1024. The result is a streaming vector of length 513.

Subsequent integration in the frequency domain of the amplitude spectrum

IntegralP

Precise integral

Prerequisite:

Only available in: imc Online FAMOS!

a = **IntegralP**(**b**)

a: result

b: input channel

The respective result values are the sum of all previous input values multiplied by the sampling interval.

Internally, the precise integral uses a numerical notation with higher precision in order to suppress the compounding of errors in longer integrations.

However, **IntegralP** is significantly slower than the function **Integral**.

IntegralP2

Precise integral with reset to 0. The return values are the respective sums of all input values accumulated since the last reset to 0 or since the start of the measurement, multiplied by the sampling interval.

Prerequisite:

Only available in: imc Online FAMOS!

Result = **IntegralP2**(**Channel**, **Reset**)

Result: Precise Integral

Reset: If this value is not 0, the integral is reset to 0

Channel: Input channel

In contrast to the function **IntegralP**, the function **IntegralP2** offers the option to reset the integral to 0 during measurement.

Otherwise, the behavior of the two functions is the same.

In contrast to the function **Integral2**, the function **IntegralP2** calculates the integral precise.

The function **IntegralP2** requires much more calculating time.



Example

```
Result = IntegralP2( Channel_001, VrtBit_01 )
```

General information about the interval funktions

Interval functions enable analysis of one or more channels in relation to the signal plot of one channel.

- For example, in taking measurements of a motor, the RPM values, vibration, etc. are only interesting if the RPM exceeds a certain level.
- Or, in conjunction with the particular angular ranges of a rotating system, the static values of associated channels are to be determined.

Using the trigger machine and imc Online FAMOS, it would be possible to do this calculation for one range. The interval functions can achieve the same thing with fewer command lines, no trigger and for a variety of ranges.

The example below illustrates this:



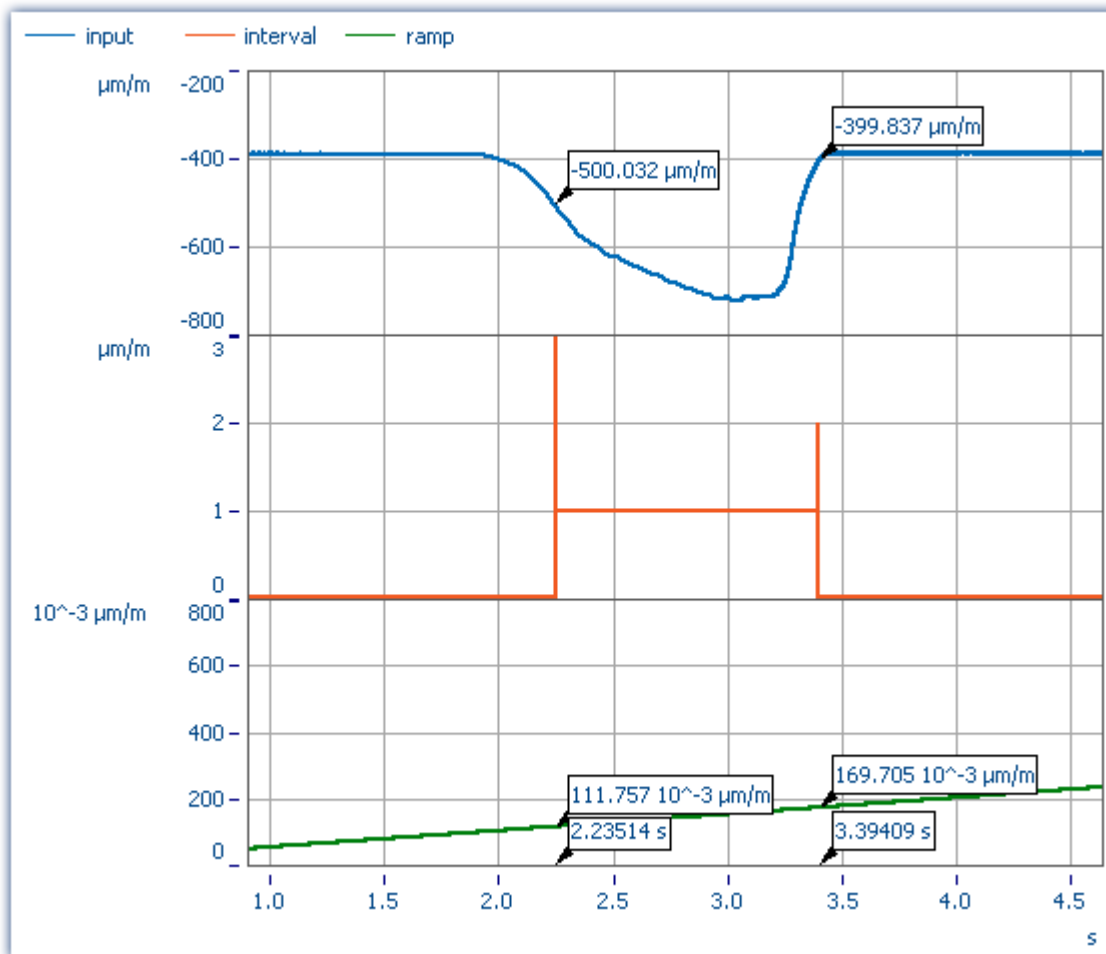
Example

An input channel is analyzed in the interval from one descending edge at $-500 \mu\text{m/m}$ until an ascending edge at $-400 \mu\text{m/m}$. In this interval, the maximum and minimum of a different channel (ramp) are determined.

The program for this appears as follows:

```
ramp      = sawtooth( input, 0, 0.00001, 62800 )
interval  = IntervalFromLevels( input, -500, -400, 1)
Maximum   = IntervalMax( ramp, interval )
Minimum   = IntervalMin( ramp, interval )
```

The interval channel is only an auxiliary channel which can only be used internally.



The functions `IntervalMax`, `IntervalMin`, etc. return exactly one result per interval and have no more time reference.

Maximum =	0.1697 $\mu\text{m/m}$
Minimum =	0.1118 $\mu\text{m/m}$



Note

Result channels created with `IntervalMax`, `-Min`, `Mean` and `RMS` no longer have a time reference and **cannot be sent via XCPoE!**

IntervalFrom1Level

Generates a channel specifying whether the signal's values are within a specified interval.

Result = **IntervalFrom1Level**(**Signal**, **Level**, **Levelcode**)

Result: results channel

Level: interval limit

Signal: input channel

Levelcode: Reserved, always set to 0

Depending on the specified Levelcode, the interval limit must be crossed either in the rising or falling direction to count as entering the interval.

Subsequent interval functions use the results of **IntervalFrom1Level** to limit their calculations to the resulting intervals. The interval channel that is generated by **IntervalFrom1Level** contains, unlike **IntervalFromLevels**, additional data that interpolate the exact point in time when the interval limit is crossed between 2 samples of the given input signal. Subsequent interval functions may evaluate these data to increase their precision.



Example

```
; Determine the mean value of a squared sine signal
; Theoretical value = 0.5, independent of the interval limit
; set in IntervalFrom1Level
saw1 = SawTooth(Channel_001, 0, 0.01, 628)
signal1 = Sin(saw1)
signal2 = signal1 * signal1
interval = IntervalFrom1Level(saw1, 1.0, 0)
iMean1 = IntervalMean(signal2, interval)
```


IntervalFromLevels

Generates a channel stating whether the signal's values are located within the specified interval.

Result = **IntervalFromLevels**(Signal, Level1, Level2, Levelcode)

Result: results channel

Level1: first interval limit

Signal: input channel

Level2: second interval limit

Levelcode: Levelcode

Depending on the specified Levelcode, the first interval limit must be crossed either in the rising or falling direction to count as entering the interval. Correspondingly, the second interval limit must be crossed in the rising or falling direction in order to count as leaving the interval.

Permissible values for Levelcode:

0	Level1 must be crossed in rising direction, so that the interval is considered entered, Level2 must be crossed in rising direction, so that the interval is considered left.
1	Level1 must be crossed in falling direction, so that the interval is considered entered, Level2 must be crossed in rising direction, so that the interval is considered left.
2	Level1 must be crossed in rising direction, so that the interval is considered entered, Level2 must be crossed in falling direction, so that the interval is considered left.
3	Level1 must be crossed in falling direction, so that the interval is considered entered, Level2 must be crossed in falling direction, so that the interval is considered left.

IntervalFromLevels designates the beginning and end of the resulting intervals by special values so that it is possible to append the intervals directly to each other.

Subsequent interval functions use the results of **IntervalFromLevels** to limit their calculations to the resulting intervals.



Example

```
; Determine min and max of a periodic signal between 20 and 40 degrees.
; the angle is obtained from an incremental encoder sensor in Angle-absolute-mode
intervall = IntervalFromLevels( angle, 20, 40, 0 )
iMin = IntervalMin(signal, interval)
iMax = IntervalMax(signal, interval)
```

IntervalFromPulse

Creates an interval channel from a incremental encoder signal.

Result = IntervalFromPulse(Impulse time, Period duration, Multiplier)

Result: Virtual channel with result

Period duration: Length of period

Impulse time: Incremental counter channel set to measurement mode impulse time.

Multiplier: Reserved, always = 0

The input signal comes from an incremental encoder sensor in pulse-time mode. `IntervalFromPulse` uses this to generate interval codes indicating the exact pulse time which can be used by subsequently called functions, for example, ones which use these time points for interpolation.

The interval is entered upon arrival of the first pulse and is only exited when the measurement ends or when more than one pulse arrives during a sampling interval. In the latter case, the system waits for the number of pulses associated with the period length before entering the interval again.

This serves to prevent any phase shift due to periodic data. For nonperiodic data, a period length of 0 must be specified.



Example

```
ivl = IntervalFromPulse( Inc_Counter_01, 60, 0 )
res = IntervalResample( Channel_001, ivl, 360.0, 60, "Deg", 1 )
```

IntervalMax

Determines the input signal's maximum within the specified interval.

Result = IntervalMax(Signal, Interval)

Result: result channel

Interval: interval channel

Signal: input signal

The interval channel is generated by the functions `IntervalFromLevels` or `IntervalFromLevel`.

`IntervalMax` determines the input signal's maximum, while limiting this computation to the intervals specified by the interval channel. Since the time reference is lost, the result cannot be charged any further in many imc Online FAMOS functions and [cannot be sent via XCPoE](#)³³⁷.



Example

```
; Determine min and max of a periodic signal between 20 and 40 degrees.
; the angle is obtained from an incremental encoder sensor in Angle-absolute-mode
interval = IntervalFromLevels( angle, 20, 40, 0 )
iMin = IntervalMin(signal, interval)
iMax = IntervalMax(signal, interval)
```

IntervalMean

Determines the mean of the input signal within the given interval.

Result = IntervalMean(Signal, Interval)

Result: results channel

Interval: interval channel

Signal: input signal

The interval channel is generated by the functions [IntervalFromLevels](#) or [IntervalFromLevel](#).

[IntervalMean](#) determines the mean of the input signal, while limiting this computation to the intervals specified by the interval channel.

[IntervalMean](#) incorporates the additional data provided by [IntervalFromLevel](#) to increase the precision of the computed results.

Since the time reference is lost, the result cannot be charged any further in many imc Online FAMOS functions and [cannot be sent via XCPoE](#) ³³⁷.



Example

```
; Determine the mean of a squared sine signal
; Theoretical value = 0.5, independent of the interval limit specified by
; IntervalFromLevel
saw1 = SawTooth(Channel_001, 0, 0.01, 628)
signal1 = Sin(saw1)
signal2 = signal1 * signal1
interval = IntervalFromLevel(saw1, 1.0, 0)
iMean1 = IntervalMean(signal2, interval)
```

IntervalMin

Determines the input signal's minimum value within the specified interval.

Result = IntervalMin(Signal, Interval)

Result: result channel

Interval: interval channel

Signal: input signal

The interval channel is generated by the functions [IntervalFromLevels](#) or [IntervalFromLevel](#).

[IntervalMin](#) determines the input signal's minimum, while limiting this computation to the intervals specified by the interval channel. Since the time reference is lost, the result cannot be charged any further in many imc Online FAMOS functions and [cannot be sent via XCPoE](#) ³³⁷.



Example

```
; Determine min and max of a periodic signal between 20 and 40 degrees.
; the angle is obtained from an incremental encoder sensor in Angle-absolute-mode
interval = IntervalFromLevels( angle, 20, 40, 0 )
iMin = IntervalMin(signal, interval)
iMax = IntervalMax(signal, interval)
```

IntervalMult

Pulse count multiplier: From the input interval channel, generates a new interval channel with multiple pulse count.

Results = IntervalMult(Intervals, MinFrequency, Multiplier)

Result: Channel containing the multiplied interval codes

MinFrequency: Minimal frequency of the pulses contained, stated in Hz.

Intervals: Input channel containing the original interval codes. Sampling interval stated in s.

Multiplier: The pulse count is multiplied by this factor.

The function is mainly used for signals from sensors which generate one pulses per revolution. Many algorithms (such as Resampling, ...) however require a higher pulse count per revolution. For this purpose, an interval signal is generated from the pulse signal, which this function in turn multiplies by the pulse count. The results can only be processed with other interval functions.

Multiplication only occurs if the frequency of the pulses the signal contains is low enough. The reason is that the results channel can only contain a single pulse per sampling interval. If the input signal has one pulse every 10 samples, for instance, then the maximum multiplication factor is 10, since in that case each result sample contains one pulse. To compensate small variations and to avoid numerical problems, this maximum resulting pulse frequency has to be reduced by the factor 0.999. So, for an input signal of 1000 samples per pulse the multiplier may be at most 999, for 100 samples per pulse at most 99, and for the 10 samples per pulse in the example above at most 9.

If the input signal contains a too rapid sequence of pulses, the pulses are deleted. The function either multiplies, or discards. Thus, in the result, either there are no pulses or the multiplied amount.

Also, the multiplication only takes place if the signal has enough pulses per time unit. The parameter MinFrequency specifies the minimum pulse frequency which is enough for multiplication to take place.

If the pulses arrive at a slower rate, they will be deleted. Thus, the pulse frequency may only lie within the range [MinFrequency ... SamplingFrequency/Multiplier].

The function requires temporary memory of the size [SamplingFrequency / MinFrequency], for which reason the minimum frequency may not be chosen too low.



Example

```
; A drive shaft sensor connected to the incremental counter interface returns
; one pulse per revolution. For subsequent resampling by the angle, however,
; 120 pulses per revolution are needed.
; The shaft spins at 600..6000 RPM. 600 RPM = 10 Hz = MinFrequency.
; SamplingFrequency = 20 kHz > 12 kHz = 120*100 Hz, 100 Hz = 6000 RPM.
ivl1 = IntervalFromPulse(ImpulseTime, 0, 0) ; create intervals
ivl120 = IntervalMult(ivl1, 10, 120) ; multiply pulses
res = IntervalResample(Channel_001, ivl120, 360, 120, "Degree", 1) ; resampling
```

IntervalResample

Resampling of a periodic signal

Result = IntervalResample(Signal, Interval, MaxAngle, AmtPoints, Unit, IsSegmented)

Result: Results channel

Signal: Measurement signal

Interval: Interval signal

MaxAngle: Value of total angle or unit length

AmtPoints: Number of encoder pulses per total angle or unit length

Unit: Value of the result

IsSegmented: Is the result segmented?

0: no

1: yes

Two channels are captured simultaneously which contain time-domain data: one channel with the actual measured signal returning position-dependent measured data, and one channel with the associated interval data.

These interval data were generated from an incremental encoders's signal using the function IntervalFromPulse, which works in the impulse time mode. By means of IntervalResample, values for the measurement signal at the time points of interest are linearly interpolated from the pulse time data. In the process, no more values may be generated than specified by the sampling interval.

MaxAngle is the value to which the number of pulses corresponds, e.g. 60 encoder pulses per 360 degrees, per 2π or per revolution (for oscillation signals) or even 2 encoder pulses per meter.



Example

```
interval = IntervalFromPulse(Inc_Encoder_01, 60, 0) ; Interval codes
res = IntervalResample( Channel_001, interval, 360.0, 60, "Deg", 1) ; resampling
```

IntervalRMS

Determines the root mean square of the input signal within the specified interval.

Result = IntervalRMS(Signal, Interval)

Result: results channel

Interval: interval channel

Signal: input signal

The interval channel is generated by the functions [IntervalFromLevels](#) or [IntervalFrom1Level](#).

[IntervalRMS](#) determines the root mean square of the input signal, while limiting this computation to the intervals specified by the interval channel.

[IntervalRMS](#) incorporates the additional data provided by [IntervalFrom1Level](#) to increase the precision of the computed results.

Since the time reference is lost, the result cannot be charged any further in many imc Online FAMOS functions and [cannot be sent via XCPoE](#) ^[337].



Example

```
; Determine the root mean square of a sine signal
; Theoretical value = 0.707107, independent of the interval limit specified
; by IntervalFrom1Level
saw1      = SawTooth( Channel_001, 0, 0.01, 628 )
signal1    = Sin( saw1 )
interval   = IntervalFrom1Level( saw1, 1.0, 0 )
iRMS1     = IntervalRMS( signal1, interval )
```

IsSynchronized

This function returns whether the internal device clock is synchronized.

Prerequisite:

Only available in: imc Online FAMOS!

a = IsSynchronized(b)

a: Result

1: synchronized

0: not synchronized

-1: unable to determine/no clock present

b: clock, only for establishing the data rate.

JKFlipFlop

JK flip-flop function

a = JKFlipFlop(J, K)

a: Result

J: J input

K: K input

Returns a 1 for the state H and a 0 for the state L. Beginning with the state L, the state H is entered if J is not equal to zero and K equals zero. If J equals zero and K is not equal to zero, then the state L is assumed. If both J and K equal zero, the state remains unchanged. If both J and K are not unequal to zero, then the state changes.

In order to obtain useful results, the function requires a time base. When a channel is set as the 1st and the 2nd parameter, it is provided as the channel's sampling interval. When a single value (e.g. a process vector variable) is set as the 1st and the 2nd parameter, one can obtain a time reference by using timers (or on the device, in synchronous imc Online FAMOS tasks).



Example

```
; The LED flashes while the signal is larger than 9.
res = Greater( Channel_001, 9 )
LED_01 = JKFlipFlop( res, res )
```

9.2.9.3.10 L

LEQ

LEQ-value: The function determines the LEQ-value in dependence on the time. The result states the LEQ for all previously processed values of the input signal.

LEQSignal = LEQ(Signal, Frequency weighting, Reduction[, Mode])

LEQSignal: Rated signal	Frequency weighting: The signal's frequency weighting	Reduction: Reduction depending on the value of "Mode"
Signal: Signal to be evaluated	0: no rating	Mode: [Optional] Interpretation of the parameter "Reduction"
	1: A-rating	1: The parameter "Reduction" is the reduction factor expressed in value count
	2: B-rating	2: The parameter "Reduction" is the sampling interval of the result
	3: C-rating	

The optional "Mode" parameter applies only to imc Online FAMOS. It does not exist for imc Inline FAMOS and is interpreted as 1.

When the value 1 is specified for "Mode", the "Reduction"-value is interpreted as the reduction factor (RF). The function LEQ reduces the count of sample values by the reduction factor. Additionally, the data rate is reduced by the factor "Reduction".

- Reduction = 1: After each input value, a result value is calculated.
- Reduction = 2: After every other input value, a result value is calculated.
- Reduction = n: After every n-th input value, a result value is calculated.

The reduction factor must be an integer > 0.

When the value 2 is specified for "Mode", the "Reduction"-value is interpreted as the target sampling interval of the result channel. The target sampling interval must be the result of dividing the input channel's sampling interval by an integer.

Example: For an input channel sampling interval of 100 ms (0.1), the target sampling interval could be 500 ms (0.5), for example.This would represent a reduction factor (RF) of 5.

The 3th parameter "Mode" is optional.In the absence of any specification, the default value 1 is applied.

At the end of the measurement, the last value of the LEQ-signal displays the LEQ for the entire measurement. The signal is first frequency-weighted, e.g. with an A-rating according to DIN IEC 651.

The function requires a sound signal expressed in the unit Pa (Pascal). The result is referenced to 20*10^-6Pa and expressed in dB. The result can always count as the mean sound pressure level for all data captured to date.

For display in a curve window, representation as "Last value as number" is recommended. This way, the number appearing will gradually level off at the ultimate value, which can then be read at the end of the measurement.

The reduction factor can be selected to generate a few values per second.

**Example 1**

```
LEQ1 = LEQ( Signal, 1, 1000 )
```

The signal is subjected to an A-rating and resampled by the factor 1000. The signal originally has a sampling frequency of 20 kHz, the result has a sampling frequency of only 20 Hz. The result is the average sound pressure level LEQ1, stated in dB.

**Example 2****with optional parameter**

```
LEQ2 = LEQ( Signal, 2, 0.05, 2 )
```

The signal is subjected to a B-rating and resampled. The signal originally has a sampling frequency of 20 kHz, the result has a sampling frequency of only 20 Hz. The result is the average sound pressure level LEQ2, stated in dB.

Less

Checks whether the first argument is less than the second argument

a = Less(b, c)

a = 1 if b is less than c, otherwise 0.

LessEqual

Checks whether the first argument is less than or equal to the second argument

a = LessEqual(b, c)

a = 1 if b is less than or equal to c, otherwise 0.

Ln

Natural logarithm of input channel

a = Ln(b)

The logarithm with the base e (Euler number) is formed.

LogAnd

Logical AND operator applied to B and C.

A = LogAnd(B, C)

A = 1, if B is nonzero and C is nonzero.

A = 0, if B is nonzero and C = 0.

A = 0, if B = 0 and C is nonzero.

A = 0, if B = 0 and C = 0.

LogNot

Logical inverse of B.

A = LogNot(B)

A = 1, if B = 0.

A = 0, if B is nonzero.

LogOr

Logical OR operator applied to B and C.

```
A = LogOr( B, C )
A = 1, if B is nonzero and C is nonzero.
A = 1, if B is nonzero and C = 0.
A = 1, if B = 0 and C is nonzero.
A = 0, if B = 0 and C = 0.
```

LogXor

Logical XOR operator applied to B and C.

```
A = LogXor( B, C )
A = 1, if B = 0 and C is nonzero.
A = 1, if B is nonzero and C = 0.
A = 0, if B = 0 and C = 0.
A = 0, if B is nonzero and C is nonzero.
```

Lower

Returns the lesser value of the arguments

```
a = Lower( b, c )
```

9.2.9.3.11 M

Max

Maximum of a window, with reduction

```
a = Max( b, Windowwidth, Reduction[, Mode] )
```

a: result	Windowwidth: Window width in values
b: Input channel or input value	In imc Online FAMOS: 1 * RF, 2 * RF, ... ,10 * RF
	In imc Inline FAMOS: 1 * RF, 2 * RF, ... ,1000 * RF
	Reduction: Reduction depending on the value of "Mode"
	Mode: [Optional] Interpretation of the parameter "Reduction"
	1: The parameter "Reduction" is the reduction factor expressed in value count
	2: The parameter "Reduction" is the sampling interval of the result

When the value 1 is specified for "Mode", the "Reduction"-value is interpreted as the reduction factor (RF). The function `Max` then calculates the maximum among the last x values, where x is the window width. Additionally, the data rate is reduced by the factor "Reduction".

Reduction = 1: After each input value, a result value is calculated.

Reduction = 2: After every other input value, a result value is calculated.

Reduction = n: After every n-th input value, a result value is calculated.

When the value 2 is specified for "Mode", the "Reduction"-value is interpreted as the target sampling interval of the result channel. The target sampling interval must be the result of dividing the input channel's sampling interval by an integer.

Example: For an input channel sampling interval of 100 ms (0.1), the target sampling interval could be 500 ms (0.5), for example. This would represent a reduction factor (RF) of 5.

The 4th parameter "Mode" is optional. In the absence of any specification, the default value 1 is applied.

Special cases: If the window width is greater than 1*RF, the required count of values is not yet available at the start of the measurement. In this case, processing is applied only the input values available at that point in time.

In order to obtain useful results, the function requires a time base. When a channel is set as the 1st parameter, it is provided as the channel's sampling interval. When a single value (e.g. a process vector variable) is set as the 1st parameter, one can obtain a time reference by using timers (or on the device, in synchronous imc Online FAMOS tasks). When using single values as the input value (1st parameter), then for "Mode" = 1 the only possible setting for the reduction factor is 1. "Mode" = 2 is not supported with single values.



Example 1

```
Res = Max( Signal_01, 20, 4 )
```

Returns the maximum among the last 20 input values for every fourth input value.



Example 2

with optional parameter

```
Res = Max( Signal_01, 20, 0.004, 2 )
```

On a channel sampled at 1 kHz, for every fourth input value the maximum among the last 20 input values is returned. The result has a sampling rate of 250 Hz.

Mean

Mean value of a window, with reduction

a = **Mean**(**b**, **Windowwidth**, **Reduction**[, **Mode**])

a: result

b: Input channel or input value

Windowwidth: Window width in values

In imc Online FAMOS:

1 * RF, 2 * RF, ... ,10 * RF

In imc Inline FAMOS:

1 * RF, 2 * RF, ... ,1000 * RF

Reduction: Reduction depending on the value of "Mode"

Mode: [Optional] Interpretation of the parameter "Reduction"

1: The parameter "Reduction" is the reduction factor expressed in value count

2: The parameter "Reduction" is the sampling interval of the result

When the value 1 is specified for "Mode", the "Reduction"-value is interpreted as the reduction factor (RF). The function **Mean** then calculates the mean value of the last x values, where x is the window width. Additionally, the data rate is reduced by the factor "Reduction".

Reduction = 1: After each input value, a result value is calculated.

Reduction = 2: After every other input value, a result value is calculated.

Reduction = n: After every n-th input value, a result value is calculated.

When the value 2 is specified for "Mode", the "Reduction"-value is interpreted as the target sampling interval of the result channel. The target sampling interval must be the result of dividing the input channel's sampling interval by an integer.

Example: For an input channel sampling interval of 100 ms (0.1), the target sampling interval could be 500 ms (0.5), for example. This would represent a reduction factor (RF) of 5.

The 4th parameter "Mode" is optional. In the absence of any specification, the default value 1 is applied.

Special cases: If the window width is greater than 1*RF, the required count of values is not yet available at the start of the measurement. In this case, processing is applied only the input values available at that point in time.

In order to obtain useful results, the function requires a time base. When a channel is set as the 1st parameter, it is provided as the channel's sampling interval. When a single value (e.g. a process vector variable) is set as the 1st parameter, one can obtain a time reference by using timers (or on the device, in synchronous imc Online FAMOS tasks). When using single values as the input value (1st parameter), then for "Mode" = 1 the only possible setting for the reduction factor is 1. "Mode" = 2 is not supported with single values.

**Example 1**

```
Res = Mean( Signal_01, 6, 2 )
```

Returns the mean value of the last 6 input values for every other input value.

**Example 2****with optional parameter**

```
Res = Mean( Signal_01, 6, 0.002, 2 )
```

On a channel sampled at 1 kHz, for every second input value, the mean of the last 6 input values is returned. The result has a sampling rate of 500 Hz.

Median3

Median of three values: The input channel is smoothed using the median of the last three samples

a = Median3(b)

Each group of three samples is sorted by amplitude in decreasing order. The result is the middle element of the sorted list.

Median5

Median of five values: The input channel is smoothed using the median of the last five samples

a = Median5(b)

Each group of five samples is sorted by amplitude in decreasing order. The result is the middle element of the sorted list.

Min

Minimum of a window, with reduction

a = Min(b, Windowwidth, Reduction[, Mode])

a: result

b: Input channel or input value

Windowwidth: Window width in values

In imc Online FAMOS:

1 * RF, 2 * RF, ... ,10 * RF

In imc Inline FAMOS:

1 * RF, 2 * RF, ... ,1000 * RF

Reduction: Reduction depending on the value of "Mode"

Mode: [Optional] Interpretation of the parameter "Reduction"

1: The parameter "Reduction" is the reduction factor expressed in value count

2: The parameter "Reduction" is the sampling interval of the result

When the value 1 is specified for "Mode", the "Reduction"-value is interpreted as the reduction factor (RF). The function `Min` then calculates the minimum among the last x values, where x is the window width. Additionally, the data rate is reduced by the factor "Reduction".

Reduction = 1: After each input value, a result value is calculated.

Reduction = 2: After every other input value, a result value is calculated.

Reduction = n: After every n-th input value, a result value is calculated.

When the value 2 is specified for "Mode", the "Reduction"-value is interpreted as the target sampling interval of the result channel. The target sampling interval must be the result of dividing the input channel's sampling interval by an integer.

Example: For an input channel sampling interval of 100 ms (0.1), the target sampling interval could be 500 ms (0.5), for example. This would represent a reduction factor (RF) of 5.

The 4th parameter "Mode" is optional. In the absence of any specification, the default value 1 is applied.

Special cases: If the window width is greater than 1*RF, the required count of values is not yet available at the start of the measurement. In this case, processing is applied only the input values available at that point in time.

In order to obtain useful results, the function requires a time base. When a channel is set as the 1st parameter, it is provided as the channel's sampling interval. When a single value (e.g. a process vector variable) is set as the 1st parameter, one can obtain a time reference by using timers (or on the device, in synchronous imc Online FAMOS tasks). When using single values as the input value (1st parameter), then for "Mode" = 1 the only possible setting for the reduction factor is 1. "Mode" = 2 is not supported with single values.



Example 1

```
Res = Min( Signal_01, 20, 4 )
```

Returns the minimum among the last 20 input values for every fourth input value.



Example 2

with optional parameter

```
Res = Min( Signal_01, 20, 0.004, 2 )
```

On a channel sampled at 1 kHz, the minimum among the last 20 input values is returned for every fourth input value. The result has a sampling rate of 250 Hz.

Monoflop

a = **Monoflop**(**b**, **Duration**)

a: result

Duration: Pulse duration in samples; ≥ 1

b: input channel

Monoflop returns for a rising signal edge (a transition from zero to non-zero) a pulse of constant duration (value: 1 during the pulse, else zero).

Not re-triggerable, i.e. signal edges are only evaluated again after the end of a pulse.

In order to obtain useful results, the function requires a time base. When a channel is set as the 1st parameter, it is provided as the channel's sampling interval. When a single value (e.g. a process vector variable) is set as the 1st parameter, one can obtain a time reference by using timers (or on the device, in synchronous imc Online FAMOS tasks).

MonoflopRT

Retriggerable monoflop

a = **MonoflopRT**(**b**, **Duration**)

a: result

Duration: Pulse duration in samples; ≥ 1

b: input channel

MonoflopRT returns for a rising signal edge (a transition from zero to non-zero) a pulse of constant duration (Value: 1 during pulse, else zero)

Signal edges arriving during the pulse prolong it.

In order to obtain useful results, the function requires a time base. When a channel is set as the 1st parameter, it is provided as the channel's sampling interval. When a single value (e.g. a process vector variable) is set as the 1st parameter, one can obtain a time reference by using timers (or on the device, in synchronous imc Online FAMOS tasks).

9.2.9.3.12 N

NorthCorrection

Correction of north transition in a window with WF samples.

a = **NorthCorrection**(**b**, **WF**)

a: result

WF: window size in samples; ≥ 1

b: argument

The function **NorthCorrection** corrects the discontinuity at due North according to the addition method. It prevents skipping from 360° to 0° in the averaging interval.

For values fluctuating around 360°, an average of 180° is avoided.

The result of averaging could lie outside of the wind rose range 0°..360°, e.g. 365°.

The function **WindRoseCorr**^[435] returns the result to the wind rose range (0°..360°), e.g. changes 365° to 5°.

This procedure is conditional on the wind direction not changing more than 100 degrees.

The only sensible way to use the functions **NorthCorrection** and **WindRoseCorr**^[435] is in a combination as illustrated in the example below. Instead of averaging, the standard deviation, i.e. the function **StDev**^[407], can be used.



Example

```
NC = NorthCorrection( Channel, 10 )
NC_Mean = Mean( NC, 10, 10 )
Result = WindRoseCorr( NC_Mean )
```

NOT

Logical inverse of A.

Prerequisite:
Only usable with control commands!

LogicalNot = NOT A

LogicalNot: Result	A: Operand
LogicalNOT = 1, if A = 0.	
LogicalNOT = 0, if A is nonzero.	

The NOT-Operator may only be applied to individual variables.

If applied to whole expressions, for example `If NOT ( pv.x > 0 ) = 0` , the NOT-operator will not return correct results. In such cases, the function `LogNot` should be used instead; thus for example: `If LogNot (pv.x > 0) = 0` .

 Example

```
OnTriggerMeasure( BaseTrigger )
  Value = CurrentValue( Channel_001, 0, 0.0 )
  If NOT Value > 0
    VirtChannel_001 = Channel_001 + 5
  Else
    VirtChannel_001 = Channel_001 + 10
  End
End
```

NumberOfPulses

Returns the number of pulses in reduction window

a = NumberOfPulses(b, Reduction[, Mode])

a: Result	Reduction: Reduction depending on the value of "Mode"
b: Input channel or input value	Mode: [Optional] Interpretation of the parameter "Reduction"
	1: The parameter "Reduction" is the reduction factor expressed in value count
	2: The parameter "Reduction" is the sampling interval of the result

When the value 1 is specified for "Mode", the "Reduction"-value is interpreted as the reduction factor (RF). The function returns the count of pulses within the window of RF values. A pulse is a sequence of two edges, an edge being a transition from zero to non-zero or vice versa.

Additionally, the data rate is reduced by the factor "Reduction".

- Reduction = 1: After each input value, a result value is calculated.
- Reduction = 2: After every other input value, a result value is calculated.
- Reduction = n: After every n-th input value, a result value is calculated.

Analog signals should be pre-processed (Schmitt-Trigger function ([STri](#)₄₀₈)); see examples).

When the value 2 is specified for "Mode", the "Reduction"-value is interpreted as the target sampling interval of the result channel. The target sampling interval must be the result of dividing the input channel's sampling interval by an integer.

Example: For an input channel sampling interval of 100 ms (0.1), the target sampling interval could be 500 ms (0.5), for example. This would represent a reduction factor (RF) of 5.

The 3rd parameter "Mode" is optional. In the absence of any specification, the default value 1 is applied.

In order to obtain useful results, the function requires a time base. When a channel is set as the 1st parameter, it is provided as the channel's sampling interval. When a single value (e.g. a process vector variable) is set as the 1st parameter, one can obtain a time reference by using timers (or on the device, in synchronous imc Online FAMOS tasks). When using single values as the input value (1st parameter), then for "Mode" = 1 the only possible setting for the reduction factor is 1. "Mode" = 2 is not supported with single values.



Example 1

```
STri1 = Stri( Signal_01, 0.5, 4.5 )  
Res = NumberOfPulses( STri1, 100 )
```

Pre-processing with the Schmitt-Trigger function. For every hundredth pre-processed value, the amount of pulses is returned of the last hundred pre-processed signal values is returned.



Example 2

with optional parameter

```
STri1 = Stri( Signal_01, 0.5, 4.5 )  
Res = NumberOfPulses( STri1, 0.1, 2 )
```

A channel sampled at 1 kHz is pre-processed with the Schmitt-Trigger-function. Subsequently, the pulse count of the pre-processed signal is determined. The result has a sampling rate of 10 Hz.

9.2.9.3.13 O

OnECUCmdReturn_ECU_001

See [ECU-functions](#)  316.

OR

Logical OR operator applied to A and B.

Prerequisite:

Only usable with control commands!

LogicalOr = A OR B

LogicalOr: Result	A: 1st operand
	B: 2nd operand

- LogicalOR = 1, if A is nonzero and B is nonzero.
- LogicalOR = 1, if A is nonzero and B = 0.
- LogicalOR = 1, if A = 0 and B is nonzero.
- LogicalOR = 0, if A = 0 and B = 0.

The Or-operator may only be used on operands of type BOOL. A result of type BOOL is returned by the operators <, <=, >, >=, =, <>, AND, OR and NOT.

 Example

OnTriggerMeasure(BaseTrigger)
Value1 = CurrentValue(Channel_001, 0, 0.0)
Value2 = CurrentValue(Channel_002, 0, 0.0)
If Value1 > 0 OR Value2 > 0
 VirtChannel_001 = Channel_001 + 10
Else
 VirtChannel_001 = Channel_001 + 5
End
End

General information about the Otr functions

Prerequisite:

Only available in: imc Online FAMOS!

Device enabled for: "Online Order tracking"

OtrAngleAdd

Addition of an angle to an angle signal

Result = **OtrAngleAdd**(**Angle**, **Add**)

Result: results channel

Add: angle to be added

Angle: angle signal

The function adds a user-specified angle to the angle signal supplied, which is derived from an incremental encoder sensor operating in absolute angle mode. The resulting angle channel can be processed by [OtrResampleAngle](#).

Changes to the angle to be added are always made only when the resulting angle signal's negative signal edge arrives. This results in segments having a negative angle, which are ignored by [OtrResampleAngle](#).



Example

```
; angle offset for zero correction
angle = OtrAngleAdd(Inc_encoder_01, -0.1)
; resampling
res = OtrResampleAngle(Channel_001, angle, 360.0, 360, "Degrees", 1)
```

OtrEncoderPulsesToRpm

Determines the RPM-value: The function determines an RPM time history from a pulse signal.

a = **OtrEncoderPulsesToRpm**(**b**, **Signal_Type**, **EncoderType**, **EncoderPulse**, **RPM_Min**, **Reduction**)

a: Result - RPMHistory: Time-history of RPM-signal

b: Input channel - Pulse_Signal: Time-history of the pulse signal

Signal_Type: Type of pulse signal

0: Number of events

1: Time of pulse

2: Sampled rectangle

3: Sine signal

EncoderType: Type of encoder, how many missing teeth?

0: Default

1: 1 tooth missing

2: 2 teeth missing

EncoderPulse: Pulses per revolution, ≥ 1

RPM_Min: Minimum occurring RPM-value

Reduction: Factor by which the number of data points in the result is less than the input data

The time-history of the RPMs is determined. The function returns 0.0 until at least 2 pulses have been detected. Only then is a computation of RPM possible. The quality of the computation depends on the signal type. For instance, Time of pulse measurement allows very good computation. Since the function is carried out online in the device, it must return an estimated value in some instances (due to unknown location of future).

Notes on Signal_type

- 0 Number of events. The pulse signal reflects the number of pulses counted in a sampling interval. The signal is received if the device's incremental counter input is set to "Event counting". The signal comprises a sequence of integers. Each number represents the number of pulses counted within the current sampling interval. The RPM-value is assumed to be directly proportional to the number of pulses counted. The function smoothes the RPM signal in order to allow a better RPM estimate. If the signal consisted of the sequence of values { ..., 3, 4, 4, 4, 3, 4, 4, 4, 3... }, then for this portion, a RPM-value of approx. 3.75 can be estimated. The function treats each value of the signal as an number of pulses. If the signal consisted of the sequence of values { 0, 0, 1, 0, 1, 2, 1 }, this amounts to 5 pulses detected. This signal type comes with a certain amount of imprecision, since it doesn't reflect at what time within the sampling interval the pulses were counted.

- 1 Time of pulse. The signal is received if the device's incremental counter input is set to "Time of pulse". In this mode, an encoder pulse's exact position in time in the frame of the sampling interval is determined. This involves very precise capture of the pulse. Only this signal type allows such precise sampling in relation to the angle. In this mode it is also important that the measured RPM value be small enough so that there is never more than one encoder pulse per sampling interval. The sampling interval may have to be adjusted to a small enough value for this purpose. RPM_Max must then also be set to a correspondingly low value.

Maximum possible RPM [revs/min] = $60 / (\text{encoder pulses} * \text{sampling interval [s]})$

- 2 Sampled rectangle. One of the device's digital inputs is used to sample the time-history of the encoder's digital output. At every transition from zero to nonzero, the encoder is assumed to have turned by one increment. If the signal consisted of the sequence of values { 0, 0, 1, 1, 1, 0, 0, 1, 1, 0 }, this amounts to 2 pulses detected. If an analog voltage is sampled and there is no hardware comparator, the analog voltage data must be converted to a digital data sequence. If the signal is noisy, it may need to be smoothed beforehand and then have a Schmitt-Trigger performed. This signal type comes with a certain amount of imprecision, since it doesn't reflect at what time within the sampling interval the edge occurred.
- 3 Sine signal. One of the device's digital inputs is used to sample the time-history of the encoder's (sinusoid) analog output. Sinusoid or other signals with zero-crossing in a positive slope can be processed. If the signal is noisy, it may need to be smoothed beforehand and then have a Schmitt-Trigger performed. A zero-crossing in a positive slope is interpreted as a pulse from the encoder. If the signal consisted of the sequence of values { -3.0, -1.0, +1.5, +2.8, +1.3, +0.1, -0.6 }, a pulse is detected at the transition from -1.0 to +1.5.



Note

Notes on EncoderType

Special features of encoders with missing pulses:

The number of encoder pulses is always specified to include the missing tooth. E.g. for an encoder emitting a pulse every 10 degrees of rotation, which should then have 36 teeth, the number of encoder pulses to specify is then also 36. But the encoder only generates 35 pulses because one is missing. Also typical are encoders generating a pulse every 6 degrees. In this case, 60 teeth are to be specified although 2 are missing and only 58 actually present. The first tooth after the gap is interpreted as the zero-pulse. This marks the start of sampling. Encoders with missing pulses can only be used with the signal type "Time of pulse". Recognizing the missing teeth is only possible if the RPM-value is relatively constant in the region around the gap. Particularly with extremely low RPM-values, this cannot be guaranteed. Since in that case, the gap cannot be clearly recognized, the minimum RPM must be set to a non-zero value. At higher RPM-values, the gap is usually clearly recognizable due to mechanical inertia. The function tries to re-synchronize itself after any error in the pulse sequence (or a presumably incorrectly interpreted pulse sequence). However, incorrect sampled values may occasionally occur.



Note

Notes on the RPM-range

If a value for RPM_Min > 0 is set, then signal components up to that RPM-value are disregarded for the result, and whole periods of OrderRef can be skipped in the process. The minimum RPM must not be set too low since the tracking filter is dynamically limited. RPM_Max may only be set so high that the data rate after sampling isn't greater than that of the input data. When OrderRef or Oversampling have high values, this generally is accompanied by a high data rate. In this case, the maximum RPM-value must be set correspondingly low.



Note

Notes on Reduction

The number of data points can be reduced by this factor. Default is 1. Since a high sampling rate sometimes must be used to pinpoint the location of the pulses, a correspondingly large reduction factor can be used to adapt the data rate to the RPMs. This is because the RPM-value often doesn't change so quickly.

**Example 1**

The RPMs of a toothwheel are to be determined. To this end, an inductive encoder is used to generate a rectangular voltage signal. This voltage is recorded at a constant sampling rate of 1 kHz. The voltage is about 0 V .. 3 V in the tooth-gap, and about 18 V .. 22 V at the tip of the tooth. The toothwheel has 8 teeth.

```
_Pulses = stri( Voltage, 15, 5 )
Revs = OtrEncoderPulsesToRpm( _Pulses, 2, 0, 8, 0, 1 )
; _STyp = 2 ; sampled rectangle signal
; _ETyp = 0
; _EPulse = 8 ; number of teeth
; _MinRevs = 0
; _Red = 1
; Revs = OtrEncoderPulsesToRpm( _Pulses, _STyp, _ETyp, _Pulse, _MinRevs, _Red )
```

Since the voltage isn't in an appropriate format, it is altered to produce a proper 0-1 sequence.

**Example 1**

The device records the signal "Tacho1" of an encoder with 128 pulses using the Time-of-pulse mode. The sampling time is 0.1ms.

```
Revs = OtrEncoderPulsesToRpm( Tacho1, 1, 0, 128, 0, 1 )
; _STyp = 1 ; Time-of-pulse
; _ETyp = 0
; _EPulse = 128 ; Pulses per revolution
; _MinRevs = 0
; _Red = 1
; Revs = OtrEncoderPulsesToRpm( Tacho1, _STyp, _ETyp, _EPulse, _MinRevs, _Red )
```

The RPM-value may reach up to 4680 revs/min [= 60 / (128 * 0.0001)].

OtrFrequLine

Determines the frequency line: The function finds the magnitude or phase of a sinusoidal oscillating signal having a fixed period duration.

a = OtrFrequLine(VibrationSignal, PeriodLength, PeriodAmount, Option)

a: Result - Magnitude or phase time history

Option: Determines what is calculated

VibrationSignal: Signal with sinusoidal oscillation

0: Magnitude (effective value)

PeriodLength: Number of samples per period; ≥ 2.0

1: Phase (in degrees)

PeriodAmount: The number of periods over which an average is taken

The function finds the value for the magnitude or for the phase of the oscillation for each interval of length $\text{PeriodLength} * \text{PeriodAmount}$. In other words, the number of oscillation in such an interval is given by PeriodAmount . The duration of the oscillations must be fixed and constant. PeriodLength does not have to be a integer number of samples. But the Product $\text{PeriodLength} * \text{PeriodAmount}$ must be of a integer number of samples in length, determined by $\text{PeriodLength} * \text{PeriodAmount}$ divided by the sampling interval.

The function finds one line of the discrete Fourier-spectrum (DFT) using Hanning-windowing.

If the signal contains significant components at other frequencies, a large number should be set for the number of periods in order to reduce the distorting effects of these other frequencies. If only a small number of periods is used, it may be advisable to connect a bandpass filter in front.

$\text{PeriodLength} \geq 2.0$, $\text{PeriodAmount} \geq 1$. The product of these two parameters may not exceed $2e9$. The phase is stated in the range -180 degrees .. +180 degrees. the value of the phase is 0 degrees for a cosine oscillation, -90 degrees for a sine oscillation.



Example

A vibration signal is sampled in reference to the rotation angle (vib_revs), so that all oscillation components up to the 8th order are present. The signal thus contains 16 points per revolution. The phase of the 1st order is to be determined. One value for the phase is desired for every 5 revolutions.

```
Phase = OtrFrequLine( vib_revs, 16, 5, 1 )
```

OtrFrequLine2

Determining frequency lines: A signal containing an vibration of fixed period duration is approximated by a sinusoidal waveform. The magnitude and phase of this approximation are determined.

OtrFrequLine2(Magnitude, Phase, OscillationSignal, PeriodLength, PeriodCount)

Magnitude: Result of calculation (RMS-value)

VibrationSignal: Input channel with vibration

Phase: Result of calculation (in degrees).

PeriodLength: Number of sampling values in a period; ≥ 2.0

The phase is determined as in the range -180 degrees ... +180 degrees.

PeriodCount: Averging over this many periods; ≥ 1 .

The phase's value is 0 degrees for a cosine oscillation, -90 degrees for a sine oscillation.

The product of PeriodLeng and PeriodCount may not exceed $2e9$.

Within every interval of length $\text{PeriodLength} * \text{PeriodCount}$, this function determines the magnitude and the phase of the oscillation. This interval contains exactly the number PeriodCount oscillations.

The duration of the oscillations must be fixed and constant. The period length does not have to be an integer number of samples. But the Product $\text{PeriodLength} * \text{PeriodAmount}$ must be an integer number of samples in length, determined by $\text{PeriodLength} * \text{PeriodAmount}$ divided by the sampling interval.

The function determines a line of the discrete Fourier transform (DFT) with rectangular windowing.

If the signal additionally contains significant other frequency components, a large number of periods should be selected in order to reduce their distorting effects. When working with a low number of periods, it may be necessary to connect a bandpass filter upstream.



Example

A vibration signal is sampled over the angle (vib_revs), so that all vibration components up to the 8th order are included.

The signal thus composes 16 points per revolution. The magnitude and phase of the 1st order are to be determined.

Every 5 revolutions, a value for the magnitude and phase each is to be returned.

```
OtrFrequLine2( mag, phase, vib_revs, 16, 5 )
```

OtrFrequLine3

Determining frequency lines: A signal containing an vibration of fixed period duration is approximated by a sinusoidal waveform. The magnitude and phase of this approximation are determined. Also, a sinusoidal oscillation is formed based on these results.

OtrFrequLine3(Magnitude, Phase, Sine, VibrationSignal, PeriodLength, PeriodCount, SampleCount)

Magnitude: Result of calculation (RMS-value)

VibrationSignal: Input channel with sinusoidal oscillation

Phase: Result of calculation (in degrees).

PeriodLength: Number of sampling values in a period; ≥ 2.0

The phase is determined as in the range -180 degrees ... +180 degrees.

PeriodCount: Averging over this many periods; ≥ 1 .

The phase's value is 0 degrees for a cosine oscillation, -90 degrees for a sine oscillation.

SampleCount: Length of the sinusoidal oscillation found as the results (in Samples)

Sinus: Results of the calculation (sinusoidal oscillation)

The product of PeriodLeng and PeriodCount may not exceed $2e9$.

The function determines in each interval of length $\text{PeriodLength} * \text{PeriodCount}$ the value of the magnitude and phase of the oscillation. In each such interval, there are PeriodCount oscillations.

The duration of the oscillations must be fixed and constant. The period length does not have to be an integer number of samples. But the Product $\text{PeriodLength} * \text{PeriodAmount}$ must be an integer number of samples in length, determined by $\text{PeriodLength} * \text{PeriodAmount}$ divided by the sampling interval.

The function determines a line of the discrete Fourier transform (DFT) with rectangular windowing.

If the signal additionally contains significant other frequency components, a large number of periods should be selected in order to reduce their distorting effects. When working with a low number of periods, it may be necessary to connect a bandpass filter upstream.

The resulting sinusoidal oscillation can only be used to display instantaneous values. E.g. in the curve window by means of "last N samples". An interpretation of this channel to indicate time is generally not possible.



Example

A vibration signal is sampled over the angle (vib_revs), so that all vibration components up to the 8th order are included. The signal thus composes 16 points per revolution. The magnitude and phase of the 1st order are to be determined. Every 5 revolutions, a value for the magnitude and phase each is to be returned.

A resulting oscillation with 50 points resolution is to be generated.

```
OtrFrequeLine3( mag, phase, sinus vib_revs, 16, 5, 50 )
```

OtrOrderSpectrum

Order spectrum over RPMs: The order spectrum is determined from the vibration and pulse signal time-histories referenced to the RPMs.

OrderSpectrum = OtrOrderSpectrum(Vibration, RPM, RPM_Min, RPM_Max, RPM_Bin_Width, Resolution, OrderMax, AveragingType)

OrderSpectrum: Order spectrum referenced to the RPMs

Vibration: Time-history of vibration signal

RPM: Time-history of the RPMs. In scaled to U/min

RPM_Min: Lower limit of desired RPM-range

RPM_Max: Upper limit of desired RPM-range

RPM_Bin_Width: Width (interval) of an RPM-class

Resolution: Resolution of the order spectrum
[1, 1/2, 1/3, 1/4, ...]

OrderMax: The highest order (line) in the order spectrum

AveragingType: The processing of all spectra within one RPM-class

0: Arithmetic mean

1: Maximum

2: Minimum

RPM_Min, RPM_Max and RPM_Bin_Width are scaled in revs/minute. For sampling, the absolute value of the RPMs is used, but the original RPM value is used for assignment to an RPM-class.

Resolution: 0.1, if 0.1 orders is the distance between lines in the order spectrum. The resolution must be an integer fraction of 1.0, in other words, 1, 1/2, 1/3, 1/4, ... The inverse value of the resolution is a measure of how many revolutions are used to determine the order spectrum. For example, with a resolution of 0.1, each spectrum is determined from 10 revolutions.

The spectral lines are stated as rms-values. Since the FFT is internally calculated from a somewhat larger number of data (a power of 2), some spectral lines are truncated. The first line in the spectrum is always set as zero. Rectangular windowing is used for the FFT.

The averaging works on the magnitude spectrum. The average RPM-value in a spectrum determines the RPM-class.

A tracking Butterworth lowpass filter is used for anti-aliasing. The signal thus filtered is sampled at intervals of rotational angle. Then, an FFT is computed. As a result of this procedure, higher intermediate order lines may show up during sampling than are present in the ultimately returned spectrum. The 3dB-order of the anti-aliasing filter is located, i.e., at the end of the spectrum. The FFT can consist of anywhere between 16 and 8192 points. A rectangular window is used.

The function works properly for:

$\text{OrderMax} < 16 / (\text{SampleTime\_Vibration} * \text{RPM\_Max})$

...

$\text{OrderMax} < 32 / (\text{SampleTime\_Vibration} * \text{RPM\_Max})$

where SampleTime_Vibration is the sampling time of the signal 'Vibration'.

Use the value 16 for a worst case estimate. But, depending on the current number values, it can be as high as 32. The factor of 2 for feasible values is due to the internal rounding up to a power of 2 of values for the FFT used. Obtaining the highest possible order is sometimes possible by changing the resolution.

The RPM's should change only slowly. The RPM's should not fall far below 1% of capacity. The RPM's must be used to compute a good approximation of the rotational angle by means of integration (summing up). Therefore, the RPM values should be reasonably exact.

If no spectrum is computed in a certain RPM class, this spectrum is filled with zeroes. During measurement, the results are transferred like a histogram, i.e., regular but infrequent transmission of intermediary results to the PC. The result is a segmented waveform. Each segment is an order spectrum.



Example

From the time-history of a vibration vib and of the RPMs rpm, the order spectrum is to be determined in relation to the RPMs. The RPM-time trace is sampled at a rate of 0.2 ms.

```
_Min = 1000.0 ; minimum of RPM range
_Max = 6000.0 ; maximum of RPM range
_Delta = 100.0 ; bin spacing of RPM classes
_Res = 0.1 ; spectrum resolution, i.e. 1/10 order visible, calculated
; from 10 revolutions
_OMax = 6.0 ; up to this order, lines in the spectrum are to be displayed.
_MeanType = 0 ; 0 (arithmet. mean)
OSpectrum = OtrOrderSpectrum( vib, rpm, _Min, _Max, _Delta, _Res, _OMax, _MeanType)
; _OMax = 6.0 < 16 / ( 0.0002 * 6000 ) = 13.3
```

OtrOrderSpectrumP

Order spectrum over RPMs: The order spectrum is determined from the vibration and pulse signal time-histories referenced to the RPMs.

a = **OtrOrderSpectrumP**(**Vibration**, **PulseSignal**, **SignalType**, **EncoderType**, **EncoderPulse**, **RPM_Min**, **RPM_Max**, **RPM_Bin_Width**, **Resolution**, **OrderMax**, **AveragingType**)

a: Result - Order spectrum referenced to the RPMs

Vibration: Time-history of vibration signal

PulseSignal: Time-history of pulse signal

SignalType: Type of pulse signal

0: Number of events

1: Time of pulse

2: Sampled rectangle

3: Sine signal

EncoderType: Type of encoder, how many missing teeth?

0: Default

1: 1 tooth missing

2: 2 teeth missing

EncoderPulse: Pulses per revolution, ≥ 1

RPM_Min: Lower limit of desired RPM-range

RPM_Max: Upper limit of desired RPM-range

RPM_Bin_Width: Width (interval) of an RPM-class

Resolution: Resolution of the order spectrum
[1, 1/2, 1/3, 1/4, ...]

OrderMax: The highest order (line) in the order spectrum

AveragingType: The processing of all spectra within one RPM-class

0: Arithmet. mean

1: Maximum

2: Minimum

The RPM-values are derived from the pulse signal by aggregating the angle contribution of each pulse. The vibration signal is sampled in reference to the rotational angle. Then the spectrum is determined. That is the complicated, but exact algorithm.

RPM_Min, RPM_Max and RPM_Bin_Width are scaled in revs/minute. For sampling, the absolute value of the RPMs is used, but the original RPM value is used for assignment to an RPM-class.

Resolution: 0.1, if 0.1 orders is the distance between lines in the order spectrum. The resolution must be an integer fraction of 1.0, in other words, 1, 1/2, 1/3, 1/4, ...

The inverse value of the resolution is a measure of how many revolutions are used to determine the order spectrum. For example, with a resolution of 0.1, each spectrum is determined from 10 revolutions.

The spectral lines are stated as rms-values. Since the FFT is internally calculated from a somewhat larger number of data (a power of 2), some spectral lines are truncated. The first line in the spectrum is always set as zero. Rectangular windowing is used for the FFT.

The averaging works on the magnitude spectrum. The average RPM-value in a spectrum determines the RPM-class.

A tracking Butterworth lowpass filter is used for anti-aliasing. The signal thus filtered is sampled at intervals of rotational angle. Then, an FFT is computed. As a result of this procedure, higher intermediate order lines may show up during sampling than are present in the ultimately returned spectrum. The 3dB-order of the anti-aliasing filter is located, i.e., at the end of the spectrum. The FFT can consist of anywhere between 16 and 8192 points. A rectangular window is used.

The function works properly for:

$\text{OrderMax} < 16 / (\text{SampleTime\_Vibration} * \text{RPM\_Max})$

...

$\text{OrderMax} < 32 / (\text{SampleTime\_Vibration} * \text{RPM\_Max})$

where SampleTime_Vibration is the sampling time of the signal 'Vibration'.

Use the value 16 for a worst case estimate. But, depending on the current number values, it can be as high as 32. The factor of 2 for feasible values is due to the internal rounding up to a power of 2 of values for the FFT used. Obtaining the highest possible order is sometimes possible by changing the resolution.

Notes on signal type

-
- 0 Number of events. The pulse signal reflects the number of pulses counted in a sampling interval. The signal is received if the device's incremental counter input is set to "Event counting". The signal comprises a sequence of integers. Each number represents the number of pulses counted within the current sampling interval. If the signal consisted of the sequence of values { 0, 0, 1, 0, 1, 2, 1 }, this amounts to 5 pulses detected. This signal type comes with a certain amount of imprecision, since it doesn't reflect at what time within the sampling interval the pulses were counted.
-
- 1 Time of pulse. The signal is received if the device's incremental counter input is set to "Time of pulse". In this mode, an encoder pulse's exact position in time in the frame of the sampling interval is determined. This involves very precise capture of the pulse. Only this signal type allows such precise sampling in relation to the angle. In this mode it is also important that the measured RPM value be small enough so that there is never more than one encoder pulse per sampling interval. The sampling interval may have to be adjusted to a small enough value for this purpose. RPM_Max must then also be set to a correspondingly low value.
- $\text{Maximum possible RPM [revs/min]} = 60 / (\text{encoder pulses} * \text{sampling time [s]})$
-
- 2 Sampled rectangle. One of the device's digital inputs is used to sample the time-history of the encoder's digital output. At every transition from zero to nonzero, the encoder is assumed to have turned by one increment. If the signal consisted of the sequence of values { 0, 0, 1, 1, 1, 0, 0, 1, 1, 0 }, this amounts to 2 pulses detected. If an analog voltage is sampled and there is no hardware comparator, the analog voltage data must be converted to a digital data sequence. If the signal is noisy, it may need to be smoothed beforehand and then have a Schmitt-Trigger performed. This signal type comes with a certain amount of imprecision, since it doesn't reflect at what time within the sampling interval the edge occurred.
-
- 3 Sine signal. One of the device's digital inputs is used to sample the time-history of the encoder's (sinusoid) analog output. Sinusoid or other signals with zero-crossing in a positive slope can be processed. If the signal is noisy, it may need to be smoothed beforehand and then have a Schmitt-Trigger performed. A zero-crossing in a positive slope is interpreted as a pulse from the encoder. If the signal consisted of the sequence of values { -3.0, -1.0, +1.5, +2.8, +1.3, +0.1, -0.6 }, a pulse is detected at the transition from -1.0 to +1.5.
-



Note

Notes on EncoderType

Special features of encoders with missing pulses:

The number of encoder pulses is always specified to include the missing tooth. E.g. for an encoder emitting a pulse every 10 degrees of rotation, which should then have 36 teeth, the number of encoder pulses to specify is then also 36. But the encoder only generates 35 pulses because one is missing. Also typical are encoders generating a pulse every 6 degrees. In this case, 60 teeth are to be specified although 2 are missing and only 58 actually present. The first tooth after the gap is interpreted as the zero-pulse. This marks the start of sampling. Encoders with missing pulses can only be used with the signal type "Time of pulse". Recognizing the missing teeth is only possible if the RPM-value is relatively constant in the region around the gap. Particularly with extremely low RPM- values, this cannot be guaranteed. Since in that case, the gap cannot be clearly recognized, the minimum RPM must be set to a non-zero value. At higher RPM-values, the gap is usually clearly recognizable due to mechanical inertia. The function tries to re-synchronize itself after any error in the pulse sequence (or a presumably incorrectly interpreted pulse sequence). However, incorrect sampled values may occasionally occur. The RPM-value should change only slowly. It should not sink substantially below 1% of the maximum value. It follows that the minimum RPM-value must be > 0.0.

If no spectrum is determined for a particular RPM-class, this spectrum is filled with zeroes. During measurement, the result is transferred like a histogram, i.e., regular but few transfers of intermediate results to the PC. The result is a segmented data set. Each segment is an order spectrum.



Example

The order spectrum referenced to the RPMs is to be determined from a vibration signal vib and the pulse signal Inc01. The time signals are sampled at rate of 0.2 ms. Inc01 is recorded in the Time-of-pulse mode of the incremental counter input. The encoder has 8 markings per revolution.

```
OSpectrum = OtrOrderSpectrumP( vib, Inc01, 1, 0, 8, 1000, 6000, 100, 0.1, 6.0, 0 )
; _SType = 1          ; time-of-pulse
; _EType = 0
; _EPulse = 8        ; number of markings
; _Min = 1000.0      ; minimum in RPM range
; _Max = 6000.0      ; maximum in RPM range
; _Delta = 100.0     ; bin-width of RPM classes
; _Res = 0.1         ; resolution of spectrum, meaning 1/10 order is visible,
;                   ; computed over 10 revolutions
; _OMax = 6.0        ; lines in the spectrum up to this order are to be displayed
; _Mean = 0          ; 0 (arithmet. mean)
; OSpec = OtrOrderSpectrumP( vib, Inc01, _SType, _EType, _EPulse, _Min, _Max,
; _Delta, _Res, _OMax, _Mean )
; _OMax = 6.0 < 16 / ( 0.0002 * 6000 ) = 13.3
```

OtrPulseDuration

Pulse time measurement: Measurement of pulse duration for incremental encoders

Result = OtrPulseDuration(InputSignal, DxResult)

Result: Pulse duration

DxResult: Delta-X for the result channel, > 0

InputSignal: Signal from incremental encoder sensor

The time that passes between two pulses of the input signal is determined with a precision of 1/32000 of the sample time. The input signal comes from an incremental encoder sensor that operates in pulse time mode.

An output occurs only when there is a pulse on the input, that is, the result may not be combined with other channels.

When multiple pulses fall into one sample interval, a 0 is returned.



Example

```
; 1 increment per pulse
pulse_dur = OtrPulseDuration( Ink_Geber_01, 1 )

; 10 increments per revolution
pulse_dur = OtrPulseDuration( Ink_Geber_01, 0.1 )

; 10 increments per 360 degrees angle
pulse_dur = OtrPulseDuration( Ink_Geber_01, 36 )
```

OtrResample

Angle sampling: Sampling of a vibration signal in reference to the rotation angle, given a pulse signal. A tracking, anti-aliasing filter is used.

AngleSignal = **OtrResample**(**Vibration**, **Pulse_Signal**, **Signal_Type**, **EncoderType**, **EncoderPulse**, **OrderRef**, **Oversampling**, **Order3dB**, **FilterOrder**, **RPM_Min**, **RPM_Max**, **DelayTime**)

AngleSignal : The signal sampled by angle	EncoderPulse : Pulses per revolution, ≥ 1
Vibration : Time-history of a vibration signal	OrderRef : A visible order (line) in the result
Pulse_Signal : Time-history of the pulse signal	Oversampling : Oversampling for OrderRef
Signal_Type : Type of pulse signal	Order3dB : Order, at which the low-pass filter dampens by 3dB
0: Number of events	FilterOrder : The filter-order of the low-pass filter (1..10)
1: Time of pulse	RPM_Min : Minimum occurring RPM-value
2: Sampled rectangle	RPM_Max : Maximum occurring RPM-value
3: Sine signal	DelayTime : Delaytime, stated in seconds
EncoderType : Type of encoder, how many missing teeth?	
0: Default	
1: 1 tooth missing	
2: 2 teeth missing	

The rotational angle is derived from the pulse signal by aggregating the angle increment of each pulse. The resulting signal trace over the angle is scaled to the x-coordinate, reflecting the number of revolutions made. The x-coordinate starts at 0, has the value of 0.5 after half a revolution, after a full revolution 1.0, after 2 whole revolutions 2.0 etc. The data rate for the result : $0.5 / (\text{OrderRef} * \text{Oversampling})$ The absolute value for the RPMs is used. A Butterworth low-pass filter is used for anti-aliasing.

The function works properly for:

$\text{OrderMax} < 24 / (\text{SampleTime\_Vibration} * \text{RPM\_Max})$

where $\text{SampleTime\_Vibration}$ is the sampling interval of the signal 'Vibration'.

Note that the highest frequency components of the resulting signal are already very heavily dampened.

$\text{Order3dB} \ll \text{OrderMax} = \text{OrderRef} * \text{Oversampling}$

$\ll$ means: much less than.

The RPM's should change only slowly. The RPM's should not fall far below 1% of capacity. The upper cutoff frequency of the low-pass filter must always be substantially below half of the vibration signal's sampling rate. Above about $(0.4 * \text{sampling rate})$, filtering cannot be performed.

If the current RPM-value rises above RPM_Max during measurement, whole oscillations of the order given by the parameter OrderRef are rejected.

The rotation angle is derived from the pulse signal by integration (aggregating the values). The function works with constant accuracy even for long-term measurements and is preferable to the function

[OtrResampleFromRpm](#).

Linear interpolation is used to find intermediate values. Between 2 pulses, the RPM-value is assumed to be constant. Interpolation is possible up to a maximum factor of 100 .

The function disregards the first 32 (approximately) measurement values in the signal.

Notes on signal type

-
- 0 Number of events. The pulse signal reflects the number of pulses counted in a sampling interval. The signal is received if the device's incremental counter input is set to "Event counting". The signal comprises a sequence of integers. Each number represents the number of pulses counted within the current sampling interval. If the signal consisted of the sequence of values { 0, 0, 1, 0, 1, 2, 1 }, this amounts to 5 pulses detected. This signal type comes with a certain amount of imprecision, since it doesn't reflect at what time within the sampling interval the pulses were counted.
-
- 1 Time of pulse. The signal is received if the device's incremental counter input is set to "Time of pulse". In this mode, an encoder pulse's exact position in time in the frame of the sampling interval is determined. This involves very precise capture of the pulse. Only this signal type allows such precise sampling in relation to the angle. In this mode it is also important that the measured RPM value be small enough so that there is never more than one encoder pulse per sampling interval. The sampling interval may have to be adjusted to a small enough value for this purpose. RPM_Max must then also be set to a correspondingly low value.

$$\text{Maximum possible RPM [revs/min]} = 60 / (\text{encoder pulses} * \text{sampling interval [s]})$$
-
- 2 Sampled rectangle. One of the device's digital inputs is used to sample the time-history of the encoder's digital output. At every transition from zero to nonzero, the encoder is assumed to have turned by one increment. If the signal consisted of the sequence of values { 0, 0, 1, 1, 1, 0, 0, 1, 1, 0 }, this amounts to 2 pulses detected. If an analog voltage is sampled and there is no hardware comparator, the analog voltage data must be converted to a digital data sequence. If the signal is noisy, it may need to be smoothed beforehand and then have a Schmitt-Trigger performed. This signal type comes with a certain amount of imprecision, since it doesn't reflect at what time within the sampling interval the edge occurred.
-
- 3 Sine signal. One of the device's digital inputs is used to sample the time-history of the encoder's (sinusoid) analog output. Sinusoid or other signals with zero-crossing in a positive slope can be processed. If the signal is noisy, it may need to be smoothed beforehand and then have a Schmitt-Trigger performed. A zero-crossing in a positive slope is interpreted as a pulse from the encoder. If the signal consisted of the sequence of values { -3.0, -1.0, +1.5, +2.8, +1.3, +0.1, -0.6 }, a pulse is detected at the transition from -1.0 to +1.5.
-

**Note****Notes on OrderRef and Oversampling**

The maximum order in the result is OrderRef * Oversampling

For the maximum order, 2 points in the angle signal are allotted per period. OrderRef denotes the order line in the angle signal which can be clearly recognized. Also, this order should always be displayed in-phase as far as possible even if the RPM-value is so high that some signal components have to be skipped over. If the data rate of the angle signal becomes too high, whole periods of OrderRef are skipped. Oversampling may only be an integer factor ≥ 1 . OrdRef can also be a fractional order.

**Note****Notes on EncoderType****Special features of encoders with missing pulses:**

The number of encoder pulses is always specified to include the missing tooth. E.g. for an encoder emitting a pulse every 10 degrees of rotation, which should then have 36 teeth, the number of encoder pulses to specify is then also 36. But the encoder only generates 35 pulses because one is missing. Also typical are encoders generating a pulse every 6 degrees. In this case, 60 teeth are to be specified although 2 are missing and only 58 actually present.

The first tooth after the gap is interpreted as the zero-pulse. This marks the start of sampling. Encoders with missing pulses can only be used with the signal type "Time of pulse". Recognizing the missing teeth is only possible if the RPM-value is relatively constant in the region around the gap. Particularly with extremely low RPM-values, this cannot be guaranteed. Since in that case, the gap cannot be clearly recognized, the minimum RPM must be set to a non-zero value. At higher RPM-values, the gap is usually clearly recognizable due to mechanical inertia. The function tries to re-synchronize itself after any error in the pulse sequence (or a presumably incorrectly interpreted pulse sequence). However, incorrect sampled values may occasionally occur.

**Note****Notes on the filter**

If no low-pass filter is to be used, the following settings must be made:

`Order3dB = 0`

`FilterOrder = 0`

Always use a 1st or 2nd order filter.

**Note****Notes on the RPM-range**

If a value for `RPM_Min > 0` is set, then signal components up to that RPM-value are disregarded for the result, and whole periods of `OrderRef` can be skipped in the process. The minimum RPM must not be set too low since the tracking filter is dynamically limited.

`RPM_Max` may only be set so high that the data rate after sampling isn't greater than that of the input data. When `OrderRef` or `Oversampling` have high values, this generally is accompanied by a high data rate. In this case, the maximum RPM-value must be set correspondingly low.

**Note****Notes on DelayTime**

`DelayTime = 0.0` Default ≥ 0.0

A delay refers to when the vibration signal lags behind the RPM-signal because of analog circuitry such as filters and amplifiers. A constant lag in the time signal results in a RPM-dependent delay in the angle signal. This can be compensated. The delay time is assumed to be constant. This isn't true exactly, but as a good approximation. The time lag is not a property of the device, but rather of the measurement chain and must be determined empirically by a control measurement.

**Example**

A vibration signal `vib` is sampled at a rate of 1 ms. The RPMs can go up to 4000 revs/min. The vibration is to be plotted over the rotation angle. There should be signal components up to the 5th order. The device's incremental counter input is set to "Time of pulse", the pulse signal is entitled `Inc01`. The encoder has 12 markings per revolution. RPM-values below 10 revs/min are to be ignored.

```
res = OtrResample( vib, Inc01, 1, 0, 12, 5.0, 1, 2.7, 8, 10.0, 4000.0, 0.0 )
```

The 8th-order anti-aliasing filter is designed to dampen the 2.7th order by 3 dB, the 5th order by 60 dB. For the 2.3th order, the amplitude error is already less than 5%.

The following holds: $\text{OrderMax} = 5.0 * 1 \leq 24 / (0.001 * 4000) = 6.0$

Calculation of maximum order: The result has a resolution of $0.5 / \text{OrderMax} = 0.1$ revolution.

It has 10 sample values per revolution.

OtrResampleAngle

Resampling of a periodic signal

Result = OtrResampleAngle(Vibration, Angle, MaxAngle, AmtPoints, Unit, IsSegmented)

Result: results channel

Unit: unit of the results

Vibration: Measurement signal

IsSegmented: are the result data segmented?

Angle: angle signal

0: no

MaxAngle: angle maximum

1: yes

AmtPoints: number of points

Two channels containing time-based data arrive simultaneously: One channel with the actual measurement signal returning periodic measured data, and one channel with associated angle data coming from an incremental encoder sensor operating in absolute angle mode.

Resampling generates a channel which represents the measured data as a function of the angle, where the angle-based data result from interpolation from the time data.

The magnitude of the angle maximum serves the purpose of scaling the results and states whether a revolution is interpreted as 360°, 720° or 2*pi.

The amount of points determines how many measurement points on the angle axis are to be generated by resampling. This number may not be so large that the data volume would be increased by resampling.



Example 1

The signal of the incremental encoder sensor, corrected by -0.1°, is passed to the function `OtrResampleAngle`, together with the periodic measurement signal on Channel_001. The result of resampling is a segmented record that maps the measurement signal to the angle in the range of 0°...360°. The amount of points for one revolution is 360, i.e. the angle resolution is 1°.

```
; angle offset for zero-point correction
angle = OtrAngleAdd(inc_encoder_01, -0.1)
; resampling
res = OtrResampleAngle(Channel_001, angle, 360.0, 360, "°", 1)
```



Example 2

In the following example, the result is displayed over the number of revolutions, i.e. a complete revolution gets a value of 1, and the unit is set to "Rev". The result is resolved with 100 points per revolution. The resulting record is non-segmented, so it counts the number of revolutions along the measurement.

```
res = OtrResampleAngle(Channel_001, inc_encoder_01, 1, 100, "Rev", 0)
```

OtrResampleFromRpm

Angle sampling: A vibration signal is referenced to the rotation angle. The RPM signal must be given.

AngleSignal = OtrResampleFromRpm(Vibration, RPM, OrderMax, RPM_Max)

AngleSignal: Result, the vibration signal plotted over rotation angle

OrderMax: Maximum order (line) contained in result

Vibration: Time history of the vibration signal

RPM_Max: Maximum desired RPM-value in revs/min

RPM: Time history of the RPMs. Scaled in rotations/minute.

The resulting signal trace over the angle is scaled to the x-coordinate, reflecting the number of revolutions made. The x-coordinate starts at 0, has the value of 0.5 after half a revolution, after a full revolution 1.0, after 2 whole revolutions 2.0 etc. The data rate for the result: $0.5 / (\text{OrderRef} * \text{Oversampling})$. The absolute value for the RPMs is used.

The function doesn't come with anti-aliasing filtering. `OtrTrackingLowPass` should be called beforehand. Linear interpolation is used to determine intermediate values.

The function works properly for:

$\text{OrderMax} \leq 30 / (\text{SampleTime\_Vibration} * \text{RPM\_Max})$

where `SampleTime_Vibration` is the sampling interval of the signal 'Vibration'.

The RPM's should change only slowly. The RPM's should not fall far below 1% of the maximum value.

If the momentary RPM-value rises above RPM_Max during the measurement, whole revolutions are rejected. The rotation angle is computed from the RPM-values by means of integration (aggregation), therefore the RPM-values must be available and highly precise.

The signal processor works with 32 bit real numbers at a relative accuracy of 1e-7. In long-term measurements, a variable phase-shift tends to occur.

The same effect appears even more pronounced in the case of inexact RPM-values or values which cannot be integrated to a rotational angle.



Example

A vibration signal vib is sampled every 0.5 ms. The RPM-value rpm can rise to up to 3000 revs/min. The vibration is to be plotted over the angle. Signal components up to the 15th order are supposed to be included.

```
tlp = OtrTrackingLowPass( vib, rpm, 8.0, 4 )
res = OtrResampleFromRpm( tlp, rpm, 15.0, 3000.0 )
```

It is seen that: $\text{OrderMax} = 15.0 \leq 30 / (0.0005 * 3000) = 20.0$

The resolution of the result is $0.5 / \text{OrderMax} = 0.0333$ revolutions. There are 30 samples per revolution.

The anti-aliasing filter is configured to dampen the signal by 3dB in the 8th order, in the 14.4th order by 20 dB. At the 6.1th order the amplitude error is already less than 5%.

OtrRpmComplexOrder

Complex order line: Determines magnitude and phase of an order line referenced to the RPMs. The desired RPM range is divided into classes of equal width.

OtrRpmComplexOrder(Magnitude, Phase, Vibration, Pulse_Signal, Signal_Type, EncoderType, EncoderPulse, RPM_Min, RPM_Max, RPM_Bin_Width, OrderMed, WidthPercent, FilterOrder, Interpolation, DelayTime)

Magnitude: Result, magnitude referenced to RPMs

Phase: Result, phase referenced to RPMs

Vibration: Time-history of vibration signal

Pulse_Signal: Time-history of pulse signal

Signal_Type: Type of pulse signal

0: Number of events

1: Time of pulse

2: Sampled rectangle

3: Sine signal

EncoderType: Type of encoder, how many missing teeth?

0: Default

1: 1 tooth missing

2: 2 teeth missing

EncoderPulse: Pulses per revolution, ≥ 1

RPM_Min: Lower limit of desired RPM-range

RPM_Max: Upper limit of desired RPM-range

RPM_Bin_Width: Width (interval) of an RPM-class

OrderMed: The order (line) to be filtered out

WidthPercent: Width of the bandpass filter in percent, 10..100%

FilterOrder: The filter order of the bandpass filter

Interpolation: Specifies whether the result is interpolated

0: No interpolation

1: Constant interpolation (centered around reference values)

2: Linear interpolation

DelayTime: DelayTime, stated in seconds

The function finds whole periods for the specified order line. For each period, the magnitude, phase and mean RPM-value are determined. These value-pairs are filed in the result. If a value for the particular RPM-value was already filed, an average is taken. The function works with a signal plotted over rotation angle, to which a fixed-frequency bandpass filter is applied.

The root-mean-square value of the magnitude is determined. The phase is stated in the range -180 degrees ... +180 degrees. the value of the phase is 0 degrees for a cosine oscillation, -90 degrees for a sine oscillation. If a "0" is specified as the value for the Magnitude or Phase parameter, the respective quantity is not determined.

The RPM-range starts at RPM_Min, the resolution is always RPM_Bin_Width. The value for RPM_Max is used only to determine the number of result values. RPM_Min, RPM_Max and RPM_Bin_Width must all be scaled in revs/min.

For each RPM-class (of width RPM_Bin_Width) there should be enough periods in the vibration signal. If there are no values, the result for such a RPM-class is zero. If the interpolation parameter is not selected as zero, unfilled RPM-classes are filled by interpolating between adjacent values. If interpolation is selected, even unfilled outer bins (classes at the margins) are filled by means of extension.

The value range for the parameter WidthPercent is [10 ... 100.0]. E.g. at 30% width, the ratio of upper to lower cutoff frequency for the bandpass is 1.30 .

The parameter OrderMed is the order (line) at which the medium frequency of the bandpass is situated. The internally set medium frequency of the filter is:

$$\text{Mediumfrequency} = \text{OrderMed} * (\text{Current\_RPM} / 60)$$

The upper cutoff frequency is above the medium frequency and results for the filter width.

The function works properly for:

$$\text{OrderMax} << 10 / (\text{SampleTime\_Vibration} * \text{RPM\_Max})$$

where SampleTime_Vibration is the sampling time of the signal 'Vibration'.

$$\text{OrderMax} = \text{OrderMed} * \text{sqrt} (1 + \text{WidthPercent} / 100)$$

<< means that the specifiable order should be substantially smaller.

The RPM's should change only slowly. The RPM's should not fall far below 1% of capacity.

Note that the bandpasses need some time for the transients to subside. This time increases strongly with the narrowness of the filter. A bandpass of width 10% is in this sense already extremely narrow. A width of 25% corresponds to a third-octave filter, a width of 100% to an octave filter.

During measurement, the results are transferred like a histogram, i.e., regular but infrequent transmission of intermediary results to the PC. Any interpolation procedure set will only take effect upon completion of the measurement.

Notes on signal type

- 0 Number of events. The pulse signal reflects the number of pulses counted in a sampling interval. The signal is received if the device's incremental counter input is set to "Event counting". The signal comprises a sequence of integers. Each number represents the number of pulses counted within the current sampling interval. If the signal consisted of the sequence of values { 0, 0, 1, 0, 1, 2, 1 }, this amounts to 5 pulses detected. This signal type comes with a certain amount of imprecision, since it doesn't reflect at what time within the sampling interval the pulses were counted.
- 1 Time of pulse. The signal is received if the device's incremental counter input is set to "Time of pulse". In this mode, an encoder pulse's exact position in time in the frame of the sampling interval is determined. This involves very precise capture of the pulse. Only this signal type allows such precise sampling in relation to the angle. In this mode it is also important that the measured RPM value be small enough so that there is never more than one encoder pulse per sampling interval. The sampling interval may have to be adjusted to a small enough value for this purpose. RPM_Max must then also be set to a correspondingly low value.

$$\text{Maximum possible RPM [revs/min]} = 60 / (\text{encoder pulses} * \text{sampling time [s]})$$

-
- 2 Sampled rectangle. One of the device's digital inputs is used to sample the time-history of the encoder's digital output. At every transition from zero to nonzero, the encoder is assumed to have turned by one increment. If the signal consisted of the sequence of values { 0, 0, 1, 1, 1, 0, 0, 1, 1, 0 }, this amounts to 2 pulses detected. If an analog voltage is sampled and there is no hardware comparator, the analog voltage data must be converted to a digital data sequence. If the signal is noisy, it may need to be smoothed beforehand and then have a Schmitt-Trigger performed. This signal type comes with a certain amount of imprecision, since it doesn't reflect at what time within the sampling interval the edge occurred.
-
- 3 Sine signal. One of the device's digital inputs is used to sample the time-history of the encoder's (sinusoid) analog output. Sinusoid or other signals with zero-crossing in a positive slope can be processed. If the signal is noisy, it may need to be smoothed beforehand and then have a Schmitt-Trigger performed. A zero-crossing in a positive slope is interpreted as a pulse from the encoder. If the signal consisted of the sequence of values { -3.0, -1.0, +1.5, +2.8, +1.3, +0.1, -0.6 }, a pulse is detected at the transition from -1.0 to +1.5.
-

**Note****Notes on EncoderType****Special features of encoders with missing pulses:**

The number of encoder pulses is always specified to include the missing tooth. E.g. for an encoder emitting a pulse every 10 degrees of rotation, which should then have 36 teeth, the number of encoder pulses to specify is then also 36. But the encoder only generates 35 pulses because one is missing. Also typical are encoders generating a pulse every 6 degrees. In this case, 60 teeth are to be specified although 2 are missing and only 58 actually present. The first tooth after the gap is interpreted as the zero-pulse. This marks the start of sampling. Encoders with missing pulses can only be used with the signal type "Time of pulse". Recognizing the missing teeth is only possible if the RPM-value is relatively constant in the region around the gap. Particularly with extremely low RPM-values, this cannot be guaranteed. Since in that case, the gap cannot be clearly recognized, the minimum RPM must be set to a non-zero value. At higher RPM-values, the gap is usually clearly recognizable due to mechanical inertia. The function tries to re-synchronize itself after any error in the pulse sequence (or a presumably incorrectly interpreted pulse sequence). However, incorrect sampled values may occasionally occur.

**Note****Notes on DelayTime**

DelayTime = 0.0 Default ≥ 0.0

A delay refers to when the vibration signal lags behind the RPM-signal because of analog circuitry such as filters and amplifiers. A constant lag in the time signal results in a RPM-dependent delay in the angle signal. This can be compensated. The delay time is assumed to be constant. This isn't true exactly, but as a good approximation. The time lag is not a property of the device, but rather of the measurement chain and must be determined empirically by a control measurement.



Example

The magnitude and phase of the 1.5th order line, plotted over the RPMs, are to be determined from this order line's time-history.

Given: vibration signal vib sampled at a rate of 0.0005 ms and the pulse signal Inc01. The device's incremental counter input is set on "Time-of-pulse" mode. The encoder has 8 markings along its circumference.

```
OLine = OtrRpmOrder( vib, Inc01, 1, 0, 8, 1000, 6000, 100, 1.5, 30, 6, 0, 0 )
; _STyp = 1      ; Time-of-pulse
; _ETyp = 0
; _EPulse = 8    ; number of markings
; _Min = 1000.0  ; minimum of RPM-range
; _Max = 6000.0  ; maximum of RPM-range
; _Delta = 100.0 ; width of individual RPM-classes
; _om = 1.5      ; the 1.5th order is selected
; _width = 30    ; 30% of total width
; _fo = 6        ; a 6th-order bandpass filter is used
; _Ipl = 0       ; 0 default (no interpolation)
; _Delay = 0     ; no delay

; OLine = OtrRpmOrder( vib, Inc01, _STyp, _ETyp, _EPulse, _Min, _Max, _Delta, _om,
; _width, _fo, _Ipl, _Delay )
```

Calculation of maximum order: $\text{OrderMax} = 1.5 * \sqrt{1 + 30 / 100} = 1.7$

and thus: $\text{OrderMax} = 1.7 < 10 / (0.0005 * 6000) = 3.33$

OtrRpmOrder

OrderLine: Determines the effective (r.m.s.) value of an order line in relation to the RPMs. The desired RPM range is subdivided in classes of equal width.

RpmOrderLine = OtrRpmOrder(Vibration, RPM, RPM_Min, RPM_Max, RPM_ClassWidth, OrderMiddle, WidthPercent, FilterOrder, Interpolation)

RpmOrderLine: Representation in relationship to RPMs

Vibration: Time history of the vibration signal

RPM: Time history of the RPMs. Scaled in rotations/minute

RPM_Min: Lower end of desired RPM-range

RPM_Max: Upper end of desired RPM-range

RPM_ClassWidth: Bin spacing (class width) of RPM classes

OrderMiddle: The order (line) to be isolated

WidthPercent: Bandpass filter bandwidth in percent, 10% ... 100%

FilterOrder: The filter order of the bandpass filter

Interpolation: Specifies whether the results are interpolated?

0: No interpolation

1: Constant interpolation (centered around auxiliary values)

2: Linear interpolation

The RPM-range always begins with RPM_Min, the resolution is always RPM_ClassWidth. The parameter RPM_Max is only used to specify the number of values in the result. RPM_Min, RPM_Max and RPM_ClassWidth must also be scaled in RPMs.

There should be an adequate number of measurement values in the vibration signal for each RPM-class (of bin spacing RPMClassWidth). Where no values are present, the result for the RPM-class is zero. Only if a nonzero value is selected for the interpolation parameter, the unfilled RPM-classes are filled by means of interpolation of adjacent values. If a form of interpolation is selected, even unfilled classes at the margins are filled by extending the curve at a constant level.

The value range for the parameter WidthPercent is [10 ... 100.0]. For instance, at 30% width, the relationship of upper to lower cutoff frequency of the bandpass is 1.30.

The parameter OrderMiddle is the order (line) in which the medium frequency of the bandpass is located. The internally selected medium frequency of the filter is:

$$\text{MediumFrequency} = \text{OrderMiddle} * (\text{Momentary\_RPM} / 60)$$

The upper cutoff frequency is located above the medium frequency and is derived from the filter bandwidth.

The function works properly for:

$$\text{OrderMax} \ll 10 / (\text{SampleTime\_Vibration} * \text{RPM\_Max})$$

where SampleTime_Vibration is the sampling interval of the signal 'Vibration'.

$$\text{OrderMax} = \text{OrderCenter} * \text{sqrt} (1 + \text{WidthPercent} / 100)$$

<< which indicates that the order to be specified is substantially smaller.

The RPM's should change only slowly. The RPM's should not fall far below 1% of capacity.

Note that bandpasses need a certain amount of time for transients to subside. This time increases markedly as the filter bandpass bandwidth becomes more narrow. In this context, a bandpass bandwidth of 1% is extremely narrow. A width of 25% corresponds to a third-octave filter, a width of 100% to an octave filter.

During measurement, the results are transferred like a histogram, i.e., regular but infrequent transfer of intermediary results to the PC. Any interpolation specified takes effect only after completion of the measurement.

The function works on a signal which is sampled at intervals of rotational angle and to which a fixed frequency bandpass filter is applied, and from which an r-m-s value is then derived for each RPM-class.



Example

The effective (r.m.s.) value of the 1.5th order line in relationship to the RPMs. The input data are the vibration signal vib with the sampling interval of 0.0005 ms and the RPM signal rpm.

```
_RPM_Min = 1000.0 ; RPM-range minimum
_RPM_Max = 6000.0 ; RPM-range maximum
_RPM_Delta = 100.0 ; width of individual RPM-classes
_om = 1.5 ; the 1.5th order is selected
_width = 30 ; 30% of total width
_fo = 6 ; a 6th order bandpass filter is used
_Interpolation = 0 ; 0 default no interpolation)
OLine = OtrRpmOrder( vib, rpm, _RPM_Min, _RPM_Max, _RPM_Delta, _om, _width, _fo,
_Interpolation )
```

Calculation of maximum order: $\text{OrderMax} = 1.5 * \text{sqrt}(1 + 30 / 100) = 1.7$

and thus: $\text{OrderMax} = 1.7 \ll 10 / (0.0005 * 6000) = 3.33$

OtrRpmPresentation

RPM-based signal representation: The time histories of a vibration signal and of the RPM are evaluated to determine the vibration behavior in reference to the RPMs. The desired RPM-range is subdivided into classes of equal width.

RPMRepresentation = OtrRpmPresentation(Vibration, RPM, RPM_Min, RPM_Max, RPM_ClassWidth, Calculation, Interpolation)

RPMRepresentation: Signal recorded in reference to momentary RPM-value

Vibration: Time history of the vibration signal

RPM: Time history of the RPMs. Scaled in rotations/minute.

RPM_Min: Lower end of desired RPM-range

RPM_Max: Upper end of desired RPM-range

RPM_ClassWidth: Bin spacing (class width) of RPM classes

Calculation: How are the values of an RPM-class represented mathematically?

0: Root-mean square (default)

1: Arithmetic mean

2: Minimum

3: Maximum

Interpolation: Is the result interpolated?

0: No interpolation

1: Constant interpolation

2: Linear interpolation

Vibration values corresponding to RPM-values outside of the desired range are ignored (closed outer bins).

The RPM-range always begins with RPM_Min, the resolution is always RPM_ClassWidth.

The parameter RPM_Max is only used to specify the number of values in the result. The results are similar to a histogram in nature, and it is often appropriate to represent them in the form of bar or step graphs.

"Vibration" and "RPM" can either both be in the time domain or in the domain of rotational angle. The RPMs need not be scaled in revs/min. But the RPM-signal, and the parameters RPM_Min, RPM_Max and RPM_ClassWidth must all be scaled to a common y-unit.

There should be an adequate number of measurement values in the vibration signal for each RPM-class (of bin spacing RPMClassWidth). Where no values are present, the result for the RPM-class is zero. Only if a nonzero value is selected for the interpolation parameter, the unfilled RPM-classes are filled by means of interpolation of adjacent values.

If a form of interpolation is selected, even unfilled classes at the margins are filled by extending the curve at a constant level.

During measurement, the results are transferred like a histogram, i.e., regular but infrequent transfer of intermediary results to the PC.

Any interpolation specified takes effect only after completion of the measurement.



Example

The time history of a vibration is to be used to derive a representation of the rms-value of the vibration in reference to the RPMs. Given: vibration signal vib and RPM-signal rpm.

```
rms_rpm = OtrRpmPresentation( vib, rpm, 1000, 6000, 100, 0, 0 )
```

OtrRpmPresentVector

RPM-representation of a spectrum: The time-history of a spectrum and corresponding RPM-signal is determined. The desired RPM-range is divided into bins of equal width.

RPMRepresentation = **OtrRpmPresentVector**(**Spectrum_Sequence**, **RPM**, **RPM_Min**, **RPM_Max**, **RPM_Bin_Width**, **Processing**)

RPMRepresentation: Representation of the spectrum in reference to the RPMs

Spectrum_Sequence: Time-history of a spectrum

RPM: Time-history of RPM signal

RPM_Min: Lower limit of desired RPM-range

RPM_Max: Upper limit of desired RPM-range

RPM_Bin_Width: Width (interval) of an RPM-class

Processing: How are the values from a single RPM-class processed?

0: Arithmetic mean

1: Maximum

2: Minimum

This function takes a sequence of spectra as its input data. This is the spectrum's time-history, determined, for instance with the function [FFT](#). The function also requires the corresponding set of RPM-values. This is the RPM's time-history. There is, then, a corresponding spectrum for each RPM value. Each spectrum-RPM pair is written to the result-matrix. The result-matrix contains a spectrum for each RPM range.

The RPM range always begins at **RPM_Min**, the resolution is **RPM_Bin_Width**. The specification for **RPM_Max** determines the number of values for the result.

If an RPM value is outside of the specified range, the corresponding spectrum is disregarded. (closed outer bins).

There should be enough spectra for each RPM-class (of width **RPM_Bin_Width**). If there is no spectrum, the result in this RPM-class is zero. During the measurement, the results are transmitted like a histogram, i.e. regular but infrequent transfer of intermediate results to the PC.



Example

Given: a vibration signal channel "Vibration" and a RPM-signal channel "Revolutions". Both have the sampling interval 1ms. The spectrum is computed on the basis of the vibrations. This spectrum is to be represented as a function of the RPMs.

```
Spectra      = fft( Vibration, 0, 1024 )
_Revs        = mean( Revolutions, 1024, 1024 )
Spectrum_N = OtrRpmPresentVector( Spectra, _Revs, 1000, 6000, 100, 0 )
;_Min = 1000 ; Minimum of RPM-range
;_Max = 6000 ; Maximum dof RPM-range
;_Delta = 100 ; bin-width of Rpm-classes
;_Calc = 0 ; 0 = Arithmetic mean
; Spectrum_N = OtrRpmPresentVector( Spectra, _Revs, _Min, _Max, _Delta, _Calc )
```

Every 1024 values of the vibration channel, a spectrum is determined. In other words, after 1024 ms a corresponding value for the RPMs is needed. The channel Revs is averaged so as to return one RPM-value per spectrum.

OtrRpmSpectrum

Spectrum: The FFT-spectrum (rms-values!), determined from the time histories of Vibration and RPM in relationship to the RPM's. The desired RPM range is subdivided in classes of equal width.

RPMspectrum = OtrRpmSpectrum(Vibration, RPM, RPM_Min, RPM_Max, RPM_ClassWidth, WindowWidth, WindowType, AveragingType)

RPMspectrum: Spectrum in relation to RPM

Vibration: Time history of the vibration signal

RPM: Time history of the RPMs. Scaled in rotations/minute

RPM_Min: Lower end of desired RPM-range

RPM_Max: Upper end of desired RPM-range

RPM_ClassWidth: Bin spacing (class width) of RPM classes

WindowWidth: Width of the time-window in data points, 100 ... 8192

WindowType: Window function for the FFT used

0: Rectangular

1: Hamming

2: Hanning

3: Blackman

4: Blackman / Harris

5: Flat Top

AveragingType: How are the multiple spectrum computations of each RPM-class averaged?

0: Arithmetic mean

1: Maximum

2: Minimum

RPM_Min, RPM_Max and RPM_ClassWidth are scaled to revs/minute just as RPM. The window width need not be a power of 2. Window widths of, say, 500 or 1000 samples also produce "pretty" frequency line distances. The spectral lines are given as rms-values.

The averaging works on the magnitude spectrum. The mean RPM value during a spectrum determines the RPM-class. Therefore, the RPMs should change only slowly.

If no spectrum is computed in a certain RPM class, this spectrum is filled with zeroes. During measurement, the results are transferred like a histogram, i.e., regular but infrequent transmission of intermediary results to the PC. The result is a segmented waveform. Each segment is a spectrum.



Example

From the time-history of a vibration vib and of the RPMs rpm, the spectrum is to be determined in relation to the RPMs.

```
_RPM_Min = 1000.0 ; minimum of RPM range
_RPM_Max = 6000.0 ; maximum of RPM range
_RPM_Delta = 100.0 ; bin spacing of RPM classes
_WindowWidth = 1000 ; width of window for the FFT, in number of measurement values
_WindowType = 3 ; 0 Rectangular, 3 Blackman
_AveragingType = 0 ; 0 (arithmet. mean)
FFTSpectrum = OtrRpmSpectrum( vib, rpm, _RPM_Min, _RPM_Max, _RPM_Delta, _WindowWidth,
_WindowType, _AveragingType )
```

A spectrum is computed which has a frequency line spacing of 2 Hz, a sampling interval of 0.5ms and a window width of 1000 points.

OtrSynthSin

Internal function for imc Online FRAME.

OtrTrackingLowPass

Smoothing: Tracking low-pass filter. A vibration signal is low-pass filtered, and the filter cutoff frequency depends on the RPM-value.

Filtered = **OtrTrackingLowPass**(**Vibration**, **RPM**, **Order3dB**, **FilterOrder**)

Filtered: Low-pass filtered signal

Vibration: Time history of vibration signal

RPM: Time history of RPMs. Scaled in revs/min

Order3dB: Order at which the low-pass filter dampens the signal by 3dB

FilterOrder: The low-pass filter order

The absolute value for the RPMs is used. The internally set cutoff frequency of the filter is $\text{CutoffFrequency} = \text{Order3dB} * (\text{Current\_RPM} / 60)$

The cutoff frequency must always be substantially less than half the sampling rate of the vibration signal in order for the filtering effect to be achieved.

The function works properly for:

$\text{Order3dB} \ll 24 / (\text{SampleTime\_Vibration} * \max(\text{RPM}))$

where $\text{SampleTime\_Vibration}$ is the sampling interval of the signal 'Vibration' and $\max(\text{RPM})$ is the maximum RPM-value occurring.

$\ll$ means that the specifiable order should be substantially smaller.

Conversely, the maximum RPM-value may not be too high. The RPM's should change only slowly. The RPM's should not fall far below 1% of the maximum value.

The upper cutoff frequency of the low-pass filter must always be substantially less than half the sampling rate of the vibration signal. Above about 0.4 times the sampling rate, no filtering can be performed.



Example

A vibration signal vib is sampled every 0.2 ms. The RPM-signal rpm can get up to 6000 revs/min. Components above the 10th order are to be suppressed.

```
tlp = OtrTrackingLowPass( vib, rpm, 10.0, 6 )
```

A 6th order low-pass filter is used. It dampens by 3dB at the 10th order. It is seen that:

$\text{OrderMax} = 10.0 \ll 24 / (0.0002 * 6000) = 20.0$

The low-pass filter is configured to dampen the signal by 3dB in the 10th order, in the 22nd order by 40 dB. Below the 8.3th order the amplitude error is already less than 5%.

9.2.9.3.14 P

Poll

POLL-Operator: A single value is generated from a DIO-bit in the acquisition mode. This single value can be used like a DIO-bit in the input mode.

DIOBitInput = POLL DIOBitAcquisition

The generated single value can also be subjected to computation operations without the trigger having been released. With control commands, for instance, the single value can be subjected to calculations in the segment [OnTimer](#).



Example

```
LED_01 = LogNot( POLL DIO_Bit01 )
```

If DIO_Bit01 = 0, the first LED in the device lights up although no triggers were released.

Power1

Computes a single-phase power measurement

Power1(InstPower, P, S, Q, PowerFactor, uRMS, iRMS, Time, u, i)

InstPower: Name of the virtual channel with instantaneous power

P - TruePower: Name of the virtual channel with true power data

S - AppPower: Name of the virtual channel with apparent power data

Q - ReacPower: Name of the virtual channel with reactive power data

PowerFactor: Name of the virtual channel with power factor data

uRMS: Channel with RMS voltage data

iRMS: Channel with RMS current data

Time: Variable containing averaging time in seconds (single value)

u: Channel with voltage data

i: Channel with current data

If a channel (e.g. reactive power in example above) is not needed, then enter 0 for its parameter.



Example

```
Power1( IP, P, S, 0, PF, Urms, Irms, 5, Voltage, Current )
```



Reference

A general description is found in the documentation on imc Online FAMOS: "[Information and Tips](#)" > "[Power Measurement](#)".

Power2

Computes a two-phase power measurement

Power2(InstPower, P, S, Q, PowerFactor, uRMS1, iRMS1, uRMS2, iRMS2, Time, u1, i1, u2, i2)

InstPower: Name of the virtual channel with instantaneous power

P - TruePower: Name of the virtual channel with true power data

S - AppPower: Name of the virtual channel with apparent power data

Q - ReacPower: Name of the virtual channel with reactive power data

PowerFactor: Name of the virtual channel with power factor data

uRMS1: Channel with 1st phase RMS voltage data (u1)

iRMS1: Channel with 1st phase RMS current data (i1)

uRMS2: Channel with 2nd phase RMS voltage data (u2)

iRMS2: Channel with 2nd phase RMS current data (i2)

Time: Variable containing averaging time in seconds (single value)

u1: Channel with 1st phase voltage data

i1: Channel with 1st phase current data

u2: Channel with 2nd phase voltage data

i2: Channel with 2nd phase current data

If a channel (e.g. reactive power in example above) is not needed, then enter 0 for its parameter.



Example

```
Power2( IP, P, S, 0, PF, Urms1, Irms1, Urms2, Irms2, 5, Voltage1, Current1, Voltage2, Current2 )
```



Reference

A general description is found in the documentation on imc Online FAMOS: "[Information and Tips](#)" > "[Power Measurement](#)".

Power3

Computes a three-phase power measurement

Power3(InstPower, P, S, Q, PowerFactor, uRMS1, iRMS1, uRMS2, iRMS2, uRMS3, iRMS3, Time, u1, i1, u2, i2, u3, i3)

InstPower : Name of the virtual channel with instantaneous power	uRMS3 : Channel with 3nd phase RMS voltage data (u3)
P - TruePower : Name of the virtual channel with true power data	iRMS3 : Channel with 3nd phase RMS current data (i3)
S - AppPower : Name of the virtual channel with apparent power data	Time : Variable containing averaging time in seconds (single value)
Q - ReacPower : Name of the virtual channel with reactive power data	u1 : Channel with 1st phase voltage data
PowerFactor : Name of the virtual channel with power factor data	i1 : Channel with 1st phase current data
uRMS1 : Channel with 1st phase RMS voltage data (u1)	u2 : Channel with 2nd phase voltage data
iRMS1 : Channel with 1st phase RMS current data (i1)	i2 : Channel with 2nd phase current data
uRMS2 : Channel with 2nd phase RMS voltage data (u2)	u1 : Channel with 3nd phase voltage data
iRMS2 : Channel with 2nd phase RMS current data (i2)	i1 : Channel with 3nd phase current data

If a channel (e.g. reactive power in example above) is not needed, then enter 0 for its parameter.

 Example

`Power3( IP, P, S, 0, PF, Urms1, Irms1, Urms2, Irms2, Urms3, Irms3, 5, Voltage1, Current1, Voltage2, Current2, Voltage3, Current3)`

 Reference

A general description is found in the documentation on imc Online FAMOS: "*Information and Tips*" > "[Power Measurement](#)²⁷⁵".

PulseDuration

Pulse duration: A signal's average pulse duration in the window, with resampling.

a = **PulseDuration**(**b**, Reduction[, Mode])

a : Result	Reduction : Reduction depending on the value of "Mode"
b : Input channel or input value	Mode : [Optional] Interpretation of the parameter "Reduction" 1: The parameter "Reduction" is the reduction factor expressed in value count 2: The parameter "Reduction" is the sampling interval of the result

When the value 1 is specified for "Mode", the "Reduction"-value is interpreted as the reduction factor (RF). The function applies to a window of RF values. The average pulse duration is derived from all completed pulses in the window. A complete pulse is a sequence of three edges, an edge being a transition from zero to a non-zero value or vice versa.

If no pulse is completed within the window, the last pulse duration is returned; or, if the window width times the number of windows without pulses is greater than the last pulse duration, the product of the amount and width of window is returned.

Additionally, the data rate is reduced by the factor "Reduction".

Reduction = 1: After each input value, a result value is calculated.

Reduction = 2: After every other input value, a result value is calculated.

Reduction = n: After every n-th input value, a result value is calculated.

Analog signals should be pre-processed (Schmitt-Trigger function ([STri](#)⁴⁰⁸)); see examples).

When the value 2 is specified for "Mode", the "Reduction"-value is interpreted as the target sampling interval of the result channel. The target sampling interval must be the result of dividing the input channel's sampling interval by an integer.

Example: For an input channel sampling interval of 100 ms (0.1), the target sampling interval could be 500 ms (0.5), for example. This would represent a reduction factor (RF) of 5.

The 3rd parameter "Mode" is optional. In the absence of any specification, the default value 1 is applied.

In order to obtain useful results, the function requires a time base. When a channel is set as the 1st parameter, it is provided as the channel's sampling interval. When a single value (e.g. a process vector variable) is set as the 1st parameter, one can obtain a time reference by using timers (or on the device, in synchronous imc Online FAMOS tasks). When using single values as the input value (1st parameter), then for "Mode" = 1 the only possible setting for the reduction factor is 1. "Mode" = 2 is not supported with single values.



Example 1

```
STri1 = STri( Signal_01, 0.5, 4.5 )
Res = PulseDuration( STri1, 100 )
```

Pre-processing with the Schmitt-Trigger function. For every hundredth pre-processed value, the mean pulse duration of the last hundred pre-processed signal values is returned.



Example 2

with optional parameter

```
STri1 = STri( Signal_01, 0.5, 4.5 )
Res = PulseDuration( STri1, 0.01, 2 )
```

A channel sampled at 1 kHz is pre-processed with the Schmitt-Trigger- function. For every hundredth pre-processed value, the mean pulse duration of the last hundred pre-processed signal values is returned. The result's sampling rate is 100 Hz.

PulseFrequency

Pulse frequency: The average pulse frequency of a signal in the window with resampling.

a = **PulseFrequency**(**b**, **Reduction**[, **Mode**])

a : Result	Reduction : Reduction depending on the value of "Mode"
b : Input channel or input value	Mode : [Optional] Interpretation of the parameter "Reduction" 1: The parameter "Reduction" is the reduction factor expressed in value count 2: The parameter "Reduction" is the sampling interval of the result

When the value 1 is specified for "Mode", the "Reduction"-value is interpreted as the reduction factor (RF). The function applies to a window of RF values. The average pulse frequency is derived from all completed pulses within the window. A complete pulse is a sequence of three edges, an edge being a transition from zero to a non-zero value or vice versa.

If no pulse is completed within the window, the last pulse frequency is returned; or, if the result's sampling frequency divided by the number of the windows without pulses is less than the last pulse frequency, the quotient of the sampling frequency and the number of windows is returned.

- Additionally, the data rate is reduced by the factor "Reduction".
- Reduction = 1: After each input value, a result value is calculated.
 - Reduction = 2: After every other input value, a result value is calculated.
 - Reduction = n: After every n-th input value, a result value is calculated.

Analog signals should be pre-processed (Schmitt-Trigger function ([STri](#)⁴⁰⁸); see examples).

When the value 2 is specified for "Mode", the "Reduction"-value is interpreted as the target sampling interval of the result channel. The target sampling interval must be the result of dividing the input channel's sampling interval by an integer.

Example: For an input channel sampling interval of 100 ms (0.1), the target sampling interval could be 500 ms (0.5), for example. This would represent a reduction factor (RF) of 5.

The 3rd parameter "Mode" is optional. In the absence of any specification, the default value 1 is applied.

In order to obtain useful results, the function requires a time base. When a channel is set as the 1st parameter, it is provided as the channel's sampling interval. When a single value (e.g. a process vector variable) is set as the 1st parameter, one can obtain a time reference by using timers (or on the device, in synchronous imc Online FAMOS tasks). When using single values as the input value (1st parameter), then for "Mode" = 1 the only possible setting for the reduction factor is 1. "Mode" = 2 is not supported with single values.

**Example 1**

```
STri1 = STri( Signal_01, 0.5, 4.5 )
Res = PulseFrequency( STri1, 100 )
```

Pre-processing with the Schmitt-Trigger function. For every hundredth pre-processed value, the mean pulse frequency of the last hundred pre-processed signal values is returned.

**Example 2****with optional parameter**

```
STri1 = STri( Signal_01, 0.5, 4.5 )
Res = PulseFrequency( STri1, 0.01, 2 )
```

A channel sampled at 1 kHz is pre-processed with the Schmitt-Trigger- function. For every hundredth pre-processed value, the mean pulse frequency of the last hundred pre-processed signal values is returned. The result's sampling rate is 100 Hz.

PulsePhase

The average phase difference between two signals in a window, with resampling.

a = PulsePhase(b, c, Reduction[, Mode])

a: Result

b: Input channel 1 or Input value 1

c: Input channel 2 or Input value 2

Reduction: Reduction depending on the value of "Mode"

Mode: [Optional] Interpretation of the parameter "Reduction"

1: The parameter "Reduction" is the reduction factor expressed in value count

2: The parameter "Reduction" is the sampling interval of the result

When the value 1 is specified for "Mode", the "Reduction"-value is interpreted as the reduction factor (RF). The function applies to a window of RF values. The phase shift is calculated for each pair of corresponding positive slopes and the difference is stated in seconds. A positive slope refers to a transition from zero to non-zero.

Additionally, the data rate is reduced by the factor "Reduction".

Reduction = 1: After each input value, a result value is calculated.

Reduction = 2: After every other input value, a result value is calculated.

Reduction = n: After every n-th input value, a result value is calculated.

Analog signals should be pre-processed (Schmitt-Trigger function ([STri](#)⁴⁰⁸)); see examples).

When the value 2 is specified for "Mode", the "Reduction"-value is interpreted as the target sampling interval of the result channel. The target sampling interval must be the result of dividing the input channel's sampling interval by an integer.

Example: For an input channel sampling interval of 100 ms (0.1), the target sampling interval could be 500 ms (0.5), for example. This would represent a reduction factor (RF) of 5.

The 3rd parameter "Mode" is optional. In the absence of any specification, the default value 1 is applied.

In order to obtain useful results, the function requires a time base. When a channel is set as the 1st parameter, it is provided as the channel's sampling interval. When a single value (e.g. a process vector variable) is set as the 1st parameter, one can obtain a time reference by using timers (or on the device, in synchronous imc Online FAMOS tasks). When using single values as the input value (1st parameter), then for "Mode" = 1 the only possible setting for the reduction factor is 1. "Mode" = 2 is not supported with single values.



Example 1

```
STri11 = STri( Signal_01, 0.5, 4.5 )  
STri12 = STri( Signal_02, 1.0, 9.0 )  
Res = PulsePhase( STri1, STri2, 100 )
```

Pre-processing with the Schmitt-Trigger function. For every hundredth pre-processed value, the mean phase shift of the last hundred pre-processed signal values is returned.



Example 2

with optional parameter

```
STri11 = STri( Signal_01, 0.5, 4.5 )  
STri12 = STri( Signal_02, 1.0, 9.0 )  
Res = PulsePhase( STri1, STri2, 0.01, 2 )
```

Two channels sampled at 1 kHz are pre-processed with the Schmitt-Trigger-function. For every hundredth pre-processed value, the mean phase shift of the last hundred pre-processed signal values is returned. The result's sampling rate is 100 Hz.

9.2.9.3.15 R

RangeMax

Returns the input channel's **upper measurement range limit**.

a = RangeMax(**b**)

a: Upper limit of input channel measurement range (single value) **b**: input channel



Note

Balancing

After channel balancing, this function only returns the **correct value**, if **"Prepare" was performed**.

Background: Balancing can be performed either during measurement or before a measurement.

Afterwards, "Prepare" is not always necessary. In these cases, the measurement range is not determined correctly. Instead, the same measurement range which had been displayed before the balancing continues to be displayed.

RangeMin

Returns the input channel's **lower measurement range limit**.

a = RangeMin(**b**)

a: Lower limit of input channel measurement range (single value) **b**: input channel



Note

Balancing

After channel balancing, this function only returns the **correct value**, if **"Prepare" was performed**.

Background: Balancing can be performed either during measurement or before a measurement.

Afterwards, "Prepare" is not always necessary. In these cases, the measurement range is not determined correctly. Instead, the same measurement range which had been displayed before the balancing continues to be displayed.

ReadyForPowerOff

Device ready for deactivation: The device can be turned off now.

Prerequisite:

Only available in: imc Online FAMOS!

Only usable with control commands!

The function `ReadyForPowerOff` may only be called in the control command `OnPowerOff`.

ReadyForPowerOff()

This function is called in the section `OnPowerOff` which is activated when the device is turned off. The device's internal battery then provides the power for a maximum of 8 s. Within this time, the program can be closed correctly in the section `OnPowerOff`. When the function `ReadyForPowerOff` is called, the device is ready to be turned off. Before the device is really turned off, it may be necessary to conclude other processes, e.g. saving to the hard drive.



Example

```
OnSyncTask( 0.01 ) ; cycle time of 0.01s
    DisplayVar_01 = pv.Channel_001
End

OnPowerOff( 0.01 ) ; cycle time of 0.01s
    DisplayVar_01 = 0
    DOut01_Bit01  = 0
    DOut01_Bit02  = 0
    ReadyForPowerOff()
End
```

RecordEvent

Record event: Records a transition from zero to not zero as an event

Prerequisite:

There must be at least one channel assigned to the "*BaseTrigger*". Texts can be recorded only while channels assigned to the "*BaseTrigger*" are measuring.

RecordEvent(b, "c")

b: Event channel

c: Event text

As a result a TimeStampAscii channel (OfaEvents) will be created. The event messages can be displayed with a curve window as table or graphical over time.

For each device/imc Inline FAMOS-task, only **one log channel is created**. The function can be called multiple times, and all entries it makes are deposited in this channel.



Example

```
RecordEvent( Greater( Channel_001, 9 ), "High water mark passed" )
```

If the level on Channel_001 exceeds 9, this is recorded as an event.



Note

Modifying the name

Tip: Assign an appropriately distinct name to the channel. The log channel has a default name which cannot be changed by means of the function (e.g. "*OFA_Events*"). To do this, open the imc Inline FAMOS-Editor and there, open "*Properties*" (F5). Here, you can change the log-channel's name.

RecordText

Return text: The specified text is output.

Prerequisite:

For imc Online FAMOS, observe the following: There must be at least one channel assigned to the "*BaseTrigger*". Texts can be recorded only while channels assigned to the "*BaseTrigger*" are measuring.

RecordText ("Text")

Text: Text to output

This function should only be used as a message following certain events, for example when a certain value in the control command Switch is reached. If the function is called repeatedly, the output memory for texts can quickly overflow.



Example

```
OnInitAll
    Value = 1
    StartTimerPeriodic( 5, 0.1, 0.1 )
End

OnTimer( 5 )
    Switch Value
        Case 1
            RecordText( "Value = 1" )
            Value = 2
        End
        Case 2
            RecordText( "Value = 2" )
            Value = 3
        End
        Default
            Value = Value + 1
        End
    End
End
```

In contrast to the [RecordEvent](#)-function, the [RecordText](#) function always outputs the text (in the [RecordEvent](#) function, a transition of the channel parameter from 0 to non-zero is necessary for output of text).

For each device/imc Inline FAMOS-task, only **one log channel is created**. The function can be called multiple times, and all entries it makes are deposited in this channel.



Note

Modifying the name

Tip: Assign an appropriately distinct name to the channel. The log channel has a default name which cannot be changed by means of the function (e.g. "*OFA_Events*"). To do this, open the imc Inline FAMOS-Editor and there, open "*Properties*" (F5). Here, you can change the log-channel's name.

Red

Reduction: reduces the number of values in the input channel

a = **Red**(**b**, **Reduction**[, **Mode**])

a: Result

b: Input channel

Reduction: Reduction depending on the value of "Mode"

Mode: [Optional] Interpretation of the parameter "Reduction"

1: The parameter "Reduction" is the reduction factor expressed in value count

2: The parameter "Reduction" is the sampling interval of the result

When the value 1 is specified for "Mode", the "Reduction"-value is interpreted as the reduction factor (RF). The function **Red** reduces the count of sample values by the reduction factor. Additionally, the data rate is reduced by the factor "Reduction".

Reduction = 1: After each input value, a result value is calculated.

Reduction = 2: After every other input value, a result value is calculated.

Reduction = n: After every n-th input value, a result value is calculated.

The reduction factor must be an integer > 0. To avoid aliasing effects, the input channel must previously have been filtered with a low-pass filter.

When the value 2 is specified for "Mode", the "Reduction"-value is interpreted as the target sampling interval of the result channel. The target sampling interval must be the result of dividing the input channel's sampling interval by an integer.

Example: For an input channel sampling interval of 100 ms (0.1), the target sampling interval could be 500 ms (0.5), for example. This would represent a reduction factor (RF) of 5.

The 3rd parameter "Mode" is optional. In the absence of any specification, the default value 1 is applied.



Example 1

```
Res = Red( FiltLP( Signal_01, 0, 0, 4, 125 ), 4 )
```

A channel sampled at 1 kHz is reduced by the factor 4. The result has a sampling rate of 250 Hz.

Previously, low-pass filtering whose cutoff frequency is half of the resulting sampling frequency, i.e. 125 Hz.



Example 2

with optional parameter

```
Res = Red( FiltLP( Signal_01, 0, 0, 4, 125 ), 0.004, 2 )
```

A channel sampled at 1 kHz is reduced such that the result has a sampling rate of 250 Hz. Previously, low-pass filtering whose cutoff frequency is half of the resulting sampling frequency, i.e. 125 Hz.

ReduceDataRate

Reducing the data rate: A channel's data rate is reduced to the specified data rate.

Prerequisite:

Only available in: imc Online FAMOS!

a = **ReduceDataRate**(**b**, **Datarate**)

a: Result

b: Channel with measured values

Datarate: Data rate in Hz, ≥ 2

This function is applied to Fieldbus channels which were sampled at a slightly higher data rate than necessary. Additionally, the channel contains some duplicate measured values. This function eliminates these duplicate measured values as far as possible while still achieving the target data rate.

Both the input channel's data rate and the target data rate must be integer numbers. The ratio of the input channel's data rate to the target data rate may not exceed a factor of 2.5.



Example

```
res = ReduceDataRate( Signal_500Hz, 300 )
; Reduces the data rate of a 500Hz-channel Signal_500Hz to 300 Hz.
```

ReplaceFirstValues0

Replaces signal values with the value 0.0: During the phase of signal transients due to filtering at the measurement's start, the initial signal values are replaced with 0.0.

Prerequisite:

Only available in: imc Inline FAMOS!

a = **ReplaceFirstValues0**(**Signal**, **Count**)

a: Result

Count: Number of values to replace

Signal: Input channel

This function is used to suppress the undesired signal fluctuations due to filtering, which occur at the measurement's start. For this purpose, as many initial values are replaced with 0.0 as specified in the parameter Count.

The signal parameter may be either a channel or a single value (e.g. a pv-variable).

In contrast to the function [SkipFirstValues](#), the values during the transients' phase are replaced instead of skipped.



Example

```
Signal_Filtered = FiltHP( Signal_01, 1, 0, 4, 100 )
Signal_Corrected = ReplaceFirstValues0( Signal_Filtered, 1000 )
```

A high-pass filter is applied to the sine signal Signal_01 (the sampling rate of Signal_01 is 1 kHz).

When the measurement is started, a signal spike occurs (whose height depends on the phasing).

This signal spike would not occur once the filter's transients had subsided.

The transients are suppressed for the duration of one second.

For this purpose, the first 1000 values of the filtered signal are replaced with the value 0.0.

ReplaceFirstValuesN

Replaces signal values with n-th value: During the filter's transients phase at the measurement's start, the initial signal values are replaced with the n-th value.

Prerequisite:

Only available in: imc Inline FAMOS!

a = ReplaceFirstValuesN(Signal, Count)

a: Result

Count: Number of values to replace

Signal: Input channel

This function is used to suppress the undesired signal fluctuations due to filtering, which occur at the measurement's start. At the start of the measurement, this function does not return any results. After the first n values accumulated, all previous values are retroactively replaced with the n-th value.

Only channels are permitted as the signal parameter.

In contrast to the function [SkipFirstValues](#), the values during the transients' phase are replaced instead of skipped.



Example

```
Signal_Filtered = FiltHP( Signal_01, 1, 0, 4, 100 )
Signal_Corrected = ReplaceFirstValuesN( Signal_Filtered, 1000 )
```

A high-pass filter is applied to the sine signal Signal_01 (the sampling rate of Signal_01 is 1 kHz). When the measurement is started, a signal spike occurs (whose height depends on the phasing). This signal spike would not occur once the filter's transients had subsided. The transients are suppressed for the duration of one second. For this purpose, the first 1000 values of the filtered signal are replaced with the 1000th value.

ReSample

Sampling rate conversion: Re-sampling of vectorized data.

a = ReSample(b, VectorLen)

a: result

VectorLen: Length of result vectors

b: Vectorized input channel

The vectors of a vectorized input channel are modified to a new vector length by means of resampling. The new vector elements are determined by interpolating from the existing vector elements. The curve shape remains unchanged as a result of the resampling



Example

```
FFT513 = FFT( Signal, 0, 1024 )
Res500 = ReSample( FFT513, 500 )
```

Vectorization of a signal with a vector length of 1024, and calculation of the FFT. Subsequent alteration of the result's vector length from 513 to 500 by means of resampling.

RMS

Root-mean-square value of a window, with reduction

a = **RMS**(**b**, **Windowwidth**, **Reduction**[, **Mode**])

a: Result

b: Input channel or input value

Windowwidth: Window width in values

In imc Online FAMOS:

1 * RF, 2 * RF, ... ,10 * RF

In imc Inline FAMOS:

1 * RF, 2 * RF, ... ,1000 * RF

Reduction: Reduction depending on the value of "Mode"

Mode: [Optional] Interpretation of the parameter "Reduction"

1: The parameter "Reduction" is the reduction factor expressed in value count

2: The parameter "Reduction" is the sampling interval of the result

When the value 1 is specified for "Mode", the "Reduction"-value is interpreted as the reduction factor (RF). The function **RMS** then calculates the RMS-values among the last x values, where x is the window width. Additionally, the data rate is reduced by the factor "Reduction".

Reduction = 1: After each input value, a result value is calculated.

Reduction = 2: After every other input value, a result value is calculated.

Reduction = n: After every n-th input value, a result value is calculated.

When the value 2 is specified for "Mode", the "Reduction"-value is interpreted as the target sampling interval of the result channel. The target sampling interval must be the result of dividing the input channel's sampling interval by an integer.

Example: For an input channel sampling interval of 100 ms (0.1), the target sampling interval could be 500 ms (0.5), for example. This would represent a reduction factor (RF) of 5.

The 4th parameter "Mode" is optional. In the absence of any specification, the default value 1 is applied.

Special cases: If the window width is greater than 1*RF, the required count of values is not yet available at the start of the measurement. In this case, processing is applied only the input values available at that point in time.

In order to obtain useful results, the function requires a time base. When a channel is set as the 1st parameter, it is provided as the channel's sampling interval. When a single value (e.g. a process vector variable) is set as the 1st parameter, one can obtain a time reference by using timers (or on the device, in synchronous imc Online FAMOS tasks). When using single values as the input value (1st parameter), then for "Mode" = 1 the only possible setting for the reduction factor is 1. "Mode" = 2 is not supported with single values.



Example 1

```
Res = RMS( Signal_01, 10, 2 )
```

Returns the RMS value of the last 10 input values for every second input value.



Example 2

with optional parameter

```
Res = RMS( Signal_01, 10, 0.002, 2 )
```

On a channel sampled at 1 kHz, for every second input value the mean value among the last 10 input values is returned. The result has a sampling rate of 500 Hz.

Rosette1

Rosette calculation: Analysis of strain measurements with rosettes, $Eps1 \geq Eps2$.

Rosette1(Eps1, Eps2, Sig1, Sig2, SigV, Phi, Grid_A, Grid_B, Grid_C, Type, Poisson, EModul, AngleR)

Eps1: Result Principal strain ϵ_1 [$\mu\text{m}/\text{m}$]

Eps2: Result Principal strain ϵ_2 [$\mu\text{m}/\text{m}$]

Sig1: Result Principal stress σ_1 [N/mm^2]

Sig2: Result Principal stress σ_2 [N/mm^2]

SigV: Result reference stress per Mises, σ_v [N/mm^2]

Phi: Result Grid A angle $\phi_{P,Q}$ [$^\circ$]

Grid_A, _B, _C: Channel for Grid A, B, C ($\epsilon_a, \epsilon_b, \epsilon_c$) with measured strains in $\mu\text{m}/\text{m}$

Type: Rosette type

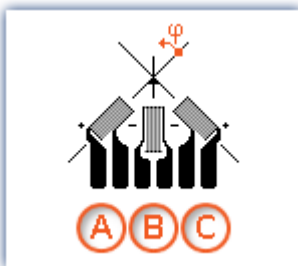
1: Rectangular rosette ($0^\circ/45^\circ/90^\circ$)

2: Delta rosette ($0^\circ/60^\circ/120^\circ$)

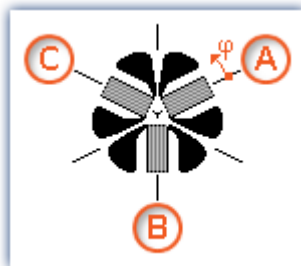
Poisson: Poisson's ratio ν

EModul: Modulus of elasticity in GPa ν

AngleR: Angle rectification below, in $\mu\text{m}/\text{m}$



Rosette type
 $0^\circ/45^\circ/90^\circ$



Rosette type
 $0^\circ/60^\circ/120^\circ$

Angle rectification below: For determining the Grid A angle, the arctan of a fraction is found. If both the numerator and denominator of this fraction are very small, small changes of the values could cause large angle changes.

If the sum of the absolute amounts of the numerator and denominator are less than or equal to the specified angle rectification, the angle is internally assigned to the value 0° .

Numbering of the measurement grid: In order to obtain correct values from measurement with 3-element rosettes, the measurement grid must be numbered in a very particular way. The numbering of the measurement grid must proceed as in the illustrations in the Formula Assistant.

Angle measurements in counterclockwise direction are assigned positive sign here, while clockwise measurements get negative sign.

The input channels for Grids A, B and C must be stated in $\mu\text{m}/\text{m}$, the angle rectification must also be stated in $\mu\text{m}/\text{m}$.

The results Eps1 and Eps2 are then also stated in $\mu\text{m}/\text{m}$, the principal stress and the reference stress according to Mises in N/mm^2 and the Grid A angle in $^\circ$ (degrees).

For the principal strains the relationship $\text{Eps1} \geq \text{Eps2}$ applies, i.e. Principal strain 1 is the greater of the two principal strains.

If no virtual channel is needed, instead of the channel name, a 0 can be entered.

For parameterization of the Rosette1-function the formula-assistant is recommended.



Example

```
Rosette1( Eps1, Eps2, Sig1, Sig2, SigV, Phi, Grid_A, Grid_B, Grid_C, 1, 0.3, 210, 5.0 )
```

Formulas used	
Principle strain at 45°	$\varepsilon_{1,2} = \frac{\varepsilon_a + \varepsilon_c}{2} \pm \frac{1}{\sqrt{2}} \sqrt{(\varepsilon_a - \varepsilon_b)^2 + (\varepsilon_c - \varepsilon_b)^2}$
Principle strain at 60°	$\varepsilon_{1,2} = \frac{\varepsilon_a + \varepsilon_b + \varepsilon_c}{3} \pm \sqrt{\left(\frac{2\varepsilon_a - \varepsilon_b - \varepsilon_c}{3}\right)^2 + \frac{1}{3}(\varepsilon_b - \varepsilon_c)^2}$
Principle stress 1	$\sigma_1 = \frac{E}{1 - \nu^2} (\varepsilon_1 + \nu \varepsilon_2)$
Principle stress 2	$\sigma_2 = \frac{E}{1 - \nu^2} (\varepsilon_2 + \nu \varepsilon_1)$
Reference stress as per Mises	$\sigma_v^2 = \sigma_1^2 + \sigma_2^2 - \sigma_1 \sigma_2$
Angle grid at 45°	$\phi_{p,q} = \frac{1}{2} \arctan\left(\frac{2\varepsilon_b - \varepsilon_a - \varepsilon_c}{\varepsilon_a - \varepsilon_c}\right)$
Angle grid at 60°	$\phi_{p,q} = \frac{1}{2} \arctan\left(\frac{\sqrt{3}(\varepsilon_b - \varepsilon_c)}{2\varepsilon_a - \varepsilon_b - \varepsilon_c}\right)$
Angle rectification	(Magnitude(Numerator)+Magnitude(Nominator)) > Angle rectification, the compute; else 0.

Rosette2

Rosette calculation: Analysis of strain measurements with rosettes, $|\text{Eps1}| \geq |\text{Eps2}|$.

Description: see [Rosette1](#) ³⁹⁶

Difference: For the principal strains the relationship $|\text{Eps1}| \geq |\text{Eps2}|$ applies, i.e. the **absolute value** of Principal strain 1 is the greater of the two principal strains.

Round

Rounds the argument to the nearest integer

a = Round(b)

This function returns the **nearest integer** for a real number input.



Example

The function returns

Input channel value	Result value
1,2	1
1,5	2
1,8	2
-1,2	-1
-1,5	-1
-1,8	-2

RSFlipflop

RS flip-flop function

a = RSFlipFlop(R, S)

a: Result

R: Reset

S: Set

Returns a 1 for the state H and a 0 for the state L. Beginning with the state L, the state H is entered if S is unequal to zero and R equals zero. If S equals zero and R is not equal to zero, the state L is entered. With the other two combinations of input values, the state remains unchanged.

In order to obtain useful results, the function requires a time base. When a channel is set as the 1st and the 2nd parameter, it is provided as the channel's sampling interval. When a single value (e.g. a process vector variable) is set as the 1st and the 2nd parameter, one can obtain a time reference by using timers (or on the device, in synchronous imc Online FAMOS tasks).



Example

An LED lights up when the signal exits the range from 0 to 8, and goes off again once the signal is in the range from 0.5 to 6.

```
_R= LogAnd( Less( Channel_001, 6 ), Greater( Channel_001, 0.5 ) )
_S= LogOr( Greater( Channel_001, 8 ), Less( Channel_001, 0 ) )
LED_01 = RSFlipFlop( _R, _S )
```

RunAutoBalance

Function for calling the automatic balancing for the selected channels.

RunAutoBalance()

The function does not require any parameters.

The channels to be balanced are selected before calling the function.

Activate the property "Balance at device startup" (Setup page: "Channel balance" or "Analog channels") for the desired channels. If you wish, you can also display the parameter of the same name as an additional column by using the "[Column chooser](#)".

Parameter	Description		
Category: Channel	long name	short name	column ID
Balance at device startup	Balance at device startup	Balance at startup	eBalanceAtDeviceStart

This parameter has two purposes:

- Before a Diskstart/Autostart measurement starts, **balancing** of the selected channels is performed.
- The channels selected can be balanced using the **imc Online FAMOS function** [RunAutoBalance](#).

This balancing operation starts automatically before every Diskstart-measurement. The type of balancing performed is set on the following page: Setup page: "Channel balance" > "Balance". Available options: Taring or bridge balancing (depending on the hardware).

[RunAutoBalance](#) should not be called at each run of the program. One way to ensure this is by means of a virtual bit.



Example

```
; Executed during a running measurement
OnTriggerMeasure( BaseTrigger )
If Virt_Bit01 = 1
    RunAutoBalance()
    Virt_Bit01 = 0
End
End
```

RunAutoShuntCalibration

Function for calling the automatic shunt calibration of the selected channels

RunAutoShuntCalibration()

9.2.9.3.16 S

SamplesGate

Gate for values: The input channel's values are only transferred, if the current value of the control channel is equal to the reference value.

```
a = SamplesGate( b, c, d )
```

a: result	c: control channel
b: input channel	d: reference value

The function sometimes yields output values at irregular intervals. Therefore, the result is generally difficult to combine with other channels in calculations.

Sawtooth

Converts input channel to a saw tooth waveform.

```
a = SawTooth( b, y0, dy, P )
```

a: Result	dy: Increment
b: Input channel	P: Cycle count in samples
y0: Start value	

An input is used to set the sampling rate of the sawtooth.

SendMessage

A message (e.g. "Message_001") is filled and sent with the specified channels.

Prerequisite:
Only available in: imc Online FAMOS!
This function will appear only, if a send message has been defined at the CAN-assistant!

```
SendMessage_[Message_001]( Mode, Parameter, CAN_001, [CAN_002], [...] )
```

Mode: Send cyclically, conditionally or always? 0: Cyclic sending 1: Conditional sending 2: Send always (only for Control commands)	Parameter: Depending on Mode: Pulse rate, condition or reserved If mode = 0: Pulse rate in s If mode = 1: Condition If mode = 2: Reserved, 0	CAN_001: 1st channel to send [CAN_002], [...]: If 2 channels, for instance, are defined in the CAN-Assistant

Operation Mode 2 can only be used with control commands.

Configuring a message to transmit is the condition for applying this function. In the CAN-Assistant, a formal framework of messages and channels is constructed. The function SendMessage is used to fill this framework and send it. The channels and single values in imc Online FAMOS are sent in the format set in the CAN-Assistant. The SendMessage function can be used, for instance, to assign a virtual channel created in imc Online FAMOS to the channel defined in the CAN-Assistant and send it in the data format defined.

The SendMessage function is automatically set up and listed in the Functions directory tree in response to the CAN-Assistant data. The actual function name SendMessage is extended with a suffix (e.g. `SendMessage_Message_001`).

The first two parameters (Mode and Parameter) are contained in each SendMessage-function. The other channel parameters are dependent on the configuration in the CAN-Assistant: The channels defined in the CAN-Assistant for this message are themselves the additional parameters of the function. If 2 channels, for instance, are defined in the CAN-Assistant, the SendMessage function has 4 parameters.

In cyclical message-sending, one message containing the current channel data is sent periodically at each sending-pulse interval. With conditional sending, exactly one message is sent if the specified condition is fulfilled (the condition is fulfilled by a transition from 0 to non-zero). Conditions must be fulfilled by channels; single values don't count. Other permissible parameters (the data to be sent) are channels and single values.



Example

The following parameter description of the function SendMessage is based on a message with 2 channels.

```
_Co = Greater( Test1, 500 )
SendMessage_Message_001( 1, _Co, Channel_001, Channel_002 )
or
SendMessage_Message _001( 0, 0.1, Channel_001, Channel_002 )
```

Sin

Sinus: Sine of the input channel

a = Sin(**b**)

a: Result

b: Input channel; in radians

SingleValueChannel

A virtual channel is created from a single value.

Prerequisite:

Only usable with control commands!

VirtualChannel = SingleValueChannel (Trigger, DataRate)

VirtualChannel: Result

DataRate: Average ... max. number of readings in a second; allowed: 1 ... 10000

Trigger: Trigger to be assigned to the channel

The function `SingleValueChannel` is used to declare a virtual channel. The declaration is made as an `OnInitAll` sequence. In the curve window, this virtual channel is also represented as a virtual channel, in imc Online FAMOS or imc Inline FAMOS it is regarded as single value. In `OnTriggerStart`, `OnTriggerMeasure` and `OnTriggerEnd` the single value can be redefined as desired. Every time a value is assigned to the single-value variable, it is written to the virtual channel as a sample. If a value is continuously assigned, the output memory for the virtual channel can overflow.

This virtual channel cannot be retrieved but may only appear on the left side of an assignment statement.



Example 1

Virtual channels can be created with integers or with real numbers.

```
OnInitAll
    VChanReell = SingleValueChannel( BaseTrigger, 1000 )
    int VChanInt = SingleValueChannel( BaseTrigger, 1000 )
End

OnTriggerMeasure( BaseTrigger )
    If Virt_Bit01 > 0
        VChanReell = 10.5
        Virt_Bit01 = 0
    End
    If Virt_Bit02 > 0
        VChanInt = 10
        Virt_Bit02 = 0
    End
End
```



Example 2

```
OnInitAll
    MinMax = SingleValueChannel( BaseTrigger, 1000 )
End

OnTriggerMeasure( BaseTrigger )
    If Channel_001 > 100
        MinMax = 100
    End
    If Channel_001 < 5
        MinMax = 5
    End
End
```

The virtual channel created is filled according to assignments of single values. Whenever an assignment is carried out, this single value is added to the virtual channel as a sample value. Thus it can happen that the virtual channel contains no samples. On the other hand, a large number of samples can accumulate in a short time. The data rate should be dimensioned correspondingly.

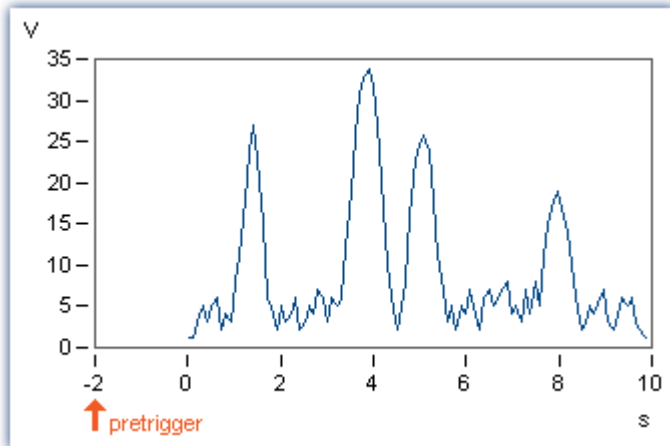
The unit and the clock rate (Delta-X) of this virtual channel can be set in the Properties dialog. The X-offset is fixed at 0, pretrigger time is not allowed for.

By default, the clock rate is set to 1s and no units are displayed in the curve window.

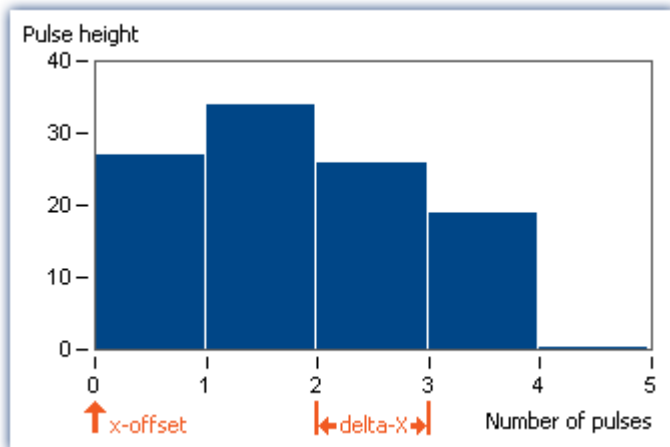


Example

As an example, voltage data (with a 2 s pretrigger) are displayed below in a curve window.



The voltage channel's big impulses are extracted using the function [SingleValueChannel](#) (here, there is one value representing each impulse, for a total of 4). The virtual channel "Pulse" is the result.



The x-offset is 0, delta-X in this example is 1.

SkipFirstValues

Skip signal values: During the filter's transients phase at the measurement's start, the initial signal values are skipped.

Prerequisite:

Only available in: imc Inline FAMOS!

a = SkipFirstValues(Signal, Count)

a: Result

Count: Number of values to skip

Signal: Input channel

This function is used to suppress the undesired signal fluctuations due to filtering, which occur at the measurement's start. For this purpose, as many values are skipped as specified in the parameter Count.

The results of the function can only be combined with channels having the same number of values to be skipped.

In contrast to the functions [ReplaceFirstValues0](#) and [ReplaceFirstValuesN](#), the values during the transients' phase are skipped instead of replaced.



Example

```
Signal_Filtered = FiltHP( Signal_01, 1, 0, 4, 100 )
Signal_Corrected = SkipFirstValues( Signal_Filtered, 1000 )
```

A high-pass filter is applied to the sine signal Signal_01 (the sampling rate of Signal_01 is 1 kHz).

When the measurement is started, a signal spike occurs (whose height depends on the phasing).

This signal spike would not occur once the filter's transients had subsided.

The transients are suppressed for the duration of one second.

For this purpose, the first 1000 values of the filtered signal are skipped.

SlopeClip

The maximum slope dy/dx between two adjacent values is clipped

a = SlopeClip(b, Max)

a: Result

Max: Maximum slope

b: Input channel

Smo3

Three-point smoothing: The input channel is smoothed by the weighted average of the three last samples

a = Smo3(b)

a: Result

b: Input channel

Smo5

Five-point smoothing: The input channel is smoothed by the weighted average of the five last samples

a = Smo5(**b**)

a: Result

b: Input channel

SoundPressureLevel

Sound pressure level: The function finds the course of the sound pressure level over time.

a = SoundPressureLevel(Signal, FrequencyRating, TimeConstant, ReductionFactor)

a: Result - The signal's sound pressure level

Signal: signal to be evaluated

FrequencyRating: signal's frequency rating

0: no rating

1: A-rating

2: B-rating

3: C-rating

TimeConstant: Time constant for averaging

In imc Online FAMOS:

≥ 0.0: Time constant is defined by the user in s

e.g. 0.125 for FAST-weighting, 1.0 for SLOW-weighting

In imc Inline FAMOS:

-1: Fast (0.125s)

-2: Slow (1s)

-3: Pulse

-4: Peak

-5: RMS in interval

-6: RMS from start

≥ 0.0: Time constant is defined by the user in s

ReductionFactor: Factor for resampling; ≥ 1

For a ReductionFactor = 1, no resampling

The function requires a sound signal expressed in the unit Pa (Pascal). The signal is first frequency rated, e.g. with an A-rating according to DIN IEC 651. Next, time-weighting (moving RMS with exponential averaging) and resampling are performed.

The result is referenced to $20 \cdot 10^{-6}$ Pa and expressed in dB.



Example

SPL = SoundPressureLevel (Signal, 1, 0.125, 1000)

The signal is subjected to A-rating.

The frequency-weighted signal is then time-weighted with a time constant of 0.125 s (FAST) and resampled with a factor of 1000.

The signal originally has a sampling frequency of 20 kHz; the A-rated result has a sampling frequency of only 20 Hz.

The result SPL is the sound pressure level over time and expressed in dB.

SpecThirds

1/3 octave spectrum in reference to time: From the time-histories of a vibration, the 1/3 octave spectrum referenced to time is determined.

Prerequisite:

Only available in: imc Inline FAMOS!

a = SpecThirds(Vibration, f1, f2, Frequency weighting, Time rating, Output interval)**a:** Result - 1/3 octave spectrum referenced to time**Vibration:** Time-history of vibration signal, the time scaled in seconds**f1:** Center frequency of lowest 1/3 octave in Hz
10, 12.5, 16, 20, 25, 31.5, 40, 50, 63, 80, 100, ...**f2:** Center frequency of highest 1/3 octave in Hz
1000, 1250, 1600, 2000, 2500, 3150, 4000, 5000, 6300, 8000, 10000, 12500, 16000, ...**Frequency weighting:** Frequency weighting for the result
0: linear
1: A-weighting
2: B-weighting
3: C-weighting
4: D-weighting**Time rating:** Averaging time, time rating of filtered data

-1: Fast (0.125s)

-2: Slow (1s)

-3: Pulse

-4: Peak

-5: RMS in interval

-6: RMS from start

> 0.0: Time constant is defined by the user, given in seconds

Output interval: 1/3 octave spectra are calculated using this time increment (delta-time, dt). Integer multiple of the vibration's sampling time. Specified in seconds.**Remarks:**

The two frequency limits f1 and f2 are to be given as the 1/3-octave center frequencies, e.g. f1 = 8 Hz and f2 = 12500 Hz. f1 < f2. The upper 1/3-3-octave with its frequency band must lie entirely within half of the sampling frequency.

The individual 1/3-octave values are stated as root mean square (RMS) values.

While the transients in the individual 1/3-octave (band-pass) filters are subsiding at the start of the measurement, the input signal's values are ignored. The transient time for the 1 kHz 1/3-octave is assumed to be 20 ms. This time interval is inversely proportional to the 1/3-octave frequency. For very low 1/3-octaves, this time interval becomes considerable. A correspondingly long measurement must then be expected.

During the settling phase the RMS is assumed to be 0.0. The result is a segmented data set. Each segment contains one 1/3-octave spectrum. The x-coordinate of the result counts the 1/3-octave bands (just like the Famos-function OctA). For good representation in the curve window, 1/3-octave labeling should be selected.

The z-coordinate of the result represents time. The first spectrum is taken where time is equal to the output interval.

The 1/3-octave filter and analyses are in accordance with DIN IEC 651 (sound level measurement), DIN 45652 (1/3-octave filter for electro-acoustic measurements) and DIN EN 61260 or IEC 1260 (band-pass filter for octaves and fractions of octaves, filter Class 0).

Time rating "Pulse": For increasing amplitudes the time constant is 35 ms, for decreasing amplitudes 1.5 s. Thus impulse-shaped signals are captured quickly, the response decays slowly.

Time rating "Peak": Extreme response for very short impulses; ensuring capture of the peak value. Time constant is zero during increasing amplitude (can be performed exactly by computer, by analog operation only in approximation); during decreasing amplitude 3 s.

 Example

The 1/3-octave spectrum is to be calculated from the time-history of the vibration every 0.1 s.

The signal is sampled every 0.025 ms.

```
fEval = 1 ; 0 (linear) 1 (A-weighting)
f1 = 10
f2 = 12500
tEval = -1 ; -1 (Fast)
tInterval = 0.1 ; [s]
Thirds = SpecThirds( vib, f1, f2, fEval, tEval, tInterval )
```

Sqrt

Square root of argument

a = Sqrt(**b**)

a : Result	b : Input channel

StDev

Standard deviation of a window, with reduction.

a = StDev(**b**, Windowwidth, Reduction[, Mode])

a : Result	Windowwidth : Window width in values
b : Input channel	In imc Online FAMOS: 1 * RF, 2 * RF, ... ,10 * RF In imc Inline FAMOS: 1 * RF, 2 * RF, ... ,1000 * RF
	Reduction : Reduction depending on the value of "Mode"
	Mode : [Optional] Interpretation of the parameter "Reduction" 1: The parameter "Reduction" is the reduction factor expressed in value count 2: The parameter "Reduction" is the sampling interval of the result

When the value 1 is specified for "Mode", the "Reduction"-value is interpreted as the reduction factor (RF). The function StDev then calculates the standard deviation among the last x values, where x is the window width. Additionally, the data rate is reduced by the factor "Reduction".

Reduction = 1: After each input value, a result value is calculated.

Reduction = 2: After every other input value, a result value is calculated.

Reduction = n: After every n-th input value, a result value is calculated.

When the value 2 is specified for "Mode", the "Reduction"-value is interpreted as the target sampling interval of the result channel. The target sampling interval must be the result of dividing the input channel's sampling interval by an integer.

Example:For an input channel sampling interval of 100 ms (0.1), the target sampling interval could be 500 ms (0.5), for example. This would represent a reduction factor (RF) of 5.

The 4th parameter "Mode" is optional. In the absence of any specification, the default value 1 is applied.

Special cases: If the window width is greater than $1 \cdot RF$, the required count of values is not yet available at the start of the measurement. In this case, processing is applied only the input values available at that point in time.

In order to obtain useful results, the function requires a time base. When a channel is set as the 1st parameter, it is provided as the channel's sampling interval. When a single value (e.g. a process vector variable) is set as the 1st parameter, one can obtain a time reference by using timers (or on the device, in synchronous imc Online FAMOS tasks). When using single values as the input value (1st parameter), then for "Mode" = 1 the only possible setting for the reduction factor is 1. "Mode" = 2 is not supported with single values.



Example 1

```
Res = StDev( Signal_01, 10, 4 )
```

Returns the standard deviation of the last 10 input values for every fourth input value.



Example 2

with optional parameter

```
Res = StDev( Signal_01, 10, 0.004, 2 )
```

On a channel sampled at 1 kHz, the mean of the last 10 input values is determined for every fourth input value. The result has a sampling rate of 250 Hz.

STri

Schmitt trigger with upper and lower threshold

a = STri(b, S_upper, S_lower)

a: Result

S_upper: Upper threshold

b: Input channel

S_lower: Lower threshold

If the previous result was 0, the new result is:

0, if the current value is less than S_upper
otherwise 1

else (if the previous result was 1) the new result is:

1, if the current value is greater than S_lower
otherwise 0

For computing the first value, the previous result is assumed to be 0.



Example

Computes a clean TTL signal

```
res = STri( Channel_001, 4, 1 ) * 5
```

Sum

Summation with reduction

a = **Sum**(**b**, **Reduction**[, **Mode**])

a: Result

b: Input channel or input value

Reduction: Reduction depending on the value of "Mode"

Mode: [Optional] Interpretation of the parameter "Reduction"

1: The parameter "Reduction" is the reduction factor expressed in value count

2: The parameter "Reduction" is the sampling interval of the result

When the value 1 is specified for "Mode", the "Reduction"-value is interpreted as the reduction factor (RF). The function calculates the running sum of the input values. Additionally, the data rate is reduced by the factor "Reduction".

Reduction = 1: After each input value, a result value is calculated.

Reduction = 2: After every other input value, a result value is calculated.

Reduction = n: After every n-th input value, a result value is calculated.

When the value 2 is specified for "Mode", the "Reduction"-value is interpreted as the target sampling interval of the result channel. The target sampling interval must be the result of dividing the input channel's sampling interval by an integer.

Example: For an input channel sampling interval of 100 ms (0.1), the target sampling interval could be 500 ms (0.5), for example. This would represent a reduction factor (RF) of 5.

The 3rd parameter "Mode" is optional. In the absence of any specification, the default value 1 is applied.

In order to obtain useful results, the function requires a time base. When a channel is set as the 1st parameter, it is provided as the channel's sampling interval. When a single value (e.g. a process vector variable) is set as the 1st parameter, one can obtain a time reference by using timers (or on the device, in synchronous imc Online FAMOS tasks). When using single values as the input value (1st parameter), then for "Mode" = 1 the only possible setting for the reduction factor is 1. "Mode" = 2 is not supported with single values.



Example 1

```
Res1 = Sum( Signal_01, 10 )
```

The sum of the input values is calculated and every tenth value outputted.

```
Res2 = Sum( IncEnc_01, 1 )
```

Converts a channel's distance increments into the complete distance.

**Example 2****with optional parameter**

```
Res1 = Sum( Signal_01, 0.01, 2 )
```

The sum of the values of a channel sampled at 1 kHz is calculated. The result's sampling rate is 100 Hz.

**Note****Adding small numbers to proportionally much larger numbers**

Observe the limitations of the resolution of numbers in 32-bit Float format.

Sum2

Sum with reset to 0: Sum of all values captured on a channel since the last reset to 0, or since the beginning of measurement, with reduction.

a = Sum2(b, Reduction, Reset[, Mode])

a: Result

b: Input channel or input value

Reduction: Reduction depending on the value of "Mode"

Reset: If unequal to 0, reset sum to 0

Mode: [Optional] Interpretation of the parameter "Reduction"

1: The parameter "Reduction" is the reduction factor expressed in value count

2: The parameter "Reduction" is the sampling interval of the result

In contrast to the function `Sum`, the sum in the `Sum2`-function can be reset during measurement to 0. Otherwise, the functions `Sum` and `Sum2` operate the same way.

When the value 1 is specified for "Mode", the "Reduction"-value is interpreted as the reduction factor (RF). The function calculates the running sum of the input values. Additionally, the data rate is reduced by the factor "Reduction".

Reduction = 1: After each input value, a result value is calculated.

Reduction = 2: After every other input value, a result value is calculated.

Reduction = n: After every n-th input value, a result value is calculated.

When the value 2 is specified for "Mode", the "Reduction"-value is interpreted as the target sampling interval of the result channel. The target sampling interval must be the result of dividing the input channel's sampling interval by an integer.

Example: For an input channel sampling interval of 100 ms (0.1), the target sampling interval could be 500 ms (0.5), for example. This would represent a reduction factor (RF) of 5.

The parameter "Reset" may be a single value (e.g. virtual bit) or a channel.

If the parameter 'Reset' is a single value, and resetting is executed, the input channel samples accumulating during this processing step are ignored in the summation. Multiple samples in one processing step can accumulate especially in the case of high sampling rates or where there are extensive calculations (e.g. use of FFTs). The summation result is afterwards zero.

If the parameter 'Reset' is a channel and resetting is executed, then any input channel samples accumulating after the reset are captured correctly. Even if they arrive during the same processing step.

The 4th parameter "Mode" is optional. In the absence of any specification, the default value 1 is applied.

In order to obtain useful results, the function requires a time base. When a channel is set as the 1st parameter, it is provided as the channel's sampling interval. When a single value (e.g. a process vector variable) is set as the 1st parameter, one can obtain a time reference by using timers (or on the device, in synchronous imc Online FAMOS tasks). When using single values as the input value (1st parameter), then for "Mode" = 1 the only possible setting for the reduction factor is 1. "Mode" = 2 is not supported with single values.



Example 1

```
S1 = Sum2( Signal_01, 10, VrtBit_01 )
```

The sum of the input values is calculated and every tenth value is returned. Reset of the sum to 0, if VrtBit_01 changes from 0 to 1.

```
S1 = Sum2( Signal_01, 1, Reset_01 )
```

The sum of entered values is calculated and each value is outputted. Resets the sum to 0, whenever a transition from 0 to 1 occurs in the channel Reset_01.



Example 2

with optional parameter

```
S1 = Sum2( Signal_01, 0.01, VrtBit_01, 2 )
```

The sum of the values captured on a channel sampled at 1 kHz is calculated. The result's sampling rate is 100 Hz. Reset of the sum to 0, if VrtBit_01 changes from 0 to 1.

SyncOverload

Synchronous task is overloaded: Sets the specified value if the synchronous task is overloaded.

Prerequisite:

Only available in: imc Online FAMOS!

Only usable with control commands!

The function `SyncOverload()` may only be called in the synchronous task.

Overloaded = SyncOverload(Value)

Overloaded: Result

Value: Fixed value to which the result is set

If the synchronous task is overloaded, the result variable is set to the value specified in the function `SyncOverload`. The function can be called at any point in the lowest level of a synchronous task. No combinations with other functions are permitted. The parameter Value must take a fixed value (e.g. 1, 2 or 3).

As results, Display variables, virtual bits, Ethernet bits, digital outputs, pv-variables and vector elements generated with the function `VectorChannel` are all permitted. If the result is a vector element, then if the synchronous task is constantly overloaded, no channels and thus no vectors can be outputted any longer (in the curve window).

If multiple synchronous tasks are going and data overload is detected in one of them, the overload will be indicated for this synchronous task or possibly for a faster synchronous task.



Example

```
OnSyncTask( 0.0001 )
    Virt_Bit01 = SyncOverload( 1 )
    pv.SyncOverloaded = SyncOverload( 10 )
    ...
End

OnSyncTask( 0.001 )
    ...
    DisplayVar_32 = SyncOverload( 20 )
End
```

9.2.9.3.17 T

Tan

Tangent of input channel

a = **Tan**(**b**)

a: Result

b: Input channel; in Radians

TextAdd

Join texts. Two texts are conjoined. Text2 is appended to Text1.

Text = **TextAdd**(**Text1**, **Text2**)

Text: Result

Text1: Text, to which to append Text2

Text2: Text, to which Text1 is appended

Instead of the function `TextAdd`, the operator '+' can be used, see example.

Text-functions require large amounts of processing time and therefore should only be used for isolated outputs of text, rather than continually.



Example

```
OnInitAll
    Text1 = "Number"
    Text3 = ""
    IsText = 0
    Number = 1000
End

OnTriggerMeasure ( BaseTrigger )
    ...
    If IsText
        Text3 = Text1 + TextFormatH( Number, 8 )
        RecordText( Text3 )
        ; The text "Number = 0x000003E8" is returned
        IsText = 0
    End
    RecordEvent( Channel_001, "Index = " + TextFormatI( 250 ) )
    ; If the event occurs, the text "Index = 250" is returned
End
```

TextFormatE

Converts a real number to text. A real number is converted to text in floating point notation.

Text = **TextFormatE**(**Number**, **DecPlace**)

Text: Result

DecPlace: Number of digits after the decimal

Number: Real number to be converted to text

The text contains as many significant digits as the parameter supplied to the function.

E.g. with four decimal places, the number 35 is represented as "3.5000E01" and the number 32767 as "3.2770E04".

Text-functions require large amounts of processing time and therefore should only be used for isolated outputs of text, rather than continually.



Example

```
OnInitAll
    Text1 = "Number"
    Text3 = ""
    IsText = 0
    Number = 1000
End

OnTriggerMeasure ( BaseTrigger )
    ...
    If IsText
        Text3 = Text1 + TextFormatE( Number, 6 )
        RecordText( Text3 )
        ; The text "Number = 1.000000E03" is returned
        IsText = 0
    End
    RecordEvent( Channel_001, "Index = " + TextFormatE( 250, 4 ) )
    ; If the event occurs, the text "Index = 2.5000E02" is returned
End
```

TextFormatF

Translates a real number to text. A real number is converted to text with fixed point notation.

Text = **TextFormatF**(**Number**, **DecPlace**)

Text: Result

DecPlace: Number of digits after the decimal

Number: Real number to be converted to text

The text contains as many decimal places as the corresponding parameter function contains.

E.g., with four decimal places, the number 35 is represented as "3.5000" and the number 32767 as "32767.0000".

Text-functions require large amounts of processing time and therefore should only be used for isolated outputs of text, rather than continually.



Example

```
OnInitAll
    Text1 = "Number"
    Text3 = ""
    IsText = 0
    Number = 1000
End

OnTriggerMeasure ( BaseTrigger )
    ...
    If IsText
        Text3 = Text1 + TextFormatF( Number, 6 )
        RecordText( Text3 )
        ; The text "Number = 1.000000" is returned
        IsText = 0
    End
    RecordEvent( Channel_001, "Index = " + TextFormatF( 250, 4 ) )
    ; If the event occurs, the text "Index = 250.0000" is returned
End
```

TextFormatH

Converts a number to text. A number is converted to text with hexadecimal notation.

Text = **TextFormatH**(**Number**, **Places**)

Text: Result

Places: Number of decimal places

Number: Number to be converted to text

The text contains as many digital places as as the parameter supplied to the function.

E.g. with four significant digits, the number 35 is represented as "0023" and the number 32767 as "7FFF".

Text-functions require large amounts of processing time and therefore should only be used for isolated outputs of text, rather than continually.



Example

```
OnInitAll
    Text1 = "Number"
    Text3 = ""
    IsText = 0
    Number = 1000
End

OnTriggerMeasure ( BaseTrigger )
    ...
    If IsText
        Text3 = Text1 + TextFormatH( Number, 8 )
        RecordText( Text3 )
        ; The text "Number = 0x000003E8" is returned
        IsText = 0
    End
    RecordEvent( Channel_001, "Index = " + TextFormatH( 250, 4 ) )
    ; If the event occurs, the text "Index = 0x00FA" is returned
End
```

TextFormatI

Converts a number to text in integer notation.

Text = **TextFormatI**(**Number**)

Text: Result

Number: Number to be converted to text

The text only contains as many digits as needed to represent the integer.

E.g. the number 35 appears as "35" and the number 32767 as "32767".

Text-functions require large amounts of processing time and therefore should only be used for isolated outputs of text, rather than continually.



Example

```
OnInitAll
    Text1 = "Number"
    Text3 = ""
    IsText = 0
    Number = 1000
End

OnTriggerMeasure ( BaseTrigger )
    ...
    If IsText
        Text3 = Text1 + TextFormatI( Number )
        RecordText( Text3 )
        ; The text "Number = 1000" is returned
        IsText = 0
    End
    RecordEvent( Channel_001, "Index = " + TextFormatI( 250 ) )
    ; If the event occurs, the text "Index = 250" is returned
End
```

Tolerance

Function that returns logical value of whether each input channel value is within the given tolerance.

a = **Tolerance**(**b**, **min**, **max**)

a: Result

min: Minimum

b: Input (channel or single values)

max: Maximum



Example

```
Res1 = Tolerance( Signal_01, pv.Min1, pv.Max1 )
Res2 = Tolerance( Signal_02, 2.5, 6.5 )
```

For example, Res2 takes the value 1 if Signal_02 is 4.1.

For example, Res2 takes the value 0, if Signal_02 is 2.1.

TransRec

Transitional Recording: Data reduction for analog data

Prerequisite:

Only available in: imc Online FAMOS!

a = TransRec(b, Precision)

a: result

Precision: Precision in units of the argument

b: input channel

To perform long-term measurements at high resolution, a special computational algorithm has been designed. The principle on which it operates is to only save such data (data points) which are necessary for reconstructing the original signal at a given level of precision.

This means that rapid signal movements are sampled at a high sampling rate. However slow-paced signal movements are sampled at as few points as will still allow the signal to be reconstructed.

Whether a measured value is saved depends on the signal and on the tolerance threshold set. Data are only saved if the measured signal can not be represented by a substitute curve within the tolerance demanded. The data-reduced channel never deviates from the input signal by more than the amplitude difference demanded.

Data reduction factors of up to 254 can be achieved by means of this procedure.



Note

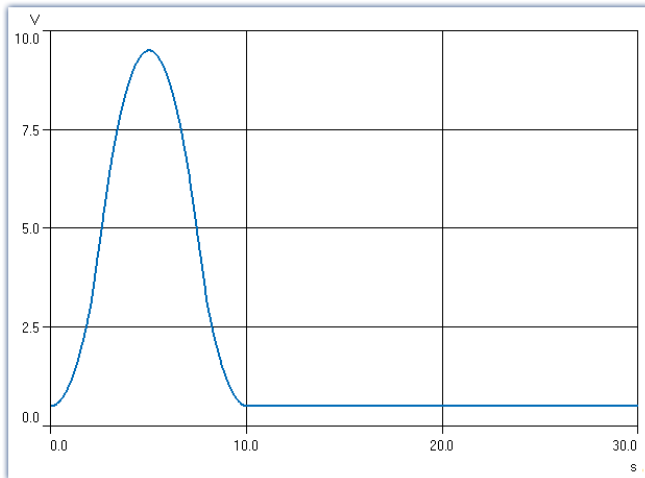
Transfer to PC is not possible

Results of the **TransRec** function and DI-port compressed channels are blocked from transfer to the PC.



Example 1

The following input signal, consisting over a period of 10s of a long period of constancy and then a peak at 9.5V, is to be measured.

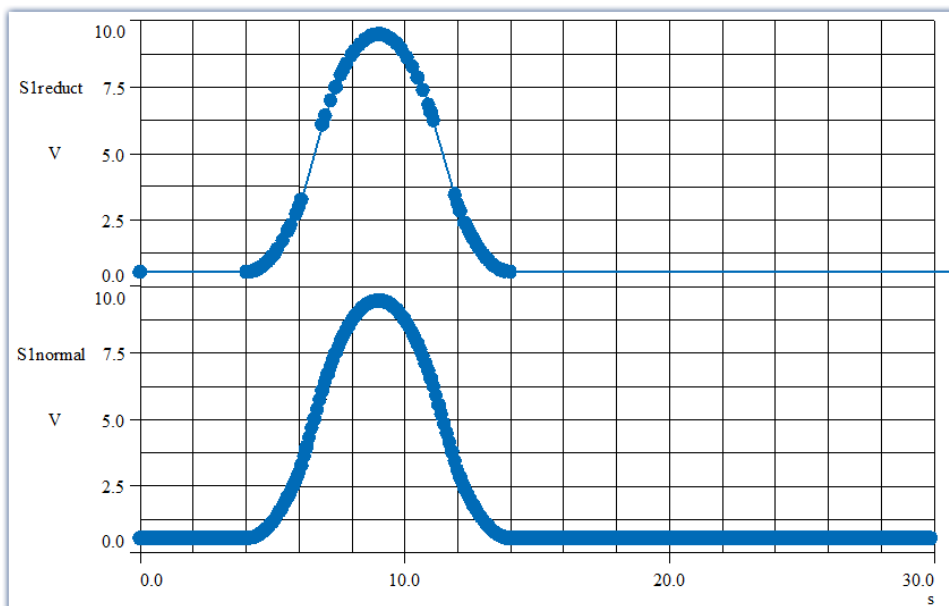


In this example, the signal is recorded once with and once without **TransRec** and saved.

As tolerance threshold set for the data reduction is 10mV.

S1reduct is measured with data reduction

S1normal is measured without data reduction



The curves are displayed on top of each other. The filled circles represent sample values, the lines between them are interpolations

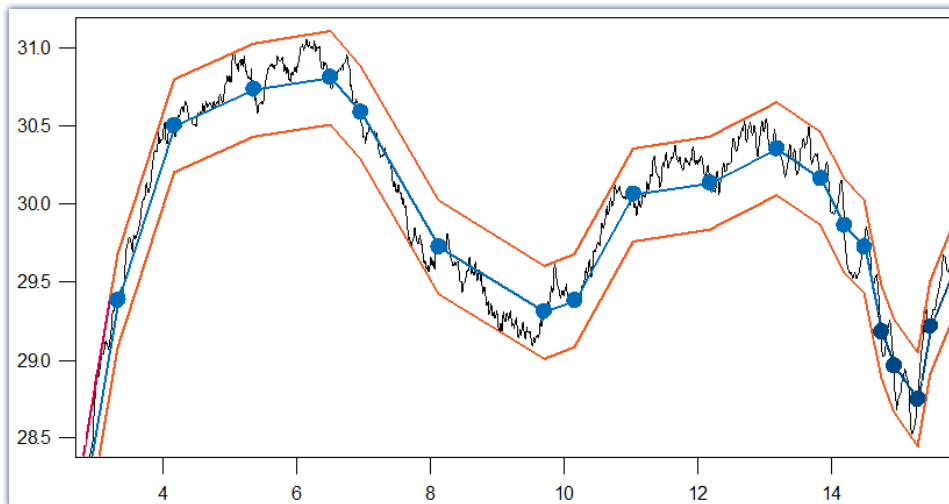
The data reduction only adds measured points wherever the curve features significant curvature. Straight-line signal plots of any slope are recognized and summarized as a straight line between two measured points.



Example 2

In the illustration below, the results from a temperature measurement are shown. The original data (zigzag line) was recorded with a tolerance of 0.3°C . The reduced data are shown using solid circles, joined by straight lines.

To demonstrate the effectiveness of the algorithm, a tolerance band of $\pm 3^{\circ}\text{C}$ has been parallel to the interpolated line. As you can see, the original signal never exceeds the tolerance bounds.



*The filled circles are derived by Transitional Recording.
The original signal never leaves the tolerance band.*

For what sort of signals is Transitional Recording used?

Signals which feature movement only rarely or at constant slope, but in which major changes need to be sampled at a high rate. If a certain amount of imprecision (e.g. 1% of the input range) is acceptable, then this procedure is especially effective. Since temperature measurement often meets these criteria, the use of this procedure is very useful for this case.

[TransRec](#) is particularly suited to long-term measurements which would probably result in data overload in the absence of data reduction.

Precision

If you have defined a scaling for the reduced channel, then the precision should be selected accordingly.



Example

Scaling factor: 0.03 A/V

Input range: 0.3 A

Precision: 10 mA

Setting to make: 0.01

The unit of your signal is "A", so the tolerance (precision) is also specified in "A". 10 mA takes the notation 0.01 A , but the unit "A" must not be stated in the function.

Post calculations

The calculation results of the function `TransRec` can not be subject to further calculation in imc Online FAMOS or imc Inline FAMOS. Therefore it is best to first have all other calculations with data from a virtual channel performed and have `TransRec` executed last.



Example

```
_Difference_tmp = Channel_001 - Channel_002  
Difference = TransRec (_Difference_tmp, 0.1)
```

In this example, the difference between two channels is first calculated.

The underline at the beginning indicates that the signal "_Difference_tmp" is not to be used as a virtual channel. Only after performing data reduction, we obtain the virtual channel with its correct name "Difference".

Maximum Reduction

The `TransRec` function produces resulting data sets with reduction factors between 1 and 254. Even with a constant signal, every 254th data point is recorded, thereby limiting the maximum reduction factor to 254.

Digital channels

The function `TransRec` can be applied to digital signals. However, this is not recommended.

Wherever possible, use the pre-processing for digital signals: **Reduction**. This option is available in the Setup dialog: **Sampling & Preprocessing**.

File format / Data type

The result of the `TransRec`-function takes the data type: **XY-waveform**. Like the other data types, it can be processed by other imc software packages (e.g. imc FAMOS).

Limitations

Buffer memory size in the device is limited to 128 kBytes. This means that the maximum buffer duration may be somewhat limited, depending on the sampling rate. However, if the technique is used properly and a significant reduction factor is achieved, that will not matter any more.



Note

Storage of measurement data always automatically generates sets of 64 samples. Therefore, when a measurement (or file) is closed after a fixed space of time, it usually happens that data points from before or after the fixed time interval are also saved in the file. If there is no data value which would normally be saved at the moment that the file is to be closed, the file is closed anyway. This means that the last data point in a file may be from a point in time significantly preceding the stopping time specified.

TSAGate

Gate for values in the case of time-stamped data: The input channel's values are only passed on as long as the control channel's value is non-zero.

Prerequisite:

Only available in: imc Inline FAMOS!

a = **TSAGate**(**b**, **c**, **p**)

a: Result

c: Control channel

b: TSA input channel

p: Pretrigger in bytes

The function sometimes yields output values at irregular intervals. Therefore, the result is generally difficult to combine with other channels in calculations.

The parameter p (pretrigger in Bytes) may be 0. Otherwise, p must be set in accordance with the size of the TSA-sample.

To do this, the size of a TSA-sample must be known. A TSA-sample fundamentally consists of a header and the data bytes. The header is always 8 Bytes long. The size of the data bytes may be a few Bytes (e.g. in the CAN-Bus up to 8 Bytes), or also several 1000 Bytes. Additionally, an overhead (e.g. for synchronization markers, etc.) can be taken into account: For smaller TSA-samples, the overhead should be only a few Bytes, while for larger ones, 10 % of a TSA-sample's data bytes, for example.

How the size of the parameters p is determined is illustrated by the example of a CAN message containing 8 data bytes:

The CAN message is sent every 100 ms. A pretrigger time of 2 s is to be taken into account. With 8 header bytes, 8 data bytes and an overhead of 4 in this example, the following results are obtained:

$$p = (8 \text{ Bytes} + 8 \text{ Bytes} + 4 \text{ Bytes}) * 2 \text{ s} / 100 \text{ ms} = 400 \text{ Bytes}.$$



Example

```
g = TSAGate ( TSA_CAN_001, c, 400 )
```

The channel g is a collection of all CAN messages appearing when the control channel c is non-zero. This takes a pretrigger of 2 s into account. For the calculation of the pretrigger in Bytes, see above.

9.2.9.3.18 U

Unequal

Checks whether the first argument is not equal to the second argument

a = Unequal(b, c)

a: Result

b: 1. argument

c: 2. argument

a = 1 if b is not equal to c, otherwise 0.

Upper

Returns the greater value of the arguments

a = Upper(b, c)

a: Result

b: 1. argument

c: 2. argument



Example

```
sound = Upper( dB( Channel_001 ), -100 )
```

The input channel is converted into decibel and restricted to a lower border of -100 dB.

9.2.9.3.19 V

General information about the VectorChannel functions

Prerequisite:

Only usable with control commands!

A virtual channel is declared with the function `VectorChannel`. All elements are initialized as zero. Declaration is performed in the OnInitAll-Block. In the curve window, this virtual channel is also represented as a virtual channel, but in imc Online FAMOS or imc Inline FAMOS it's treated as a static vector. The vector elements can be assigned and queried like single values.

Whenever the function `VectorChannelSet` is called, the complete vector is transferred to the virtual channel. The function `VectorChannelSet` can be called in `OnTriggerStart`, `OnTriggerMeasure` and `OnTriggerEnd` as well as in blocks which don't depend on triggers, e.g. in `OnSyncTask`. If the function `VectorChannelSet` is called too quickly and too often in succession, the memory allocation for the virtual channel can overflow.

Virtual channels can be generated from vectors with numbers and from vectors with real numbers (see example).

From version imc STUDIO 5.0 vector-handling has been enhanced with local vectors. Now, functions with 2 results vectors are supported, e.g. `FFTAmpitudePhase( AmlSpec_Local, PhasSpec_Local, ...)`, where `AmlSpec_Local` and `PhasSpec_Local` must be declared as local vectors in the `OnInitAll`-block.

**Example****Example, generating virtual channels from vectors**

```
OnInitAll
    int v1 = VectorChannel( BaseTrigger, 1000, 3 ) ; vector with integers
    ; v1[1] = 0, v1[2] = 0, v1[3] = 0 happens automatically
    v1[3] = 2

    v2 = VectorChannel( BaseTrigger, 1000, 3 ) ; vector with real numbers
    ; v2[1] = 0, v2[2] = 0, v2[3] = 0 happens automatically
    v2[3] = 2.5
End

OnSyncTask( 0.5 )
    v1[1] = v1[1] + 1
    v1[2] = VirtualBit_01
    v1[3] = 1
    VectorChannelSet( v1 ) ; the vector is transferred to the virtual channel

    v2[1] = v2[1] + 1
    v2[2] = VirtualBit_01
    v2[3] = 1.5
    VectorChannelSet( v2 ) ; the vector is transferred to the virtual channel
End
```

VectorChannel

Virtual channel from vector: A segmented virtual channel is generated from a vector.

Prerequisite:

Only usable with control commands!

VirtualChannel = **VectorChannel**(Trigger, DataRate, ElementCount)

VirtualChannel: Result

Trigger: Trigger to which the channel is to be assigned.

DataRate: Average... max. count of vectors in a second;
Allowed: 1 ... 10000

ElementCount: Number of elements in the vector;
Allowed: 1 ... 100000

A virtual channel is declared with the function **VectorChannel**. All elements are initialized as zero. The declaration is performed in the **OnInitAll**-Block. In the curve window, this virtual channel is also represented as a virtual channel, but in imc Online FAMOS it's treated as a static vector. The vector elements can be assigned and queried like single values.

Whenever the function **VectorChannelSet** is called, the complete vector is transferred to the virtual channel. The function **VectorChannelSet** can be called in **OnTriggerStart**, **OnTriggerMeasure** and **OnTriggerEnd** as well as in blocks which don't depend on triggers, e.g. in **OnSyncTask**. If the function **VectorChannelSet** is called too quickly and too often in succession, the memory allocation for the virtual channel can overflow.

Virtual channels can be generated from vectors with numbers and from vectors with real numbers (see example).



Example 1

```
; Generating a virtual channel from a vector with integers:
OnInitAll
    int v1 = VectorChannel( BaseTrigger, 1000, 3 )
    ; v2[1] = 0, v2[2] = 0, v2[3] = 0 performed automatically
    v1[3] = 2
End

OnSyncTask( 0.5 )
    v1[1] = v1[1] + 1
    v1[2] = Virt_Bit01
    v1[3] = 1
    VectorChannelSet( v1 )
End
```



Example 2

```
; Generating a virtual channel from a vector with real numbers:
OnInitAll
    v2 = VectorChannel( BaseTrigger, 1000, 3 )
    ; v2[1] = 0, v2[2] = 0, v2[3] = 0 performed automatically
    v2[3] = 2.5
End

OnSyncTask( 0.1 )
    v2[1] = v2[1] + 1
    v2[2] = Virt_Bit02
    v2[3] = 1.5
    VectorChannelSet( v2 )
End
```

VectorChannelSet

Writing a vector: A vector is written to a segmented virtual channel.

Prerequisite:

Only usable with control commands!

VectorChannelSet(Vector)

Vector: A segmented virtual channel generated with the function `VectorChannel`

VectorFromFile

Read vector from file

Vector = VectorFromFile("Filename")

Vector: Resulting vector	"Filename": Name of a file containing the vector elements

Load the file

In imc STUDIO, the file of the vector data is permanently integrated in the experiment. Import this file using the ribbon.

Ribbon	View
Setup-Configuration > Supplemental Files	Complete

 Notes

- The vector data is limited to 50000 values.
- Data formats (Float, Integer, Bool) can under certain circumstances be converted by imc Online FAMOS during loading. See also the notes on [Boolean variables in IF conditions.](#)

 Example

```
Vector = VectorFromFile( "Vector_01.DAT" )
```

VectorizeAndSkip

Create vector series: groups input samples into a series of vectors

Vector = VectorizeAndSkip(b, VLength, SkipLength)

Vector: Output vector series	VLength: Number of samples per vector; > 1
b: Input channel	SkipLength: Number of samples to be skipped; ≥ 0

The input samples are grouped into a series of vectors.
A fixed number of samples can be skipped between successive vectors.

VectorizeOverlapped

Creates an overlapping sequence of vectors from the input channel

Vector = **VectorizeOverlapped**(**b**, **VLength**, **Increment**)

Vector: Result vector sequence

VLength: Length of a vector; > 1

b: input channel

Increment: Distance to the next vector; $\geq 1, \leq VLength$

The input channel is converted to a vector sequence.

The vectors can overlap each other, i.e. values from the input channel are entered multiple times in the vector sequence.

After every 'Increment' (e.g. 100) input channel values the subsequent 'Vlength' (e.g. 1024) input channel values are converted to a vector.

At a sampling rate of 1kHz, a vector of the length 1024 is created from these samples every 100ms.



Example

The vector a returns a vector with 1024 values for every 100 input values.

This results in an overlap ratio of $(1024-100) / 1024 * 100\% = 90.23\%$.

```
a = VectorizeOverlapped( Channel_001, 1024, 100 )
b = FFT( a, 0, 1024 )
```

VectorStatic

Creates a static vector. A static vector whose elements can be changed during the measurement is created.

Prerequisite:

Only usable with control commands!

VectStat = **VectorStatic**(**Trigger**, **Element Count**)

VectStat: Result

Element count: Number of the vector's elements;
Allowed: 1 ... 100000

Trigger: Trigger to which the vector is to be assigned

A static vector is declared using the function **VectorStatic**. All elements are initialized at "0". The declaration is made in the **OnInitAll** block.

The vector elements are treated as single values. Whenever an assignment is made to a vector element, the vector element's current value is transferred to the result vector. Assignments to vector elements can be called in **OnTriggerStart**, **OnTriggerMeasure** and **OnTriggerEnd** as well as in the blocks which do not depend on triggers, such as **OnTimer** or **OnSyncTask**.

Static vectors can be generated with integers and with real numbers (see examples).



Example

```

OnInitAll
    int vs1 = VectorStatic( BaseTrigger, 3 ) ; vector with integers
    ; vs1[1] = 0, vs1[2] = 0, vs1[3] = 0 performed automatically

    vs2 = VectorStatic( BaseTrigger, 3 ) ; vector with real numbers
    ; vs2[1] = 0.0, vs2[2] = 0.0, vs2[3] = 0.0 performed automatically
End

OnSyncTask( 0.5 )
    If Virt_Bit01 > 0
        vs1[1] = 4
        vs1[2] = 5
        vs1[3] = 6
    Else
        vs1[1] = 1
        vs1[2] = 1
        vs1[3] = 2
    End

    vs2[1] = pv.Channel_001
    vs2[2] = pv.Channel_002
    vs2[3] = pv.Channel_003
End

```

Access to vectors via variables

It is possible to access the elements of the vector with the help of variable. Thus, different values can be read according to the variable's value.



Example

In the following example, the value is **written** to the vector by means of the "pointer"-variable:

"DisplayVar_01":

```

; writing to a vector at a variable position
Vektor[DisplayVar_01] = DisplayVar_02+10

```

In the following example, the value is **read** from the vector by means of the "pointer"-variable:

"DisplayVar_01":

```

; reading from the vector at a variable position
DisplayVar_03 = Vektor[DisplayVar_01]

```



Note

Notes on assignment

The first element is addressed with the value "1". The second with "2", etc.

Size of vector

The maximum number of values is limited to 100 000 values.

VibrationFilter

Vibration-weighting: A signal is filtered according to the frequency weighting set. Additionally, retroactive time weighting (moving RMS-value with exponential averaging) and resampling can be performed.

VibSignal = VibrationFilter(Signal, FrequencyWeighting, TimeConstant, ReductionFactor)

VibSignal: Filtered signal	TimeConstant: Time constant for the exponential RMS-calculation in seconds, ≥ 0 . e.g. 0.125 with FAST-weighting 1.0 with SLOW-weighting	Reduction factor: Factor for resampling, ≥ 1 . With ReductionFactor = 1, no resampling. This enables the data volume of the result to be conveniently reduced when the time constant is large.
Signal: Input signal to be weighted; the time scaled in seconds.		
FrequencyWeighting: Frequency weighting for the signal		

For a time constant = 0, no subsequent time weighting is performed. The reduction factor must in this case be exactly 1.

For numerical reasons, there are constraints on the input signals' sampling intervals which depend on the frequency weighting (see above).

Frequency weighting for the signal

per ISO 2631-1, 2nd edition, 1997:	
10	Wk, z direction and for vertical recumbent direction, except head. Recommended signal's sample time: $\geq 0.5\text{ms}$ and $\leq 5\text{ms}$ or $\geq 200\text{Hz}$ and $\leq 2\text{kHz}$
11	Wd, x and y directions and for horizontal recumbent direction. Recommended signal's sample time: $\geq 0.5\text{ms}$ and $\leq 2\text{ms}$ or $\geq 500\text{Hz}$ and $\leq 2\text{kHz}$
12	Wf, motion sickness. Recommended signal's sample time: $\geq 5\text{ms}$ and $\leq 500\text{ms}$ or $\geq 2\text{Hz}$ and $\leq 200\text{Hz}$
13	Wc, seat-back measurement. Recommended signal's sample time: $\geq 0.5\text{ms}$ and $\leq 10\text{ms}$ or $\geq 100\text{Hz}$ and $\leq 2\text{kHz}$
14	We, measurement of rotational vibration. Recommended signal's sample time: $\geq 0.5\text{ms}$ and $\leq 2\text{ms}$ or $\geq 500\text{Hz}$ and $\leq 2\text{kHz}$
15	Wj, vibration under the head of a recumbent person. Recommended signal's sample time: $\geq 0.5\text{ms}$ and $\leq 2\text{ms}$ or $\geq 500\text{Hz}$ and $\leq 2\text{kHz}$

per DIN 45671 Part 1, Sept. 1990:

- 16 Hx, Whole-body vibrations, standing, seated position, measurement direction: x, y, Recumbent position, measurement direction: y, z.

Recommended signal's sample time: $\geq 0.2\text{ms}$ and $\leq 50\text{ms}$ or $\geq 20\text{Hz}$ and $\leq 5\text{kHz}$

- 17 Hz, Whole-body vibrations, standing, seated position, measurement direction: z

Recommended signal's sample time: $\geq 0.5\text{ms}$ and $\leq 10\text{ms}$ or $\geq 100\text{Hz}$ and $\leq 2\text{kHz}$

- 18 Hxl, Whole-body vibrations, recumbent position, measurement direction: x

Recommended signal's sample time: $\geq 0.5\text{ms}$ and $\leq 2\text{ms}$ or $\geq 500\text{Hz}$ and $\leq 2\text{kHz}$

- 19 Hb, Whole-body vibrations, body position not specified

Recommended signal's sample time: $\geq 0.5\text{ms}$ and $\leq 10\text{ms}$ or $\geq 100\text{Hz}$ and $\leq 2\text{kHz}$

- 20 Hh, Hand transmitted vibration, for all measurement directions

Recommended signal's sample time: $\geq 0.02\text{ms}$ and $\leq 2\text{ms}$ or $\geq 500\text{Hz}$ and $\leq 50\text{kHz}$

per ISO 7505, 1st edition, 1986-05-01:

- 20 hand transmitted vibration, weighting filter.

Recommended signal's sample time: $\geq 0.02\text{ms}$ and $\leq 2\text{ms}$ or $\geq 500\text{Hz}$ and $\leq 50\text{kHz}$

per ISO 2631-1, 1st edition, 1985:

- 21 Weighting factors for transverse (x, y) vibrations, see table 3.

Recommended signal's sample time: $\geq 0.5\text{ms}$ and $\leq 100\text{ms}$ or $\geq 10\text{Hz}$ and $\leq 2\text{kHz}$

- 22 Weighting factors for longitudinal (z) vibrations, see table 3.

Recommended signal's sample time: $\geq 0.5\text{ms}$ and $\leq 50\text{ms}$ or $\geq 20\text{Hz}$ and $\leq 2\text{kHz}$

per ISO 2631-4, 2001:

- 23 Wb, passenger and crew comfort in fixed-guideway transport systems.

Recommended signal's sample time: $\geq 0.5\text{ms}$ and $\leq 5\text{ms}$ or $\geq 200\text{Hz}$ and $\leq 2\text{kHz}$

per ISO 2631-2, 2003:

- 24 Wm, human exposure to vibration in buildings.

Recommended signal's sample time: $\geq 0.2\text{ms}$ and $\leq 5\text{ms}$ or $\geq 200\text{Hz}$ and $\leq 5\text{kHz}$

per ISO 6954, 2000:

- 25 Acceleration input.

Recommended signal's sample time: $\geq 0.2\text{ms}$ and $\leq 5\text{ms}$ or $\geq 200\text{Hz}$ and $\leq 5\text{kHz}$

- 26 Velocity input.

Recommended signal's sample time: $\geq 0.5\text{ms}$ and $\leq 2\text{ms}$ or $\geq 500\text{Hz}$ and $\leq 2\text{kHz}$

per ISO 5349-1, 2001:

- 27 Hand transmitted vibration, weighting filter.

Recommended signal's sample time: $\geq 0.02\text{ms}$ and $\leq 2\text{ms}$ or $\geq 500\text{Hz}$ and $\leq 50\text{kHz}$

per ISO 8041, 2005:

28	Wb, passenger and crew comfort in fixed-guideway transport systems. Recommended signal's sample time: $\geq 0.5\text{ms}$ and $\leq 5\text{ms}$ or $\geq 200\text{Hz}$ and $\leq 2\text{kHz}$
29	Wc, seat-back measurement. Recommended signal's sample time: $\geq 0.5\text{ms}$ and $\leq 10\text{ms}$ or $\geq 100\text{Hz}$ and $\leq 2\text{kHz}$
30	Wd, x and y directions and for horizontal recumbent direction. Recommended signal's sample time: $\geq 0.5\text{ms}$ and $\leq 2\text{ms}$ or $\geq 500\text{Hz}$ and $\leq 2\text{kHz}$
31	We, measurement of rotational vibration. Recommended signal's sample time: $\geq 0.5\text{ms}$ and $\leq 2\text{ms}$ or $\geq 500\text{Hz}$ and $\leq 2\text{kHz}$
32	Wf, whole body low-frequency vibrations, motion sickness. Recommended signal's sample time: $\geq 5\text{ms}$ and $\leq 500\text{ms}$ or $\geq 2\text{Hz}$ and $\leq 200\text{Hz}$
33	Wh, hand transmitted vibration. Recommended signal's sample time: $\geq 0.02\text{ms}$ and $\leq 2\text{ms}$ or $\geq 500\text{Hz}$ and $\leq 50\text{kHz}$
34	Wj, vibration under the head of a recumbent person. Recommended signal's sample time: $\geq 0.5\text{ms}$ and $\leq 2\text{ms}$ or $\geq 500\text{Hz}$ and $\leq 2\text{kHz}$
35	Wk, z direction and for vertical recumbent direction, except head. Recommended signal's sample time: $\geq 0.5\text{ms}$ and $\leq 5\text{ms}$ or $\geq 200\text{Hz}$ and $\leq 2\text{kHz}$
36	Wm, human exposure to vibration in buildings. Recommended signal's sample time: $\geq 0.2\text{ms}$ and $\leq 5\text{ms}$ or $\geq 200\text{Hz}$ and $\leq 5\text{kHz}$

**Example**

```
SignalVib = VibrationFilter( Signal, 10, 0.125, 1000 )
```

Wk Weighting per ISO 2631-1

The signal is subjected to vibration weighting. The frequency-weighted signal is then time-weighted at a time constant of 0.125 s (FAST) and resampled with a factor of 1000.

The signal originally has a sampling frequency of 20 kHz, the vibration-weighted result has a sampling frequency of only 20 Hz.

VisAnyGreater

Return value is 1 if a vector element of b is larger than the corresponding element in c. Otherwise, the return value is 0.

a = VisAnyGreater(b, c)

a: Return value

b: Input vector series

c: Input vector series

**Example**

```
_tmpV1 = VectorizeAndSkip( Channel_001, 512, 0 )
_tmpV2 = VectorizeAndSkip( Channel_002, 512, 0 )
a = VisAnyGreater( _tmpV1, _tmpV2 )
```

a: 1 when a sample of Channel_001 larger than Channel_002

VMax

Maximum from vector: returns the maximum value from each input vector

a = **VMax**(**b**)

a : Output value	b : Input vector series

VMaxV

Maximum from vector series: maximum values from a reduced vector series

a = **VMaxV**(**b**, **WindowOption**, **RF**)

a : Output vector series	WindowOption : Compute maximum vector from	RF : Reduction factor (RF) in vectors; ≥ 1
b : Input vector series	1: RF input vectors 2: All current input vectors	

For each element of the vector, the maximum value of the element for a vector sequence is determined. The result is another vector and over time a vector sequence.

For a window option of 1, the maxima over RF vectors are found and one output vector is generated after each RF input vectors.

For a window option of 2, the maxima of all previous input vectors are found. After every RF input vectors, one result vector is found.

VMean

Mean from vector: returns the mean value from each input vector

a = **VMean**(**b**)

a : Output value	b : Input vector series

VMeanV

Mean from vector series: mean values from a reduced vector series

a = **VMeanV**(**b**, **WindowOption**, **RF**)

a : Output vector series	WindowOption : Compute mean vector from	RF : Reduction factor (RF) in vectors; ≥ 1
b : Input vector series	1: RF input vectors 2: All current input vectors	

For each element of the vector, the mean value of the element for a vector sequence is determined. The result is another vector and over time a vector sequence.

For a window option of 1, the means over RF vectors are found and one output vector is generated after each RF input vectors.

For a window option of 2, the means of all previous input vectors are found. After every RF input vectors, one result vector is found.

VMin

Minimum from vector: returns the minimum value from each input vector

a = **VMin**(**b**)

a : Output value	b : Input vector series

VMinV

Minimum from vector series: minimum values from a reduced vector series

a = **VMinV**(**b**, **WindowOption**, **RF**)

a : Output vector series	WindowOption : Compute minimum vector from	RF : Reduction factor (RF) in vectors; ≥ 1
b : Input vector series	1: RF input vectors	
	2: All current input vectors	

For each element of the vector, the minimum value of the element for a vector sequence is determined. The result is another vector and over time a vector sequence.

For a window option of 1, the minima over RF vectors are found and one output vector is generated after each RF input vectors.

For a window option of 2, the minima of all previous input vectors are found. After every RF input vectors, one result vector is found

VRedV

Reduction from vector series: reduces a vector series

a = **VRedV**(**b**, **RF**)

a : Output vector series	RF : Reduction factor as vectors; ≥ 1
b : Input vector series	

Every RF input vectors are reduced to a single output vector.

VRMS

RMS from vector: returns the RMS value of each input vector

a = **VRMS**(**b**)

a : Output value	b : Input vector series

VSum

Sum from vector: returns the sum of the elements in each input vector

a = **VSum**(**b**)

a : Output value	b : Input vector series

VValueAtXValue

Y-value at x-value: returns the y-value for a specified x-value

a = **VValueAtXValue**(**b**, **XValue**)

a : output value	XValue : X-value
b : Input vector series	range: x-offset up to delta-x * vector_length - x-offset

 Example

```
frqPos = VXValueOfMax( SPEK_PosA ) ; frqPos = Display variable
phaPos = VValueAtXValue( SPEK_PosP, frqPos )
```

VXValueOfMax

X-value of maximum: returns x-value of the maximum element in a vector

a = **VXValueOfMax**(**b**)

a : Output value	b : Input vector series

VXValueOfMin

X-value of minimum: returns x-value of the minimum element in a vector

a = **VXValueOfMin**(**b**)

a : Output value	b : Input vector series

VXValueWithYValue

X-value with y-value: Returns x-value for a specified y-value

a = **VXValueWithYValue**(**b**, **YValue**, **ErrorValue**)

a : Output value	YValue : Y-value to find
b : Input vector series	ErrorValue : Error value to return if y-value not found

9.2.9.3.20 W

WindRoseCorr

To Values less than zero 360 is added and from values greater than 360, 360 is subtracted.

a = WindRoseCorr(b)

a: result

b: argument

The function [NorthCorrection](#)^[354] corrects the discontinuity at due North according to the addition method. It prevents skipping from 360° to 0° in the averaging interval.

For values fluctuating around 360°, an average of 180° is avoided.

The result of averaging could lie outside of the wind rose range 0°..360°, e.g. 365°.

The function [WindRoseCorr](#) returns the result to the wind rose range (0°..360°), e.g. changes 365° to 5°.

This procedure is conditional on the wind direction not changing more than 100 degrees.

The only sensible way to use the functions [NorthCorrection](#)^[354] and [WindRoseCorr](#) is in a combination as illustrated in the example below. Instead of averaging, the standard deviation, i.e. the function [StDev](#)^[407], can be used.



Example

```
NC = NorthCorrection( Channel, 10 )
NC_Mean = Mean( NC, 10, 10 )
Result = WindRoseCorr( NC_Mean )
```


9.2.9.4 Functions with Control commands

Prerequisite:

Only usable with control commands!

Function Groups	Operators and Symbols	Additional functions
Measurement Process States 	(=) Equal Operator (Comparison) 	AND 
Loops, Conditions 	(<>) Inequality operator 	CloseSaveInterval 
Timer Functions 	(<) Less than operator 	CurrentValue 
	(<=) Less than or equal operator 	DelayBuffer* 
	(>) Greater than operator 	ECU* 
	(>=) Greater than or equal operator 	GetDateTime 
		NOT 
		OnECUCmdReturn ECU * 
		OR 
		ReadyForPowerOff 
		SyncOverload 
		VectorChannel 
		VectorChannelSet 
		VectorStatic 

9.2.9.4.1 Measurement process states

Operations can be carried out in response to different states of the measurement process. Calculations of channel data can only be performed in [OnTriggerMeasure](#). In the other states, only operations with single values (state variables (single values), bits, DACs, LEDs etc.) are allowed.

Variables for single values must be initialized in the [OnInitAll](#)-block if they are used in a formula. Each measurement state must be concluded with an [End](#) command.

OnInitAll

Executed once at the start

OnInitAll

All commands within this section are executed once at the start. This section is executed exactly one time after the configuration is loaded to the device. This makes this section suitable for initializations. Global variables (single values) in particular are initialized here for later use. Digital bits also can be initialized here.

Loops and conditions are supported within the `OnInitAll` routine.

All global variables later used for calculations in imc Online FAMOS or imc Inline FAMOS must first be initialized in the `OnInitAll` block.



Example

```
OnInitAll
  Virt_Bit01 = 1
  Voltage    = 10
  Counter    = 0
  LED_01     = 0
  int v[8]
  int i       = 0

  For i = 1 Till 8 Step 1
    v[i] = 1
  End
End
```

OnAlways

Executed constantly

OnAlways

The command lines in this section are constantly executed after the commands in `OnInitAll`.

This is not a one-off event, but is constantly repeated whether or not measurements are in progress.

This section works with global variables (single values) and digital bits. If calculations are performed on global variables (single values), they must first be initialized in the section `OnInitAll`.



Example

```
OnInitAll
  Reset = 0
End

OnAlways
  Virt_Bit10 = 1
  If DisplayVar_01 > 0
    DisplayVar_02 = 0
    Reset = 1
  Else
    DisplayVar_02 = 1
    Reset = 0
  End
End
```



Note

Writing to the device like "LED_01 = 1" should be avoided in `OnAlways`. It could cause an bus-overload.

OnMeasureStart

Executed at the push of the Start-button

OnMeasureStart

The command lines in this section are carried out one time only if the measurement is started, meaning that the button for starting the measurement is pressed.

This section works with global variables (single values) and digital bits. If calculations are performed on global variables (single values), they must first be initialized in the section [OnInitAll](#).



Example

```
OnInitAll
    Reset = 0
End

OnMeasureStart
    Virt_Bit10 = 1
    If DisplayVar_01 > 0
        DisplayVar_02 = 0
        Reset = 1
    Else
        DisplayVar_02 = 1
        Reset = 0
    End
End
```

OnMeasureEnd

Executed at the push of the Stop-button

OnMeasureEnd

The command lines in this section are carried out one time only if the measurement is completed, meaning that the button for ending the measurement is pressed or the test carried out finished independently.

This section works with global variables (single values) and digital bits. If calculations are performed on global variables (single values), they must first be initialized in the section [OnInitAll](#).



Example

```
OnInitAll
    Reset = 0
End

OnMeasureEnd
    Virt_Bit10 = 1
    If DisplayVar_01 > 0
        DisplayVar_02 = 0
        Reset = 1
    Else
        DisplayVar_02 = 1
        Reset = 0
    End
End
```

OnSyncTask

Run exactly upon elapse of the time interval set.

Prerequisite:

Only available in: imc Online FAMOS Professional!

OnSyncTask(CycleTime)

This segment forms the synchronous task, which is activated cyclically in synchronization with the device's AD converter. The command lines contained in this segment are performed exactly one time once the specified cycle time has elapsed again. A maximum of 4 synchronous tasks are possible.

The parameter Time is stated in s.

The following entries are allowed: 0.0001s, 0.0002s, 0.0005s, ..., 0.1s, 0.2s, 0.5s, 1s.



Example

```
OnSyncTask( 0.1 ) ; Cycle time of 0.1s
  If VirtBit_01 <> 0
    VirtBit_02 = 1
  Else
    VirtBit_02 = 0
  End
End
```

If the cycle time is insufficient for performing the commands, the beeper and LED_06 are activated.

Alternatively, this can be monitored using the command [SyncOverload](#)⁴¹².

OnTriggerStart

Run upon start of the measurement

OnTriggerStart(TriggerName)

The command lines in this section are executed a single time at the start of every measurement, at the exact moment the trigger is released.

This section works with global variables (single values) and digital bits. If calculations are performed on global variables (single values), they must first be initialized in the section [OnInitAll](#).



Example

```
OnInitAll
  Counter = 0
  VirtBit_02 = 1
End

OnTriggerStart( BaseTrigger )
  Counter = Counter + 1
  If VirtBit_01 > 0
    VirtBit_02 = 0
  End
End
```

OnTriggerEnd

Executed at the end of every measurement

OnTriggerEnd(TriggerName)

The command lines in this section are carried out one single time at the end of every measurement, at the moment that no more data are arriving due to the trigger.

This section works with global variables (single values) and digital bits. If calculations are performed on global variables (single values), they must first be initialized in the section [OnInitAll](#).



Example

```
OnInitAll
    Counter = 0
    VirtBit_02 = 0
End

OnTriggerEnd( BaseTrigger )
    Counter = Counter + 1
    If VirtBit_01 < 1
        VirtBit_02 = 1
    End
End
```

OnTriggerMeasure

Executed during a running measurement

OnTriggerMeasure(TriggerName)

The command lines in this section are executed during a running measurement. This is not a one-off event, instead, the execution of this section is constantly repeated between the trigger start and the trigger end.

This section works with channels and virtual channels, but also with single values and digital bits.

If calculations are performed on global variables (single values), they must first be initialized in the section [OnInitAll](#).



Example

```
OnTriggerMeasure( BaseTrigger )
    VirtChannel_001 = Channel_001 + 5
    If VirtBit_01 > 0
        VirtChannel_002 = Channel_002 + 10
    Else
        VirtChannel_002 = Channel_002 + 5
    End
End
```

OnTimer

Run upon elapse of the time interval set

OnTimer(Timer_ID)

The command lines in this section are executed once for each time the appropriate timer is released. A timer can be created using `StartTimerPeriodic` or `StartTimerSingle` (see [Timer functions](#) ⁴⁴⁶).

If calculations are performed on global variables (single values), they must first be initialized in the section `OnInitAll`.

The specified timer-ID must exactly match the one for the corresponding timer start function; the values permitted are 1 ... 8.



Example

```
OnTriggerStart( BaseTrigger )
    StartTimerPeriodic( 5, 0.1, 0.1 )
End

OnTimer( 5 )
    If VirtBit_01 > 0
        VirtBit_02 = 1
    Else
        VirtBit_02 = 0
    End
End
```

OnPowerOff

Run upon switch off the device.

Prerequisite:

Only available in: imc Online FAMOS!

OnPowerOff(cycle time)

This section performs the power off procedure which comes into effect when the device switched off. The battery in the device then takes over supply for a maximum of 8 s. Within this time, the application can be closed properly in the section `OnPowerOff`. When calling the function `ReadyForPowerOff`, device supply via the internal battery is ready to disabled.

One `OnPowerOff` section per synchronous task is allowed. The `OnPowerOff` section associated with the synchronous task is identified by the stated cycle time.

The cycle time parameter is stated in seconds. Any of the following cycle times may be specified: 0.0001 s, 0.0002 s, 0.0005 s, ..., 0.1 s, 0.2 s, 0.5 s, 1 s.



Example

```
OnSyncTask( 0.01 ) ; Cycle time: 0.01s
    DisplayVar_01 = pv.Channel_001
End

OnPowerOff( 0.01 ) ; Cycle time: 0.01s
    DisplayVar_01 = 0
    DOut01_Bit01 = 0
    DOut01_Bit02 = 0
    ReadyForPowerOff()
End
```

9.2.9.4.2 Loops, conditions

If condition, Else

This control command sets the operations appearing underneath to be carried out conditionally.

The operations are only executed if the condition is fulfilled. If the single value passed to it is not equal to 0, the condition is fulfilled. The operations must be concluded by End or Else.

Else denotes the start of operations to be performed if the prior If-condition is not fulfilled. The operations must be concluded by End.



Example

```
OnTriggerMeasure( BaseTrigger )
  If VirtBit_01 > 0
    VirtChannel_001 = Channel_001 + 1
  End
  If VirtBit_05 = 0
    VirtChannel_002 = Channel_002 + 5
  Else
    VirtChannel_002 = Channel_002 + 10
  End
End
```

A condition is usually employed in the manner shown in the example: two operands are compared according to a comparison operator. Naturally, a condition in the form of "If a" is also possible. In such a case, a must be a single value of type BOOL, or the Boolean result of a calculation.

The comparison operators "<", ">", "=" etc. also return variables of type BOOL. Instead of "If VirtBit_01 > 0", the command lines "a = VirtBit_01 > 0" and "If a" can be coded. Combinations of different conditions are also allowed, e.g. "If VirtBit_01 > 0 AND VirtBit_02 = 0".



Example

Functions and Bit-variables in conditions

Functions in conditions

```
If Greater(Current_RPMs, 2200) = 1
```

or the following alternative:

```
If Greater(Current_RPMs, 2200)
```

Bit-variables in conditions

```
If Virt_Bit01 = 1
```

or the following alternative:

```
If Virt_Bit01
```

**Example****Calculation results in conditions**

If the results of a calculation are certain to be in Boolean format, the variable can be used in the subsequent conditions without any assignment, as if it were a bit-variable.

```
a = b > c
If a = 1
```

or the following alternative:

```
a = b > c
If a
```

The variable "a" is a single value variable which is created in `OnInitAll`. "b > c" always returns either 1 or 0 as its result. Even if the variable contains a non-boolean result at a different location, it can be used in this case.

This behavior only applies to single value variables and not to device-variables.

**Note****Boolean variables from files**

See also the notes when using [imported variables in If conditions](#) ²⁶⁸.

Switch

This control command can be used to perform operations in dependence upon the value of the parameter b. In the list of case-blocks, different values for the parameter b are treated.

A Case-block is executed each time the numerical value of the parameter b is equal to the Case-command's numerical value. Any numerical values for the parameter b not explicitly listed are treated in the default case.

The Switch command is concluded with an End-command.

The parameter for a Case-command must be an integer ≥ 0 . Real numbers are not allowed as Case-command parameters.

**Example**

```
OnTriggerMeasure( BaseTrigger )
  Value = CurrentValue( Channel_001, 0, 0.0 )
  Switch Value
    Case 1
      VirtChannel_001 = Channel_001 + 1
      VirtChannel_002 = Channel_002 + 1
    End
    Case 2
      VirtChannel_001 = Channel_001 + 10
      VirtChannel_002 = Channel_002 + 2
    End
    Default
      VirtChannel_001 = Channel_001 + 100
      VirtChannel_002 = Channel_002 + 3
    End
  End
End
```


For

For Counter = Start Till End Step Iteration

A For-loop enables repeated execution of operations. The number of loop runs is always fixed in For-loops. The loop is carried out, starting when Counter = Start, until Counter is smaller than or equal to the value "End". After each run of the loop, the Counter is iterated upward by the number "Iteration".

The counter must be defined as an **integer single value** in the `OnInitAll`-block. The **maximum loop count** can also be specified by a **variable**. In that case it must be declared in an the `OnInitAll`-block as an integer and can be changed during the measurement.

Start, End and Iteration must be integers. The For-block is concluded by an "End" command. Single values and digital bits can be processed in loops, channel definitions are not permitted here.



Example

```
OnInitAll
    int i = 0
    HistoSum = 0
    int MaxLoops= 15
End

OnTriggerMeasure( BaseTrigger )
    Histo = Histogramm( Channel_001,  -10, 10, 16, 1, 1 )
    HistoSum = 0
    For i = 0 Till MaxLoops Step 1
        HistoSum = HistoSum + GetHistoValue( Histo, i )
    End
    VrtSum = Channel_001*0 + HistoSum
End
```

While condition

While condition

A While-loop enables repeated execution of operations. The number of loop runs for "While" depends on the condition, so the number isn't necessarily known when the loop call comes. The operations in the While-block are carried out until the condition is met, i.e. the parameter of the While instruction is not equal to 0. The While-block is concluded by an "End" command.

Single values and digital bits can be processed in loops, channel definitions are not permitted here.



Example

```
OnInitAll
  int i = 0
  HistoSum = 0
End

OnTriggerMeasure( BaseTrigger )
  Histo = ClHistogram( Channel_001,-10,10,16,1,1)
  HistoSum = 0
  i = 0
  While i <= 15
    HistoSum = HistoSum + GetHistoValue( Histo, i )
    i = i + 1
  End
  VrtSum = Channel_001*0 + HistoSum
End
```

The While-loop is carried out until the condition is fulfilled. If the condition is always fulfilled, the While-loop is never exited. While-loops are commonly used in the manner illustrated by the example: two operands are compared using a comparison operator. Combinations of conditions are also allowed: e.g "While $i \leq 15$ AND $HistoSum \leq 5000$ ".

Default

The Case-command states the particular value of the variable specified by a prior switch command. In the list of case-blocks, different values for the parameter of the switch command are treated. A Case-block is executed each time the numerical value of the parameter of the switch command is equal to the Case-command's numerical value. Any numerical values for the parameter of the switch command not explicitly listed are treated in the default case. The Default command is concluded with an End-command. The parameter for a Case-command must be an integer ≥ 0 . Real numbers are not allowed as Case-command parameters.



Example

```
OnTriggerMeasure( BaseTrigger )
  Value = CurrentValue( Channel_001, 0, 0.0 )
  Switch Value
    Case 1
      VirtChannel_001 = Channel_001 + 1
      VirtChannel_002 = Channel_002 + 1
    End
    Case 2
      VirtChannel_001 = Channel_001 + 10
      VirtChannel_002 = Channel_002 + 2
    End
    Default
      VirtChannel_001 = Channel_001 + 100
      VirtChannel_002 = Channel_002 + 3
    End
  End
End
```

9.2.9.4.3 Timer functions

Up to eight different timers can be implemented. Each timer must be started by a function. Both periodic timers (the timer is released periodically at fixed intervals) and one-off timers (the timer is released after a specified interval) can be realized. To stop a periodic timer, the function StopTimer must be used. Timers can be started and stopped in any state of the measurement process.

In the `OnTimer` block, only single values (e.g. single value variables, bits, LEDs, DACs, ...) can be subjected to calculations.

StartTimerPeriodic

Periodic timer. Creates a periodic timer.

StartTimerPeriodic (Timer_ID, TimeInterval, TimeInterval_1)

Timer_ID: Identifies the timer (1..8)

TimeInterval: Release upon elapse of time interval, in seconds

TimeInterval_1: Time in seconds, after which release occurs

The time-count begins when this function is called. When the time TimeInterval_1 has elapsed, the timer is released one single time. After that, the timer is released repeatedly after each interval of duration TimeInterval. When the timer is released, the `OnTimer` section of the corresponding Timer_ID is executed.

The time intervals can be specified to a precision of 0.0001 s (0.1 ms). TimeInterval must be at least 0.0001 s; TimeInterval_1 can also be 0 s.

The permitted maximum values for the parameters TimeInterval and TimeInterval_1 depend on the device types used.



Example

```
OnTriggerStart( BaseTrigger )
; Create periodic timer
StartTimerPeriodic ( 5, 0.5, 0.0 )
End

OnTimer( 5 )
If VirtBit_01 > 0
    LED_01 = 1
Else
    LED_01 = 0
End
End
```

StartTimerSingle

Single-use timer. Creates a timer for one-off use.

StartTimerSingle (Timer_ID, TimeInterval)

Timer_ID: Identifies the timer (1..8)

TimeInterval: Time in seconds, after which release occurs

The time-count begins when this function is called. When the time TimeInterval has elapsed, the timer is released one single time. Then, the `OnTimer` section of the corresponding Timer_ID is executed.

The time interval can be stated to a precision of 0.0001 s (0.1 ms) and must be at least 0.0001 s.

The permitted maximum value for the parameter TimeInterval depends on the device type currently in use.



Example

```
OnTriggerStart( BaseTrigger )
; Create a one-off timer
StartTimerSingle( 1, 5.0 )
End

OnTimer( 1 )
If VirtBit_01 > 0
    LED_01 = 1
Else
    LED_01 = 0
End
End
```

StopTimer

Deletes a timer. The timer cannot be released. It no longer exists.

StopTimer (Timer_ID)

Timer_ID: Identifies the timer (1..8)



Example

```
OnTriggerMeasure( BaseTrigger )
If VirtBit_01 > 0
    ; Create periodic timer
    StartTimerSingle( 5, 0.0 )
End
If VirtBit_02 > 0
    StopTimer( 5 ) ; Delete timer
End
End

OnTimer( 5 )
If VirtBit_05 > 0
    LED_01 = 1
Else
    LED_01 = 0
End
End
```

9.2.9.4.4 CanMsg

Prerequisite:

Only available in: imc Online FAMOS!

9.2.9.4.5 Controller functions

Prerequisite:

Only available in: imc Online FAMOS Professional!

CtPID

Create and initialize a PID-controller: Initializes a PID-controller with the specified values.

Controller = CtPID(P_component, I_component, D_component)

Controller: the controller-structure to be created

I_component: I-component

P_component: P-component

D_component: D-component

This function is called once for each controller used in the section `OnInitAll`. It creates a controller-structure. The constants passed are stated for a continuous (non-discrete) controller. They serve to initialize the elements `.KP`, `.KI`, `.KD`. All other elements are initialized as 0.0.

Except for `.DCutOff`, which is set to approx. 0.1 / [controller's cycle time].

The PID-controller's structure has been enhanced: In particular, input control, controller output limiting, and band limiting of the D-component are all possible.

Computation of the controller is performed in the `OnSyncTask`-block with the function `.Calc`. The elements of the structure can be queried and changed at any time, and the set point `.SetPoint` as well.



Example

```
OnInitAll
    EngineController = CtPID( 20.0, 0.5, 0 ) ; P, I, D
    EngineController.SetPoint = 6000
End

OnSyncTask( 0.1 )
    DAC_VoltageEngine = EngineController.Calc( pv.Speed )
    If VirtBit_01 <> 0
        EngineController.Reset = 1
    End
    If VirtBit_02 > 0
        EngineController.SetPoint = 2000
    End
End
```

CtPID.Calc

Compute PID-controller: calculation of the PID-controller.

ControllerOutput = CtPID.Calc(ProcessValue)

ControllerOutput: the controller's output

ProcessValue: instantaneous measurement value,
process value for the controller

The function calculates exactly one step of the controller. For the currently passed process value, the new controller output is determined as the return value. The return value is then typically passed to a DAC. The controller output is subsequently available in `.PV`; the controller output also in `.CO`.

When the controller's parameters are changed, these changes only take effect upon the next call of the function `.Calc`, because that is the first (and only) time a new controller output is computed.

The function `.Calc` is only allowed in synchronous tasks.



Example

```
OnInitAll
    EngineController = CtPID( 20.0, 0.5, 0 ) ; P, I, D
    EngineController.SetPoint = 6000
End

OnSyncTask( 0.1 )
    DAC_VoltageEngine = EngineController.Calc( pv.Speed )
    If VirtBit_01 <> 0
        EngineController.Reset = 1
    End
    If VirtBit_02 > 0
        EngineController.SetPoint = 2000
    End
End
```

CtTwoPos

Create and initialize a two-point controller: Initializes a two-position controller with the specified values.

Controller = CtTwoPos(Hysteresis, Output_Inversion)

Controller: The newly created controller structure

Output_Inversion: Should the output be inverted?

Hysteresis: The distance between two reversal points

0: Default

1: Invert output

This function is called once for each two-position controller used in the section `OnInitAll`. It creates a controller structure.

The controller structure contains a number of elements which are initialized by this function. Thus, the controller's set point `.SetPoint` is set = 0, and the controller output `.CO` is set = 0. With the output inverted, `.CO` is set = 1.

The hysteresis is the distance between the upper and the lower reversal point. It should be > 0.

The two reversal points lie symmetrically around the setpoint. If the setpoint is = 20, for instance, and the hysteresis = 2, then the resulting reversal points are 21 and 19.

The controller's method of functioning: If the process value is greater than the upper reversal point, then the controller output = 1 is generated and returned by the function `.Calc`. If the process value is lower than the lower reversal point, then the resulting controller output is = 0. If the process value is between the two reversal points, the controller output remains unchanged.

By means of the parameter `Output_Inversion`, the controller output can normally be inverted, so that when the process value is too high, the resulting controller output is = 0; and when the process value is too low, the resulting CO is = 1.

Computation of the controller is performed in the `OnSyncTask`-block with the function `.Calc`. The elements of the structure can be queried and changed at any time, and the set point `.SetPoint` as well.



Example

```
OnInitAll
    Thermostat = CtTwoPos( 2, 0 ) ; Hysteresis, invert output
    Thermostat.SetPoint = 20 ; optional: set setpoint
    DigitalOut_01 = 0 ; optional: output in rest state
End

OnSyncTask( 0.1 )
    DigitalOut_01 = Thermostat.Calc( pv.Temperature_01 ) ; set setpoint
    ...
    If VirtBit_01 <> 0
        Thermostat.SetPoint = 22.0 ; change setpoint
    End
End
```

CtTwoPos.Calc

Calculate two-point controller: computation of a two-point controller.

ControllerOutput = CtTwoPos.Calc(ProcessValue)

ControllerOutput: the controller's output

ProcessValue: instantaneous measurement value,
process value for the controller

The function computes exactly one controller step. For the currently passed process value, the new controller output is determined as the return value. The return value is then typically passed to a digital output. The process value is subsequently available in `.PV`; the controller output in `.CO`.

The controller output can take values between 0 and 1.

When the controller's parameters are changed, these changes only take effect upon the next call of the function `.Calc`, because that is the first (and only) time a new controller output is computed. The function `.Calc` is only permitted in synchronous tasks.



Example

```
OnInitAll
    Thermostat = CtTwoPos( 2, 0 ) ; Hysteresis, invert output
    Thermostat.SetPoint = 20 ; optional: set setpoint
    DigitalOut_01 = 0 ; optional: output in rest state
End

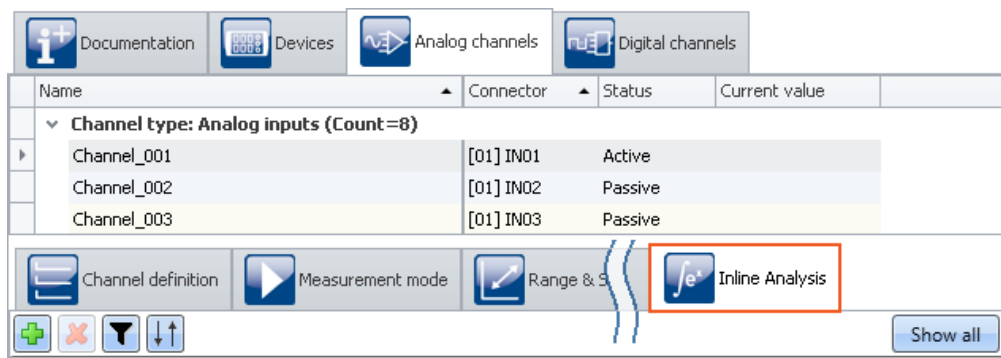
OnSyncTask( 0.1 )
    DigitalOut_01 = Thermostat.Calc( pv.Temperature_01 ) ; set setpoint
    ...
    If VirtBit_01 <> 0
        Thermostat.SetPoint = 22.0 ; change setpoint
    End
End
```

9.3 Inline Analysis

By using the following dialog, it is possible to compute a derivative or calculated channel from a input channel.

Setup page: "Analog channels"

Dialog: "Inline Analysis"



Note

Which Inline Functions are visible

Which Inline Functions are available depends on the product configuration and licensing.
(see "[Product Configuration / Licensing](#)" or Technical Data Sheet)

9.3.1 Inline Functions - operation

Display in the dialog

The dialog always displays all Inline Functions pertaining to the channels selected. If any selected channel has no Inline Functions, that area is empty. Clicking on the button "Show all" opens a tool window in which all Inline Functions for all channels are displayed.

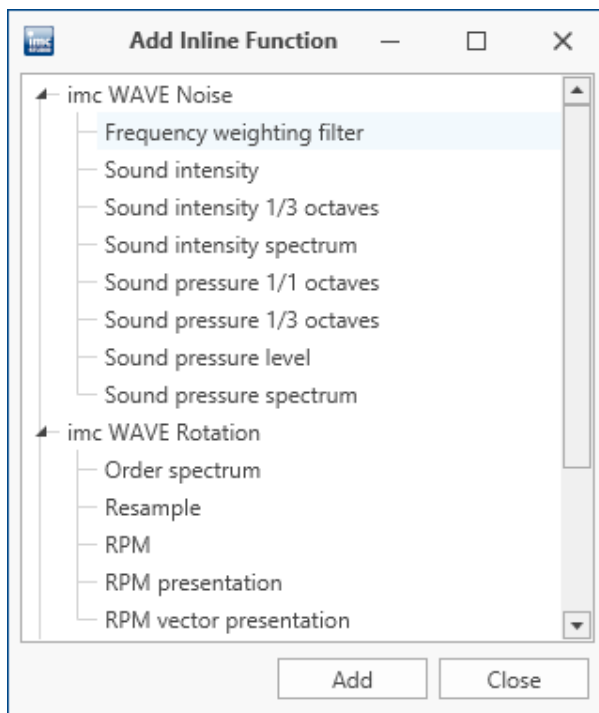


Note

Applying changes

Changes are only applied once you perform the action "[Process configuration](#)" (✓). This involves adding and deleting Inline Function as well as modifying the parameters. If the last device has been removed, then configuration processing is no longer possible, which means that the functions can no longer be cleared away.

Adding Inline Functions



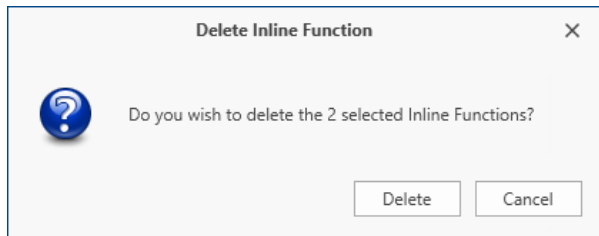
Adding new Inline Functions

- Select the channel from the table of channels or multiple channels if you wish to add a Inline Function to multiple channels at the same time.
- Click on the plus-sign . A dialog appears in which all possible Inline Functions for the channel are presented.
- Select a Inline Function and click on "Add". The Inline Function is added to the list. Alternatively, you can add the Inline Function by double-clicking it.

In this way you can add more Inline Functions, or close the dialog.

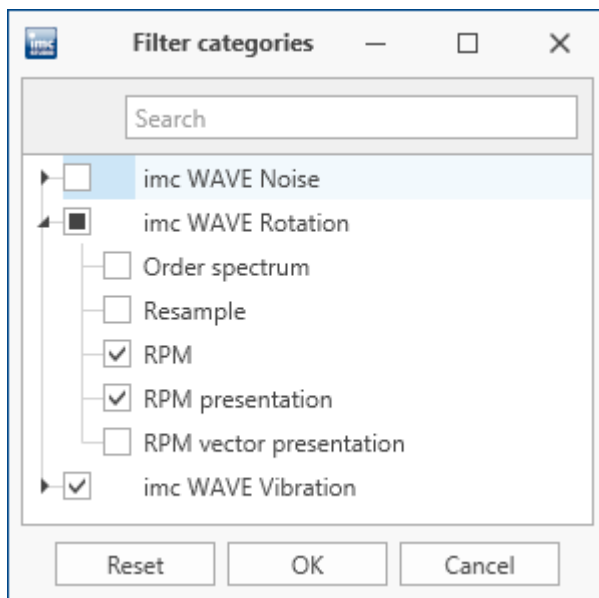
Deleting Inline Functions

- Select the derived channels to be deleted.
- Click on the X-symbol (✖). Ensure that the amount of Inline Functions to be deleted matches your intentions:



Deleting Inline Functions

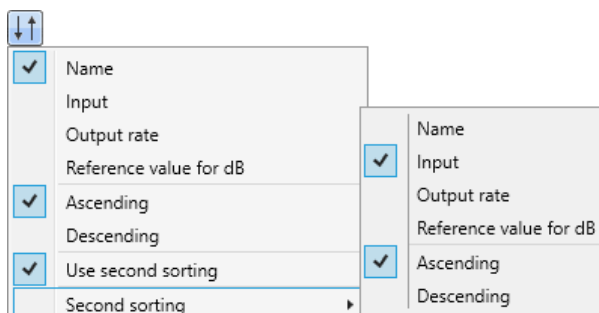
Filtering Inline Functions



Filtering Inline Functions

- Large numbers of Inline Functions and large channel counts can cause the list to become unwieldy. For better ease of managing, you can filter the view:
- Click on the Filter-symbol (🔍). In the dialog, select which categories to include.

Sorting Inline Functions



Sorting Inline Functions

- Within a single Function group, you can sort the entries in either ascending or descending order according to the name, input, output rate or reference value.
- Click on the Sorting symbol (↕). By means of the "Second sorting", you can refine the sorting order to the next level.

Parameters - Modifying the Inline Function

You can modify the parameters for the Inline Function. Depending on the type of Inline Function, a variety of input boxes or selection lists are available for this purpose.

Parameter	Description
Name	The parameter: "<i>Name</i>" defines the output-variable's name. This appears in the Data Browser. Once you set up a Inline Function, the "<i>Name</i>" reflects the name of the input channel (e.g. "<i>Channel_001_Leq_01</i>"). But when you subsequently change the input channel, the "<i>Name</i>" does not also change.
Input	The parameter: "<i>Input</i>" defines the input channel on which the calculation is performed. This, too, can be changed subsequently. Along with the inputs captured by the device (analog-, incremental counter-, 3PDI-), calculated channels such as imc Online FAMOS and imc Inline FAMOS can be used. The result of some Inline Function calculations can be used as input for another calculation (first level). However, a further calculation (second level) is not supported. Some inline functions have additional input channels, e.g. when determining the sound intensity from two microphone signals, the order time plot of the RPMs is also required for the order tracking analysis.

Additional parameters may also be available depending on the Inline Function type.

9.3.2 Introduction to the imc WAVE functions

The Inline Analyses "[imc WAVE Noise](#)⁴⁵⁵", "[imc WAVE Vibration](#)⁴⁶⁵" and "[imc WAVE Rotation](#)⁴⁶¹" offer numerous capabilities for spectral examinations in the fields of **acoustics**, **vibration analysis** and **order-tracking analysis**. What this means is taking signals measured over the time domain, typically generated by microphones and accelerometers, and performing spectral frequency analyses, such as Fast Fourier Transformation or 1/3- and octave spectrum analysis, on them in real time. For this purpose, a comprehensive variety of settings options are available in order to allow parameterization of the analyses according to established standards. This includes time- and frequency-weighting with standard procedures and filters, window functions, etc.

The Inline Analysis "[imc WAVE Structure](#)⁴⁷¹" offers functions for 2 channel **spectrum analysis**, e.g. frequency response function or coherence.

9.3.3 imc WAVE Noise

Analyzer functionalities

Functions for noise analysis, e.g. sound pressure level, Leq, sound intensity, FFT-spectra, 1/3-octave and 1/1-octave spectrum.

- Sound level meter according to Standard IEC 61672-1
- 1/3- and 1/1-octave analysis as 3D or averaged according to Standard IEC 61260-1
- FFT analysis as 3D or averaged
- Determining the sound intensity (time plot, narrow band spectrum and 1/3-octave spectrum) for measurements with an intensity probe

Functions	Description
Basic analysis functions	• FFT analysis (up to 131072 data points) as 3D or averaged • simultaneous calculation of octaves and 1/3-octaves in real time • calculation of spectra with constant absolute and relative bandwidth • selectable frequency range for the spectrum analysis • A-, B-, C- and linear frequency weighting selectable • time-weighting: Fast, Slow • window functions: rectangular, Hamming, Hanning, Flattop, Blackman and Blackman-Harris • frequency -weighted sound level with exponential time weighting • calculation of frequency-weighted averaging levels

Typical applications

- acoustics measurements, fully conformant to applicable industry standards
- acceptance and certification measurements with respect to noise emission and sound levels
- product qualification tests
- product optimization in the R&D field
- in-vehicle noise levels: tests for quality assessment and optimization
- comprehensive and wholistic investigations: root cause analysis of origins, propagation paths and effects of noise and vibration
- incorporation of acoustics expertise into general application fields of physical measurement engineering

Calculations

Z, A, B, C are standards-compliant designations for the frequency weighting (frequency filtering); standardized (acoustic) filters with specific frequency responses. "L" stands for (sound-) level.

Calculation	Description
Frequency weighting filter	Weighted time signal • Frequency-weighting: Z (without) or A, B or C. The result is also a time-domain signal.

Calculation	Description
Sound intensity level	Determination of sound intensity by means of a sound intensity probe according to the two-microphone technique. • Frequency-weighting: Z (without) or A, B or C • Distance (spacer): distance between microphones • Density: air density • Averaging: • <i>Interval:</i> The averaging is performed over the defined output interval (1 / output rate) as the arithmetic mean of the intensity plot. • <i>from start:</i> The averaging is performed over the entire plot over time, from the start of the measurement. The last value is the arithmetical mean of the plot of the intensity up until this point in time. • Output rate: for the averaging type "Interval", the averaging interval = 1 / output rate.
Sound intensity 1/3 octaves	Determination of sound intensity as 1/3-octaves by means of measurement by a sound intensity probe according to the two-microphone technique. • Frequency-weighting: Z (without) or A, B or C • Distance (Spacer): distance between microphones, length of spacer • Density: air density • Averaging: • <i>Interval:</i> The averaging is performed over the defined output interval (1 / output rate) as the arithmetic mean of each 1/3-octave band. • <i>from start:</i> The averaging is performed over all 1/3-octave bands from the start of the measurement as the arithmetic mean of each 1/3-octave band. • Lower/upper center frequencies: center frequencies for 1/3-octaves • Output rate: for the averaging type "Interval", the averaging interval = 1 / output rate.
Sound intensity spectrum (FFT)	Determination of the sound intensity as a narrow band spectrum by means of measurement with a sound intensity probe according to the two-microphone technique. • Frequency-weighting: Z (without) or A, B or C • Distance (spacer): Distance between microphones, length of spacer • Density: air density • Averaging: • <i>none:</i> no averaging • <i>Count:</i> Instead of <i>none</i> and <i>from start</i>, a number can be entered. The averaging is performed over the specified count of spectra as the arithmetic mean of the intensity spectrum for each frequency line. • <i>from start:</i> The averaging is performed over all spectra from the start of the measurement, as the arithmetic mean for each frequency line. • Samples (number): Count of input signal samples used to determine a spectrum. Sample value counts which are not powers of two are allowed. • Window (function): Rectangle, Hamming, Hanning, Blackman, Blackman-Harris, Flatop • Overlap: 0%, 10%, 25%, 33.33%, 50%, 66.67%, 75%, 90% • Log. axis: Generates an X-axis scaled for display with 1/3-octaves.

Calculation	Description
Sound pressure 1/1 octaves	Leq or LeqT 1/1-octaves: time-averaged sound level, equivalent continuous sound level over the measurement duration or with time interval T • for Time-weighting: "Leq from start" or "Leq in interval" • Frequency-weighting: Z (without) or A, B or C Lt 1/1-octaves: time weighted sound level • for Time-weighting: F (Fast), S (Slow), I (Impulse) or Peak • Frequency-weighting: Z (without) or A, B or C
Sound pressure 1/3 octaves	Leq or LeqT 1/3-octaves: time-averaged sound level, equivalent continuous sound level over the measurement duration or with time interval T • for Time-weighting: "Leq from start" or "Leq in interval" • Frequency-weighting: Z (without) or A, B or C Lt 1/3-octaves: time weighted sound level • for Time-weighting: F (Fast), S (Slow), I (Impulse) or Peak • Frequency-weighting: Z (without) or A, B or C
Sound pressure level	Mean level, equivalent continuous sound level of the frequency-weighted sound pressure signal over the measurement duration or with time interval T • for Time-weighting: "Leq from start" or "Leq in interval" • Frequency-weighting: Z (without) or A, B or C Lt and Lt max: frequency- and time-weighted sound level • for Time-weighting: F (Fast), S (Slow), I (Impulse) or Peak • Frequency-weighting: Z (without) or A, B or C • Maximum: Maximum of Lt

Calculation	Description
Sound pressure spectrum (FFT)	Fast Fourier Transform: Amplitude spectrum • Frequency-weighting: Z (without) or A, B or C • Result: Output of the results values as: • <i>RMS</i>: harmonic as RMS-value • <i>RMS²</i>: harmonic as squared RMS-value • <i>Peak</i>: harmonic as peak value/amplitude • Averaging: • <i>none</i>: no averaging • <i>Number</i>: averaging performed according to the "Avg.-type" selected, over the specified count of spectra • <i>from start</i>: averaging performed according to the "Avg.-type" selected, over all spectra as of the measurement start • Avg.-type: RMS (root mean square), Arithmetic Mean, Maximum (of all spectra), Minimum (of all spectra) • Samples (number): Count of input signal samples used to determine a spectrum. Sample value counts which are not powers of two are allowed. • Window (function): Rectangle, Hamming, Hanning, Blackman, Blackman-Harris, Flattop • Overlap: 0%, 10%, 25%, 33.33%, 50%, 66.67%, 75%, 90% • Log. axis: Creates an X-axis for the representation scaled with thirds. • Diff./Int.: • <i>without</i>: The time signal's amplitude spectrum is generated without differentiation or integration. • <i>differentiation</i>: The derivative of the amplitude spectrum is found. This is performed by multiplication of the spectrum by the angular frequency ω ($2\pi \cdot f$). • <i>double differentiation</i>: The second derivative of the amplitude spectrum is found. This is performed by multiplication of the spectrum by the square of the angular frequency ω ($2\pi \cdot f$). • <i>integration</i>: The integral of the amplitude spectrum is found. This is performed by division of the spectrum by the angular frequency ω ($2\pi \cdot f$). • <i>double integration</i>: The second integral of the amplitude spectrum is found. This is performed by division of the spectrum by square of the angular frequency ω ($2\pi \cdot f$). Note: In all derivation and integration procedures, the result for the DC component (frequency: 0 Hz) is 0 and thus has no significance.

Notes on Sound intensity

Sound intensity is determined by a two-microphone-based technique. For this purpose, an intensity probe is used in which two microphones are mounted at a small and fixed distance from each other.

The course of the sound pressure over time at each microphone is provided as a signal.

The calculation is performed as the multiplicative product of the sound pressure and the sound particle velocity.

The sound pressure used is the average of the sound pressures at the two microphones.

The sound particle velocity is determined by integration of the pressure differential. The pressure differential is an approximation of the derivative of the sound pressure appearing in the Euler equation.

To prevent the calculated integral from drifting away due to initial values and (small) offset errors, and thereby (strongly) distorting the result, a high-pass filter is applied.

For the purpose of the output, an average is taken, for which the parameter Time Weighting sets the averaging region.

All result values are intensities and stated in W/m^2 if the input data are scaled in Pa over s.

Sign

The calculated sound intensity is an average with an arithmetical sign.

The sign expresses the direction.

The sign is positive when the energy first hits the first microphone and next the second. This means it is positive when Microphone 1 is aimed at the source.

Even when the result is expressed in dB, the sign is included in the result. In order for this to be possible and remain clearly interpretable, the unsigned dB-result is limited to 0 dB. Additionally, for each of the three calculations, an extra sign-channel is generated.

- For the "overall" sound intensity, a time-domain channel with the current sign.
- For the 1/3-octave sound intensity, a segmented channel with a sign for each 1/3-octave.
- For the sound intensity spectrum, a segmented channel with a sign for each frequency reference point.

Limitations of the procedure

The microphone distance determines what frequency range is possible for the analysis:

When frequency components become very low, they are no longer possible to evaluate precisely because their phase differential is either very difficult or impossible to determine.

When frequency components become very high, they are no longer possible to evaluate precisely because their wavelength approaches the size of the distance between microphones.

The approximation of the derivative by a pressure differential, mentioned above, only works well when the frequency components are significantly less than 1/4 of the sampling frequency.

1/3-octave-based analysis

The calculation is performed for each 1/3-octave: First the 1/3-octave filtering is performed, next the calculation of the sound intensity.

The two frequency boundaries f_1 and f_2 should be specified as 1/3-octave midband frequencies, e.g. $f_1 = 8 \text{ Hz}$ and $f_2 = 12500 \text{ Hz}$. $f_1 < f_2$. The top 1/3-octave along with its frequency band must lie completely within one-half of the sampling frequency.

The top 1/3-octave should lie clearly within 1/4 of the sampling frequency.

The duration of transient oscillations for the 1kHz 1/3-octave is assumed to be 20ms. This duration is inversely proportional to the 1/3-octave frequency. For very low 1/3-octaves, this duration becomes substantial. The duration of the measurement must be set accordingly.

The result is a segmented data set. Each segment contains one 1/3-octave spectrum. The result's x-coordinate counts the 1/3-octaves (just like the FAMOS-function OctA). To achieve a useful display in the curve window, 1/3-octave labeling should be selected there.

The z-coordinate of the result represents the time. The time difference between the segments, Δz , corresponds to the parameter "Output-Interval".

The 1/3-octave filter and weightings are in accordance with IEC 651 (sound level meter), DIN 45652 (1/3-octave filter for electroacoustic measurements) and EN61260-1:2014 or IEC61260-1:2014 (band filter for octaves and fractions of octaves, filter Class 1).

In addition to the signal transients due to the 1/3-octave filter, the high-pass filter used also causes transient oscillations.

Total sound intensity

The input signals sound intensity is determined. There is no 1/3-octave filtering. The cutoff frequency of the high-pass filter used internally is 50 Hz.

The calculation is possible either without frequency weighting (Z) or with A, B or C weighting.

The signal transients are determined by the high-pass filter. Their duration is inversely proportional to the filter's cutoff frequency.

The result is a time-domain channel. The output interval specified is its sampling interval.

Sound intensity spectrum

The sound intensity spectrum is calculated via the cross-power spectrum from the complex spectra of the input signals. The integration is performed over the frequency domain.

Notes

The elapse time between the two microphones is of decisive importance for the analysis. All filters and pre-processing applied to the microphone signals, which affect the signal phase, must always proceed identically for both channels.

The transient oscillation procedure depends on what filters are used.

During the transient oscillation procedure, the intensities are assumed to be 0.0.

9.3.4 imc WAVE Rotation

Analyzer functionalities

Functions for analysis of rotating machinery

- Resampling of time-domain channels
- Order-tracking analysis as a 3D or averaged
- FFT analysis as a 3D or averaged
- Calculation of an input signal over time to a channel over RPMs (RPM-presentation)
- Calculation of spectrum over time or revolutions, to a 3D-channel over RPMs (RPM-vector-presentation)
- Calculation of 3D cross sections

Functions	Description
Basic analysis functions	• FFT analysis (up to 131072 data points) as 3D or averaged • Calculation of order-tracking spectra • Determining of RPMs, and Resample-functions • Window functions: rectangular, Hamming, Hanning, Flattop, Blackman and Blackman-Harris • Class-counting and representation of time data and spectra plotted over the RPMs

Typical applications

- product qualification tests
- product optimization in the R&D field
- End-Of-Line Quality Assurance
- comprehensive investigations of rotating machinery

Calculations

Calculation	Description
Frequency spectrum (FFT)	Fast Fourier Transform: Amplitude spectrum • Result: Output of the results values as: • <i>RMS</i>: harmonic as RMS-value • <i>RMS²</i>: harmonic as squared RMS-value • <i>Peak</i>: harmonic as peak value/amplitude • Averaging: • <i>none</i>: no averaging • <i>Number</i>: Averaging is performed according to the selected "<i>Avg.-type</i>" over the specified count of spectra • <i>from start</i>: Averaging is performed according to the selected "<i>Avg.-type</i>" over all spectra as of the measurement start • Avg.-type: RMS (root mean square), Arithmetic Mean, Maximum (of all spectra), Minimum (of all spectra) • Samples (number): Count of input signal samples used to determine a spectrum. Sample value counts which are not powers of two are allowed. • Window (function): Rectangle, Hamming, Hanning, Blackman, Blackman-Harris, Flattop • Overlap: 0%, 10%, 25%, 33.33%, 50%, 66.67%, 75%, 90% • Log. axis: Creates an X-axis for the representation scaled with thirds. • Diff./Int.: • <i>without</i>: The time signal's amplitude spectrum is generated without differentiation or integration. • <i>differentiation</i>: The derivative of the amplitude spectrum is found. This is performed by multiplication of the spectrum by the angular frequency ω ($2 \cdot \pi \cdot f$). • <i>double differentiation</i>: The second derivative of the amplitude spectrum is found. This is performed by multiplication of the spectrum by the square of the angular frequency ω ($2 \cdot \pi \cdot f$). • <i>integration</i>: The integral of the amplitude spectrum is found. This is performed by division of the spectrum by the angular frequency ω ($2 \cdot \pi \cdot f$). • <i>double integration</i>: The second integral of the amplitude spectrum is found. This is performed by division of the spectrum by square of the angular frequency ω ($2 \cdot \pi \cdot f$). Note: In all derivation and integration procedures, the result for the DC component (frequency: 0 Hz) is 0 and thus has no significance..
RPM	Determining the RPMs from, for example, squarewave, sinusoidal or incremental encoder signals, with the measurement mode "Impulse Time" • Schmitt trigger: upper and lower threshold • Counter: encoder, impulse time, sampled squarewave signal, sinusoidal signal, 1 missing tooth, 2 missing teeth • Counter pulses: tooth/impulse count, including "missing teeth" • without zero pulse: When this selection is deactivated, RPM-values are only determined as of the first impulse (before this, the RPM-value = 0 RPM). • Minimum RPM: bottom boundary of RPM-range

Calculation	Description
RPM presentation	The plot of a signal over the RPM-values is generated from the signal's time plot and the RPM-value's time plot. The desired RPM-range is subdivided among classes of equal width. • RPM: Time plot of RPMs • Averaging: RMS-value, arithmetic mean, minimum, maximum Specifies how values within the same RPM-class are subjected to calculations. • Interpolation: Tooth/impulse count, including "missing teeth" • Minimum RPM: bottom boundary of RPM-range • Maximum RPM: top boundary of RPM-range • RPM class width: width of one RPM-class
RPM vector presentation	The plot of a spectrum over the RPM-values is generated from the spectrum's time plot and the RPM-value's time plot. The desired RPM-range is subdivided among classes of equal width. • RPM: Time plot of RPMs • Averaging: Arithmetic mean, minimum, maximum, RMS-value Specifies how values within the same RPM-class are subjected to calculations. • Interpolation: Tooth/impulse count, including "missing teeth" • Minimum RPM: bottom boundary of RPM-range • Maximum RPM: top boundary of RPM-range • RPM class width: width of one RPM-class
Order spectrum (FFT)	Determination of the order spectrum • RPM: channel containing the RPM information for purpose of re-sampling • Filter type: low-pass (fixed) - Type of RPM-tracking filter serving as the anti-aliasing filter for resampling • Characteristic: Butterworth (fixed) • Order: integer between 1 and 10; filter order of the tracking filter • 3dB order: integer $\leq$ maximum order; order (line), at which the tracking low-pass filter dampens by 3dB • Maximum RPM: top boundary of desired RPM-range • Window (function): Rectangle, Hamming, Hanning, Blackman, Blackman-Harris, Flattop • Overlap: 0%, 10%, 25%, 33,33%, 50%, 66,67%, 75%, 90% • Maximum order: the highest order (line) displayed in the order spectrum • Samples (number): Count of sampled values of the resampled input signal used to determine a spectrum. Sample counts which are not powers of two are allowed. Derived as $2 * \text{"Maximum Order"} \text{ divided by "Resolution"}$ • Resolution: distance between lines in the order spectrum

Calculation	Description
Resample	Sampling of the input channel over the angle. The X-unit of the result is "revolutions". The values in the result are to be interpreted as RMS-values. • RPM: channel containing the RPM information for purpose of re-sampling • Filter type: low-pass (fixed) - Type of RPM-tracking filter serving as the anti-aliasing filter for resampling • Characteristic: Butterworth (fixed) • Order: integer between 1 and 10; filter order of the tracking filter • 3dB order: integer $\leq$ maximum order; order (line), at which the tracking low-pass filter dampens by 3dB • Maximum RPM: top boundary of desired RPM-range • Window (function): Rectangle, Hamming, Hanning, Blackman, Blackman-Harris, Flattop • Maximum order: the highest order contained in the sampled signal
Spectrum cut	By means of this function, it is possible to make a vertical cut in the spectrum at a given X-position. Thus for example, it is possible to determine the signal plot in the z-direction at a specific frequency or a sampling point. The input can be a frequency spectrum or an order tracking spectrum, as well as 1/3-octave and octave spectra. • X position: Position of the vertical cross section; or for a frequency spectrum, a frequency in Hz; or for an order spectrum, an order. In the case of a 1/3-octave or octave spectrum, the desired center frequency is selected.

9.3.5 imc WAVE Vibration

Analyzer functionalities

Functions for vibration analysis of human vibration and machine diagnostics

- Filters (LP, HP, BP, BS, simple/double integration, simple/double differentiation, envelope curve)
- Machine diagnostics according to Standard ISO 10816 and ISO 20816
- Human vibration filters according to Standard ISO 2631-1, ISO 8041, DIN EN 12299
- 1/3- and 1/1-octave analysis of vibration as 3D or averaged according to Standard IEC 61260-1
- FFT analysis as 3D or averaged

Functions	Description
Basic analysis functions	• FFT analysis (up to 131072 data points) as 3D or averaged • simultaneous calculation of octaves and 1/3-octaves in real time • calculation of spectra with constant absolute and relative bandwidth • selectable frequency range for the spectrum analysis • Human vibration filter, high-, low-, bandpass and band rejection • time-weighting: Fast, Slow • window functions: rectangular, Hamming, Hanning, Flattop, Blackman and Blackman-Harris • frequency -weighted sound level with exponential time weighting

Typical applications

- standards-conformant measurements for human vibration and machine diagnostics
- acceptance and certification measurements in conjunction with impact of vibration on humans
- in-vehicle seating comfort: Measurements for qualification and optimization
- hand-arm vibration: handheld machinery
- acceptance and certification measurements in conjunction with impact of vibration on machinery
- product qualification tests
- product optimization in the R&D field
- comprehensive and wholistic investigations: root cause analysis of origins, propagation paths and effects of vibration

Calculations

"L" stands for (vibration-) level.

Calculation	Description
Filter	• Filter type: without, Low-pass filter, High-pass filter, Band-pass filter, Band-stop filter • Characteristic: Butterworth, Bessel, Chebychev, Critical damping • Order: integer ≤ 100 • Cutoff frequency: up to one-half of the sampling rate • Diff./Int.: • <i>without:</i> Filtering performed without subsequent differentiation or integration of the time-domain signal. • <i>differentiation:</i> The filtered signal is differentiated in the time domain. • <i>double differentiation:</i> The filtered signal is differentiated twice in the time domain. • <i>integration:</i> The filtered signal is integrated in the time domain. • <i>double integration:</i> The filtered signal is integrated twice in the time domain. • <i>envelope:</i> The absolute value of the filtered signal is determined.
Filter rms	The filtering is performed in accordance with a specified frequency-weighting. Subsequently, calculation of the moving exponential RMS value (time weighting) is performed on the results. By means of re-sampling, the result can be reduced by an integer factor which is determined by the output rate. • Frequency-weighting: without or see "Possible filters"⁴⁶⁹ • Time-weighting: F (Fast = 0.125 s) or S (Slow = 1 s), I (Impulse), Peak, "<i>Leq from start</i>" or "<i>Leq in interval</i>" • Averaging time interval: Duration of interval with "<i>Leq in interval</i>" time weighting • Maximum: Determines the maximum among all values since the measurement start (Peak-Hold-Max) • Characteristic: For Frequency-weighting LP, HP, BP, BS - See "Filter" • Order: For Frequency-weighting LP, HP, BP, BS - See "Filter" • Cutoff frequency: For Frequency-weighting LP, HP, BP, BS - See "Filter" • Diff./Int.: See "Filter" • Output rate: same as input channel (maximum), or reduced by integer factor

Calculation	Description
Machine diagnosis	In accordance with ISO 10816/20816 - The function has two results channels. 1. Result (with specified name) in accordance with the settings for the filtering, time weighting, etc. 2. "Alert"-channel, containing the zone values (A = 0, B = 1, C = 2, D = 3), according to which zone the result (1.) lies in • Filter type: without, Low-pass filter, High-pass filter, Band-pass filter, Band-stop filter • Characteristic: Butterworth, Bessel, Chebychev, Critical damping • Order: integer ≤ 100 • (Lower/Upper) cutoff frequency: up to half the sampling rate • Time-weighting: • <i>RMS from start:</i> RMS (root mean square) values from start of measurement • <i>RMS in interval:</i> RMS (root mean square) values in the interval • <i>Peak value absolute in interval:</i> absolute value of the amplitude • <i>Peak value positive in interval:</i> absolute value of the positive values • <i>Peak value negative in interval:</i> absolute value of the negative values • <i>Peak-to-peak value from start:</i> positive - negative values as of start of measurement • <i>Peak-to-peak value in interval:</i> positive - negative values within the interval • <i>Crest factor from start</i> • <i>Crest factor in interval</i> $\text{Crest factor} = \frac{ \text{Input signal} }{\text{Input signal}_{\text{RMS}}}$ • Averaging time interval: Duration of interval with "in interval" time weighting • Maximum: Determines the maximum among all values since the measurement start (Peak-Hold-Max) • Diff./Int.: • <i>without:</i> The time signal is filtered without subsequent differentiation or integration. • <i>differentiation:</i> The derivative of the filtered signal is taken. • <i>double differentiation:</i> The second derivative of the filtered signal over the time domain is taken. • <i>integration:</i> The integral of the filtered signal over the time domain is taken. • <i>double integration:</i> The second integral of the filtered signal over the time domain is taken. • <i>envelope:</i> The absolute value of the filtered signal is determined. • Zone: A - very good (Value = 0), B - suitable (Value = 1), C - not suitable (Value = 2), and D - critical (Value = 3)  Boundary values can be specified for the zones. The results channels values are classified according to these boundary values and allocated to the corresponding zones. The result of the classification is represented in the "Alert"-channel by the values 0 (Zone A) through 3 (Zone D) and thus offers the ability to perform zone monitoring. • Output rate: max. like input channel, or reduced by integer factor

Calculation	Description
Vibration 1/1 octaves	Leq or LeqT 1/1-octaves: time-averaged vibration level, equivalent continuous vibration level over the measurement duration or with time interval T • for Time-weighting: "<i>Leq from start</i>" or "<i>Leq in interval</i>" • Frequency-weighting: without or human vibration filter (e.g. Hx, DIN 45671, ...) L_T 1/1-octaves: time weighted vibration level • for Time-weighting: F (Fast), S (Slow), I (Impulse) or Peak • Frequency-weighting: without or human vibration filter (e.g. Hx, DIN 45671, ...)
Vibration 1/3 octaves	Leq or LeqT 1/3-octaves: time-averaged vibration level, equivalent continuous vibration level over the measurement duration or with time interval T • for Time-weighting: "<i>Leq from start</i>" or "<i>Leq in interval</i>" • Frequency-weighting: without or human vibration filter (e.g. Hx, DIN 45671, ...) L_T 1/3-octaves: time weighted vibration level • for Time-weighting: F (Fast), S (Slow), I (Impulse) or Peak • Frequency-weighting: without or human vibration filter (e.g. Hx, DIN 45671, ...)

Calculation	Description
Vibration spectrum (FFT)	Fast Fourier Transform: Amplitude spectrum • Frequency-weighting: without or human vibration filter (e.g. Hx, DIN 45671, ...) • Result: Output of result values as: • <i>RMS</i>: harmonic as RMS-value • <i>RMS²</i>: harmonic as squared RMS-value • <i>Peak</i>: harmonic as peak value/amplitude • Averaging: • <i>none</i>: no averaging • <i>Number</i>: Averaging is performed according to the selected "Avg.-type" over the specified count of spectra • <i>from start</i>: Averaging is performed according to the selected "Avg.-type" over all spectra as of the measurement start • Avg.-type: RMS (root mean square), Arithmetic Mean, Maximum (of all spectra), Minimum (of all spectra) • Samples (number): Count of input signal samples used to determine a spectrum. Sample value counts which are not powers of two are allowed. • Window (function): Rectangle, Hamming, Hanning, Blackman, Blackman-Harris, Flattop • Overlap: 0%, 10%, 25%, 33.33%, 50%, 66.67%, 75%, 90% • Log. axis: Creates an X-axis for the representation scaled with thirds. • Diff./Int.: • <i>without</i>: The time signal's amplitude spectrum is generated without differentiation or integration. • <i>differentiation</i>: The derivative of the amplitude spectrum is found. This is performed by multiplication of the spectrum by the angular frequency ω ($2\pi \cdot f$). • <i>double differentiation</i>: The second derivative of the amplitude spectrum is found. This is performed by multiplication of the spectrum by the square of the angular frequency ω ($2\pi \cdot f$). • <i>integration</i>: The integral of the amplitude spectrum is found. This is performed by division of the spectrum by the angular frequency ω ($2\pi \cdot f$). • <i>double integration</i>: The second integral of the amplitude spectrum is found. This is performed by division of the spectrum by square of the angular frequency ω ($2\pi \cdot f$). Note: In all derivation and integration procedures, the result for the DC component (frequency: 0 Hz) is 0 and thus has no significance.

Possible filters

- Low-pass, High-pass, Band-pass or Band-stop
- Wk, z direction and for vertical recumbent direction, except head. As per ISO 2631-1:1997
- Wd, x and y directions and for horizontal recumbent direction. As per ISO 2631-1:1997
- Wf, motion sickness. As per ISO 2631-1:1997
- Wc, seat-back measurement. As per ISO 2631-1:1997
- We, measurement of rotational vibration. As per ISO 2631-1:1997
- Wj, vibration under the head of a recumbent person. As per ISO 2631-1:1997
- Hx, whole body vibration, standing, sitting position: x- and y-directions. Recumbent position: y- and z-directions. As per DIN 45671-1:1990-09
- Hz, whole body vibration, standing, sitting position: z-direction. As per DIN 45671-1:1990-09

- Hxl, whole body vibration, recumbent position: x-direction. As per DIN 45671-1:1990-09
- Hb, whole body vibration, body posture not specified. As per DIN 45671-1:1990-09
- Hh, hand-arm-vibration, for all directions. As per DIN 45671-1:1990-09
- hand transmitted vibration, weighting filter. Nach ISO 7505:1986-05
- Weighting factors for transverse (x, y) vibrations. As per withdrawn ISO 2631-1:1985
- Weighting factors for longitudinal (z) vibrations. As per withdrawn ISO 2631-1:1985
- Wb (passenger and crew comfort in fixed-guideway transport systems). As per ISO 2631-4:2001
- Wm (human exposure to vibration in buildings). As per ISO 2631-2:2003
- Acceleration input. Nach ISO 6954:2000
- Velocity input. Nach ISO 6954:2000
- Wh (hand transmitted vibration, weighting filter). As per ISO 5349-1:2001
- Wb (passenger and crew comfort in fixed-guideway transport systems). As per ISO 8041:2005
- Wc (seat-back measurement). As per ISO 8041:2005
- Wd (x and y directions and for horizontal recumbent direction). As per ISO 8041:2005
- We (measurement of rotational vibration). As per ISO 8041:2005
- Wf (Whole body low frequency, motion sickness). As per ISO 8041:2005
- Wh (hand transmitted vibration). As per ISO 8041:2005
- Wj (vibration under the head of a recumbent person). As per ISO 8041:2005
- Wk (z direction and for vertical recumbent direction, except head). As per ISO 8041:2005
- Wm (human exposure to vibration in buildings). As per ISO 8041:2005
- Wb (railway applications, passenger traveling comfort, Z ground, Z seat shell). As per EN 12299:2009
- Wc (railway applications, passenger traveling comfort, X seat backrest). As per EN 12299:2009
- Wd (railway applications, passenger traveling comfort, X ground, Y ground, Y seat shell). As per EN 12299:2009
- Wp (railway applications, passenger traveling comfort, Y ground, phi ground). As per EN 12299:2009

9.3.6 imc WAVE Structure

Analyzer functionalities

Functions for 2-channel analysis in the frequency domain

- Generation of output signals for the purpose of subsequent modal analysis
- Calculation of transfer functions with noisy input- and/or output signals
- Calculation of the coherence as a quality indicator
- Power rating by means of the cross-power-spectrum and spectral power density
- FFT analysis as 3D or averaged

Functions	Description
Basic analysis functions	• FFT analysis (up to 131072 data points) as 3D or averaged • Power density spectrum (up to 131072 points) as 3D or averaged • Coherence (up to 131072 points) averaged • Cross power density (up to 131072 points) spectrum as 3D or averaged • Frequency response (up to 131072 points) function as 3D or averaged

Typical applications

- product qualification tests
- product optimization
- evaluation of signal power and structural behavior

Calculations

Calculation	Description
Frequency spectrum (FFT)	Fast Fourier Transform: Amplitude spectrum • Result: output of the results values as: • <i>RMS</i>: harmonic as RMS-value • <i>RMS²</i>: harmonic as squared RMS-value • <i>Peak</i>: harmonic as peak/amplitude value • Averaging: • <i>none</i>: no averaging • <i>Number</i>: Averaging is performed according to the "Avg.-type" selected, over the specified count of spectra • <i>from start</i>: Averaging is performed according to the "Avg.-type" selected, over all spectra since the measurement start • Avg.-type: RMS (root mean square), Arithmetic Mean, Maximum (of all spectra), Minimum (of all spectra) • Samples (number): Count of input signal samples used to determine a spectrum. Sample value counts which are not powers of two are allowed. • Window (function): Rectangle, Hamming, Hanning, Blackman, Blackman-Harris, Flattop • Overlap: 0%, 10%, 25%, 33.33%, 50%, 66.67%, 75%, 90% • Log. axis: Creates an X-axis for the representation scaled with thirds. • Diff./Int.: • <i>without</i>: The time signal's amplitude spectrum is generated without differentiation or integration. • <i>differentiation</i>: The derivative of the amplitude spectrum is found. This is performed by multiplication of the spectrum by the angular frequency ω ($2 \cdot \pi \cdot f$). • <i>double differentiation</i>: The second derivative of the amplitude spectrum is found. This is performed by multiplication of the spectrum by the square of the angular frequency ω ($2 \cdot \pi \cdot f$). • <i>integration</i>: The integral of the amplitude spectrum is found. This is performed by division of the spectrum by the angular frequency ω ($2 \cdot \pi \cdot f$). • <i>double integration</i>: The second integral of the amplitude spectrum is found. This is performed by division of the spectrum by square of the angular frequency ω ($2 \cdot \pi \cdot f$). Note: In all derivation and integration procedures, the result for the DC component (frequency: 0 Hz) is 0 and thus has no significance..
Coherence	Determines the coherence between the input and the output time plots, over the frequency. • Input: time plot of input (1st input signal) • Output: time plot of output (2nd input signal) • Samples (number): Sample count of the input signals which are used to determine a spectrum. Sample value counts which are not powers of two are allowed. • Window (function): Rectangle, Hamming, Hanning, Blackman, Blackman-Harris, Flattop • Overlap: 0%, 10%, 25%, 33.33%, 50%, 66.67%, 75%, 90% • Log. axis: Generates an X-axis scaled for display with 1/3-octaves.

Calculation	Description
Cross power spectrum	Determines the cross-power spectrum of the input and output signals over the frequency. The function has two results channels: 1. Amplitude spectrum 2. Phase spectrum • Input: time plot of input (1st input signal) • Output: time plot of output (2nd input signal) • Averaging: • <i>none</i>: no averaging • <i>from start</i>: Averaging is performed separately for the Real- and Imaginary parts over all spectra for each frequency line, beginning at the measurement start. • Samples (number): Sample count of the input signals which are used to determine a spectrum. Sample value counts which are not powers of two are allowed. • Window (function): Rectangle, Hamming, Hanning, Blackman, Blackman-Harris, Flattop • Overlap: 0%, 10%, 25%, 33.33%, 50%, 66.67%, 75%, 90% • Log. axis: Generates an X-axis scaled for display with 1/3-octaves.
Power spectrum density (PSD)	Determines the power density spectrum (normalized power spectrum with frequency resolution) of the input and output signals over the frequency. The function has two results channels: 1. Amplitude spectrum 2. Phase spectrum • Input: time plot of input (1st input signal) • Output: time plot of output (2nd input signal) • Averaging: • <i>none</i>: no averaging • <i>from start</i>: Averaging is performed separately for the Real- and Imaginary parts over all spectra for each frequency line, beginning at the measurement start. • Samples (number): Sample count of the input signals which are used to determine a spectrum. Sample value counts which are not powers of two are allowed. • Window (function): Rectangle, Hamming, Hanning, Blackman, Blackman-Harris, Flattop • Overlap: 0%, 10%, 25%, 33.33%, 50%, 66.67%, 75%, 90% • Log. axis: Generates an X-axis scaled for display with 1/3-octaves.

Calculation	Description
Frequency response function (FRF)	Determines the transfer function (Frequency response function) from the input time domain signal to the output time domain signal over the frequency. The function has two results channels: 1. Amplitude spectrum 2. Phase spectrum • Input: time plot of input (1st input signal) • Output: time plot of output (2nd input signal) • Averaging: • <i>none</i>: no averaging • <i>from start</i>: Averaging is performed separately for the Real- and Imaginary parts over all spectra for each frequency line, beginning at the measurement start. • Type: • <i>H1: G_{xy} / G_{xx}</i>: Measurement with noisy output signal, calculated from the auto- and cross-spectrum • <i>H2: G_{yy} / G_{yx}</i>: Measurement with noisy input signal, calculated from the auto- and cross-spectrum • <i>HV: $(G_{xy} / G_{xy}) * \sqrt{G_{yy} / G_{xx}}$</i>: Geometric mean of H1 and H2. The input and output are noisy • Samples (number): Sample count of the input signals which are used to determine a spectrum. Sample value counts which are not powers of two are allowed. • Window (function): Rectangle, Hamming, Hanning, Blackman, Blackman-Harris, Flattop • Overlap: 0%, 10%, 25%, 33.33%, 50%, 66.67%, 75%, 90% • Log. axis: Generates an X-axis scaled for display with 1/3-octaves.

9.3.7 Monitoring the overmodulation and under range

Overmodulation of "analog channels"

Monitoring the overmodulation of "analog channels" (violations of the specified measurement range) as per the standard: "DIN EN 61672-1".

For purposes of display and evaluation of the overmodulation a variable is provided: "WAVE_OverloadVariable". It indicates that one measurement channel is reporting overmodulation but not which channel.

When overmodulation is detected, then for **one second** the variable "WAVE_OverloadVariable" takes the value "1". After that one second the system checks whether the overmodulation is still happening.

- if "no", the variable reverts to "0".
- if "yes", the value "1" is retained until there is no more overmodulation. Then the variable reverts to "0".

The following channels are monitored: active analog channels belonging to the firmware groups A and B (imc DEVICES and imc DEVICEcore), which are subjected to **calculations in an Inline Analysis** function. No other channels such as incremental counters or 3rd-party device channels.

Sound pressure level under range (imc WAVE Noise)

Fall of the calculated sound pressure level under range as per the standard: "DIN_EN_61672".

In order to determine the level linearity according to the standard, it is important that the under range is observed and evaluated taking into account the measuring range of the input channels. For this purpose, a supplemental variable, "WAVE_UnderrangeVariable", is available (in parallel) for all existing sound level calculations in a measurement. This variable indicates that a calculated sound pressure level reports under range, but not which calculation.

If an under range is detected, the variable "WAVE_UnderrangeVariable" is set to "1" at least for one second. After elapse of that one second, the system checks whether any calculation of sound pressure level is still reporting under range.

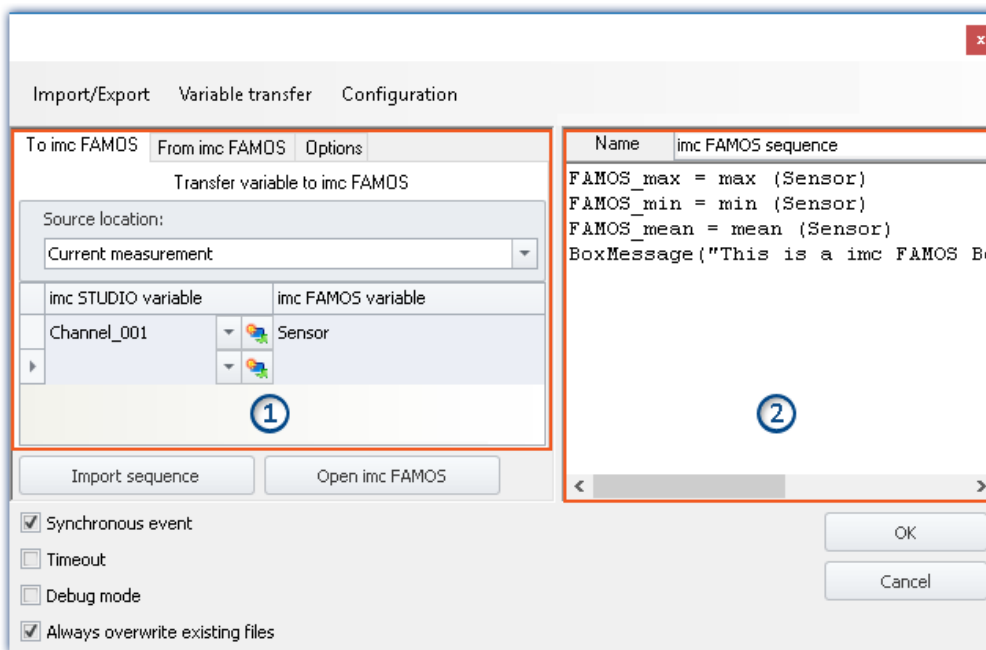
- If "no", the variable is set back to "0".
- If "yes", the value "1" is retained until the level is no longer under range. Then the variable is set back to "0".

Only such sound pressure level calculations are monitored for under range whose input channels originate from devices belonging to firmware groups A or B (imc DEVICES and imc DEVICEcore).

Sound pressure level calculations whose input channels have no measurement range such as 3rd-party device channels are not monitored.

9.4 imc FAMOS dialog

The dialog consists of multiple areas:



imc FAMOS Sequence Editor and transfer table of imc STUDIO variables to imc FAMOS

- [Transfer table](#) ⁴⁷⁸ (1): **Transfer of the imc STUDIO-variables** is accomplished by means of this table.
- imc FAMOS Sequence Editor (2): Here, the **imc FAMOS sequence in use** is displayed and can be edited.

Procedure

- The **measured variables** set in the [Transfer Table](#) ⁴⁷⁸ under "To imc FAMOS" **are transferred to imc FAMOS**.
- The **calculations** belonging to the sequence are **performed**.
- The **result variables** set in the [Transfer Table](#) ⁴⁷⁸ under "From imc FAMOS" **are transferred to imc STUDIO**.

Editing the imc FAMOS sequence

In order to create a imc FAMOS sequence, there are a variety of techniques:

- By editing in the imc FAMOS **Sequence-Editor**.
- Using the **Start command for imc FAMOS** via the button "*Open imc FAMOS*". (Preferable when not knowing the imc FAMOS functions and their parameters)
- By **importing** an existing **sequence file** using the button "*Import Sequence*".

To edit or test a sequence in imc FAMOS, use the button "*Open imc FAMOS*".

Dialog interface

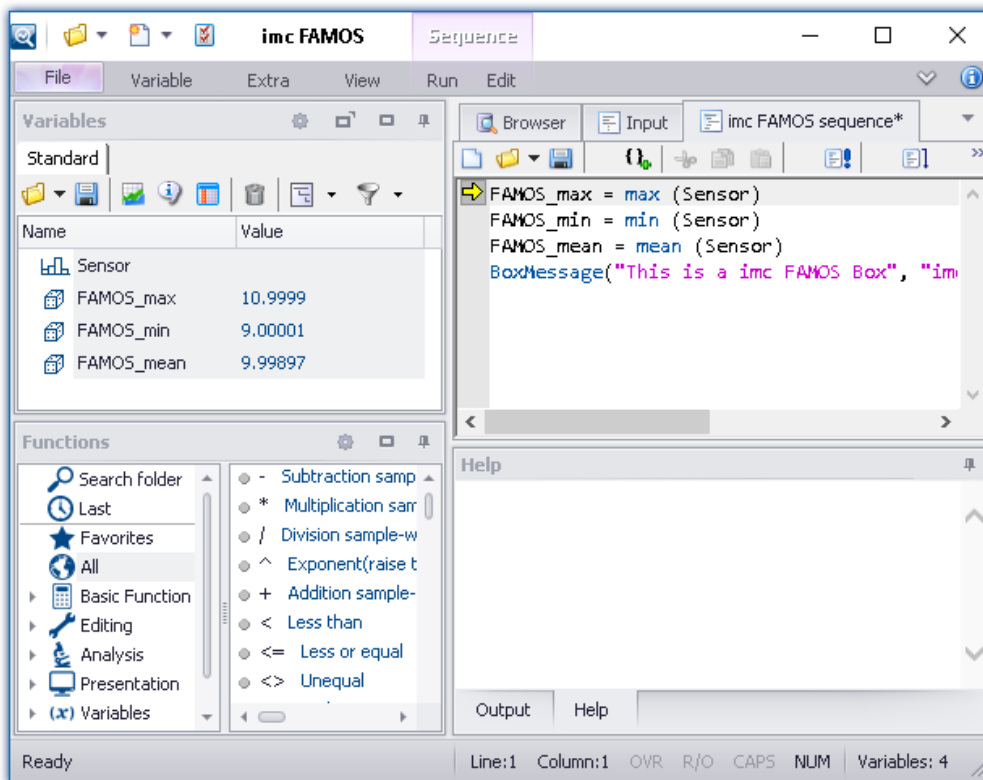
Checkbox	Description
Synchronous event	"<i>Synchronous event</i>" means that the source waits until imc FAMOS has concluded the sequence. This means: • Sequencer: The next command line is only started when imc FAMOS has concluded the sequence. • Automation: The next step only starts when imc FAMOS has concluded the sequence. If "<i>Synchronous event</i>" is deactivated, the new command line or step runs immediately. Meanwhile, the sequence continues to run in parallel. See also the notes on "Parallel processing of sequences"⁴⁸⁴.
Timeout	The option "<i>Timeout</i>" (only available if the dialog is called from a command) only appears if "<i>Synchronous Event</i>" is activated. • At the latest, after a specified timeout time, the Sequencer runs the next line.
Debug Mode	If the command is executed while the option "<i>Debug Mode</i>" is activated, then imc FAMOS is opened. In this case, the sequence is not run automatically. In Debug Mode, you can run the sequence step-by-step, and make changes which are saved in the command. If imc FAMOS is closed, the results are transferred to imc STUDIO, if they are available in the table "<i>From imc FAMOS</i>". The command is only concluded as of this moment in time. Please note that in Debug Mode, the command needs longer to execute: • With the option: "Synchronized event" active: The subsequent commands wait until the imc FAMOS-command has concluded. • With the option: "Synchronized event" deactivated: Required results may not exist in time since the command is still working. As well, the option: "Timeout" should not be activated, since it can cause premature closing of the command, although imc FAMOS has not yet been ended.
Always overwrite existing files	If this option is activated, files having the same name in the destination folder will be overwritten without a confirmation prompt. This option is useful for automated routines.

Open imc FAMOS

Alternatively, **imc FAMOS** can be started using the button "*Open imc FAMOS*". In this case, the imc FAMOS Editor is used for making entries. You have the entire scope of the Function Assistant's functionality, help texts, etc., at your disposal.

In imc FAMOS, you can run the sequence directly for testing purposes. To this end, the variables in the table: "*To imc FAMOS*" are automatically set up with the current values. They appear in imc FAMOS's Variables list. If the variables do not yet exist in imc STUDIO at that moment, an empty variable is used in substitute (see [note](#) ⁴⁷⁸ below).

The results of the test evaluation are not transferred back to imc STUDIO.



imc FAMOS as editor

Save the sequence if you wish to use the changes in imc STUDIO.

 Note
Variable not present

If a variable does not exist in imc STUDIO when it is to be transferred to imc FAMOS, an "empty" variable is created in imc FAMOS as a substitute.

- Data type: Normal data set
- X-Delta: 1
- Total size: 0

Using the imc FAMOS-paths

The default paths set in imc FAMOS also apply to the command.

E.g. "SEQ MySequence" is premised on "MySequence.seq" being located in the default sequence folder of the installed imc FAMOS program. Alternatively, it is possible to specify the absolute path, e.g. "SEQ "D:\SEQ\MySequence.seq"".

Transferring the files used when exporting to other computers

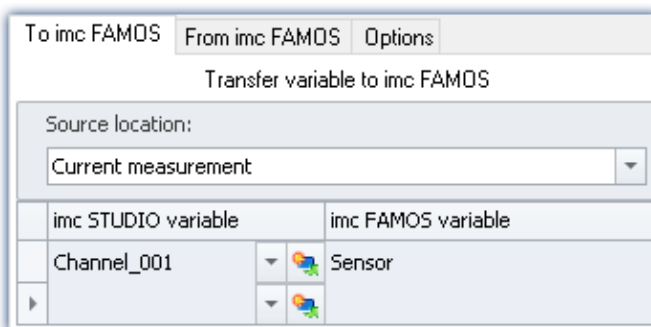
When exporting an experiment, the external files called in the sequence are not automatically included! Copy these files separately.

The exception is the files located in the experiment's subfolder "Meta". There, you are able to save your own files about the experiment, such as your own metadata, imc FAMOS-sequences, info-files, ... (for more information, see the section "[Experiments, projects and the database](#)"⁴⁶).

9.4.1 Transfer table

Transfer of the imc STUDIO variables or the imc FAMOS variables is accomplished by means of the transfer table. It is possible to rename the variables. In this way, existing sequences can be adopted directly.

Passing variables to the sequence ("To imc FAMOS")



imc STUDIO variable	imc FAMOS variable
Channel_001	Sensor

The variables entered in the column: "*imc STUDIO variable*" are passed to the imc FAMOS sequence. There, they are assigned the name of the associated cell in the column: "*imc FAMOS variable*".

The variable does not need to be already registered in imc STUDIO. In this case, it is created as an "empty" variable in imc FAMOS (see [note](#)⁴⁷).

In the example shown, the measured variable "*Channel_001*" is passed to the imc FAMOS sequence

as the variable "*Sensor*".

You are able to transfer placeholders

Where is the measurement "x" saved? Which test object is used? Such information can be transferred directly to imc FAMOS by means of placeholders (). When the command is run, the placeholder is resolved and transferred.

Source: Selecting the source (measurement)

All variables defined are transferred from the source selected.

Source	Description
Current measurement	The current measured data are transferred. Note that in this case, only the amount of circular buffer memory set for the curve window display is available for use. This may be only a fraction of the entire measurement.
Last completed measurement	If the measured data are saved, the last measurement saved is loaded automatically. The associated variables are passed to the imc FAMOS sequence.  More detailed information on the "Last measurement" is presented in the chapter: "<i>Storage options and directory structure</i>" > "The last measurement"²¹⁹
Measurement number Measurement#<No>	The variables belonging to the measurement with the respective measurement number are passed.

 Note

Variables from varying measurements

In some cases, variables from varying measurements need to be transferred to imc FAMOS. Independent of what "Source" is set, you can define in the Variables list whether a variable is to be transferred from a different measurement.

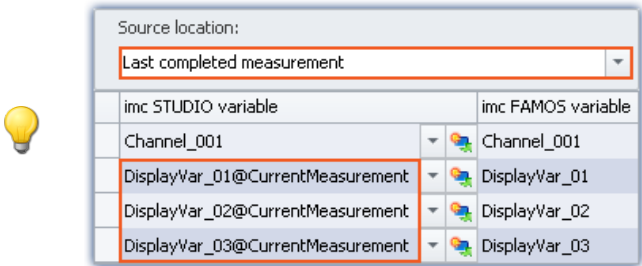
In this case, use the following syntax for the imc STUDIO variable:

Syntax	Example
@<MeasurementName>	<VariableName>@<MeasurementName> e.g. Channel_001@2010-11-15 15-44-03 (1)
@Measurement#<MeasurementNumber>	<VariableName>@Measurement#<MeasurementNumber> e.g. Channel_001@Measurement#1
@LastMeasurement	corresponds to: " <i>Last completed measurement</i> "
@CurrentMeasurement	corresponds to: " <i>Current Measurement</i> "

For measurement names, an **input support tool** is available: in the input box behind the channel name, click on the keyboard combination: <CTRL> + <SPACE>. You are then provided with a list of various input options. Select one and modify the results if appropriate.

Sample application: Variables from a saved measurement in combination with variables from "*Current Measurement*"

The analysis of measured data often requires the use of additional parameters which are saved with variables under "*Current Measurement*". With this imc FAMOS command, you are able to transfer variables both from a saved measurement and from "*Current Measurement*".

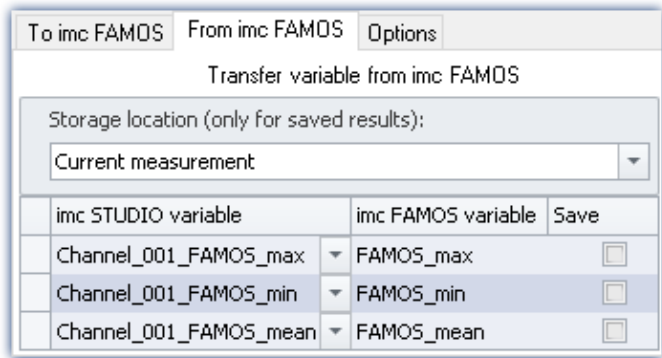


In this example, a particular measurement is selected. Additionally, variables from "*Current Measurement*" are also transferred. In this case, use the following syntax for the imc STUDIO variable:

<VariableName>@CurrentMeasurement

Example: DisplayVar_01@CurrentMeasurement

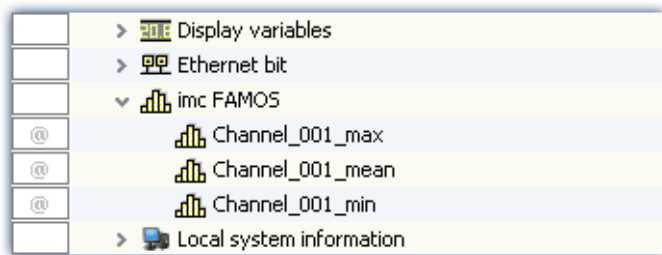
Receiving the sequence's results ("From imc FAMOS")



The variables calculated by imc FAMOS are transferred to imc STUDIO along with the transfer table. On this occasion they can also be renamed.

Note: A channel can not be transferred back to a "User-defined Variable" of the type "Numeric" (single value). For this purpose, please modify the type in imc FAMOS if you wish to obtain a single value. If you require a channel, use as the target a variable created using imc FAMOS. This will then always be assigned the appropriate data type.

The results appear in the Data Browser:



Storage location without saving: Selection of the target (measurement)

The result appears in the Data Browser, but is not saved.

Storage location	Description
Current measurement	If the target variable already exists (e.g.: a user-defined variable), the result is copied to the existing variable, provided that the variable's type is correct. If the target variable does not yet exist, the result is created under the category "imc FAMOS" and is assigned the scope (Validity range): "Temporary". The variable's type is selected on the basis of the content.
Last completed measurement	The memory location is ignored. The result is saved under "Current Measurement" and treated the exact same way as for the selection "Current Measurement"
Measurement#<No>	

Storage location with saving: Selection of the target (measurement)

The result appears in the Data Browser and is stored as a.dat-file.

Speicherort	Beschreibung
Last concluded measurement	(recommended) If the option Save is selected, use Last concluded measurement or Measurement#<No> . The result is saved to the folder with the last concluded measurement respectively to the folder of measurement with selected number #<No>. After loading the measurement, the result is always available. Please note that at least one channel must be saved on PC.
Measurement#<No>	Otherwise the measurement folder does not exist. If in Data Browser the selected measurement number is assigned to "Current measurement" , the system acts as if " <i>Current Measurement</i> " is selected.
Current measurement	(not recommended) If Current measurement is selected as target, no saved measurement is selected. A subfolder reflecting the current time stamp (e.g. " <i>imcFAMOSResult_2014-07-31 12-50-43</i> ") is set up in the experiment folder for the results.

"Options"

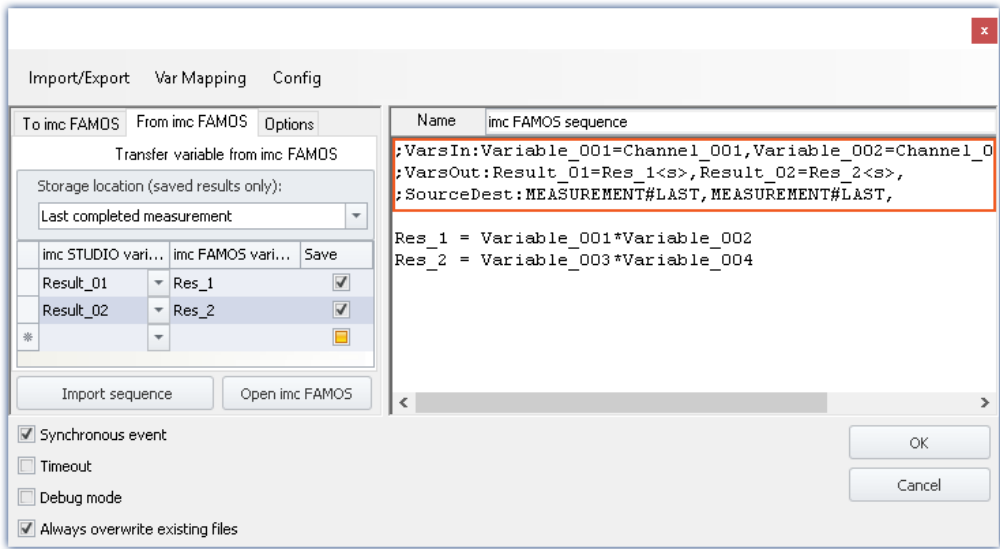
On this page, you can define fixed parameters which are used in the sequence.

Passing (assignment in the Transfer table) of variables can be incorporated in the sequence

Passing (assignment in the Transfer table) of variables can be incorporated in the sequence. Once the assignment is in a sequence, it can easily be passed (copied) to other imc FAMOS-commands.

The assignment is expected/entered as a "Header" in the first three lines of the sequence.

By means of the menu, the assignment can be passed from the transfer table into the sequence, or conversely from the sequence to the table.



Below, the header is set up (with sample names as per the picture):

 Example

Variables: To imc FAMOS:

```
;VarsIn:Variable_001=Channel_001,Variable_002=Channel_002,Variable_003=Channel_003,
```

Variables: From imc FAMOS:

```
;VarsOut:Result_01=Res_1<s>,Result_02=Res_2<s>,
```

Source and data storage location:

```
;SourceDest:MEASUREMENT#LAST,MEASUREMENT#LAST,
```

Variables

Description	Variables: To imc FAMOS:	Variables: From imc FAMOS:
Start	;VarsIn:	;VarsOut:
First variable name	Name in imc FAMOS	Name in imc STUDIO
Assignment character	=	=
Second variable name	Name in imc STUDIO	Name in imc FAMOS
Activation of data storage (optional)		<s>
Separator from next assignment	,	,

Source and data storage location:

Description	Syntax
Start	;SourceDest:
First name	Source of the page "To imc FAMOS"
Second name	Storage location of the page "From imc FAMOS"
Separator	,

Possible syntax:

Source or data storage location	Syntax
Last concluded measurement	MEASUREMENT#LAST
Measurement number 3	Measurement#3
Current measurement	empty, so only ", "
Fixed measurement name (as in the Data Browser)	2017-02-08 16-42-41 (1)

**Example**

Examples	Description
;SourceDest:MEASUREMENT#LAST,Measurement#1,	Source: Last concluded measurement Data storage location: Measurement number 1
;SourceDest:MEASUREMENT#LAST,2017-02-08 16-42-41 (1),	Source: Last concluded measurement Data storage location: Measurement with the name 2017-02-08 16-42-41 (1)
;SourceDest:;,MEASUREMENT#LAST,	Source: Current measurement Data storage location: Last concluded measurement

9.4.2 Information and tips

Parallel processing of sequences

Only one sequence can ever be processed at the same time. If multiple sequences are started in parallel, they are processed one at a time in succession. This can cause "congestion" if there are very many requests.

The data for the sequence (which are valid at its starting time) are saved temporarily one single time and are available at the start of the analysis.

Closing imc STUDIO Monitor

Upon closing imc STUDIO Monitor, the sequence currently running is not interrupted. However, any return values will not be processed. Any additional sequences in the queue for processing are canceled. A corresponding notification is posted and you have a choice whether to cancel closing the program.

Breakpoints and Bookmarks

If the imc FAMOS-sequence contains any defined breakpoints or bookmarks, these are stated along with the sequence. When you close imc FAMOS, comments on the bookmarks/breakpoints appear in the imc FAMOS Sequence-Editor (on the right-hand side).

The following is an example

```
;@# imc FAMOS Sequence
;@# VERSION:      V4.0
;@# DESCRIPTION:
;@# BREAKPOINTS: 20 94
;@# BOOKMARKS:   87 0 0 112 0 0 0 0 0 0
```

Here, breakpoints have been defined for lines 20 and 94, and bookmarks for lines 87 (Bookmark 0) and 112 (Bookmark 3). Upon opening imc FAMOS, comment lines are recognized and removed. Additionally, the corresponding lines are supplied with bookmarks and breakpoints accordingly.

**Note****Error source**

When changes are not made in imc FAMOS, but are instead made directly via the imc FAMOS Sequence-Editor (on the right-hand side), the line numbers do not automatically change accordingly. Even if, for instance, the [transfer of variables](#)⁴⁸² (assignment in the transfer table) is saved with the sequence.

In this case, the data points need to be corrected manually.

To the extent possible, only make any changes using imc FAMOS whenever using any breakpoints/bookmarks in the sequence.

9.5 Data Processing (general)

imc STUDIO DataProcessing is a imc STUDIO plug-in by means of which it is possible during a running measurement to outsource various data processing and computational tasks, which previously were performed by the measurement device, to the PC. This means that the measurement device's computational resources are freed up, and the high computing power of the PC is exploited.

A number of functions packages for Data Processing are available:

Functions package	Description
imc Inline FAMOS  245	Processing and analysis of measured data during running measurement
Inline Analysis  451	Analysis of measured data during running measurement. It is possible to compute a derivative or calculated channel from a input channel. For this purpose, a variety of functions packages are available.
imc STUDIO BusDecoder  487	Expansion package for decoding Fieldbus log channels
Powertrain Monitoring  492	The component imc STUDIO Powertrain Monitoring is developed for the diagnosis of powertrains. The powertrains can consist of motors, shift gearboxes and engines as well as devices for braking. The diagnosis can be used in field scenarios, test plants or end of line tests in a production.

Opening Data Processing

The plug-in can be opened via the Navigation Pane: .

Tasks

For each activated functions package, a **Task**, represented by a card tab, is opened. In consequence, you can jump from one functions package to another. All Tasks are processed in parallel. How many tasks can be performed depends on the demands made by the functions used and on the PC's capacities.

For each functions package, you can set up additional tasks. Here again, the amount possible depends on the capacities of the PC used.

Configuring results channels in the Setup

For quick and clearly organized configuration of the results channels belonging to Data Processing-tasks, all results channels appear in the channel table in the main window: Setup. Here, the channels can be configured like the virtual channels belonging to imc Online FAMOS

Note

Wherever the documentation refers to a device, the statements also apply analogously for multiple devices.

9.5.1 Ribbon - Data Processing

Task

Menu item	Description
 New	Adds a new Task. Select from which Functions Package the new Task is to be generated and enter an appropriate name for the task.
 Edit	Opens a dialog in which, for example, the selected task's name can be changed.
 Delete	Deletes the selected Task. The configuration is discarded and can no longer be restored.

Configuration

Menu item	Description
 Restore	Restores the most recently applied state of the selected task's configuration. The current configuration is discarded and can no longer be restored.
 Check	Checks the configuration of the selected Task. If an error occurs, a notification will appear in the status bar! Special feature of imc Inline FAMOS: The faulty portion will appear highlighted in the Editor. If the syntax-check concludes without an error, all newly defined virtual channels and variables will be adopted in the Channels list.
 Apply	Applies the configuration of the selected Task. As soon as the configuration has been applied , the Task starts and the calculations are performed. In the table of channels in the Setup, the newly defined channels appear as additional virtual channels .

Clipboard

Menu item	Description
 Cut	Removes the selection (the highlighted portion in the Editor) to the Clipboard.
 Copy	Copies the selection (the highlighted portion in the Editor) to the Clipboard.
 Paste	Inserts the Clipboard contents at the position highlighted.

Edit

Menu item	Description
 Undo	Reverses the effect of the last change (in the Editor).
 Redo	Restores the previously reversed change.

Search

Menu item	Description
 Find	Find specific text
 Find and Replace	Find and replace specific text

Im-/Export

Menu item	Description
 Import	Imports Data Processing Task configurations.
 Export	Exports the configuration of the selected Data Processing Task.

9.6 Bus Decoder - Decoding fieldbus log channels

There is a limit on the count of active fieldbus channels decoded in the device. You have the option of outsourcing the decoding of individual channels from the device to the PC. There are two methods of doing this:

- using imc FAMOS (post-processing - after the measurement), and
- using the imc STUDIO component: Bus Decoder (during the measurement)

The decoding is performed on the fieldbus log channel. By means of the Fieldbus-Assistant, you can specify whether the particular channel is to be decoded either in the device or via the Bus Decoder, for example.

- By means of the Bus Decoder, you can select which among the selected fieldbus channels to decode from the log channel during the measurement.
- When using imc FAMOS, all fieldbus channels are always decoded (not only specifically highlighted ones).

Bus Decoder

The decoding is performed on the basis of the data streams of the measurement currently running on the PC. This utilizes the PC's computation resources.

The decoding information which is usually located in separate configuration files (e.g. with CAN in *.dcb) is instead embedded in the log channel. Thus, the log channel contains all information necessary for decoding.

The following functions are available:

- Decoding of either all or individual channels from a log-channel
- Resampling of the channels
- Saving of the result channels

The results generated can be displayed on Panel pages and saved with the associated measurement data. Subsequent processing by means of imc Inline FAMOS is also possible.



Note

Notes on activating

Bus Decoder is a package of functions for Data Processing.

Activate the Bus Decoder and the "imc STUDIO DataProcessing Editor" using the [Product Configurator](#)²⁷, so that it will appear in the software. You will find the Bus Decoder in the group: "imc STUDIO DataProcessing".

Supported fieldbus types (Bus Decoder)

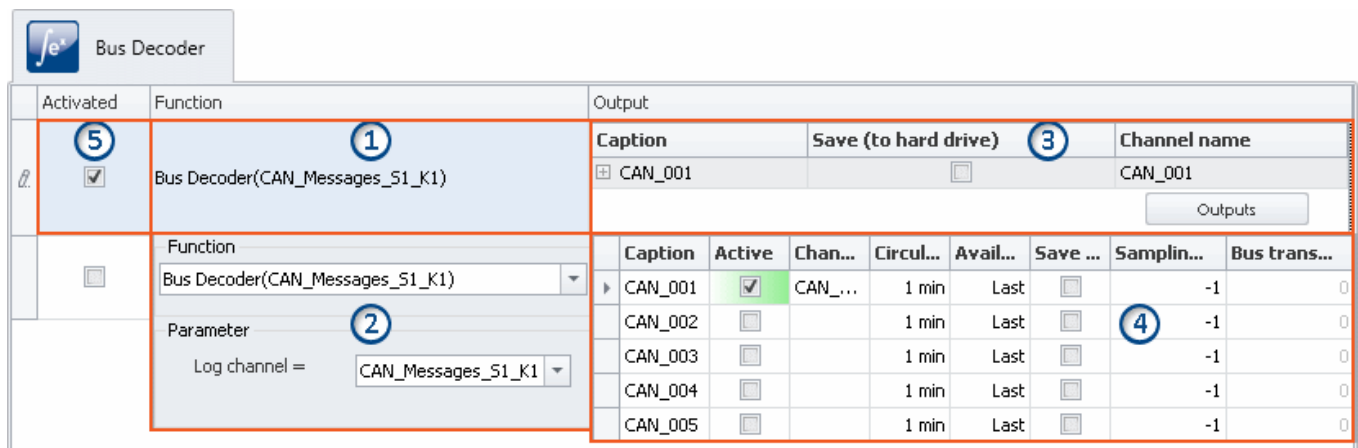
Decoding of the log channels is possible for multiple fieldbusses. Some fieldbusses need to be configured separately for this purpose.

Fieldbus type	FW group		Note
	A	B	
CAN	•	---	Per-channel adoption in the blob. Channel settings: See in the imc STUDIO manual under " <i>Blob - Decoding with the Bus Decoder</i> " Logging of the data must be activated.
SPI	•	---	All channels are adopted in the blob. No selection is possible.
MVB	•	---	All channels are adopted in the blob. No selection is possible. Logging of the data must be activated.
FlexRay	•	---	Per-channel adoption in the blob. Channel settings: See in the imc STUDIO manual under " <i>Properties of Signals</i> " Logging of the data must be activated.

9.6.1 Opening Bus Decoder

Open the plug-in Data Processing via the Navigation pane. In the plug-in's main window, there will be a tab (Task: ) for the functions package Bus Decoder. If it is not selected already, left-click the mouse on the Task name. The Bus Decoder Editor will appear in the main window.

9.6.2 User interface



The window is subdivided into five regions:

1. Function: Lists any functions whose parameters have already been set (decoding of a log channel)
2. Function: Selection of the log channel
3. Output: Displays the activated outputs (the decoded channels)
4. Output: Activate and parameterize the outputs here (the decoded channels)
5. Activate/deactivate functions

Region 1: Display of parameterized functions

Until the settings have been made for any function, this line appears empty. Once the settings for any functions have been made, these functions are displayed here.

Decoding a log channel (setting up a function)

In order to decode a desired log channel, click in the column **"Function"** in the first empty row. In consequence, Region 2 opens. There, select the log channel.

Changing a function

To alter a function which has already been set up, click on it corresponding cell.

Deleting a function

To delete a function which has already been set up, open the context menu in the corresponding cell and select *"Delete function"*.

Region 2: Selecting the log channel

Here you select the **"Log channel"**.

In the bottom input box having the arrow button (▼), select the desired log channel.

Function

Bus Decoder(CAN_Messages_S1_K1) ▼

Parameter

Log channel = CAN_Messages_S1_K1 ▼

When you click the mouse anywhere outside of the dialog, the settings are concluded.

Region 3: Displaying the activated outputs (the decoded channels)

Under **"Output"**, all channels belonging to the log channel, for which decoding is configured in the Bus Decoder, are listed. Each output quantity generates a channel (virtual channel), which is available in the Setup and in the Panel and can be linked with a Widget.

Activating (decoding) and making settings for outputs

In order to activate and parameterize outputs (channels), click in the column **"Output"** on the button **"Outputs"**. Immediately, the parameterizing window (Region 4) opens. There, select and configure the channels.

	Activated	Function	Output		
			Caption	Save (to hard drive)	Channel name
	☒	Bus Decoder(CAN_Messages_S1_K1)	CAN_001	☐	CAN_001
			Outputs		

For each function, the selected channels are set separately, for which reason there is an *"Outputs"* button for each function. For each function, any number of channels can be activated. Here, too, as with the functions, the amount is limited only by the performance power of your PC.

Region 4: Window for parameterizing and activating the channels

Here, you can activate and parameterize the channels. The channels available are listed in a table.

	Caption	Active	Chan...	Circul...	Avail...	Save ...	Samplin...	Bus trans...
▶	CAN_001	☒	CAN_...	1 min	Last	☐	-1	0
	CAN_002	☐		1 min	Last	☐	-1	0
	CAN_003	☐		1 min	Last	☐	-1	0
	CAN_004	☐		1 min	Last	☐	-1	0
	CAN_005	☐		1 min	Last	☐	-1	0

Activating channels

Put a check mark in the column "**Active**" for the channel desired.

Checking this box activates the decoding for the channel. The channel name is set automatically, but can be edited as desired subsequently. Activated channels appear as "**Virtual channels**" in the Channels table (Setup) and in the Data Browser (Panel).

Parameterizing channels

Each channel has parameters which can be adapted to the respective requirements.

E.g., storage of channels on the PC can be activated. To do this, put a checkmark in the box for the column "*Save (HD)*". If necessary, adjust the other settings as well, by editing the inputs.

In the channel table (Setup), the outputs defined appear as "**Virtual channels**". Here, the channel can be parameterized like any other virtual channels (e.g. the data storage). The changes made in the Setup have a direct effect on the Bus Decoder settings, since the same settings, in part, can be made here.

For more information on the individual parameters, see the section: "[Output Parameters](#)" .

Region 5: Activating/deactivating functions

Here, the function can be activated/deactivated.

Only such log channels are decoded which have a checkmark in the column "**Activated**". All newly added functions are activated by default.

9.6.3 Output parameters

Every output can be parameterized. The parameters affect the results returned. The table contains the following columns, in which settings can be made:

- Parameters denoted by ● can also be set in the channel table in the Setup.

Parameter	Description
- Caption	Name of the output quantity (permanently assigned)
	Activates/deactivates decoding of the channel
- Active	● Deactivated: the channel is not decoded ● Activated: the channel is decoded
- Channel name	Name of the virtual channel on which the output quantity is to be outputted. This name appears in the channel table (Setup) and in the Data Browser (Panel).

Parameter	Description
• Circular buffer duration (curve window)	For the time period specified, the data are retained on the PC for the purpose of display in the Panel. The Widgets (e.g. the curve window) only contain data for the duration of the buffer period set by the user.
• Available events (curve window)	With this setting, you determine whether all events triggered in the Panel are to be available, or only the last one. If "<i>Circular buffer duration (curve window)</i>" is set, then only "<i>Last</i>" is available. For a detailed explanation, see "Storage options"¹⁹⁷.
• Save (to hard drive)	Saves the output quantities to the PC
- Averaging time	>0 Sampling of the decoded channel at a defined sampling time -1 The original channel's setting (Feldbus-Assistenten) is applied. 0 Recorded with a time stamp
Bus transmission cycle [s]	Defined by the Assistant >0 Sampling interval/transfer cycle of the channel 0 Recorded with a time stamp

9.7 Powertrain Monitoring - Machine and bearing diagnostics

General

The component imc STUDIO Powertrain Monitoring has been developed in cooperation with the company **GfM (Gesellschaft für Maschinendiagnose mbH)** for power train diagnostics.

It is dedicated to diagnosis of powertrains. The powertrains can consist of motors, shift gearboxes and engines as well as devices for braking. The diagnosis can be used in field scenarios, test rigs or end of line tests in production.

Powertrain Monitoring offers two different kinds of diagnosis of vibrations: a [Base diagnosis](#)^[496] on the basis of characteristic values and a [Advanced Diagnosis](#)^[503] on the basis of a frequency selective search of kinematic pattern. A configuration for a specific gear type will be created via the **Powertrain Monitoring Assistant**. This configuration will be saved in the project of imc STUDIO and can be distributed via this project on different test locations. The configuration is used by the **Editor** by applying configured inputs to physical measurement channels. This system allows to use the same configuration on different measurement systems if multiple test locations are driven with the same powertrain type.



Note

Notes on activating

Powertrain Monitoring is a package of functions for Data Processing.

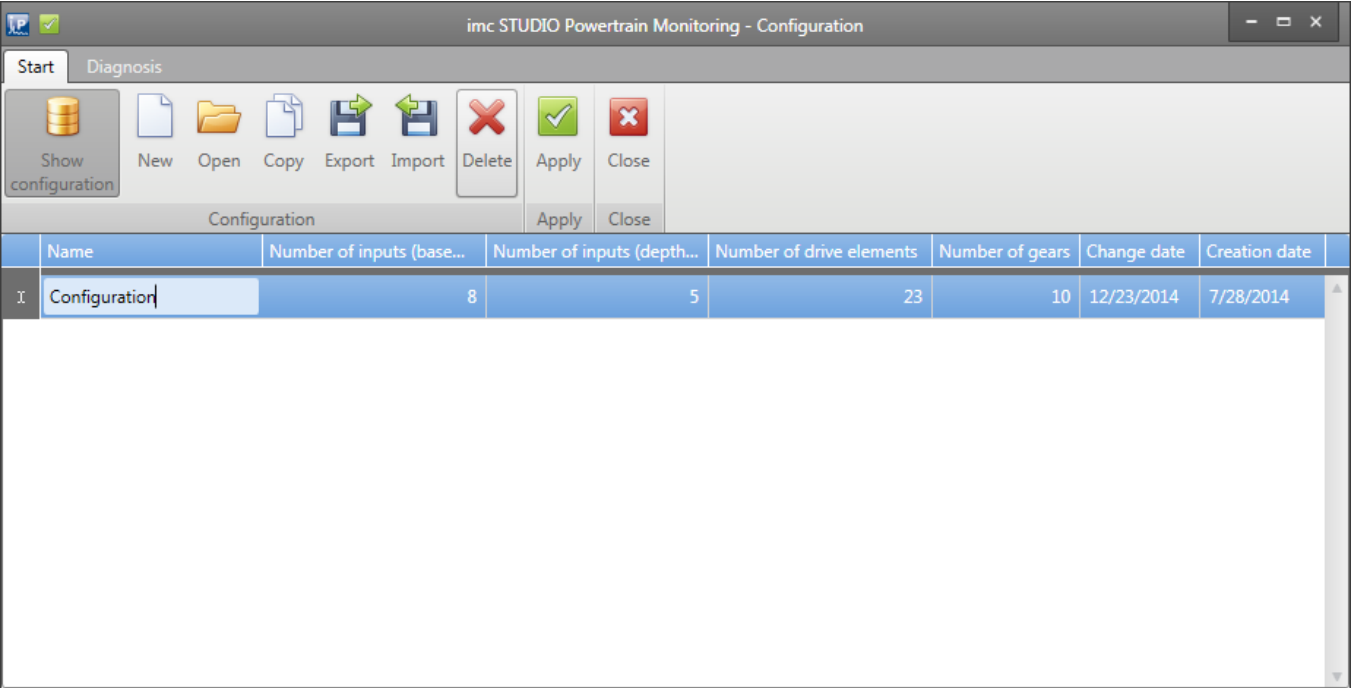
Activate the Powertrain Monitoring and the "imc STUDIO DataProcessing Editor" using the [Product Configurator](#)^[27], so that it will appear in the software. You will find the Powertrain Monitoring in the group: "imc STUDIO DataProcessing".

Licensing

For Powertrain Monitoring, an extra license is required.

9.7.1 Assistant

9.7.1.1 Configuration

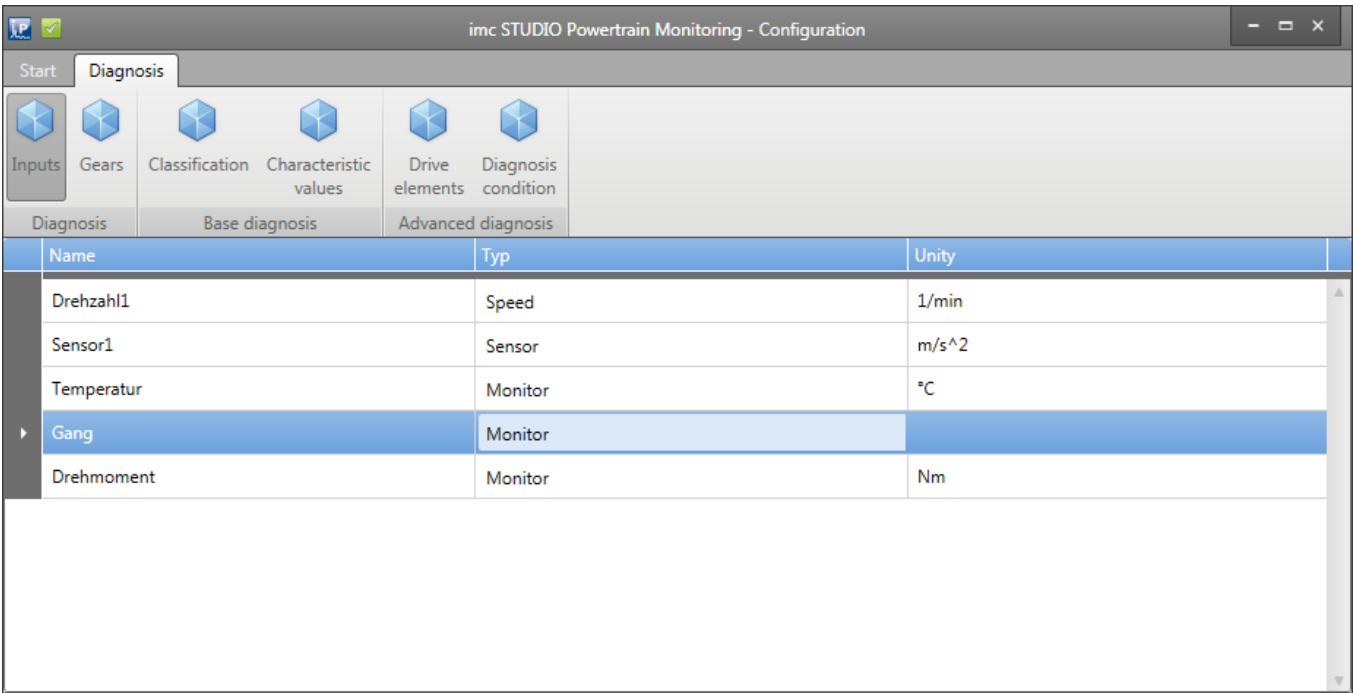


Picture 1: Manage Configurations

Via the menu point show configurations, all configurations that are saved in the active imc STUDIO project can be managed. That means creating new, open, copy, export, import and delete configurations.

One configuration can be applied to the active project by the menu point apply. **To have this persistent the imc STUDIO project have to be saved.**

9.7.1.2 Inputs



Picture 2: Assistant Inputs

An essential part of both diagnosis procedures are the measurement of structure-borne sound of the test item via accelerometers. These will be added as sensors in the assistant.

Particularly for the advanced analysis the accurate measurement of the speed is required. This speed input will be assigned to sensors as reference speed.

Furthermore up to 12 process information can be measured with the Powertrain Monitoring. These serve for the classification of the operational state in the base diagnosis and thus a description of steady states of the system. In the advanced diagnosis the process information can be used to limit the analysis to certain operational states.

In the menu point inputs, the accelerometer, speed and process inputs can be configured. For a speed input the maximum occurring speed must be set in the properties. For a sensor the reference speed must be chosen from the already configured speed channels. For the [advanced analysis](#)⁵⁰³ the measurement time for this sensor must also be set.

9.7.1.3 Gears

imc STUDIO Powertrain Monitoring - Configuration

Start Diagnosis

Inputs Gears Classification Characteristic values Drive elements Diagnosis condition

Diagnosis Base diagnosis Advanced diagnosis

Number of gears: 7

		G1	G2	G3	G4	G5	G6	R1
Kupplung A	(Coupled)	Yes	Yes	Yes	Yes	Yes	No	No
PG 1	(Annulus)	Decoupled	Coupled	Coupled	Coupled	Decoupled	Decoupled	Coupled
PG 1	(Sun)	Coupled	Coupled	Coupled	Fixed	Decoupled	Decoupled	Coupled
PG 2	(Annulus)	Fixed	Coupled	Coupled	Coupled	Fixed	Fixed	Fixed
PG 2	(Sun)	Coupled	Decoupled	Decoupled	Decoupled	Coupled	Decoupled	Coupled
PG 3	(Annulus)	Coupled	Coupled	Coupled	Coupled	Coupled	Fixed	Coupled
PG 3	(Sun)	Fixed	Fixed	Coupled	Coupled	Coupled	Coupled	Fixed
PG 4	(Sun)	Fixed	Fixed	Coupled	Coupled	Coupled	Coupled	Fixed
Kupplung B	(Coupled)	No	Yes	Yes	No	No	No	Yes
Kupplung E	(Coupled)	No	No	Yes	Yes	Yes	Yes	No

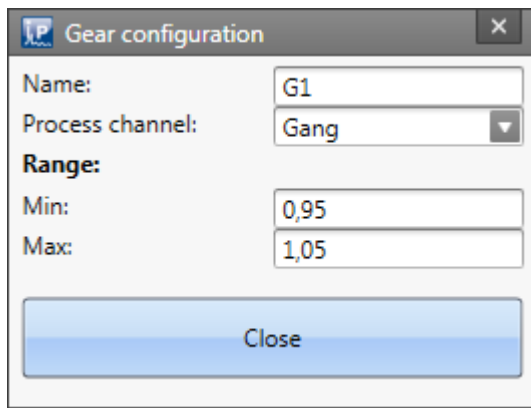
	G1	G2	G3	G4	G5	G6	R1
Motorwelle	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000
Zwischenwelle 1	1,0000	1,0000	1,0000	1,0000	1,0000	0,0000	-1,3182
Zwischenwelle 2	-0,7586	1,0000	1,0000	0,0000	-0,7586	0,0000	1,0000
Zwischenwelle 3	0,4052	1,0000	1,0000	0,6618	0,4052	0,0000	-0,5341
festeWelle	0,0000	1,0000	1,0000	0,4314	0,0000	0,0000	0,0000
Zwischenwelle 4	0,0000	0,0000	1,0000	1,8859	2,5579	3,6190	0,0000
Zwischenwelle 5	0,2932	0,7237	1,0000	1,0000	1,0000	1,0000	-0,3865
Zwischenwelle 6	0,2122	0,5237	1,0000	1,2448	1,4305	1,7237	-0,2797
Zwischenwelle 7	-1,0968	-1,0968	-1,0968	-1,0968	-1,0968	-1,0968	-1,0968
Zwischenwelle 8	1,0968	1,0968	1,0968	1,0968	1,0968	1,0968	1,0968
Zwischenwelle 9	-0,1782	-0,4399	-0,8400	-1,0456	-1,2016	-1,4479	0,2350
Ausgangswelle	0,0653	0,1611	0,3076	0,3829	0,4400	0,5302	-0,0860

Picture 3: Assistant Gears

A special process information, always required by Powertrain Monitoring, is the gear information. This process information is also used in the classification like all other process information. It also serves as output of status information for the base diagnosis. Thus a status information is written whether a certain characteristic value is exceeded in a certain gear. This allows to control the test procedure for example skipping a certain gear.

For the advanced analysis the gear information serves as information for calculating the transmission of the individual drive elements correctly and therefore allows a frequency selective search of kinematic patterns.

The gears can be configured in the menu point gears (see Picture 3).



Picture 4: Gear configuration

To identify the current gear in the analysis, a unique range of one process information must be defined. The same process input must be used for all gears. This can be configured by clicking the button besides the gear name (see Picture 4).

9.7.2 Editor

To apply the configuration built with the assistant to the current experiment, the Powertrain Monitoring [Editor](#)⁵⁰⁰ has to be used.

One of the configurations that are saved in the current project must be chosen in the **configuration selection box**. The executing function should be set depending on the chosen configuration to base or advanced diagnosis. Thus all sensor, speed and process inputs are read from the configuration and are provided as inputs. These must be assigned to measurement channels that are available in the current experiment.

The measurement channels must meet the following requirements:

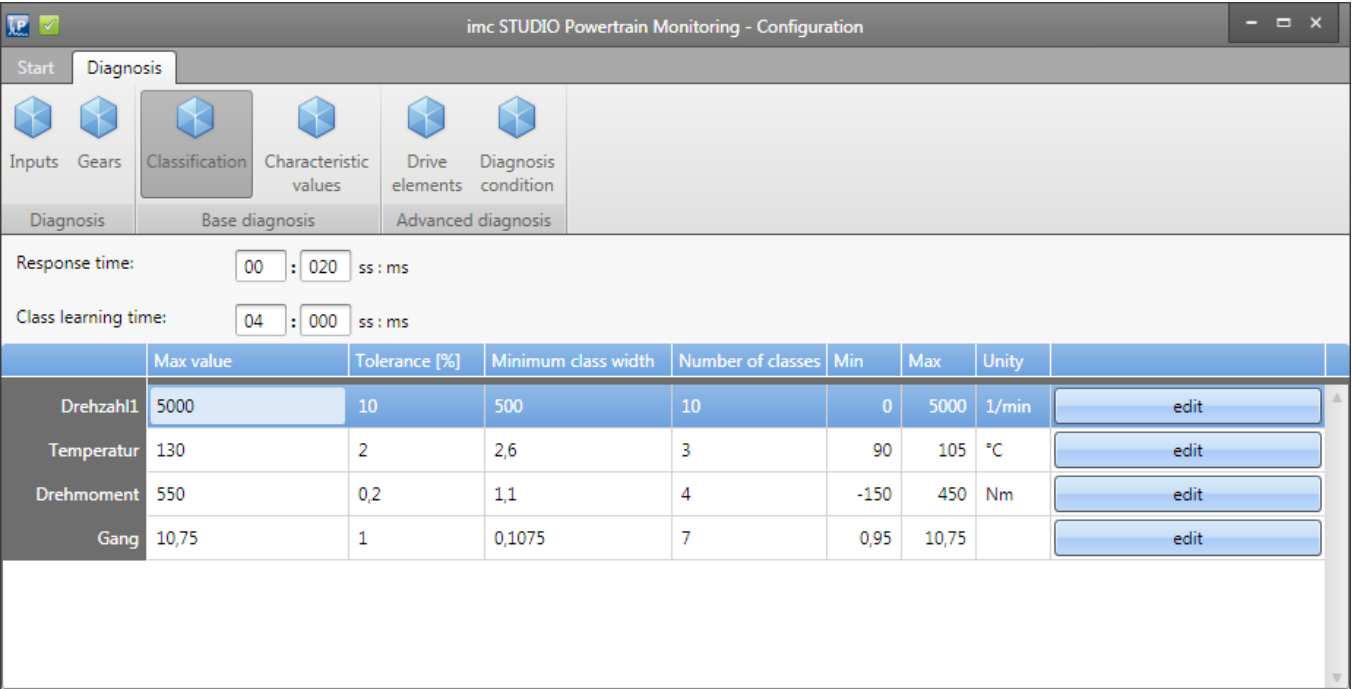
- Accelerometers and speed channels must be measured with 50 kHz.
- Sampling frequencies of process channels must be an integer divider of 50 kHz (e.g. 100 Hz, 1 kHz, ...)
- All channels must be set to undefined measurement time.

9.7.3 Base diagnosis

The base diagnosis consists of a characteristic value calculation with self-learning thresholds. For the learning phase a classification is used where an individual warn- and alarm threshold is learned for each class. The [classification](#)⁴⁹⁷ can be configured with the Powertrain Monitoring Assistant.

9.7.3.1 Assistant

9.7.3.1.1 Classification



Picture 5: Assistant Classification

The following general settings must be configured in the menu point **Classification**:

Settings	Description
Reaction Time	If a certain class is active continuously over this time it counts as stable. The learning time for unlearned classes starts counting from this time point. For already learned classes, the learned warn- and alarm threshold will be used from this time point. The user defined threshold will be used between the actual class change and the reaching of the reaction time.
Class Learning Time	A class will be considered as learned if the system was in this state over the defined class learning time. The class learning time can also be reached by multiple time parts that sum up to the class learning time. During the learning phase the maximum occurred characteristic values will be saved. At the end of the learning phase warn- and alarm thresholds are calculated by multiplying the occurred maximum with a percent factor.

Class details of the channel: Drehzahl1

	Min	Max	Unity	Ignore	
Class 1	0,00	1.000,00	1/min	☐	
Class 2	1.000,00	1.300,00	1/min	☐	
Class 3	1.300,00	1.600,00	1/min	☐	
Class 4	1.600,00	2.000,00	1/min	☐	
Class 5	2.000,00	2.500,00	1/min	☐	
Class 6	2.500,00	3.000,00	1/min	☐	
Class 7	3.000,00	3.500,00	1/min	☐	
Class 8	3.500,00	4.000,00	1/min	☐	
Class 9	4.000,00	4.500,00	1/min	☐	
Class 10	4.500,00	5.000,00	1/min	☐	

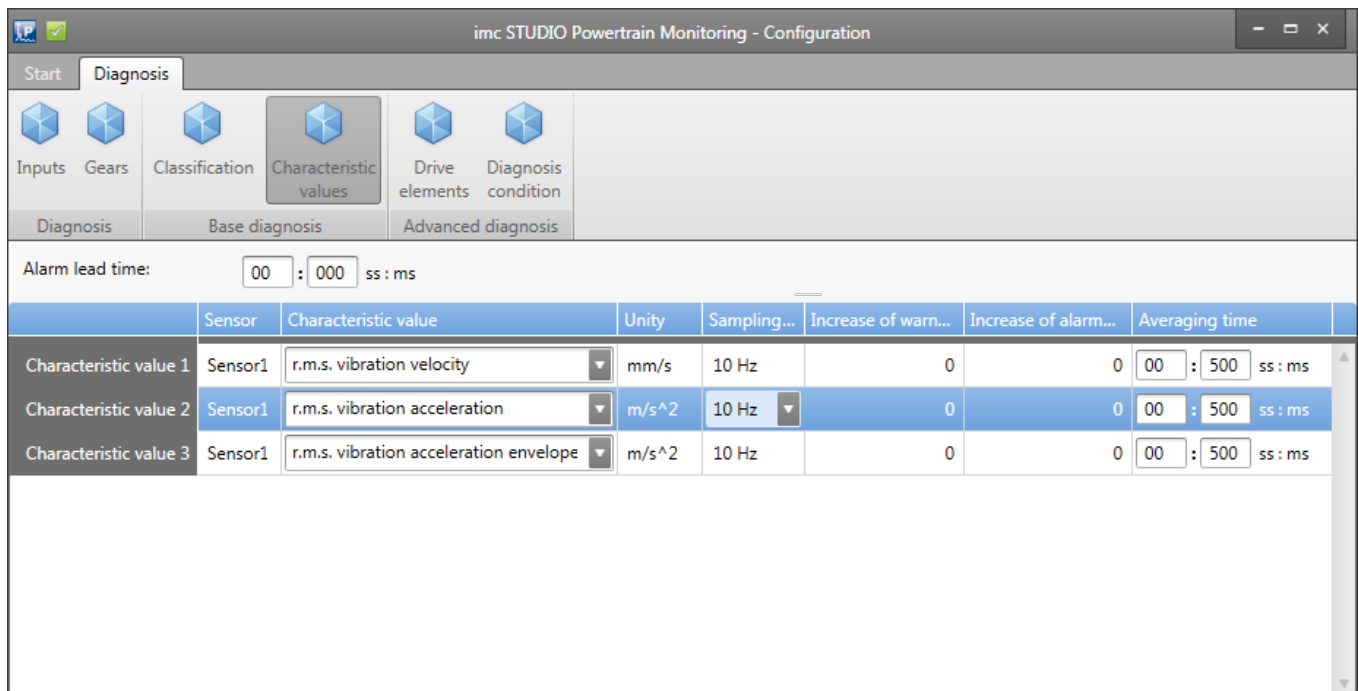
Schließen

Picture 6: Assistant Classification Details

For the classification each process information is used that were defined (see chapter "[Configuration](#)⁴⁹³"). Each process information can be divided in up to 1000 classes. Thus the class borders can also be learned automatically by only configuring the class width of each process information. Furthermore the classes can be configured individually by clicking the Edit button (see Picture 6).

An exception is the **gear** information. This process information can be configured as described in chapter "[Gears](#)⁴⁹⁵" and is not modifiable in this section.

9.7.3.1.2 Characteristic values



Picture 7: Assistant Characteristic Values

The characteristic values can be configured in the corresponding menu point (see Picture 7).

The following characteristic values are available for the base diagnosis for each configured accelerometer:

- R.M.S. of vibration velocity based on DIN-ISO 10816-3 in the frequency range 10 Hz - 1 kHz
- R.M.S. of vibration acceleration in the frequency range 0 Hz - 25 kHz
- R.M.S. of vibration acceleration of the envelope in the frequency range 0 Hz - 25 kHz

The characteristic value will be calculated from the measured signal of the accelerometer with the given average time. A prerequisite is a configured Anti-Aliasing filter for the sampling frequency of 50 kHz for the accelerometers in imc STUDIO.

The configured percent increase of warn- and alarm thresholds are multiplied to the maximum characteristic value at the end of the learning phase of the class.

9.7.3.2 Editor

Please define the new variable

Variable

Name:

Type:

Extended

Initial value:

Category:

Unit:

Validity range:

Comment:

☐ Measured data for display, calculations

Available events:

Circular buffer duration:

Trigger name:

☐ Save measured data

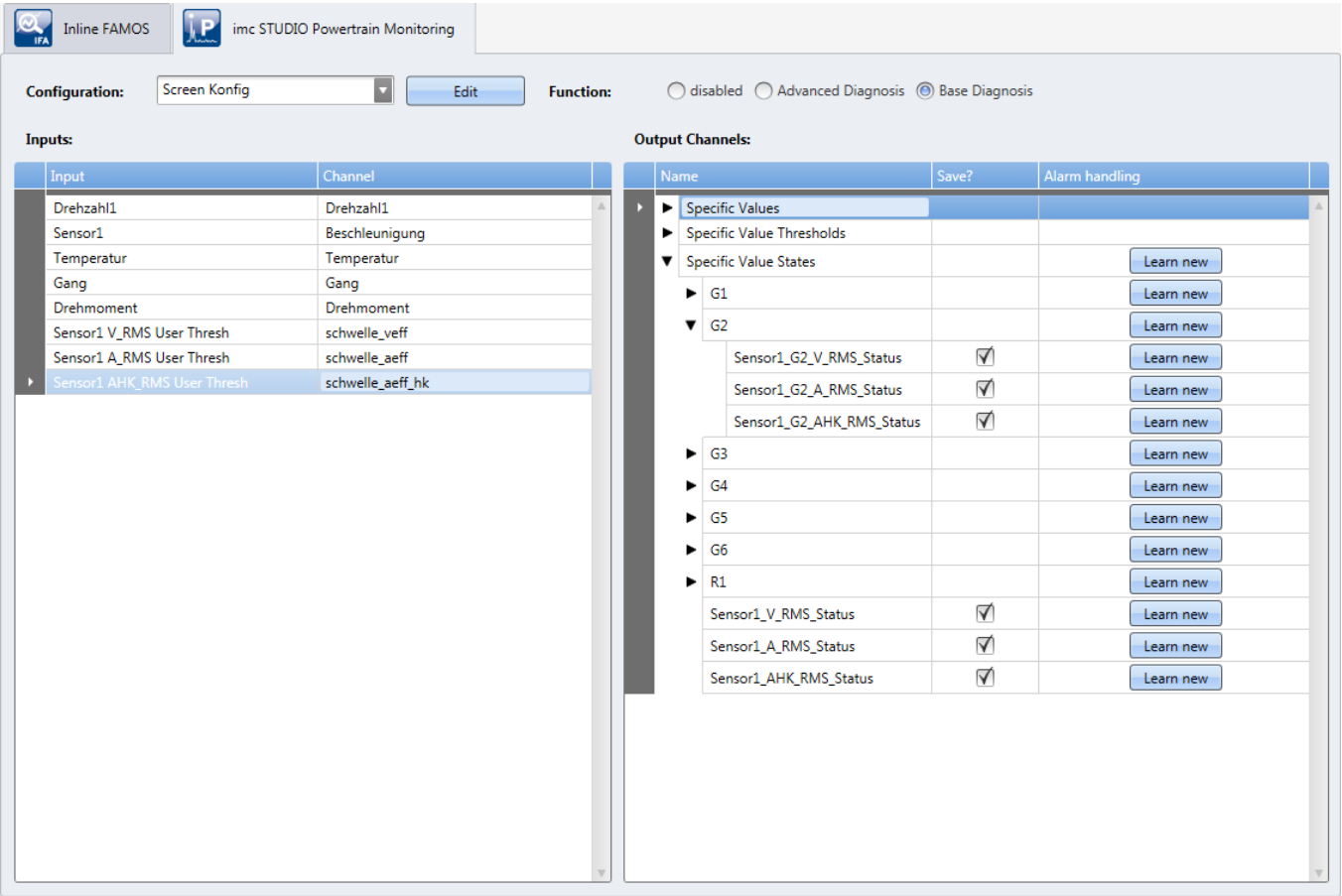
Saved events:

Circular buffer duration:

Apply OK Cancel

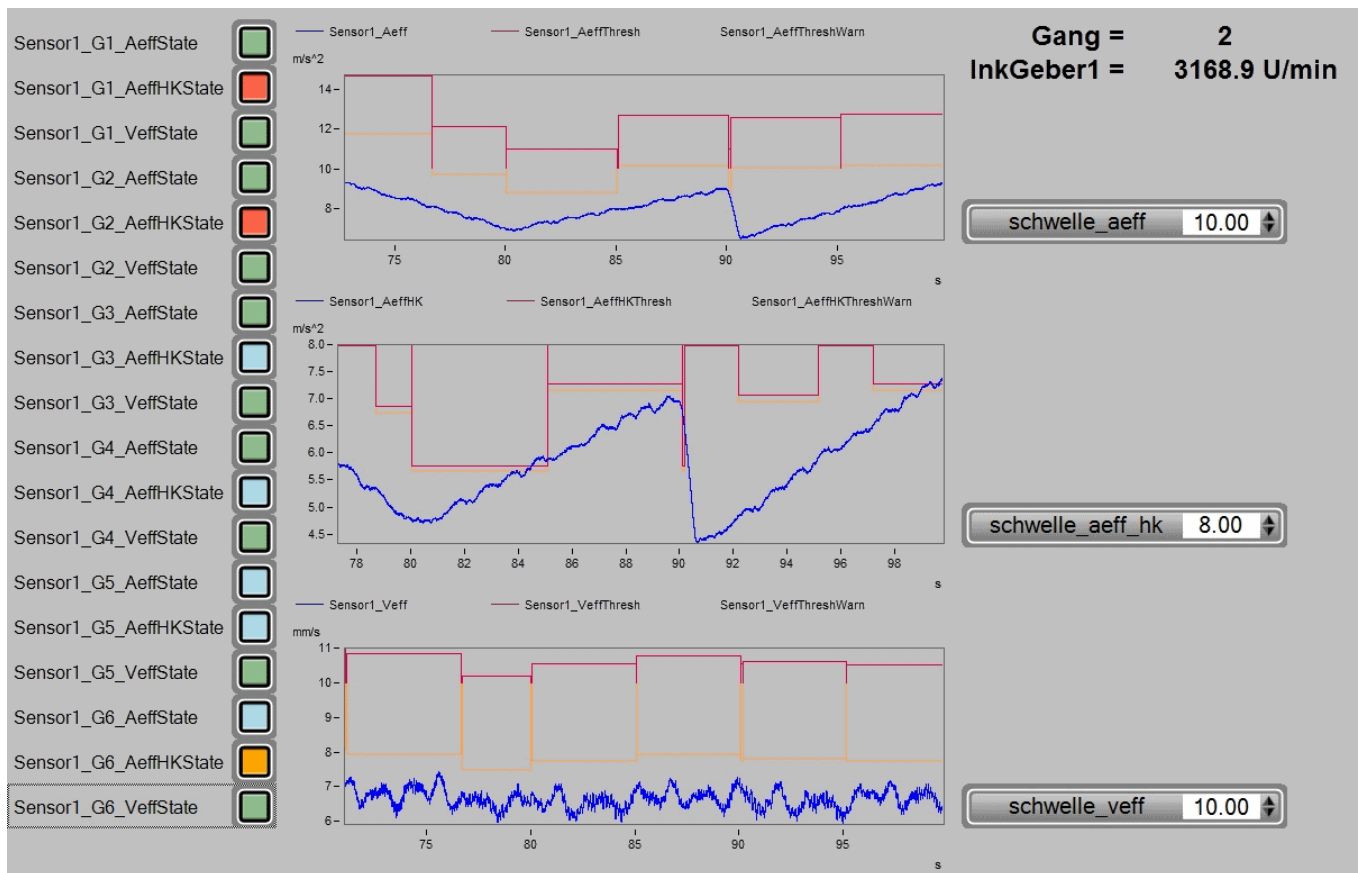
Picture 8: Create User defined variables

The inputs of the configuration must be assigned to the physical measurement channels for the base diagnosis (see also general [description of the editor](#)). Furthermore each characteristic value has a user defined threshold that must be assigned. These are assigned to user defined variables that can be configured in the Data Browser by right clicking in the Data Browser (see Picture 8).



Picture 9: Base Diagnosis Editor

9.7.3.3 Measurement data



Picture 10: Example: Base Diagnosis

For each configured characteristic value a warn- and alarm threshold channel is provided.

If a certain class is learned, the corresponding warn- and alarm threshold is put out to the channel. If the class is not yet learned or the reaction time is not yet elapsed, the user defined threshold will be put out.

Additionally to the characteristic value, warn threshold and alarm threshold, state information are provided.

These signalize an exceedance of a threshold in the following manner:

- No Exceedance of any threshold = 0
- Exceedance of the user defined threshold = 1
- Exceedance of the warn threshold = 2
- Exceedance of the alarm threshold = 3

The state information are written for each characteristic value and for each individual gear.

By using existing visualization tools the state information can be visualized for example as traffic light.

If the user defined threshold is changed during the measurement, the user defined thresholds will be reevaluated and reset if the new threshold does not exceed the characteristic value. The warn- and alarm thresholds are based on learned classes and will not be reset due to a change of the user defined threshold. They must be reinitialized by resetting and relearning the classes. This can be achieved in the Powertrain Monitoring [Editor](#) ⁵⁰⁰ on the right side at the various state channels.

Picture 10 shows a possible way of visualizing the data of the base diagnosis. On the left side the state information are shown for each characteristic value and each gear. The curve windows contain the characteristic values and corresponding warn- and alarm thresholds. The user defined thresholds are placed on the right side.

9.7.3.4 Summary of function

- Calculating the characteristic value of one accelerometer with the configured average time and bandwidth.
- Determining the current class
 - A class counts as stable if it is active the whole reaction time.
 - A class is learned if the learning time is reached. This can also be reached in multiple time parts.
 - At the end of the learning phase the maximum characteristic value during the learning phase is used to calculate the warn- and alarm thresholds.
- If the class is not yet learned or not stable the user defined threshold will be written to the threshold channels.
- If the class is learned and stable the learned warn- and alarm threshold will be written to the corresponding threshold channels.
- Depending on the exceedance of a threshold the state is written to the state channels for the characteristic value.

9.7.4 Advanced diagnosis

The advanced diagnosis of machine monitoring evaluates abnormal states in spectra and envelope spectra. It provides early evidence of evolving damages. The methods are well known and accepted. They are partly described in the VDI 3832. If an abnormal state of a drive element is recognized it is possible to view the damage evolution long before an actual failure. So it is possible to evolve knowledge about approaches of improvement.

Basis for the advanced diagnosis are the order and envelope order spectra. These spectra are calculated from the vibration signal and the speed signal if defined conditions are fulfilled for a certain measurement time. Possible conditions are speed or process ranges to allow an analysis in well-defined states.

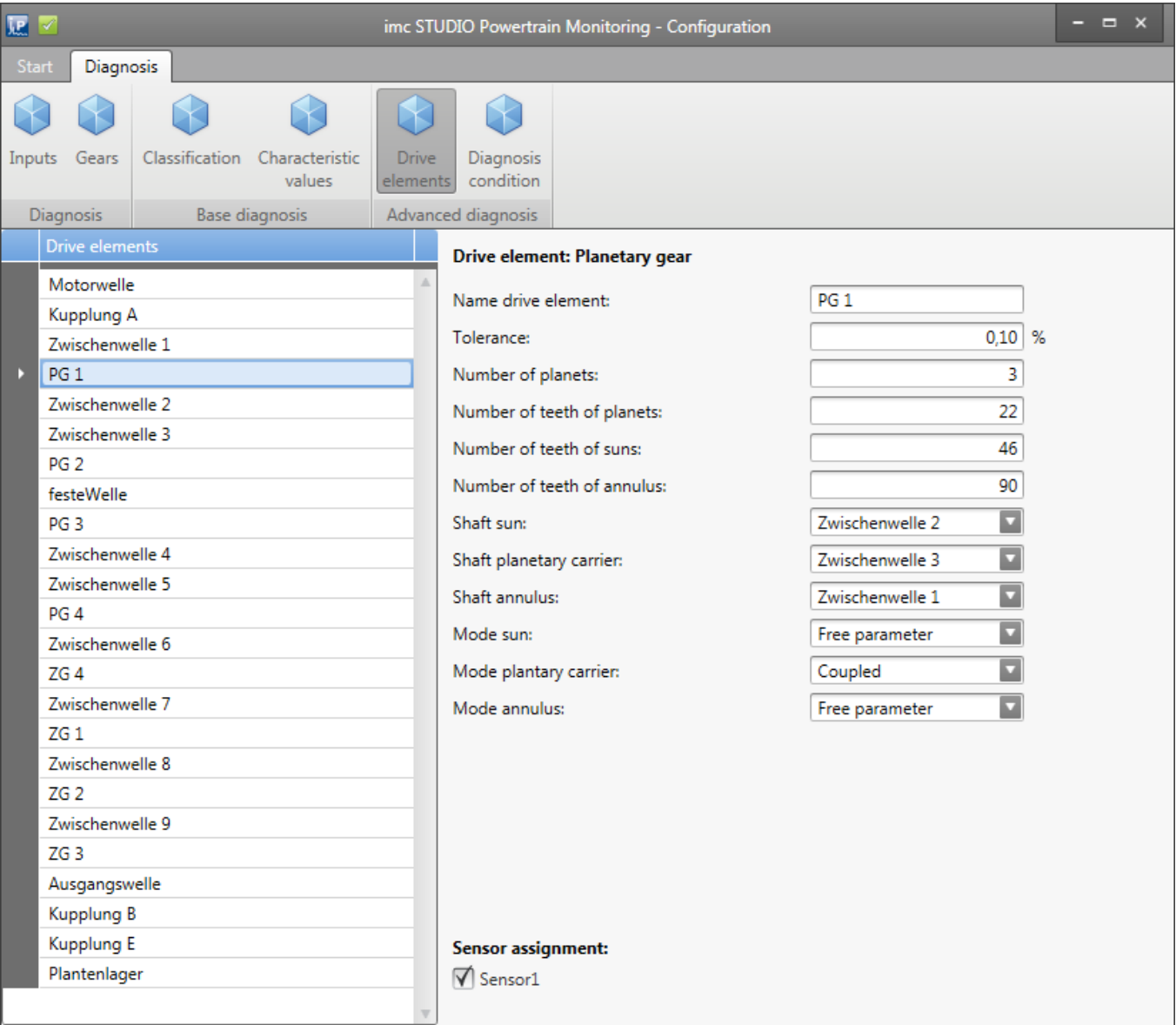
The calculated spectra are then analyzed according to a significance analysis. This is done by first calculating a moving significance threshold that represents the average amplitude over the abscissa. The spectrum is then divided by this threshold and all values below one are set to zero. The remaining peaks are significant and thus with high possibility a consequence of a deterministic mechanic cause, in contrast to random excitations. If this significant spectral components fit to kinematic pattern of a drive element, they are called diagnostic attributes below.

For further diagnosis the spectra can be visualized as single spectrum, color map or waterfall diagram.

9.7.4.1 Assistant

The advanced diagnosis is based on a cyclic evaluation of the measured vibration signal. In order that an analysis can take place, the predefined [diagnosis conditions](#)^[508] must be fulfilled over a configured measurement time. The measurement time is configured in the menu point **inputs**, as described in chapter "[Configuration](#)"^[493]. The longer the measurement time the better the resolution of the calculated spectra. This increases the diagnosis quality. However the maximum order or frequency is lower. This can possibly lead to not examining all kinematic patterns.

9.7.4.1.1 Drive elements



Picture 11: Assistant Drive Elements

The monitoring gear has to be configured in the Powertrain Monitoring Assistant. Thus the complete power train must be represented under the menu point Drive Elements (see Picture 11). Thereby at some drive elements free parameter can be chosen. These free parameters must then be set in the menu point Gears. This allows an individual configuration for each gear. For example a clutch can coupled in one gear and decoupled in another and thus influencing the transmission of the shafts. That a certain drive element can be monitored by the advanced diagnosis, it has to be assigned to one ore more sensors.

The following drive elements can be configured:

Drive elements	Description
Shaft	Exactly one shaft in the system must be configured as reference shaft. This is also the shaft where the speed is measured. It has always the transmission of 1. The transmission of all other shafts in the machine are in relation to this reference shaft. Normally a tolerance of 0.1 % is suitable. In the analysis it is searched for loose particles and unbalance on the configured shafts.

Drive elements	Description
Drive Shaft	The drive shaft can be used as in the configuration like a normal shaft. This can also be used as reference shaft for speed measurement. Normally a tolerance of 0.1 % is suitable. In the analysis it is searched for loose particles, unbalance and link damage or misalignment on the configured drive shafts.
Clutch	A clutch can be configured as fixed connection or as shifting clutch between two shafts. Therefore the input and output shaft must be set to one shaft of the already configured shafts. If this clutch is fixed the option coupled has to be set to yes. If the clutch is decoupled due to gear switching the option must be set as free parameter. Normally a tolerance of 0.1 % is suitable. In the analysis it is searched for misalignment on the configured clutches.
Gear Drive	The gear drive describes a spur gear and bevel gear. It must be chosen a shaft for the pinion and wheel respectively. It has to be set whether the pinion and wheel are coupled with the shaft or whether the coupling between pinion and shaft or wheel and shaft is decoupled due to gear switching. Furthermore the number of tooth for the pinion and wheel has to be entered. Normally a tolerance of 0.1 % is suitable. In the analysis it is searched for revolving deviation of flank shapes and local deviation of flank shapes at the pinion and wheel on the configured gear drives.
Belt Gearing	For the belt gearing the shaft of the driven and driving wheel. It has to be configured whether the driven and driving wheel is fixed coupled with the shaft or whether the connection is modified due to gear switching. Furthermore the diameter of the driven and driving wheel as well as the length of the belt must be defined. Normally a tolerance of 0.1 % is suitable. In the analysis it is searched for local irregularities of the belt on the configured belt gearings.
Planetary Gear	The planetary gear covers planetary gears with single planets (in contrast to stage planets). The number of planets, number of tooth of the planets, sun and annulus must be configured. Furthermore the shafts must be selected for the annulus, sun and planetary carrier. The mode of the planetary gear must be configured. One of the following combinations can be used: • sun and planetary carrier rotate and annulus is fixed • sun and annulus rotate and planetary carrier is fixed • planetary carrier and annulus rotate and sun is fixed • all shafts rotate The modes can also be set as free parameter if the state changes due to gear shifting. Normally a tolerance of 0.1 % is suitable. In the analysis it is searched for local irregularities of the sun, annulus and planets as well as for revolving deviation of flank shapes on the configured planetary gears.

Drive elements	Description
Rolling Bearing	For the rolling bearing the kinematic parameters must be configured. These are the rotational cage order, the ball spin order, the double ball spin order and the ball pass order of the outer and inner ring. These data can be obtained from the manufacturer of the rolling bearing. The input of the rotational orders can be done with respect to order 1 or with respect to the relative speed of the rolling bearing. Furthermore the shafts for the outer and inner ring must be chosen. If one of these is fixed the shaft can be left empty. If the rolling bearing is configured as planetary rolling bearing no shafts must be set. Normally a tolerance of 1 % is suitable to compensate slippage of the rolling bearing. In the analysis it is searched for outer and inner ring damage, cage damage and rolling element damage on the configured rolling bearings.
Three-Phase Motor	For the three-phase motor the net frequency must be configured. Normally a tolerance of 0.1 % is suitable. In the analysis it is searched for magnetic imbalance on the configured three-phase motors.
Steady Frequency	The elements steady frequency and steady frequency envelope can be used to monitor individual frequencies in the frequency spectrum and envelope frequency spectrum respectively. The frequency must be configured.
Steady Order	The elements steady order and steady order envelope can be used to monitor individual orders in the order spectrum and envelope order spectrum respectively. The order must be configured.

9.7.4.1.2 Gears

The screenshot shows the 'imc STUDIO Powertrain Monitoring - Configuration' window. The 'Diagnosis' tab is active, and the 'Gears' sub-tab is selected. The 'Number of gears' is set to 7, and the 'Übernehmen' button is visible. The main table shows the configuration for 7 gears (G1-G6, R1) with columns for 'Coupled', 'Decoupled', 'Fixed', and 'Coupled' states. Below this, a table shows the resulting transmission of each shaft, including Motorwelle, Zwischenwelle 1-9, and Ausgangswelle, with values for speed and torque.

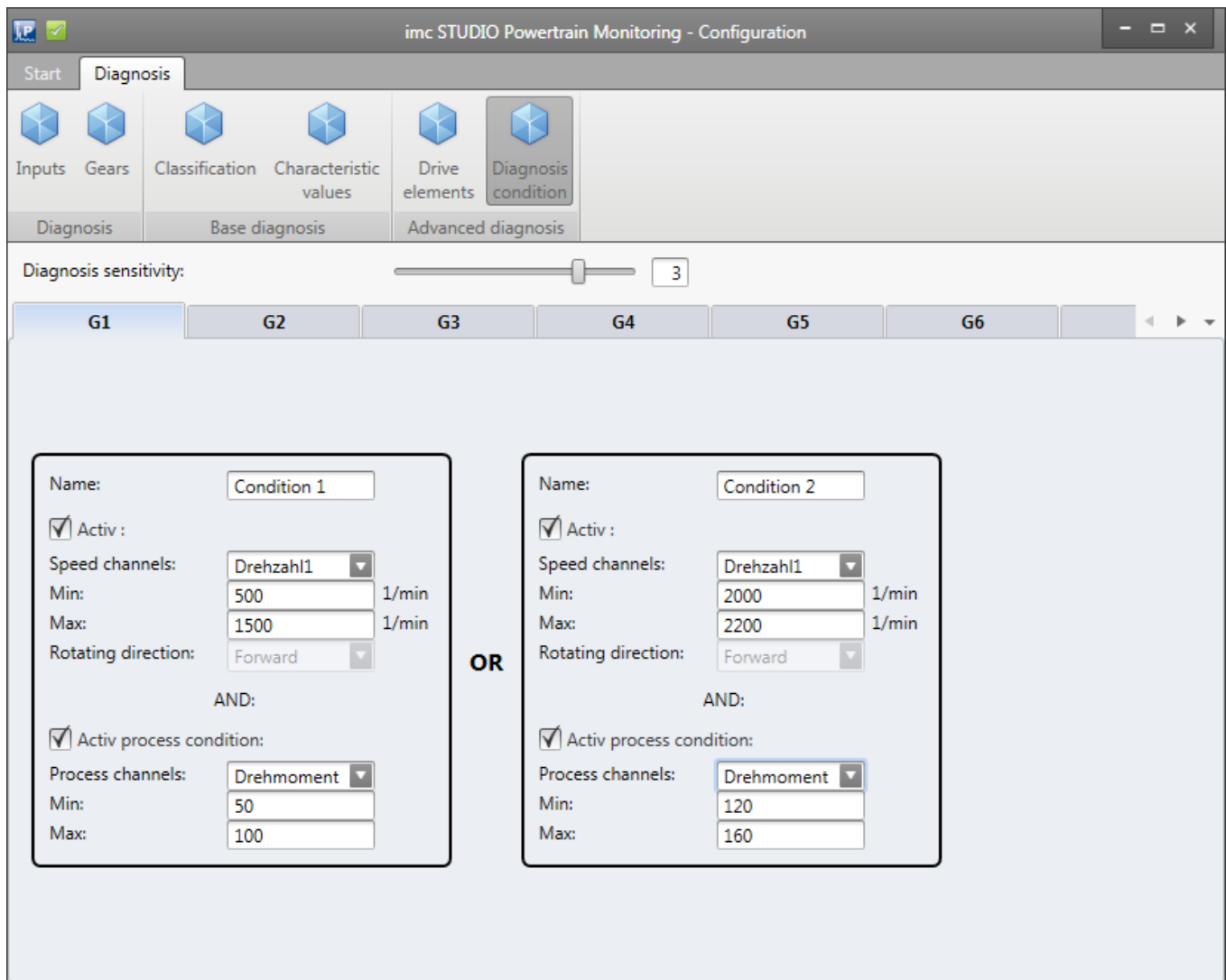
	G1	G2	G3	G4	G5	G6	R1
Kupplung A (Coupled)	Yes	Yes	Yes	Yes	Yes	No	No
PG 1 (Annulus)	Decoupled	Coupled	Coupled	Coupled	Decoupled	Decoupled	Coupled
PG 1 (Sun)	Coupled	Coupled	Coupled	Fixed	Decoupled	Decoupled	Coupled
PG 2 (Annulus)	Fixed	Coupled	Coupled	Coupled	Fixed	Fixed	Fixed
PG 2 (Sun)	Coupled	Decoupled	Decoupled	Decoupled	Coupled	Decoupled	Coupled
PG 3 (Annulus)	Coupled	Coupled	Coupled	Coupled	Coupled	Fixed	Coupled
PG 3 (Sun)	Fixed	Fixed	Coupled	Coupled	Coupled	Coupled	Fixed
PG 4 (Sun)	Fixed	Fixed	Coupled	Coupled	Coupled	Coupled	Fixed
Kupplung B (Coupled)	No	Yes	Yes	No	No	No	Yes
Kupplung E (Coupled)	No	No	Yes	Yes	Yes	Yes	No

	G1	G2	G3	G4	G5	G6	R1
Motorwelle	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000
Zwischenwelle 1	1,0000	1,0000	1,0000	1,0000	1,0000	0,0000	-1,3182
Zwischenwelle 2	-0,7586	1,0000	1,0000	0,0000	-0,7586	0,0000	1,0000
Zwischenwelle 3	0,4052	1,0000	1,0000	0,6618	0,4052	0,0000	-0,5341
festeWelle	0,0000	1,0000	1,0000	0,4314	0,0000	0,0000	0,0000
Zwischenwelle 4	0,0000	0,0000	1,0000	1,8859	2,5579	3,6190	0,0000
Zwischenwelle 5	0,2932	0,7237	1,0000	1,0000	1,0000	1,0000	-0,3865
Zwischenwelle 6	0,2122	0,5237	1,0000	1,2448	1,4305	1,7237	-0,2797
Zwischenwelle 7	-1,0968	-1,0968	-1,0968	-1,0968	-1,0968	-1,0968	-1,0968
Zwischenwelle 8	1,0968	1,0968	1,0968	1,0968	1,0968	1,0968	1,0968
Zwischenwelle 9	-0,1782	-0,4399	-0,8400	-1,0456	-1,2016	-1,4479	0,2350
Ausgangswelle	0,0653	0,1611	0,3076	0,3829	0,4400	0,5302	-0,0860

Picture 12: Assistant Gear configuration

In the menu point Gears the free parameter can be set for each configured gear as described in chapter "[Gears](#)". The resulting transmission of each shaft is shown in the table below as an overview. Thus it can be traced whether the complete power train was entered correctly. Picture 12 shows an example configuration of a power train with 10 gears.

9.7.4.1.3 Diagnosis conditions

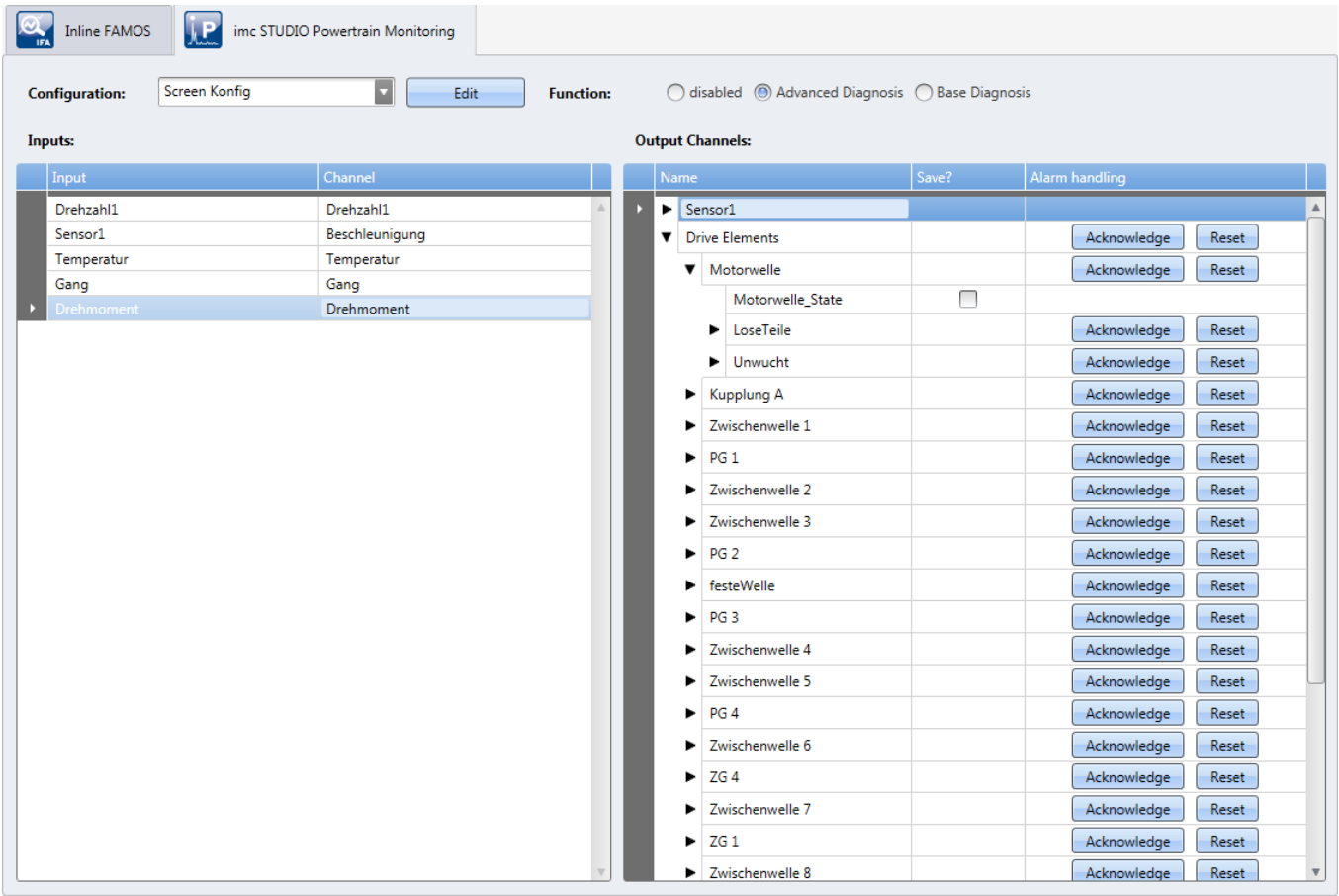


Picture 13: Assistant Diagnosis Conditions

In the menu point diagnosis condition the sensitivity of the advanced analysis must be set. This parameter corresponds to the number of detections before an irregularity is signaled.

Furthermore 5 different diagnosis conditions can be set for each gear individually. One diagnosis condition can consist of a speed range and one process range (see Picture 13). If one of the conditions are fulfilled over the complete measurement time, the analysis of kinematic patterns will be started.

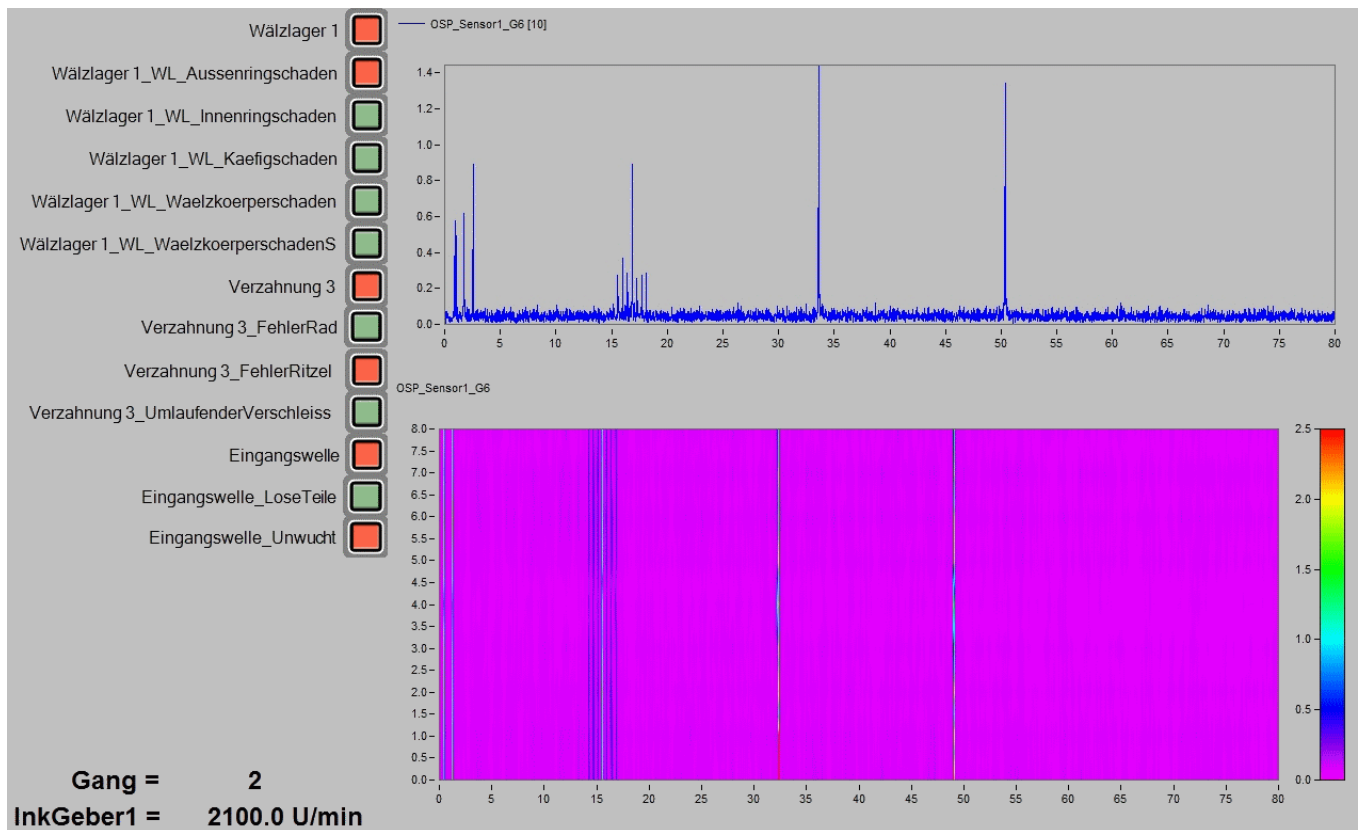
9.7.4.2 Editor



Picture 14: Editor Advanced Diagnosis

For the configuration of the advanced analysis the inputs of the configuration must be assigned to the physical measurement channels (see also general [description of the editor](#)⁴⁹⁶).

9.7.4.3 Measurement data



Picture 15: Example: Advanced Diagnosis

The following output channels are provided for each configured sensor in the advanced analysis:

- Order spectra for each configured gear
- Envelope order spectra for each configured gear
- Frequency spectra for each configured gear
- Envelope frequency spectra for each configured gear

Furthermore the following output channels are provided for each configured drive element:

- For each possible irregularity of one [drive element](#)⁵⁰⁴ a **state channel** whether this irregularity is detected (0 or 1)
- **One state channel** for each drive element whether one or multiple irregularities are detected (0 or 1)
- For each diagnostic attribute **three channels** with the **order**, **amplitude** and **significance** of the kinematic pattern

All these channels are triggered channels. Thus the time point of the measurement is saved in the channel.

The signaled irregularities can be acknowledged or reset in the Powertrain Monitoring Editor (see picture [Editor](#)⁵⁰⁹). An Acknowledgement increases the significance threshold for the searched orders by the given percent value and resets the alarm state. Thus in the following analysis this irregularity is signaled only if the new threshold is exceeded. A reset of an alarm resets the significance threshold to its origin value.

Panel pages offer various possibilities to visualize the input and processed data. Picture 15 shows one Panel page exemplary. The last calculated order spectrum of Sensor1 in the 6th gear is visualized. Furthermore all events of the same order spectrum are visualized in a color map. Irregularities of the drive elements are visualized on the left part of the Panel page with Widgets. On the left bottom corner the current gear and the current speed is shown.

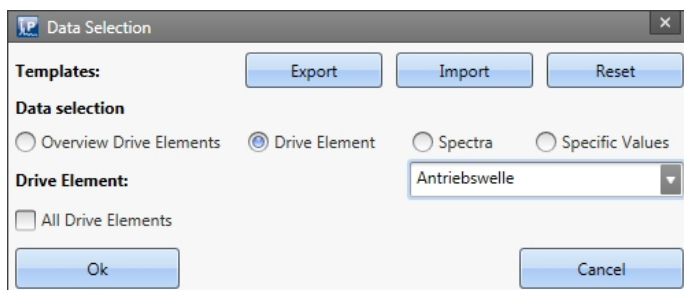
9.7.4.4 Summary of function

- Evaluating whether one of the diagnosis condition is fulfilled over the configured measurement time.
- If one diagnosis condition is fulfilled an evaluation of the vibration signal is done.
 - Calculation of all frequency, envelope frequency, order and envelope order spectra of all vibration signals.
 - Calculation of diagnostic attributes of all kinematic frequencies or orders.
 - Determining whether an irregularity is detected with respect to the preceding analysis.
- Output of the spectra, diagnostic attributes and alarm states in the corresponding channels.

9.7.5 Visualization

9.7.5.1 Create Panel pages via menu action

In imc STUDIO an additional action "Powertrain Monitoring Panel" can be added to a Ribbon via the function "Customize". If this new action is executed, Panel pages with selected data from the Powertrain Monitoring can be generated automatically (Picture 16).



Picture 16: Powertrain Monitoring Panel

The dialogue directly accesses the tasks of the Powertrain Monitoring of the current experiment. Thus all data of the configured tasks are available in the selected category. It can be chosen between the following categories:

- **Overview Drive Elements:** Generates an overview page with the alarm state of all configured drive elements.
- **Drive Element:** A Panel page for the selected drive element is generated with state information for each damage pattern.
 - If the option "All Drive Elements" is chosen, this Panel page is created for each drive element in the list. This can lead to lot of Panel pages, depending on the configuration.
- **Spectra:** For the selected sensor and gear one Panel page with order spectrum, envelope order spectrum, frequency spectrum and envelope frequency spectrum is generated. These are visualized as color map.
- **Specific Values:** A Panel page with all configured specific values at the chosen sensor is created. For each specific value the alarm state for each gear, an aggregated alarm state as well as a curve window with the specific value and the corresponding warn and alarm thresholds are visualized.

In order that the channel assignment is working properly, the default channel names in the Powertrain Monitoring Tasks must be used. These are automatically generated when a task is applied. Additionally the used Panel pages can be exported, imported and reset via the "Powertrain Monitoring Panel". Further information can be found in chapter [Manage Panel Templates](#) ⁵¹².

9.7.5.2 Create Panel pages via scripting

The described functionality in "[Create Panel Pages via Menu Action](#)"⁵¹¹ is also accessible via the imc STUDIO Scripting Editor. Thus the generated Panel pages can be individualist e.g. by adding further channels that do not come from the Powertrain Monitoring tasks.

To have the Powertrain Monitoring accessible from one Script, a reference to the .NET Assembly `imc.Studio.PlugIns.GfM.PowertrainMonitoring.Scripting.dll` must be added to the project. This can be done via the menu `Project > AddReference > GAC`.

A short example for creating a Panel page for the first drive element that can be found in the Powertrain Monitoring configuration follows.

```
using System.Linq;
using System.Collections.Generic;
using imc.Studio.Interfaces.V2.Core;
using imc.Studio.PlugIns.GfM.PowertrainMonitoring.Scripting;

// get reference to PowertrainMonitoring component
ISharedComponentBase shared_comp;
SharedComponents.TryGetValue("PowertrainMonitoring", out shared_comp);
var pm_ref = shared_comp as IPowertrainMonitoringBase;

if (pm_ref == null)
{
    //no reference to Powertrain Monitoring found
    return;
}

//use template directory of current project
string template_dir = pm_ref.GetTemplateDir();
//get first element of DriveElement List
var element = pm_ref.GetDriveElementList().FirstOrDefault();

if (element != null)
{
    //get Panel template file and insert Panel
    string panel_file = template_dir + pm_ref.GetDriveElementType(element) + ".dbv";
    string page_name = Panel.InsertPageBefore(0, panel_file, "", element);

    //get reference to Panel and apply Powertrain Monitoring variables
    PanelPage pp = Panel.Pages.FirstOrDefault(p => p.Name == page_name);
    pm_ref.ApplyVariablesDriveElement(pp.Base, element);

    //e.g. add additional channels to Panel page
}
```

9.7.5.3 Manage Panel templates

With the installation of the Powertrain Monitoring there are also Panel templates for the different data types and their visualization installed. These can be redesigned as needed. In the menu action "Powertrain Monitoring Panel" there is a function available to export all templates to a directory. The exported Panel pages can be loaded in imc STUDIO like normal Panel pages. After redesigning the pages the new Panel pages can be activated via the import function of the menu action.

These are then saved with the current project. Thus they are also exported if the imc STUDIO project is exported. If the project is changed the templates may be imported again by the user. Via the function reset the user designed templates are overwritten with the default ones.

When designing the Panel templates it is crucial to maintain the exact file names.

Table 1: Panel templates

Template File	Description
AntriebOverview.dbv	Overview Drive Elements
Welle.dbv	Diagnosis Data Shaft
KardanWelle.dbv	Diagnosis Data Drive Shaft
Kupplung.dbv	Diagnosis Data Clutch
Waelzlager.dbv	Diagnosis Data Rolling Bearing
Zahnradgetriebe.dbv	Diagnosis Data Gear drive
Planetengetriebe.dbv	Diagnosis Data Planetary gearing
Riemengetriebe.dbv	Diagnosis Data Belt gearing
Drehstrommotor.dbv	Diagnosis Data Three-phase motor
FesteFrequenz.dbv	Diagnosis Data Steady Frequency
FesteFrequenzHK.dbv	Diagnosis Data Steady Frequency Envelope
FesteOrdnung.dbv	Diagnosis Data Steady Order
FesteOrdnungHK.dbv	Diagnosis Data Steady Order Envelope
Spektren.dbv	Frequency and order spectra of one sensor in one gear
Grunddiagnose.dbv	R.M.S. of Base diagnosis of one sensor

The Panel templates contain Widgets with specific names. On the basis of these names it will be determined which channels are assigned to the Widgets. Furthermore there are placeholder that are replaced with the instantiated data types. Generally it is crucial to take care of the exact spelling of the placeholders and Widget names. Otherwise a correct assignment of the channels to the Widgets is not possible.

In the following tables an overview of all names and their channel assignment is shown.

Table 2: Used placeholders

Placeholder	Description	Usage
{Sensor}	Name of the sensor in the configuration	Specific values
{Antrieb}	Name of the drive element in the configuration	Diagnosis data of one drive element
{Antrieb1} .. {Antrieb30}	Name of the chosen drive element	Overview of drive elements

Table 3: Widget names for spectra

Widget name	Channel assignment	Example Channel name
Signal_OSP	Order spectrum of one sensor in one gear	OSP#Sensor1#G1
Signal_HKOSP	Envelope order spectrum of one sensor in one gear	HKOSP#Sensor1#G2
Signal_FSP	Frequency spectrum of one sensor in one gear	FSP#Sensor1#G3
Signal_HKFSP	Envelope frequency spectrum of one sensor in one gear	HKFSP#Sensor1#G4

Table 4: Widget names for Base Diagnosis

Widget name	Channel assignment	Example Channel name
Status_VEff	Gear dependent status channel of r.m.s. of vibration velocity	Sensor1#G1#VEff#Status
Signal_VEff	r.m.s. of vibration velocity and corresponding warn and alarm thresholds	Sensor1#VEff#Alarmschwelle
Status_AEff	Gear dependent status channel of r.m.s. of vibration acceleration	Sensor1#G2#AEff#Status
Signal_AEff	r.m.s. of vibration acceleration and corresponding warn and alarm thresholds	Sensor1#AEff#Warnschwelle
Status_AEffHK	Gear dependent status channel of r.m.s. of vibration acceleration envelope	Sensor1#G3#AEffHK#Status
Signal_AEffHK	r.m.s. of vibration acceleration envelope and corresponding warn and alarm thresholds	Sensor1#AEffHK

Table 5: Widget names for Diagnosis data

Widget name	Channel assignment	Example Channel name
Signifikanz_{Damage}	All significance values of the chosen drive element at the current damage pattern	Antriebswelle#LoseTeile#S2#Grundordnung_OSP#Signifikanz
Amplitude_{Damage}	All amplitude values of the chosen drive element at the current damage pattern	Antriebswelle#LoseTeile#S2#Grundordnung_OSP#Amplitude
Ordnung_{Damage}	All order values of the chosen drive element at the current damage pattern	Antriebswelle#LoseTeile#S2#Grundordnung_OSP#Ordnung

The following listing shows all drive elements and their available damage pattern that are used in the diagnosis data. It must be used the exact given spelling of the name for these damage pattern.

Table 6: Names for damage pattern

Drive Element	Damage Pattern	Name
Shaft	Imbalance	Unwucht
	Loose particles	LoseTeile
Drive shaft	Imbalance	Unwucht
	Loose particles	LoseTeile
	Link damage or misalignment	Gelenkfehler
Clutch	Misalignment	Ausrichtfehler
Rolling Bearing	Cage damage	WL_Kaefigschaden
	Rolling element damage	WL_Waelzkoerperschaden
	Rolling element damage spin	WL_WaelzkoerperschadenS
	Outer ring damage	WL_Aussenringschaden
	Inner ring damage	WL_Innenringschaden
Gear Drive	Revolving tooth damage	UmlaufenderVerschleiss
	Local irregularity wheel	FehlerRad
	Local irregularity pinion	FehlerRitzel
Planteray Gear	Revolving tooth damage	UmlaufenderSchaden
	Local irregularity planet	FehlerPlanet
	Local irregularity sun	FehlerSonne
	Local irregularity annulus	FehlerHohlrad
Belt Gear	Local irregularity belt	FehlerRiemen
Three-Phase Motor	Magnetical imbalance	MagnetischeUnwucht
Steady Frequency	Steady frequency	FesteFrequenzSchaden
Steady Frequency Envelope	Steady frequency envelope	FesteFrequenzHKSchaden
Steady Order	Steady order	FesteOrdnungSchaden
Steady Order Envelope	Steady order envelope	FesteOrdnungHKSchaden

9.8 imc WFT Assistant

The WFT-Plug-in provides a convenient Assistant for configuration of wheel force transducers, plus pre-made Panel pages designed for optimal display of measurement results.

Reference

Manuals for CRFX/WFT-2 and WFT-sensor

For info on utilization and operation of

- the CRFX/WFT-2 module, see the imc CRONOS Manual
- the WFT-sensor, see imc WFT Manual.

Prerequisites:

- Device software: imc STUDIO 5.2 R25 and as of imc STUDIO 2024 R2
- The "imc WFT Assistant" is activated in the "*Product configurator*".
- Only one device may be selected for measurement.
- The base unit **CRFX-2000GP** is supported (**recommended: CRFX-2000GP-WFT**) with at least one WFT-2 module.
- When using remote control:
 - The **CRFX-2000GP-WFT** base unit is required. This special base unit includes a terminal for remote control.
 - Remote control requires 1 DIIn-bit and 4 DOut-bits. For this purpose, the last bits of the respective port are reserved. They may not be utilized for any other purpose.
 - The control commands in imc Online FAMOS must be activated.
 - Timer 8 may not be in use.
- During configuration, the connection with the device must be active; no measurement may be running.

Opening the Assistant

Select the item "WFT" in the menu ribbon. The WFT Assistant is then displayed in a dialog.

Ribbon	View
Start > WFT 	all
Setup-configuration > WFT 	Complete

Menu item: WFT Load

- loads the previous configuration, if applicable
- checks whether any WFT-sensors are connected

When there are more than two WFT-2 modules, the first two module-IDs are always used.

The parameter "Status" lists the respective results.

Status	Color	Description
Active	green	sensor recognized • all channels are active and • all channel names match the prefix/postfix-pattern⁵¹⁷
Connected	yellow	sensor recognized • not all channels are active, or • at least one channel name does not match the prefix/postfix-pattern⁵¹⁷, or • the serial number does not match that of the last one connected
Missing	red	sensor missing
Passive	transparent	no sensor recognized

! Note

A different WFT-sensor is connected

When the WFT-sensor is connected, if its serial number does not match the serial number of the sensor, this sensor is displayed as "connected". Additionally, the serial number of the previously connected WFT-sensor is listed.

When the sensor information is imported, the serial number in the experiment is updated so that in consequence, updating the configuration again causes the serial number to no longer be displayed!

Rename channels - Prefix/postfix-pattern

A WFT-sensor's channels can be renamed to feature a prefix before and a postfix after the specified middle portion of the name. Empty elements are allowed, as long as all WFT-sensors each have different names. Any changes to the channel names are highlighted in yellow.

! Notes

Renaming via the Channels table not recommended

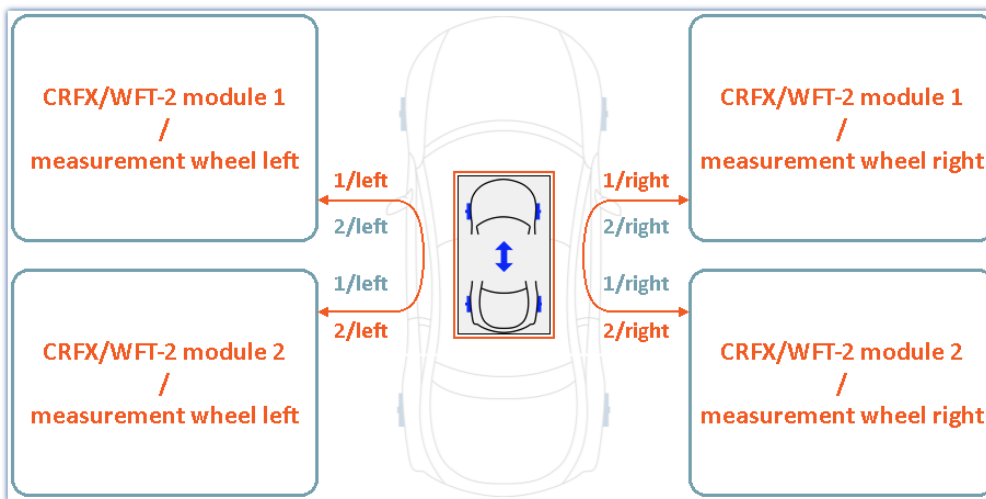
Changing the channel names in the Channels table is not recommended. The changes will be reversed upon the next call of the WFT-Assistant. Ideally you would use the prefix and postfix in order to be able to jointly modify the channel names of associated channels.

Positioning modules/sensors and modifying the axis configuration

Positioning of the WFT-modules and sensors is performed by means of the buttons in the middle of the Assistant.

Two WFT-modules

When using **two WFT-modules**, it is only possible to switch the assignment from front to back. A module having two sensors is always assigned to an axis.

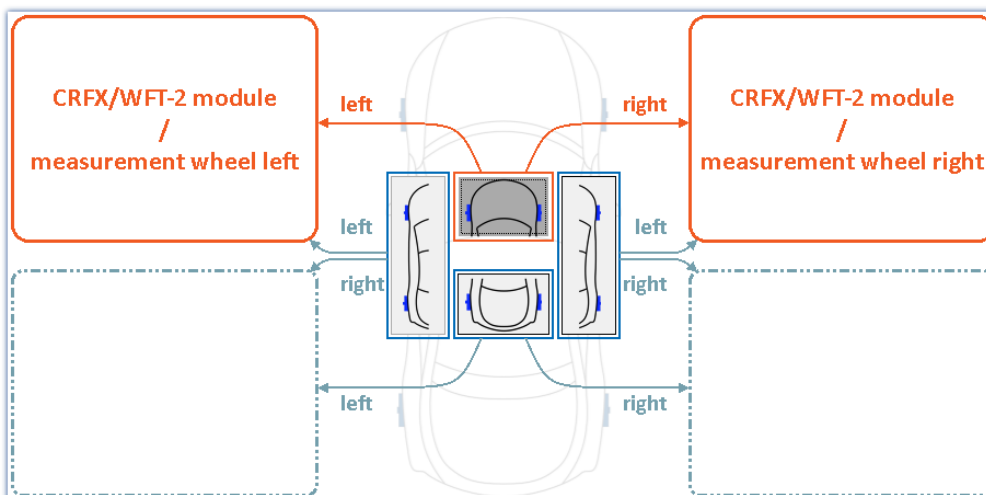


Two CRFX/WFT-2 modules, each with two wheel force transducers

One WFT-module

When using **one WFT-module**, the two sensors can be positioned in front, back, right or left.

When only one CRFX/WFT-2 module and a single-sided configuration (left/right side of vehicle) is used, the system automatically makes an internal correction to apply the reversal of the WFT's coordinate system to the new wheel position. For a configuration for the left vehicle side, the right-side sensor is corrected, and for a configuration for the right vehicle side, the left-side sensor is corrected.



One CRFX/WFT-2 module with two wheel force transducers

Correction of names upon change of axis assignment

When the axis assignment is changed in this way, the following actions are performed automatically:

- Default postfix names are automatically modified. For instance, if the sensor is now located at "rear left", then "_fl" (front left) is automatically changed to "_rl".
 - Any manually modified postfix names remain unchanged. The channel names "migrate" with the axis assignment.
- Any manual changes to the channel names via the Channels table are reset (see notes on "[Renaming via Channels table](#)"⁵¹⁷).
- Any missing or wrong names are replaced with the default names (e.g. "WFT_", "_fl").

Menu item: Apply Setup

Various actions are performed when the item "Apply Setup" is selected. The system checks the configuration and applies it in accordance with the "Status". The channels' names are set, and the actions for angle compensation are enabled. Closing options (Panel/Remote Control) can now also be enabled.

Status	Actions which depend on the Status	Additional actions
Active	all of the sensor's channels 1. are renamed, as defined by the prefix and postfix⁵¹⁷ and 2. are activated	3. if applicable, modification of the algebraic sign in case of single-sided configuration with only one WFT-2 module 4. channel parameter "Balance at device startup" is deactivated 5. imc Online FAMOS source text of the remote control is deleted (see explanation regarding imc Online FAMOS⁵²²) 6. if appropriate, the Digital-In/Out-bits used are reset to their default names
Connected		
Missing	all of the sensor's channels 1. are reset to their default names 2. are deactivated (passive)	
Passive		

 Notes

Re-activating remote control

When remote control is used, it must be re-activated before closing the Assistant.

Menu item: Show Angle

The angles of the individual wheel force transducers are displayed. It is possible to check the position and, if appropriate, to adjust it.

- The button is active as soon as the current configuration is loaded.
- Only WFT-sensors having the Status "Active" are displayed. If necessary, first run the function "Apply Setup".

Menu item: Angular Zero

The spirit level angle adjustment is started. The Assistant guides you through the necessary steps. Once adjustment of the angle has been performed successfully, the display of the wheel force transducers should match the alignment of the wheel force transducers to the vehicle.



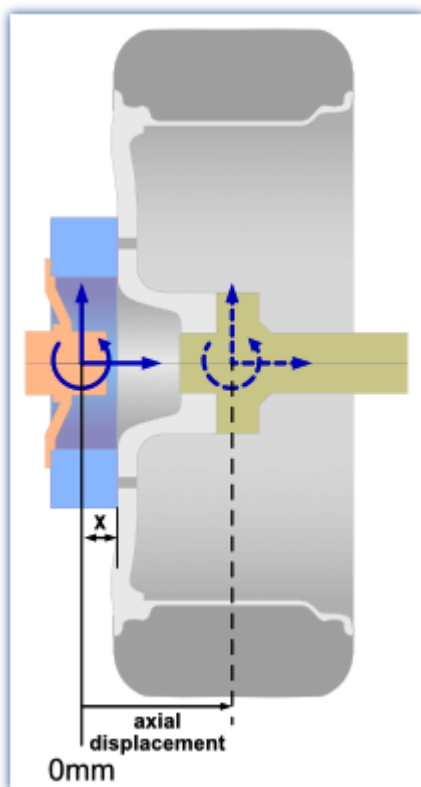
Reference

Spirit level angle adjustment

More information on the spirit level angle adjustment is presented in the WFT manual.

Axial displacement

The wheel force transducer is not mounted axially in the center of the vehicle wheel, in consequence of which there is an offset along the wheel's axis (see graphic for axial displacement). Since the forces are exerted orthogonally to this axis, torque effects result in the x- and z-planes in proportion to the offset. If the offset is known, the software can compensate for the discrepancies.



Axial displacement

In order to apply a calculated shift of the coordinate system along the axis, follow the instructions in the dialog for the axial offset.

Menu item: Close

This performs concluding actions and closes the Assistant. There are two options for closing the Assistant: "Remote Control" and "Panels"

With "Remote Control" option activated:

- Activates the Remote Control (imc Online FAMOS-source text is integrated into the existing text - see explanations regarding [imc Online FAMOS](#)⁵²²).
- The Digital-In/Out-bits are activated and renamed.
- The channel parameter "Balance at device startup" is activated for the active sensors.

With "Panels" option activated:

- The Panel pages are imported (and [overwritten if appropriate](#)⁵²¹).
- The channel names in the curve windows are modified according to the current configuration. Channels which are not active are deleted from the curve window.
Any curve windows which no longer contain channels once the channel names have been replaced are hidden. Any Panel pages which no longer have visible curve windows after replacement of the channel names are deleted.

Panel page

There are pre-configured Panel pages which are imported when the pertinent option offered upon closing the Assistant is activated. These Panel pages are located in the following folder:

"C:\Users\Public\Documents\imc\imc STUDIO\WFT"

The pages are named according to the following pattern: "WFT_WIDGET_*.dbv". If no pages exist there yet, they will be newly created there by the WFT-Assistant.



Notes

Pages with "WFT_WIDGET_" in their name will be overwritten

Upon importing, any Panel pages having the same internal "Name" (not the same "Title") as one of the pages to be imported are deleted without any warning message and replaced with the new page. If you do not intend for the pages generated by the WFT-Assistant to be deleted, the internal "Name" of the page must be changed. The WFT-Assistant does not change or delete any pages which do not match the name pattern.

Changing the templates

The Panel page templates can be modified and changed in accordance with your personal wishes. During import, the channels in the curve windows are assimilated to the current names of the channels. For this purpose, the names specified must adhere to the following pattern:

"WFT_WIDGET_*_FL", "WFT_WIDGET_*_FR", "WFT_WIDGET_*_RL", "WFT_WIDGET_*_RR"

The asterisk * stands for the channel name in English (e.g. "Angle", "RPM", "Revs"). Important! Only the channels in the **curve windows** are replaced by the WFT-Assistant! Other Widgets remain unaffected.

Deleting templates

Individual Panel-files can be deleted from the folder. They are then no longer imported by the WFT-Assistant. Only once there is no longer any pertinent Panel-file present will the WFT-Assistant save the files in the folder again.

Preallocation of the Digital In/Out-bits

Remote control requires 1 DIn-bit and 4 DOut-bits. For this purpose, the last bits of the respective port are allocated automatically. In order for the bits to be recognizable by the WFT-Assistant, they are renamed accordingly.

imc Online FAMOS

Some of the remote control options require their own imc Online FAMOS-source text. For this purpose, the existing imc Online FAMOS-source text is modified. The WFT-Assistant modifies only regions within the keywords `;WFT_WIDGET` and `;/WFT_WIDGET`. The other regions remain unchanged and can be used as accustomed. Only the "Timer (8)" is needed. This may not be used for other purposes.

The control commands are used and must be activated manually. In order for exchange of the source text to be possible, the control commands must be uniquely identifiable. For this reason, please do not insert any comments in the control command's keyword line. The source text elements are always inserted at the beginning of a control command by the WFT-Assistant. Any control commands which are not yet present are appended at the end.

The template for the imc Online FAMOS-source text is saved in the following folder:

```
"C:\Users\Public\Documents\imc\imc STUDIO\WFT"
```

The template may not be changed. If the file is missing, it is saved there again upon the next start of the WFT-Assistant. Additionally, all temporary files such as the exported and unmodified imc Online FAMOS-file are saved there.

10 Channel types, variable types, data types and data storage formats

In imc STUDIO Monitor, the various components work with variables, and some work with files as well. Variables can take a variety of formats and types.

This chapter provides an overview of the various possibilities and applications.

Overview

Synopsis	Section
What kinds of data types are available? What are the differences between them?	• Data types ⁵²³ • Structured data sets - Event-based or sequenced channel ⁵²⁴ • Time-stamped channels - Setting the RAM-buffer duration, circular buffer duration and pretrigger duration ⁵²⁵
What variable types and channel types are there? Where do the variables come from?	• Monitor channels ⁵²⁷ and GPS variables ⁵³³ • Trigger, devices and system variables ⁵²⁸ • Process vector variables ⁵³¹ and Ethernet bits ⁵²⁷ • User-defined variables ⁵³⁶
In what format are the variables saved? How to export/import configurations?	• Data file format - Saving measured data during measurement ⁵⁴⁰ • Memory requirements for measurement data ⁵⁴² • Parameter set ⁵⁴⁶

10.1 Data types

The main characteristic of a variable is its data type. The following list presents a select number of data types. Additional, personally designed types are also possible. Some can be generated by means of Scripting, others by using imc FAMOS.

Data type	Description
Normal data set	A normal channel (equidistant) is a one-dimensional field (vector) of real numbers with units, names etc. A sequence of sampled values of a measured quantity with a fixed sampling rate represents a normal channel. These include for example the analog channels, equidistantly captured fieldbus channels, ...
Single values	A single value is an individual real number having a name and in some cases a unit as well. These include, for example: Display-, Process vector variables, user-defined numeric variables, ...
Bits	A variable having the values 0 or 1. These include, for example: Ethernet-bits, virtual bits
Text-variables	A user-defined text. These include, for example: user-defined text-variables

Data type	Description
XY-data sets	Time-stamped channel with numerical values. XY-data sets arise where two values must be stored for each data point, namely, the (amplitude) value and additionally the time- or x-value. Example: CAN-channel with the setting "x-axis": "<i>Time-stamp</i>"  Observe the notes on making the settings for RAM-buffer duration, circular buffer duration and pretrigger duration ⁵²⁵.
TimeStamp-ASCII (TSA)	Time-stamped channel with ASCII-values, also known as TSA. Such data sets contain a list of texts (or generally, blocks of ASCII-code), each of which comes with its own time stamp (similar to XY data sets). TSA-channels are used for the message protocols of the various fieldbusses. Example: RecordEvent/RecordText, various fieldbus protocols, ...  Observe the notes on making the settings for RAM-buffer duration, circular buffer duration and pretrigger duration ⁵²⁵.
Vectors	A defined set of single values. These include, for example: various vectors from imc Online FAMOS or imc Inline FAMOS.
Text array	Text arrays from imc FAMOS can't be read and evaluated in imc STUDIO Monitor by ordinary means. Widgets are not able to handle text arrays. It is possible to access variables via a script. A text array can be imported from a file. Exchange in both directions can be accomplished by means of an imc FAMOS-command. Thus, text arrays can be created using imc FAMOS, transferred to imc STUDIO Monitor and then passed back to imc FAMOS at a later time, in order to be filled by imc FAMOS with additional entries.

10.1.1 Structured data sets - Eventierter oder sequentierter Kanal

Normally a data set consists of a series of measurement values sampled at a fixed sampling rate. Various applications require options for concatenating data sets.

Segments

Example: The imc Online FAMOS function "FFT" - This function generates one new segment for each time a user-specified number of points accumulates.

A data set may be segmented, with each segment having the same length and the entire data set's length being a whole number multiple of the segment length. The segmented data sets also include matrices.

Events

Example: Multiple trigger-events in one data set.

A data set can also be structured as several events, which, unlike segments, can take different lengths. Thus, an additional list containing the beginning and length of each event must be managed along with the data set. Various data set characteristics, such as trigger time and sampling rate, are also managed in the capacity of characteristics of particular events.



Reference

See also

Storage options and directory structure > "[Events and circular buffer](#)" ²⁰¹

10.1.2 Time-stamped channels - Setting the RAM-buffer duration, circular buffer duration and pretrigger duration

For time-stamped numeric channels (XY-data sets)

The RAM-buffer duration, circular buffer duration and pretrigger duration are specified as a time duration on the Setup pages. For time-stamped channels, the imc STUDIO Monitor software assumes an average sampling rate $S_{assumed}$ of 100 samples per second. This is necessary in order for the software to convert the specified time duration into a buffer size expressed in samples.

If the user knows the actual average sample rate S_{real} , then the duration to set can be calculated according to the following equation:

$$t_{to\text{set}} = t_{real} \cdot \frac{S_{real}}{S_{assumed}}$$

$t_{to\text{set}}$	the RAM-buffer duration, circular buffer duration or pretrigger duration [s]
t_{real}	the desired actual RAM-buffer duration, circular buffer duration or pretrigger duration [s]
$S_{assumed}$	the average sample rate assumed by the software $S_{assumed} = 100 \text{ Samples/s}$
S_{real}	the actual average sample rate in Samples/s



Example

The actual sample rate S_{real} is 1200 Samples/s

The desired buffer duration t_{real} is 1min.

$$t_{to\text{set}} = t_{real} \cdot \frac{S_{real}}{S_{assumed}}$$

$$t_{to\text{set}} = 60 \text{ s} \cdot \frac{1200 \text{ Samples/s}}{100 \text{ Samples/s}}$$

$$t_{to\text{set}} = 720 \text{ s}$$

For time-stamped text channels (TSA)

Time-stamped text channels (TSA-channels) are used for, among other things, message protocols of the various fieldbusses. A sample in such a channel consists of a time stamp and text or payload byte of arbitrary length, plus multiple bytes specific to the TSA format. The RAM-buffer duration, circular buffer duration and pretrigger duration for these channels is specified in the dialogs *Configuration or Storage* as a time duration. The program assumes an average sampling rate $S_{assumed}$ of 100 samples per second, where for each sample a size $L_{assumed}$ of 20 bytes per sample is assumed. These 20 bytes correspond to about the size of one sample (or entry) in the fieldbusses message protocol. These presumed values are necessary for the program to be able to convert the specified buffer duration to a buffer size in bytes.

In order to determine the RAM-buffer duration, circular buffer duration and pretrigger duration $t_{to\text{set}}$ to set, the actual average data rate D_{real} must be calculated. This data rate depends on two estimated values.

$$D_{real} = S_{real} \cdot (L_{real} + 12 \text{ Bytes/Sample})$$

D_{real}	the actual average data rate [Bytes/s]
S_{real}	the actual average sample rate [Sample/s]
L_{real}	the actual average number of payload bytes per sample [Bytes/Sample]

If the user knows the actual average data rate D_{real} , then the duration to set can be calculate according to the following equation:

$$t_{tose} = t_{real} \cdot D_{real} / D_{assumed}$$

t_{tose}	the RAM-buffer duration, circular buffer duration or pretrigger duration [s]
t_{real}	the desired actual RAM-buffer duration, circular buffer duration or pretrigger duration [s]
$D_{assumed}$	the average data rate assumed by the program [Bytes/s]
	$D_{assumed} = S_{assumed} \cdot L_{assumed}$
	$D_{assumed} = 100 \text{ Sample/s} \cdot 20 \text{ Bytes/Sample}$
	$D_{assumed} = 2000 \text{ Bytes/s}$
D_{real}	the actual average data rate [Bytes/s]



Example

The channel is a TSA channel.

The actual average data rate S_{real} is 150 Samples/s

On average, 48 payload bytes per sample are contained: $L_{real} = 48 \text{ Bytes/Sample}$

$$D_{real} = S_{real} \cdot (L_{real} + 12 \text{ Bytes/Sample})$$

$$D_{real} = 150 \text{ Samples/s} \cdot (48 \text{ Bytes/Sample} + 12 \text{ Bytes/Sample})$$

$$D_{real} = 9000 \text{ Bytes/s}$$

The desired buffer duration t_{real} is 1 min.

$$t_{tose} = t_{real} \cdot \frac{D_{real}}{D_{assumed}}$$

$$t_{tose} = 60 \text{ s} \cdot \frac{9000 \text{ Bytes/Sample}}{2000 \text{ Bytes/s}}$$

$$t_{tose} = 279 \text{ s}$$

10.2 Various variable types - Channel type

Along with the data type, the variable's source impacts how the variables can be used.

When the source is a device, one usually speaks of channel types. These variables are configured in the Channel-table on the respective Setup-page. This applies to analog channels, Monitor channels, Ethernet-bits, ... A list of all channel types is presented in the section "[Channel table](#)"^[182].

There are also local variables on the PC. These are generated using the dialog "[User-defined variables](#)"^[536].

In the next few section you will be introduced to a variety of variable types which require a detailed look and which can not be described adequately just in terms of their data type.

- [Monitor channels](#)^[527]
- [Ethernet bits](#)^[527]

- [Trigger, devices and system variables](#) ⁵²⁸
- [Process vector variables \(pv-variables\)](#) ⁵³¹
- [GPS](#) ⁵³³
- [User-defined variables](#) ⁵³⁶

10.2.1 Monitor channels

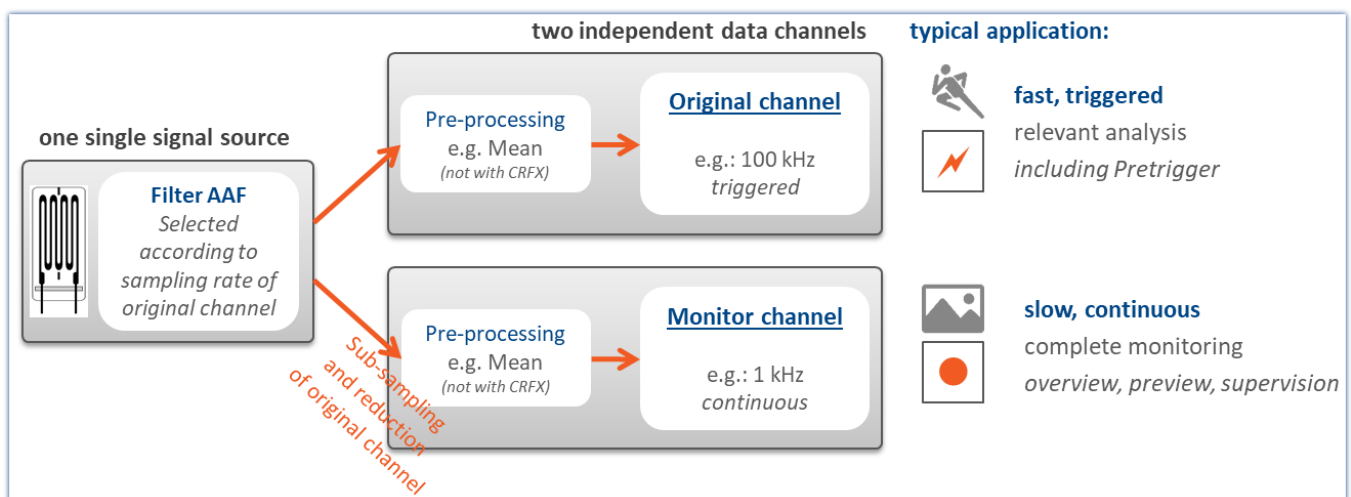
Monitor channels - For devices belonging to the [firmware group A](#) ¹¹⁶ (imc DEVICES)

Monitor channels are digital copies of the input signals and serve the purpose of continuous monitoring. Their settings for triggering, data storage and the **sampling rate** do not need to match those of the respective original channel. By means of monitor channels, it is possible to display the plot of captured signals before occurrence of the trigger event, at a low sampling rate.

Their **maximum sampling rate** is that of the input channel. The monitor channel does not have its own independent low-pass filter. For this reason, at low resampling rates, **aliasing-effects can occur**.

When an input channel is configured for "*preprocessing*", the monitor channel receives the unprocessed raw data.

The following channel types can have Monitor channels: Analog inputs, Digital inputs and Incremental counter inputs. Most fieldbus interfaces also offer an option for creating monitor channels.



Concept for monitor channels

10.2.2 Ethernet bits

imc STUDIO Monitor can derive trigger conditions from its own analog and digital channels as well as from AND/OR combinations of these. When working with multiple devices, it's often necessary to announce a trigger event to all devices.

When working with multiple channels, it is often necessary to announce a trigger event to all the devices. It is often difficult to lay digital lines in a widely dispersed network. One solution is to use an Ethernet-connection, if available. This is accomplished using the so-called "*Ethernet bits*". One way to think of them is as 32 hard-wired digital lines running parallel to the Ethernet, which connect all devices mutually. Thus the device has an Ethernet-DIO port, in addition to the physical DIO. These digital inputs and outputs can be used for triggering just like the unit's physical DIO-bits.

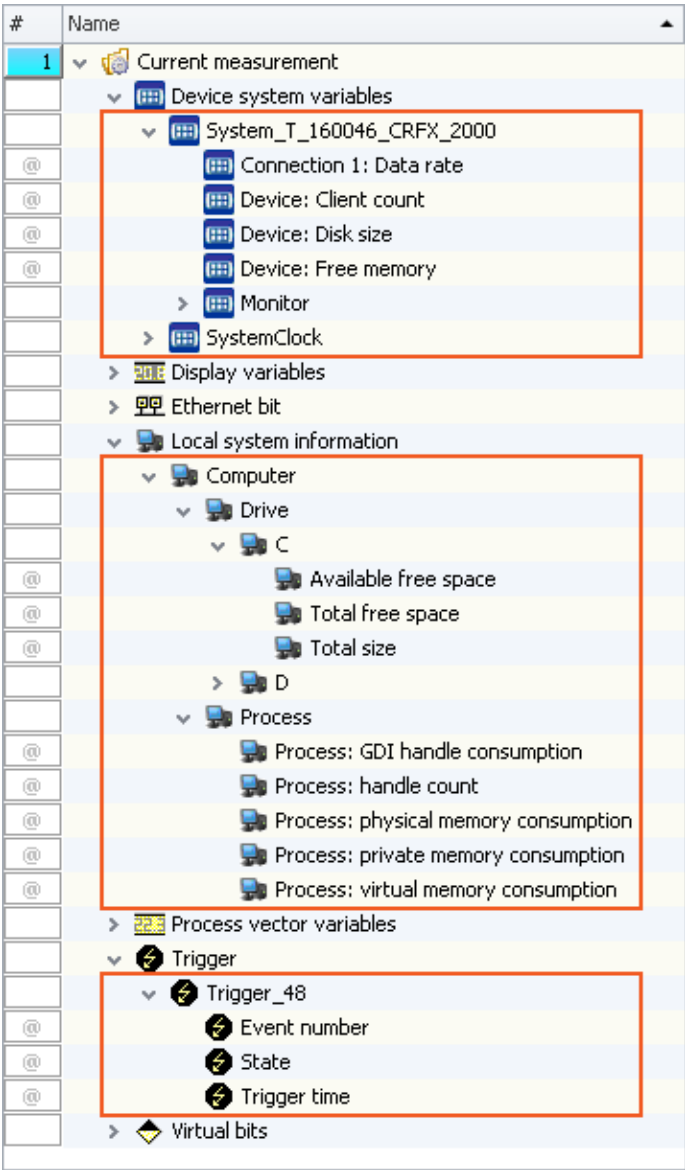
In this context, note the following:

- Ethernet bits only take effect within the scope of one protocol.
- The ethernet bits of TCP/IP devices only take effect within their Subnet.
- Another analogy to the physical DIO also applies: just as in the Ethernet, there is a delay between the setting of a bit and the recognition by another device of the new state.

 **Note**

When a measurement is prepared, all bits, including the Ethernet bits, are reset. This also affects devices on the network which are currently performing measurements. Any Ethernet bits which had been set in the process of the running measurements will be reset when any device in the network is prepared

10.2.3 Trigger, devices and system variables



In an expanded tree diagram, the following variables are displayed:

Variables	Description
Device system variables	Utilization and info on device status, storage space, measurement status, connection status and synchronization status, as well as time information
Local system information	User information, system load and hard disk status
Trigger	Status and information about the triggers

These variables contain various kinds of information which are each displayed separately and which can easily be moved to the Panel by means of Drag&Drop.

Trigger		Description
Trigger Name e.g. Trigger_01 or BaseTrigger	Event number	With multi-triggering: count of previous trigger releases
	State	Current state of the trigger: armed, released, stopped
	Trigger time	Time when the state last changed
Device system variable <Device name>	Description	
Free memory	Available memory in internal data storage medium	
Disk size	Capacity of internal data storage medium	
Client count	Count of active connections to the device. E.g. imc STUDIO Monitor, imc STUDIO Monitor or imc REMOTE connections or a data storage medium used. If imc REMOTE WebServer is enabled in the device properties, a connection is permanently occupied.	
Connection 1	First detected connection with the device (imc STUDIO or imc WAVE).	
Data rate	Currently transmitted Bytes/s at the receiver (instantaneous value)	
Critical channel fill level	Proportional (in percent) fill level of the "critical channel's" FIFO. The "critical channel" is the channel with the highest fill level. The fill level is allowed to rise temporarily and have high peaks. However, it may never reach 100%, or a data overflow occurs in consequence. If the fill level rises not temporarily but constantly, this indicates an overflow will occur later.	
Critical channel name	Name of "critical channel", see " <i>Critical channel fill level</i> ".	
Connection 2 Monitor 1	First connection with imc STUDIO Monitor (corresponds to " <i>Connection 1</i> ").	
Connection x Monitor n	Additional connections	

Device system variable System	Description
Experiment	
Measurement status	Returns the state of the measurement: measurement running (1), measurement stopped (0), unclear (-1) 1: Measurement running applies when at least one device is performing a measurement 0: Measurement stopped applies when all devices are known not to be performing measurement -1: Unclear applies when at least one device is not connected, and no connected device is running
Synchronization status	Returns the device's synchronization status: Synchronized (1), not synchronized (0) 1: Synchronized, when all devices are synchronized 0: Not synchronized, when at least one device is not synchronized Devices which are not intended to be synchronized are not included in the status check.
Connection status	Returns the status of the device's connection with imc STUDIO Monitor: connected (1), not connected (0) 1: Connected, when all devices are connected 0: Not connected when at least one device is not connected
SystemClock	
PC time	Current PC clock time
System time	Current clock time of imc STUDIO Monitor. This time is used for all components of imc STUDIO Monitor.
Local system information	Description
User	Using this variable, you can automatically include an indication in your report of which user is logged-in.
Name	Name of the logged-in user.
Role	Allocated role of the logged-in user.
Computer	
Drive n	Information about the local data carrier. Pay attention to the free memory space if you are trying to save large volumes of data.
Total free space	Free memory on the local data carrier. This property reflects the total amount of free memory on the drive, not just the portion available to the current Windows user. In most cases, the parameter "Available free space" is to be used.
Total size	Data storage capacity of the local data carrier.
Available free space	Free memory space on the local data carrier. This property reflects the amount of free memory space available to the current Windows user.
Process	Various kinds of system information. For information and value limits, see the associated Internet forums. Monitoring these values can be of interest in long-term testing.

10.2.4 Process vector variables (pv-variables)

The "Process vector" is a collection of **single-value variables** (process vector variables or pv-variables). **For all measurement channels** of the device, these single-value variables are already **pre-defined**.



Note

Values are available immediately

Updating of the device pv-variables occurs immediately after preparing the measurement device. Thus, the pv-variables always display the current value, even if the measurement is not yet running, or has already concluded, or the channel-trigger has not yet released.

imc Online FAMOS works independently of the trigger mechanism and can thus access the measured values currently being captured by means of the variables. These can then be used, in conjunction with the "Synchronous Tasks", for monitoring and control purposes.

Variants of the pv-variables

Variants	Description
pv-variables of the data acquisition channels	pv-variables of the data acquisition channels (e.g. analog channels, CAN, ...) always show the currently captured measurement value starting from the action "Prepare".
User-defined pv-variable	pv-variables personally defined by the user, e.g. by means of imc Online FAMOS. Prerequisite: A firmware group A ¹¹⁶ device.
Display GPS Information ⁵³³	GPS signal are made available as pv-variable and fieldbus channels (with GPS interface)
State of the external power "pv.State.ExternalPower"	The pv-variable takes the following value: 0 = state unknown/not yet found (initialization value) 1 = external current supply present ("power ok") 2 = external current supply not present/deactivated ("power fail")
n of system time "pv.State.SyncTimeDeviation"	Only in conjunction with NTP- and PTP-synchronization Current time deviation of devices from the NTP-or PTP-Server in seconds. The return value is exactly 0 if the server is either not activated or cannot be reached; otherwise it is in the range of a few milliseconds.

User-defined pv-variable

Prerequisite: A [firmware group A](#) ¹¹⁶ device.

In parallel to the already existing device-pv-variables, you can create your own pv-variables. This is conditional on the following requirements:

Component Rights	read	create	write
imc Online FAMOS	yes	no	no
imc Online FAMOS Professional	yes	yes	yes
imc Inline FAMOS	yes	no	yes (if they were created using imc Online FAMOS Professional)

Creating a user-defined pv-variable

Not possible in imc STUDIO Monitor, only with imc STUDIO/imc WAVE.

Properties of the pv-variables

Properties	Description										
Trigger independence	Updating of the device pv-variables occurs immediately after preparing the measurement device. Thus, the pv-variables always display the current value, even if the measurement is not yet running, or has already concluded, or the channel-trigger has not yet released.										
Disallow read-/write-access	By means of the variable's "Status" (Setup page "Variables"), you are able to set whether other additional components are able to see the variable. <table> <tr> <th>Status</th><th>Description</th></tr> <tr> <td>Active</td><td> The variable exists in the device and is transferred to the PC. The following rules apply to user-defined pv-variables: No write-permission on the PC: The value can only be written via imc Online FAMOS. Other components such as the Panel can only read the value but not write it. If the initialization in imc Online FAMOS is removed, the variable remains in the list. The measurement can only be started if the variable is initialized in at least one location (e.g. imc Online FAMOS or Automation). Otherwise this error message appears: Error number: 6408 <i>"There is at least one process vector variable that is not being used. Make sure that it has write permission (Setup) or is being used (e.g. imc Online FAMOS, Automation, ...)"</i> </td></tr> <tr> <td>Passive</td><td> The variable exists in the device, but is not transferred to the PC. It is only possible to access the value via imc Online FAMOS. For user-defined pv-variables, the following applies: If the initialization in imc Online FAMOS is removed, the variable is deleted from the list. </td></tr> <tr> <td>read/write</td><td>Like "active", but with write-permission on the PC; e.g. the value of the variable can be changed by means of the Panel.</td></tr> <tr> <td>write</td><td><i>not recommended</i> - The status has no significance and usually behaves like "read/write"</td></tr> </table>	Status	Description	Active	The variable exists in the device and is transferred to the PC. The following rules apply to user-defined pv-variables: No write-permission on the PC: The value can only be written via imc Online FAMOS. Other components such as the Panel can only read the value but not write it. If the initialization in imc Online FAMOS is removed, the variable remains in the list. The measurement can only be started if the variable is initialized in at least one location (e.g. imc Online FAMOS or Automation). Otherwise this error message appears: Error number: 6408 <i>"There is at least one process vector variable that is not being used. Make sure that it has write permission (Setup) or is being used (e.g. imc Online FAMOS, Automation, ...)"</i>	Passive	The variable exists in the device, but is not transferred to the PC. It is only possible to access the value via imc Online FAMOS. For user-defined pv-variables, the following applies: If the initialization in imc Online FAMOS is removed, the variable is deleted from the list.	read/write	Like "active", but with write-permission on the PC ; e.g. the value of the variable can be changed by means of the Panel.	write	<i>not recommended</i> - The status has no significance and usually behaves like "read/write"
Status	Description										
Active	The variable exists in the device and is transferred to the PC. The following rules apply to user-defined pv-variables: No write-permission on the PC: The value can only be written via imc Online FAMOS. Other components such as the Panel can only read the value but not write it. If the initialization in imc Online FAMOS is removed, the variable remains in the list. The measurement can only be started if the variable is initialized in at least one location (e.g. imc Online FAMOS or Automation). Otherwise this error message appears: Error number: 6408 <i>"There is at least one process vector variable that is not being used. Make sure that it has write permission (Setup) or is being used (e.g. imc Online FAMOS, Automation, ...)"</i>										
Passive	The variable exists in the device, but is not transferred to the PC. It is only possible to access the value via imc Online FAMOS. For user-defined pv-variables, the following applies: If the initialization in imc Online FAMOS is removed, the variable is deleted from the list.										
read/write	Like "active", but with write-permission on the PC ; e.g. the value of the variable can be changed by means of the Panel.										
write	<i>not recommended</i> - The status has no significance and usually behaves like "read/write"										
Direct read-access via some fieldbusses	Appropriate variants of the CAN-Bus and LIN-Bus are able to read the current values of the pv-variables and to output these on the bus.  More detailed information including the prerequisites are presented in the descriptions for the respective fieldbusses.										

10.2.5 GPS

At the GPS socket it is possible to connect a GPS-receiver. This makes it possible to achieve absolute **synchronization to GPS time**. If the GPS-mouse has reception, the measurement system synchronizes itself automatically. **Synchronization with a NMEA source** is possible. The precondition for this is that the clock must return the GPRMC-string along with the one-second-interval clock signal.

All **GPS information** can be **evaluated** and subjected to **subsequent processing** by imc Inline FAMOS.

GPS signals are **available as**: process vector variables and fieldbus channels.

GPS information	Description
pv.GPS.course	Course in °
pv.GPS.course_variation	Magnetic declination in °
pv.GPS.hdop	Dilution of precision for horizontal
pv.GPS.height	Height over sea level (over geoid) in meter
pv.GPS.height_geoidal	Height geoid minus height ellipsoid (WGS84) in meter
pv.GPS.latitude pv.GPS.longitude	Latitude and longitude in degree (Scaled with 1E-7)
pv.GPS.pdop	Dilution of precision for position
pv.GPS.quality	GPS quality indicator 0 Invalid position or position not available 1 GPS standard mode, fix valid 2 differential GPS, fix valid ...
pv.GPS.satellites	Number of used satellites.
pv.GPS.speed	Speed in km/h
pv.GPS.time.sec	The number of seconds since 01.01.1970 00:00 hours UTC. For this reason, it is no longer possible to assign the value to a Float-format channel without loss of data. This count of seconds can be transformed to absolute time under Windows and Linux. To do this, use the function below. <code>MySeconds = CreateVChannelInt(Channel_001, pv.GPS.time.sec)</code>
pv.GPS.vdop	Dilution of precision for vertical see e.g. www.iota-es.de/federspiel/gps_artikel.html (German)

 Note**Scaling of the latitude and longitude**

pv.GPS.latitude and pv.GPS.longitude are **INT32 values, scaled with 1E-7**. They must be **treated as Integer channels**, otherwise the **precision is diminished**.

By means of imc Online FAMOS, you are able to generate virtual channels from them. However, due to the reversal of the scaling, precision is lost:

```
latitude = Channel_001*0+pv.GPS.latitude *1E-7
```

Recommendation: Use the corresponding fieldbus channel: "*GPS.latitude*" or "*GPS.longitude*". Here, no scaling is required, so that the precision is preserved.

Sampling rate

Due to system limitations, GPS channels for determining the fastest sampling rate in the system are not taken into account. For an working configuration, at least **one other channel** (fieldbus, digital or analog) must be sampled at either the **same** sampling rate as the GPS-channel, or a **faster** one.

Internal variables; do not use

- pv.GPS.counter
- pv.GPS.test
- pv.GPS.time.rel
- pv.GPS.time.usec

GPS-Receiver

The **GARMIN GPS receivers** supplied by imc are set ready for operation and provide a 1 Hz or 5 Hz pulse, depending on the model.

The following conditions must be met in order to use other GPS receivers from imc devices:

- **RS232 port settings**
 - **Baud rate:** Possible values are 4800, 9600, 19200, 38400, 57600 or 115200
 - 8 bit, 1 stop bit, no flow control
- The following **NMEA strings** must be sent: **GPRMC, GPGGA, GPGSA**. The order of the strings must be adhered to.
Additional strings should be deactivated. If this is not possible, all other strings must be **before** the GPGSA string!
- The receiver must deliver a **1 Hz clock**.
- The rising edge of the clock must mark the second specified in the next GPRMC string.
- All three strings should be sent as soon as possible after the 1 Hz clock, so that there is sufficient time for processing between the last string and the next 1 Hz clock.

NMEA-Talker IDs

Supported NMEA-Talker IDs:

- GA: Galileo Positioning System
- GB: BeiDou (BDS) (China)
- GI: NavIC (IRNSS) (India)
- GL: GLONASS, according to IEIC 61162-1
- GN: Combination of multiple satellite systems (GNSS) (NMEA 1083)
- GP: Global Positioning System (GPS)
- GQ: QZSS regional GPS augmentation system (Japan)

10.2.6 User-defined variables

imc STUDIO Monitor provides the user with the ability to **create** and use **user-defined variables**.



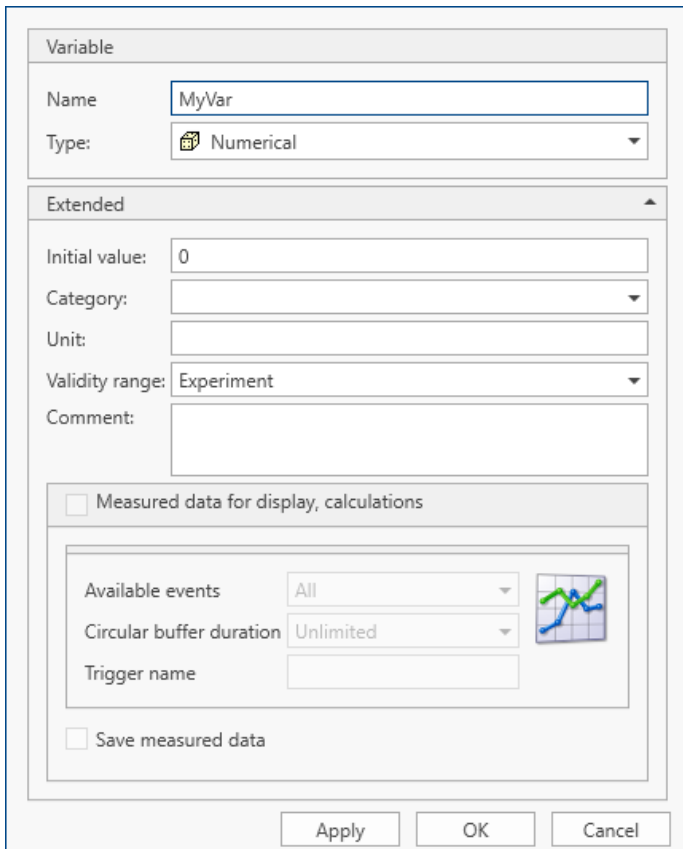
Note

Variables are not available in the device

User-defined variables only exist on the PC. It is not possible to access them from the device.

Creating variables

You can create user-defined variables in the Data Browser by clicking on the symbol . Furthermore, at certain locations where variables can be used, you will find a button with the caption "Create", for example in the command "[Set Variables](#)" . A dialog opens which helps you in setting up the variable.



The dialog box is titled "Create User-defined Variable" and is divided into two main sections: "Variable" and "Extended".

Variable Section:

- Name:** A text input field containing "MyVar".
- Type:** A dropdown menu showing "Numerical" with a small cube icon.

Extended Section:

- Initial value:** A text input field containing "0".
- Category:** A dropdown menu.
- Unit:** A text input field.
- Validity range:** A dropdown menu showing "Experiment".
- Comment:** A large text area.

Measured data for display, calculations: A checkbox that is currently unchecked.

Available events: A dropdown menu showing "All".

Circular buffer duration: A dropdown menu showing "Unlimited".

Trigger name: A text input field.

Save measured data: A checkbox that is currently unchecked.

Buttons: "Apply", "OK", and "Cancel" are located at the bottom right.

Create User-defined Variable dialog

The dialog contains two regions. "Variable" contains an input box for the name and a selection box for the type. In the lower region "Extended", you can specify additional properties such as the "Initial value" and "Unit".

Variable

Parameter	Description						
Name	Here, enter a name for your variable. All alphanumeric characters are permitted. If you use the name of an already existing variable, the existing one is deleted and a new one with the specified properties is created. ▼ User-defined ▼ Result a1 a2 To join multiple variables together in a group, you can use the character '.' in the variable's name. E.g. <i>Result.a1</i>, <i>Result.a2</i> results in this display in the Data Browser. <i>Display in the Data Browser</i>						
Type	For your user-defined variable, you can choose among various types: <table><tr><th>Typ</th><th>Description</th></tr><tr><td>Numerical</td><td>Saves a number; this type suitable both for integer values and for floating points numbers. Under <i>Initial value</i>, you can set the value with which to initialize the variable (see the section <i>Extended</i>). The default for this value is 0.0.</td></tr><tr><td>Text</td><td>Saves an arbitrary text. Under <i>Initial value</i>, you can set the value with which to initialize the variable (see the section <i>Extended</i>). The default for this value is an empty string, represented by the readout <i>Empty Value</i> in gray letters.</td></tr></table>	Typ	Description	Numerical	Saves a number ; this type suitable both for integer values and for floating points numbers. Under <i>Initial value</i> , you can set the value with which to initialize the variable (see the section <i>Extended</i>). The default for this value is 0.0.	Text	Saves an arbitrary text . Under <i>Initial value</i> , you can set the value with which to initialize the variable (see the section <i>Extended</i>). The default for this value is an empty string, represented by the readout <i>Empty Value</i> in gray letters.
Typ	Description						
Numerical	Saves a number ; this type suitable both for integer values and for floating points numbers. Under <i>Initial value</i> , you can set the value with which to initialize the variable (see the section <i>Extended</i>). The default for this value is 0.0.						
Text	Saves an arbitrary text . Under <i>Initial value</i> , you can set the value with which to initialize the variable (see the section <i>Extended</i>). The default for this value is an empty string, represented by the readout <i>Empty Value</i> in gray letters.						

Extended

Parameter	Description
Initial value	Your variable is initialized with this value. This means that when creating the variables, when loading an experiment and upon resetting of the variables ("Set Variables"⁹⁹⁹), the value entered here is written in the variable. In this box, it is only possible to enter anything if the variable's type is Numerical or Text. With all other types, this box is disabled and the variable is empty when created.

Parameter	Description
Category	Assigning a category gives you the ability to structure the variables in the Data Browser. If there are already categories of user-defined variables, then to display them, click on the symbol  in the box <i>Category</i> after entering the name. This expands a selection list of all previously existing categories. The categories only serve structuring purposes and do not influence what values the variables can take. Custom variables created in custom main category When you create a variable having no category, it is automatically placed among the "<i>User-defined variables</i>". When you create a variable having a category, this variable is not created under "<i>User-defined variables</i>" but parallel with that category. This allows you to create your own custom folder structures. If you wish to create a category within the category "<i>User-defined variables</i>", do the following: write a "\" before the category name.  Category: "MeasurementPoint_1". A category called "MeasurementPoint_1" is created in parallel with the other categories, such as "<i>Analog Inputs</i>". It contains the variable. Category "\"MeasurementPoint_1". This creates a category "MeasurementPoint_1" within the category "<i>User-defined variables</i>". It contains the variable. This applies, for example, to variables created by means of the command: "Load variable"  or to generating a user-defined variable by means of the Data Browser.
Unit	You can assign a unit to your variables.

Parameter	Description												
Validity range	The setting Validity Range determines the availability of your variable. What is saved is always only the variable itself, not the current value! <table> <tr> <th>Validity Range</th><th>Description</th></tr> <tr> <td>Temporary</td><td>Temporary variables are associated with the session, which means they only exist until imc STUDIO is closed. Upon the next start they are no longer available.</td></tr> <tr> <td>Experiment</td><td>Variables with this validity range are associated with the current experiment. They are saved along with it and are available again after loading.</td></tr> <tr> <td>Sequencer</td><td>Variables with this validity range are associated with the Sequencer. You should use this validity range if you intend to load other experiments via the Sequencer. Your variables remain intact even if the new experiment has its own variables.</td></tr> <tr> <td>Project</td><td>Variables with this validity range are associated with the current project. They are thus available for all of a project's experiments.</td></tr> <tr> <td>Persistent</td><td>Persistent variables are project-dependent and globally available for all projects and experiments.</td></tr> </table> Any "higher-level" variables are not overwritten if a variable of the same name but from a different validity range is loaded.  Example: Suppose a variable exists with the validity range "Project" or "Sequencer". An experiment is loaded by means of the command: "Open experiment". A variable having the same name exists in this experiment and has the validity range "Experiment". This variable is not loaded from the experiment when the command is run.  Please also observe the notes on the action: "Load/Refill variable" in regard to the "Validity range".	Validity Range	Description	Temporary	Temporary variables are associated with the session, which means they only exist until imc STUDIO is closed. Upon the next start they are no longer available.	Experiment	Variables with this validity range are associated with the current experiment. They are saved along with it and are available again after loading.	Sequencer	Variables with this validity range are associated with the Sequencer. You should use this validity range if you intend to load other experiments via the Sequencer. Your variables remain intact even if the new experiment has its own variables.	Project	Variables with this validity range are associated with the current project. They are thus available for all of a project's experiments.	Persistent	Persistent variables are project-dependent and globally available for all projects and experiments.
Validity Range	Description												
Temporary	Temporary variables are associated with the session, which means they only exist until imc STUDIO is closed. Upon the next start they are no longer available.												
Experiment	Variables with this validity range are associated with the current experiment. They are saved along with it and are available again after loading.												
Sequencer	Variables with this validity range are associated with the Sequencer. You should use this validity range if you intend to load other experiments via the Sequencer. Your variables remain intact even if the new experiment has its own variables.												
Project	Variables with this validity range are associated with the current project. They are thus available for all of a project's experiments.												
Persistent	Persistent variables are project-dependent and globally available for all projects and experiments.												
Comment	It is possible to add a comment to a variable. It is seen in the Data Browser in the column <i>Comment</i> , or can be displayed in a Widget with the help of placeholders , for example.												
Measured data for display, calculations	This option currently has no function.												
Save measured data	This option currently has no function.												

10.3 Data file format - Saving measured data and export/import formats

With imc STUDIO Monitor, you are able to save a variety of elements (variables, configurations) in various formats. You can either save the measured data directly during measurement, or export them subsequently in a different format. You can save device configurations and later import them again.

Next, a selection of available formats will be presented. Significantly more formats are available for export of variables than can be mentioned here. Please refer to the information on the respective "export template", if provided, for more details.

10.3.1 Data file format - Saving measured data during the measurement

By default, the measured data are saved in the file format imc FAMOS. The desired file format can be set in the file format selection list:

Format	Description
imc FAMOS	Default format for all device variables.
imc FAMOS ZIP 	Compressed format for many fieldbus variables.
CANAllyse-MDF 	CANAllyser-MDF format for CAN-Log data.



Note

"imc FAMOS ZIP" and "CANAllyse-MDF": Applicable with devices of the [firmware group A](#)  (imc DEVICES).

imc FAMOS

Each channel is saved in a separate file with the extension *.raw. raw stands for raw data, meaning the original data returned by the measurement device.



Reference

Detailed documentation on this file format is presented in the manual:

"imcSharedComponents.pdf" > File Assistant > File Format.

There you will find descriptions of the modules which jointly use imc FAMOS and imc STUDIO Monitor.

Measured data saved by your device can be read directly by imc FAMOS (required imc FAMOS version: see "Technical data sheet").

With which version and which device were the measured data generated?

The saved measurement files record with which version of imc STUDIO and with which firmware/3rd-party device/DataProcessing and which device the measurement was performed. For this purpose, the *.raw-file can be opened with a text editor. This info is readable whichever data format (imc2 or imc3) is used.



Note

Conversion to EXCEL or ASCII

If you want to convert the imc measurement data into a EXCEL table or a ASCII file, use the component:

"[imc Format Converter](#)" .

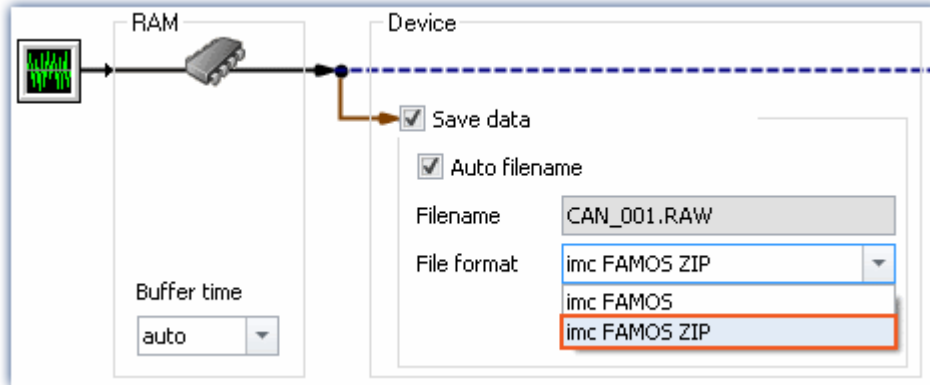
imc FAMOS ZIP

For certain channel types, it is possible to **reduce the disk space requirements** (file format: "imc FAMOS ZIP"). In the background, the file size is minimized by means of zip-compression. The result depends accordingly on the nature of the signal.

The filename remains unchanged. As well, the handling of the file, for instance with imc FAMOS is still the same (required imc FAMOS version: see "Technical data sheet").

Data compression is possible for the following channel types:

- analog/digital fieldbus channels
- log channels of the CAN-fieldbus
- digital input-ports



! Note

limitation: computing power of the basic system

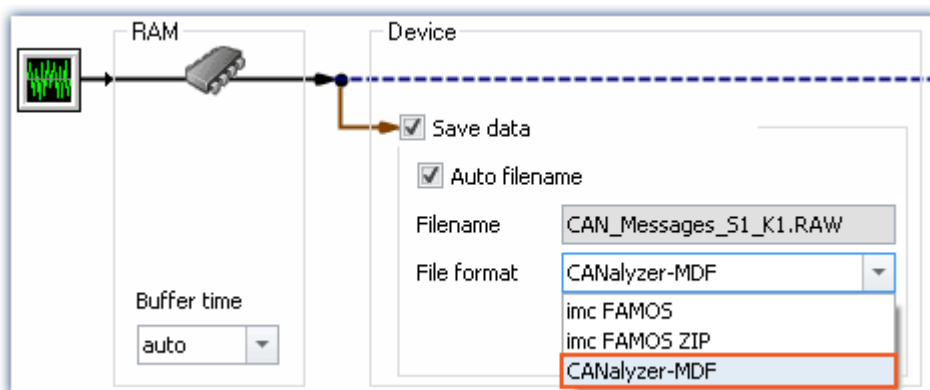
The compression of the data is limited by the computing power of the basic system. An **overload due to too many individually compressed channels can lead to a restart of the system!** The **number of channels** is the most important factor. Typical measurements with an imc BUSDAQ-4 showed that up to **60 channels** can be successfully compressed permanently.

If the compression is limited to the log channels, an overload does not lead to a restart. However, it is possible that compression may take so long that the data cannot be retrieved in time. In this case, the system reports a data overflow.

CAN-Log Data in CANalyzer-MDF format

Logs of CAN messages can be saved either in imc FAMOS format or in the CANalyze-MDF file format.

This option is only available for the device.



! Note

- The maximum possible measurement duration is 11 hours, 55 minutes and 49.67 seconds.
- This format is only available if "Log channel" is activated for the node in the CAN Assistant.

10.3.2 Memory requirements for measurement data

For measurements saved to the **internal storage medium**, note:

If the available memory is exhausted during measurement, the measurement continues anyway (display and additional saving to PC-hard disk, in some cases).

In order to estimate the memory requirement for a measurement of a particular length, one must first distinguish among the various data types. There are distinctions between **analog** and **digital**, **reduced** and **equidistant** data, as well as between **untriggered** and **triggered measurement**. Additionally, the "*cluster size*" of the storage medium used plays a role.

Data type:

- Analog data require 16 bits per sample (or for the data type: Float 4 Byte pro Sample (whether your device supports the data type: Float is stated in the respective device manual)
- Digital data require 16 bits per 16 bit port
- Virtual channels in imc Online FAMOS require 32 Bit per sample
- The same applies to channels reduced using Transitional Recording (only possible on the device).
- When estimating the memory requirements for virtual waveforms, it is necessary to take the function used into consideration. For example, the simple RMS-value of 1000 readings reduces the data amount by that factor.

Trigger:

- For each triggering event, a file header is saved, which provides additional information on the stored data (imc FAMOS file format).
- The size of this header depends on the particular settings (e.g. header of the first Event: 1536 Byte, header of the subsequent Events: 512 Bytes)

Note

- The calculated memory requirement is the minimum memory space requirement.
- For each file on the hard drive, there is a series of entries in the FAT (file allocation table) which determine which clusters are occupied by the file, i.e., where the file contents are physically located on the disk.
- The folders, subfolders and files they contain are administered separately and occupy at least one cluster.

Data Types

Data Type	Memory Requirement	Comment
Analog channels	2(4) byte / sample *	4 Byte for the data type: float
Digital channels (16 bit DIO port)	2 byte / sample *	
Virtual channels (Data computed in imc Online FAMOS)	4 byte / sample *	
Analog channels reduced using Transitional Recording	4 byte / saved sample *	depends also on imc Online FAMOS functions
Digital channels reduced using Transitional Recording	4 byte / saved sample *	2 byte / saved sample + 2 byte for the time information

* + any file header for trigger events (size is variable)

Cluster Size

With **FAT16** formatting of the **storage medium**, the cluster size depends on the size of the storage medium. It is clear that small clusters make better use of the memory space, since less space is wasted at the ends of files by partially empty clusters.

The **PC-Hard disk** (e.g. WIN98), by contrast, has **FAT32**-formatting, in which the cluster size is constant at **4k**.

Cluster Size (FAT16)	Maximum Logical Drive Size
2 k = 2048 byte = 4 sectors	128 MB
4 k = 4096 byte = 8 sectors	256 MB
8 k = 8192 byte = 16 sectors	512 MB

DOS configures the cluster size to be as small as possible when formatting a drive. For example, if you format a 500 MB partition, you get 8 K clusters, for a 170 MB medium 4 K clusters, for a 340 MB medium 8 K clusters.

Examples



Example 1

- An analog channel is recorded with multiple triggered events. Each event is to comprise 2000 samples. A total of 100 events are recorded.
- A "naive" calculation which assumes that the data can be arbitrarily compressed would yield the following results:

Analog channel: 2 byte / sample	
100 events x 2000 samples x 2 byte = 100 x 2000 x 2 byte	400 000 byte
1st file header: 1 x 1536 byte*	+ 1 536 byte
Each additional file header: 99 x 512 byte*	+ 50 688 byte
1 cluster for the directory allocation info and 32 bytes there for file allocation info	+ 32 byte
Data size (cannot be stored this compactly!)	= 452 256 byte*

* File header for trigger events (size is variable)

- In actual fact, the data can not be packed with any arbitrary degree of compression. Instead, the storage medium's cluster structure must be taken into account. Thus, the storage of each individual measurement (trigger release, event) starts at a new cluster.
- Thus in reality the following results would emerge in an example of a 340MB hard drive:

1st event:	
1536 bytes (header) + 4000 bytes (samples) = 5536 bytes => 2 x 4 KB	8 KB
Next events:	
512 bytes (header) + 4000 bytes (samples) = 4512 bytes => 2 x 4 KB => 99 x 8 KB	792 KB
Directory and file allocation info in FAT (1 cluster)	8 KB
True memory requirement	= 808 KB*

* File header for trigger events (size is variable)

The results illustrate that the space requirements for this example's files are almost twice the actual data volume.



Example 2

If the measured data are not recorded as multiple triggered events, then a folder is created for each measurement (340MB hard drive). In total, 100 measurements each comprising 2000 samples are performed:

For each measurement:	
1536 byte + 4000 byte = 5536 byte = 2 x 4 KB => 100 x 8 KB	800 KB
Directory and file allocation info in FAT (1 cluster)	
=> 100 x 8KB	800 KB
True memory requirement	= 1 600 KB

**Example 3**

In this example, 3 analog channels are recorded with multiple triggered events. Each event contains 2000 samples. A total of 100 events are recorded. Two of the channels are saved directly to the internal storage medium and the other is filtered online (imc Online FAMOS) before being saved (only filtered results are saved). The data is to be stored to an unpartitioned 340MB hard drive (8KB cluster size):

2 analog channels à 2 byte + 1 virtual channel à 4 byte

1st event:

$1536 \text{ byte}^* + 1 \times 2000 \times 2 \text{ byte} = 5,5 \text{ KB} \Rightarrow 8 \text{ KB}$ 8 KB

$1536 \text{ byte}^* + 1 \times 2000 \times 2 \text{ byte} = 5,5 \text{ KB} \Rightarrow 8 \text{ KB}$ 8 KB

$1536 \text{ byte}^* + 1 \times 2000 \times 4 \text{ byte} = 9,5 \text{ KB} \Rightarrow 16 \text{ KB}$ 16 KB

Next events:

$512 \text{ byte}^* + 1 \times 2000 \times 2 \text{ byte} = 4,5 \text{ KB} \Rightarrow 8 \text{ KB} \Rightarrow 99 \times 8 \text{ KB}$ 792 KB

$512 \text{ byte}^* + 1 \times 2000 \times 2 \text{ byte} = 4,5 \text{ KB} \Rightarrow 8 \text{ KB} \Rightarrow 99 \times 8 \text{ KB}$ 792 KB

$512 \text{ byte}^* + 1 \times 2000 \times 4 \text{ byte} = 8,5 \text{ KB} \Rightarrow 16 \text{ KB} \Rightarrow 99 \times 16 \text{ KB}$ 1 584 KB

1 cluster for directory and 3 file allocation infos 8 KB

True memory requirement: **= 3 208 KB**

* File header for trigger events (size is variable)

10.3.3 Parameter set

What is a parameter set?

A parameter set is a group of parameters which can be used for various purposes. For instance, parameters may be **settings for the device and channel configuration**. They can also be **metadata** and **variables values**.



Example

Example: Channel configuration

Channel name	Save (PC)	Sampling time	Duration	...
Channel_001	True	1 ms	40 s	...
Channel_002	True	100 ms	10 s	...

Example: Metadata

Company name	Test object number	Test part number	Test station photo	Hint
My company	X1	X1_123	.\Picture1.png	speed test 1

Example: Variable values

Name	Category	Value	Typ	...
DisplayVar_01	DisplayVar	1	System.Single	...
DisplayVar_02	DisplayVar	10	System.Single	...
DisplayVar_03	DisplayVar	777	System.Single	...

The parameters can be exported to a file. This file is in most cases read- and writable using a text editor or EXCEL (various data types are available: see *Export formats*). In this way, the parameters can be edited and imported back at a later time. When editing, please observe the notes under: "[Editing the parameter set](#)".

What can a parameter set be used for?

Parameterizing the device- and channel configuration by means of the file's contents

There are various ways to import the file and to configure the measurement device by this means. You can save multiple parameter sets and import a particular one before the start of each measurement, depending on the requirements.

Thus, the configuration of the measurement device and of the channels can always be correctly adapted without needing to edit it manually each time.

Initializing variable values by means of the file contents

Just as you can parameterize the device configuration, you can also initialize variable values. The files contain values for the variables, which you can import at the desired time.

Saving the configuration in a file as documentation on the experiment

For purposes of documentation on the measured data or the experiment, you can save the parameter sets of device/channel configurations, or current variable values. These are available for view without the need to start imc STUDIO or even for it to be installed.

Metadata on the experiment or the measurement

You can save metadata on the experiment or measurement. Experiment metadata are displayed in the project dialogs for the selected experiment, and measurement metadata in the Data Browser for the measurement. You can perform filtering and sorting by them.



Note

Trigger-settings and Application module assignments

Trigger-settings and Application module assignments can not be exported as parameter sets in a useful way.

What contents may a parameter set have?

Content	Description
Texts and numbers	E.g. device and channel configuration (device name, channel name, terminal, coupling, sampling rate, ...), metadata (notes, e-mail address, date), texts and numeric variable values
Paths	E.g. metadata: information on the project documentation, pictures. When a parameter set is created which contains pictures or document, these are saved in an extra folder. The folder name corresponds to the filename with the extension: ".data". The path information provides a relational reference to the file. Upon export, the paths are resolved and the file can thus be imported.

File formats

A variety of file formats is available. Each format has certain advantages and should be selected according to the application.

Format	Description
Parameter set files (*.csv)	Tab-separated Text-file. Easily readable in EXCEL.
Horizontal Parameter set files (*.csv)	Tab-separated Text-file. Easily readable in EXCEL.
Parameter set files (*.txt)	Particularly suitable for metadata having only one line of text (e.g. for the Setup-page <i>Documentation</i>). Easily readable with any text editing program

Export parameter set via

Command: Export parameters

The command: "*Export parameters*" allows automated export of the parameters at desired times and according to previously adjusted settings.

There are pre-made export-configurations available such as "Export all channels" (the values of the active channels' variables) or "Export all channels and settings" (the values of the active channels' variables, and channel settings of all analog channels). Or you can manually select which variable values and settings to export.

Ribbon: Project > Import/Export > Export > Export parameter set

Corresponds to the command: "*Export parameters*", with the limitations that there are no pre-made configurations and that export is performed when you close the dialog.

Ribbon: Setup-Configuration > Export configuration

This function provides a variety of exporting options.

For most of the selection options, only the parameters (the content of the rows and columns) of the table on the currently open Setup page are exported. The table is associated with a so-called [table description](#)¹⁷⁴. In the table description is where you set which parameters (e.g. Channel name, Channel status, ...) are present. Not all parameters which would be possible are displayed in the table. For this reason, there are a variety of export options, so that it is possible to export either the visible, or the selected, or even the invisible parameters.

When export of the parameter set is performed via the menu ribbon item "Setup-Configuration" > "Export configuration", the selection dialog appears as follows:

Channels

☒ Only the selected ones

☐ Active ones on "Analog channels"

☐ All on "Analog channels"

Parameters

☒ Only the selected ones

☐ All on "Analog channels"

☐ All existing ones

☐ Balancing settings

☐ Choose manually

The **left column** presents the **selection of channels** whose parameters are to be exported. The choices offered are only the selected channels, or only the channels set to "active", or all channels.

The **right column** presents the **selection of the parameters** to be exported for the channels.

Selection	Description
Only the selected ones	All parameters which are selected are exported.
All on "Page name"	All parameters are exported which are visible on the page. These included all columns in the table, but also all parameters in the dialogs and on additional tabs (e.g. the three tabs on the page: "Channel balance")
All existing ones	All parameters defined in the table description are exported. (Including such which are not present in the table.)
Balancing settings	The balance settings are exported, regardless of the Setup page selected. The record of the experiment includes the respective balance values saved for the various settings. For this reason, the balance values for each measurement range, for example, are exported to the same file. For a detailed explanation, see "Exporting balance values (taring, scaling, bridge, ...)". Upon importing, the appropriate balancing settings are imported in accordance with the measurement range set.
Choose manually	A dialog for selecting the desired columns appears. In this dialog, select all parameters which you wish to export. All parameters selected are exported.

**Note****Export of adjustment values**

When parameters which are relevant for adjustment are imported, inconsistent states can occur if not all parameters for the adjustment are appropriate. E.g. if only the compensation values are imported. When these values are exported the system ensures that all adjustment information is available when one of these relevant columns is exported.

All of the available selections also export the "*Balance scaling information*" as soon as a relevant adjustment column is included. Exception: "*Choose columns manually*"; here only items which are actually selected are exported.

The possibility to edit the export file manually still remains, but in consequence, the adjustment information may become inconsistent.

Import parameter set via**Command: Import parameters**

The command: "*Import parameters*" allows automated import of the parameters at desired times and according to previously adjusted settings. You can also select from which file to import which variable's values and/or settings.

Ribbon: Project > Import/Export > Import > Import parameter set

Corresponds to the command: "*Import parameters*", with the limitation that import is performed when you close the dialog.

Ribbon: Setup-configuration > Import Configuration

Corresponds to the command: "*Import parameters*", with the limitation that import is performed when you close the dialog. Everything is imported, regardless of the Setup page opened.

Parameter set import-column

With the "*Parameter set import-column*", you can quickly load a variety of settings. When entries in the Setup-table are selected, the associated saved parameter set (*.csv) is loaded. See also "[Creating and configuring additional columns](#)" ¹⁵¹ > "[Parameter set import-column](#)" ¹⁶⁷.

**Note****Importing variables**

For importing variables' values, instead of the command: "*Import parameter*", the command "[Load/Refill variable](#)" ⁹⁹⁵ should be used. This command is specially designed for variable values.

Using the command: "*Import parameter*" can cause unintended alteration of the device control.

 Note**How the correct target is found**

Parameter sets are saved in the form of tables. When importing previously saved parameter sets, the following rules apply in normal cases:

- **Target table:** As the target table (table description), the table in which the most columns are found is used. Note that the columns are found on the basis of their display names. It doesn't matter whether the short or long display name is used for a column.
- **Language:** The file name will contain a language-indicator abbreviation in most cases. By means of this, a parameter set can also be imported from a different language (see "[Note: Language abbreviation](#)"⁵⁵¹).
- **Device-/channel configuration or datapool (variable values):** The datapool is recognized as the target as soon as at least the columns "Name" and "value" are present in the parameter set table.
- **Target row:** The target rows of the parameters to be imported are found on the basis of the names of the corresponding elements. (See "[Mapping for import and export](#)"⁵⁵¹)
- If only one row exists in a target table (e.g. experiment description), then one row is always imported from the parameter set table.

 Note**Language abbreviation**

The csv-files are designed to be language-independent. No in-house designations are used, which makes them easily readable. A language-indicating abbreviation is automatically appended to the file name. Thus, upon import it is possible to recognize for which language the file was generated. Additionally, a parameter set can also be imported from a different language (as long as the source language is installed).

Language recognition: If the abbreviation is not present, the language with the most matching titles is used as the source language.

Special case: There is no language abbreviation, but there are multiple parallel, valid files.

If when importing a file is specified only with the ending ".csv" and without any language-indicator abbreviation, then the system automatically searches for valid files having an abbreviation (Filename.Language.csv). If there is any corresponding file, it will be imported.

If multiple such files exist (varying languages), the file in the current language is used preferentially. If it is not available, the first one found is imported. The command generates an entry in the logbook on which file was imported.

Mapping for import and export

Mapping: Import

Here you can select the parameters according to which to import. At least one parameter must be used for assignment purposes to ensure that the parameters are assigned to the correct channel.

Selection	Description
Import mapping by name	For all parameters (all table descriptions), the settings are imported. The assignment is made on the according to the name. This means: All parameters belonging to a source-row (from the parameter set) are imported to a row in the target, if the parameter Name is identical in both the source and the target. Special case: If the parameter set contains the columns: Name and Channel name with different names: In this case, the column Name is still used for the assignment, but the column Channel name for setting the new name! Then it is possible to adapt the name by means of the name assignment. But this means that the parameter set can no longer be used since the previous name then no longer exists. By this method it would be possible, for example, to initialize the channel names. Then assign the new names of the parameters using a second parameter set.
Uniformly per module type	Set up all channels of a module type uniformly. The assignment is based on module type or Device order name, respectively, and channel type (e.g. "ARGFT/UTI-6-SUP", "Analog inputs"). Only one parameter set per module type is required.
Multiple-device operation (Firmware group A)	For all channels/variables, the settings are imported. The assignment is made according to the connector designation and the device serial number. When using multiple devices, this selection is preferable to "<i>Single-device operation</i>", since each connector designation is unique within a device. Disadvantage: it is not possible to assign parameters to other devices this way. If necessary, edit the parameters beforehand, in order to adapt the serial number to the target device. Cannot be used with devices of firmware group B because the connector does not distinguish between the main channel and the monitor channel.
Single-device operation (Firmware group A)	For all channels/variables, the settings are imported. The assignment is made according to the connector designation. When using multiple devices, the connector designation is not always unique; the same one could appear in every device. Use this setting only when using only one device. Cannot be used with devices of firmware group B because the connector does not distinguish between the main channel and the monitor channel.
...	Select a user-defined mapping. See " Mapping - Modifications of the default behavior ".

Assignment: Export

Here you can select what is to be exported.

Selection	Description
Export all channels	All variables having the channel type: " analog inputs " are exported.
Export all channels and settings	All content of the table description: " analog channels " is exported and all variables having the channel type: " analog inputs " are exported.
...	Select a user-defined mapping. See " Mapping - Modifications of the default behavior ".

Editing the parameter set

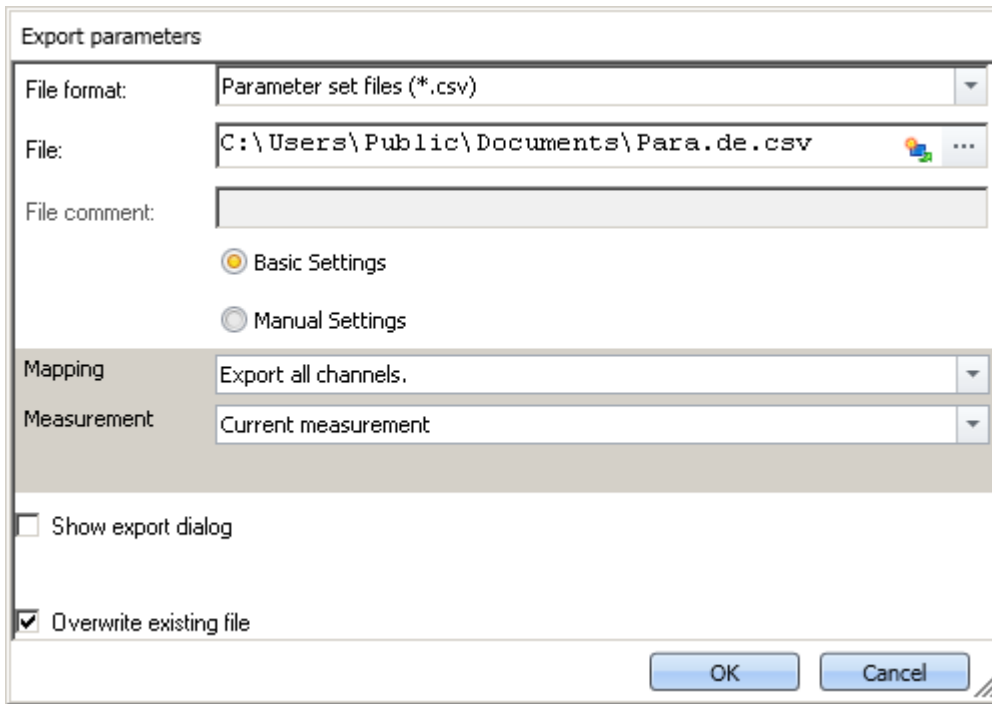
Program	Description
Editing by means of a text editor (Notepad, Notepad+, ...):	Here there are no known limitations.
Editing a csv-file with EXCEL	Load the file into EXCEL by double-clicking or by means of EXCEL's menu. If EXCEL's text conversion wizard appears, select "tab stop" as the separator. When you save the file after having made changes in EXCEL, this message appears: <i>"The file may contain features which are not compatible with Unicode text. Do you want to keep the workbook in this format?"</i> • Confirm with "Yes" When you close EXCEL, another message appears: <i>"Do you wish to save changes?"</i> • Select "Don't save" The following formats are not appropriate since the automatically save with the separator character ";": • CSV (separator character) (*.csv) • CSV (Macintosh) (*.csv) • CSV (MS-DOS) (*.csv) Known problems: Unless different settings have been made and no accompanying unit has been specified, EXCEL automatically interprets numbers with the decimal separator: <i>Decimal period</i> as a date and converts the accordingly. In this case, set the formatting of the cell to "Text". Or use the <i>Decimal comma</i> as the separator character with making entries.
Editing a csv-file by means of OpenOffice or LibreOffice	Load the file by double-clicking or via the program's menu. If the Text Import Assistant appears, select as the separator character/separator option: "Tabulator". If you save the file after having made changes, a message appears asking for the data saving format. • Click on "Keep current format" Known problems: • The character ' ' is used as an internal character in the editing process and it is deleted if it is at the beginning of a cell. This character otherwise has significance, for instance, as the unit for "feet" or "minutes of angle". • Unless different settings have been made and no accompanying unit has been specified, numbers with the decimal separator: <i>Decimal period</i> are automatically interpreted as a date and converted accordingly. In this case, set the formatting of the cell to "Text". Or use the <i>Decimal comma</i> as the separator character with making entries.

10.3.3.1 Export parameters

A parameter set is a group of parameters which can be used for various purposes. For instance, parameters may be **settings for the device and channel configuration**. They can also be **metadata** and **variables values**.

The command: "*Export parameters*" allows automated export of the parameters at desired times and according to previously adjusted settings.

There are pre-made export-configurations available such as "Export all channels" (the values of the active channels' variables) or "Export all channels and settings" (the values of the active channels' variables, and channel settings of all analog channels). Or you can manually select which variable values and settings to export.



Command: *Export parameters*



Note

Import of variables preferably via the command: **Export variable**

For **exporting variables' values**, instead of the command: "*Export parameter*", the command "**Export variable**" should be used. This command is specially designed for variable values.

Certain variable types can only be exported by means of the command "*Export variables*"; this included text-variables.

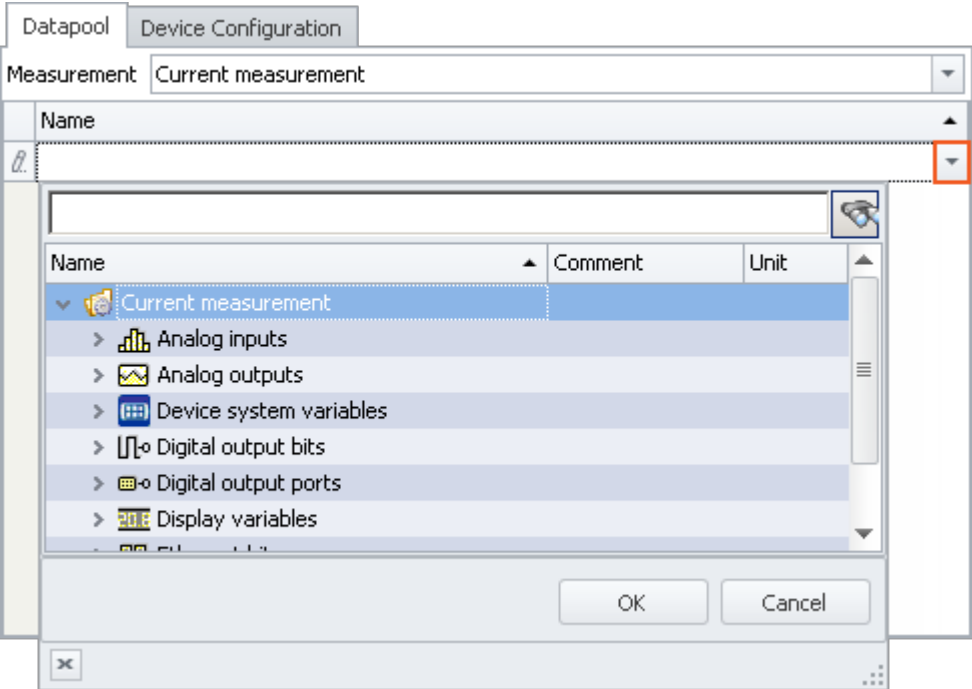
Option	Description
File format	Selection of the export format. For more information, see the description of the parameter set (File formats ⁵⁴⁷).
File	Specifies the target file including the pathname.
Basic/Manual Settings	In order to configure the export, you can choose between basic and manual settings. • Basic Settings⁵⁵⁵: Here you find prepared export settings, for instance, one where only all values of the analog channels are exported. • Manual Settings⁵⁵⁶: Here you can define exactly what to export.
Measurement	You can export either parameters belonging to the current measurement, or those of a different measurement (identified by the measurement name or the measurement number).
Overwrite existing file	If this option is activated, files having the same name in the destination folder will be overwritten without a confirmation prompt. This option is useful for automated routines.
Show export dialog and Show element list	If one of the two options is activated, then when executing the commands the dialog appears in which to make settings in these two areas: • Show export dialog: File options (Save as dialog) • Show element list: Datapool (variables) and device configuration. All settings can be inspected and edited. Any changes have no effect on the configuration of the command in the command list, but only on the current export.

Basic Settings

Option	Description
Mapping	Here you can select what to export. • Export all channels: All variables having the channel type: "analog inputs" are exported. • Export all channels and settings: All content of the table description: "analog channels" are exported and all variables of the channel type: "analog inputs" are exported. • ...: Select a user-defined assignment. For more information, see the description of the parameter set (Mapping for import and export ⁵⁵¹).

Manual Settings

Tab	Description
Datapool	Selection of the variables to be saved. Here you create a list of the variables to be exported. At runtime, these are exported automatically. The list is not dynamic, it does not adapt when variables are added or deleted. In some cases, the variables do not exist yet when the list needs to be filled. In order to add a variable to the next empty line: • enter the corresponding name, or • click on the button () to the right of the line. When you click on the button, a strip-down Data Browser with a list of all available variables appears. If needed, run a "Process configuration ¹⁴²" () to update the list of variables it contains.



List of the variables to export

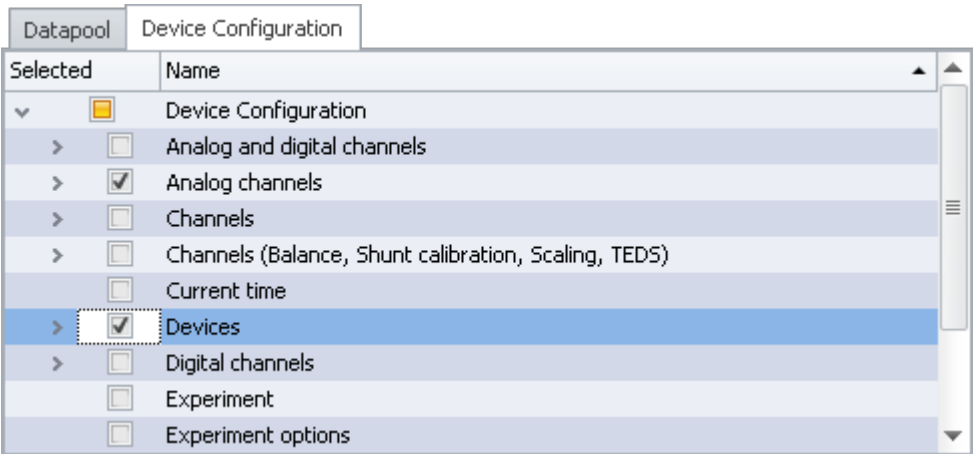
Select all desired variables (multi-selection is also allowed) and click on the button *OK*. All selected variables are added to the list of variables to export.

Tab	Description

Device Configuration

Selection of the device and channel configurations to be saved.

Here, select the settings to be exported.



List of device settings

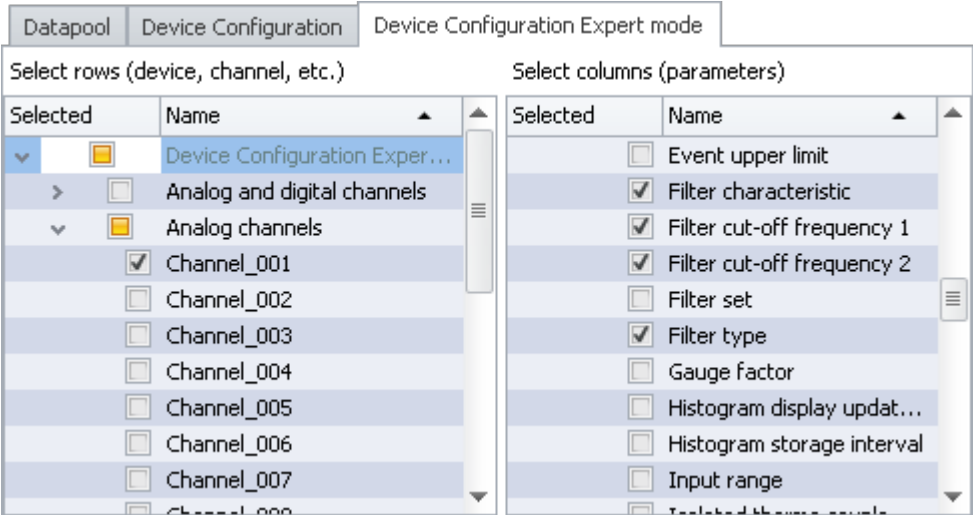
The individual table descriptions are available for selection. These define which parameters are exported. To limit the export to individual variables, devices or other elements, you can expand the table descriptions and select the desired elements.

Device Configuration Expert mode

This tab is only visible if [expert mode for export of parameter sets](#) was activated in the options.

Here, you can select individual parameters for export.

- In the region on the left, select the rows of a table description: the element (e.g. variable or device) whose parameters are to be exported.
- In the region at right, select the columns of the associated table description: the parameters which are to be exported.



Left: List of the rows in a table description (devices and variables)
Right: List of the columns of a table description (parameters)

In the example, the parameters "Filter characteristic", "Filter cut-off frequency 1", "Filter cut-off frequency 2" and "Filter type" belonging to the variable: "Channel_001" are exported.

**Note****Keeping variables' values and the device configuration separate**

You are able to export variables' values and device configurations together to one file. You are also able to combine the selection in the individual tabs. You can also combine "Device configuration" and the expert mode.

But please note: You will have greater clarity if you keep these things separate. To minimize error sources and to find errors more easily, we recommend exporting them separately.

Important parameters are always exported

Certain important parameters, which are necessary for assignment purposes when importing, are always included in the export process. Even if they are not selected, the following parameters are always included in the export of channel parameter:

Device name, Device identifier, Channel name, Name.

Trigger-settings and Application module assignments

Trigger-settings and Application module assignments can not be exported as parameter sets in a useful way.

Exporting balance values (taring, scaling, bridge, ...)

The balancing process is dependent on various parameters such as the amplifier type, the measurement range, measurement mode and 16 or 24 bit resolution. In order for it to be possible to re-import the balance value, these values need to be compatible. For this reason, when exporting balance values, many additional parameters are exported along with them.

**Note**

All necessary balance values are recorded in the file in xml-notation.

The balance values are associated with certain channel settings and only apply if these settings are previously restored when importing. E.g., when changing the measurement range, a new balancing procedure must be performed.

The record of the experiment includes the respective balance values saved for the various settings. For this reason, the balance values for each measurement range, for example, are exported to the same file.

**Example**

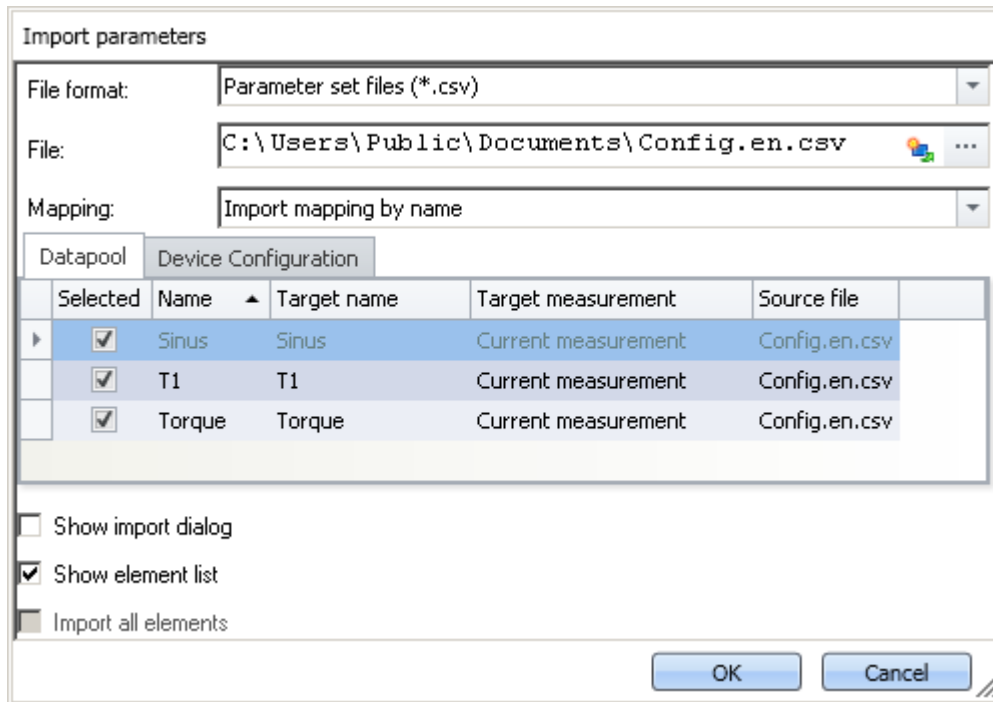
You perform balancing for Channel_001 with the measurement range 10 V. Next, you perform another balancing for the 5 V measurement range and export the result also to the file Balance.en.csv; the file contains both balancing values.

Upon importing, the appropriate balancing settings are imported in accordance with the measurement range set.

10.3.3.2 Import parameters

A parameter set is a group of parameters which can be used for various purposes. For instance, parameters may be **settings for the device and channel configuration**. They can also be **metadata** and **variables values**.

The command: "*Import parameters*" allows automated import of the parameters at desired times and according to previously adjusted settings. You can also select from which file to import which variable's values and/or settings.



Command: *Import parameters*



Note

Import of variables preferably via the command: **Load/Refill variable**

For **importing variables' values**, instead of the command: "*Import parameter*", the command "**Load/Refill variable**"^[995] should be used. This command is specially designed for variable values.

Using the command: "*Import parameter*" can cause unintended alteration of the device control.



Reference

Description of the parameter set

Basic information on working with a parameter set, as well as about its content and structure, is presented in the section: "**Parameter set**"^[546].

Option	Description
File format	Selection of the file format. For more information, see the description of the parameter set (File formats) ^[547] .
File	Specification of the file to import, including the pathname.

Option	Description
Mapping	Here you can select according to which parameters to perform the import. At least one parameter must be used for assignment purposes in order to ensure that the parameters are also assigned to the desired channel. An unsuitable assignment can cause either no parameters or incorrect ones to be imported! For more information, see further below under: "Mapping for import".
Show export dialog and Show element list	If one of the two options is activated, then when executing the commands the dialog appears in which to make settings in these two areas: • Show export dialog: File options (open-dialog) • Show element list: Datapool (variables) and device configuration. All settings can be inspected and edited. Any changes have no effect on the configuration of the command in the command list, but only on the current import.
Import all elements	This option is only available if Show element list is not selected. When it is activated, then the complete parameter set, meaning the datapool and the device configuration is always imported according to the assignment instruction. The advantage is that at the moment you create the command, you don't need to know what elements the parameter set contains. Note: If the box is checked, then the system no longer distinguishes between the device configuration and the datapool. If only the datapool is exported (e.g. for Channel_001), then when importing, the channel configuration is also imported. E.g. in this case, the unit is also set.

Tab	Description																																		
Datapool	Selection of the variables to be saved. Here you create a list of the variables to be imported. At runtime, these are imported automatically. The list is not dynamic, it does not adapt when variables are added or deleted. <table><tr><td colspan="2">Datapool</td><td colspan="4">Device Configuration</td></tr><tr><td></td><td>Selected</td><td>Name ▲</td><td>Target name</td><td>Target measurement</td><td>Source file</td><td></td></tr><tr><td>▶</td><td>☒</td><td>Sinus</td><td>Sinus</td><td>Current measurement</td><td>Config.en.csv</td><td></td></tr><tr><td></td><td>☒</td><td>T1</td><td>T1</td><td>Current measurement</td><td>Config.en.csv</td><td></td></tr><tr><td></td><td>☒</td><td>Torque</td><td>Torque</td><td>Current measurement</td><td>Config.en.csv</td><td></td></tr></table> <i>List of variables to import</i>	Datapool		Device Configuration					Selected	Name ▲	Target name	Target measurement	Source file		▶	☒	Sinus	Sinus	Current measurement	Config.en.csv			☒	T1	T1	Current measurement	Config.en.csv			☒	Torque	Torque	Current measurement	Config.en.csv	
Datapool		Device Configuration																																	
	Selected	Name ▲	Target name	Target measurement	Source file																														
▶	☒	Sinus	Sinus	Current measurement	Config.en.csv																														
	☒	T1	T1	Current measurement	Config.en.csv																														
	☒	Torque	Torque	Current measurement	Config.en.csv																														

Tab	Description

Device Configuration

Selection of the device- and channel configurations to be imported.

Here you select which configurations are to be imported from the file.

Datapool		Device Configuration
Selected	Name	Source file
☒	Device Configuration	
☒	Analog channels, Channels (Balance, Shunt cal...	Config.en.csv
☒	Sinus	Config.en.csv
☒	T1	Config.en.csv
☒	Torque	Config.en.csv
☒	Digital channels	Config.en.csv
☒	Sinus	Config.en.csv
☒	T1	Config.en.csv
☒	Torque	Config.en.csv

List of device settings

Some table descriptions are already available for selection. The displayed table descriptions do not have to match the original table description. The file does not state from which table description the export came.

Example: In the picture above, you see the table descriptions: "Analog channels, Channels (Balance, Shunt calibration, Scaling, TEDS)", and "Digital channels". These were determined from the parameters which the file contained.

However, the parameter set was generated from the table description "Analog channels". Since the table descriptions mostly contain the same parameters, these two are the first ones found which cover the import. If you ensure that all parameters are present on the target computer, then a correct table description will always be selected for import.

To limit import to individual variables, devices or other elements, you can expand the table descriptions and select the desired elements.

 **Note**

Importing a configuration

Make certain that the configuration can be imported.

Often certain settings depend on other parameters. For example, the measurement range depends on the offset/factor. When the measurement range is imported, then either the appropriate offset/factor-values must also be imported with it, or be set already.

Many other such interdependencies exist.

If a parameter can not be imported, a corresponding message appears in the logbook.

 **Note**

Importing to the Datapool / Variables

It is not possible to import to the **device channels** belonging to the current measurement (Data Browser: *Current measurement*). This applies to analog channels, incremental counter channels, ...

This does not apply to single value variables, such as Display-variables, DACs, ... It is possible to import to these.

Imprecisions in importing

Import of the measurement range

Due to imprecision from rounding, the measurement is not adopted correctly in some cases.

When there is no match, the next larger measurement range is selected which completely covers the range required. To handle rounding problems, a tolerance of 1% is provided. If a smaller measurement range is below the desired range by only 1%, it is selected.



Example

Possible measurement ranges	0.1 and 0.25	0.75 .. 1.25 and 0.5 .. 1.5
Measurement range in the import file	0.12	0.75 .. 1.263
Import generated	0.25	0.5 .. 1.5
	because it is the next larger range	because it is the next larger range

In case of rounding problems

Possible measurement ranges	0.1 and 0.25	0.75 .. 1.25 and 0.5 .. 1.5
Measurement range in the import file	0.10000001 to 0.101	0.75 .. 1.2500001 bis 0.75 .. 1.262
Import generated	0.1	0.75 .. 1.25
	because this deviates from a measurement range by 1% or less	because this deviates from a measurement range by 1% or less

Importing balance values (taring, scaling, bridge, ...)

The balancing process is dependent on various parameters such as the measurement range and measurement mode. In order for it to be possible to re-import the balance value, these values need to be compatible. For this reason, when exporting balance values, many additional parameters are exported along with them.



Warning

Bridge balance

When you import bridge balancing values for a channel whose amplifier type or device model is different (e.g. CRONOScompact with 16-bit Integer resolution to CRONOSflex with 24-bit Float resolution), the resulting measurement values are implausible! By contrast, balancing values from taring only correct the measured value and for this reason they can be imported without any problems.

Mapping for import

Here you can select according to which parameters to perform the import. At least one parameter must be used for assignment purposes, in order to ensure that the parameters are actually assigned to the desired channel.

Selection	Description
Import mapping by name	For all parameters (all table descriptions), the settings are imported. The assignment is made on the according to the name. This means: All parameters belonging to a source-row (from the parameter set) are imported to a row in the target, if the parameter Name is identical in both the source and the target. Special case: If the parameter set contains the columns: Name and Channel name with different names: In this case, the column Name is still used for the assignment, but the column Channel name for setting the new name! Then it is possible to adapt the name by means of the name assignment. But this means that the parameter set can no longer be used since the previous name then no longer exists. By this method it would be possible, for example, to initialize the channel names. Then assign the new names of the parameters using a second parameter set.
Uniformly per module type	Set up all channels of a module type uniformly. The assignment is based on module type or Device order name, respectively, and channel type (e.g. "ARGFT/UTI-6-SUP", "Analog inputs"). Only one parameter set per module type is required.
Multiple-device operation (Firmware group A)	For all channels/variables, the settings are imported. The assignment is made according to the connector designation and the device serial number. When using multiple devices, this selection is preferable to "<i>Single-device operation</i>", since each connector designation is unique within a device. Disadvantage: it is not possible to assign parameters to other devices this way. If necessary, edit the parameters beforehand, in order to adapt the serial number to the target device. Cannot be used with devices of firmware group B because the connector does not distinguish between the main channel and the monitor channel.
Single-device operation (Firmware group A)	For all channels/variables, the settings are imported. The assignment is made according to the connector designation. When using multiple devices, the connector designation is not always unique; the same one could appear in every device. Use this setting only when using only one device. Cannot be used with devices of firmware group B because the connector does not distinguish between the main channel and the monitor channel.
...	Select a user-defined mapping. See " Mapping - Modifications of the default behavior ".



FAQ

Question: Why do my changes to the parameter set have no effect following the import?

Answer: There are multiple possible reasons:

- An unsuitable assignment is used.
For example, the name assignment is used, but the names in the parameter set file do not match the names in the channel list. This can be the case if the name is to be modified by via the parameter set.
Solution: Use instead, for instance, import according to the connector designation, or observe the notes on the mapping by name (see: "[Mapping for import](#)"⁵⁶³).
- The settings are reset by other parameters.
For instance, if you export the parameter set of the page: *Analog channels*, you obtain among others the parameters for the analog channels and analog Monitor-channels. The parameters of these two channel types are interrelated. For example, if you change the coupling of the analog channel in the parameter set file from "Full bridge" to "DC", but leave the Monitor channel unchanged, the importing the Monitor channel will overwrite the analog channel.
Solution: Always minimize the parameter set to the most important parameters. Delete everything which is not needed.

10.3.3.3 Mapping - Modifications of the default behavior

Note

Advanced notes

This chapter provides advanced information on the import- and export mechanism. In most cases this chapter can be skipped.



Warning

The following description is of a modification of the default import and export behavior. For most applications, the existing mapping files are adequate.

Please don't change the existing mapping files, but rather make new assignments, at most.

The **mapping** determines how import and export are governed, for instance, according to what pattern data are imported. In this chapter, you will find some mapping examples. According to this pattern, you can define your own mapping files to design the import and export to corresponds to your requirements.

Each assignment defined will be saved in an XML file.

Pre-made mapping files for the parameter set commands are located following installation (depending on the installation and possibly the version) under the this path: (hot key <win> +r)

shell:common appdata\imc\imc STUDIO\Applications_1\Extensions\Parameterset

The assignment files used are located in the respective import or export folder. **Please do not change the original files!** If you save your own files there, they will be available for import/export.

Scenarios for import

If only certain rows of a table are to be imported, or the parameters are to be imported according to criteria other than the name.

Scenarios for export

For instance, if all channels are to be exported, they must be selected and their names known. However, as soon as the configuration is changed and new channels are added, the export list must be modified. With the help of the export mapping rules, it is possible to export all channels no matter how many and what names are used, or conversely to export only variables belonging to one type.

Defining a mapping rule

This mapping rule is saved in an XML file with the extension ".mapping.xml". **Be aware that the file must be UTF-8 encoded.** In every case, the language for which the mapping rule is used must also be specified.

In such a file it is possible to specify any desired mapping rules, and a name and a description for the mapping rule.



Example

Introductory example of import, with explanations

```
<import>
  <languageblock language="en">
    <caption>
      Here, a designation can be specified
    </caption>
    <description>
      Here, an explanation of this mapping rule can be supplied.
    </description>

    <mapping>
      <source>
        Here is where you define which rows from which table of the parameter set
        are to be imported.
      </source>
      <target>
        Here is where you define on which rows and columns the source row data
        are to be mapped.
      </target>
    </mapping>
  </languageblock>
</import>
```

Languages in the mapping rule

In order for assignment to be possible, the language of the file to import and of the software in the mapping file must be available. English should be used as the fallback language in case some parameters in the currently used software language are not available.

For each language, create a separate **languageblock**-block.



Example

languageblock-block

```
<import>
  <languageblock language="de">
    <caption>
      ...
    </caption>
    ...
  </languageblock>
  <languageblock language="en">
    ...
  </languageblock>
  <languageblock language="fr">
    ...
  </languageblock>
  <languageblock language="ja">
    ...
  </languageblock>
  <languageblock language="zh-CN">
    ...
  </languageblock>
  <languageblock language="zh-TW">
    ...
  </languageblock>
  ...
</import>
```

Syntax-examples for the import:

Syntax <source>	Description
<code>SELECT * FROM Channels</code>	All rows are used for import from the source file which can be assigned to the table description Channels . Device parameters and parameters of other table descriptions are not imported.
<code>SELECT *</code>	All rows are used for the import from the source file.
<code>SELECT * FROM Channels WHERE Name = 'My_TemplateChannel_Bridge'</code>	From the source file, all rows are used for the import 1. which were able to be assigned to the table description Channels, 2. and have My_TemplateChannel_Bridge for the parameter: Name.
Syntax <target>	Description
<code>SELECT * WHERE Name = source.Name</code>	All parameters of a source row are imported to a row in the target, if the parameter Name is identical in the source and in the target.
<code>SELECT * WHERE Connector = source.Connector</code>	All parameters of a source row are imported to a row in the target, if the parameter Connector is identical in the source and in the target.
<code>SELECT * WHERE Connector = source.Connector AND Device SN = source.Device SN</code>	All parameters of a source row are imported to a row in the target, 1. if the parameter Connector is identical in the source and in the target and 2. if the parameter Device SN is identical in the source and in the target.
<code>SELECT *\Name,Channel name WHERE Coupling = 'Full bridge'</code>	All parameters of a source row are imported to a row in the target if the parameter Coupling is Full bridge in the target. The parameters Name and Channel name are ignored and not set in the target.

**Example****Import 1: Channels - Import mapping by name**

All rows are used for import from the source file which can be assigned to the table description **Channels**. Each row in the parameter set is mapped to a row in the target table having the same **Name** as the source row.

```
<import>
  <languageblock language="en">
    <caption>
      Channels - Import mapping by name
    </caption>
    <description>
      The channels of the channel table are ordered by their names
    </description>

    <mapping>
      <source>
        SELECT * FROM Channels
      </source>
      <target>
        SELECT * WHERE Name = source.Name
      </target>
    </mapping>
  </languageblock>
</import>
```

**Example****Import 2: Import mapping by name**

Similar to Example 1.

From the source file, all rows are used for the import, not only for the table description: **Channels**. Each row in the parameter set is mapped to a row in the target table having the same **Name** as the target row.

```
<import>
  <languageblock language="en">
    <caption>
      Import mapping by name
    </caption>
    <description>
      All parameters are assigned according to their name.
    </description>

    <mapping>
      <source>
        SELECT *
      </source>
      <target>
        SELECT * WHERE Name = source.Name
      </target>
    </mapping>
  </languageblock>
</import>
```

**Example****Import 3: Single-device operation (Firmware group A)**

File: ConnectorAssignment_one_Device.mapping.xml

From the source file, all rows are used for the import.

The settings for all channels are imported. The assignment is performed on the basis of the connector designation. Each row in the parameter set is mapped to a row in the target table having the same **connector designation** as the source row.

```
<import>
  <languageblock language="en">
    <caption>
      Single-device operation (Firmware group A)
    </caption>
    <description>
      Import settings for the device selected for measurement, if the Connector
      designations do not include serial numbers.
      The assignment is based on the connector (e.g. "[02] IN01").
    </description>

    <mapping>
      <source>
        SELECT *
      </source>
      <target>
        SELECT * WHERE Connector = source.Connector
      </target>
    </mapping>
  </languageblock>
</import>
```



Cannot be used with devices of firmware group B because the connector does not distinguish between the main channel and the monitor channel.

**Example****Import 4: Multiple-device operation (Firmware group A)**

File: ConnectorAssignment.mapping.xml

All rows from the source file are used for the import.

The settings for all channels are imported. The assignment is performed on the basis of the connector designation and the device serial number. Each row in the parameter set is mapped to a row in the target table having the same **connector designation** and **serial number** as the source row.

```
<import>
  <languageblock language="en">
    <caption>
      Multiple-device operation (Firmware group A)
    </caption>
    <description>
      Import settings for particular devices, if the Connector designations do not
      include serial numbers.
      The assignment is based on connector and device serial number (e.g. "[02] IN01",
      "144163").
    </description>

    <mapping>
      <source>
        SELECT *
      </source>
      <target>
        SELECT * WHERE Connector = source.Connector AND Device SN = source.Device
      SN
      </target>
    </mapping>
  </languageblock>
</import>
```



Cannot be used with devices of firmware group B because the connector does not distinguish between the main channel and the monitor channel.

**Example****Import 5**

This mapping rule consists of two parts. First, the row of the parameter set which was imported for the channel table and which has the name "My_TemplateChannel_Bridge" is loaded. Next, this row is mapped to all rows in the target table for which "Full bridge" was set as the coupling.

In the second step, the row of the parameter set which was imported for the channel table and which has the name "My_TemplateChannel_DC" is loaded. Next, this row is mapped to all rows in the target table for which "DC" was set as the coupling.

In both cases, the name and the channel name are not set in the target.

```
<import>
  <languageblock language="en">
    <caption>
      Initializing analog channels
    </caption>
    <description>
      Creating a base configuration for all analog channels.
    </description>

    <mapping>
      <source>
        SELECT * FROM Channels WHERE Name = 'My_TemplateChannel_Bridge'
      </source>
      <target>
        SELECT *\Name,Channel name WHERE Coupling = 'Full bridge'
      </target>
    </mapping>

    <mapping>
      <source>
        SELECT * FROM Channels WHERE Name = 'My_TemplateChannel_DC'
      </source>
      <target>
        SELECT *\Name,Channel name WHERE Coupling = 'DC'
      </target>
    </mapping>
  </languageblock>
</import>
```


**Example****Sample files for example Import 5****Example: csv-file**

Name	Channel status	Measurement mode	Coupling	Input range
My_TemplateChannel_Bridge	Active	Voltage	Full bridge	±100 "mV/V"
My_TemplateChannel_DC	Active	Voltage	DC	±5 V
My_Channel_without_effect	Active	Voltage	Quarter bridge	±10 "mV/V"

Example: Channel-configuration prior to import

Name	Channel status	Measurement mode	Coupling	Input range
Channel_001	Passive	Voltage	DC	±1 V
Channel_002	Passive	Voltage	DC	±1 V
Channel_003	Passive	Voltage	DC	±1 V
Channel_004	Passive	Voltage	Quarter bridge	±1000 "mV/V"
Channel_005	Passive	Voltage	Quarter bridge	±1000 "mV/V"
Channel_006	Passive	Voltage	Full bridge	±1000 "mV/V"
Channel_007	Passive	Voltage	Full bridge	±1000 "mV/V"
Channel_008	Passive	Voltage	Full bridge	±1000 "mV/V"

Example: Channel-configuration after import

Name	Channel status	Measurement mode	Coupling	Input range
Channel_001	Active	Voltage	DC	±5 V
Channel_002	Active	Voltage	DC	±5 V
Channel_003	Active	Voltage	DC	±5 V
<i>Channel_004</i>	<i>Passive</i>	<i>Voltage</i>	<i>Quarter bridge</i>	<i>±1000 "mV/V"</i>
<i>Channel_005</i>	<i>Passive</i>	<i>Voltage</i>	<i>Quarter bridge</i>	<i>±1000 "mV/V"</i>
Channel_006	Active	Voltage	Full bridge	±100 "mV/V"
Channel_007	Active	Voltage	Full bridge	±100 "mV/V"
Channel_008	Active	Voltage	Full bridge	±100 "mV/V"

The names are ignored. The channels: "Channel_004" and "Channel_005" remain as they are, since the coupling "Quarter bridge" is not in the assignment rule for the target.

The source row "My_Channel_without_effect" is also not in the assignment rule and is not used as a source.

Syntax-examples for the export:

Syntax <source>	Description
<code>SELECT * WHERE Caption = 'analog inputs'</code>	All contents of the table description and/or all variables of the channel type: "analog inputs" are exported. In this case, there is no table description of the name "analog inputs". However, there are variables of the channel type "analog inputs"
<code>SELECT * WHERE Caption = 'analog inputs' OR Caption = 'analog channels'</code>	All contents of the table description and/or all variables of the channel type: "analog inputs" or "analog channels" are exported. In this case, there is a table description of the name "analog channels". And there are also variables of the channel type "analog inputs"
<code>SELECT * WHERE Caption = 'analog channels'</code>	All contents of the table description and/or all variables of the channel type: "analog channels" are exported. In this case, there is a table description of the name "analog channels". However, there is no variable of the channel type "analog channels".
Syntax <target>	Description
<code>SELECT *</code>	obligatory specification

 **Example**

Export 1: Export all channels

All variables of the channel type: **"Analog inputs"** are exported.

```
<import>  
  <languageblock language="en">  
    <caption>  
      Export all channels  
    </caption>  
    <description>  
      Export data from all active channels.  
    </description>  
  
    <mapping>  
      <source>  
        SELECT * WHERE Caption = 'analog inputs'  
      </source>  
      <target>  
        SELECT *  
      </target>  
    </mapping>  
  </languageblock>  
</import>
```

**Example****Export 2: Export all channels and settings**

All contents of the table description: "**analog channels**" are exported and all variables of the channel type: "**analog inputs**" are exported.

The only distinction made is between the names: "analog inputs" vs. "analog channels". The names must be unique if they are used for the export.

```
<import>
  <languageblock language="en">
    <caption>
      Export all channels and settings
    </caption>
    <description>
      Export data and settings from all active channels.
    </description>

    <mapping>
      <source>
        SELECT * WHERE Caption = 'analog channels' OR Caption = 'analog inputs'
      </source>
      <target>
        SELECT *
      </target>
    </mapping>
  </languageblock>
</import>
```

10.4 imc Data format

The imc data format (*imc3*) is applied and generated.

One major advantage of this format is the display of large data volumes in the curve window. It is not necessary to load large data volumes completely before they can be viewed. Instead, only so much data are loaded for the curve window as can also be displayed. In this way, you are able to quickly scroll and zoom, no matter how large the volume of data is.

Storage of the data is resilient to interruptions; even incomplete files can be used without the need for repairs.

Selecting the data format: imc2 or imc3

In imc FAMOS, it is possible to set whether the files are to be saved in the old imc2 or in the new imc3 format. imc STUDIO Monitor applies this setting accordingly. We recommend saving the measured data in imc3 format.

**Reference****The imc data formats**

Information on the imc data formats is presented in the imc FAMOS manual.

10.5 imc Format Converter

The imc Format Converter provides the ability to convert measured data to a different format, e.g. to EXCEL data format. The imc Format Converter is integrated in **imc STUDIO** within the functionality of **Sequencer's** [Data Saving Assistant](#)  and [Format Converter](#) .

Furthermore, the conversion capabilities of the Format Converter can be used either within a **stand-alone program** by means of the **Windows-Explorer's context menu**, if imc STUDIO or imc FAMOS has been installed.

Export formats

The same export formats are supported which are also offered by imc FAMOS. These include:

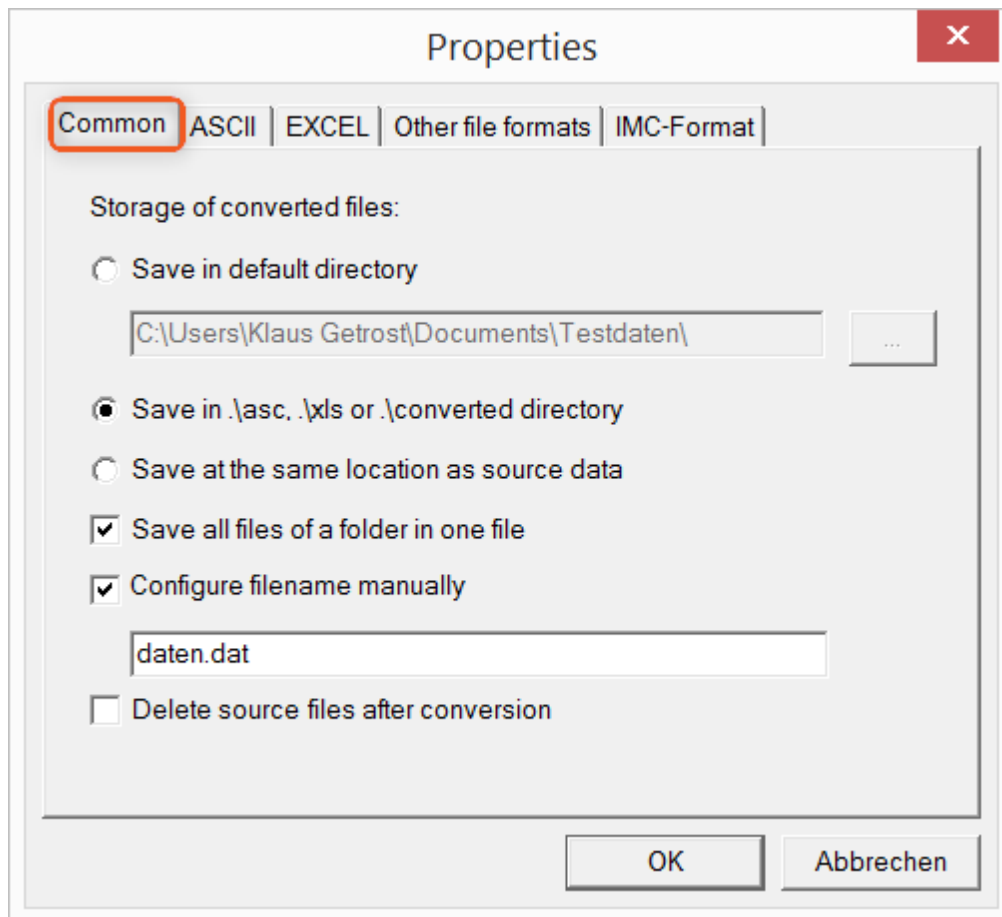
- user-defined ASCII export-formats
- user-defined EXCEL export-formats
- MDF2.0, MDF3.0, MDF4, ASAM ATFX, ASAM ATFX NVH, Catman 5.0
- Google Earth Export, HEAD acoustics (also 4.5 compatible)
- Matlab 4 and 5, nSoft-DAC, RPC-3, Somat SIF (nCode output)
- TEAC TAFFormat and TEDAM, DIAdem TDM and TDMS
- Binary Universal File Format
- Universal File Format (UFF)

10.5.1 Settings

In the settings you specify whether the data are saved either individually or together, which templates are used for ASCII and EXCEL, or what other formats are used.

The settings remain intact, so that under normal circumstances you can immediately select data and convert them.

Data saving



Storage location for converted files:

Save in default directory: Free choice of a **target folder**. Even network drives are possible.

Save in .\asc, .\xls oder .\converted directory: The data are saved in their source folder but additionally in a **subfolder**. For ASCII in the *asc* folder, *xls* for EXCEL and *converted* for any other export formats.

Save at same location as source data: The data are saved on the same **level of the tree diagram as the source data, without subfolders**.

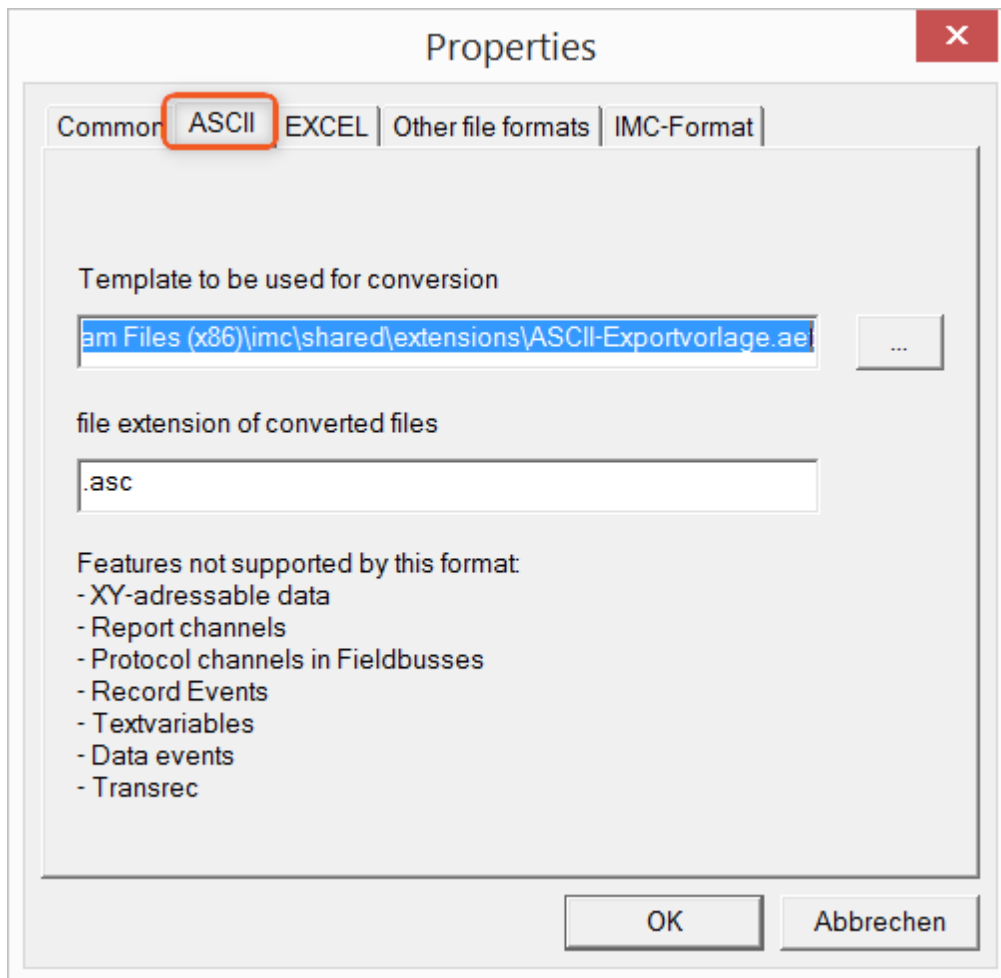
Manner of data saving:

Save all files of a folder in one file: All of a channel's files are saved **together in one file**.

Configure file name manually: **Name of the target file** in which all channel files are saved together.

Delete source files after conversion: Once the data have been converted, the **source files can be deleted automatically**, in order to make memory space free. This setting can make sense when all data are saved together in the imc format.

ASCII



Format template to be used for conversion:

By default, ASCII export templates are located in the folder "*C:\Program Files (x86)\imc\Shared\Extensions*" and use the file extension **.aet*. Owners of imc FAMOS are also able to personally create or modify these templates. In this case, the aet-files are saved in the folder "*C:\ProgramData\imc\Common\Def*".

By default, the file extension is "*asc*", but any arbitrary one can be specified here.

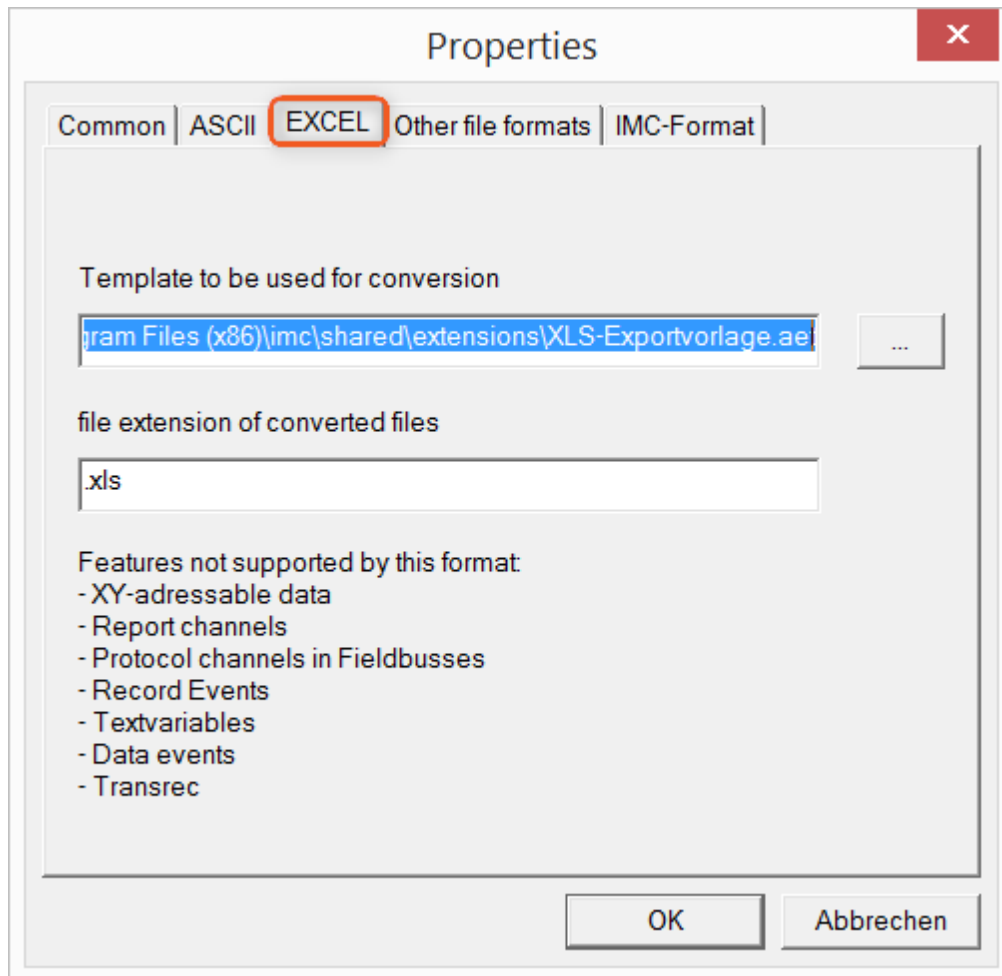
Special variable types which can **not** be exported as ASCII data are listed on this tab page.

! Notes

File extensions

In imc FAMOS the file extension is specified with the ASCII template. Within imc FAMOS this file extension is added to the ASCII file. With imc Format Converter the file extension specified here is used instead.

EXCEL



Format template to be used for conversion:

For EXCEL files, a similar technique is used as for [ASCII](#)⁵⁷⁷. By default, the file extension is "XLS", but can be arbitrarily specified here.

Variable types which are **not** exported correspond to those for [ASCII-export](#)⁵⁷⁷.

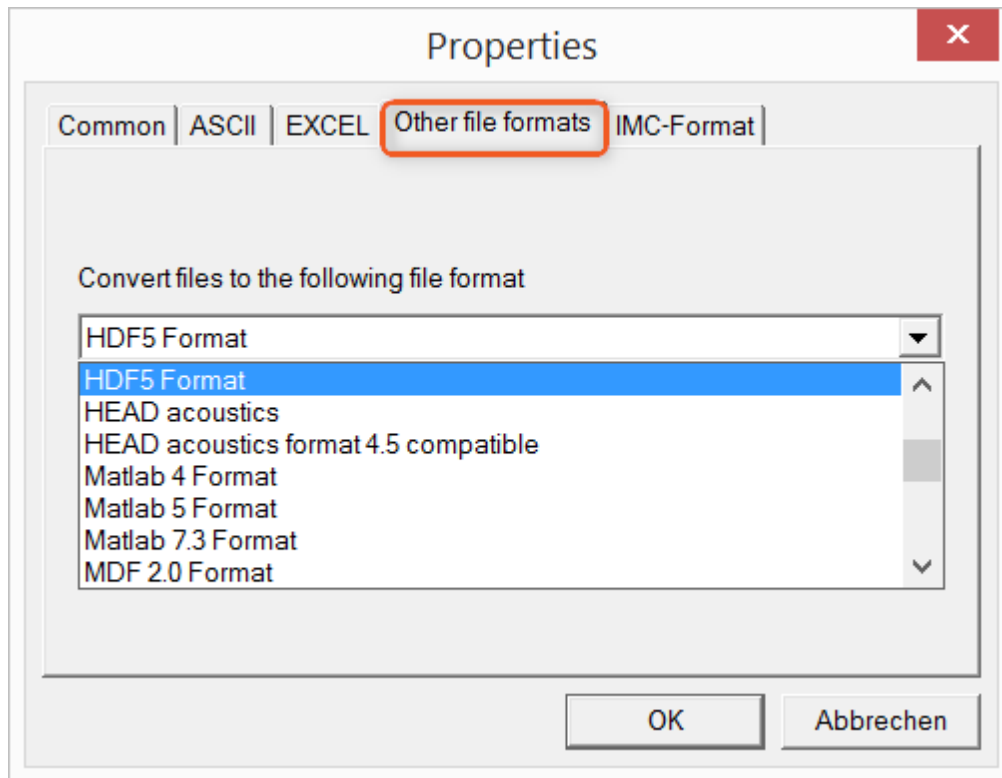


Notes

File extensions

If the XLSx format was selected as the export template, it must be here as the file extension. The imc Format Converter does not import the specified file extension from the AET file.

Other file formats



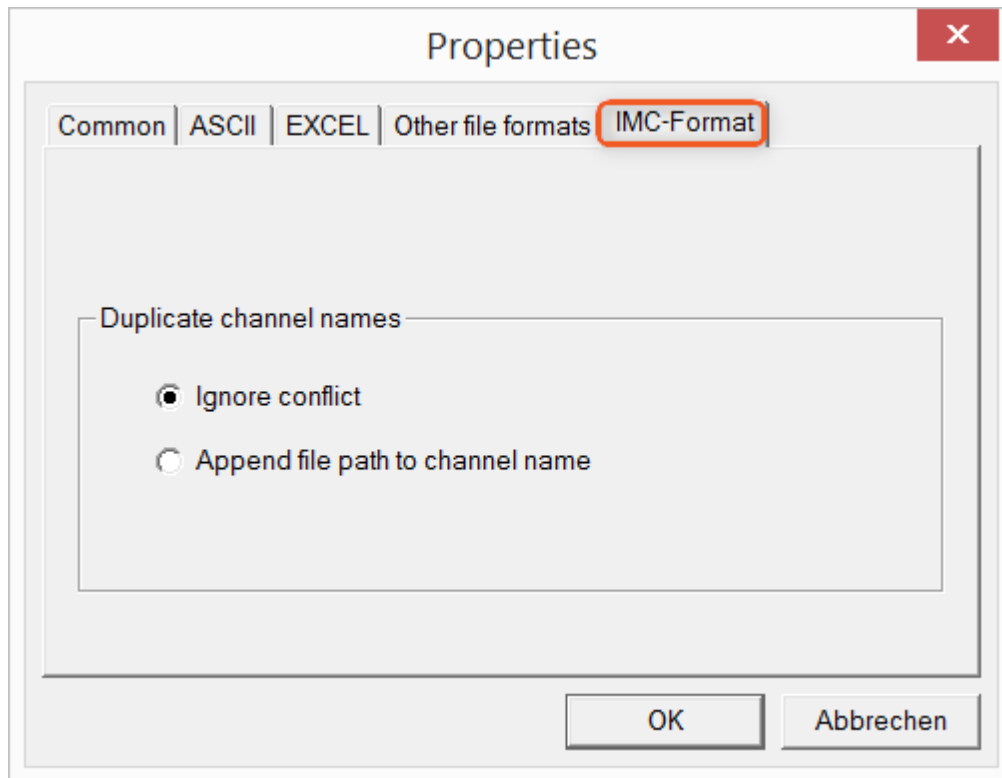
Convert files to the following file format:

Selection of the format in which the data are to be saved. Installation of the imc Format Converter makes certain export formats available.

Options

If the format selected comes with additional export options, these can be set by means of the dialog opened by clicking on the button "Options". Descriptions of the options appear in the document *ImportExportFilter.pdf*, of which a copy is imported along with the imc FAMOS installation. Alternatively, the document is available from the [imc homepage](#).

imc-Format



Handling duplicate channel names

Ignore conflict: If files are overwritten due to duplicate channel names, the conflict is not reported.

Append file path to channel name: A unique channel name is generated from the file path.

! Notes

Why save in the imc-format?

The imc Format Converter is designed for imc data, so why is there an imc-Format option? imc devices allow complex data structures in which channels can be recorded at different moments in time (using triggers). For this reason, the channels are saved as individual files. Using the imc Format Converter, you are able to join the data sets accumulated into one single file after the measurement. One appropriate group of settings on the page "[Common](#)" to accomplish this would be:

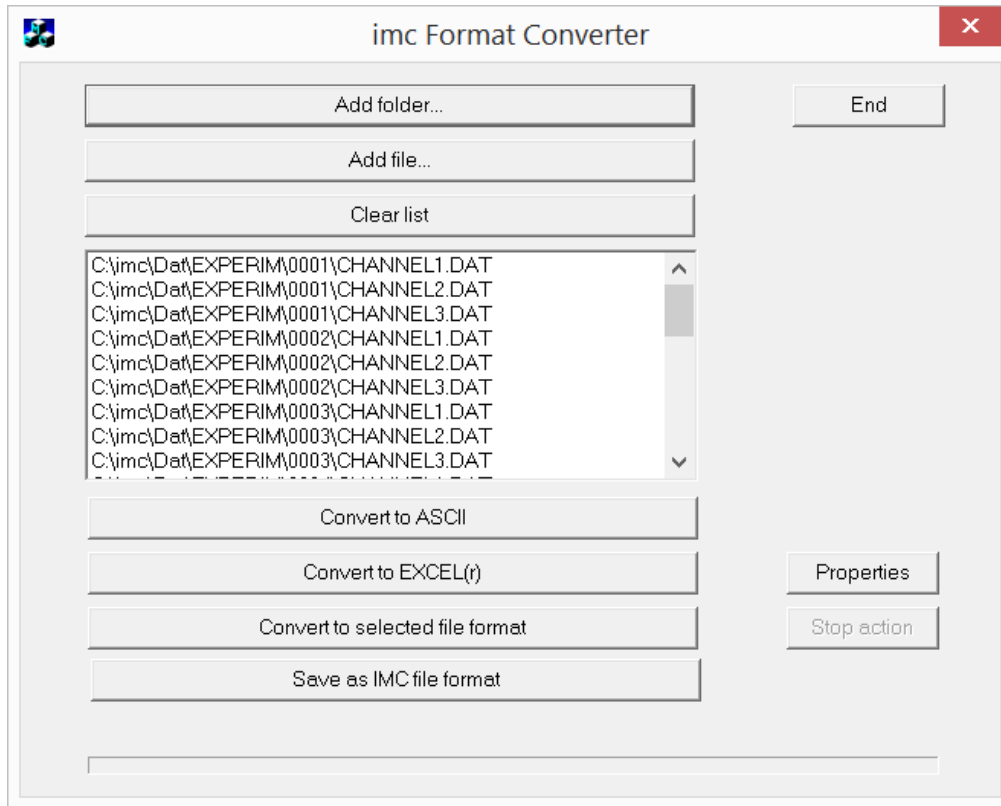
Save at same location as source data = active

Save all files of a folder in one file = active

Delete source files after conversion = active

10.5.2 Format converter as standalone-program

There is no shortcut for starting the standalone version. Start "*imcFrmtCvrt.exe*" directly from the installation folder, normally "*C:\Program Files (x86)\imc\imc Format Converter*"



Add folder...

Select the folder of measured data desired. Subfolders will also be imported and appear in the Files list.

Add file...

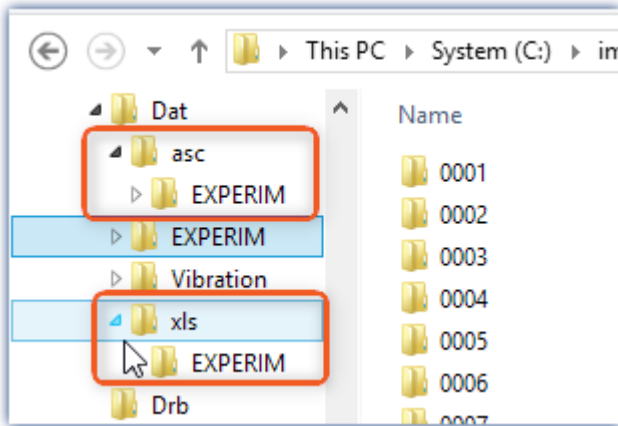
Selection of individual files.

Clear list

All entries are deleted from the Files list.

Convert to ASCII/EXCEL(r)

The standard variant for converting data to [ASCII](#)^[577] or [EXCEL](#)^[578]-files. Toward this end, a "asc" or "xls" folder is created on the same level as the folder of measured data, in which the data are saved in the same folder structure. Depending on the settings, either individual files or tables are generated, in which all data are saved in a file.



Convert to selected file format

The data are saved in the format which was selected under "[Settings](#)"^[579]. The target folder is a subfolder of the same folder, with the name "*converted*". This option allows all signals to be saved jointly in a single file. Furthermore, a filename with file extension must be entered in the [settings](#)^[576].

Save as IMC file format

For saving all files in a folder to a single file in imc format. Observe the necessary [settings on the Common tab](#)^[580].

10.5.3 Command line parameters

The imc Format Converter can be run via command lines. In the simplest case, the settings last applicable in the imc Format Converter are used, e.g. including the target file. If multiple files are to be converted, these are defined in an [XML-file](#)⁵⁸³.

Command line options

	Description	Example
-a	conversion to "ASCII"	imcFrmtCvrt.exe -a <files or paths>
-e	conversion to "EXCEL"	imcFrmtCvrt.exe -e <files or paths>
-k	conversion to "Other data formats"; The format selected in the settings is used, e.g. MDF	imcFrmtCvrt.exe -k <files or paths>
-i	import and use configuration from a setup XML file (imc Devices mode) enables complete remote control. Under some circumstance, a variety of settings can be required for the export. Beyond the default settings, it is possible to provide a Setup.xml by means of -s. A description of the XML is then no longer needed.	imcfrmtcvrt -i -s d:\setup.xml imcfrmtcvrt -i d:\setup.xml <files or paths>
-s	(Setup) without any additional parameters opens the familiar settings dialog	
-h	(hidden) only in conjunction with -i: Suppression of all windows and error messages	imcfrmtcvrt -i -h d:\setup.xml <files or paths>
-l	Write all information from windows and error messages to a LOG-file	
-h -l	Write all information from windows to a LOG-file in spite of suppression	imcfrmtcvrt -i -h -l d:\setup.xml d:\Logfile.log <files or paths>

For the purpose of determining **target files**, the imc Format Converter **settings** are used.

Specifying the files to convert in an XML file

You can specify in an XML file which files are to be converted. This can be necessary for instance when the command line is not sufficiently long. Depending on the Windows version, the line length is between 2000 and 8000 characters. The files to be converted are then are specified in an XML file.

**Example****Sample XML-file**

This XML file must have the following content:

```
<Converter_V1>
<DestinationPath>
  <SourceFile>d:\Test\data\e01 (08).RAW</SourceFile>
  <SourceFile>d:\Test\data\e02 (08).RAW</SourceFile>
</DestinationPath>
</Converter_V1>
```

Calling the file**Example 1:**

```
imcfrmtcvrt -a d:\data.xml
```

All files specified in the file: "*d:\daten.xml*" are converted to "ASCII" (-a)

Example 2:

```
imcfrmtcvrt -i -h -l d:\setup.xml d:\Logfile.log d:\Data.xml
```

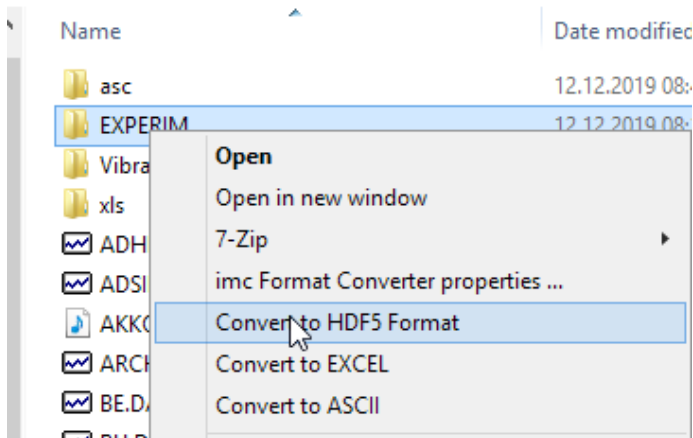
All files specified in the file: "*d:\daten.xml*" are converted.

The parameters for the conversion are defined in the file "*d:\setup.xml*" (-i).

All notifications are suppressed (-h) and saved in the log-file (-l): "*d:\Logfile.log*".

10.5.4 Conversion via the Windows Explorer

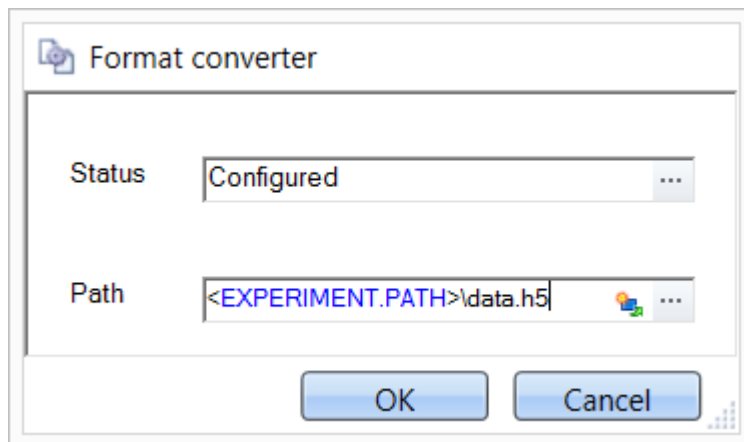
Installation of the imc Format Converter expands the Windows Explorer's context menu. The current templates for export of [ASCII](#)^[577] and [EXCEL](#)^[578], as well as the export format last selected on the page "[Other file formats](#)"^[579] can then be applied directly to a folder or files:



10.5.5 Format Converter as imc STUDIO command

Calling

Via the Sequencer: "External calls" > "Format converter":



Configuration of the command: Format Converter

Status

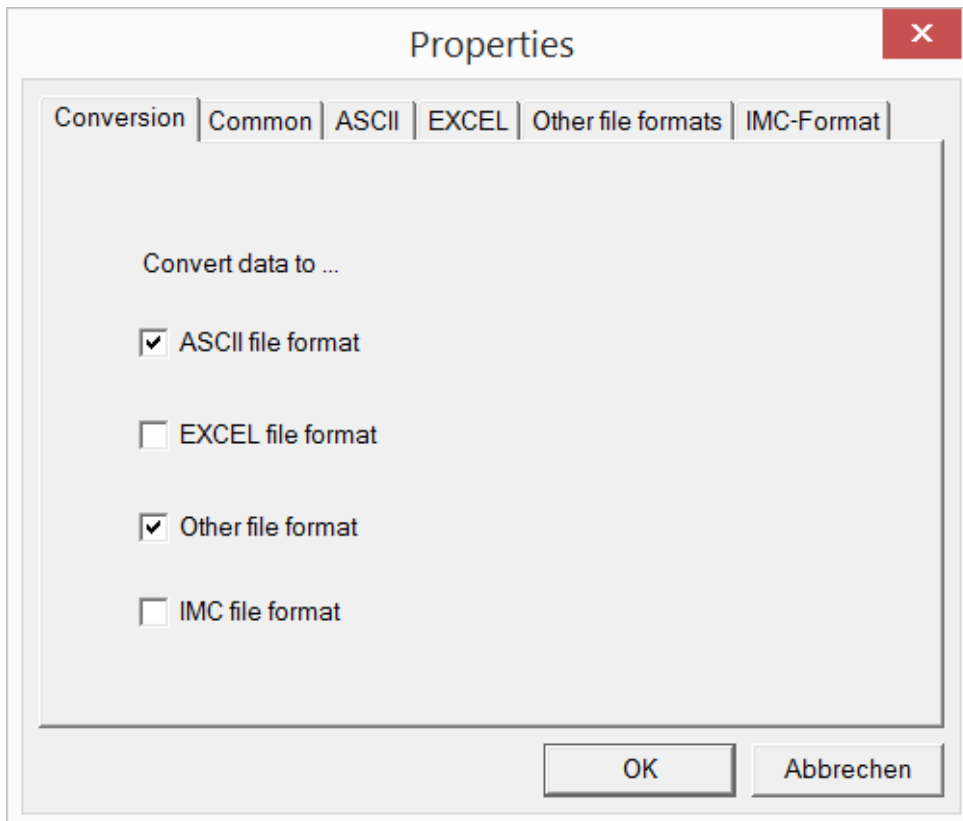
Immediately upon opening the command's configuration, the status is set to "not configured".

Path

- If just **one** file (*.dat, *.raw) of measured data is specified here, you can select whether to [convert either only the single file, or all files of measured data](#)⁵⁷⁶ which the **folder** contains.
- If a **folder** is specified here, then **all files of measured data in the folder** are converted:
 - If the option "Save all files of a folder in one file" among the settings on the tab page "[Common](#)"⁵⁷⁶ (see the segment on "Status") is selected, then **all files of measured data** present in the specified folder are converted to a **single target file**.
 - If the option is **not selected**, then each file of measured data which the folder contains, the data are converted to a **separate file**.

Using the button  in the box "Status" you can open a dialog via which you can make all the settings for the conversion:

- Conversion: Target format. It is also possible to select multiple formats.
- [Common](#)⁵⁷⁶: Storage location and whether individual files or a multi-file is to be created.
- [ASCII](#)⁵⁷⁷: Selection of the export template and specifying of the file extension.
- [EXCEL](#)⁵⁷⁸: Selection of the export template and specifying of the file extension xls orxlsx.
- [Other file format](#)⁵⁷⁹: Selection of the format and if necessary its formatting options.
- [IMC-file format](#)⁵⁸⁰: Save the data to a file in the imc format, or as a copy to a different memory location.



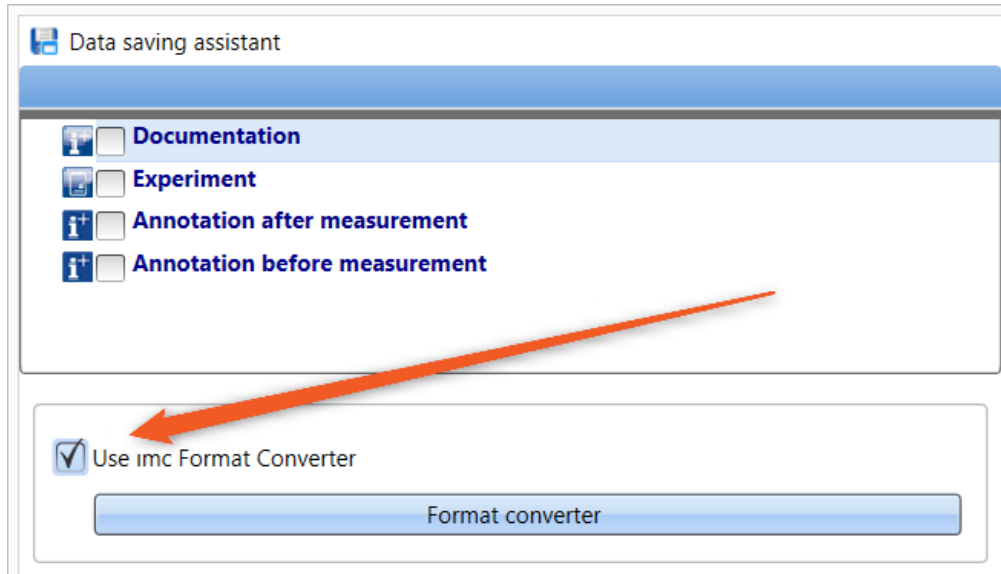
Setting the Format Converter

10.5.6 Converting via the Data saving assistant

Calling

Via the Sequencer: "Read/write data" > "Data saving assistant"

Along with supplemental information on the measurement, the "Data saving assistant" also allows the use of the imc Format Converter. Clicking on the button "Format Converter" opens its settings dialog, with which you can configure the imc Format Converter in accordance with the description of the [imc STUDIO-command](#) ⁵⁸⁵.

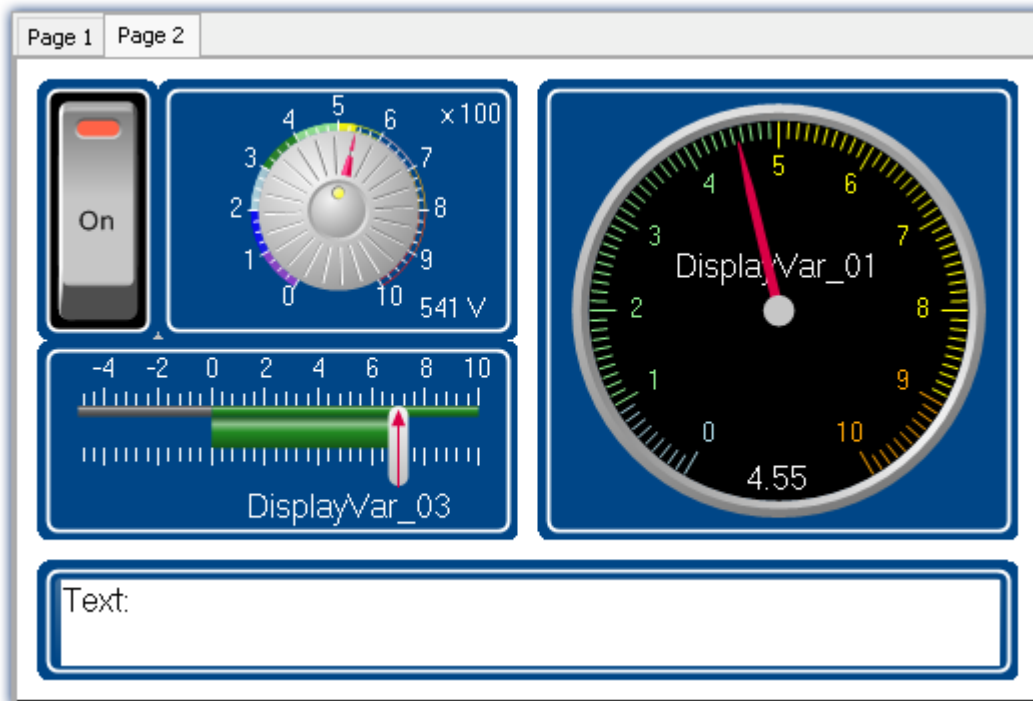


Format Converter as component of the Data Saving Assistant

11 Panel - Data visualization and controls

Panel is the imc STUDIO Monitor component with which **measured data and events** can both be **displayed** in instruments, and influenced using other controls (both controls and instruments are called Widgets).

With the Panel you can achieve **personalized user interfaces** (called [Panel pages](#) ^[924]), which are optimized for your specific needs.



imc STUDIO Panel with two pages (example)

Panel page

The Panel consists of one or more Panel pages. On a page, it is possible to layout various Widgets.

There are two kinds of Panel pages available:

- [Dialog pages](#) ^[924] optimized for on-screen display
- [Report pages](#) ^[924] optimized for printouts

Widgets

Various [Widgets](#) ^[596] are available as instruments and controls. The amount of Widgets varies by the product configuration. Even the [curve window](#) ^[647], familiar from other programs, can be placed on a page.

By means of the Widgets, you can affect the measurement interactively. It is also possible to assign [actions \(commands\)](#) ^[645] to Widgets such as switches and buttons.

Operation

For the [positioning of Widgets](#) ^[617] and the [assignment of channels and variables](#) ^[939], a variety of operation techniques can be used, such as Drag&Drop or context menus.

The Widgets can be operated by keyboard or mouse.

11.1 Ribbon

11.1.1 Control

Device control

Here you find the menu actions for device control.



For a detailed description, see the manual:
"Setup - Manual" > "Ribbon" > "[Control](#)".

View

Menu item	Description
 Panel Fullscreen	The Panel is displayed in fullscreen mode. The imc STUDIO user interface is hidden and can thus no longer be operated or accessed (excepting the Panel pages). See chapter " Fullscreen ".

Export

Menu item	Description
 As PDF (Export)	Saves the current Panel page in PDF format.
 As Graphics (Export)	Saves the current Panel page as a graphics file. You can select among the following graphics formats: • Portable Network Graphics (*.png) • Windows Bitmap (*.bmp) • JPEG (*.jpg)

Print

Menu item	Description
 Print Preview	Opens a print preview. Here you can see the possible appearance of printing results. This can be helpful when Widgets such as the curve window are used, which have their own color settings for the printout.
 Print	Opens the dialog for printing the Panel pages.

11.1.2 Navigation

Control

Menu item	Description
 Post-Processing	This option currently has no function.
 Find in Variables	This option currently has no function.
 To Beginning	This option currently has no function.
 Start	This option currently has no function.
 Stop	This option currently has no function.
 To End	This option currently has no function.
 Repeat	This option currently has no function.
 Faster	This option currently has no function.
 Slower	This option currently has no function.

Curve window

Menu item	Description
 Zoom	This option currently has no function.
 Absolute Time	This option currently has no function.
 Relative Time	This option currently has no function.
 Scroll	This option currently has no function.
 Stretch	This option currently has no function.
 Pause	This option currently has no function.

Design

Menu item	Description
 Design Mode	- Activated: the Panel page can be edited, the Widgets can not be operated. - Deactivated: The Panel page can not be edited. The Widgets can be operated.
 Navigation Bar	The navigation bar currently has no function.
 Selection	The navigation bar currently has no function.
 Variables	This option currently has no function.

Export

Menu item	Description
 As PDF (Export)	Saves the current Panel page in PDF format.
 As Graphics (Export)	Saves the current Panel page as a graphics file. You can select among the following graphics formats: • Portable Network Graphics (*.png) • Windows Bitmap (*.bmp) • JPEG (*.jpg)

Print

Menu item	Description
 Print Preview	Opens a print preview. Here you can see the possible appearance of printing results. This can be helpful when Widgets such as the curve window are used, which have their own color settings for the printout.
 Print	Opens the dialog for printing the Panel pages.

11.1.3 Design

New

Menu item	Description
 Insert Page	Selection window for creating a new Dialog- or Report-page is opened. (see: Insert - Dialog / Report ⁹²⁴⁾)
 Standard Dialog	Creates a new Standard dialog page.

Edit and Clipboard

Menu item	Description
 Design Mode	- Activated: the Panel page can be edited, the Widgets can not be operated. - Deactivated: The Panel page can not be edited. The Widgets can be operated.
 Lock/Unlock Page	Lock/Unlock the current page. Individual pages can be protected against changes (see Lock and unlock page ⁹³²⁾).
 Undo	Reverses the last change.
 Redo	Restores a previously reversed change.
 Delete	Deletes the selection (the page or the selected Widgets).
 Select all	Selects all of the page's Widgets.
 Cut	Clips out the selection or the selected Widgets and moves them to the Clipboard.
 Copy	Copies the selection or the selected Widgets to the Clipboard.
 Paste	Inserts the content of the Clipboard at the selected position.

Navigation

Menu item	Description
 Navigation Bar	The navigation bar currently has no function.
 Selection	The navigation bar currently has no function.
 Variables	This option currently has no function.
 Zoom	This option currently has no function.
 Absolute Time	This option currently has no function.
 Relative Time	This option currently has no function.
 Scroll	This option currently has no function.
 Stretch	This option currently has no function.
 Pause	This option currently has no function.

View

Menu item	Description
 Panel Fullscreen	The Panel is displayed in fullscreen mode. The imc STUDIO user interface is hidden and can thus no longer be operated or accessed (excepting the Panel pages). See chapter " Fullscreen ".

Im-/Export

Menu item	Description
 Import Page	This lets you import a saved page (file type: dbv). If the page's designation is identical to that of an already existing one, you are prompted whether to overwrite the existing page. If you answer "No", the imported page is appended along with a new page title (incremented number).
 As Panel Page (Export)	Saves the current Panel page as a file (file type "Panel pages (*.dbv)").
 As PDF (Export)	Saves the current Panel page in PDF format.
 As Graphics (Export)	Saves the current Panel page as a graphics file. You can select among the following graphics formats: • Portable Network Graphics (*.png) • Windows Bitmap (*.bmp) • JPEG (*.jpg)

11.2 Tool windows

The following tool windows are available for the purpose of configuring the Panel pages:

Tool Windows	Description
Widgets	Depending on the product configuration, a certain amount of display and control elements (Widgets) in various groups.
Data Browser	The tool window Data Browser shows the available data such as measurement channels, Display variables or pv-variables in a hierarchical tree diagram. With this tool window you can also set the variable linkage  for a Widget.
Properties 	The Properties window changes its contents in response to the object selected. If no Widget is selected, the page properties  are displayed.
Page templates 	Ti create new pages there are templates for dialogs (optimized for screen display) and reports (optimized for printout).

11.2.1 Properties (general)

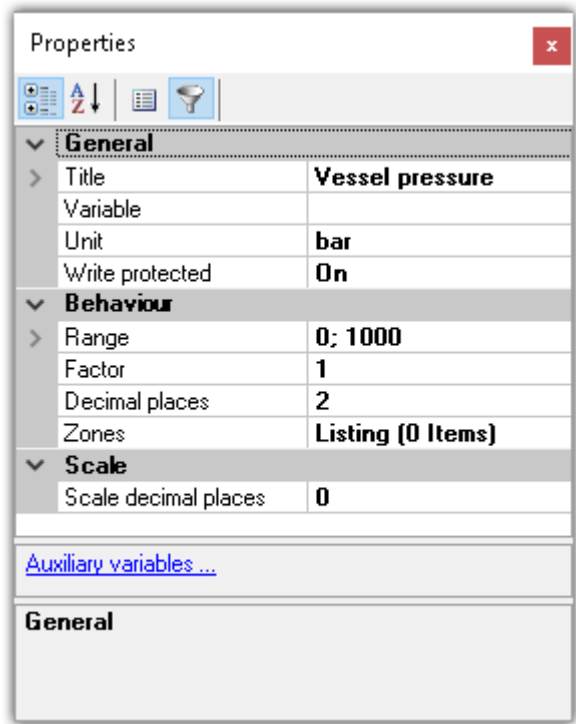
In the tool window "Properties" (or via the context menu of the selected Widget, (see [context menus of the Widgets](#) ⁶¹⁹)), the individual Widgets or the Panel pages can be adapted. The content of the Properties window depends on the respective element.

There are two display modes:

Icon	Description
	Display of the important properties. Properties which normally are adapted to the specific requirements are displayed (e.g. measurement range and variable).
	Display of all properties.

Example

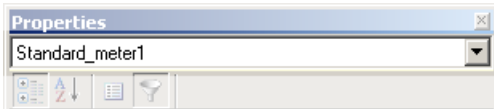
Below, one possible configuration of a *Standard meter (Automotive)* is described, with the important properties indicated.



In the window *Properties*, these properties can be adapted to your wishes. Here, for instance, the title *Vessel pressure* has been added, the unit set to *bar*, and the display range set to 0 through 1000.

Structure and operation

Selection list



Shows the name of the selected element. By means of the selection list, it is possible to select a different element. The selection on the Panel page assimilates itself to the selection.

Only shown in the tool window. Not in the "free-floating" "Properties" window.

Toolbar



Alphabetical sorting of the properties list.



Grouping of the properties by property type.

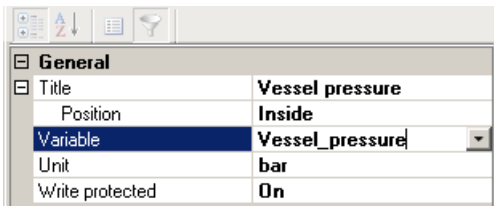


Display of all properties.



Display of the important properties. Properties which normally are adapted to the specific requirements are displayed (e.g. measurement range and variable).

Properties



List of all properties.

- Left column: Name of the property
- Right column: Current setting

To edit a setting, click in the associated text box. The desired values can be entered on the keyboard. If there are selection from which to choose, a corresponding button appears at the right margin:



Opens a selection dialog (a list or the Data Browser)



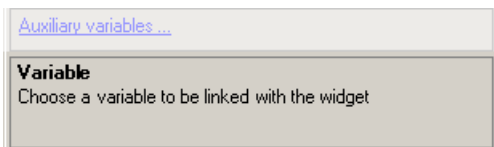
Opens a dialog (for instance Zones-dialog or event)

More dialogs



This area lists Options-dialogs which can be opened at the click of the mouse. Here you will find dialogs in addition to that for the Properties.

Description



Here you have a description of the selected property.

11.2.2 Widgets

The tool window "Widgets" provides the building blocks from which to construct a Panel page. The window contains multiple groups. In general, the groups **Curve window**, **Standard**, **Shapes** and **Devices handling** are available. Furthermore, there are specially designed Widgets in groups such as **Automotive**, **Industrial**, **Designer**, **Aviation**.

**Note****Which groups are visible**

Which groups are available depends on the product configuration.
(see [Product Configuration / Licensing](#) ^[27] or Technical Data Sheet)

Widgets are [placed \(inserted\)](#) ^[617] on the Panel page.

- [Widgets - Operation and Properties](#) ^[617]: Select this link to go to further information on operation of the Widgets.
- [Special Widgets](#) ^[910]: The descriptions of certain special Widgets are presented here.

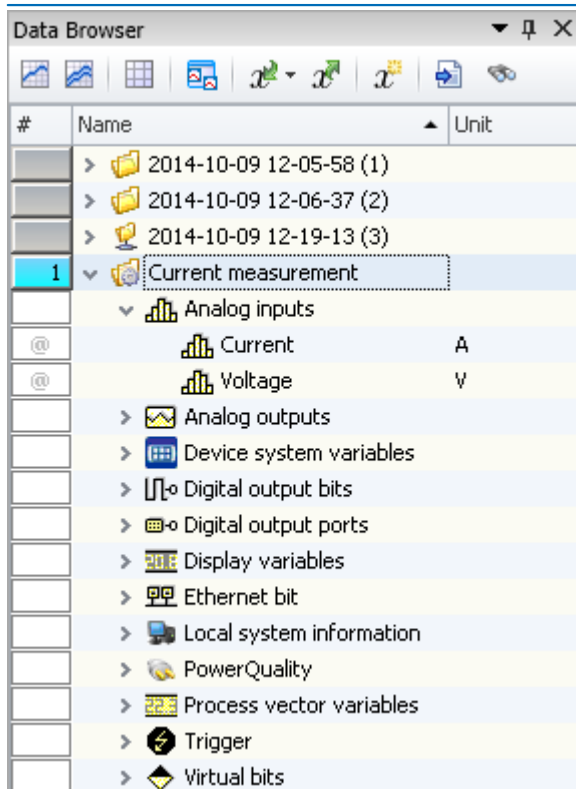
11.2.3 Data Browser

The tool window Data Browser **shows the available data (called variables)** such as measurement channels, Display variables or pv-variables in a hierarchical tree diagram. From this tool window, you can move the [variables by means of drag&drop](#) ^[940] to a Panel page and link them with Widgets. You can also [navigate through measurements](#) ^[604] and [assign measurement numbers](#) ^[603].

**FAQ**

Why are the current channels not displayed?

Answer: The tool window Data Browser **only shows changed device configurations (e.g. channels) once** the menu action "**Process configuration**" has been selected. On this topic, see the chapter "[Process configuration](#)" ^[142] in the documentation on the component Setup.



Data Browser in the Panel

Structure

The Data Browser has a [toolbar](#)⁶⁰⁵ and a table with multiple columns. Columns can be hidden or shown, according to the user's choice.

The **Table** presents the measurements and measured data in a tree diagram. The table features the following columns:

Column	Description
Numbering column (#)	Here, the assigned measurement numbers are displayed. Additionally, it is possible to lock in the measurement number (see " Assigning / Locking measurement numbers " ⁶⁰³).
Name	In this column, the measurement name and the variable name are displayed. The measurement name is determined by the data storage settings, but can be edited subsequently.
Event time	If a measurement has multiple channel-start-times (event-times), one entry appears in the column " <i>Event time</i> " for each different event time. Click on the column entry in order to select the desired event from the dropdown-list. The associated Widgets then show the selected event along with the associated variables/channels (see " Event time " ⁶⁰¹). Preselection: " <i>All available</i> "
Comment	Displays the content of the variable's property: <i>Comment</i> . With device variables, this can be defined by means of the Setup pages.
Category	Shows the content of the variable's property: <i>Category</i> .
Unit	Displays the content of the variable's property: <i>Unit</i> . With device variables, this can be defined by means of the Setup pages.
Metadata columns (optional)	If metadata about the measurement are exported, they can be added by means of additional columns in the Data Browser (see " dev man Metadata " ¹⁴⁴).

 **Note**

Showing columns

By default, not all columns are shown. You are able to add columns by means of the contextmenu item "Column selection". After any changes, use **save** to preserve the **View**.

Variable groups

The following is a list of the most important variable groups:

Groups	Description
Device variables such as: analog inputs, Display variables, Fieldbus variables, ...	A list of the device variables is provided here: "Setup - Device Configuration" > "Configuring Channels and Variables" > " Channel table " ¹⁸² .
User-defined variables	You can create your own local (PC-) variables. You can also create them in their own separate category. For more info, see the section: " User-Defined Variables " ⁵³⁶ .
System variables, such as: Trigger, measurement status, system-information and user name	In an expanded tree diagram, a variety of information is presented. For more info see the section: " Trigger, devices and system variables " ⁵²⁸ .
Virtual channels	Results channel of imc Online FAMOS and imc Inline FAMOS.
Bus Decoder	Results channel of Bus Decoder.
imc FAMOS	Results channel of imc FAMOS.
...	...

The groups can be expanded. All the variables they contain are then listed.

Using personally customized groups

You can deactivate grouping according to category and create your own grouping system instead.

The setting "Grouping by categories" can be deactivated in the Options (under "Data Browser" > "Categories"):

Ribbon	View
Extra > Options (🔧)	All

Options - Categories	Description
Grouping by categories	The variables are grouped in the Data Browser according to their respective categories (e.g. "Analog inputs", "User-defined variables"). This can be deactivated in order to sort the variables by their names. You can create your own groups by naming the variables according to the following syntax: "Group.Name". For instance, if there are two channels: MeasPoint1.Temperature and MeasPoint1.Voltage, these variables are automatically grouped in the group: "MeasPoint1".

Measurements and measurement names

The Data Browser, along with the entry for the **Current measurement** (🔧), displays all **measurements saved** with the experiment.

#	Name	Category	Unit
1	> 2014-10-09 12-05-58 (1)		
	> 2014-10-09 12-06-37 (2)		
	> 2014-10-09 12-19-13 (3)		
	> Current measurement		

Measurement	Description
 Current measurement	The variables under "Current measurement" always reflect the current status. While a measurement is being performed, you can view the current measured data as these variables under "Current measurement".  Notes on circular buffer memory By default, the circular buffer memory in the plug-in Setup is activated for display of the channels. Under "Current measurement", only the content of the circular buffer memory is displayed.
 Saved measurements	The measurements are arranged as list entries on the same level as Current Measurement. Among other factors, the measurement's name depends on the data storage settings (Setup). It may, for example, consist of a time stamp and take the form "yyyy-MM-dd HH-mm-ss". (year-month-day hour-minute-second)  Notes regarding measurement - You are able to expand the entries of all measurements without needing to load^[602] the measurements first. The information on which channels belong to the measurement is also available before the measurement has been loaded. - The measurement name can be arranged to be defined by the user. On this topic, see the option "Storage location measurement data"^[62]". Here, you can save measured data in subfolders in order to improve the sorting of data in the Data Browser. - Display of saved measurements in the Data Browser can be deactivated. On this topic, see the option: "Variables" > "Measurement Administration" > "Access to stored measurements"^[67]". This can be helpful, for instance when very many measurements are performed. In such a case, very many entries would appear in the Data Browser. If these measurements do not need to be loaded, then it can provide better clarity of overview to deactivate the display of the saved measurements.

Measurement name, and renaming measurements

The names of the saved measurements correspond to the folder names as they appear on the hard drive. It is possible to rename the measurements subsequently via the Data Browser.

- To do this, open the measurement's context menu and click on "Rename".
- Or select the measurement and press the key <F2>.

When you perform renaming, the corresponding **folder on the hard drive is also renamed**.

 Note	Renaming
Note that you may only use symbols which are allowed. If a sub-folder structure was created using the option "Storage location measurement data", it can not be subsequently modified in this way. Only the measurement folder itself can be renamed.	

Event time

If a measurement has multiple channel-start-times (event-times), one entry appears in the column "*Event time*" for each different event time. Click on the column entry in order to select the desired event from the dropdown-list. The associated Widgets then show the selected event along with the associated variables/channels.

The selection is available if the individual channels' **start times** are **different**. For instance, this is the case when in a triggered measurement the individual channels are triggered in succession by different events.

Display All

The selection list provides the entry "*All available*". This entry is selected by default. It causes all channels and their associated events to be displayed, even if these were started at different points in time.

Exception: "*Save trigger events in individual files*". With this setting, each trigger-event for the channels is saved in a separate folder. In consequence, the same channel-name can exist in multiple trigger folders. In this case, only the channel of the first applicable event is displayed.

Show relationships

When a defined event is selected, the system will make every attempt to display as many channels as possible. The channels displayed have some time overlap.

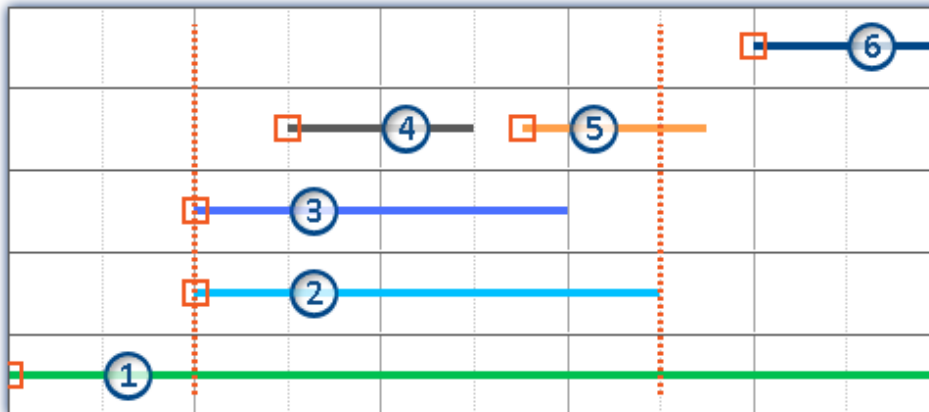
Example: A channel starts at 14:00 and ends at 15:00.

If this channel's event time is selected, all channels are displayed which comprise measured data in the time between 14:00 and 15:00.

Exception: "*Save trigger events in individual files*". See the explanation above.



Example



Dotted lines: Start and stop-times of channel 2; e.g. 14:00 and 15:00 from the above example

5 channels are recorded. Counting from the bottom to the top:

- Event 1: The first channel responds to "BaseTrigger" (start button).
- Event 2 and 3: The second and third channels are linked to the same trigger (e.g. "Trigger_01"). This produces one single event-time.
- Event 4 and 5: The fourth channel has two events. This produces 2 event-times if "Save trigger events in individual files" is activated. Otherwise, only one event time.
- Event 6: The fifth channel was triggered once all other channels governed by the triggers have concluded (except the first channel).

Thus, the Events-list shows 4 or 5 event times, depending on the setting "Save trigger events in individual files".

Depending on the selection, the relevant individual channels among these are loaded and displayed.

Example: The second event time is selected (Channels 2 and 3). The space of time corresponding to the longest channel belonging to this event is used for selecting the other channels (2). All channels which have values within this space of time are displayed (1, 2, 3, 4 and possibly 5).

Channel 5 is displayed if "Save trigger events in individual files" is not activated. When this option is activated, it is only possible to display one event time at once. In that case only the first channel (4) is displayed.

Operation - Loading and unloading measurements/variables

Saved measurements are located on the hard drive. Operating from the Data Browser, it is possible load and display these measurements. Since the saved measurements can be arbitrarily large, by default they are not loaded. The software knows the channel name and the path to their storage location, but the data located there are not in the software's own memory.

However, only variables which are already loaded can be displayed or processed.



Note

Resources

It is best to only load measurements/variables which are currently needed. Loaded elements make demands on the memory. If any measurement is no longer needed, it is best to unload it.

Automatic loading on demand: If a Widget is linked with a variable belonging to a saved measurement, this variable is loaded automatically. All of the measurement's other variables remain unloaded. It is exactly the same way with all other linkages to variables (e.g. a imc FAMOS-sequence). As soon as the value of a variable is required, that variable is loaded immediately.

To **load** the **complete measurement**, open the measurement's context menu and click on the corresponding item.

To **load a single variable**, display the variable in a Widget for example.

To **unload** the **complete measurement**, open the measurement's context menu and click on the corresponding item.

Folder symbol - Measurement status

The folder symbol in front of the measurement's entry reflects the measurement's status:

Status	Description
 Current measurement	The is the current measurement, not a stored measurement.
 Saved measurement status: unloaded	This stored measurement is not loaded. It is not used and not displayed.
 Saved measurement status: loading	This stored measurement is in the process of being loaded. This is a temporary condition.
 Saved measurement status: loaded	This is a saved measurement to which at least one loaded variable belongs. Loaded measurements can be viewed and processed.



Note

Loaded variable - Apply changes

When a variable is loaded, its associated file is loaded from the hard drive in a single action. To apply changes made to these files from outside/by other programs, the file must be re-loaded.

To do this, open the measurement's context menu and select the item: "*Reload measurement(s)*".

Operation - Assigning/Locking measurement numbers

When a saved measurement or "*Current measurement*" is **selected** in the Data Browser, a **measurement number is automatically assigned** to this entry. The measurement number is displayed in the left-hand numbering column (#). E.g., for the measurement number 1: . You may assign numbers from 1 through 99.

By means of the measurement number, it is possible to access measurements without using or even knowing their respective assigned names.



Example

- You intend to **display all saved measurements in succession** on a Report page. You have a curve window which always displays the channel: "Channel_001". However, instead of showing the channel belonging to the current measurement here, you want to display the saved measurement with measurement number: 1. When any measurement is selected, the curve window shows Channel_001 of the selected measurement, since it has the number 1.
- You wish to perform an imc FAMOS analysis on a saved measurement. The imc FAMOS sequence is configured so that the measurement results are obtained from the measurement with the measurement number 1. The imc FAMOS results are then saved back with the measurement having the number 1.
- Each measurement in a curve window is to be compared with a comparison measurement. The comparison measurement is associated with the measurement number 1. The number is locked and remains with this measurement. Any other selection will now be assigned the number 2. The curve window is linked with two channels: one from the measurement having the number 1, and one from the measurement having the number 2. If a different measurement is selected, the reference channel remains in the curve window and the second measurement corresponds to the current selection.

Assigning a measurement number

- To assign a measurement the measurement number 1, click on the cell in the numbering column (#) next to the measurement names or on an entry in the tree diagram.

#	Name	Category	Unit
	> 2014-10-09 12-05-58 (1)		
1	2014-10-09 12-06-37 (2)		
	Analog inputs		
	Current	Analog	A
	Voltage	Analog	V
	> 2014-10-09 12-19-13 (3)		
	> Current measurement		

- To assign multiple measurement numbers, hold down the SHIFT key while clicking on the cells.

Locking/unlocking a measurement number

- If you click again on a previously assigned measurement number, the assignment is locked (🔒 1). As a result, the respective number can no longer be assigned.
- To undo the locking, click on the same cell again.

Operation - Control and Navigation

To open or close a branch (e.g. "Analog inputs"),

- use the arrow keys: left/right (cursor),
- use the "+" and "-" keys on the numeric keypad
- click on the corresponding symbol (>/v)

To navigate between the measurements, use

- the mouse or
- the arrow up/arrow down (cursor) keys

11.2.3.1 Toolbar

Toolbar: Variable

Menu item	Description
 Show individually	All selected channel variables (e.g. analog inputs) are opened in separate "free-floating" curve windows ^[906] . All other selected variables (e.g. Display variables) are added to the window "Current values" ^[610] .
 Show together	All selected channel variables (e.g. analog inputs) are opened in a shared "free-floating" curve window ^[906] . All other selected variables (e.g. Display variables) are added to the window "Current values" ^[610] .
 Show current values	Opens the dialog Current values ^[610] , which displays the values of selected variables.
 Always on top	If this button is pressed, new display windows ("Curve windows" and "Current values") are always displayed in the foreground
 Load/Refill variable	Load: Creates or replaces variables with values from a file (user-defined). Refill: Imports values from a file to existing variables (device or user-defined). (See "Load/Refill variable" ^[605])
 Export variable	Exports variables' values to a file. (See "Export variable" ^[993])
 Create user defined variable	Creates user-defined variables, including single value- and text-variables. (More info is provided in the section "User-defined variables" ^[536])
 Find	Displays a search box for the purpose of locating elements in the Data Browser. The Data Browser displays only the entries found. All others are hidden as long as the search box is open.

11.2.3.1.1 Load/Refill variable

Load Variable: The action enables **creation or replace** of user-defined **variables with pre-defined values** from saved files. The user-defined variable inherits all of the properties of the variable from the saved file. (Possible targets: User-defined variables)

- If the variable doesn't exist yet, it is created.
- If the variable exists already, the existing variable is replaced with the new one.

Refill variable: The action enables **setting the values of an existing variable** from saved files. The variable must already exist before the command is run. Only the values are adopted. (Possible targets: among others device or user defined variables)

The values must be saved in a suitable file, for example one which was created using the commando: *"Export Variable"*. It is possible to select one or more files. If you wish to use multiple files, they must all be located in one folder.

**Note****Background information**

A function "**Load variable**" is now implemented, which resembles "Load Data" in imc FAMOS. Overwriting existing variables is allowed. A confirmation prompt appears for each conflict of variables. If any of the confirmation prompts is canceled, the entire import automatically cancels. In this way, the system prevents the loading of an incomplete set of controller parameters, for example.

When an existing variable is overwritten in the process of loading, then it is effectively completely replaced, including all of its properties. In consequence, the old variable is neither deleted nor is any new one created, so that no "events" are triggered by the deletion or creation. This is important for scripts, curve windows, etc.

In the process, the user-defined variables' scope is also overwritten. But when a variable for which no scope is defined is imported, the variable becomes "*temporary*".

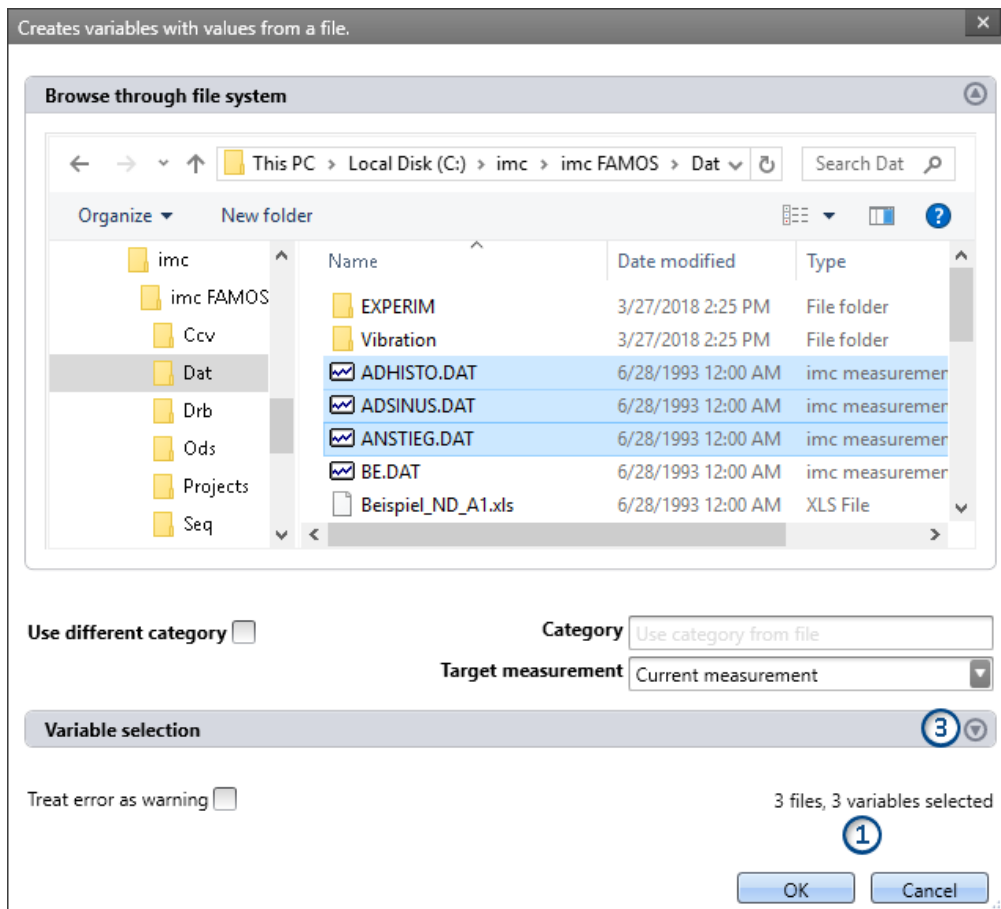
In contrast to imc FAMOS, there are variables which cannot be overwritten, e.g. channels or pv-variables. The reason is that, for instance, these variables belong to the specific device, or possess specific data formats and other properties which must be retained. If the user attempts to overwrite them, an error message is posted indicating that they can't be overwritten.

With the "**Load variable**" command, there is an option for determining whether existing variables can be overwritten with or without a confirmation prompt. By default, this option is activated, meaning that there is no confirmation prompt for overwriting.

By using "**Refill variable**", it is possible to change the content of existing variables. In this way it is possible, for instance, to change the content of pv-variables or user-defined variables. Since this only changes the variables' content and not their type, the target variable and the variable to be loaded must have the same properties, including, for example, the unit. Thus, only values and the sample count may be different. Therefore, please always use "**Load variable**" in cases where changes are possible.

Brief description:

In the upper region, there is an Explorer for selecting files:



Here you can use the "Explorer"-functions to **navigate to the files and select them**. After making a selection, you can close the dialog by clicking "OK". **All data selected are loaded** (or, when using the command, selected for loading).

Exception: If you don't select any file in the Explorer, all importable files belonging to the current folder are automatically selected.

File count (1): At the bottom right there is a readout of how many files are selected and how many variables belonging to these files are selected.

Variables-box (Variable selection) (3): Using the buttons (⊕) near the right edge, you can, for example, expand the Variables area. You can also make additional settings such as

- **de-/selecting** specific files or **variables** using the checkbox (☐), for cases where not all variables belonging to a file are to be loaded. This means that they are present in the list but will not be loaded. They can be activated again at any time.

Variables list	Description
Variable name	Name of the file/variable as found in the file.
Target variable name	Name of the variable as it is to be found in the Data Browser following load.

Other options

Category	Description
Only for: Load Variable	
Use different category	Put a checkmark in the box if you wish to create the variables in a deviating category. An input box for the new category appears. This category is applicable for all variables loaded. • deactivated: The variable is displayed in the variable's category, or without any category under "no category". • activated: The variable is displayed in the specified category or without any specification under "User-defined variable". If you wish to create a category within the category "<i>User-defined variables</i>", do the following: write a "\" before the category name.  Category: "MeasurementPoint_1". A category called "MeasurementPoint_1" is created in parallel with the other categories, such as "<i>Analog Inputs</i>". It contains the variable. Category "\"MeasurementPoint_1". This creates a category "MeasurementPoint_1" within the category "<i>User-defined variables</i>". It contains the variable. This applies, for example, to variables created by means of the command: "<i>Load variable</i>" or to generating a user-defined variable ^[536] by means of the Data Browser.
Target measurement	Description
Current measurement	Load: The variable appears in the Data Browser under "<i>Current measurement</i>". Refill: The variable overwrites the value of any variable existing under "<i>Current measurement</i>".
Last completed measurement	Load: The variable appears in the measurement which is in the Data Browser and is saved in that measurement's folder. The variable remains always available once the measurement has been loaded.
Measurement number Measurement#<No>	
Fixed measurement name	
Other options	Description
Treat error as warning	If the command is executed even though not all selected variables exist, error messages are posted. In normal cases, the Sequencer is exited when an error message appears. This also causes no variable to be imported. In order that the command always imports all variables, it is possible to activate this option. In this case, no error messages appear in the logbook, but warnings instead. Warnings do not cause the Sequencer to close.

11.2.3.2 Context menu

Context menu in the table - General

By right-clicking on the mouse over the table, the following context menu appears. Depending on the particular position, some of the items are hidden.

Menu item	Description
 Filter	Opens the Editor for creating a filter. The Data Browser then shows only the appropriate content according to the filtering.
 Transfer to imc FAMOS	The selected variables/measurements are transferred to imc FAMOS. imc FAMOS will start if it hasn't already.
Measurement number #x	Only in the Numbering column. Opens an input box for entering the desired lowest measurement number for the subsequent selection. I.e. if the number "5" is selected, at the next selection the "1" is not assigned to any measurement, but instead "5", if it is still available.
 Export the selected variables	Exports values of variables to a file.
 Create variable	Creates user-defined variables, including single value- and text-variables. (More info is provided in the section " User-defined variables " ^[536])
 Delete selected variables	Deletes the selected variables from the Data Browser. Note that device variables and system variables can not be deleted.
 Edit selected variables	Opens the Editor for changing the properties of user-defined variables. The dialog matches that of "Create variable". With some changes, the variable is newly created. In these cases, the current value is discarded. A corresponding warning is issued first. When a device or system-variable is selected, the dialog opens. Here you can view the variable's properties, but not make any changes. (More info is presented in the section "User-defined Variables"^[536])
 Show individually	All selected channel variables (e.g. analog inputs) are opened in separate "free-floating curve windows"^[906]. All other selected variables (e.g. Display variables) are added to the window "Current values"^[610].
 Show together	All selected channel variables (e.g. analog inputs) are opened in a shared "free-floating curve window"^[906]. All other selected variables (e.g. Display variables) are added to the window "Current values"^[610].
 Find	Displays the Search box. See the matching description for the toolbar ^[605] .
Add Metadata column	Metadata columns can be displayed if saved along with the measured data. Here, you can activate existing metadata columns for display.
Delete Metadata columns	Opens the dialog for deactivating the display of activated metadata column.

Context menu table - Save measurement

By right-clicking the mouse over the table on saved measurements, more context menu items become available:

Menu item	Description
 Load measurement(s)	Loads saved measurements for viewing and/or processing purposes (see " Operation - Loading/unloading measurements " ^[602])
 Unload measurement(s)	Unloads loaded, saved measurement(s) (see " Operation - Loading/unloading measurements " ^[602])
 Reload measurement(s)	Reloads already loaded measurements in order to undo changes or to load changes of the data to the hard drive (see " Operation - Loading/unloading measurement " ^[602])
 Delete measurement(s)	Deletes saved measurement(s). When a measurement is deleted, it is deleted from the hard drive. Undo is not possible.
 Load measurement settings	No function
Rename	Opens the input dialog for changing the measurement's name. If the name is changed, the associated folder on the hard drive is changed as well (see: " Measurements and measurement names " ^[599]).
Copy name	Copies the name of the measurement to the Clipboard.

11.2.3.3 Current values - Inspecting/changing variables

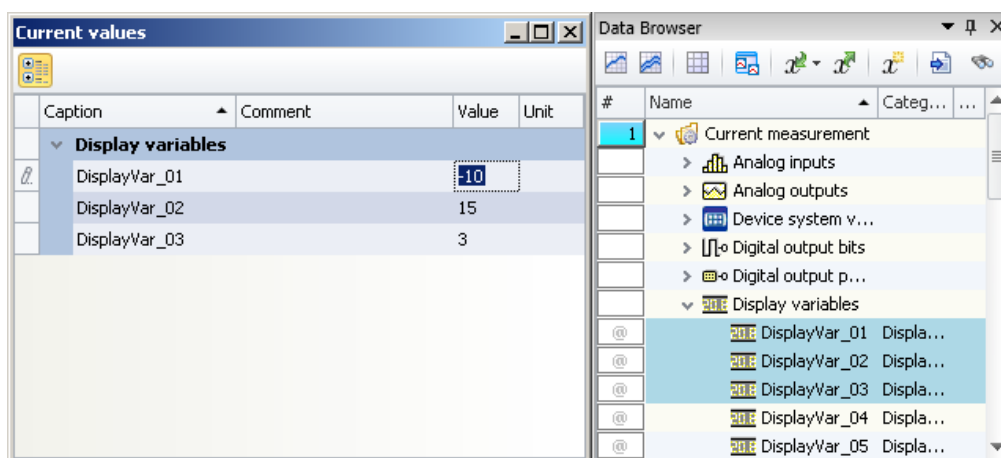
Note

The following description applies to variables which can not be viewed in the curve window, e.g. Display variables.

To directly **display** a variable's **values**, or to interactively alter them independently of a Widget,

- double-click on the entry in the variable in the tool window Data Browser
- or, select the desired variables and click on the "[Show](#)"^[605]-Button ( / ) in the toolbar of the tool window Data Browser.

Subsequently, a dialog opens, as shown below:



Inspecting or interactively altering variables
Example: three Display variables

When a **variable** is **deleted** from the Data Browser, it is automatically deleted from the Current Values-dialog.

Changing values

To change the value of variables (except bits), proceed as follows:

- click in Edit box
- enter a new value or the new text
- confirm the input with the <ENTER> key

To change a bit's value, proceed as follows:

- click in the checkbox

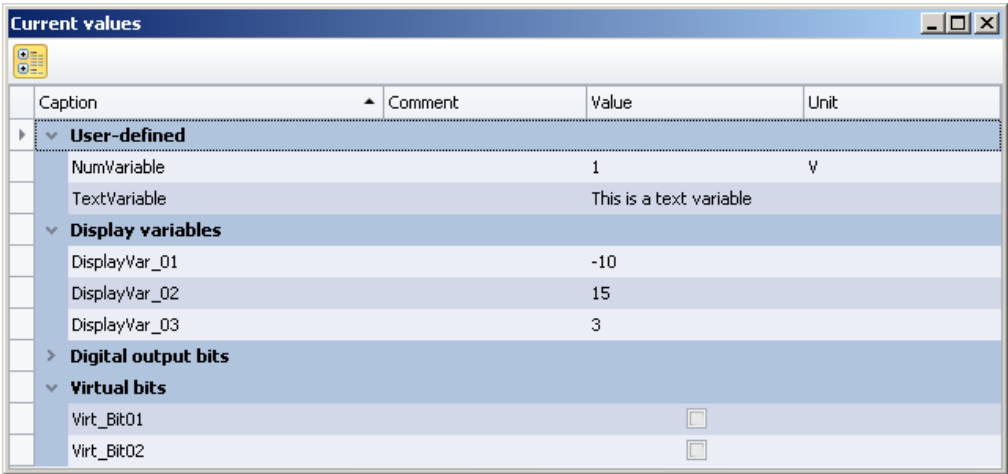
Structure and configuration

The dialog **current values** shows the desired variables in a table with multiple **columns**.

The variables are organized in groups. To activate or terminate the grouping, click on the Grouping button ().

The **table** contains, among other things, the variable names and their current value. The value is displayed in the Edit box as a number or text. For Ethernet bits there is a checkbox.

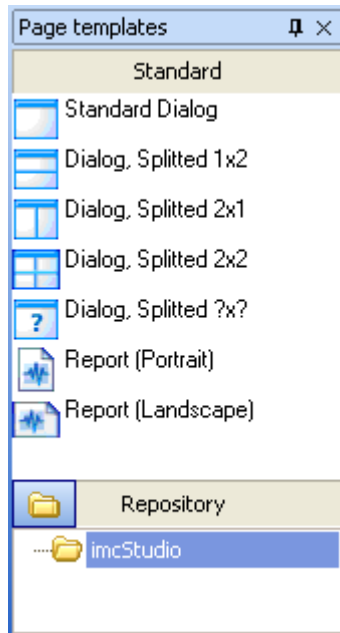
The dialog settings are saved with the experiment and displayed along with it too, if it was open.



Caption	Comment	Value	Unit
User-defined			
NumVariable		1	V
TextVariable		This is a text variable	
Display variables			
DisplayVar_01		-10	
DisplayVar_02		15	
DisplayVar_03		3	
Digital output bits			
Virtual bits			
Virt_Bit01			☐
Virt_Bit02			☐

Dialog: Current Values
Sample variables

11.2.4 Page templates



Page templates

In this tool window, you find templates for a new page. The available choices are the same as in the dialogs for [adding](#) ⁹²⁴ a new page.

11.2.5 Repository

The tool window Widgets and page templates each have their own repository. You can export completed pages and Widgets, in order to be able to insert them at other locations later.

Both repositories use the file system for the structuring and naming of the elements in the repository.

By default, the stored files are located here:

- Widget repository: %PUBLIC%\Documents\imc\imc STUDIO\Widgets
- Page repository: %PUBLIC%\Documents\imc\imc STUDIO\PanelPages

Adding elements to the Repository

Elements can be added to the Repository via the context menu or by using the Drag&Drop technique while holding down the <CTRL>-key. The elements in the Repository are assigned a distinctive name, for instance, the current page's name. If the name already exists, a prompt appears regarding whether to overwrite the existing file. If overwriting is denied, a "Save as" dialog appears.

It is also possible to add multiple Widgets to the Repository together as a single element. For this purpose, all desired Widgets must be selected when performing the adding procedure.

Importing elements from the Repository

You can use the Drag&Drop technique to move elements from the Repository to the desired location.

Menu items

You can perform the following actions by means of the Repository's menu items:

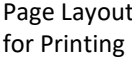
Menu item	Description
 Set the Root Directory	This enables you to change the path to the respective repository. Note that the existing repository is not automatically copied in the process. You can copy it manually.
 Delete	Deletes the selected element. This action can not be reversed.
 Rename	Renames the element. If the name of a saved page is changed in this way, the page title also adapts itself accordingly when the page is re-imported from the Repository back to the Panel.
 Delete	Deletes the selected folder along with all elements belonging to it. This action can not be reversed.
 New Folder	Creates a folder which can be used for the purpose of grouping elements.
 Rename	Renames the folder.

11.3 Context menu

Context menu: The Panel page's tab

Right-clicking the mouse over the page's tab or on the empty region at right calls the following context menu:

Menu item	Description
 New Page	Selection list for creating a new Dialog or Report-page is opened. (see: Insert - Dialog / Report ⁹²⁴)
 Import Page	This lets you import a saved page (file type: dbv). If the page's designation is identical to that of an already existing one, you are prompted whether to overwrite the existing page. If you answer "No", the imported page is appended along with a new page title (incremented number).
Export Page:	
 As Panel Page	Saves the current Panel page as a file (file type "Panel pages (*.dbv)").
 As Graphics	Saves the current Panel page as a graphics file. You can select among the following graphics formats: • Portable Network Graphics (*.png) • Windows Bitmap (*.bmp) • JPEG (*.jpg)
 As PDF	Saves the current Panel page in PDF format.
Into the Repository	Saves the current Panel page in the repository (Tool Window: Page templates ⁶¹²).
 Show Page on Monitor	• embedded: The page is displayed in the plug-in's Panel main window (default selection) • <Monitor name>: The page is displayed in fullscreen mode on the monitor selected. Independent of the opened main window.

Menu item	Description
 Zoom Zoom (All pages)	Zooms the Panel page for the purpose of better display of, for example, Report-pages (see Zooming Panel pages ^[933]).
Page Options:	
 Page Layout for Printing	Opens the dialog: " <i>Page Setup</i> " Here, you can change the setup and printout settings for Report-pages (including paper size, source, orientation, margin).
The authorization to edit and display specific pages can be linked to specific user roles.	
 Access Rights to the Page	What settings for user rights apply to a page when it is created in "<i>imc Standard User</i>" status? Example: All pages are write-protected. You can edit any pages which you yourself have made. With the access rights level "<i>Panel</i>" it is possible, for instance, for even the "<i>imc Standard User</i>" to create or import custom pages (by default, this is refused). The rights to the individual page depend on the respective user level. Thus, an "<i>imc Standard User</i>" who creates a page also has the right to edit and delete it. If a user on a different level creates the page, then these rights remain with an "<i>imc Advanced User</i>". The rights to execute these menu actions can be modified at a later time.
 Lock Page	Lock/Unlock the current page. Individual pages can be protected against changes (see Lock and unlock page ^[932]).
 Page	Deletes the selection (the page or the selected Widgets).

Context menu: Panel page

Right-clicking the mouse over the Panel page when the Design-mode is active calls the following context menu:

Menu item	Description
Insert element	Creates a Widget (operating or display widget) at the selected location.
 Undo	Reverses the last change.
 Redo	Restores a previously reversed change.
 Paste	Inserts the content of the Clipboard at the selected position.
 Design Mode	- Activated: the Panel page can be edited, the Widgets can not be operated. - Deactivated: The Panel page can not be edited. The Widgets can be operated.
Skin:	You can choose among multiple pre-defined color schemes (Skin). The color scheme influences the foreground and background colors of the Widgets and pages.
Blue, Grey, ...	Widget-colors and the background colors of the current page align to the selected skin.
Report Standard	Skin specially adapted to printout. Widget-colors and the background colors of the current page align to the selected skin.

Menu item	Description
Save skin as	A user-defined skin is generated from the current colors of the Widgets. For this purpose, a dialog appears which prompts for confirmation before saving, since this affects all Widgets on the current page. The effects are displayed temporarily on the current page.
Delete skin	Deletes the user-defined skin used.
 Grid	For the purpose of more easily positioning and moving Widgets, a grid can be imposed over the page (see Aligning to grid ⁹³²).
Adapt Page Size	See Adapt/automatically adapting the size of pages and Widgets ⁹³⁰ .
 Properties	Opens the Properties window of the Panel page ⁹²⁷ .

Context menu: Widgets

Right-clicking the mouse on the "crosshairs" () within the selected Widget calls the following context menu:

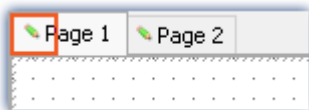
Menu item	Description
 Copy	Copies the selection or the selected Widgets to the Clipboard.
 Cut	Clips out the selection or the selected Widgets and moves them to the Clipboard.
 Delete	Deletes the selection (the page or the selected Widgets).
Arrangement:	
 To Foreground	Moves the selected Widget all the way to the foreground, on top of all other Widgets. (see: Widget-levels - Foreground/Background ⁶²³)
 To Background	Moves the selected Widget all the way to the background, beneath all other Widgets. (see: Widget-levels - Foreground/Background ⁶²³)
Align	Aligns all selected Widgets to a line. (see: Align and Cascade ⁶²⁴)
 Cascade	Widgets are in stacked at the upper left corner of the Panel page according to their respective Widget-levels. (see: Align and Cascade ⁶²⁵)
Group	Groups Widgets together (see: Grouping Widgets ⁶²⁷)
 Remove Unused Channels from Curve Windows	If curve windows are associated with channels which no longer exist, these associations can be deleted.
Curve Window Toolbar	- Activated: The curve window's toolbar is always displayed if a curve window is selected. - Deactivated: The toolbar is not displayed. All functions can be accessed via the context menu.
 Events	Opens the Editor for linking commands with switches/buttons (see Linking commands with switches ⁶⁴⁵)
Into the Repository	Saves the selection or the selected Widgets in the repository (Tool Window: Widgets ⁵⁹⁶).
 Properties	Opens the Properties window of the Widget ⁶²⁷ .

11.4 Design mode

To edit Widgets on a Panel page, the Panel must be in "*Design Mode*". The Design Mode is recognized by the appearance of the associated symbol.

Ribbon	View
Panel-Navigation > Design Mode ()	Complete
Panel-Design > Design Mode ()	Complete
Home > Design Mode ()	Standard

Or at the associated symbol next to the page name.



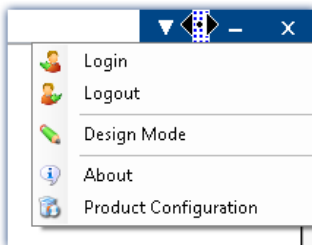
By default, this mode is **deactivated following the program start**. Activation/deactivation of the mode always applies **for all pages**.

Activating Design-mode

- via the menu item, or
- by moving a Widget to a Panel page by means of Drag&Drop.

Activation while in Full-screen mode:

Open the context menu over the bar on the top. In this menu, select the item "*Design Mode*".



Deactivating Design-mode

- by means of the associated menu item, or
- when you start the measurement.

Protecting against changes

You can have a selected page completely [locked out of editing](#) ⁹³² (independently of Design Mode).



Note

No protection against deletion of the page

Even when Design Mode is deactivated you can delete or add pages. The Design Mode is not a way to "lock" the Panel component. Only the content of the pages is protected from changes.

11.5 Widgets - operation and properties

Widgets are [placed \(inserted\)](#) ^[617] on the Panel page.

They offer a variety of [context menus](#) ^[619] and [properties](#) ^[627]

In general, there are multiple ways to operate the Widgets:

- [Move and change size](#) ^[620]
- [Select](#) ^[620]
- [Editing of Widget-elements](#) ^[621]
- [Move to background/foreground \(Widget-levels\)](#) ^[623]
- [Align and Cascade](#) ^[624]
- [Docking](#) ^[626]
- [Grouping](#) ^[627]

Some Widgets have special functions, such as:

- link to variables (see the chapter: [Variable linkage](#) ^[939])
- Extra variables (multiple scales/pointers and calculations of variables, e.g. Difference)
- [Zones](#) ^[634]
- [Linking commands with switches](#) ^[645]
- Use of [placeholders](#) ^[95]

The structure

Each Widget consists of different [elements](#) ^[621]:

- Boxes displaying the **title** or the current **value**
- **Scales**
- **Numbers**
- **Meters, bar meters, ...**

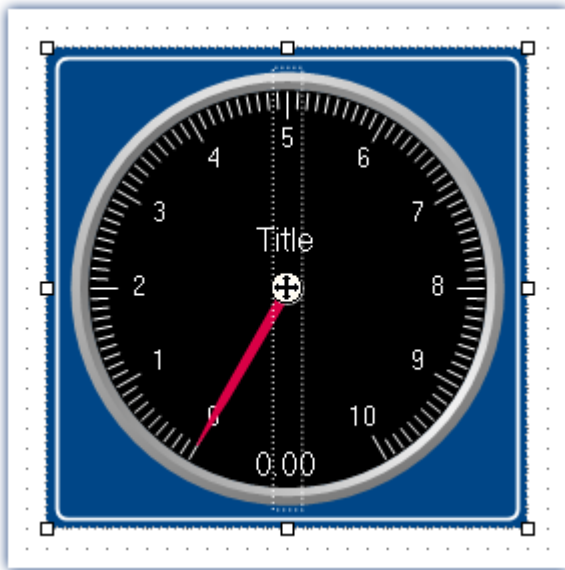
Some of these element have their own [context menu](#) ^[619] and can be [edited](#) ^[621]. Among other things, they can be moved, or their size and color can be changed. Some of these changes can also be made through the use of the Widget's [properties](#) ^[627].

11.5.1 Insert Widget

There are various ways to create a Widget on the page:

- Use Drag&Drop to move the Widget from the tool window "Widgets" to the Panel page
- Select the desired Widget in the tool window "Widgets". Use the mouse to drag a frame (of the desired size) to the Panel page
- Open the [context menu](#) ^[614] belonging to the Panel page and select "Insert Element". Select the desired Widget.
- Use Drag&Drop to move a variable from the tool window [Data Browser](#) ^[597] to Panel page and select the desired Widget (see: [Variable linkage by Drag&Drop](#) ^[940])

Once you are finished positioning, the Widget appears as selected, as shown in the image below:



*Widget in selected state
(example)*



Note

Aligning to the grid can help

An easy way to position and move Widgets is to use a grid on the page (see [Aligning to grid](#)⁹³²⁾).

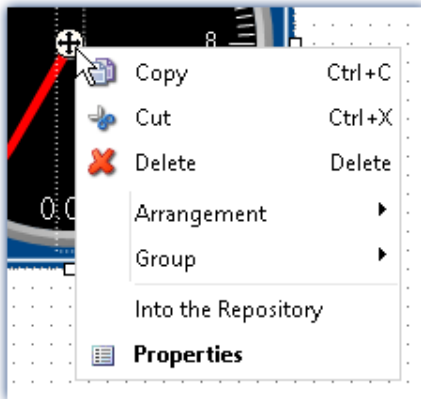
Commonly used Widgets

When you drag a variable to the Panel, a choice of Widgets is offered. The list is automatically modified according to your previous selection. Widgets which you had recently inserted frequently have priority among the choices available.

In the subgroups, all Widgets are available for selection, as accustomed.

11.5.2 Context menus of the Widgets

Widget Opening the context menu



Context menu of a Widget
(example)

To open this context menu, select the Widget and right-click on the "crosshairs" (⊕).

The context menu for the Widget appears.

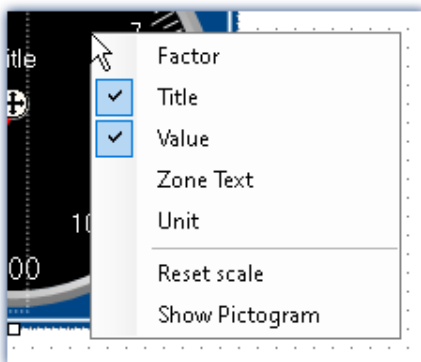
By this means, you can also open the [Properties](#) ⁶²⁷ without using the tool window.

Other context menus

Some Widgets also have additional context menus:

- to show or hide [Widget-elements](#) ⁶²¹, such as boxes, and for resetting some properties
- to edit selected [Widget-elements](#) ⁶²¹

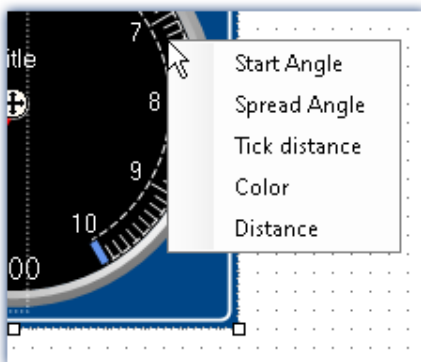
Here, using the example of a *standard meter*.



Context menu for the background

To open this context menu, select the Widget and right-click over the background.

A context menu appears, by which it is possible to show/hide [Widget-elements](#) ⁶²¹, for example.



Context menu for the Widget element

To open this context menu, select the [Widget-elements](#) ⁶²¹ and right-click over the background or the control.

A context menu appears, using which the [Widget-elements](#) ⁶²¹ can be edited.

11.5.3 Operation

Moving and changing size

Moving a Widget

You can move the Widget by clicking on it and dragging it by the "crosshairs" (⊕).

Changing the size of a Widget



Changing size

To change the size/proportions, use the mouse to drag it by the square grips on its edge.



Note

Aligning to the grid can help

An easy way to position and move Widgets is to use a grid on the page (see [Aligning to grid](#)  932).

Selection

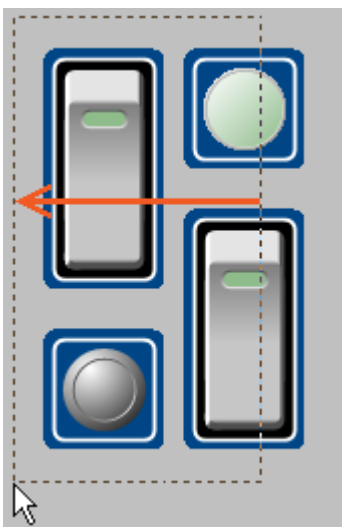
On a single page, you can select either one or multiple Widgets. The operations below can be performed as desired in any order.

- To **select a Widget**, click on the desired Widget.
- To **de-select an individual Widget**, press the <Ctrl>-key while clicking on the desired Widget.
- To **de-select all Widgets**, click on an empty region of the Panel page.

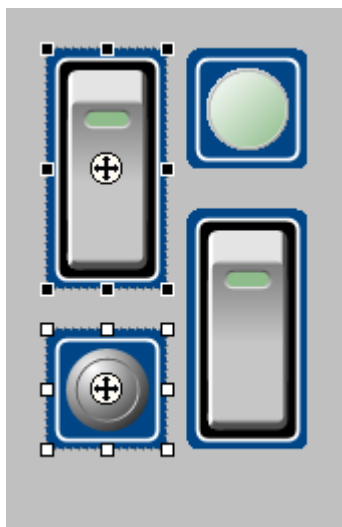
Selecting multiple Widgets

- Method 1, dragging the frame:

To select multiple Widgets, use the mouse to drag a rectangular frame around the desired Widgets. The starting point for dragging must be in an empty area of the page:

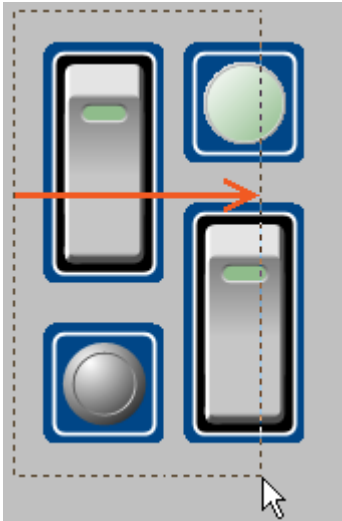


Dragging the frame from right to left

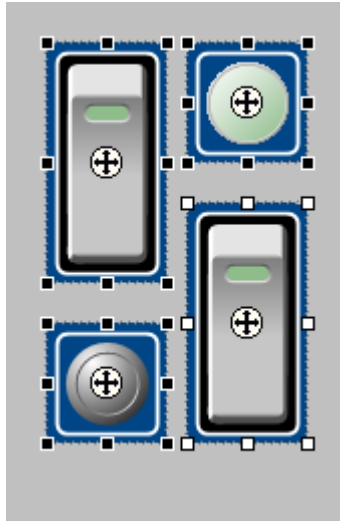


Completely framed Widgets are selected

If you drag the frame from right to left, only Widgets are selected which are completely encompassed within the frame.



Dragging the frame from left to right

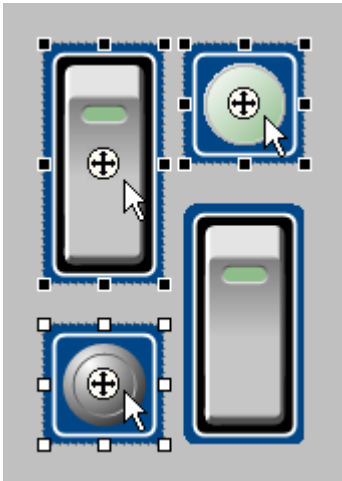


Any Widgets touched are selected

If you drag the frame from left to right, even such Widgets are selected which are only touched by the frame.

- Method 2, <Ctrl>+click

To select multiple Widgets, you can also hold down the <Ctrl>-key and click on the desired Widgets.



Multiple selected Widgets

Editing Widget-elements

Each Widget consists of various **elements**:

- Boxes indicating the **title** or the current **value**
- **Scales**
- **Numbers**
- **Meters, bar meters, ...**

Editing elements

The various Widgets offer many ways to vary the appearance without having to set the [Properties](#)⁶²⁷. You can:

- move and enlarge boxes
- edit various Widgets elements



Example

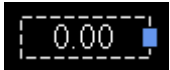
Example of three edited Widgets

The one at right consists of two *Standard meters*



Example of three edited Widgets

Each Widget also has other options. To edit the element, you must first select it. Normally a frame around the element and a blue bar, for gripping it, appear.



To illustrate, the images below use the Widget *Standard meter* as an example:



Example 1

Move and change the size of boxes



Move and change the size of boxes

 Example 2

Changing the scale angel and the sweep angle



(see also [Properties](#) ⁶²⁷)

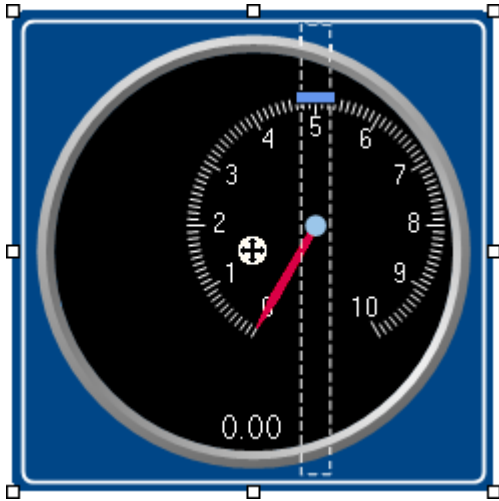


 Example 3

Changing scale position and radius



(see also [Properties](#) ⁶²⁷)



Widget-levels - Foreground/Background

The Widget-level is redefined upon inserting the Widgets. Each newly inserted Widget is positioned in front of all other Widgets. The levels are important if Widgets overlap or multiple Widgets are [docked](#) ⁶²⁶.
By means of the level, you determine whether a Widget appears above or behind another one.
To do this, open the context menu of the "crosshairs" (⊕) within the selected Widget.

- Select *Arrangement*:

Action	Description
 To Foreground	Moves the selected Widget all the way to the foreground, in front of all other Widgets
 To Background	Moves the selected Widget all the way to the background, behind all other Widgets

 Example

Thus, for instance, you can place a *Graphical switch* in the empty space belonging to a *Standard meter*:



Example of Widget-levels

Align and Cascade

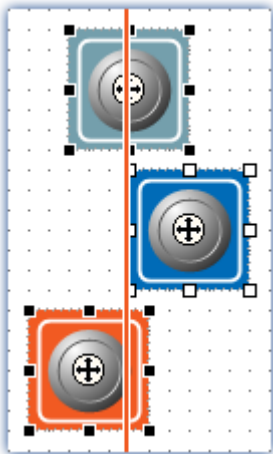
Aligning Widgets

- Widgets can be aligned along a line.
- To do this, open the context menu of the "crosshairs" (⊕) within the selected Widget.
- Select *Arrangement*:

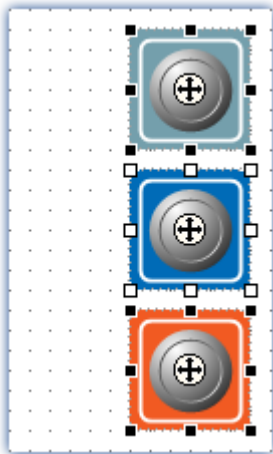
Action	Description
 Align top	Alignment to the upper edge of the last Widget selected
 Align left	Alignment to the left edge of the last Widget selected
 Align Right	Alignment to the right edge of the last Widget selected
 Align Bottom	Alignment to the bottom edge of the last Widget selected

 Example

Three Widgets are selected. The middle one was selected last. The Widgets are to be aligned to the left edge:



Widgets are aligned to the left edge of the last Widget selected



Alignment results

Cascade Widgets

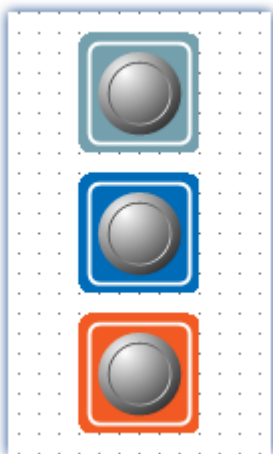
Widgets can be stacked on top of each other.
To do this, open the context menu of the "crosshairs" (⊕) within the selected Widget.

- Select *Arrangement*:

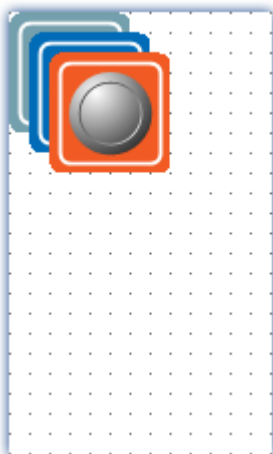
Action	Description
 Cascade	Selected Widgets are stacked in the upper left corner of the Panel page according to their respective Widget-levels.

 Example

Three Widgets are selected. The Widgets are to be stacked:



Widgets to be stacked



Results of stacking

Docking

With these commands, you can dock a Widget at your choice of edges. The dimensions of the Widget are adapted to the available width. The size can only be reduced from there and depends on the other Widgets and the page size.

To do this, open the properties of the Widget.

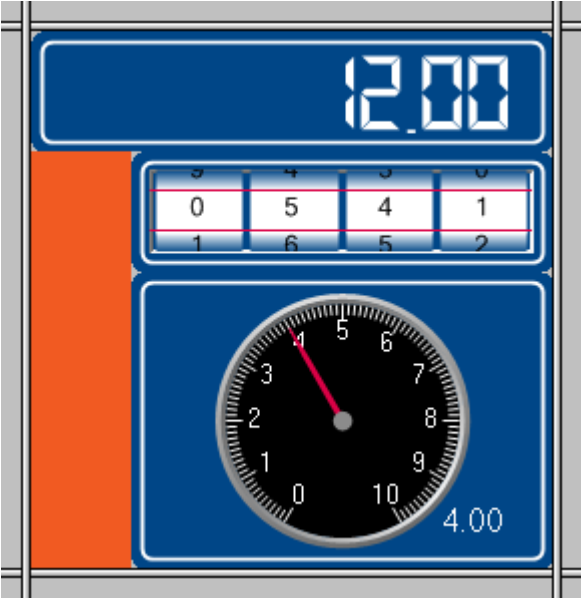
- Select under "Dock":

Action	Description
 Free	The Widget is not permanently docked to any edge, but can be freely moved. The size can be varied toward any side.
 Top	The Widget is permanently docked to the upper edge. The size can only be varied toward downward.
 Left	The Widget is permanently docked to the left edge. The size can only be varied toward rightward.
 Right	The Widget is permanently docked to the right edge. The size can only be varied toward leftward.
 Below	The Widget is permanently docked to the top edge. The size can only be varied toward upward.
 Fill	The Widget is extended in size to fill all available room. The size cannot be varied.

If multiple Widgets are docked, their respective positions depend on the **Widget-level**. Widgets positioned further in the background are closer to the edge.

 **Example**

Example for docking in a cell of the split Panel page



Widget-levels:

1	
2	3
	4

The docking of the Widgets is as follows:

- 1 - Top
- 2 - Left
- 3 - Top
- 4 - Fill

Grouping Widgets

Widgets can be grouped together. Grouped Widgets can be moved and magnified jointly. Shared properties can be set in concert.

To do this, select the desired Widgets and open the context menu of the "crosshairs" (⊕) within a selected Widget.

- Select *Group*:

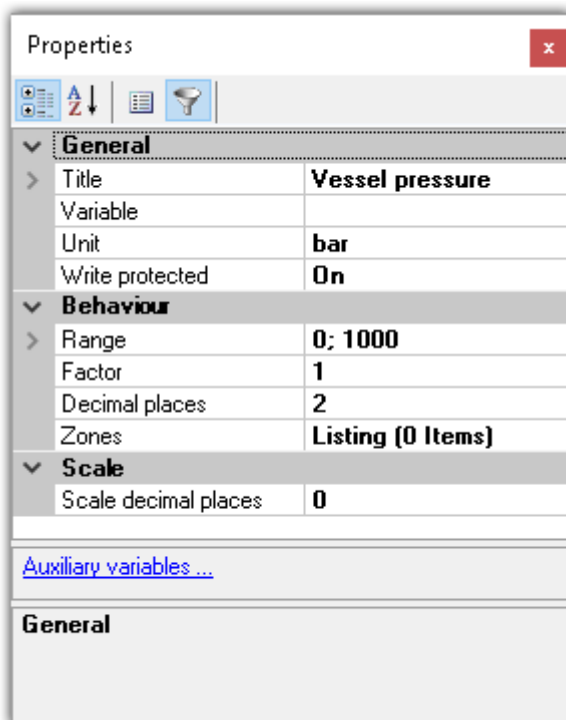
Action	Description
	Create Group Selected Widgets are joined to a group.
	Dissolve Group Dissolves the grouping. The Widgets can then each be edited separately.
	Enter Group Opens the existing group. The properties of the individual Widgets can be edited separately.
	Leave Group Closes an open group.

In order to edit the respective Widgets individually, select *Enter Group*. Then it is possible to edit the Widgets belonging to the groups. In order to exit the group again, select *Leave Group* or reverse the selection.

11.5.4 Properties - Widget

If you have selected a Widget, its properties can be adjusted in the [tool window](#)⁵⁹⁴ **Properties** (or via the context menu of the selected Widget, see "[context menus of the Widgets](#)"⁶¹⁹).

The content of the Properties window depends on the respective Widget selected. For info on the window structure, see "[Properties \(general\)](#)"⁵⁹⁵.



Below, certain commonly occurring properties are listed.

**Note****Special properties**

- The Widgets each possess only some of these properties.
- Special Widgets have additional properties not listed here.

General

Property	Description
Name	Name of the Widget. Each Widget possesses has a unique name on every page. The name may contain the following characters: letters, underlines, numbers.
Refresh rate	Updating of data. Placement of many Widgets burdens the PC's capacity. You may need to adjust the refresh rates of some Widgets if, for instance, the curve in the curve window shakes.
	The times are defined as follows:
	• very fast 50ms 20Hz
	• fast 200ms 5Hz
	• standard 1s 1Hz
	• slow 10s 0.1Hz
	• never -- --
	Reading and outputting of the new values may be delayed in proportion to the utilization of the PC's resources.
	With the option " Refresh rate of newly created Widgets ⁶⁴ ", you specify the default refresh rate of newly created Widgets.
Status	Switches the Widget active/passive.
	• Active: Widget works normally. • Passive: Widget can not be operated and its color is altered to indicate that it is passive. The current value remains displayed. • ...: depends on a variable
	0: Passive 1: Active
Title	Displayed title of the Widget.
Color	Title font's color
Default Source	Source for the title
	• User-defined: Any text • Name: Name of the variable • Long name: Name of the variable followed by the measurement name. If the Widget is linked to a variable via a measurement number, the measurement name displayed always assimilates to the measurement selected. • Comment: Comment of the variable
Font	Here the name, size and font style can be changed.
Multilingual Title	The title can be pre-defined for multiple languages ⁹⁴⁴ .
Position	Position of the title (can change the position of the Widget.)
Unit	Displayed unit. If this property is empty, the variable's unit is displayed in the Widget. Otherwise, this unit is always displayed.

Property	Description
Variable	How to link a Widget with a variable is described in the chapter " Variable linkage ".
Visible in printout	The Widget is hidden in the printout. this also applies to export as PDF or graphic. • Show: Displayed in the export or printout • Hide: Omitted from the export or printout
Write protected	Protects against changed values. • On: The value cannot be changed by this Widget (display element) • Off: The value can be changed by this Widget (control and display element) • ...: depends on a variable 0: Off 1: On

Layout

Property	Description
Dock	Docks Widget at the specified edge. See " Docking ".
Operable	Protects against editing (supported by the curve window) • Yes: The curve window can be operated and configured. • No: The curve window can no longer be modified. All options are disabled, e.g. axis scaling, Zoom, ... • ...: depends on a variable 0: operable: no 1: operable: yes
Orientation	Displayed when the Widgets can be oriented either horizontally or vertically.
Position	Position and size of the Widget.
Visible	Shows/hides the Widget (not enabled in Design Mode). • Yes: shown • No: hidden • ...: depends on a variable 0: No 1: Yes

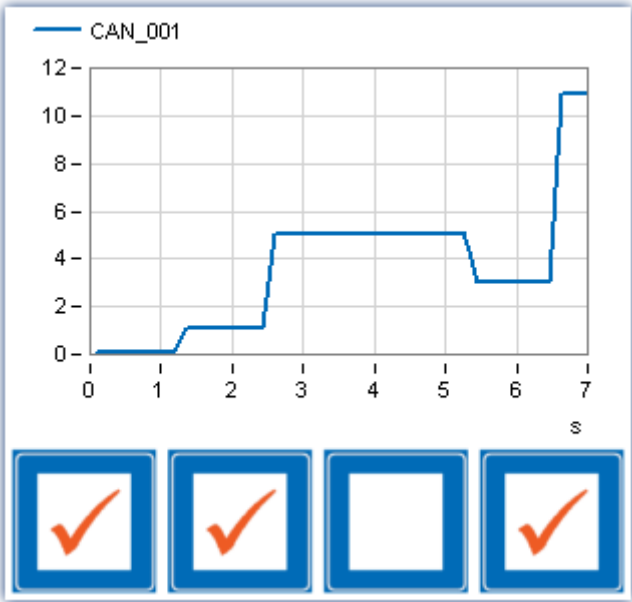
Appearance

Property	Description
Appearance	Different ways to display the Widgets. Usually only affects the appearance (2D/3D). An exception is the clock: here you can toggle between analog and digital clocks.
Background	Some Widgets can change their design. E.g. some Automotive-Widgets can adopt the design of Industrial-Widgets. Note: The Widgets will not look exactly the same and offer all the same settings as the corresponding Widget from the other design. Additionally, not all combinations are possible.
Border	This option lets you activate/deactivate the borders. Additionally, some Widgets can be displayed with sharp and some with round corners. (Only affects Automotive-Widgets)

Property	Description
Coloring	There are different color properties for the individual elements of the Widgets. The colors of most elements can be set separately. In order to apply the colors for other Widgets and for future ones, save the colors in a color scheme.
Font	Here you can adjust the font's name, size and style. This setting overwrites the title's font style.
Pictogram	The standard meter can be provided with a pictogram. This can be an icon for quick visual association. Similar to a speedometer or fuel gauge in a car.
Polygon needle	The design of the standard meter's needle can be modified. You can personally define the needle in terms of various x-y-coordinates. As an example, the coordinates of the standard needle can be loaded (form: "Standard").
Text Off/On	Texts for switch-states. There are advanced settings in the Zones-dialog  634.
Value representation	If "Value representation" is activated, the value is displayed larger than the caption. (Input, output > Numeric ...)

Behavior

Property	Description
Autosize	Adapts the size of the Widget to the size of the current picture. (e.g. Graphical switch)

Property	Description
Bitmask	Displaying variables' individual bits Selected Widgets offer the option of only displaying individual bits in a variable. Example: A Fieldbus channel returns multiple channel states with: • 0th bit: Sensor connected • 1st bit: Value exceeded • 2nd bit: Error • ... With the property, it is possible to select which bit to display. If the 1st bit is selected, the Widget only shows the value of the 1st bit. Thus, with status indicators on the Panel page, it is easy to present an overview of the status of the various channels.  <i>A Fieldbus channel returns multiple channel states at one time. In the end, the bits 0, 1, and 3 are set for a total of = 11.</i>
Decimal places	Number of decimal places for display of the value. (See also "Scale decimal places") • Automatic formatting: As many decimal places are displayed as are necessary to make a total of up to 5 significant digits visible, including pre-decimal digits. • 1, 2, or n: Only takes effect if "Factor" is set to a concrete value; otherwise the behavior always matches "Automatic formatting".  Example of "Automatic formatting". For the value 0.001213141 V, the displayed value is as follows: • For factor: "auto" -> 1.2131 mV • For factor: "micro" -> 1213.1 µV
Edit box color	Graphical display of the zones in text boxes. (Input, output > Numeric ...) The text background can be made to depend on the zones' properties. When you select "Zone color", the corresponding zone color is displayed as the background. When you select "Standard color", the background color set for text is displayed.
Events	Certain Widgets can be linked with commands ⁶⁴⁵ which are carried out at state transitions (called events).

Property	Description
	Factor with which the value is to appear. The available selection options depend on the Widget and on the unit. When a unit is specified, you can select "<i>milli</i>", "<i>kilo</i>", "<i>mega</i>", ... for example. If no unit is specified, you can select "<i>1e-3</i>", "<i>1e3</i>", "<i>1e6</i>", ... Additionally, the following selection options are available: • 1: Output of the variable's unit • Automatic formatting: Depending on the value, an appropriate factor is selected. For "<i>0.01 V</i>" for example, the factor "<i>milli</i>" is used, and "<i>10 mV</i>" is displayed. Example: The variable's value is "<i>0.1</i>" and the unit "<i>V</i>".  If you select "<i>milli</i>" as the factor, then "<i>100 mV</i>" is displayed. If you select "<i>1</i>" as the factor, no conversion calculation is performed; the displayed output is "<i>0.1 V</i>".
Factor	Unit conversion calculations The following procedure is applied by the Widgets "<i>Table</i>" and "<i>Numeric input</i>" belonging to the groups: Automotive, Industrial and Designer. 1. Certain units containing a factor, e.g. "<i>kg</i>" are converted in calculations. Other units which by chance have the same initial letter as a factor are not processed in this way; for instance "<i>Gallon</i>", where the "<i>G</i>" is not interpreted as "<i>giga</i>". 2. Using the property "<i>Factor</i>", you can specify the target factor, even if the variable's unit is "<i>kg</i>", for instance. Correct conversion to grams or tons is performed. By means of the factor "<i>1</i>", the exact unit entered is outputted. The same applies to many other units. Example: The unit is set to "<i>kV</i>", the value to "<i>0.005</i>". If the factor is set to "<i>1</i>" or to "<i>kilo</i>", the Widget displays "<i>0.005 kV</i>". If the factor is set to "<i>milli</i>", then "<i>5000 mV</i>" is displayed. For "<i>Automatic Formatting</i>", the displayed output is "<i>5 V</i>" (the factor selected by the system then depends on the magnitude of the value).
Increment	Increment size for Widgets with buttons for up and down.

Property	Description
Range	The display range, or the range of valid entries; e.g., for a standard meter, the min- and max-values for the scale.
	From Variable By default, the range is the measurement range if the variable has its own range (e.g. analog channels). The channels can also transfer other Min- and Max-values. To do this, change the settings on the following page: Setup page " <i>Analog channels</i> "; Dialog: " Curve properties ".
	All other variables automatically use the values defined, even if " <i>From Variable</i> " is selected.
	Special treatment of the scales of the "Potentiometer" and "Standard meter": The Widget adjusts the edge-ticks in such a way as to display "optimum" numbers. Example: For a range of "-97...103", the scales are rounded up to "-100...120". Defined values: Maximum and Minimum No values outside of the range specified can be entered. Scale-Widgets modify the scale according to these values. Any channel presets (" <i>Curve properties</i> ") are ignored. Infinite If the range is to be indefinite, you can enter " <i>Infinite</i> " or " <i>-Infinite</i> " (not possible with scale-Widgets).
Maximum	Maximum value of range
Minimum	Minimum value of range
Scale center	For some bar charts and Widgets with level indicator, the midpoint of the scale can be modified. The display of the bar is based on this value. As the midpoint, any value can be specified or the fixed minimum or maximum value.
Switching behavior	Behavior of the switches/buttons upon clicking the mouse. Some of the Widgets are defined by default as switches, others as pushbuttons. The behavior can be changed by this setting. • Switch: Mouse click -> Widget pressed (on); Mouse click -> Widget comes back (off) • Push-button: mouse button pressed -> Widget pressed (on); Mouse button released -> Widget comes back (off)
Text format	Here you can format the value display. Some Widgets also support placeholders .  In cases of parallel use with zone-texts, see the notes regarding: " Zone dialog ".
Zones	Displays existing zones. See " Zone dialog ". See also the display settings: "Zone representation" and "Edit box color"
Zone representation	Graphical display of the zones. Some Widgets offer various display types for the zones, e.g. standard meter, or potentiometer.

Scale

Property	Description
Interval	Smallest step between two values. The values in between are not displayed. E.g. standard meter with an "interval" of 0.5: The pointer always jumps at interval steps: 0; 0.5; 1; 1.5; ...

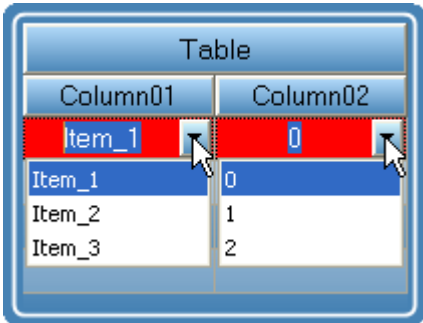
Property	Description
Scale angle	Opening and starting angle for the scale display. - Start angle: 0° -> Horizontal rightwards - Spread angle: clockwise
Scale decimal places	Number of decimal places for the scale display. The count of decimal places is automatically adjusted when the range is specified by the variable.
Scale position	- X; Y: Positioning of the scale midpoint to the Widget-midpoint - Radius: distance of the scale from the scale-midpoint
Tick distance	Distance between main ticks. The value is automatically aligned to the "range". I.e. when the range is changed, the changed value is discarded.

11.5.4.1 Zone dialog

In this dialog, you can **define which values** are available for selection in a "*selection box*", for example. You can also specify **colored display** for the **value ranges** of various Widgets:



Example of a standard meter

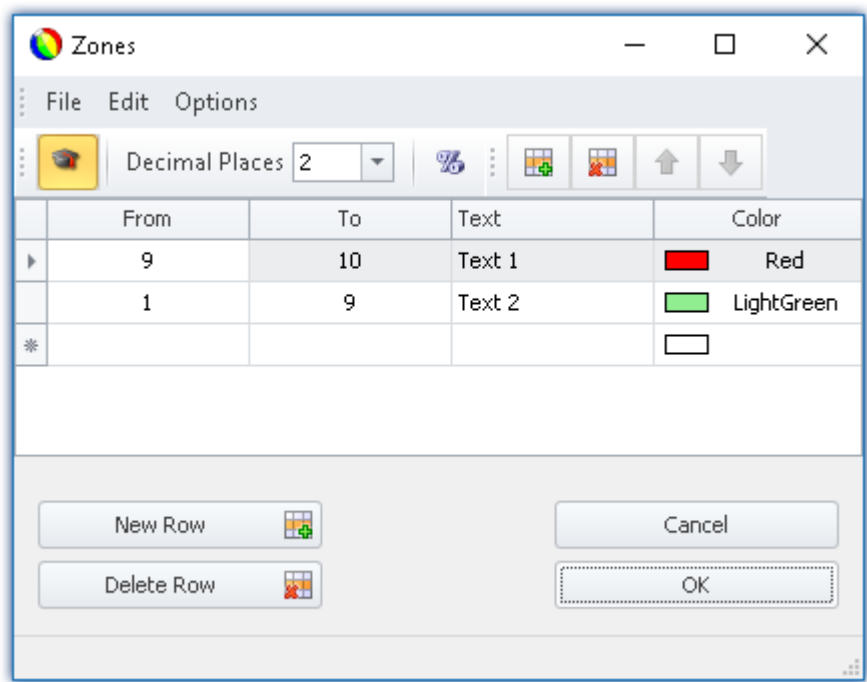


Example of a table with list boxes

To open the dialog, proceed as follows:

- Open the properties of the Widget.
- Click on the button  in the box "Zones"

The Zones-dialog is called, which appears as shown below:



Setting zones for a Widget (example: standard meter)

 **Note**

The dialog sometimes appears different

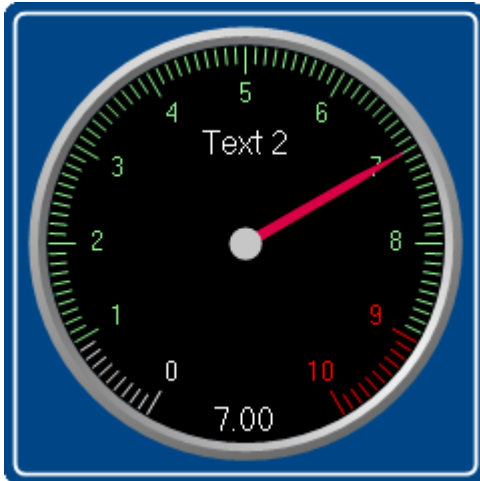
The dialog's structure and the displayed [columns](#)⁶³⁷⁾ depend on:

- the Widget (control elements require different settings than display elements)
- the Widget property "[Write protected](#)"⁶²⁷⁾ (preset values are not required)
- the option "[Advanced display](#)"⁶⁴¹⁾ (hidden and automatically calculated columns are displayed)

Here, the value ranges of the Widget can be distinguished by color. In the example, the "*standard meter's*" value range is subdivided as follows:

< 1	default (not defined) e.g. white
1 - 9	green
9 - 10	red
> 10	default (not defined) e.g. white

The image below shows the "standard meter" set accordingly:



Example: standard meter

Changing/creating zones

Changing values, texts and colors

To change an entry,

- click in the corresponding box
- perform the change desired

By default, the values are to be entered as absolute quantities. For some Widgets, the values can also be entered as a [percentage](#) ⁶⁴¹ of the range.

Adding or deleting a zone

In order to create or delete a zone, use the items: "New Row"  / "Delete Row"  (e.g. via the context menu or the menu items).

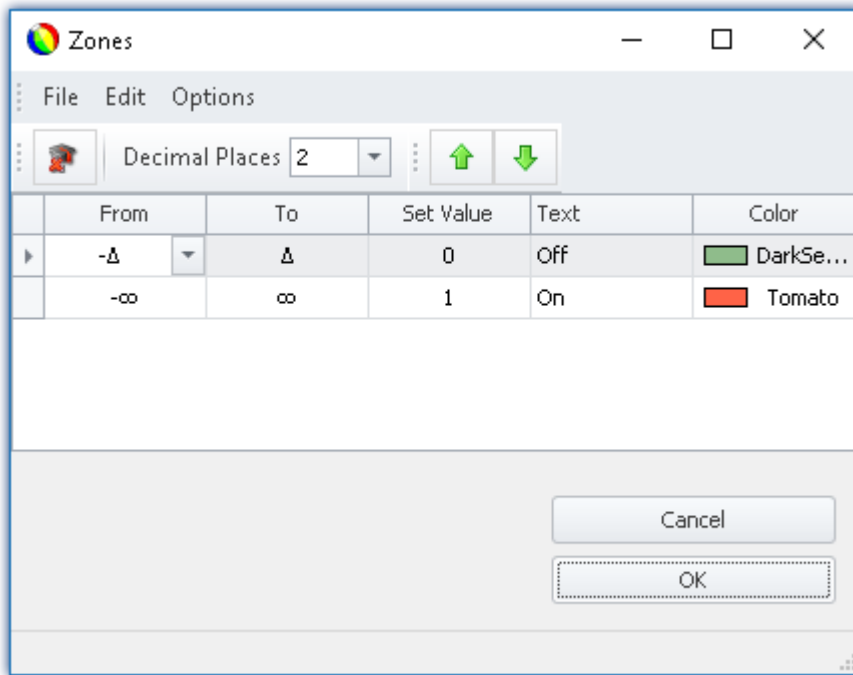
Moving the zones

The top zone has top [priority](#) ⁶³⁹.

To change the order, use the items: "Move Row Up"  / "Move Row Down"  (e.g. via the context menu or the menu items).

Infinity and zero

In special cases, symbols are used for *Zero* and *Infinity*. Below, the zones for a **switch** are shown with "[Advanced display](#)" ⁶⁴¹.



Zones of a switch with "Advanced display"

The switch should be "Off" when near "Zero" and for all other values "On".

11.5.4.1.1 The columns and their interdependencies

Some columns are hidden

The structure of the dialogs depends on the Widget:

- If **values are set** with Widget, the column "Set Value" is displayed (e.g. "Listbox" and "Switch")
- If the Widget usually **displays values**, the columns "From" and "To" are displayed (e.g. "standard meter")

The dialog's structure depends on the property "Write protected":

- If the Widget is **write-protected**, then **no value can be set** with this Widget. In this case, even for a Switch only the columns "From" and "To" are displayed.

If the "[Advanced display](#)₆₄₁" is activated, the columns "Set Value", "From" and "To" are displayed if present.

Relationship between set values and value range

The values in the columns are mutually interrelated. The values entered in the column "Set Value" should normally lie within the range between "From" and "To".

If only the column "Set Value" is displayed and you change the value, the values in the columns "From" and "To" change accordingly.



Note

Automatic adaptation

- If you have **activated** "[Advanced display](#)₆₄₁", **this automatic adaptation is deactivated**. This means you can also set values outside of the range. If you close and then re-open the dialog, the "[Advanced display](#)₆₄₁" is deactivated again. Each change to the values re-adjusts all the values again!
- Make note that because of the above fact, the Widget may get into an inoperable state.

**Example****Example of automated adjustment of the ranges: "Listbox"**

The Widget has four zones:

Set Value	Text	From (not displayed)	To (not displayed)
0	Eintrag_1	-0,5	0,5
1	Eintrag_2	0,5	1,5
2	Eintrag_3	1,5	2,5
3	Eintrag_4	2,5	3,5

If you change the value sin the column **Set Value** and have not activated the **Advanced display**, the other values adjust themselves.

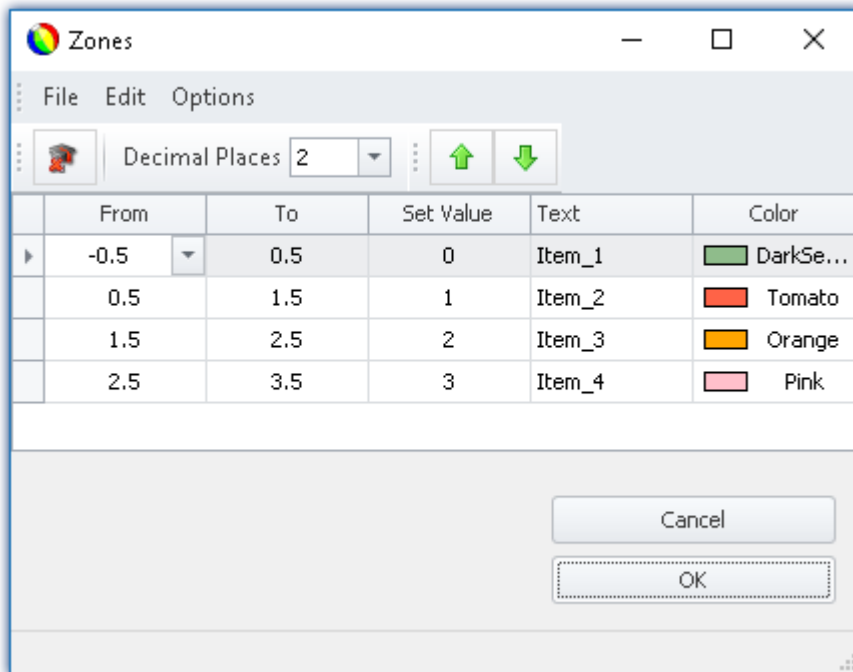
Set Value	Text	From (not displayed)	To (not displayed)
0 > 1	Entry_1	-0.5 > -∞	0.5 > 1.5
1 > 2	Entry_2	0.5 > 1.5	1.5 > 2.5
2 > 3	Entry_3	1.5 > 2.5	2.5 > 3.5
3 > 4	Entry_4	2.5 > 3.5	3.5 > ∞

Additional example of automatically calculated values:

Set Value	Text	From (not displayed)	To (not displayed)
1	Entry_1	-∞	1.5
2	Entry_2	1.5	3.5
5	Entry_3	3.5	12.5
20	Entry_4	12.5	∞

11.5.4.1.2 Order and priority / overlapping zones

In order that no values be omitted, the zones mostly **overlap**. The **top zone has higher priority**. Below an example:



Zone dialog with "Advanced display"

The first zone has the range -0.5 through 0.5. The second has 0.5 through 1.5. The value 0.5 appears in both zones. However the top zone has higher priority.

If the value is exactly 0.5, the zone is displayed with the text "Item_1".



Example

Example of an LED with overlapping zones

We intend for the LED to shine "red" for a value of 5. And "green" for all other values:

This amounts to three zones:

- 1: -infinity to 5
- 2: 5 to 5
- 3: 5 to infinity

Ensure that Zone 2 (5 to 5) is not covered over by the others. For that reason, it must be placed on top:

- 1: 5 to 5
- 2: -infinity to 5
- 3: 5 to infinity

It is then also possible to define just two zones:

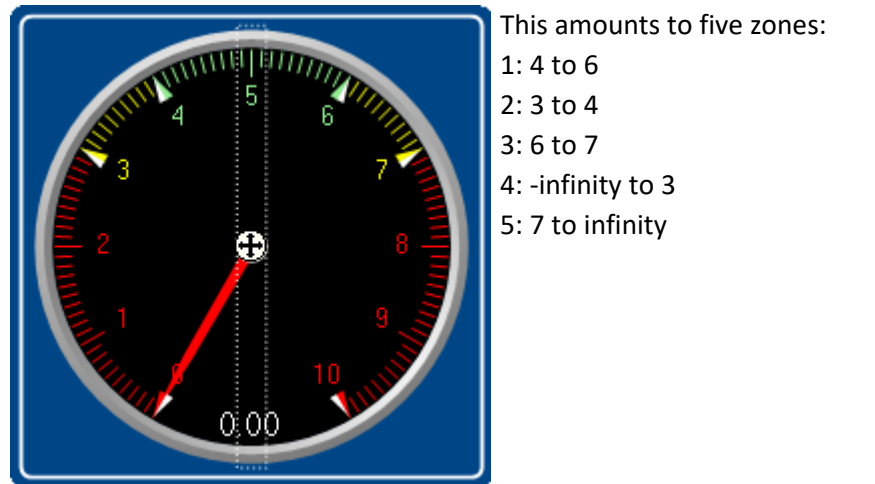
- 1: 5 to 5
- 2: -infinity to infinity

If the value is "5", it lies in the first zone. If it is not "5", it lies in the second zone.

 Example

Example of a pointer instrument with overlapping zones

The ideal value lies between 4 and 6 ("green").
One value beyond that range in either direction is still OK ("yellow").
Anything else is either too high or too low ("red").



One can make these into three zones:

- 1: 4 to 5 ("green")
- 2: 3 to 7 ("yellow")
- 2: -infinity to infinity ("red")

Zones

File Edit Options

 Decimal Places 2 

	From	To	Text	Color
▶	4	6	Text 1	 LightGreen
	3	7	Text 2	 Yellow
	-∞	∞	Text 3	 Red
*				

New Row 

Delete Row 

Cancel

OK

11.5.4.1.3 Other options and settings

Advanced display

By default the Zones dialog has a **simple display**. The structure is adapted to the normal operating style for the Widget. Accordingly, not all columns are always displayed.

The "*Advanced display*" is required for some complex applications. When the display is activated, all columns are displayed.

Application examples:

- The "*Listbox*", which is to be used as an input/output element (ranges and set values need to be defined).
- Switches or LEDs having more than two zones (e.g. multi-colored lighted pushbutton)
- Ranges and set values can be defined by variables (e.g. value limits which depend on certain settings)

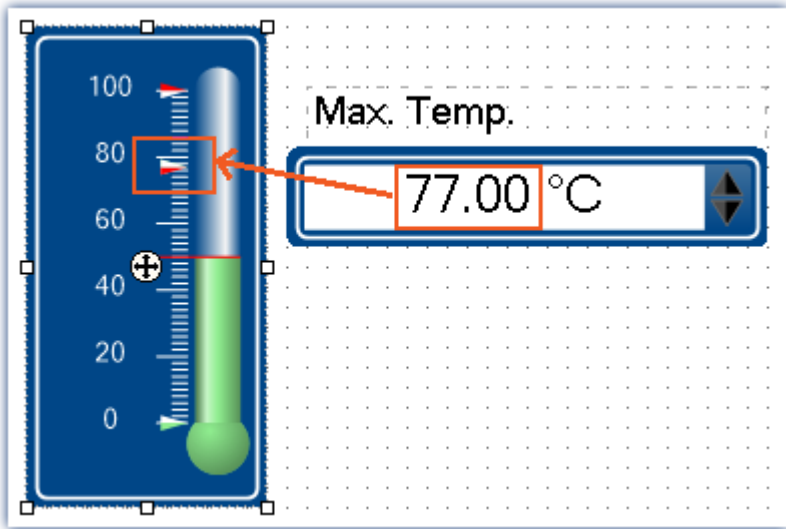
Activating Advanced Display

In order to activate "*Advanced display*", use the entry: "Extended display"  (e.g. via the menu entry).

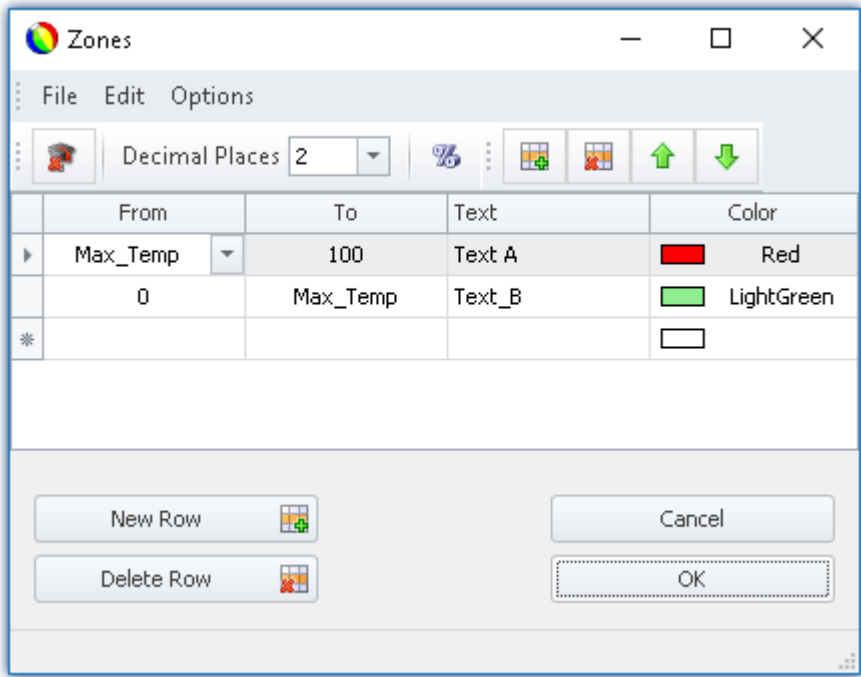
 Example

Variable value limits in a Widget

When a maximum temperature level is exceeded, the thermometer is to turn "red".



Example: Thermometer with user-defined ranges



Zone-settings
The variable: "Max_Temp" is used as the range boundary

Percentage

By default, the values are entered in absolute terms. For some Widgets, you can also enter the values as a percentage of the range.

If at a later time the Widget's display range changes, the zones adjust proportionally. However this means that fixed midpoint values can not be specified (all values adapt to the range).

Activating Percentage Entry

In order to activate "Percentage entry", use the entry: "Percentage"  (e.g. via the menu entry).

 Example

Display range of the Widget: 0-10
Zone: 0-5, 5-7, 7-10 (0%-50%, 50%-70%, 70%-100%)



The range changes to: 0-50
New zones: 0-25, 25-35, 35-50



Numeric Zone / Textual Zone

Some Widgets can **display texts or numerical values** (for instance "Tables" belonging to the group Automotive, Industrial and Designer).

Column01	Column02
Item_1	0
Item_1	0
Item_2	1
Item_3	2

Table with numerical and textual zones

The variable is assigned the same value in both cases. However, it is often necessary to have the user select a text while internally writing a number.

Switching between numeric and textual display

In order to switch between "Numeric" and "Textual Zones" display modes, use the entry: "Numerical zones"  / "Textual zones"  (e.g. via the menu entry).



Note

Zone-texts are ignored when ...

- Zone texts are ignored when the **property "Text"** is used. The property "Text" defines how the value is to be displayed. If you wish to define **a separate display for each zone**, delete the content of the property "Text" and specify the complete representation of the values by means of the zone texts.
- Zone texts are ignored when a **text-variable** is used. Any desired texts can be displayed by means of the text-variable. In order to avoid any duplicates, the zone texts are ignored.

Text as set value

Some Widgets can **set texts** (e.g. "Tables" of the groups Automotive, Industrial and Designer). By contrast, most Widgets can only set numbers.

With this setting, all that remains is the set value, and when "Textual Zone" is activated, the display value also. It is no longer possible to enter a range. The set value is written to the variable as text, for which purpose a text-variable is needed.

Activating the writing of texts

In order to activate "Text as set value", use the entry: "Text as set value"  (e.g. via the menu entry).

Decimal places

Here you set how many decimal places are to be shown in the dialog. The values can also be expressed in exponential notation. For instance, select "E3".

The following values can be entered: 0, 1, 2, ..., E0, E1, E2, ...

This setting only refers to the display of numbers in this dialog and has no effect on the Widget.

Setting the amount of decimal places

To change the number of decimal places, select the menu item: "*Decimal places*" (e.g. via the context menu or the menu).

Zone ring

The pointer instrument can have multiple "*Zone rings*".

To **show** the zone ring, the Widget-property: "Zone display" is used. By default, the ring is deactivated.

By means of the menu item "*Edit*" > "*Create Zone Ring*" **you can create an additional zone ring**. If zone display had previously been deactivated, it is activated automatically. Now, two rings are shown (if the second zone ring is supplied with zones).

In the "*Edit*" menu, you can **switch between the zone rings** or **delete a zone ring**.



Two zone rings are displayed. The dark green area is the second ring.

11.5.4.2 Linking commands with switches

Certain Widgets can be linked with **commands** which are carried out at state transitions (called **events**).

As an example, the two events (⚡) belonging to a **Switch** from the group **Standard**:

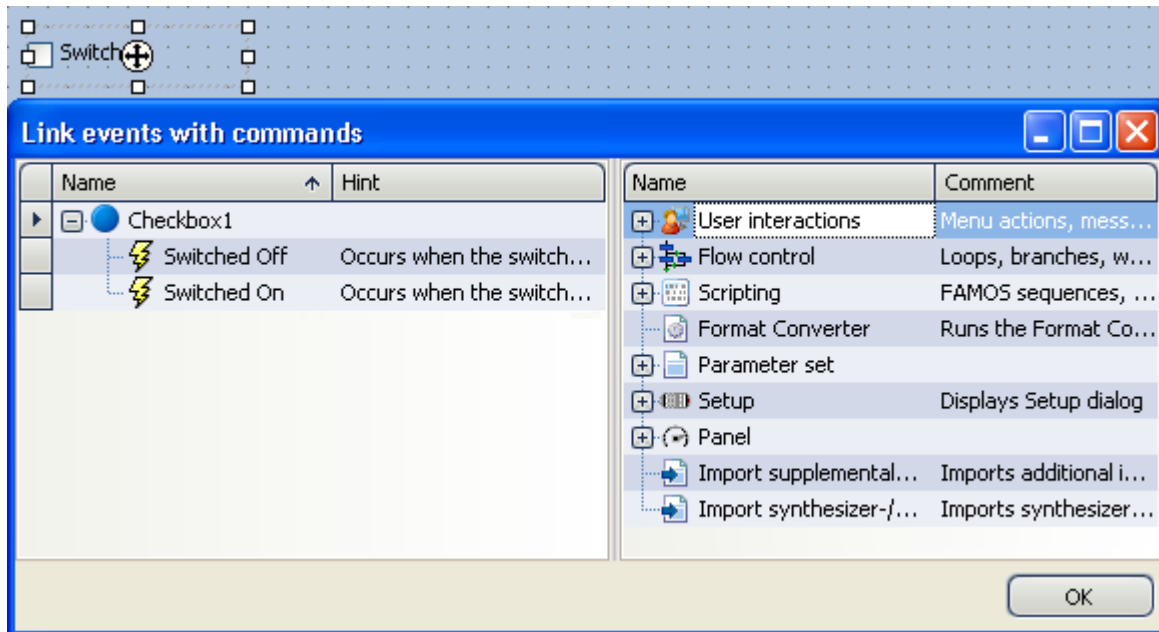
- ⚡ **Switched off** (from On to Off)
- ⚡ **Switched on** (from Off to On)

The Widgets include **Pushbutton** and **Switch** from the group **Standard** and various switches/buttons from other Widgets (**Automotive**, **Industrial**, **Designer** if these are installed).

To assign a **command**:

- Select the Widget
- Open the [Properties](#)  of the Widget

- In the Properties window, click on "Events ..." at the lower left. Subsequently, the dialog: **Event Dialog** opens, as seen in the image below:



Event dialog:
Events belonging to a switch from the group Standard



Reference

Commandos and Sequences

- The workings of the individual commands are described in the chapter [Command Reference](#)⁹⁵⁴.
- A **sequence of commands** can be created in various imc STUDIO plug-ins and is described separately (see the chapter "[Sequencer, events and commands](#)"⁹⁴⁶).



Example

Below an example with multiple commands

Name	Hint
Checkbox1	
Switched Off	Occurs when the signal is switched off
#01 Execute menu action: Stop	Execute menu action: acDeviceStop
#02 Execute FAMOS sequence	...
#03 Browse in workspace 'Report'	
Switched On	Occurs when the signal is switched on
#01 Execute menu action: Start	Execute menu action: acDeviceStart
#02 Browse in workspace 'Measurement'	

After switch on:

- the measurement starts
- the Panel page *Measurement* is opened

After switch off:

- the measurement stops
- a imc FAMOS analysis is performed (if installed; see *Technical data sheet > Additional imc Software Products*)
- the Panel page *Report* opens

11.6 Curve window

Curve Window shows the graphic representation of measurement signals and computed waveforms, in the simplest case as a labeled coordinate system and a curve. The curve window displays individual and independent objects, which can be positioned, enlarged and edited as desired.

The curve window has an extensive menu offering a multitude of options for display and graphical evaluation of waveforms. All curve windows always display the current values.

The following offers a short overview of the main features of the curve window:

Function	Description
scaling	Automatic or manual scaling of the x- and y-axes
Zoom	allows detailed viewing of a curve section
Unzoom	returns to entire curve window display
Measurement values	Two independent measurement cursors are available, which display the x- and y-values of the curves, as well as value differences and slopes. These cursors can be used to select sections of a curve for special processing. Measurement values can be transferred to a report.
Markers	markers can be set to display comments or indicate the coordinates of individual points of a curve in the curve window.
Overview Window	Shows the position within the entire curve of a zoomed curve section.
More Curves	Displays additional curves in the curve window to allow comparisons.
Print	Prints out the curve window's content with the resolution of the designated printer.
Clipboard	Curve window graphs can be copied to the Windows Clipboard. These can be pasted to word processing and DTP applications and combined with text and additional graphics. Several options are available for copying curve graphics to the clipboard or the printer, including font type, line width and dimensions.
Export	The waveform displayed in the curve window can be exported to other imc applications. This is a fast and simple method of data transfer analogous to the Windows clipboard.
Transfer to imc FAMOS	Waveforms can be transferred directly from the curve window to imc FAMOS. A sequence in imc FAMOS can be automatically initiated with a transfer. Since the curve window is used by all imc products, it would be possible, for example, to send data from imc STUDIO to imc FAMOS in this way, and to process them there directly by means of a sequence.
Logarithmic, dB and linear axes	
XY, polar and locus plots	
Colors	The colors for graphics can be set as desired
Grid	The coordinate system can be overlaid with a grid.
Configurations	Curve window configurations can be saved in a file and loaded again.
1/3-octave or octave	The x-axis of a curve window can be set to normal scaling for a 1/3-octave or octave analysis.
overlapping, stacked or waterfall diagram	Several curves can be displayed together for comparison: overlapping , stacked or in a waterfall diagram . Various options are available for waterfall display , e.g. the angle of view .

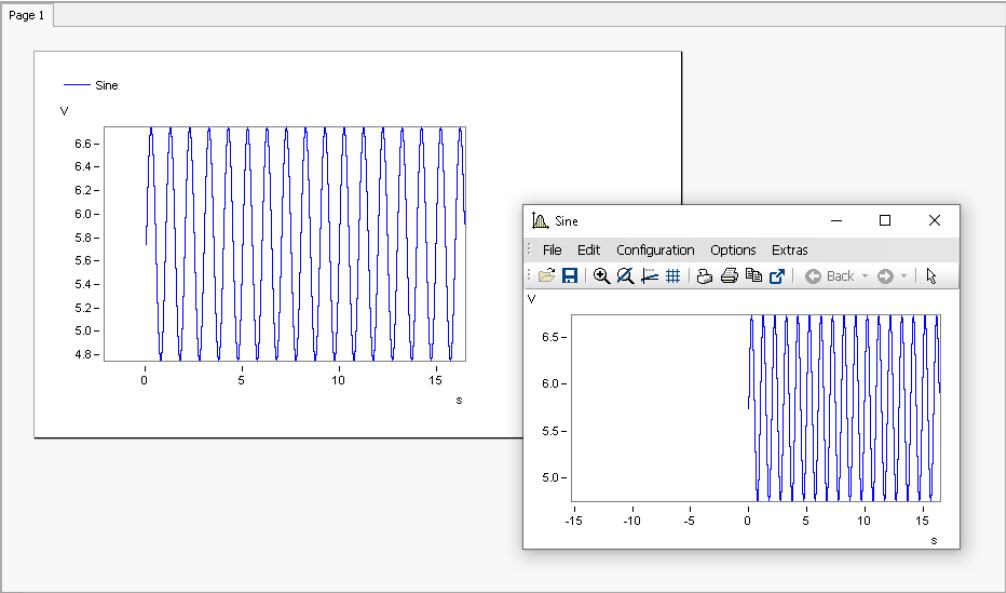
Function	Description
symbols	Curves can be marked with symbols to help differentiate between curves in a monochrome display.
periodic waveforms	An option is available for periodic waveforms in which all periods are compared with each other. This helps to identify trends and aberrations.
Color maps	A "bird's-eye view" of a colored 3D plot; color-coding indicates height.

Curve Window - Embedded or "Free-floating"

The curve window comes in two varieties:

- "free-floating" and
- embedded on a Panel page

The "free-floating" curve windows are associated with the experiment. They remain open until closed (X) and are displayed even when the Panel is not open.



Example: curve window - embedded and "free-floating"

The embedded curve windows are only visible if the associated Panel page is opened..

11.6.1 Display variants of the curve window

Function

The display mode of a curve window can be defined in a variety of ways. The display options include the appearance of the coordinate system, selection of date/time labeling or 1/3-octave labeling. Several special attributes are also available, including number of secondary ticks, period-comparison, marking of lines with symbols and definition of the reference value for dB displays.

Calling the dialog

Select menu option "*Configuration*" > "*Display*" in the curve window. A dialog appears in which the curve window display can be defined.

Tap	Description
Display ⁶⁴⁹	Display variants of the curve window
Legends ⁷⁶⁸	All curve windows can include a legend, with the channel (variable) name, line color etc.
Map ⁷⁷¹	Map lets you place a picture in the background. It is additionally possible to layout the picture in specified coordinates.
Color palette ⁷⁸⁴	A channel's color can be governed by a reference signal's amplitude.

Display

Display

Legend

Map

Color palette

Text

Display type

☐ Default

☒ Stack y-axes

☐ Waterfall diagram

☐ Last value as number

☐ Table

☐ Bar meter

☐ Color map

☐ 3D

☐ Polar plot

Scale of x-axis

☒ Chosen x-unit

☐ Date/time: absolute

☐ Days/hours/minutes: relative

☐ Third/octave labeling

Reference for dB-display:

1.00000E+0

☒ Labeling

Coordinate system size:

Default

☐ Show x = 0 (trigger)

☐ Number trimming (useful when scrolling)

☐ Grid

OK

Cancel

☒ Apply changes

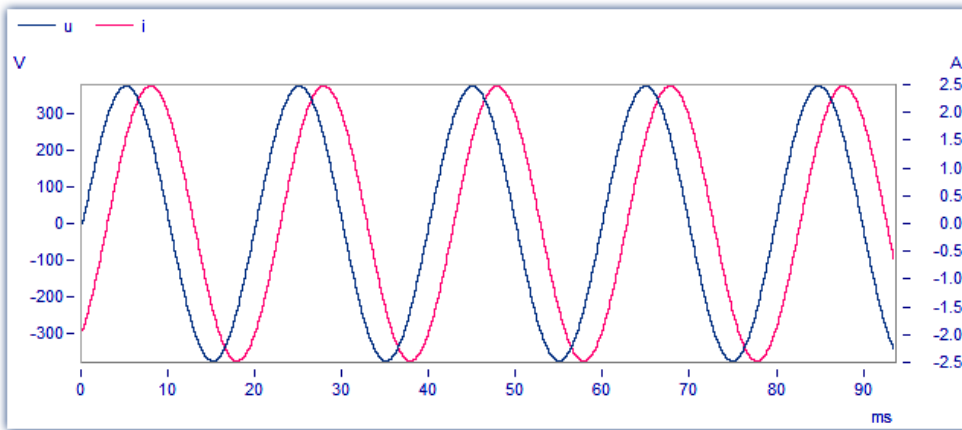
Set as default

Topic

Display type

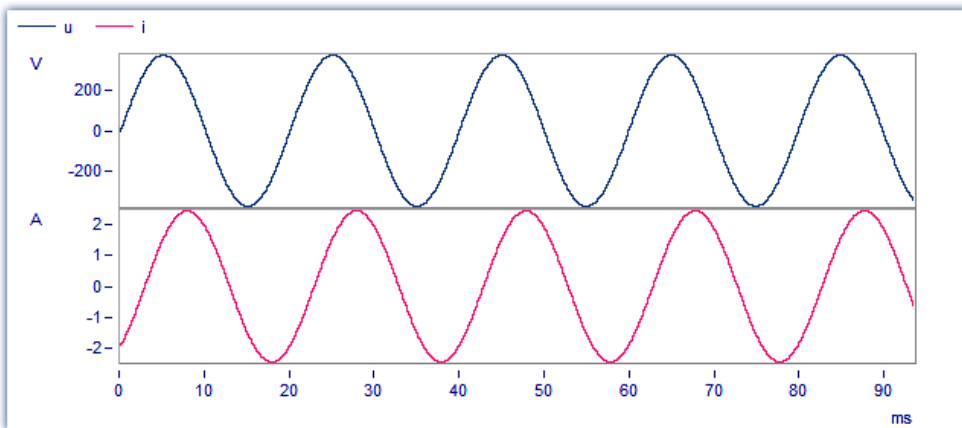
Default

In *Default* display, curve windows with more than one y-axis will show the different axes side-by-side. Several curves are then displayed in the same space.



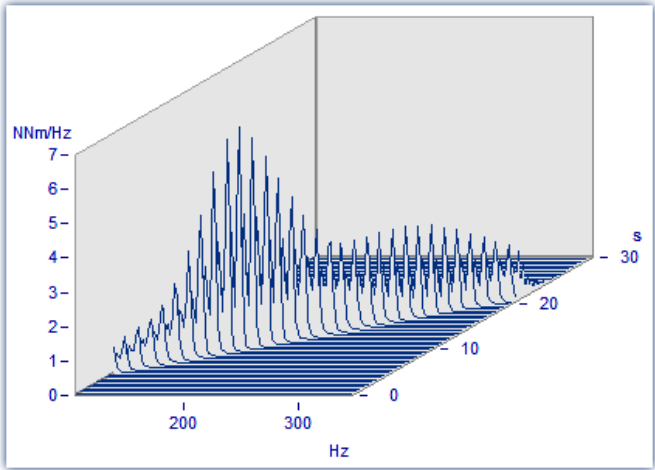
Y-axes stacked

An alternative to the *Default* setting is the *y-axes stacked*. Here, the y-axes are displayed stacked vertically, so that each curve is displayed in its own space, separated from the other curves in the window.



Waterfall Diagram

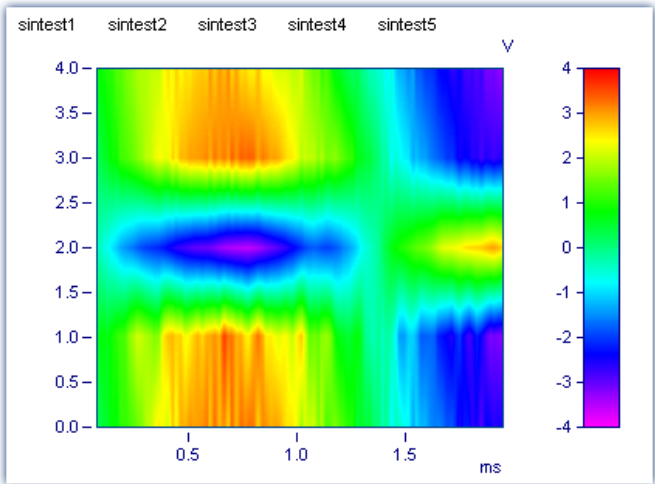
Another alternative is the waterfall diagram, which is especially suited to comparing similar curves. The curves are drawn in a diagonal procession from the front of the screen, with the curves at the back (partly) obscured by those at the front, creating the illusion of a three-dimensional perspective.



See "[Waterfall Diagrams](#)"

Color Map

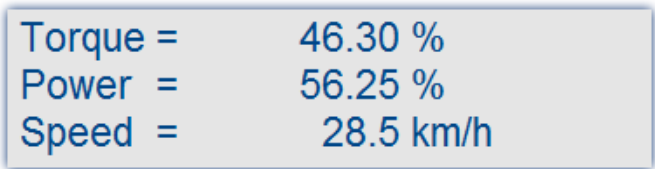
This is a bird's-eye-view of a colored, three-dimensional region. Multiple data sets are each plotted along the horizontal direction, stacked over each other along the y-direction, while their amplitude values are represented by different colors. This style of display is analogous to a geographical map where the landscape contours are indicated by shading or coloring.



See "[Color Map Display](#)"

Last value as number

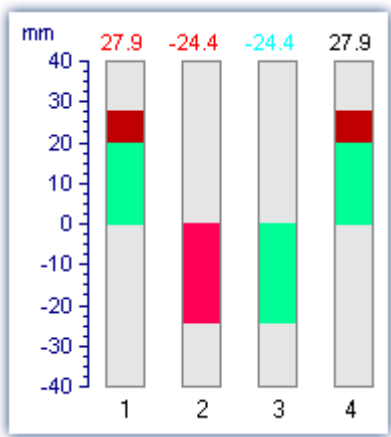
Particularly for online display of measured data, in this mode the last value is displayed as a number. The last value in a waveform is always the most recent measurement value in online display. This mode of display is particularly suited for slowly changing quantities. For example, the amplitude of a spectral line, effective power, or temperature can be monitored as a numerical value. The display of numerical values can be formatted. The corresponding numerical value is displayed for each curve in the curve window.



See "[Last Value As Number](#)" ⁶⁶⁵.

Bar meter

Particularly for online display of measured data, in this mode the last value is displayed in a bar meter.



Section "[Bar gauge](#)" ⁶⁷⁶

Table

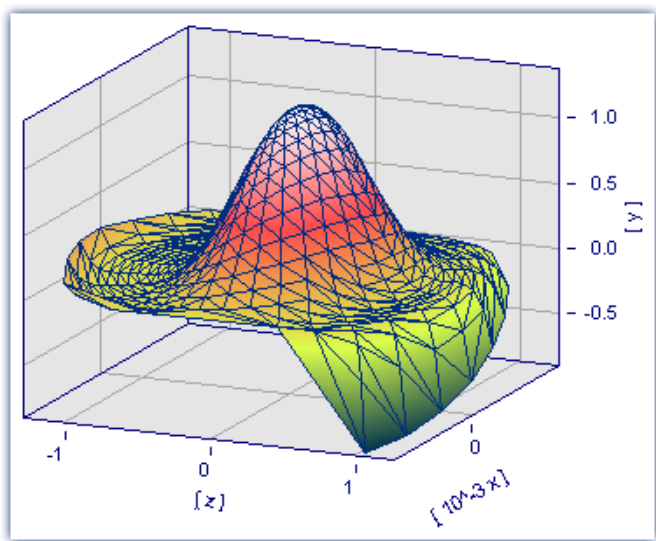
The readings are displayed in tabular form, in chronological order.

t [s]	sintest1 [V]
0.00000	-0.2544
0.00002	0.0349
0.00004	0.0259
0.00006	0.1652
0.00008	0.3037
0.00010	0.4410
0.00012	0.5766
0.00014	0.8592
0.00016	0.6923
0.00018	1.1185
0.00020	1.0945
0.00022	1.2168
0.00024	1.3388
0.00026	1.4670

Section "[Tabular display](#)" ⁶⁷⁰

3D

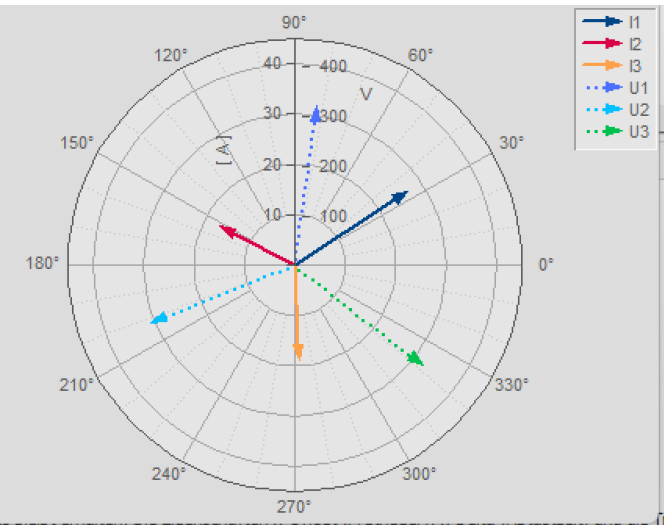
3-D display of waveforms having x,y,and z coordinates with various color map surface patterns, and rotating perspective in all dimensions.



Section on [3D display](#) ⁶⁹⁷

Polar plot

Complex data sets can be displayed as a polar diagram.



Complex single values as vector diagram



Section on ["Polar plot"](#) ⁶⁹⁸

Scale of x-axis

The following settings for scaling the x-axis can also be made using the x-axis' own settings dialog; see the section "[Axes](#)".

Selected x-unit

The x-axis is labeled in the x-unit of the waveform, e.g. in "s" (seconds) for measurement over time, or in "A" (amperes) for a characteristic dependent on the current. If several curves are displayed in the window, all waveforms should have the same x-unit.

For example, if a normal waveform with x-offset = 10s, x-delta = 1s and 20 samples is displayed, the x-axis is displayed from 10s to 30s and labeled accordingly.

Date/time: Absolute

For data in the time domain, instead of labeling of the value expressed in seconds, a display of the absolute date and time can be selected. The absolute time is not determined exclusively from the measurement points, but also gives consideration to the generation time of each waveform. This time is recorded in files in the imc FAMOS format and can be inquired and changed with imc FAMOS functions.

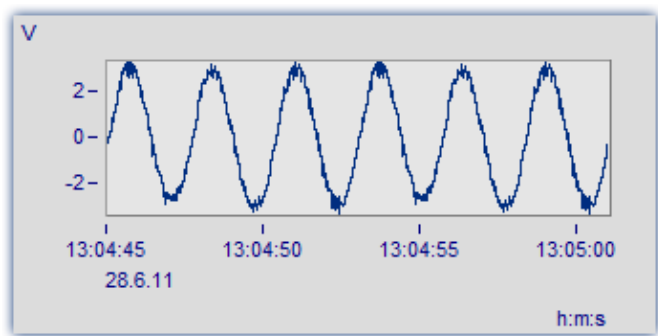
The generation time of a waveform along with the x-coordinate in "s" determines the resulting axis labels.

For example, if the generation time of a waveform is 1.1.92 at 12:00:00, and the waveform has an x-offset of 3600s and a sampling rate of 1800s for 48 measurement values, the range from

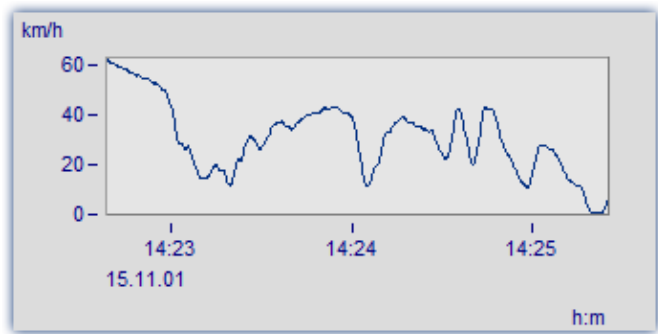
1.1.92, 13:00:00 to 2.1.92, 13:00:00

is displayed (3600s equals one hour). The waveform extends over 48 measurement values, each with a half-hour time difference, resulting in exactly one day of samples.

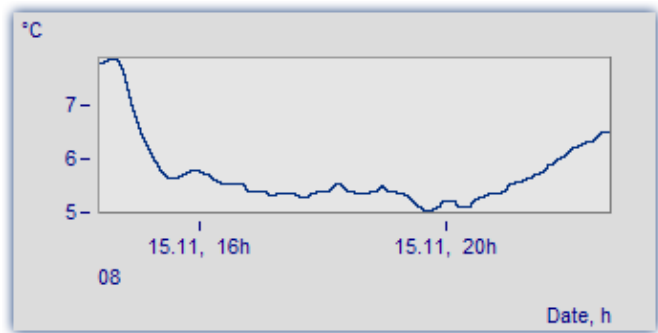
The labeling of the x-axis depends on the length of the displayed time interval, for example:



The segment between 13:04:45 and 13:05:00 encompasses several seconds.



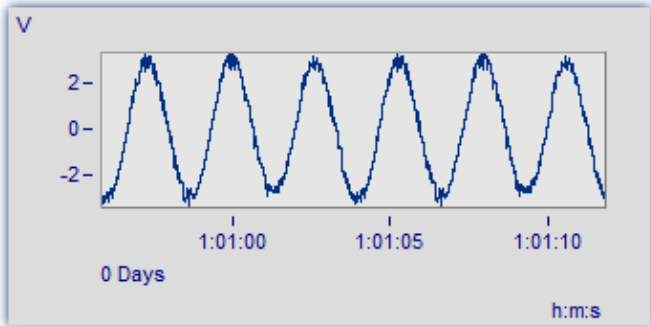
The segment between 14:23 and 14:25 comprises a few minutes.



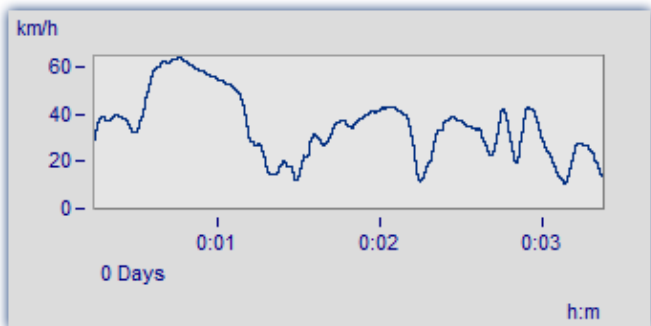
The segment between 4 am and 8 pm encompasses a few hours.

Days/hours/minutes: Relative

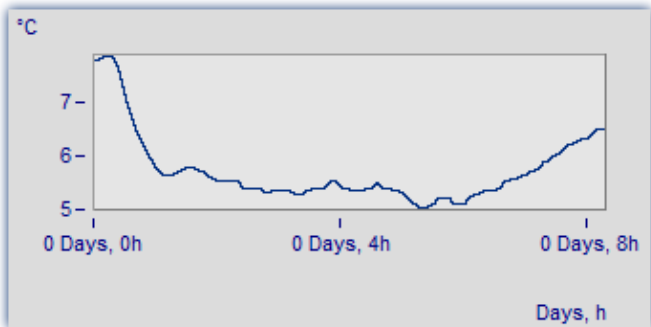
If you wish to view the whole duration of a long-term measurement, then display of relative time in days, hours, minutes and seconds is recommended. The selected unit depends on the interval. Possible variations of this labeling are as follows:



A very narrow section is shown, significantly less than one minute



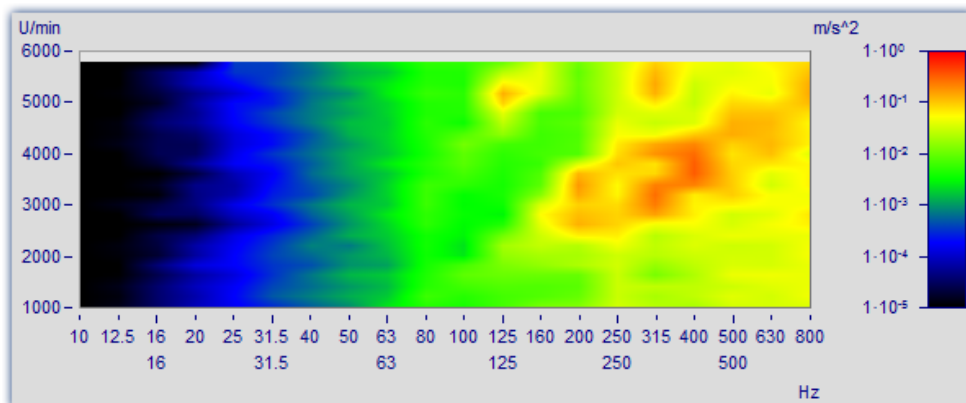
This section is 3 minutes long. The display no longer includes decimal places after the seconds.



The section from 0 to 8 hours is displayed without minutes

1/3-octave, octave labeling

When a waveform contains a 1/3-octave or an octave spectrum, and the x-axis of the waveform is scaled in 1/3-octave, the x-axis can be drawn with the numerical values of the 1/3-octave and octaves according to industry standards.



See "[Third-Octave Display](#)".

Further settings

Settings	Description
Axis labeling	Axis labels can be switched off completely, allowing the curve to be displayed in its maximal expanse. When curve windows are reduced significantly in size, the display automatically switches off display of the axis labels.
Show x=0 (trigger)	When this option is selected, a dotted vertical line is displayed in the curve window at x=0. The trigger is very often located at x=0.
Number trimming (useful when scrolling)	When this option is selected, the labeling at the ends of the axes are displayed partially, since the labeling box is too small to display it completely. When scrolling, the numbers "flow" into view. This option may also remain inactive if it is not needed.
Grid	Grid  as default setting.
Reference for dB value	All dB displays in a curve window have the same reference value. This value is "1" as a standard, but can be changed. Each number greater than zero in the valid range can be a reference value. For example, if a value of 10 is specified, all dB values on the axes become 20dB smaller. When decibels are calculated, values are first divided by this reference value. The displayed waveforms are not changed; only the axes labeling is adjusted.
Coordinate system size	Along with the <i>Default</i> -view, it is also possible to select a <i>Maximum</i> -view in which the axis labels are placed within the coordinate system and the coordinate system fills the entire curve window.

OK, Cancel, Set as default

For details on operation, see the chapter [Confirmation bar](#) .

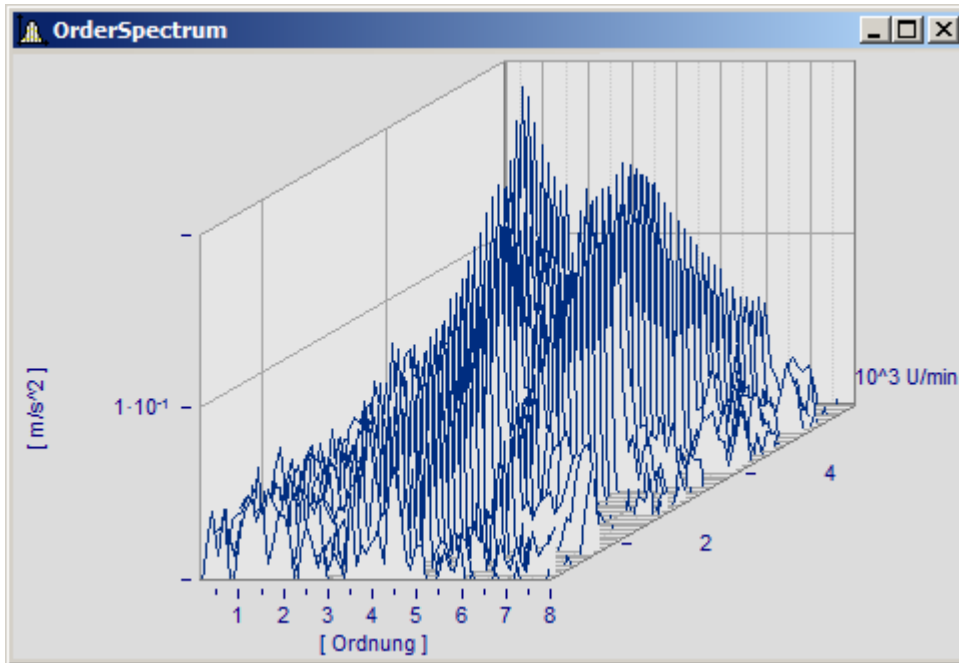
Remarks

Note

- If a waveform is not scaled in seconds, but is to be displayed in absolute or relative time, adjust the sampling time, the x-offset and the x-unit to provide the desired unit "s". For instance, if the waveform has a sampling time of 1h (1 hour), divide the sampling time by 3600 to set the x-unit to "s".
- The modes *Date/time: Absolute* and *Days/hours/minutes: Relative* cannot be implemented if an interval of less than 1ms is set for display of absolute or relative time. The program switches automatically to axis labeling in seconds.

11.6.1.1 Waterfall diagrams

The waterfall diagram serves as a simple tool for displaying a series of curves as arrayed one behind the other, and viewed from a vantage point that makes their line-up appear diagonal. This allows a close comparison of the individual curves, so that trends and deviations are easily recognized. Curves are drawn in a rectangular coordinate system with three axes (x, y, z). The z-axis has a directional component parallel to the angle of view. It is drawn diagonally toward the upper right of the screen at an angle specified by the user.



Not all curves are completely visible in waterfall display. The viewer sees a range of curves from above, with some valleys hidden by peaks. The relative angles and distances between the curves are not always apparent due to overlapping.

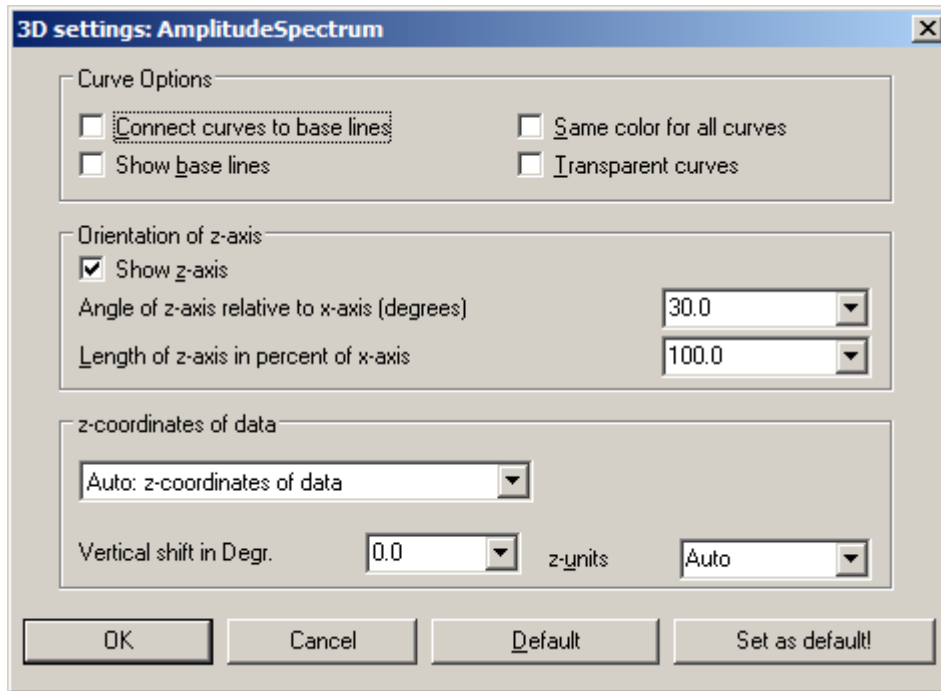
A waterfall display is drawn from back to front. When a new curve is drawn, the area under the curve is filled in with the color of the screen background, covering all lower parts of curves behind it.

The waterfall display is especially useful for comparing:

- multiple spectra; spectra are drawn at regular intervals, so that their development over time can be observed.
- multiple periods in a waveform.
- signals of several sensors on the same object, e.g. multiple temperature recorders arranged on a long pipe, delivering similar signals at different points in time.
- several consecutive measurements of a channel measured with the same trigger.

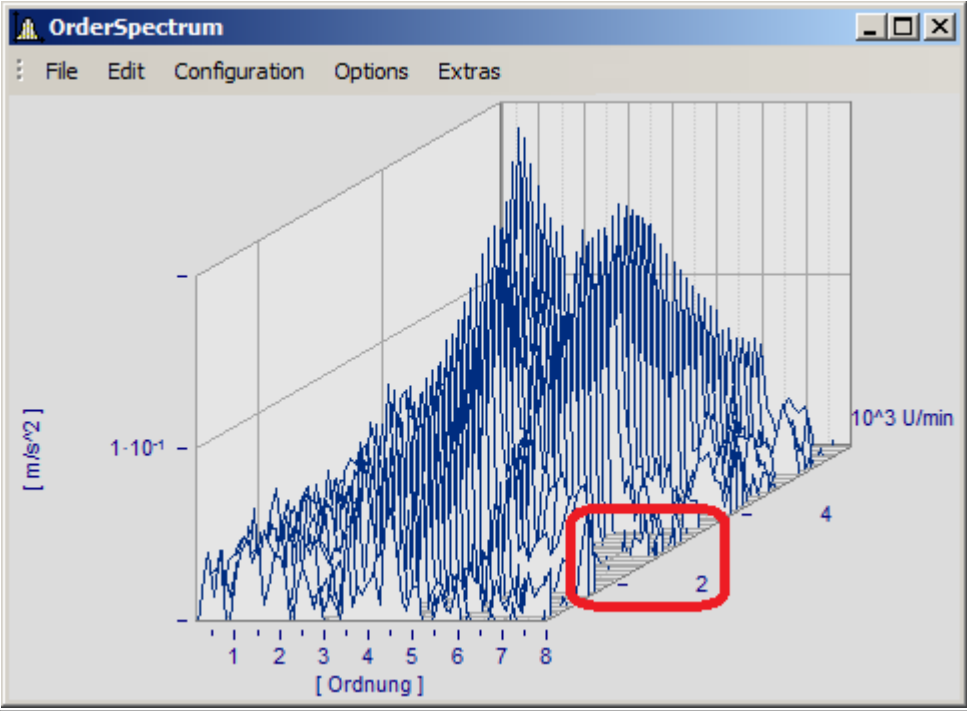
Mouse Operation

- Call the dialog for setting the type of display using the menu option *Configuration/ Display..* Select *Waterfall* and end the dialog with *OK*.
- The curves displayed in the window are shown in a waterfall diagram with the effective presets.
- A dialog is available to change the settings of the waterfall diagram. Select menu option *Configuration/ 3D....* The following dialog appears:



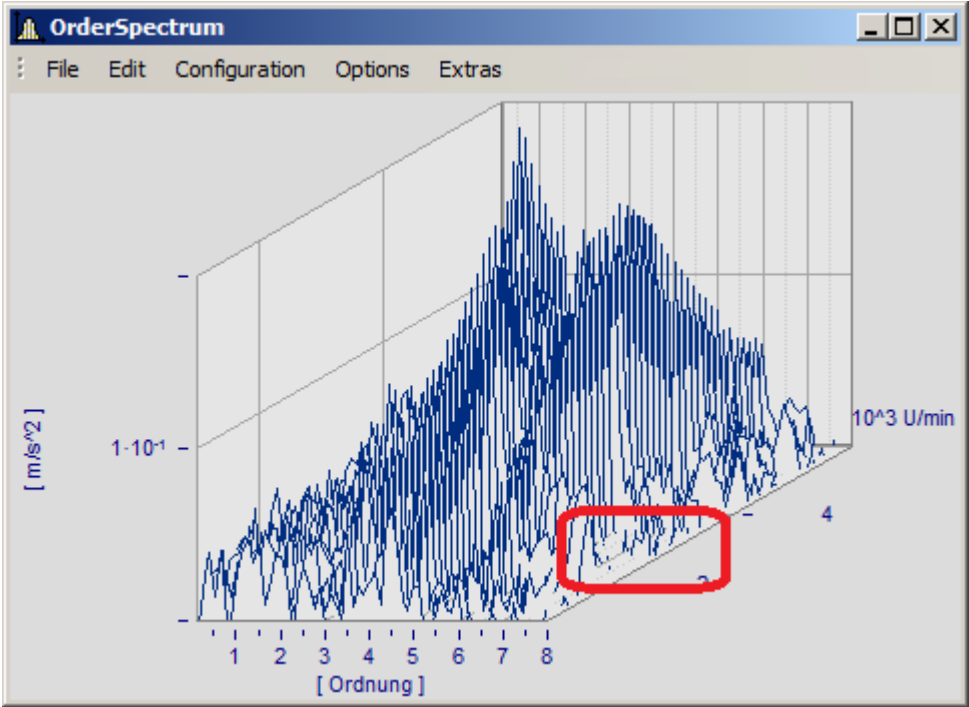
Curve options

Enter here the settings for the appearance of the curves in the window.

Parameter - Curve options	Description
Connect curves to base lines	This option connects the ends of the curve plot to the base line. The base line runs under the curve, parallel to and in the height of the x-axis. It is the projection of the curve on the plane defined by the labeled x- and z-axes. This option is not active in the following illustration: 

Parameter - Curve options	Description

Show base lines	Base lines are drawn when this option is selected. A base line runs under the curve, parallel to and at the height of the x-axis. It is a projection of the curve on the plane defined by the labeled x- and z-axes. Base lines are also obscured by curves in front of them. Base lines are present in the above illustration; the illustration below shows no base lines:



Same color for all curves	Select this option to draw all curves in the color of the first (foremost) curve. Otherwise the colors are assigned according to the selected color settings. The color composite is important for a three-dimensional appearance.

Transparent	When this option is selected, there are no hidden lines. However, the three-dimensional effect of the graph is reduced significantly. When this option is not selected, analog data sampled equidistantly are drawn in overwrite mode.

Orientation of z-axis

The options in this group refer to the display of the z-axis.

Parameter - Orientation of z-axis	Description

Show z-axis	Enter here whether the z-axis should be displayed.

Angle in degrees	The angle of the z-axis can be set to between 1 and 89 degrees. An angle of 30 degrees is recommended.

Length of z-axis in percentage of x-axis	The length of the z-axis is specified in percentage of the length of the x-axis. The minimum value is 10%

z-coordinate of curve

Select here between the options *Fixed values: 0,1,2,3,...* and *Fixed range z0, dz,....* In the former case, the first (foremost) curve takes the z-coordinate 0, the next takes 1, etc. The following settings are available when the latter option is selected:

Parameter - z-coordinate of curve	Description
z0	Enter the z-value here; the foremost curve will be displayed at this value.
dz	Enter the z-distance of adjacent curves in this box.
z-unit	A unit for the z-axis can be specified here.

Other options are also provided

Options	Description
Auto: z-coordinates of data	This employs a waveform's z-coordinate. In the imc FAMOS-dialog <i>Properties</i> of variables you can assign a z-coordinate to a waveform. The default is "0" The waveforms thus displayed must have unique and increasing z-coordinates. When handling structured waveforms (segmented, events..), this is the best choice. No further settings are necessary; the matrix will automatically be plotted with correct z-scaling.
Ditto with events: relative to first event	Only for multishot waveforms. The z-coordinate is determined from the various trigger times. The first shot is assigned the time 0 and all subsequent ones are designated accordingly. The other, related options listed make reference to other shots (events). E.g., the last, the first displayed, or the last displayed.  The z-coordinate of each waveform absolutely must be greater than the last. Otherwise the display will not include all waveforms!
z-axis scaling	To scale the z-axis, double click on this option. Settings for the x- and y-axes, as well as for ticks and marking of the range can be made here.

 Reference	Scaling the x-axis

Section '[Scaling the x-axis](#)'  736

Default

Clicking on this button sets all dialog elements to default values.

OK, Cancel, Set as default

For details on operation, see the chapter [Confirmation bar](#) 

Remarks

- Select an appropriate viewing angle. An angle of 30 degrees is a good starting point.
- A distorted display results if the angle is very small or very large; often the axes can no longer be labeled.
- The axes are labeled only when the coordinate system is large enough in every direction, i.e. when each axis exceeds a certain length. Enlarge the curve window if no labels appear. If there are still no labels, make any necessary changes to the angle and then the length of the z-axis. If an extremely long z-axis results, the window often has no room left for the labels.
- The area below the curve is **NOT** filled for XY-plots, digital data and reduced data. These displays always use the *Transparent* mode.
- Waterfall diagrams are often more legible when the y-axis is scaled upward very generously. Assume that the curves displayed have a value range from 3 to 12, the y-axis could be scaled from 3 to 30 (or 20 or 40). The axes need not be extended downward, since this <Shift>s the curves upward. Depending on the kind of data, appropriate y-axis scaling can greatly improve legibility.
- Not all data can be represented in a waterfall diagram. For example, noisy data can produce a chaotic image of overlapping lines. Waveforms which are relatively smooth and similar are best for waterfall display.
- When several curves are displayed, the first curve is always drawn in the foreground of the window; all additional waveforms appear behind the first curve.
- Waveforms with different x-axis scaling can be displayed together (i.e. waveforms sampled at different speeds). All curves are displayed correctly with the respect to time on the x-axis, as in all other curve windows.
- Only one y-axis is used for all curves in a waterfall diagram.
- A line display is always implemented for waterfall diagram, even if data were represented with symbols. The symbols are not displayed. The selection of *representation with symbols* is recorded and will be reactivated whenever the waterfall mode is ended.
- Waveforms displayed with a dotted line are interpreted as transparent in the waterfall display.
- Curve display settings such as dots, bar graphs or stair-steps can be set for each waveform individually, or for all waveforms in a window with the option *Valid for all axes* selected in the dialog for scaling the y-axis.
- Only transparent graphics can be outputted to plotters. The algorithm used here, which draws lines and subsequently draws over them with the next plane of display, does not produce the desired effect with plotters.
- Drawing lines which are then covered by the next graphics takes considerably more time than drawing transparent graphics. First try to output transparent graphics before experimenting with various settings and scaling methods.

11.6.1.2 Last value as number

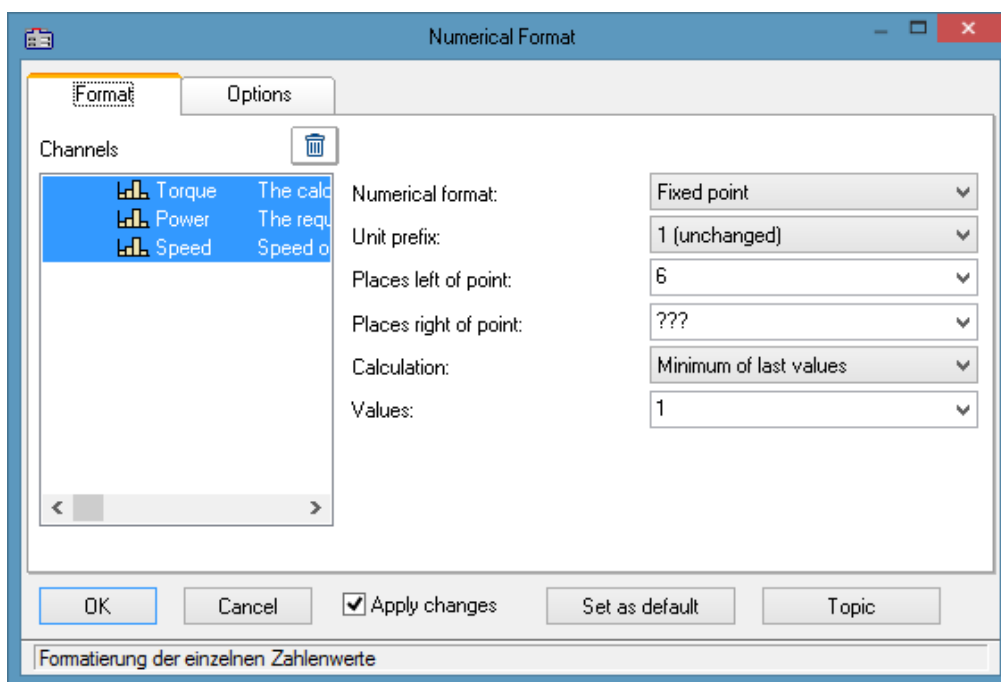
Torque =	46.30 %
Power =	56.25 %
Speed =	28.5 km/h

This mode, displaying the last value in a waveform as a numerical value, is especially suited for use of the curve manager in data acquisition and on-line display of these data. The last value in a waveform is always the most recent measurement value in on-line display. This mode is especially convenient for display of slowly changing quantities, allowing the user to monitor the amplitude of a spectral line, the effective power or a temperature as a numerical value. The display of the numerical value can be formatted and several channels can be compared with each other.

Mouse Operation

- Select *Configuration/ Display...*
- Select the option *Last value as number* and end the dialog with *OK*
- The curve window now shows the variable's name along with a numerical value. If you display more curves in the window, multiple numerical values appear stacked one above the other:

The format of this display can be changed by double-clicking on the curve window or selecting *Configuration/ Numerical Format....* The following dialog appears for defining the number format:



Waveforms

A list of all waveforms displayed in the window is found on the left side of the dialog. Select waveforms from this list by moving the mouse to the desired waveform's listing while holding the mouse button. Multiple waveforms can be selected by holding down the <Ctrl> button and clicking on the desired waveforms.

Format (For all selected waveforms)

All settings made here only affect waveforms in the *Waveforms* list.

If the the waveforms selected have differing format settings, question marks appear in the associated boxes. Any settings you enter to replace the question marks will be applied to all waveforms selected.

Settings - Format	Description										
Numerical format	Representation as <i>fixed point</i> is customary. But the numerical values can also be formatted as <i>Floating point</i> , in <i>Hex-format</i> , or as <i>Date/Time</i> . <table> <tr> <th>Numerical format</th><th>Description</th></tr> <tr> <td>Floating point</td><td><i>Floating point</i>-representation uses the E-character to express the exponent, followed by the sign and a two-digit exponent.</td></tr> <tr> <td>Hex (1, 2, 4 Byte)</td><td>Hex always displays the lowest selected Bytes, thus for example, the 1027 and 1 Byte is represented as 03h.</td></tr> <tr> <td>Time relative-, Date/time auto</td><td>The point in time when the last sample was recorded is formatted in accordance with the display option applicable to the x-axis ⁶⁴⁹.</td></tr> <tr> <td>Time relative-, Date/time fixed</td><td>The formatting of the time and date is specified by means of placeholders, see the chapter "Scale" ⁷³⁷.</td></tr> </table>	Numerical format	Description	Floating point	<i>Floating point</i> -representation uses the E-character to express the exponent, followed by the sign and a two-digit exponent.	Hex (1, 2, 4 Byte)	Hex always displays the lowest selected Bytes, thus for example, the 1027 and 1 Byte is represented as 03h.	Time relative-, Date/time auto	The point in time when the last sample was recorded is formatted in accordance with the display option applicable to the x-axis ⁶⁴⁹ .	Time relative-, Date/time fixed	The formatting of the time and date is specified by means of placeholders, see the chapter " Scale " ⁷³⁷ .
Numerical format	Description										
Floating point	<i>Floating point</i> -representation uses the E-character to express the exponent, followed by the sign and a two-digit exponent.										
Hex (1, 2, 4 Byte)	Hex always displays the lowest selected Bytes, thus for example, the 1027 and 1 Byte is represented as 03h.										
Time relative-, Date/time auto	The point in time when the last sample was recorded is formatted in accordance with the display option applicable to the x-axis ⁶⁴⁹ .										
Time relative-, Date/time fixed	The formatting of the time and date is specified by means of placeholders, see the chapter " Scale " ⁷³⁷ .										
Magnitude	Numerical values can be associated with a fixed order of magnitude. Take as an example a current of 0.1A. To display the current in mA, select <i>milli</i> . The numerical value is then multiplied by 1000 and the prefix milli (m) is set in front of the unit, if it is not empty. Selectable prefixes range from pico (p) to giga (G). Select 1 if numerical values and unit should be displayed unchanged. Selecting a fixed order of magnitude is recommended for fixed-comma notation; if numbers vary greatly in their order, it is recommended to use floating-point notation with unchanged order.										
Fixed-point, floating-point notation	Floating point notation uses the exponential notation, "e" and a two-digit signed power of 10. Fixed-point example: 0.123, -123, 888.987 Floating-point example: 1.45E+03, -1E-01, 1.4444E+00 Select Yes for Fixed-point or no for Floating point. With floating point, one digit always appears before the decimal.										
Places left of point	The amount of digits before the decimal point, max. 15. With Fixed-point notation, sufficient digits must always be specified, otherwise a black bar appears. With floating-point format, one digit before the decimal point is sufficient.										
Places right of point	The amount of digits after the decimal point, between 0 and 20. If no digits (0) appear after the decimal point, it is omitted.										

Settings - Format	Description								
Calculation	The numerical value displayed can be calculated from the most recent values: <table><tr><th>Calculation</th><th>Description</th></tr><tr><td>Maximum of the last values</td><td>The maximum is calculated from the last n values in the waveform. This function can be interpreted in on-line-display as peak hold.</td></tr><tr><td>Minimum of the last values</td><td>The minimum is calculated from the last n values in the waveform.</td></tr><tr><td>Average of last values</td><td>The arithmetic mean is calculated from the last n values in the waveform. The reduces possible noise in display of the measurement data.</td></tr></table>	Calculation	Description	Maximum of the last values	The maximum is calculated from the last n values in the waveform. This function can be interpreted in on-line-display as peak hold.	Minimum of the last values	The minimum is calculated from the last n values in the waveform.	Average of last values	The arithmetic mean is calculated from the last n values in the waveform. The reduces possible noise in display of the measurement data.
Calculation	Description								
Maximum of the last values	The maximum is calculated from the last n values in the waveform. This function can be interpreted in on-line-display as peak hold.								
Minimum of the last values	The minimum is calculated from the last n values in the waveform.								
Average of last values	The arithmetic mean is calculated from the last n values in the waveform. The reduces possible noise in display of the measurement data.								
Values	Specify here the number of values from which the average, minimum or maximum is calculated. Any number from 1 to 30000 can be specified. If the waveform has fewer values than specified here, all values in the waveform are used for the calculation. Note that with larger numbers, the calculation must be performed for display of the curves. When 1 is specified, the result of calculating the minimum, maximum and mean value is always the last value in the waveform.								

Options (Curve Window Defaults)

Format

Options

Font:

Arial

☒ Size automatic

☒ =

☒ Columns

☒ Flush right

Names

Yes

Settings made in this group of elements are effective for the entire curve window and are not made individually for each waveform. The font and several other basic display types can be set here.

Settings - Options	Description
Font	For selecting the font, font size and style. The size set here is ignored if the option Size: auto is activated.
Size automatic	The font size is automatically selected to be as large as will fit. However, only the size is adjusted.
Columns	The name, equals sign, numerical values, and units are displayed in columns for a clearer presentation. Without the option <i>Columns</i> , the entries are separated by spaces.
"=" (equal sign)	An equal sign appears after the variable's name, or else a colon. The colon appears directly with the name, but the equal sign appears in its own column.
Flush right	All numerical values appear flush along the right margin, independently of the format set. Without <i>Flush right</i> , all decimal periods are aligned one above the other. The condition for the aligned writing is that the <i>Columns</i> option is selected!
Names	Variables names and or comments which appear before the numerical value. If only one variable is in the window, the designator can be switched off since it is already in the title bar.

Set as default

The settings made initially only apply to the current window. If you click on the button *Set as Default*, any new curve windows created apply these settings whenever the display stye *Last value as number* is selected. You should only click on this button if no question marks are showing in the dialog.

The defaults for the format are **NOT** saved for each data set individually.

OK, Cancel

For details on operation, see the chapter [Confirmation bar](#)⁸⁰⁹.

Remarks

- Several menu items in the curve window, such as Measure, Scale and Overview Window cannot be selected when the numerical value display option is selected.
- Displaying numbers in fixed-point notation may have two effects in connection with the range of numbers. If the size of a number is too small, only zeros are displayed. On the other hand, when a number is too large to be displayed in the available space, a symbol appears which indicates an overflow. In this case, select more digits left of the decimal point or floating-point notation.
- With numerical display of online measurement data at high sampling rates, it can occur that the numbers change so fast that they are no longer visible. In that case, select a calculation and control the update rate for the values by making the appropriate setting under Values. Then select an appropriately small number of places right of the decimal. Usually only the digits to the right of the decimal point change strongly because the measurement signal changes only slightly.
- When time-stamped data are subjected to data reduction, the amount of values on which the calculation is performed is the same as the amount of measured points before the data reduction. Thus, specifying the amount of values to which the calculation is applied always fixes the time segment on which the calculation is performed, independently of how much the measured data are reduced.
- The color of the texts and the background in numerical display can be changed uniformly for all curve windows in the dialog *Options/ Colors....* The following elements are available for selection:
 - *Numbers: foreground or background*

11.6.1.2.1 Single value display

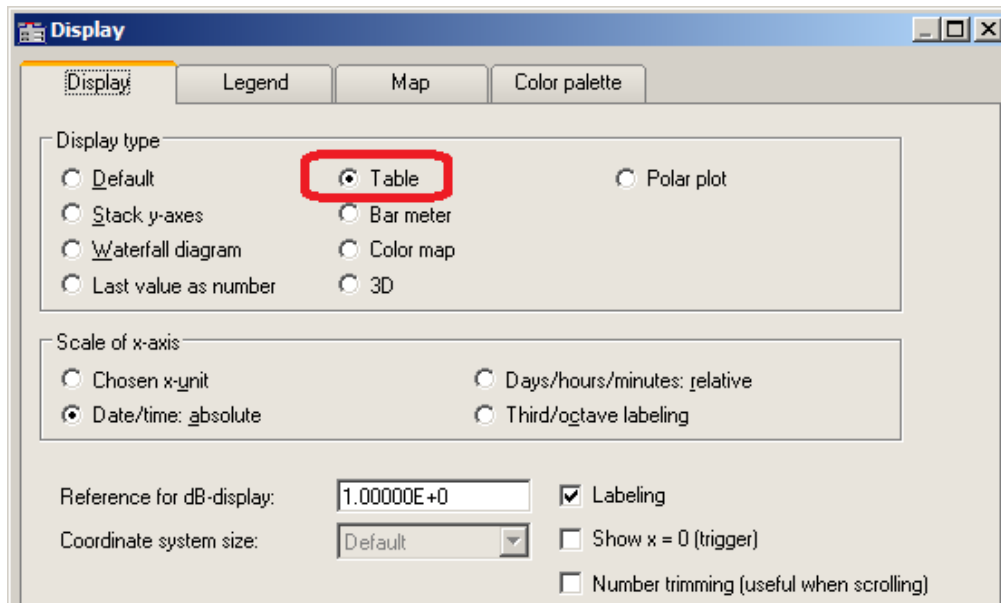
The display mode *Last value as number* is automatically activated if real waveforms with a length of 1 are displayed.

11.6.1.3 Tabular representation

Measurement readings can also be displayed in a table, where the readings are listed in chronological order. This can appear as a protocol readout or as a log book.

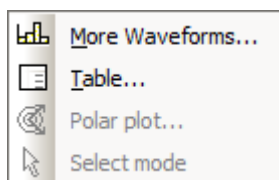
Operation

Select the curve window menu item *Configuration / Display...*. This calls a dialog for defining the curve window's display style. Select the entry, *Table*. Even the time display types *Chosen x-unit* and *Date/time absolute*, are applied in a tabular display.



The curve window then appears as shown below:

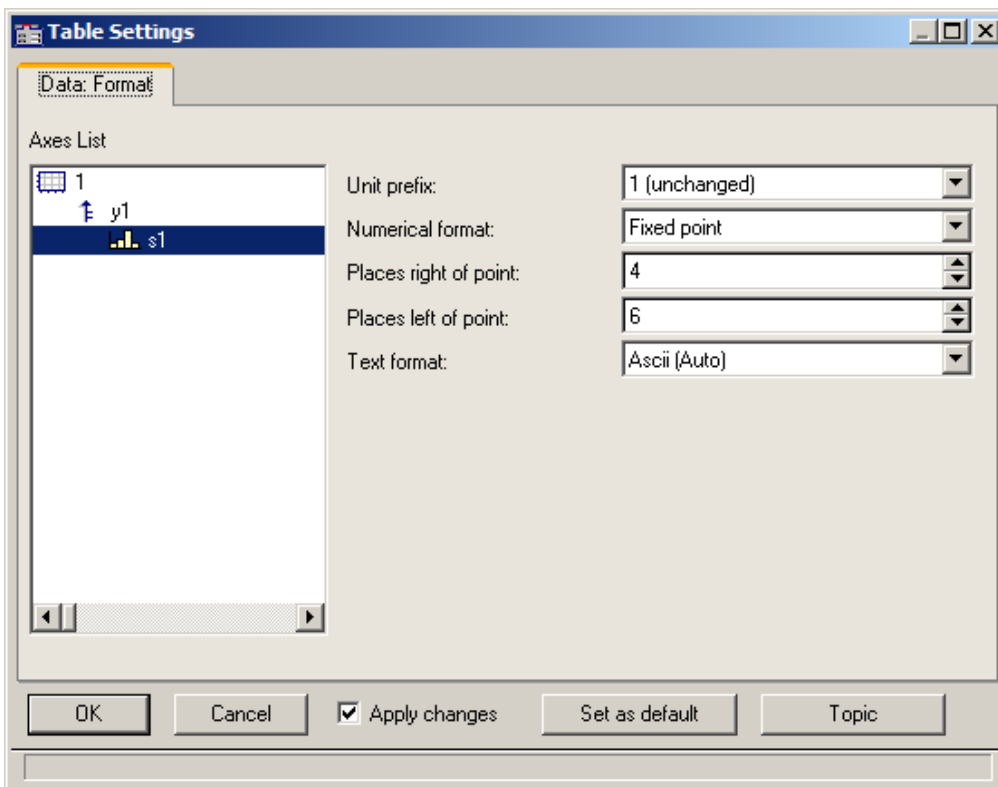
t ["order"]	OrderSpectrum [m/s ²]
0.125	0.0092
0.125	0.0157
0.125	0.0161
0.125	0.0101
0.125	0.0083
0.125	0.0150
0.125	0.0194
0.125	0.0107
0.125	0.0142



It's possible to move through the list using the scrollbar at right.

Right-clicking the mouse on the curve window calls the context menu which now looks like this.

The menu item *Table...* calls the dialog offering controls as shown below:

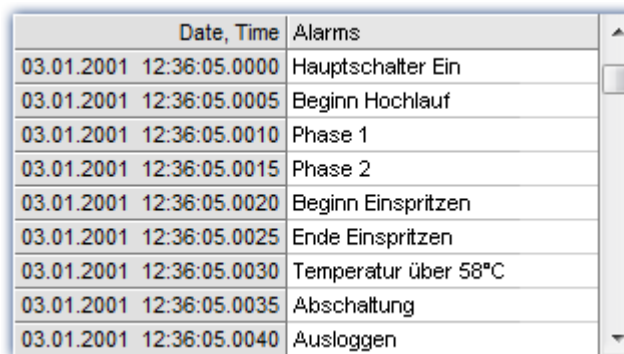


The left portion of the dialog indicates all the channels displayed in the curve window. How these channel listings are arranged in terms of axes and coordinate systems doesn't matter here. On the right side of the dialog, the controls are located, which are for making settings for the channels selected in the left part.

11.6.1.3.1 Settings

If the curve window displays channels with measurement readings (default), then you need to deal with the settings for the *numerical format*, namely *Unit prefix*, *Numerical format*, *Places*, *right/left of point*. If texts are displayed (time stamp-ASCII data), then the text format becomes relevant.

Settings	Description
Unit prefix	This consists of the order of magnitude and, if appropriate, a prefix such as kilo, milli, Mega or even just 1 (default, use of unit without prefix), which appears after the number before the unit. If the reading is 10 and takes the unit V (Volts), then for a selection of "m" (milli), a value of 10000mV is returned.
Numerical format	Fixed point can be selected for readout with a fixed number of decimal places and without an exponent; e.g. 17, 17.35, -0.0017. Floating point can be selected for readout with fixed relative precision and an exponent, e.g. -1.28E-7 or 3.4E+0
Places right of point	This is the number of decimal places to show. 0..15 places are possible.
Places left of point	Only for fixed point format, a specified amount of places before (to the left of) the decimal point. The number should be large enough to ensure that the numerical value is indicated with all its digits.
Text format	This setting only pertains to channels with information presented as strings, in other words on the case of time stamp-Ascii data. Such data are, for instance, generated by imc Online FAMOS , but also by functions belonging to the Time Stamp Ascii Kit (TSA-Kit). This data type contains texts, where each text has a time stamp, e.g. log-book entry representation with absolute time selected:



Date, Time	Alarms
03.01.2001 12:36:05.0000	Hauptschalter Ein
03.01.2001 12:36:05.0005	Beginn Hochlauf
03.01.2001 12:36:05.0010	Phase 1
03.01.2001 12:36:05.0015	Phase 2
03.01.2001 12:36:05.0020	Beginn Einspritzen
03.01.2001 12:36:05.0025	Ende Einspritzen
03.01.2001 12:36:05.0030	Temperatur über 58°C
03.01.2001 12:36:05.0035	Abschaltung
03.01.2001 12:36:05.0040	Ausloggen

Text format

The following options are available:

Text format	Description
Ascii (Auto)	Display of ASCII characters in legible form (see above.).

Text format	Description

Hex	Display of hexadecimal values 00H .. FFH. <table><tr><th>t [s]</th><th>Alarms</th></tr><tr><td>-4.8</td><td>48 61 75 70 74 73 63 68 61 6c 74 65 72 20 45 69 6e</td></tr><tr><td>-4.1</td><td>42 65 67 69 6e 6e 20 48 6f 63 68 6c 61 75 66</td></tr><tr><td>-4.1</td><td>50 68 61 73 65 20 31</td></tr><tr><td>-1.1</td><td>50 68 61 73 65 20 32</td></tr><tr><td>2.3</td><td>42 65 67 69 6e 6e 20 45 69 6e 73 70 72 69 74 7a 65 6</td></tr><tr><td>4.6</td><td>45 6e 64 65 20 45 69 6e 73 70 72 69 74 7a 65 6e</td></tr><tr><td>6.7</td><td>54 65 6d 70 65 72 61 74 75 72 20 fc 62 65 72 20 35 3</td></tr><tr><td>10.3</td><td>41 62 73 63 68 61 6c 74 75 6e 67</td></tr><tr><td>12.4</td><td>41 75 73 6c 6f 67 67 65 6e</td></tr></table>	t [s]	Alarms	-4.8	48 61 75 70 74 73 63 68 61 6c 74 65 72 20 45 69 6e	-4.1	42 65 67 69 6e 6e 20 48 6f 63 68 6c 61 75 66	-4.1	50 68 61 73 65 20 31	-1.1	50 68 61 73 65 20 32	2.3	42 65 67 69 6e 6e 20 45 69 6e 73 70 72 69 74 7a 65 6	4.6	45 6e 64 65 20 45 69 6e 73 70 72 69 74 7a 65 6e	6.7	54 65 6d 70 65 72 61 74 75 72 20 fc 62 65 72 20 35 3	10.3	41 62 73 63 68 61 6c 74 75 6e 67	12.4	41 75 73 6c 6f 67 67 65 6e
t [s]	Alarms																				
-4.8	48 61 75 70 74 73 63 68 61 6c 74 65 72 20 45 69 6e																				
-4.1	42 65 67 69 6e 6e 20 48 6f 63 68 6c 61 75 66																				
-4.1	50 68 61 73 65 20 31																				
-1.1	50 68 61 73 65 20 32																				
2.3	42 65 67 69 6e 6e 20 45 69 6e 73 70 72 69 74 7a 65 6																				
4.6	45 6e 64 65 20 45 69 6e 73 70 72 69 74 7a 65 6e																				
6.7	54 65 6d 70 65 72 61 74 75 72 20 fc 62 65 72 20 35 3																				
10.3	41 62 73 63 68 61 6c 74 75 6e 67																				
12.4	41 75 73 6c 6f 67 67 65 6e																				

CAN, LIN message	For all devices with a CAN/LIN-connection: display with identifier and contents of the CAN/LIN-messages. The identifier (appearing before the colon) and the bytes belonging to the message are displayed as hexadecimal values. <table><tr><th>t [s]</th><th>CanBus_Messages</th></tr><tr><td>1.3</td><td>034: 00 00 00</td></tr><tr><td>2.9</td><td>034: 78 88 0F</td></tr><tr><td>3.4</td><td>129: 00 00 00 00 11 11 00 00</td></tr><tr><td>3.5</td><td>001: 01</td></tr><tr><td>4.4</td><td>034: 90 90 00</td></tr><tr><td>4.9</td><td>034: 78 88 0F</td></tr><tr><td>5.4</td><td>129: 00 00 00 00 11 11 00 00</td></tr><tr><td>6.5</td><td>001: 01</td></tr></table>	t [s]	CanBus_Messages	1.3	034: 00 00 00	2.9	034: 78 88 0F	3.4	129: 00 00 00 00 11 11 00 00	3.5	001: 01	4.4	034: 90 90 00	4.9	034: 78 88 0F	5.4	129: 00 00 00 00 11 11 00 00	6.5	001: 01
t [s]	CanBus_Messages																		
1.3	034: 00 00 00																		
2.9	034: 78 88 0F																		
3.4	129: 00 00 00 00 11 11 00 00																		
3.5	001: 01																		
4.4	034: 90 90 00																		
4.9	034: 78 88 0F																		
5.4	129: 00 00 00 00 11 11 00 00																		
6.5	001: 01																		

 **Note**

Messages with extended identifier

In [Hex](#)⁶⁷³ representation of the content of a message, the extended bit is always visible.

In case of CAN, LIN message format, the bit is taken into account to output 4 or 8 hex characters for the identifier. But in the output table the bit is hidden.

4 byte abs. time	For all devices which can read in 4 Byte time stamp data (seconds since 1.1.1980), there is a specialized display style. <table><tr><th>t [s]</th><th>TimeStampUmsx4Byte</th></tr><tr><td>6.0</td><td>14.04.2001 19:41:55</td></tr><tr><td>7.0</td><td>14.04.2001 19:41:55</td></tr><tr><td>8.0</td><td>14.04.2001 19:41:55</td></tr><tr><td>9.0</td><td>14.04.2001 19:41:55</td></tr><tr><td>10.0</td><td>14.04.2001 19:41:55</td></tr><tr><td>11.0</td><td>14.04.2001 19:41:55</td></tr><tr><td>12.0</td><td>14.04.2001 19:41:55</td></tr><tr><td>13.0</td><td>14.04.2001 19:41:55</td></tr></table>	t [s]	TimeStampUmsx4Byte	6.0	14.04.2001 19:41:55	7.0	14.04.2001 19:41:55	8.0	14.04.2001 19:41:55	9.0	14.04.2001 19:41:55	10.0	14.04.2001 19:41:55	11.0	14.04.2001 19:41:55	12.0	14.04.2001 19:41:55	13.0	14.04.2001 19:41:55
t [s]	TimeStampUmsx4Byte																		
6.0	14.04.2001 19:41:55																		
7.0	14.04.2001 19:41:55																		
8.0	14.04.2001 19:41:55																		
9.0	14.04.2001 19:41:55																		
10.0	14.04.2001 19:41:55																		
11.0	14.04.2001 19:41:55																		
12.0	14.04.2001 19:41:55																		
13.0	14.04.2001 19:41:55																		

Text format	Description																				
3-number sets (hex)	Display of hexadecimal values in 3-number sets.<table><tr><th>t [s]</th><th>Hex_3er_Messages_S1_K2</th></tr><tr><td>2.70843</td><td>64 00 00 00 00 00 00 00 f e f f</td></tr><tr><td>2.72640</td><td>65 00 00 00 00 00 f e f f 00 00</td></tr><tr><td>2.79394</td><td>66 00 00 00 22 00</td></tr><tr><td>2.80843</td><td>64 00 00 00 00 00 00 00 f e f f</td></tr><tr><td>2.82641</td><td>65 00 00 00 00 00 f e f f 00 00</td></tr><tr><td>2.89394</td><td>66 00 00 00 23 00</td></tr><tr><td>2.90843</td><td>64 00 00 00 00 00 00 00 f e f f</td></tr><tr><td>2.92641</td><td>65 00 00 00 00 00 f e f f 00 00</td></tr><tr><td>2.99396</td><td>66 00 00 00 24 00</td></tr></table>	t [s]	Hex_3er_Messages_S1_K2	2.70843	64 00 00 00 00 00 00 00 f e f f	2.72640	65 00 00 00 00 00 f e f f 00 00	2.79394	66 00 00 00 22 00	2.80843	64 00 00 00 00 00 00 00 f e f f	2.82641	65 00 00 00 00 00 f e f f 00 00	2.89394	66 00 00 00 23 00	2.90843	64 00 00 00 00 00 00 00 f e f f	2.92641	65 00 00 00 00 00 f e f f 00 00	2.99396	66 00 00 00 24 00
t [s]	Hex_3er_Messages_S1_K2																				
2.70843	64 00 00 00 00 00 00 00 f e f f																				
2.72640	65 00 00 00 00 00 f e f f 00 00																				
2.79394	66 00 00 00 22 00																				
2.80843	64 00 00 00 00 00 00 00 f e f f																				
2.82641	65 00 00 00 00 00 f e f f 00 00																				
2.89394	66 00 00 00 23 00																				
2.90843	64 00 00 00 00 00 00 00 f e f f																				
2.92641	65 00 00 00 00 00 f e f f 00 00																				
2.99396	66 00 00 00 24 00																				
Flexray-message	For all devices with a Flexray-connection: display with identifier and contents of the Flexray-messages. The identifier (appearing before the colon) and the bytes belonging to the message are displayed as hexadecimal values.<table><tr><th>t [s]</th><th>Flexray_Messages_S1_K2</th></tr><tr><td>33.81970</td><td>65 0000 00: 00 00 00 00 f e f f 00 00</td></tr><tr><td>33.88721</td><td>66 0000 00: 2d 00</td></tr><tr><td>33.90171</td><td>64 0000 00: 00 00 00 00 00 00 f e f f</td></tr><tr><td>33.91970</td><td>65 0000 00: 00 00 00 00 f e f f 00 00</td></tr><tr><td>33.98723</td><td>66 0000 00: 2e 00</td></tr><tr><td>34.00171</td><td>64 0000 00: 00 00 00 00 00 00 f e f f</td></tr><tr><td>34.01970</td><td>65 0000 00: 00 00 00 00 f e f f 00 00</td></tr><tr><td>34.08724</td><td>66 0000 00: 2f 00</td></tr><tr><td>34.10171</td><td>64 0000 00: 00 00 00 00 00 00 f e f f</td></tr></table>	t [s]	Flexray_Messages_S1_K2	33.81970	65 0000 00: 00 00 00 00 f e f f 00 00	33.88721	66 0000 00: 2d 00	33.90171	64 0000 00: 00 00 00 00 00 00 f e f f	33.91970	65 0000 00: 00 00 00 00 f e f f 00 00	33.98723	66 0000 00: 2e 00	34.00171	64 0000 00: 00 00 00 00 00 00 f e f f	34.01970	65 0000 00: 00 00 00 00 f e f f 00 00	34.08724	66 0000 00: 2f 00	34.10171	64 0000 00: 00 00 00 00 00 00 f e f f
t [s]	Flexray_Messages_S1_K2																				
33.81970	65 0000 00: 00 00 00 00 f e f f 00 00																				
33.88721	66 0000 00: 2d 00																				
33.90171	64 0000 00: 00 00 00 00 00 00 f e f f																				
33.91970	65 0000 00: 00 00 00 00 f e f f 00 00																				
33.98723	66 0000 00: 2e 00																				
34.00171	64 0000 00: 00 00 00 00 00 00 f e f f																				
34.01970	65 0000 00: 00 00 00 00 f e f f 00 00																				
34.08724	66 0000 00: 2f 00																				
34.10171	64 0000 00: 00 00 00 00 00 00 f e f f																				

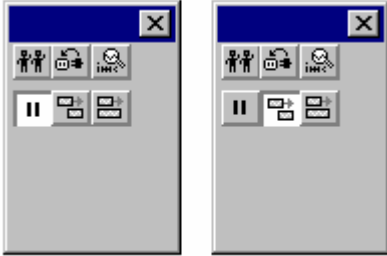
Multi-channel displays

Display of multiple channels at once also uses correctly synchronized ordering. For every measurement reading, the source channel is indicated:

t [s]	Channel	Value
0.0000	s2	0.8399 m/s^2
0.0000	s1	0.1601 m/s^2
0.0005	s2	0.4382 m/s^2
0.0005	s1	0.5618 m/s^2
0.0010	s2	0.7121 m/s^2
0.0010	s1	0.2879 m/s^2
0.0015	s2	0.5478 m/s^2
0.0015	s1	0.4522 m/s^2

Online-Display

If a table is displayed during a running measurement, it may be advantageous to have a readout of the current readings, at the bottom of the table. For this purpose, the roll-mode is provided, just like for time-based displays in coordinate systems.



OK, Cancel, Set as default

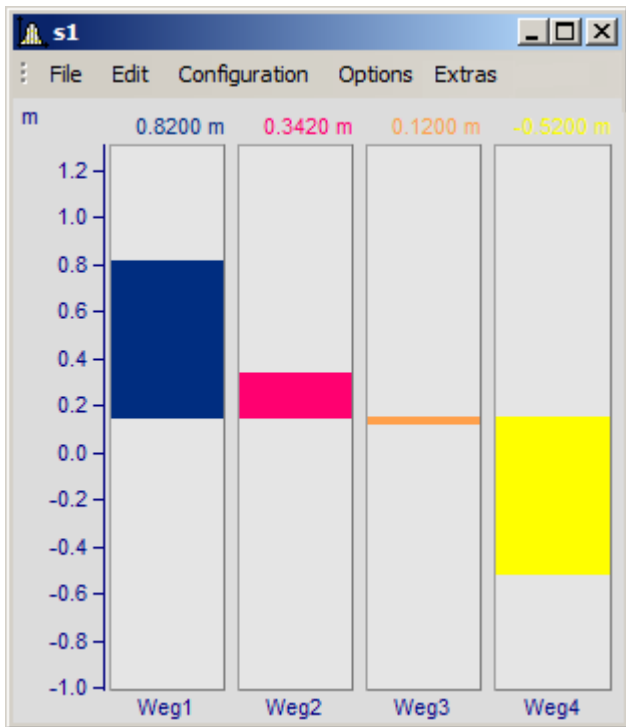
For details on operation, see the chapter [Confirmation bar](#) ⁸⁰⁹

Remarks

- The entries are always ordered chronologically.
- The condition for using this display type is that the time coordinate in each channel behaves like time, in other words, increases continuously.
- If the time coordinate decreases in a data set, for example, in the case of a XY-waveform, this display type cannot be used.
- For information on the buttons *Ok*, *Cancel* and *Set as default*, refer to the chapter on [Confirmation bar](#) ⁸⁰⁹.

11.6.1.4 Bar meter

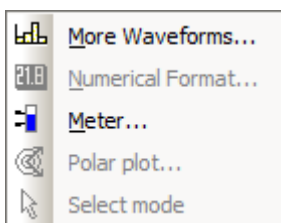
In addition to the numerical data display *Last value as number*, there is also a data display style having the form of a vertical meter bar. The height of the bar is determined from the last value in the waveform. This representation style is particularly useful for online viewing, as it offers a convenient overview of signal developments in multiple channels. The representation style incorporates graphical features such as different colors, slave pointers, margin checking etc.



Settings

Select menu item *Configuration/ Display* in the curve window. Select the *Bar Meter* radio button.

It is possible to display several channels together using a common y-axis. Select the channels to be displayed in the curve window.



To configure the bar meter itself, right-click on the curve window to open the context menu:

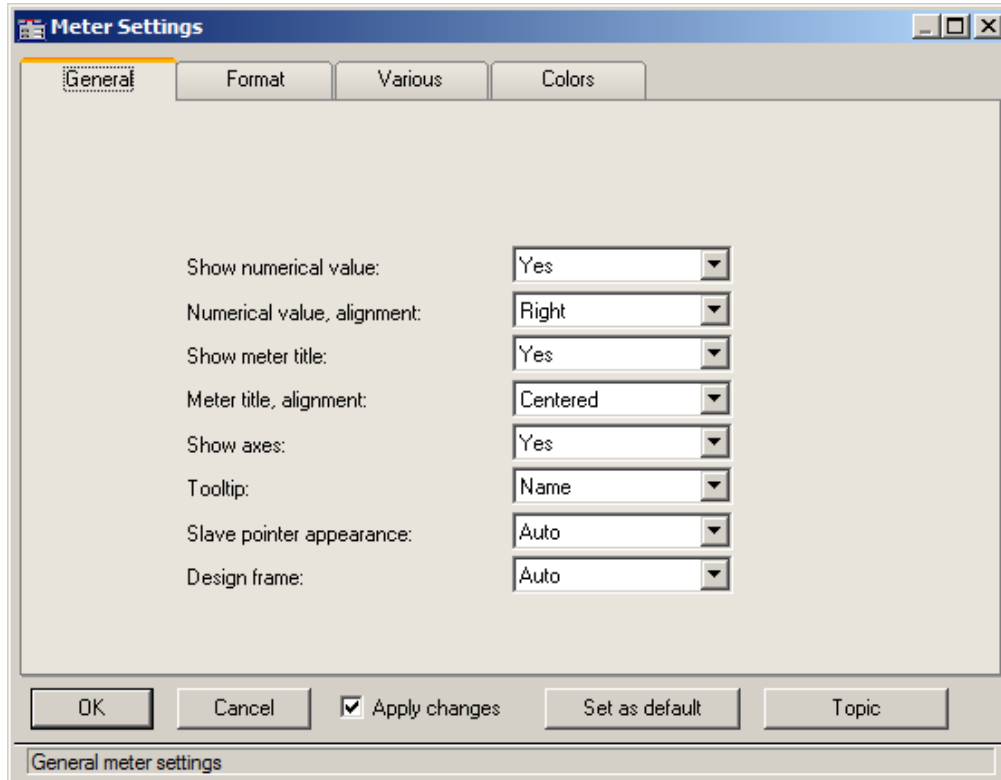
Select *Meter...* to open the *Meter Settings* dialog. This dialog consists of several sub-dialogs, each of which contains controls for different settings.

If you have preferences for certain settings, use the *Set as default* button. These settings will be implemented automatically each time bar meter display mode is used.

The following is a description of each sub-dialog in the *Bar Meter Settings* dialog.

Settings: General

This is where settings are made which affect all bars in the curve window alike.



Settings	Description
Show numerical value	This setting governs whether to display the numerical value of a signal together with its bar meter representation. The numerical value is written above the bar.
Numerical value, alignment	This controls the horizontal alignment of the numerical value (in case it is specified to appear). The recommended setting is <i>right</i> , but <i>left</i> and <i>centered</i> are offered.
Show meter title	This determines whether to display the corresponding channel name under the bar meter of a signal. This is particularly useful when multiple channels' signals are being shown.
Meter title, alignment	The horizontal alignment of the title text, which is always written in one line. Choices offered are <i>left</i> , <i>right</i> and <i>center</i> .
Show axes	A y-axis can be specified to appear adjacent to the bar meter at left. The scaling of the axis can be set by the usual method for curve windows. This axis determines the value range of the bar (lower-limit physical value at the bottom edge, upper-limit physical value at the top edge). This is also where to specify whether the scale is indicated or not. Even if the scale is not depicted, it is still applied to the bar's value range. The settings in the curve window's menu item Configuration/ Axes must be made afterwards.
Tooltip	For <i>Name</i> , the channel name is displayed; for <i>no</i> , it is not.
Slave pointer appearance	<i>Line</i> , or a clamp for <i>auto</i> .

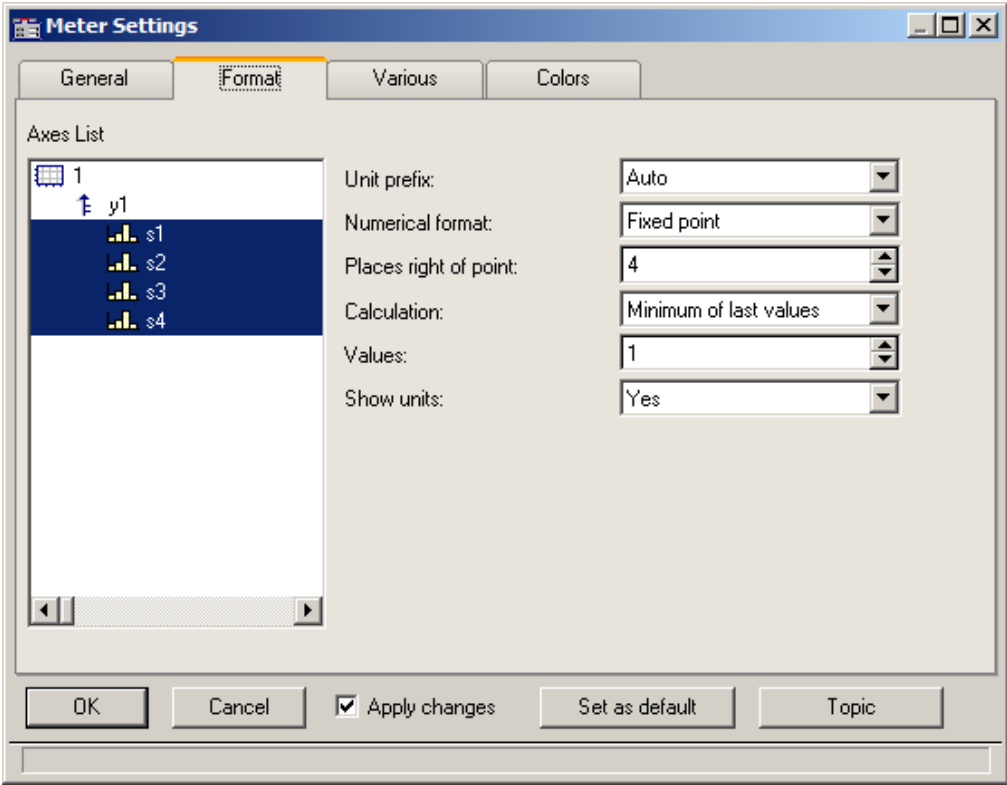
Settings	Description
Design frame	With <i>auto</i> , each channel has its own frame; with <i>Common frame</i> , not.



Settings: Format

This filecard mainly concerns the settings for the numerical format and for computing the numerical value. The numerical value is used for the readout above the bar and for determining the height of the bar.

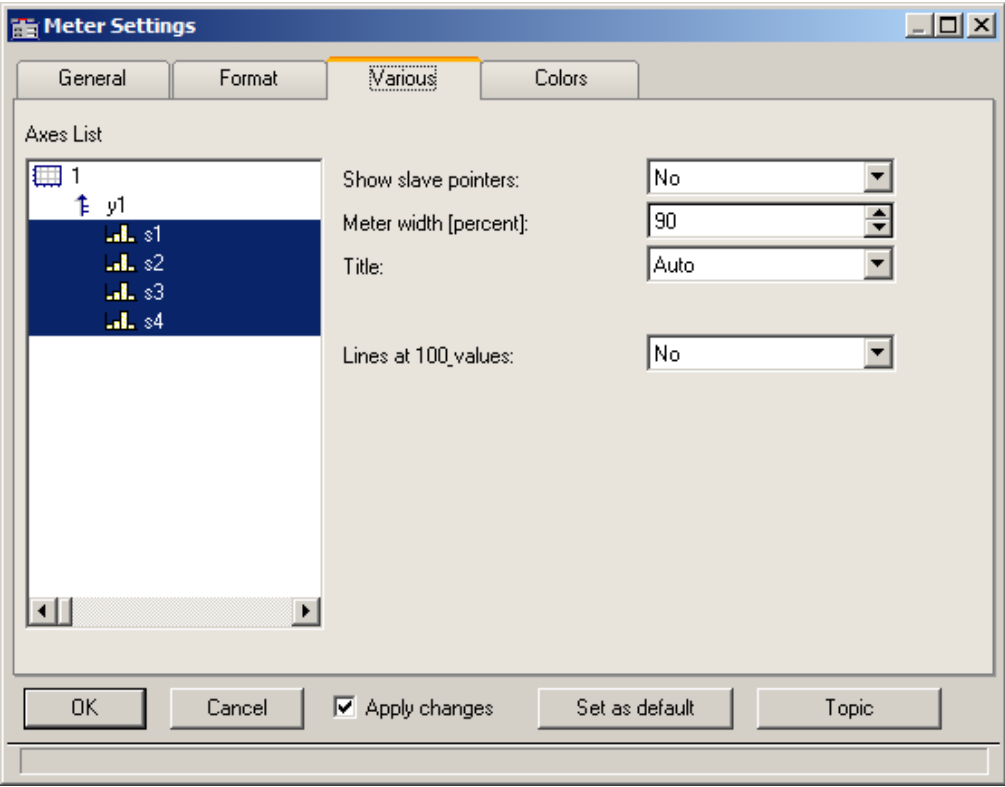
On this and the following sub-dialogs, the first step is to select all the channels to which the settings made should apply in common. Multi-selection can be performed as described in the instructions for the [axes list](#)⁷²⁷. If multiple, selected channels have mutually different settings for a property, the setting indicated for that property will be "???".



Settings	Description
Unit prefix	You can specify a unit prefix, reflecting the 10-based exponent of the signal value, to accompany the displayed numerical value. Prefixes from piko (p) to Giga (G) are supplied. When <i>auto</i> is set, the exponent is oriented to the exponent of the y-axis scaling, which is the recommended setting. For fixed point notation, a fixed exponent is recommendable, as well as for the y-axis scaling - see the description of the y-axis. Example: the waveform involved is a power measurement taking units of Watts and extending in range from 1e4 to 1e6 W. As the fixed unit prefix we choose "kilo" (k), resulting in values of 10kW to 1000kW. This setting is unnecessary if the numerical value of the signal isn't set to be displayed.
Numerical format	Fixed point (e.g., 0.01, 100, 365.25) or floating point notation (e.g., 3.5E-3) are available. If the value range of the signal is known, fixed point notation is better. This setting is unnecessary if the numerical value of the signal isn't set to be displayed.
Places right of point	Zero to fifteen places after the decimal point can be specified. This setting is unnecessary if the numerical value of the signal isn't set to be displayed.
Calculation	The displayed numerical value can be the most recently measured value, or can be a function of the N most recent values. The functions of N values offered are mean, minimum and maximum. If the waveform has fewer than N values, all available values are used in the calculation. If N = 1, the particular function used is irrelevant. The height of the meter bars, as well, is derived from the setting made here.
Values	Enter the number N of values, from which the mean, minimum, or maximum is to be calculated. Only integers ≥ 1 are permitted. A large value for N needs much time for computation, but 100 values is an amount the system can comfortably and quickly handle.
Show units	This determines whether the physical unit, together with the exponential prefix (p, n, μ, m, ...) is to be displayed with the numerical measurement value. It is sometimes necessary to include an indication of the physical unit, in order to identify the value's order of magnitude. The unit should be omitted only if it is already accurately reflected by the y-axis scaling.

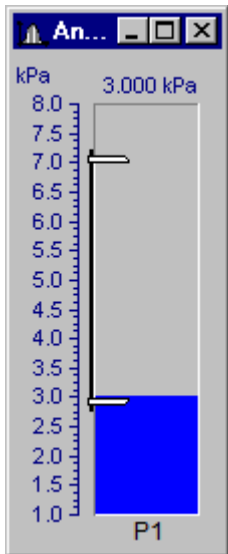
Settings: Various

This sub-dialog contains an assortment of setting controls, such as the width of the meter bars, or settings concerning the slave pointer.



Settings	Description
Show slave pointers	Specify here whether a meter bar should be equipped with slave pointers. These are markings appearing along a bar, which at any given time are situated where the signal's minimum and maximum encountered values were.
Meter width (percent)	Specify here the width of the meter bar relative to the width of the bar's "cell", which is the approximate width of the curve window divided by the number of bars in the window. Any fraction of the cell width, from 0% to 100% can be entered; a setting of 100% causes the bars to touch each other and thus 90% is recommended for greatest clarity.
Title	The title of the bar meter can be the channel to which its signal belongs (setting: <i>auto</i>), or any arbitrary name (<i>Fixed title</i>). If you select <i>Fixed title</i> from the list box, the entry box of the same name will appear below. Fixed title: Enter here an arbitrary designation for a bar meter. Up to 20 characters are allowed.

Slave Pointers



Slave pointers display the largest and smallest levels of a channel. At the beginning of a measurement, they lie together. They generally move away from each other during a measurement - the top one follows the largest values while the bottom one follows the smallest values. The slave pointers may also lie outside of the currently visible region.

The bar's slave pointers are represented by a graphic resembling a bar clamp. The slave pointers may also be located outside of the visible range.

Whenever a curve window is first opened or a new curve configuration is loaded, the slave pointers in that window are reset. The slave pointers are also not displayed when no data is present.

Slave Pointer: Computation

The slave pointers are rapidly computed by simply tracking the minimum and maximum values in a waveform. Therefore, slave pointers can be updated quickly even with large amounts of data. In typical online measurement applications, where the incoming data is simply appended to the data record, the use of slave pointers is appropriate. In off-line data analysis, however, they are not particularly useful because the entire data record is loaded at once.

Starting a new measurement on a channel with slave pointers does not reset them. They retain their former value until new data forces them to a new position.

Slave pointers are computed from the actual measurement data and not from a value computed from the channel for the bar meter. For example, consider a bar meter which is configured to display the average of the last 10 values. Although the meter is unlikely to show transient maximum values, these will affect the position of the slave pointers.

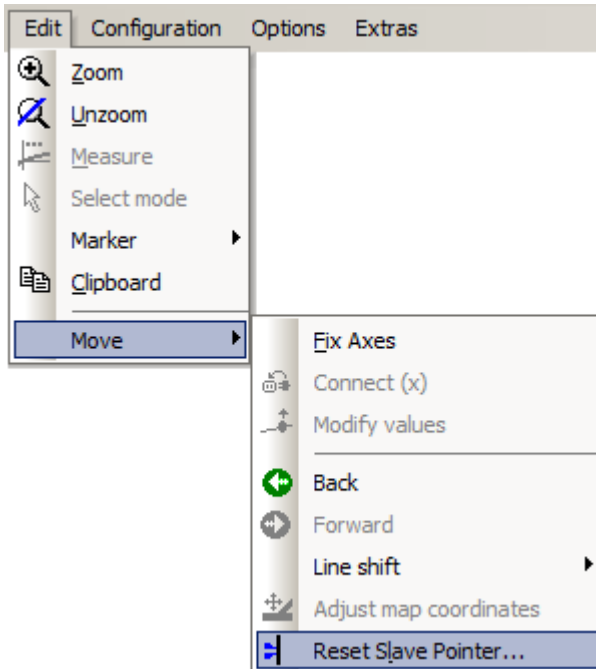
To avoid this effect from occurring, compute the average *online* and then display the resulting *virtual channel* in a bar meter. The slave pointers will then agree with the displayed maximum and minimum values.

Slave pointers are accurate to 6 significant figures, i.e. they use 4 Byte representation.

Slave Pointer: Reset

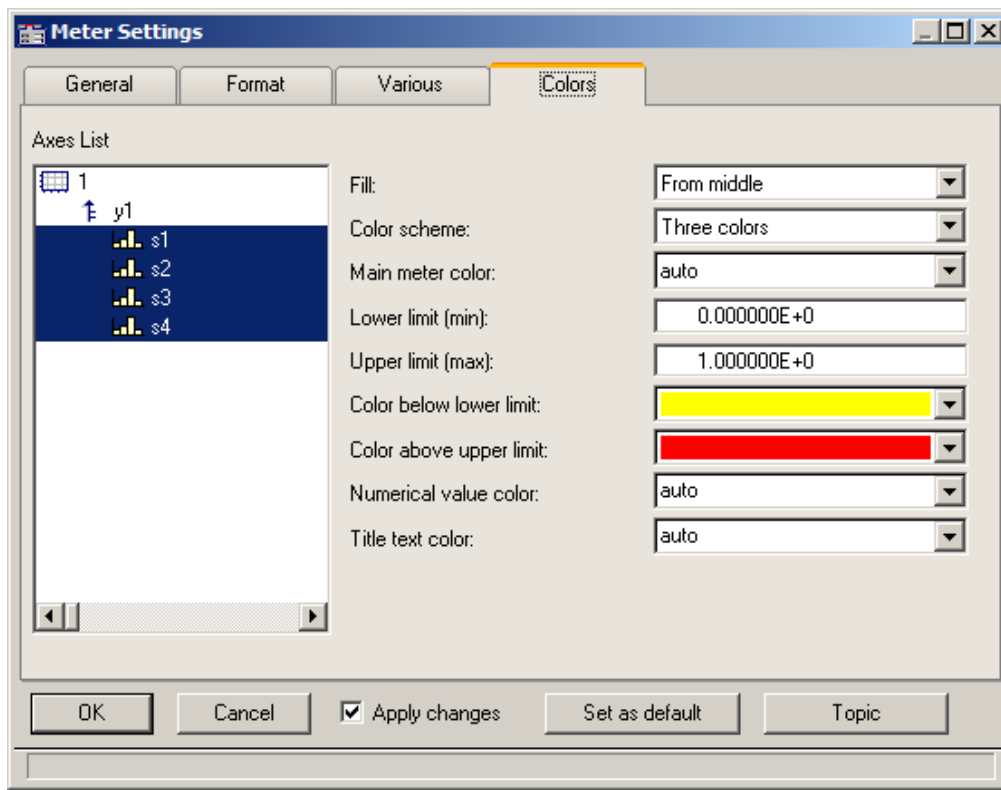
Resetting a slave pointer causes *all* slave pointers in that curve window to be reset. When you reset a slave pointer, it slides back to the current measurement value. Slave pointers have to be reset manually in each window. When a new measurement is started, this does *not* automatically reset the slave pointers!

To reset the slave pointers in a curve window, select *Edit / Move / Reset Slave Pointer...*



Settings: Colors

This sub-dialog lets you decide on colors for the bars.

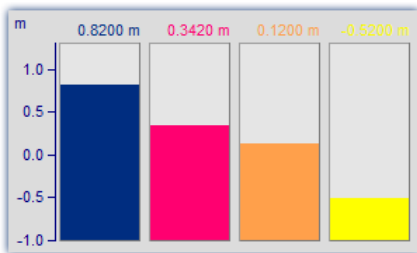


Fill

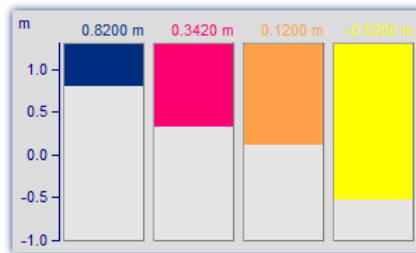
This setting determines from where the meter bars grow with increasing signal measurement values, as represented visually by a column filling with color.

The available options are from the top, from the bottom, from the center, and from $y = 0$ (the height along the bar, where the y-axis is equal 0.0).

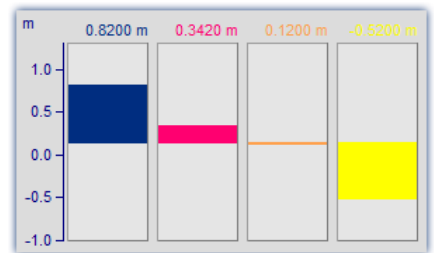
From below



From above



From $y = 0$

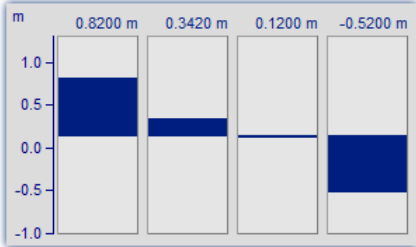


Alternating up and down from the 0 line of the axis

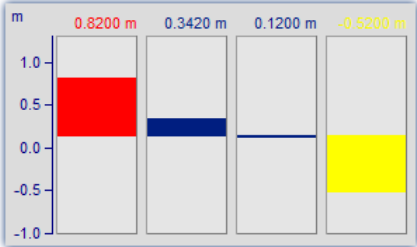
Color scheme

The bar can have only one color, three colors in succession or up to three colors at once. Three colored bars are used to indicate whether the measurement signal is currently within a tolerance range, which can be specified by the user in terms of an upper and a lower threshold value.

One color



Three colors



Three colors



A one-colored bar has the same color at all times.

The three color bar (*one color at a time*) shows the whole filled area in one color, which depends on whether the signal is below range, within range, or above range.

- The 1st color, if: Value <= Lower limit
- The 2nd color, if: Lower limit < Value < Upper limit
- The 3rd color, if: Value <=Upper limit

Where a bar has *Three colors together*, each of its colors is shown only within a certain range of the bar's height, as outlined below:

- The 1st color, in the range: fill height <= Lower limit
- The 2nd color, in the range: Lower limit < fill height < Upper limit
- The 3rd color, in the range: Upper limit <= fill height

Depending on the circumstances, therefore, up to all three of the colors can be visible in a bar.

Settings	Description
Main meter color	This is the only color of the meter bar if only one color style is specified; for three color style it is the middle color (the color of the bar indicating a signal within the limits set by the user). When the valid setting is <i>auto</i>, the automatic selection which would have been made for a waveform in the curve window is applied to the bar. The background color of the bar is the background color of the coordinate system, which is set via dialog Options/ Colors ⁸⁸² of the curve window. If a waveform doesn't contain any values, the bar meter remains empty as well.
Lower limit (Minimum)	For three-color display modes, a lower limit must be specified in this entry box. If there shouldn't be any lower limit, it is possible to enter -1e30, which is generally lower than any signal which will ever be measured.
Upper limit (Maximum)	For three-color display modes, an upper limit must be specified in this entry box. If there shouldn't be any upper limit, it is possible to enter 1e30, which is generally higher than any signal which will ever be measured. The upper limit must be greater than the lower limit.
Color below lower limit	In three-colored bars, this is the first color.
Color above upper limit	In three-colored bars, this is the last color.
Numerical value color	The numerical value, if it is specified to be displayed above the bar meter, can be assigned a fixed color. Or, if set to <i>auto</i>, the color is the same as of the bar itself.

Settings	Description
Title text color	The title below the bar can be assigned a fixed color. Or, if set to <i>auto</i> , the color is the same as of the bar itself.

Note

- **Reducing flicker:** If flickering occurs in a bar meter during a measurement, then you should use averaging (Meter Settings dialog: *Format/ Calculation/ Average of last values*). Be sure to set the numerical format to floating point. Restrict the decimal places right of the comma to just a few and use right alignment.
- **Appearance:** In general, tall, slim meters appear better than short, wide ones. If necessary, the curve window should be resized.

Limitations

- With data containing events or segments, the structure is ignored. If results are being computed from the last N values, the data set is simply considered a long waveform. Thus, computations can be performed on more than one event. IMPORTANT: Because this behavior may be changed in upcoming versions, we do not recommend that you use it.
- Only one y-axis is possible. All bar meters in a curve window have to use the same y-scaling. If you wish to use different y-scaling, than use separate curve windows.
- Slave pointers are only appropriate for online measurements. If data is processed off-line (e.g. with imc FAMOS), the slave pointers will not always function properly, i.e. they will not display the maximum / minimum values. However, if values have only been appended to a data record, the slave pointers will function properly. At the beginning of a new measurement, the data record has to be emptied before appending new data. This allows the slave pointers to correctly interpret the new data.
- With XY-data as well as with data reduced using Transitional Recording, computations over the last N values always use a different time interval. This behavior may be changed in upcoming versions.
- The slave pointers can only properly interpret data which has been correctly appended to a data record. For example, data which has not been recorded via the measurement device and therefore has not passed through the Data Manager is not properly interpreted.

11.6.1.5 Color map display

This style of data representation is analogous to a geographical map with color-coding of surface relief.

Multiple waveforms are arranged in a stacked layout (in the y-direction), where each one's values are plotted from left to right (in the x-direction). The amplitude is plotted in the z-direction, in other words, extending vertically out from the screen. The result is a contoured surface extending out from the xy-plane, onto which the observed looks with a bird's-eye view. The height of the surface is indicated by color coding, where different ranges of height correspond to specific colors.

There is additionally the possibility of superimpose x-y waveforms over the color map, and to have the map bordered by the waveforms. For more on this topic, see the section [Lines / Extras](#)⁷⁶³.

Prerequisites

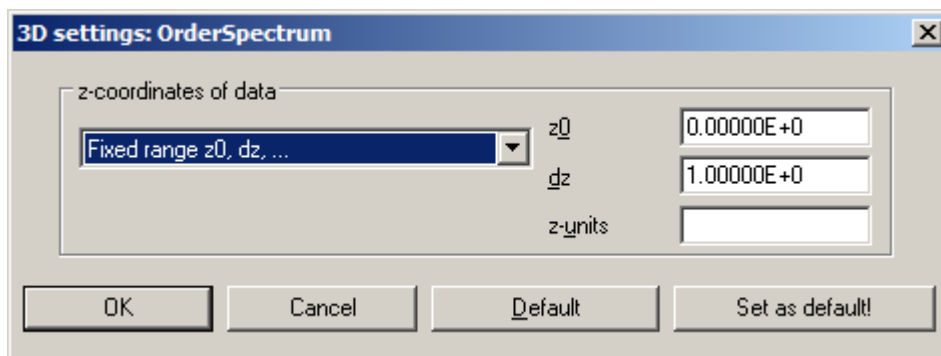
- Color video card with more than 256 colors, in other words 16 bit or 24 bit colors.
- Color printer of DIN A4 format.
- A fast PC (≥ 586 , 100MHz) with plenty of RAM (≥ 32 MB)
- Since the graphics display is based on large bitmaps (pixel graphics), large displays in particular require much storage space and computation
- With a DIN A4 color printer having 300dpi, one has about 2000 by 3000 pixels and with 24 bits per pixel one already arrives at about 18Mbytes!

Operation

First display all desired channels in the curve window. Then select the menu option *Color Map* in the dialog *Configuration / Display....*

Setting the Z-coordinate

In the dialog: [Configuration / 3D](#) ⁶⁹⁷ of the curve window, select an appropriate z-coordinate (see below):

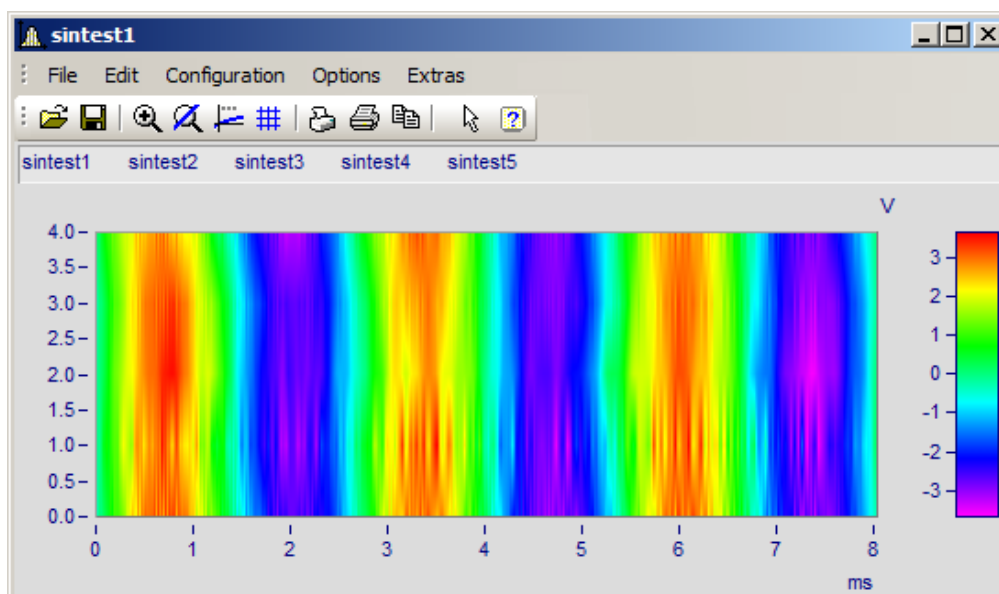


Reference

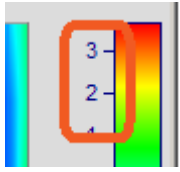
Waterfall diagrams z-coordinate

More on the topic of [Waterfall diagrams z-coordinate](#) ⁶⁵⁹

Result



Z-Axis Scaling



To adjust the color axis, double click on the scaling of the colored legend. The color axis is the same thing as the z-axis and is scaled as such.

Color Selection

The colors are set by means of the Color Map dialog: [Configuration/Color Map Display](#)  688.

Interpreting the Z-coordinate

The z-coordinate of the data (an otherwise undefined coordinate in imc FAMOS) is to be set via the dialog Configuration/ 3D in the curve window (see above). For example, The z-coordinate of the first waveform (channel) in a group could be set to z-min, the next to z-min + dz, the next to z-min + 2dz etc.

Orientation of the Coordinates

The component waveforms in the map are represented as multicolored horizontal lines, whose x-range is plotted from the x-minimum on the left edge of the graph to the x-maximum on the right edge. The height on the screen of a waveform's line or strip is determined by that waveform's specified z-coordinate; a waveform's z-coordinate is thus plotted along the y-axis, in a manner of speaking. Conversely, a waveform's y-coordinate, being its measurement value, is represented by a color; thus one can say a waveform's y-coordinate is plotted along the color axis, which for these purposes is also referred to as the z-axis. The scaling of the y-axis, then, serves to landmark regions of the pictorial representation, without pinpointing the location of any physical value in that representation.

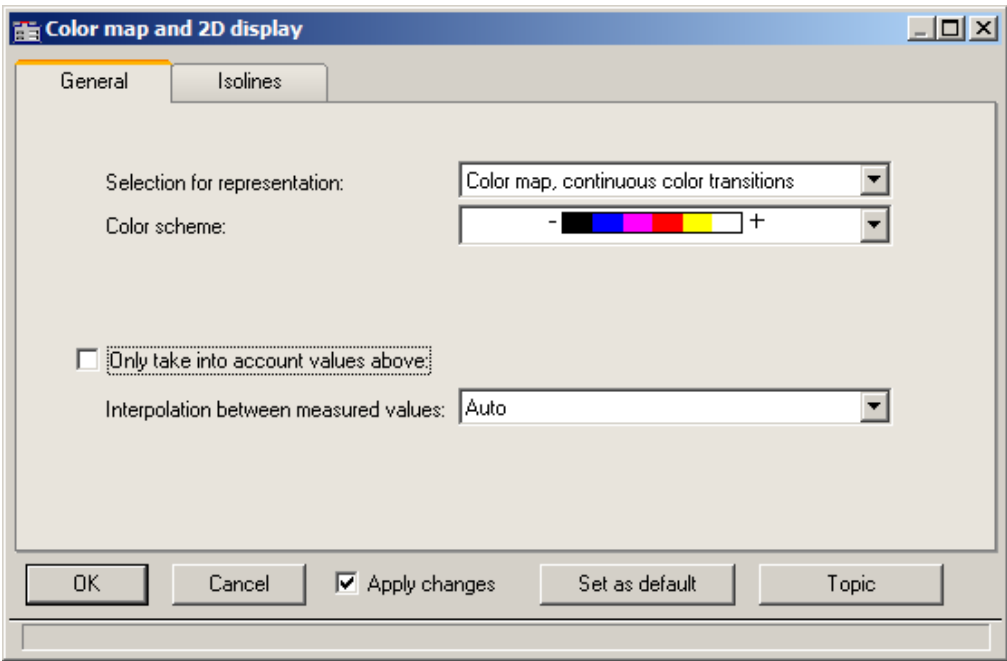
We thus achieve the illusion of a colorful landscape depicted on a map with green signifying valleys, the mountains brown and the snowy peaks white (in our case the colors follow the order of the physical spectrum).

Limits

- With locus and XY plots a measurement cursor is not available for the z-coordinate.
- The colored map representation is somewhat slow by comparison with other representation styles. Printing in particular requires large amounts of storage capacity. This may also require patience, therefore.
- The interpolation between two neighboring waveforms is fixed.
- The data's z-coordinate must increase strictly monotonically.

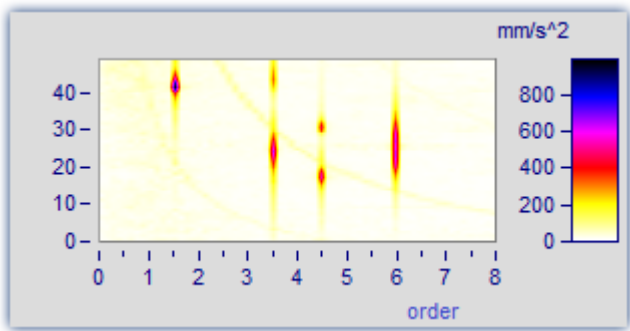
11.6.1.5.1 Color map dialog, "General"

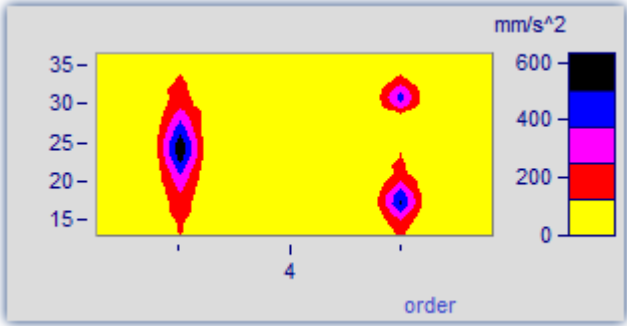
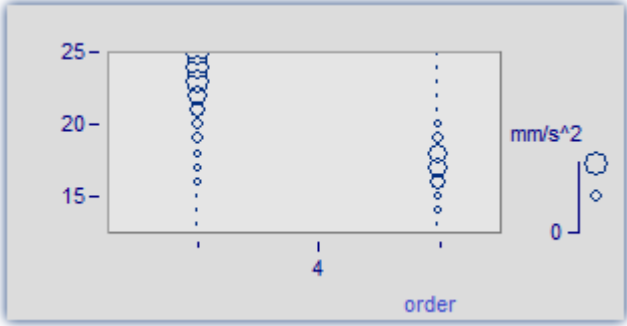
The options for color map display are set in the dialog called by means of the menu item *Configuration/Color Map*:



Selection for representation

Representation	Description
auto	Default colors from magenta to red, or from white to black for B/W display. No other options.
Color map, continuous color transitions	Interpolated color transitions. For each amplitude to represent, an appropriate color is calculated. The color scheme can be selected.



Representation	Description
Color map, graduated colors	Specified fixed number of colors without interpolation. The color scheme can be selected. Number of different colors: The amount of different colors must be between 2 and 1000. The suggested values in the list can be overwritten. 
Symbols, size corresponds to filling (Campbell)	Each measured value is represented by a symbol, e.g. a circle. The size of the symbol represents the respective amplitude. The minimum symbol size represents a value of zero, while the maximum represents the maximum value. The intermediate transitions are linear. The symbol is centered around the measured value.   The <i>Color of the symbol</i> corresponds to that of the first curve and can not be changed here.  Under <i>Symbol selections</i>, ellipses, rectangles and diamonds, either drawn as outlines or filled, are available.   
Remarks on display as symbols	• With symbol display, the appearance of the symbol's size in the legend is as true to reality as is possible, however, in consideration of how much space in the curve window is taken up by the legend, there are limits to the size. • With symbol display, the readings must increase in the x-direction strictly monotonously. Otherwise, no display is possible. • If XY-data having varying dx increments, or data with varying distance in the z-direction are graphed, then there is no determined size for symbols with (actually constant) size but variable filling. In the legend, an automatically determined value is used for the size.

Representation	Description
Symbols, filling corresponds to amplitude	Each reading is represented by a symbol, e.g. a box. The degree of the symbol's filling depends on the signal's amplitude, where an empty symbol (box) corresponds to the minimum and the maximum is represented by the largest size symbol available. The scaling between these extremes is linear. The <i>Fill color</i> and <i>Frame color</i> can be adjusted. For the frame, the setting <i>auto</i> is available, which represents the color of the first curve. For <i>transparent</i>, the frame is omitted.   The filling increases, depending on the style chosen, either from the outside to the inside, or vice-versa, or from one side or corner to the other.  The measurement value is located at the symbol's lower left corner.  For the display with symbols of varying size, the maximum symbol size can be set. The size then depends on the font size (of text appearing as labels in the curve window): The value is specified in percent. The size denotes the maximum diameter of the symbols. A size of 50% .. 200% is usually best.

Color scheme

The options depend on what display type is selected.

The color scheme can be set for color map display with graduated or continuous color transition.

All color schemes (except *auto*) are independent of the capabilities of the output device.

Color scheme	Description
auto	For color output devices, the colors Magenta to Red are used; for b&w printers, the shades from White to Black.

Color scheme	Description

Transition between 2, 3 or 4 colors

Here you have a choice of using 2, 3 or 4 fixed colors. Then additional controls appear to let you select the particular colors.

Color scheme:

Transition between 2 colors

Color transition from (small ampl.):

Color transition to (large ampl.):

Dialog with 2 colors.

Color scheme:

Transition between 4 colors

Color transition from (small ampl.):

Color transition via:

Color transition via:

Color transition to (large ampl.):

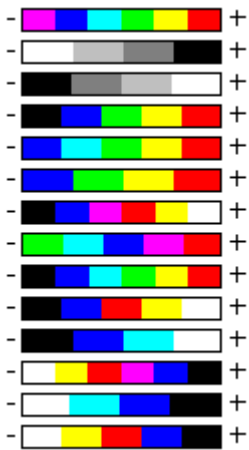
Dialog with 4 colors.

Between the fixed colors, any intermediate colors are generated by linear interpolation of the fixed colors. For instance, if color map display with 5 graduated colors is chosen, and a transition between two fixed colors is also set, the two colors at the ends of the scale are fixed and the other 3 are determined by interpolation.

In the dialog, the color for the smallest amplitude is indicated first (above), and the color for the highest amplitude in the lowest position.

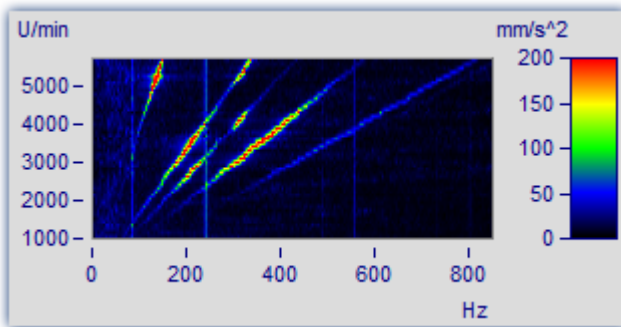
Fixed color schemes

These color schemes are available for direct selection. At left (where "-" appears), the color for the smallest amplitude is shown, at right ("+") the color for the highest.



Only take into account values above

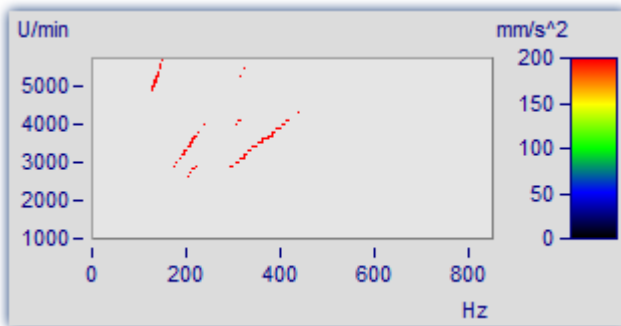
The options depend on what display type is selected.



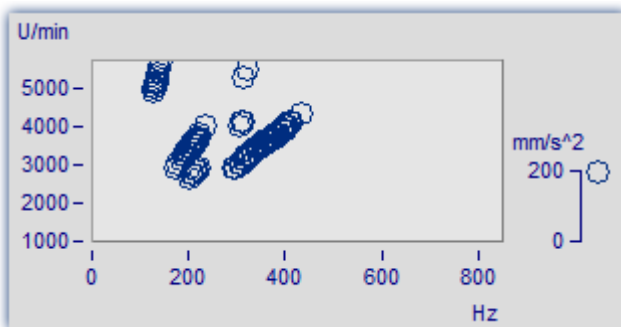
If this option box is not check-marked, all readings are taken into account (default).

If the checkbox is marked, then a lower limit for the amplitudes to be displayed can be specified.

☒ Only take into account values above:



Only relatively large reading values are given consideration. All others merge into the background.

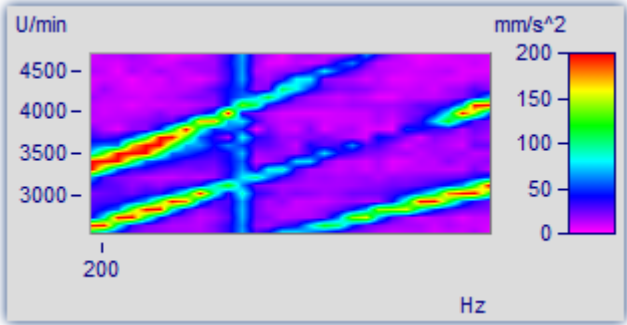
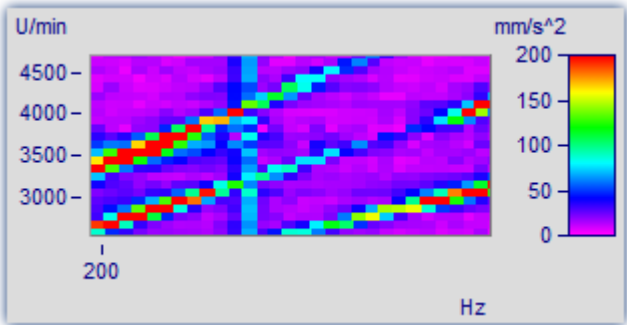
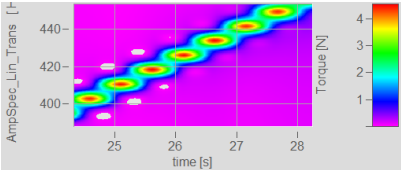


The same graph in the mode *Symbols*, size corresponding to amplitude (Campbell)

Interpolation between measured values

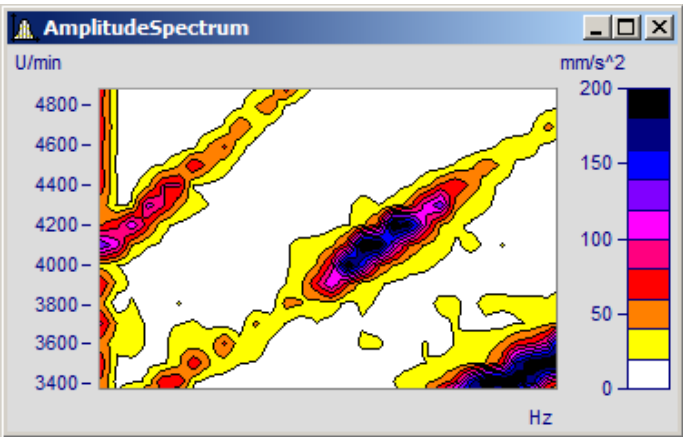
The options depend on what display type is selected.

This refers to interpolation in xy-plane between readings. Not to be confused with interpolation in the z-direction (interpolation of colors).

Interpolation	Description
auto, linear	In the x- and y-directions (horizontal and vertical on-screen, respectively) interpolated values between the specific readings are determined. This corresponds to linear interpolated lines between plotted points. 
Constant extension to upper right	In the x- or respectively, the y-direction, all measured values are extended at constant magnitude until the next measured value. This corresponds to stair-step or bar meter display of measured values. 
Interpolation, cubic polynomials	In x- and y-direction (horizontal and vertical on the screen respectively) interpolation is done between all measured values with cubic splines. 

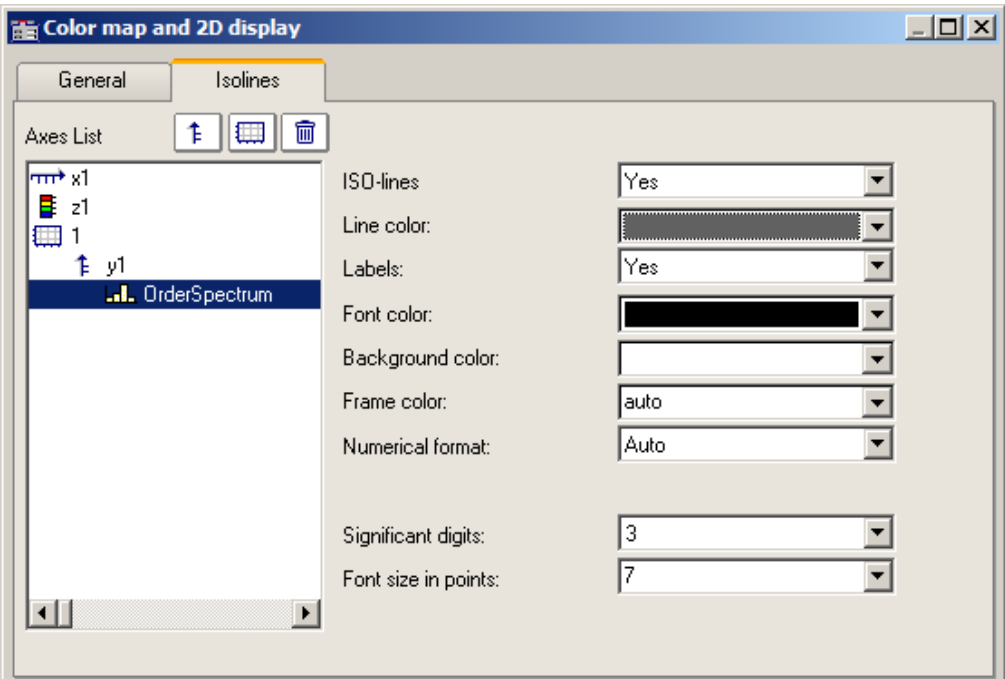
11.6.1.5.2 Color map dialog, ISO-lines

ISO-lines are contours, in other words, lines joining coordinates of equal height. They are used here in the same way as in maps. Since the lines appear in conjunction with graduated colors, they mark the boundaries between the colors.



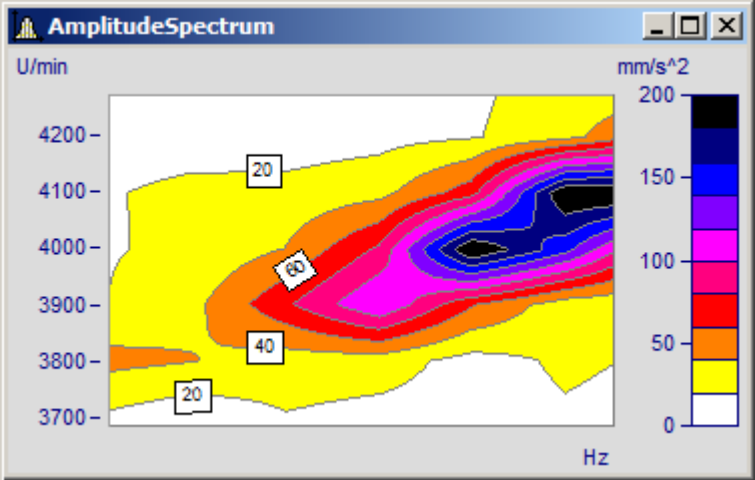
ISO-lines can only be set for [color map display with graduated colors](#) .

The "Isolines" index card of the Color map settings dialog



The following options can be selected:

Options	Description
ISO-Line	Display of the isolines. This option is only available for the display types <i>Color maps</i> , <i>graduated colors</i> .
Line color	The color of the isolines is set here.

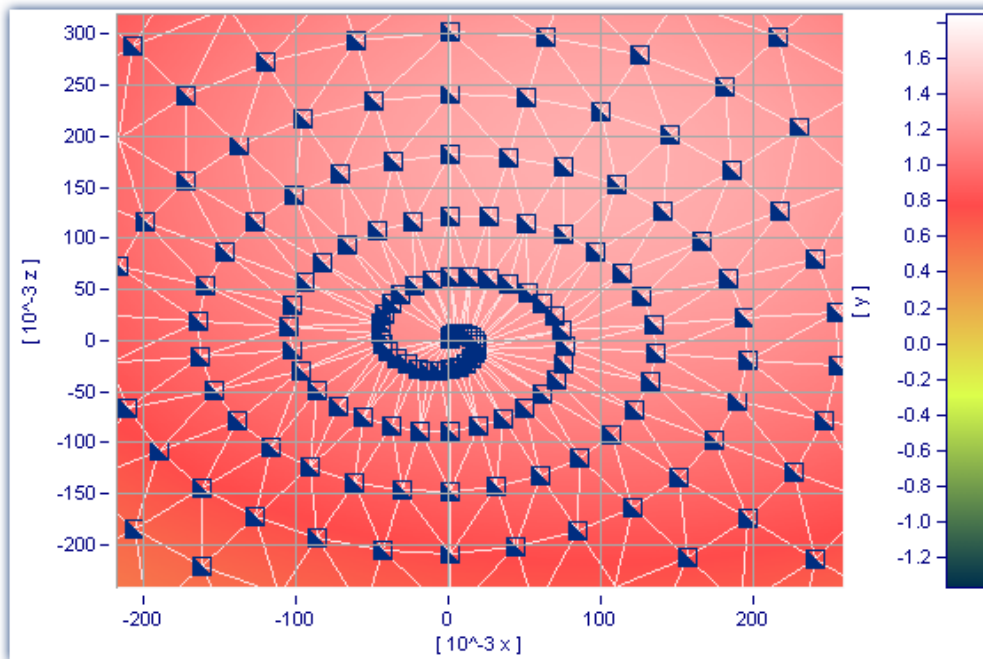
Options	Description
Labels	<i>auto, no</i>: The isolines are not labeled. <i>yes</i>: The isolines are labeled; small tags are attached to the lines, as space allows. 
All further options only pertain to the labeling of the isolines.	
Font color	Determines the color of the label text.
Background color	Determines the color of the label background.  <i>auto</i>: Transparent background, but covering the contour line.  <i>transparent</i>: The label background is totally transparent. The contour line between the colors is visible as well.  <i>Fixed color</i>: The label background is given a fixed color and is thus not transparent.
Numerical format	<i>Fixed point</i>: Fixed point with a fixed number of decimal places to the right of the point e.g. 17, 17.35, -0.0017. <i>Floating point</i>: Floating point with a fixed relative precision and exponent, e.g. -1.28E-7 or 3.4E+0. <i>auto</i>: Automatic selection, partially dependent on the numerical representation of the z-axis.
Unit prefix	This consists of the order of magnitude and, if appropriate, a prefix such as kilo, milli, Mega or even just 1 (default, use of unit without prefix), which appears after the number before the unit. If the reading is 10 and takes the unit V (Volts), then for a selection of "m" (milli), a value of 10000mV is returned. For <i>automatic numerical format</i>, the same order of magnitude is set as for the labeling of the z-axis.
Places right of point	Number of decimal places for <i>Numerical format Fixed point</i> or <i>Floating point</i> : 0 to 15.
Valid digits	For <i>numerical format auto</i> the number of decimal places cannot be set, instead, the number of significant digits. Thus, the numbers 3.4, 3.4E-4 and 0.034 each have 2 significant digits.
Font size in points	The font size in points (pt): 4pt to 10pt. Typical values are 6 and 8 points.



See chapter [Waterfall Diagrams](#) 659

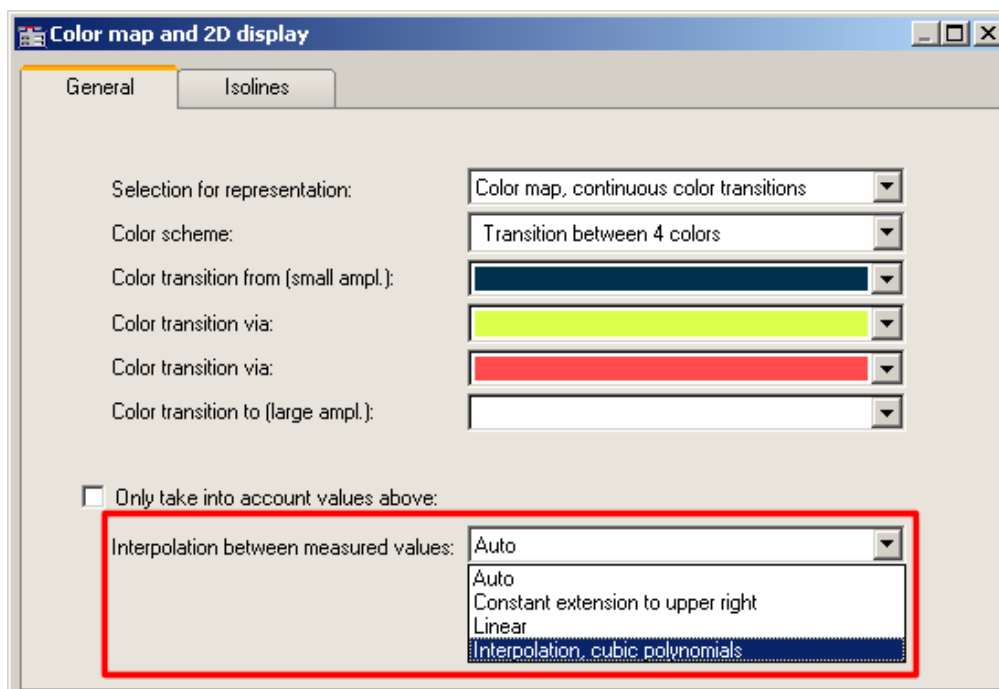
11.6.1.5.3 Waveforms with associated x-,y-,z-variables

As an alternative to 3D-display, waveforms with associated variables can also be displayed in a color map.



Color map display of a data set with multiple dimensions

It is also possible to specify interpolation between measured data points for this display style. To do this, open the color map's Properties dialog and on the *General* page select an interpolation type from the dropdown-list.



Interpolation between measured values in a color map

11.6.1.6 3D display

Another display style for datasets with [associations of x, y, and z variables](#)⁷³², for segmented waveforms is 3D display.

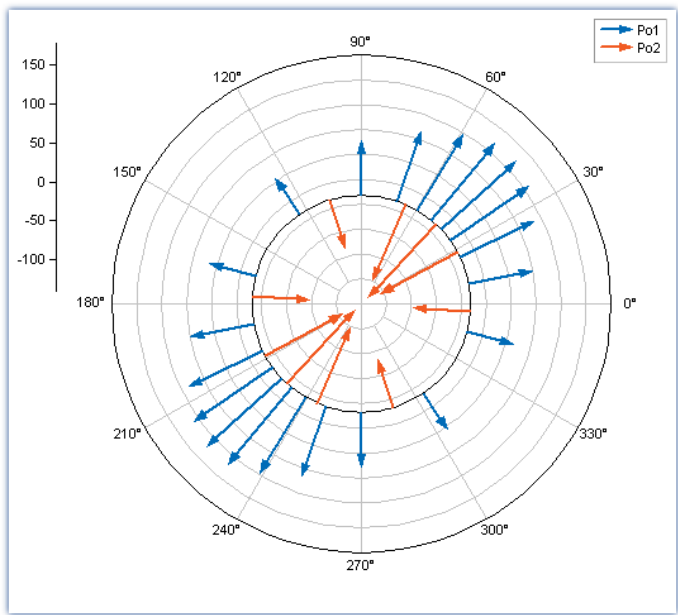
In this display style, the perspective can be freely rotated, and the [Axis Navigation Bar](#)⁹⁰⁵ can be used to change the view. It is possible to display more than one dataset with associations of x, y and z variables. This makes **comparison** of multiple superimposed 3D data sets possible.

Other curve window functions such as *zooming*, *setting markers* or *line-shifting* are not available here.

For more on using 3D-display, see the section on [Configuration / 3D](#)⁷⁹⁰.

11.6.1.7 Polar plot

Complex data sets can be displayed as a *polar plot*. To do this, select the *polar plot* display type in the *Configuration\Display* menu.



The dialog presenting the polar diagram's properties is opened via the context menu *Configuration\Polar plot...* :

Base

Polar plot

Position of axis:

Angle:

Degree [°]

Sky direction:

N,NE,E,SE ..

Direction of rotation:

Left rotating

Position of 0°:

Right

Interpretation of X-Component:

Auto

Polar plot settings

Option	Description
Position of axis	Axis scaling arrangement: "half" or "doubled". The doubled axis scaling appears as mirrored in the lower portion. If the origin is not located in the center, only the "half"-style of the axis scaling is displayed.
Angle	Display of the angle label in <i>degree [°]</i> . <i>On</i> or <i>Off</i> .
Sky direction	In the respective display language: German or English. • 4x: North, East, South, West • 8x: North, Northeast, East, Southeast, ... • 16x: North, North-northeast, Northeast, East-northeast, East, East-southeast,...

Option	Description
Direction of rotation	Default = leftward; alternatively rightward.
Position of 0°	right, top, left, bottom; or freely defined.
Interpretation X-Component	Degree (0-360°) or RAD (0-2π)

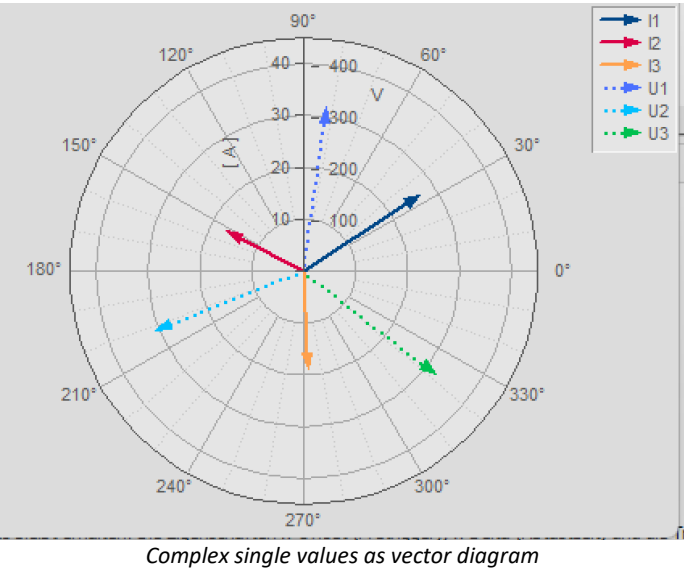
 Notes

Sample project PolarPlots1

The included sample project "PolarPlots1" displays the following examples in a Panel.

11.6.1.7.1 Example: Pointer diagram

Pointer diagram

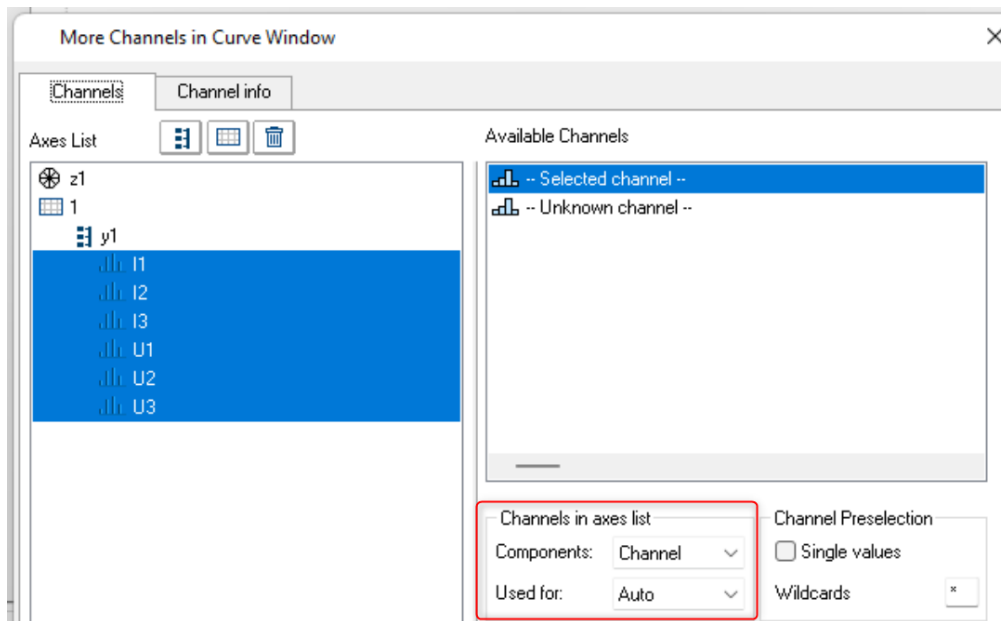


In imc FAMOS you can create a **complex** single value using the function `Compl()`.

The pointer graph above is created as follows:

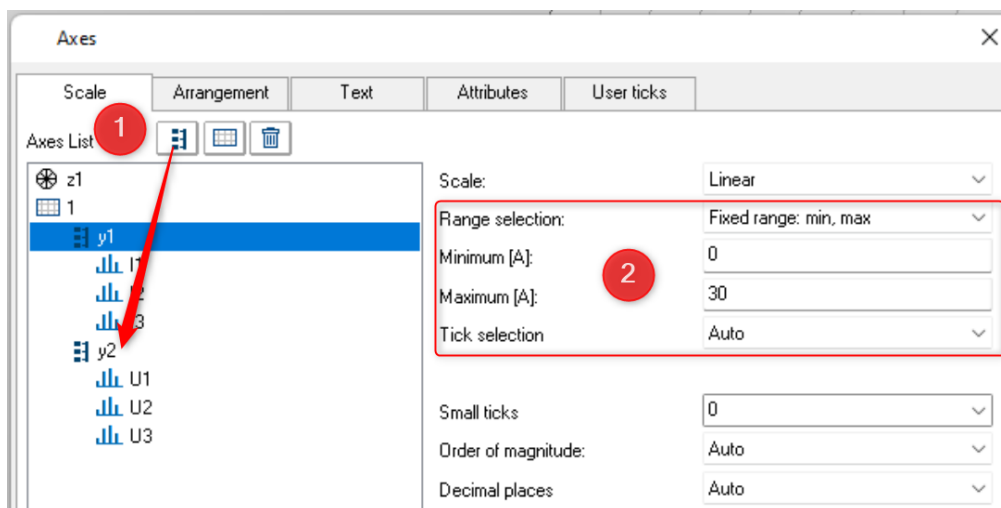
- Sequence for generating the complex variables:
`I1= compl (27'A', 33'Degr')`
`I2= compl (17'A', 153'Degr')`
`I3= compl (19'A', 273'Degr')`
`U1= compl (327'V', 82'Degr')`
`U2= compl (315'V', 202'Degr')`
`U3= compl (331'V', 322'Degr')`
- Display the single values in a curve window and select the display style *Polar plot* in the menu `Configuration\Display`.

3. By default, the curve window only shows the magnitude component of a variable. For this reason, all arrows point upward. With the right mouse button, open the context menu and there select *More Channels...*. Select the variables and select *Channels in axes list\Components: Channel*. The arrows will then be drawn according to the specified angle.



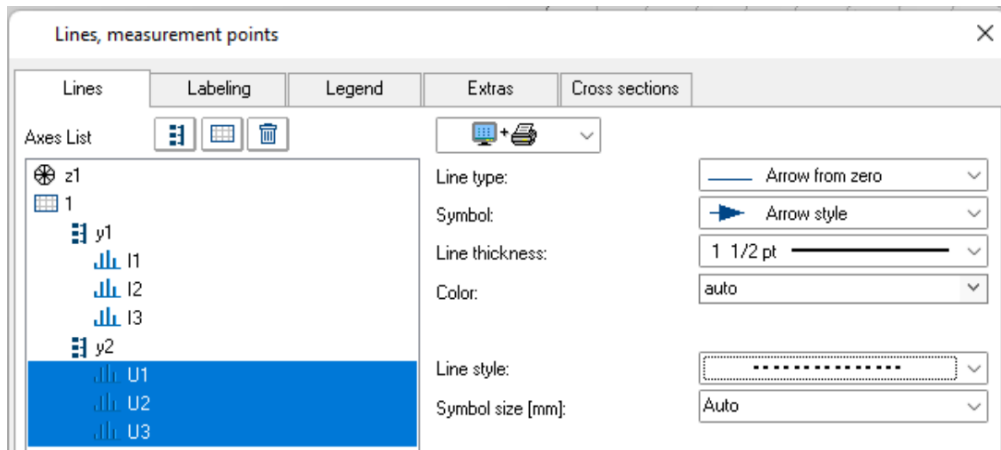
Observing each component of a complex waveform

4. As in a normal curve window, all variables are assigned to an axis. The values for amperage which are very low compared to the voltage values therefore appear barely visible. So now we want to display the amperage values and voltage values each with a separate respective axis: Menu item: *Configuration -> Axis...*. Using the items: *Range selection: Fixed range: min, max*, the scaling for the axes can be set.



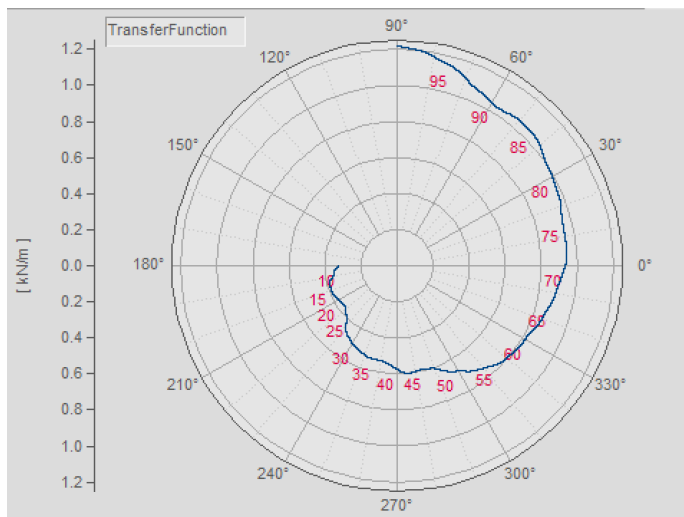
(1) An additional axis is added. (2) Optimize the scaling of the axes.

5. Using the Lines dialog, you can now adapt the display of the arrows.
Right mouse button: "Lines"



Dialog lines: Display with arrows

11.6.1.7.2 Example: Transfer function



Example transfer function

There are further possibilities for displaying a complex data set:

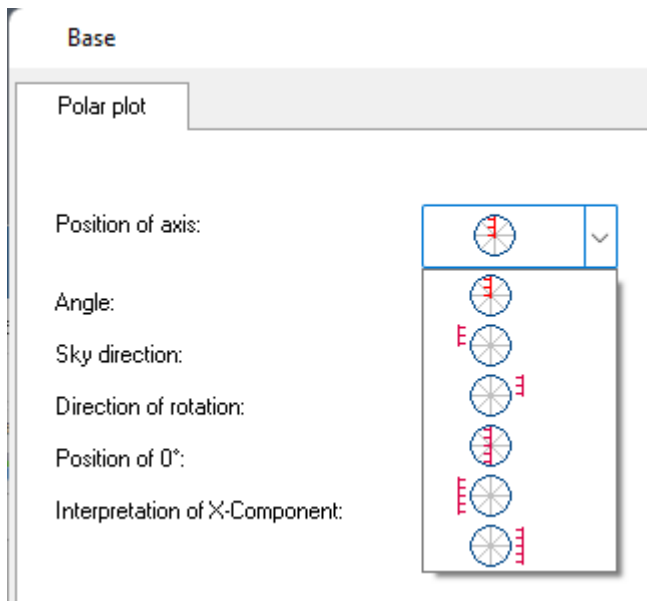
1. Use imc FAMOS to create a complex data set:

```
; transfer function diagram
t = ramp ( 0, 1, 11000 )
m = t * 0.1 + 200 + 1000*smo ( Random(leng?(t), 2, 0, 0, 3), 1000 )
p = t * 0.03 + 150
m = xoff ( xdel ( CutIndex ( m, 1001, 10000 ), 0.01 ), 10 )
p = xoff ( xdel ( CutIndex ( p, 1001, 10000 ), 0.01 ), 10 )
yUnit p Degr
yUnit m N/m
xUnit p Hz
xUnit m Hz

TransferFunction = compl ( m, p )
Labels = red ( TransferFunction, 500 )
```

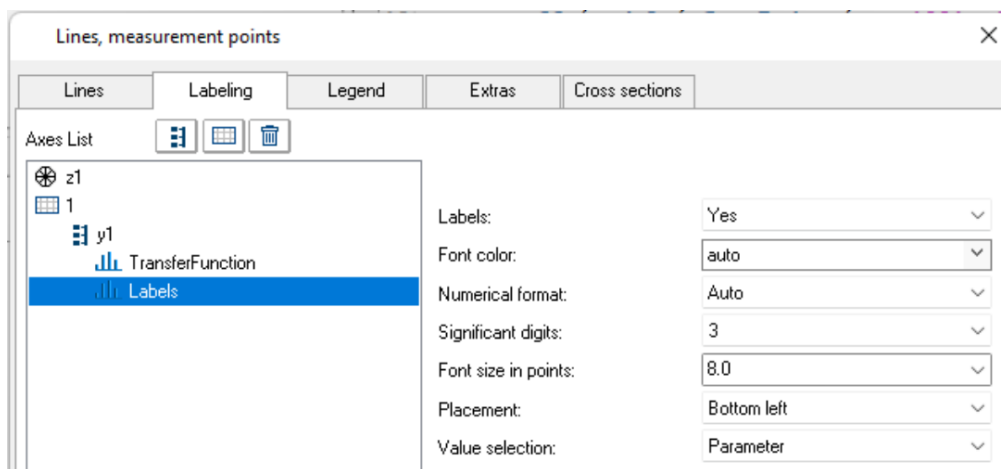
Important Note: The angle specification distinguishes between **degrees** and **RAD** as the unit. If no unit is specified, then RAD is applied, meaning 2PI instead of 360°. For this reason, it is important to use **yUnit** to set the unit as Degr, as shown in the example.

2. Display the data set *TransferFunction* in a curve window and select the display style **Polar plot** in the menu **Configuration\Display**.
3. Using the button "Topic" at the bottom right, go to the dialog *More Channels...*. In order for the settings to take effect on both components, the setting selected under *Channels in axes list* must be **Components: Channel**. The angle data is now reflected in the display.
4. **Axes position:** By default, the Y-axis is displayed only in the positive direction. You can change both the axis' position and its range: Right mouse click: *Polar plot*. The double-axes are only plotted when the value range is positive and starts at 0. The scaling of the axes may need to be adjusted accordingly.



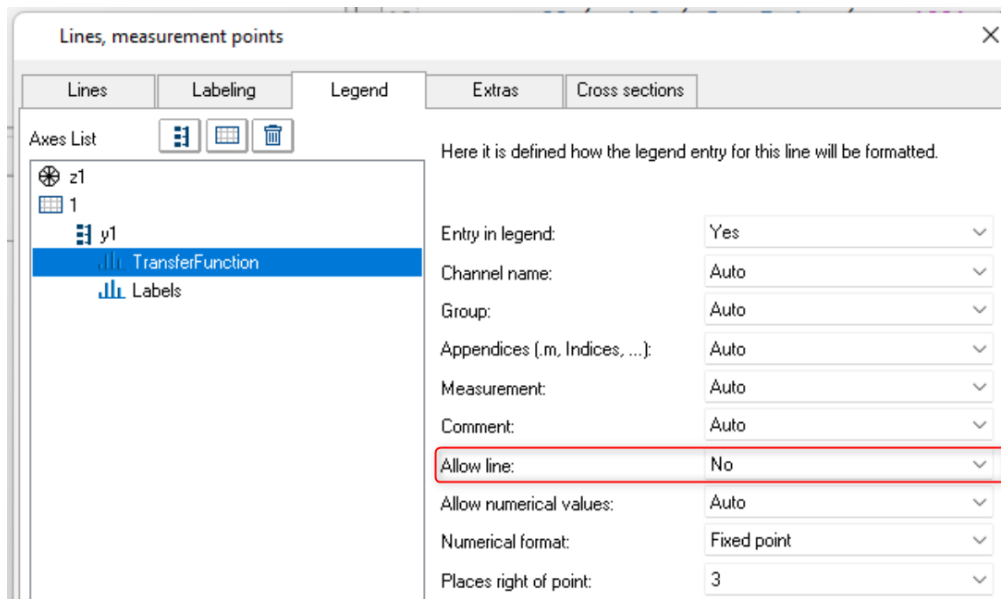
Position of axis

5. **Labeling the values:** The individual data points can be labeled along with their values. However, in this example the amount of data would make that impossible to read. For this reason, the data set *Labels* was created, which is derived from the data set *TransferFunction* by applying a reduction factor of 500.
 - *More Channels...* assign *Labels* to the same axis.
 - Click the button *Topic* and select *Lines: Line type: Only symbols, Symbol: empty*
 - Go to the page *Labeling* and activate it: By default, *Labeling* displays the magnitude. In our example we wish to display the frequency. For this reason, select for *Value selection* the entry *Parameter*.



Labeling the values

6. **Legend:** The data set *Labels* is not to appear in the legend. For this reason, in the *Lines* dialog on the page *Legend*, we select the data set *Labels* and select *Entry in legend: no*. For the data set *TransferFunction*, we select *Allow line: no*.

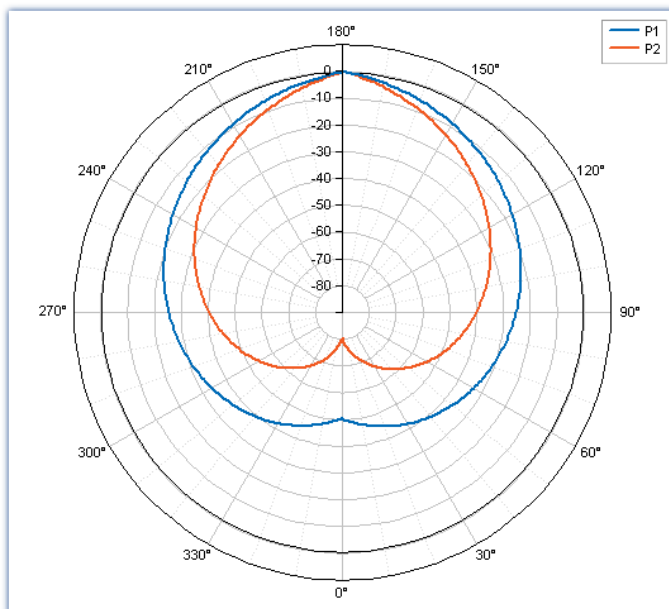


Legend display on/off for each channel separately

The position of the legend can be directly optimized in the curve window by using the mouse.

11.6.1.7.3 Example: Microphone polar pattern

Acoustics diagram with negative value



Display of XY data, with interpretation of the X component as an angle [°]

With certain kinds of applications, **negative values** are typical. Thus it is also possible to represent negative mechanical tension and traction forces as a polar plot. The following is a typical example from the field of acoustics, featuring negative dB-values.

1. Creating the complex data sets:

;

```

local Level1= !CreateSignal.Ramp(-50,0,1,180) * 1'dB'
local angle1= !CreateSignal.Ramp(0,179,1,180) * 1'degree'
local P_1 = XYof(angle1,Level1)

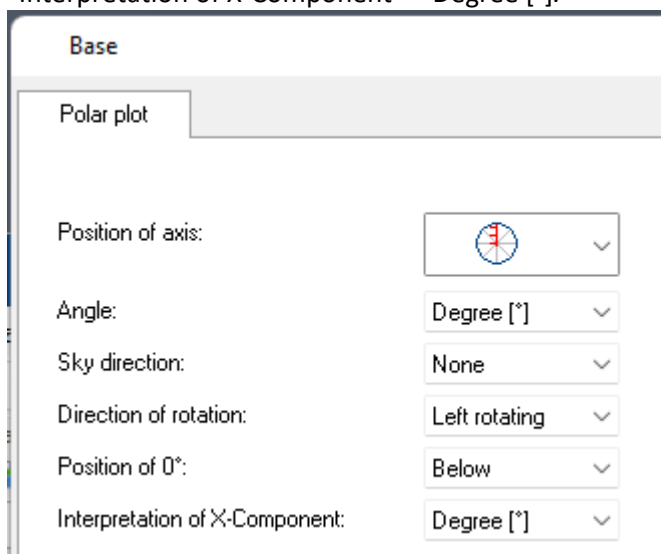
local level2= !CreateSignal.Ramp(0,-50,1,180) *1'dB'
local angle2= !CreateSignal.Ramp(180,359,1,180) *1'degree'
local P_2 = XYof(angle2,level2)
P1 = Append(P_1,P_2,0)

;-----
local level= !CreateSignal.Ramp(-80,0,1,180) * 1'dB'
local angle= !CreateSignal.Ramp(0,179,1,180) * 1'degree'
P_1 = XYof(angle,level)

level= !CreateSignal.Ramp(0,-80,1,180)
angle= !CreateSignal.Ramp(180,359,1,180)
P_2 = XYof(angle,level)
P2 = Append(P_1,P_2,0)

```

- The two data sets P1 and P2 are displayed in a curve window and represented by a *Polar plot*: Menu item: *Configuration\Display -> Polar plot*.
- Using the button "Topic" at the lower right, go to the dialog *Polar plot*. "Position of 0°" = "Below" and "Interpretation of X-Component" = Degree [°].

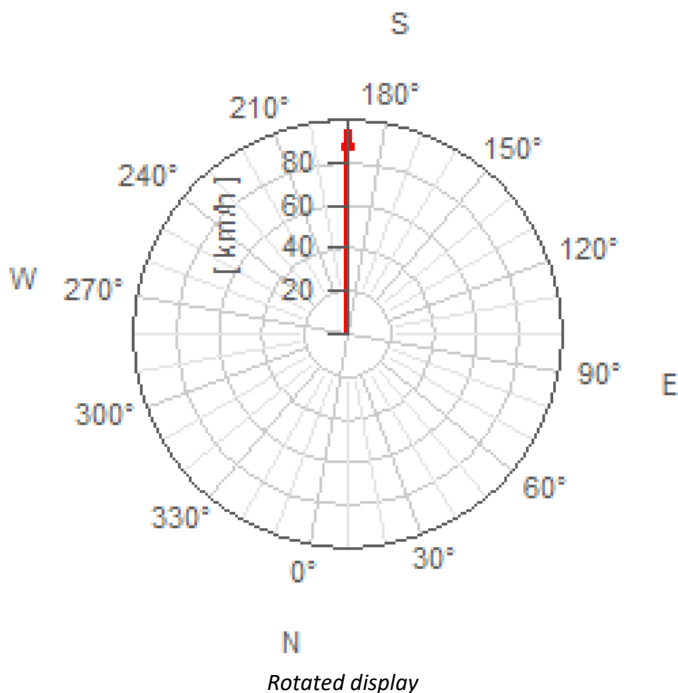


Polar plot settings

- Axes:** Using the button "Topic" at the lower right, go to the dialog *Axes*. Select the axis y1 and for *Range selection*, select *Fixed range: min, max*, and enter for *Minimum*: -90 and for *Maximum*: 10. For *Small ticks*: set "Auto".
- Lines:** Using the button "Topic" at the lower right, go to the dialog *Axes*. Select the two data sets P1 and P2 and adjust the line thickness according to your wishes.

11.6.1.7.4 Example: Orientation

Example: Display of compass directions and changed orientation



1. Creating a complex single value:

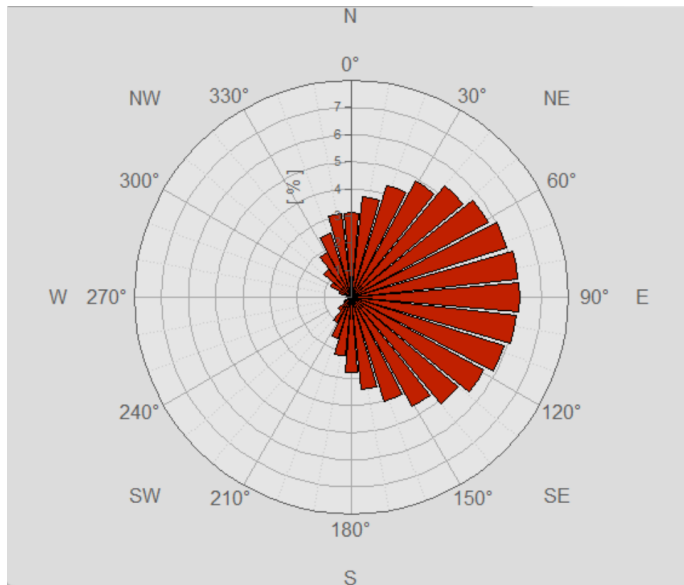
```

; -----
W1= comp1 (97 'N', 190 'degree')

```
2. Display the variable W1 in a curve window and represent it by a *Polar plot*: Menu item: *Configuration\Display -> Polar plot*.
3. Using the button "Topic" at the lower right, go to the dialog *More Waveforms*. In order for the settings to take effect on both components, the setting selected under *Channels in axes list* must be **Components: Channel**. The angle information is now applied in the display.
4. **Line**: Using the button "Topic" at the lower right, go to the dialog *Lines....* Adjust the settings for *Line thickness* and *Color*.
5. **Orientation**: Using the button "Topic" at the lower right, go to the dialog *Polar plot*. Set "*Position of 0°*" to 260°.

11.6.1.7.5 Example: Rosette (wind force)

Example: Wind force represented by a rosette display



Example: Distribution of wind direction in percentages

The polar representation with the **line type: Rosette** is used with a normal data set (ND). The angles are determined by the xDelta of the data set.

If only the amplitudes are available, the xDelta can be set using the following formula for a 360° distribution:

Data= **XDel**(Data, **PI2/leng?**(Data))

1. Create a data set of wind values, subdivided into 32 partitions:

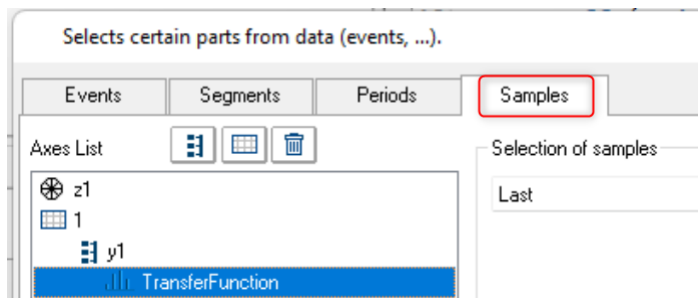
```
; -----
; with 2PI the xDelta is calculated for the full circle.
local temp= !CreateSignal.Ramp(0, 2*PI, 2*PI/32, 32)
temp= (sin(temp)+1)/2 ; Amplitudes
Wind32= temp/sum(temp)*100'%' ; normalized to 100%
```

2. Display the variable *Wind32* in a curve window and represent it as a *Polar plot*: Menu item *Configuration\Display -> Polar plot*.
3. Using the button "Topic" at the lower right, go to the dialog *Lines....* *Line type= Rosette*, make a selection for *Color*.
4. **Labeling**: Using the button "Topic" at the lower right, go to the dialog *Axes....* On the page *Attributes*, select axis *z1*, set the *Font size* to 12 and modify the *Font color* if desired.
5. **Orientation**: Using the button "Topic" at the lower right, go to the dialog *Polar plot....* Set "*Position of 0°*" to *Top*, *Sky direction= N, NE, E, SE...*, *Direction of rotation= right rotation*.

11.6.1.7.6 Pointer diagram during measurement

For measurements with imc CANSAS, the measurement device **does not** generate complex data sets. For instance, with an online FFT, the magnitude and phase are each transferred as separate channels. These can be combined to form complex data sets in the curve window under [More Channels...](#) ⁷³².

If one wishes to display the complex data as a [pointer diagram](#) ⁶⁹⁸ like above, then besides the procedure outlined above, the following step is also necessary: Since only the channel's current value, meaning the most recent one, is to be displayed, select *Configuration \ Events, segments periods...Go to the page Samples* and select *Last* under *Selection of samples*.



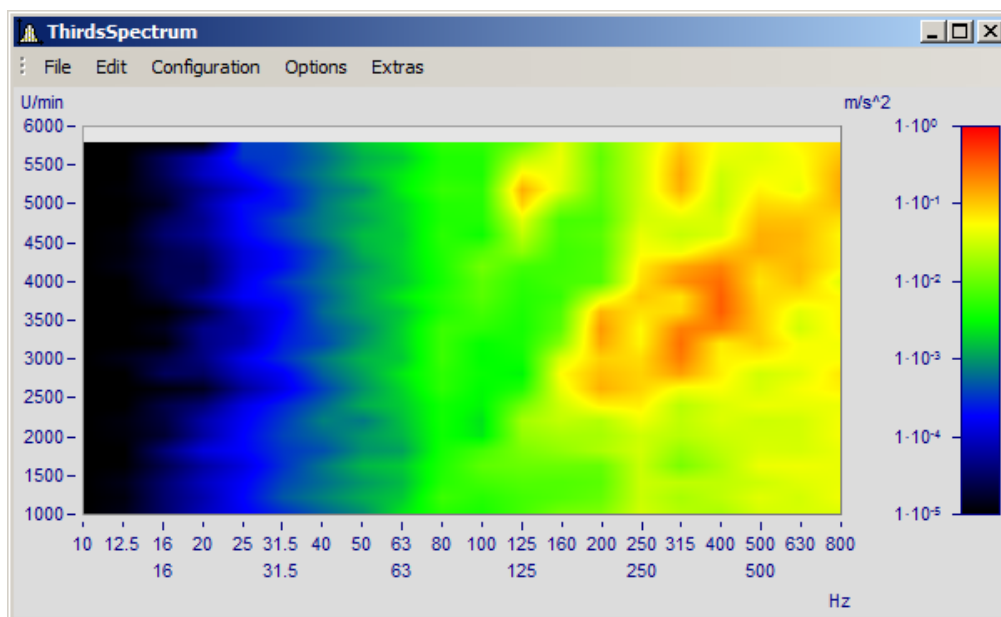
Only the last sample is displayed

11.6.1.8 Third-octave display

Certain methods and forms of representation in the analysis of sounds and vibrations have gained widespread acceptance in industry. 1/3-octave, octave and narrow-band analysis require mathematical operations to be performed, in which the spectra in question are computed from the time signals of vibration pick-ups. The x-axis of the coordinate system is labeled according to the selected frequency bands; the frequencies themselves are determined by industry standards.

Example

Using the mathematical functions in imc FAMOS, the 1/3-octave spectrum was calculated over time. The following color map diagram displays the result, with the x-axis labeled in 1/3-octaves and octaves.



Description

The following nominal pass ranges apply to octave filters. Higher and lower values can be computed by taking the first ten values from the list and multiplying them by ten-to-the-power-of any multiple of three.

Octaves

Center frequency (Hz)	Lower limit (Hz)	Upper limit (Hz)
16	11.2	22.4
31.5	22.4	45
63	45	90
125	90	180
250	180	355
500	355	710
1000	710	1400
2000	1400	2800
4000	2800	5600
8000	5600	11200
16000	11200	22400

The following frequencies apply to 1/3-octaves. Higher and lower values can be computed by taking the first ten values from the list and multiplying them by ten raised to an integer exponent.

1/3-octaves

Center frequency (Hz)	Lower limit (Hz)	Upper limit (Hz)
1000	900	1120
1250	1120	1400
1600	1400	1800
2000	1800	2240
2500	2240	2800
3150	2800	3550
4000	3550	4500
5000	4500	5600
6300	5600	7100
8000	7100	9000
10000	9000	11200

The center frequencies of 1/12- and 1/24-octave bands are located at the center frequencies of the 1/3-octaves and at the intermediate values located at logarithmically equal distances. Here the edges of the 1/3-octave range are used as additional frequency check points.

1/12 octaves

Center frequency (Hz)	Lower limit (Hz)	Upper limit (Hz)
1000	974	1029
1058	1029	1089
1120	1089	1151
1183	1151	1216
1250	1216	1286
1323	1286	1361
1400	1361	1448
1497	1448	1547
..	..	..

1/24 octaves

Center frequency (Hz)	Lower limit (Hz)	Upper limit (Hz)
1000	987	1014
1029	1014	1043
1058	1043	1073
1089	1073	1104
1120	1104	1135
1151	1135	1167
1183	1167	1200
1216	1200	1233
1250	1233	1268
1286	1268	1304
1323	1304	1342
..	..	..

When waveforms in the frequency range are displayed as the result of a frequency-band analysis, the program expects the x-axis to have a certain kind of scaling. Since the frequency bands are spaced logarithmically, the x-axis is labeled with the logarithm of the frequency to correspond to the mathematical functions. The logarithm is then expanded again for display, and the frequencies are marked along the axis, according to the German industry standard. The following table illustrates by a few examples the relation between the x-scaling of the data and the frequency bands, based on the following rule: the $\log_{10}$ of the center frequency is multiplied by ten and then this value is rounded.

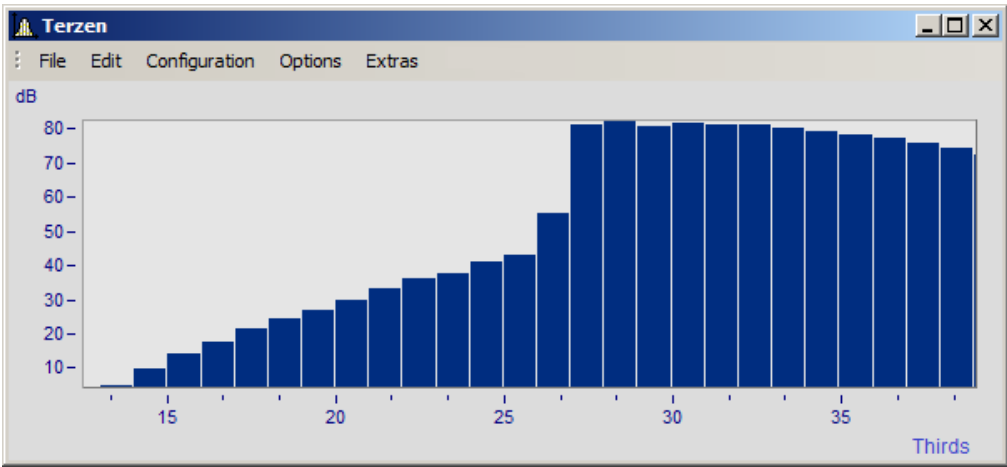
x-scaling	Center frequency (Hz)
..	..
-3	0.5
-2	0.63
-1	0.8
0	1
1	1.25
2	1.6
3	2
4	2.5
5	3.15
6	4
7	5
8	6.3
9	8
10	10
11	12.5
..	..
20	100
30	1000
40	10000
41	12500
43	20000
..	..

The 1/3-octaves are found at the x-positions 0, 1, 2..., the octaves at the positions 0, 3, 6, 9, 12..., 1/12-octaves at the positions 0, 0.25, 0.5, 0.75, 1, 1.25..., and 1/24-octaves are found at all multiples of 1/8.

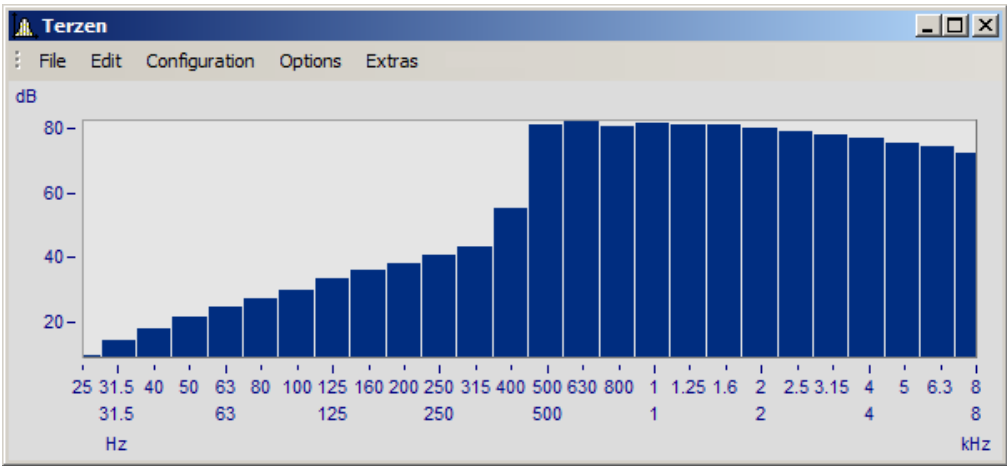
This means that the various bandwidths are expressed as a multiple of a 1/3-octave. Therefore, the delta-x of the x-axis is set to the following values for frequency scaling:

Bandwidth	Delta-X
Octave	3
1/3-octave	1
1/12 octave	0.25
1/24 octave	0.125

Example



A curve window with standard labeling

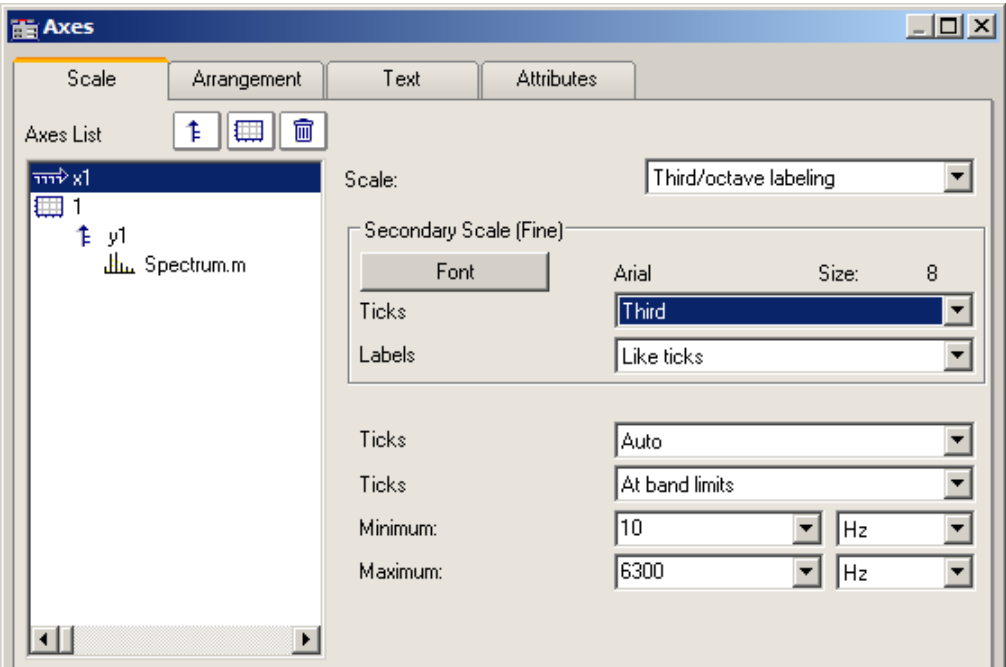


In 1/3-octave / octave labeling

Here, the connection between labeling with the logarithm and the expanded 1/3-octave scaling is apparent.

Mouse Operation

- Set the labeling of the x-axis in 1/3-octave / octaves in the Axes dialog called via the menu item *Configuration/ Axes...* menu. Alternatively, you can double-click on the x-axis labeling.

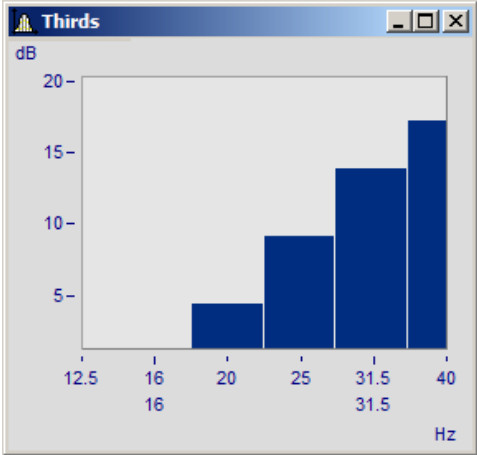
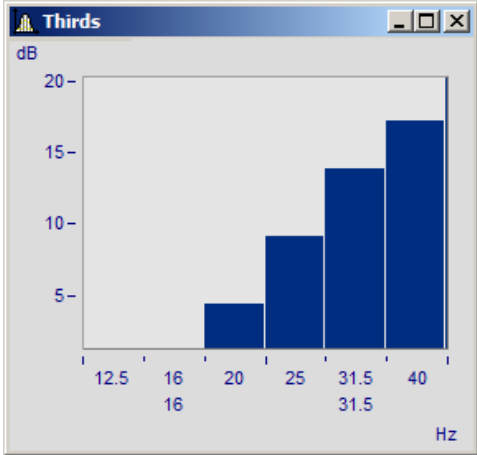


- In the list at left, select the x-axis. Then select under the control *Scale* the option *Third/ octave labeling*

General information

The x-axis scaling is organized into three horizontal layers. The upper layer consists of small ticks and labels (e.g. 1/3-octaves). The middle layer consists of large ticks and the standard font type for the curve window (large labels). The units (Hz, kHz...) are found in the lowest line.

Settings	Description
Secondary Scale (Fine): Font	Establish the font type for the small labels. For true-type fonts, the six-point size is generally quite legible for numbers on the screen.
Ticks	For the small and the large labels, the user can specify the interval between ticks on the axis. Choose from the following options: • Auto: Depending on the domain and the window size, one of the following options can be chosen. • Octaves • 1/3-octaves • 1/12-octaves • 1/24-octaves

Settings	Description
Ticks	Options for centering or aligning ticks. <i>Centered ticks</i> are placed exactly at the <i>center</i> frequency of the corresponding band, with labeling centered under the tick. The edges of the frequency bands are not visible. In the option <i>Ticks to edge</i>, the ticks are located exactly at the edges of the frequency bands, with labels centered between the ticks and at the center frequency.  Ticks centered  At band limits
Minimum, Maximum	The limits of the frequency range to be displayed are set using the combobox <i>Minimum</i> and <i>Maximum</i>. The contents of the list with numerical values are based on the previous selection of ticks for small labeling, which the program assumes is the highest resolution. The entries in the list boxes for <i>Minimum</i> and <i>Maximum</i> are always center frequencies of the band. If numerical values are entered using the keyboard, the precise center frequency will be automatically selected. The range of selectable numbers extends from 10μHz to 400THz, ensuring that the significant physical ranges are covered. Conventional applications will always lie within the range from 10Hz to 20kHz.

Remarks

- **Curves:** When a curve is displayed in a stair-step or bar graph display, the curve is <Shift>ed in the x-direction so that the center of the bars or steps are centered over the labels of the center frequency. The curve is moved to the left by one-half of the sampling time. This applies only to data sampled equidistantly, not to XY-displays.
- Note that in a "steps"-representation of a waveform, the sample-points of the conventional linear display would normally coincide with the corners of the steps. This is not the case for an x-axis labeled with frequency bands, where the sample point coincides with the center frequency, i.e. the midpoint of the step!
- When a small window is scaled over several decades, it is generally not possible to label the axis in 1/24 octaves. Select a coarser resolution in labeling, e.g. octaves.

11.6.1.9 XY-plots

Normally, curve windows display waveforms as time functions. This means the signal's values are indicated as y-coordinates plotted over time represented by the x-axis.

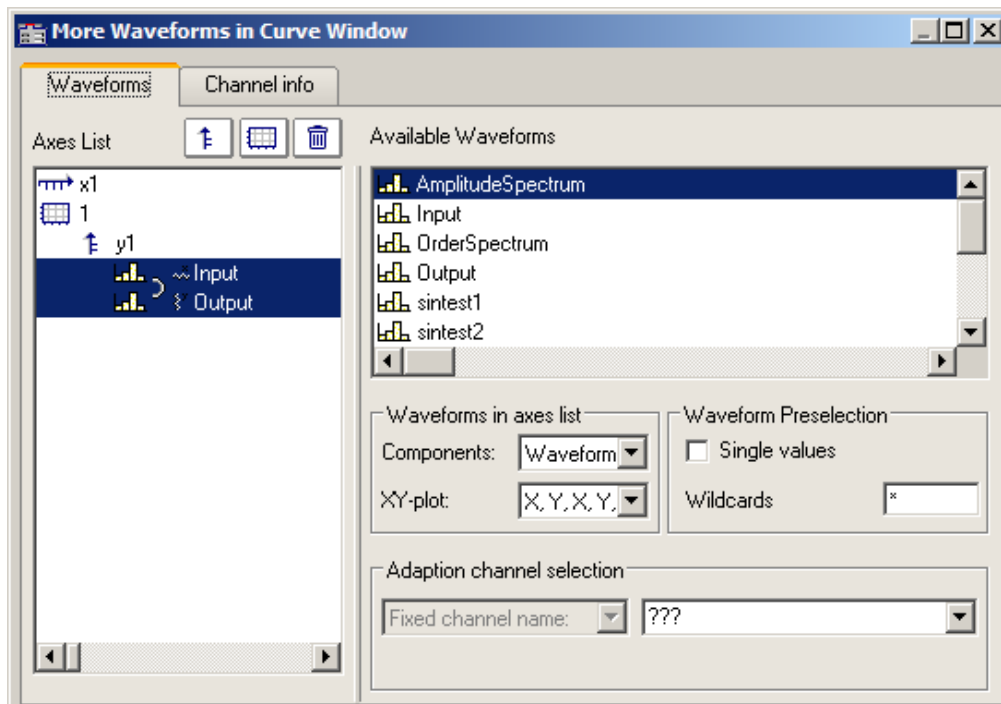
The curve window is also able to display XY-plots, meaning plotting one waveform's values in correlation to another waveform's values.

Examples of XY-plots include **Lissajous figures**, which consist of an XY-plot of sine shaped waveforms of differing frequencies and phases. **Hysteresis curves** for magnetic materials display the correlation between magnetic flux density and magnetic field intensity. **Characteristic curves** enable the assignment of an output variable value to a corresponding input variable value.

Polar plots are another type of XY data. Polar plots are used to display complex waveforms. This type of display can be used for a complex waveform in rectangular coordinates (real and imaginary part) or polar coordinates (magnitude and phase).

Operation

Select "[More Channels in Curve Window](#)" (by right-clicking the mouse in the curve window). Stack the XY-plots vertically. Select the two waveforms and then in the box *Waveforms in axes list* Select the entry *x,y,x,y* from the drop-down list for *XY-plot*:. If you wish to change the axes, select instead the other order: *y,x,y,x*.



Remarks

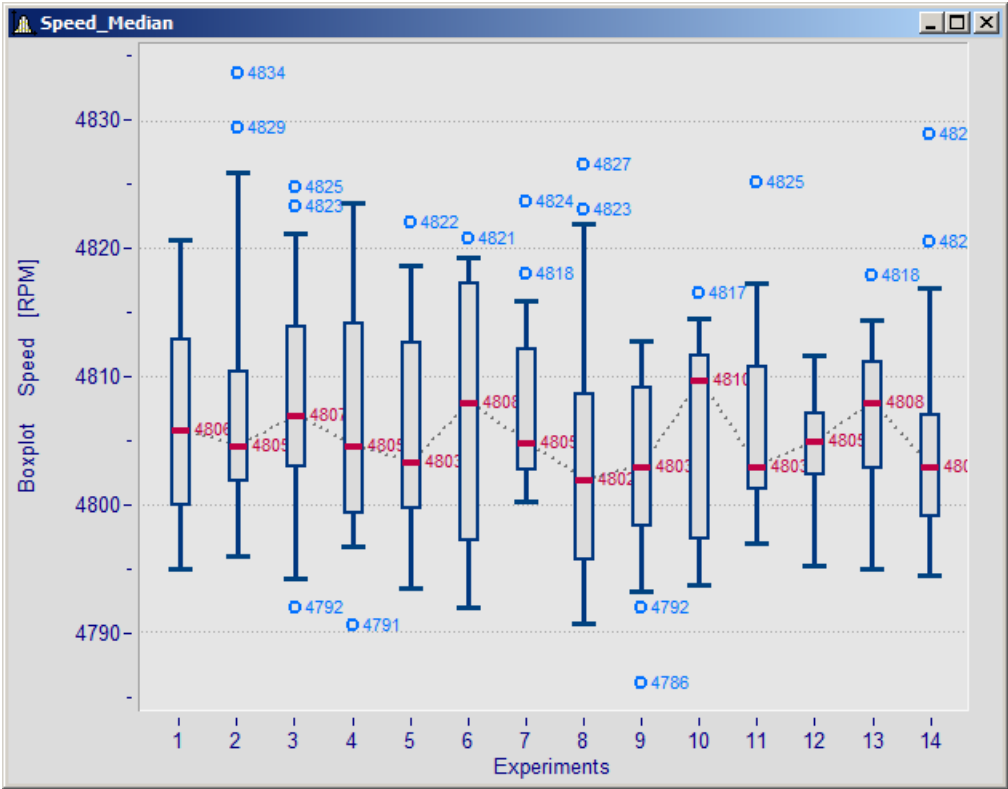
- In all XY-plots, the measurement cursors follow the parameters during operation in "Measure" mode. See chapter section '[Measure](#)'.
- Computing a polar plot requires less computation if the complex waveform is provided in rectangular coordinates. To increase the speed of the graphic display for a waveform in polar coordinates, transfer it into rectangular coordinates.
- The x- and y-values in an XY plot are paired *correctly* with respect to time, *not* point by point! Thus, the two waveforms used for an XY-plot do not have to have the same x-scale! Therefore x-offset and x-delta (sampling rate) are considered, but not the trigger time.
- If a waveform is already defined as complex or of XY-type, simply click on the curve window icon to display it.

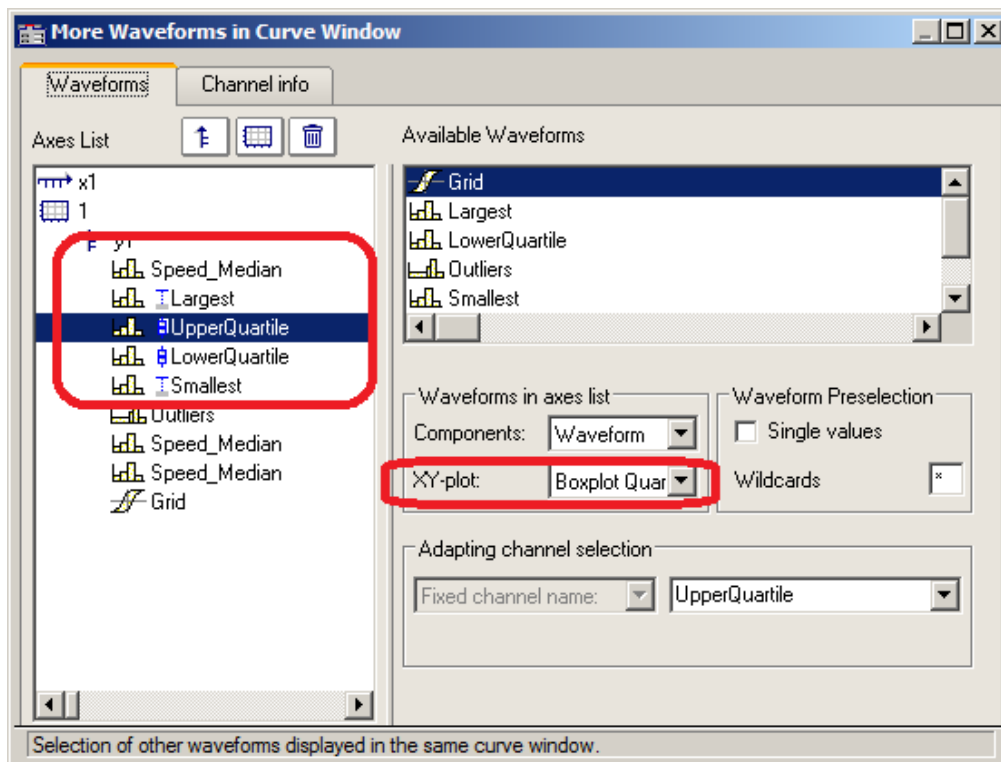
11.6.1.10 Boxplot

The Boxplot (also called Box Whisker Diagram) depicts a statistical distribution by displaying the minimum value, the maximum value, bottom quartile, top quartile and the median.

In this display, the minimum and maximum values are represented by whiskers, the quartile boundaries as a box, the median as a line.

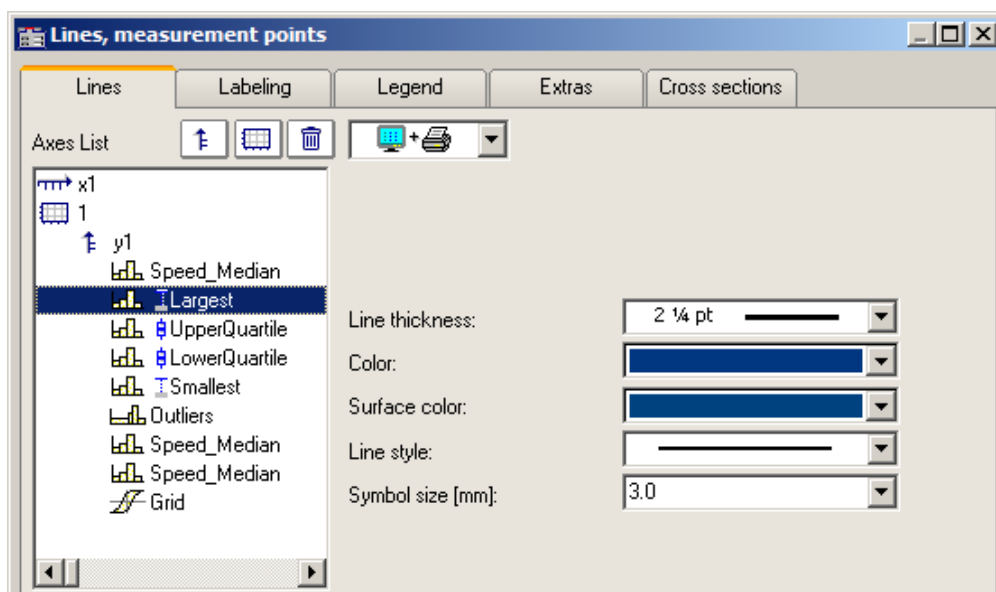
For a curve window display, each of these statistics is saved in a separate channel, e.g. one channel with the median values, another channel of maximum values...



Making the settings:

In the dialog "More Channels... in Curve Window", the channel with the median value is initially added. For this channel, no special settings need to be made. Next, the channels for the whiskers and the quartiles are added. For each of these channels, the property XY-plot is set to "Boxplot Whisker" or "Boxplot Quartile", where it does not matter in this case whether a whisker is the top or bottom one; the curve window finds out on the basis of the numerical value. The order of the channels below is arbitrary. They do not even need to be complete, however, the median must always be the first channel.

Subsequently, the line properties are defined:



Depending on the line selected, different properties are available for selection on the right side. The line structure is only applied to the vertical whisker line. The surface color is used for filling the box representing the quartiles and for filling the whiskers' horizontal bar. Automatic color selection for the surface results in transparent fill of the quartiles' box. The color (line color) is used for connecting- and borderlines. Automatic color selection causes this color to be same as the median. Most properties can be set separately, however, for the 2nd quartile defined, only the fill color needs to be selected, since the rest has already been determined by the 1st quartile defined.

The display is performed point by data set point. All data sets must have the same sampling rate. The median itself may be an XY-channel, but all others may only be normal, equidistant data sets (it may be necessary to display the .y-component alone). In the case of an XY-channel, the sampling time of the y-component must match that of the other channels.

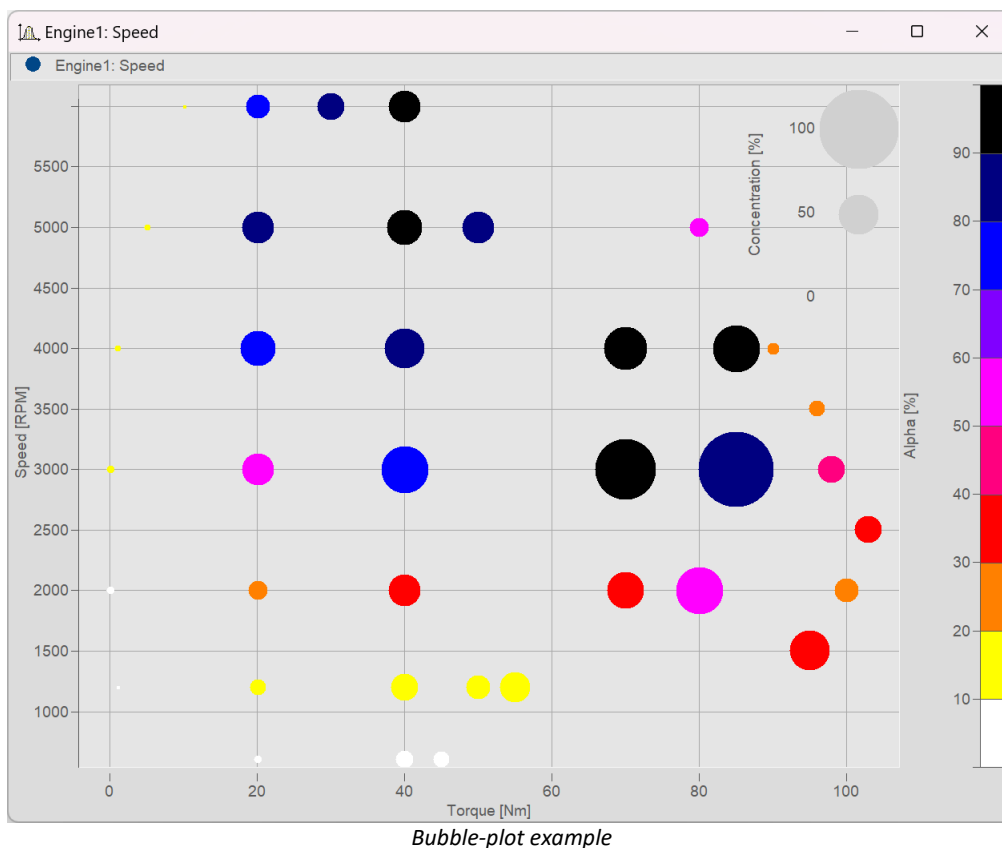
Remarks

- A combination with a color-determining channel is not possible.
- Measurement cursors and other curve window functions have only restricted functionality.
- To display any outliers, supplemental channels are used, for which the legend entry can be suppressed, for instance.

11.6.1.11 Bubble Plot

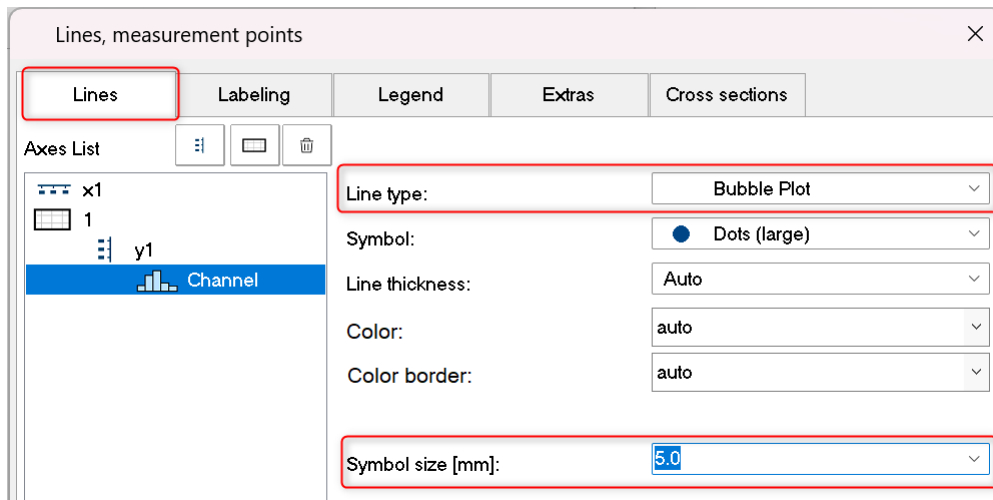
In a **Bubble Plot** (also called "Bubble Chart"), symbols are displayed which represent three (or optionally four) numerical values. Each point in the diagram is symbolized by a disk shape (bubble) in a typically Cartesian coordinate system. One data set (or in the case of XY, two data sets) determines the **position** of the bubble's center point. One data set determines its **size**, and optionally an additional data set, which could represent a kind of categorization of the data points, determines the **color** of the bubble.

The **necessary condition** is that the **sampling rate** of all the data sets be identical.

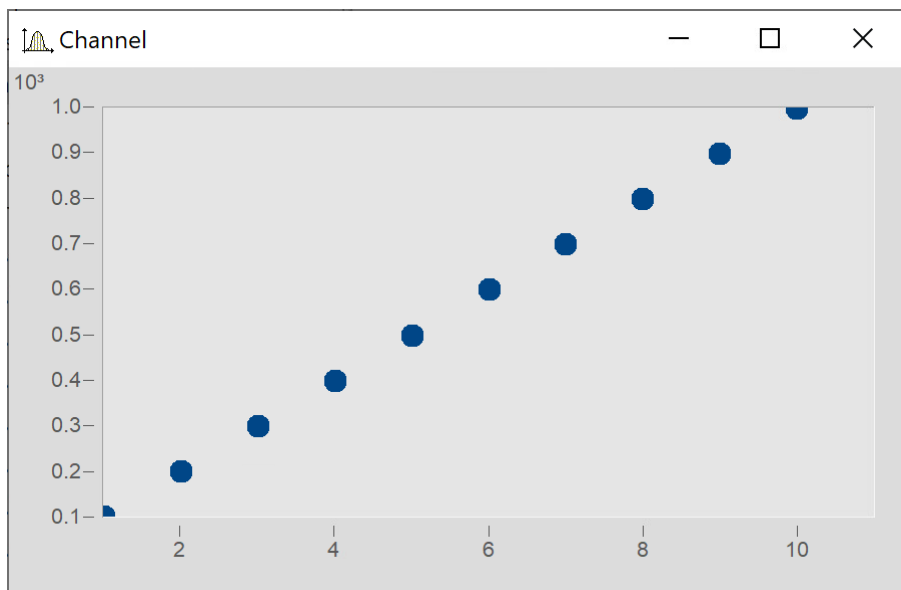


Settings:

Within the **Display type** "Default", begin by setting the *Line type* to "Bubble Plot" in the settings under **Lines**. The initial size can be modified using *Symbol size [mm]*:

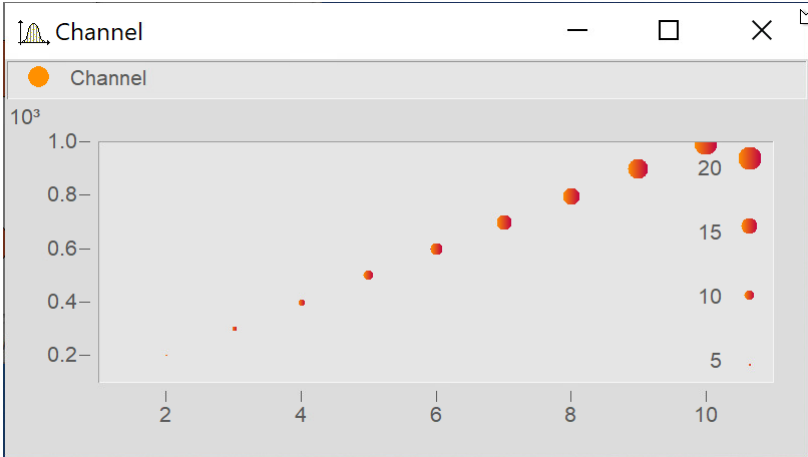


Depending on the *Symbol* selected, all values are displayed as *Dots (large)* etc. in the same size.

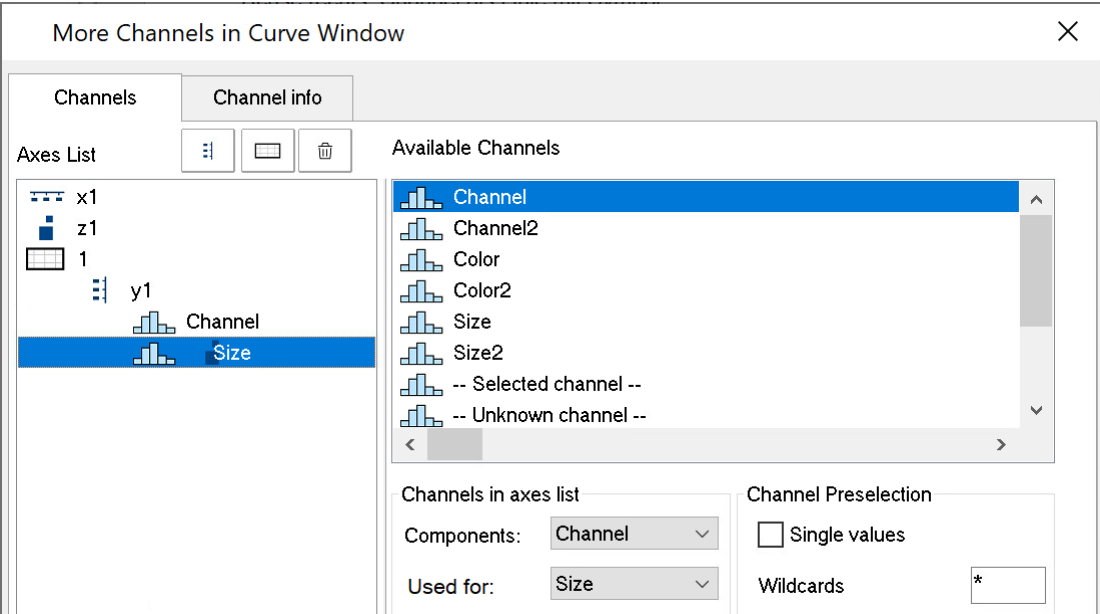


Size of bubbles corresponding to values of another data set

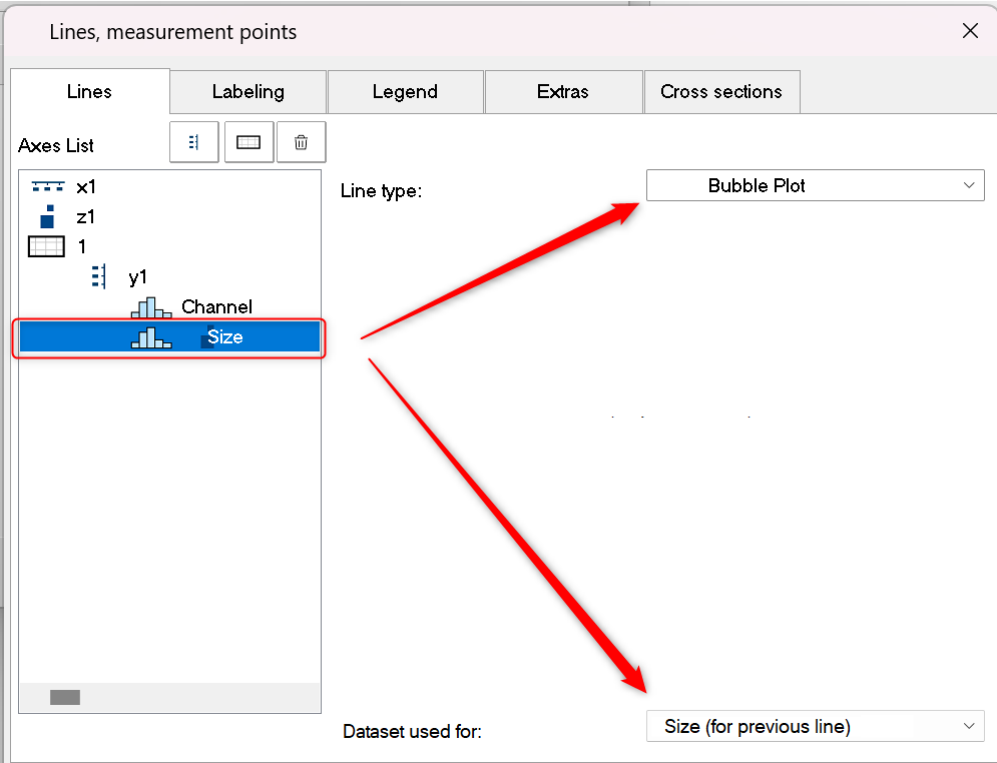
Next, the variable "Size" is intended to determine the size of the bubbles, as illustrated below:



In the dialog **More Channels in Curve Window...** add the variable "Size".



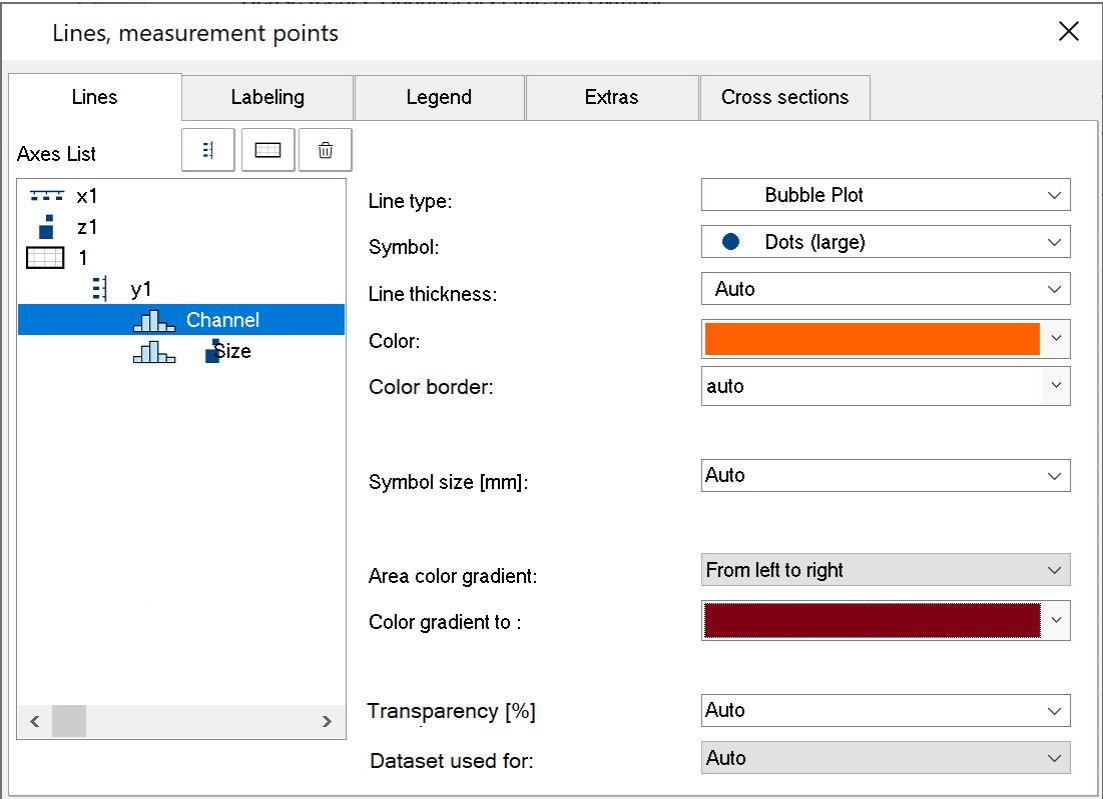
Go to the dialog **Lines** by clicking on the button *Topic* and set the *Line type* for the variable "Size" also to *Bubble Plot*.
Next, under *Dataset used for:* select *Size (for previous line)*:



Next, in the dialog **More Channels...** set *Used for:* to *Size*

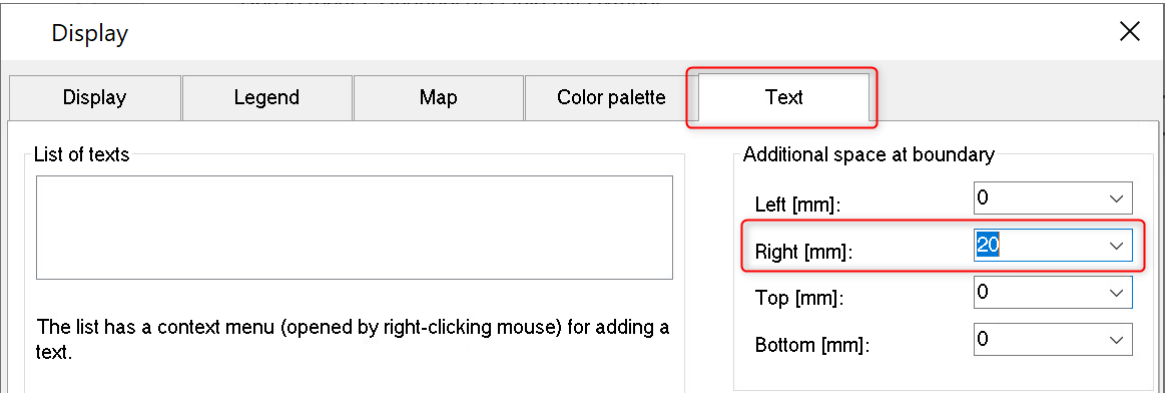
Bubbles with color gradient

Coloring the symbols with a color gradient:

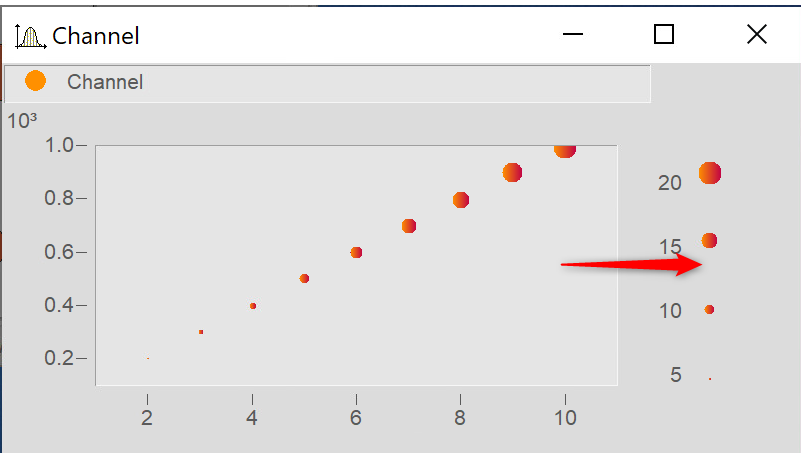


Adjusting the boundary

By default, the scaling axis for the variable is located within the curve window. In order to position it on the outside, it is necessary to first expand the boundary. On the tab *Text* in the dialog for the menu item **Display**, enter a right-side boundary:

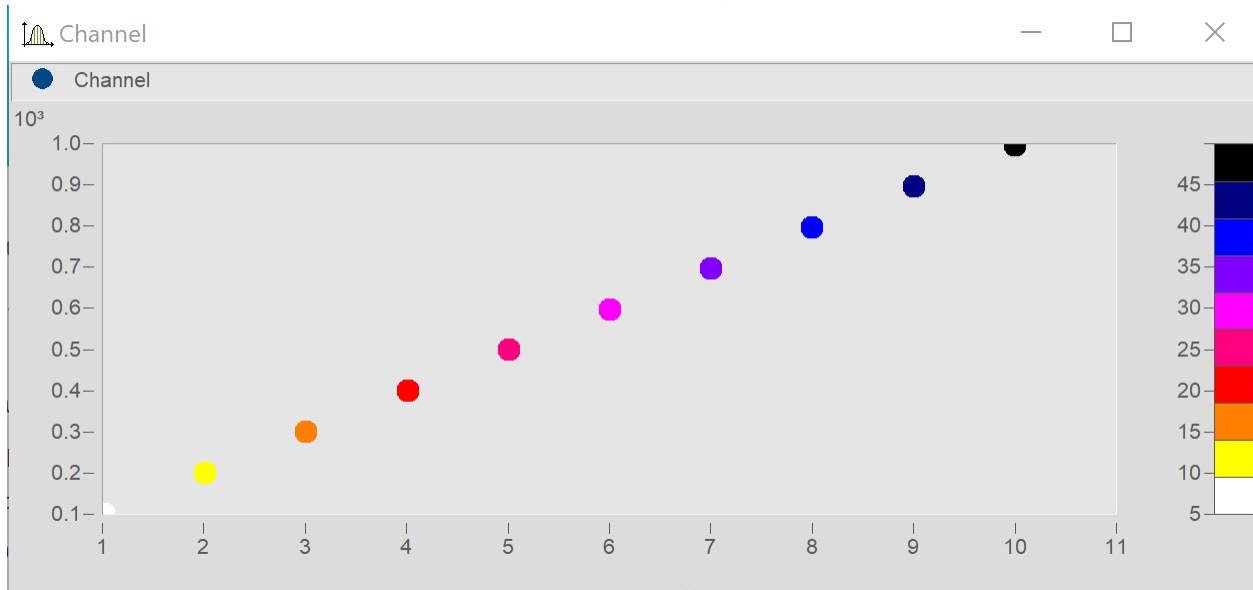


Subsequently, use the mouse to move the axis to the edge:



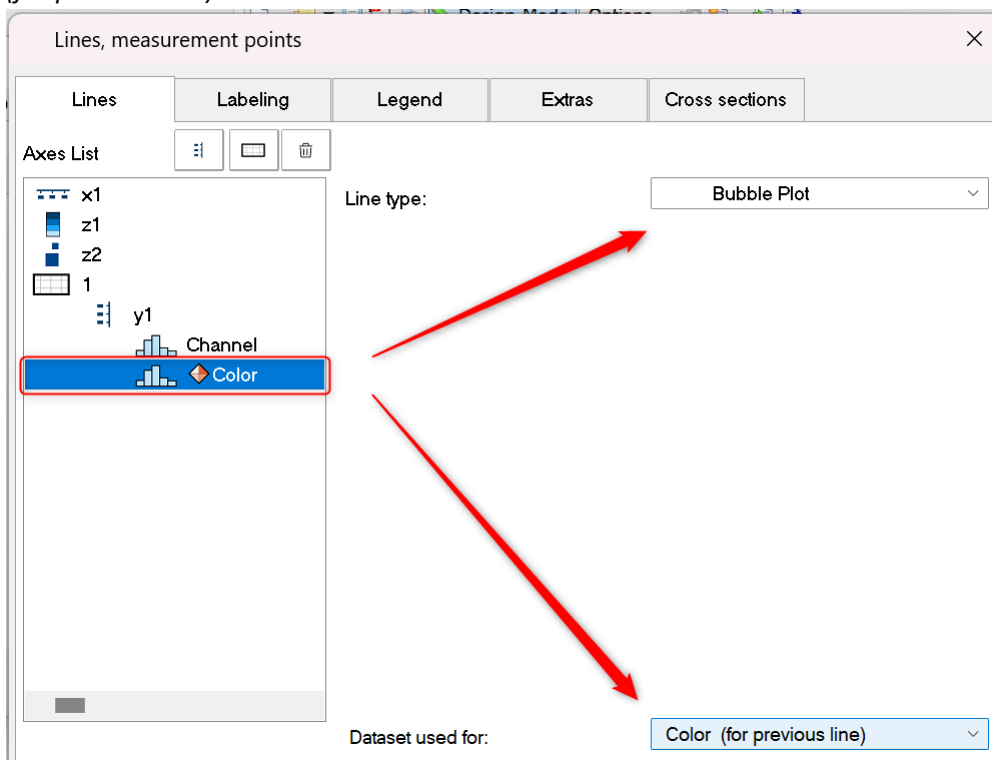
11.6.1.11.1 Additional Bubble Plot displays

Bubble Plot with color palette

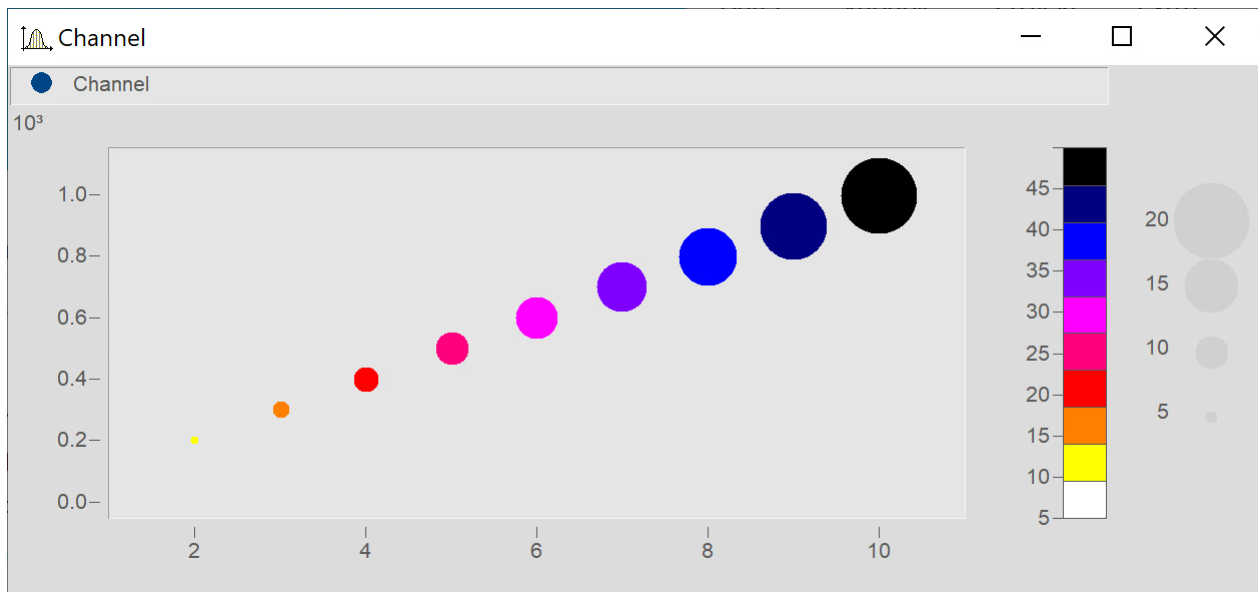


Matches "[Size of bubbles corresponding to values of another data set](#)"⁷¹⁹, except that the second data set is used for the color palette.

In the dialog **Lines**, set the color-determining channel to *Line type: Bubble Plot* and *Dataset used for Color (for previous line)*.



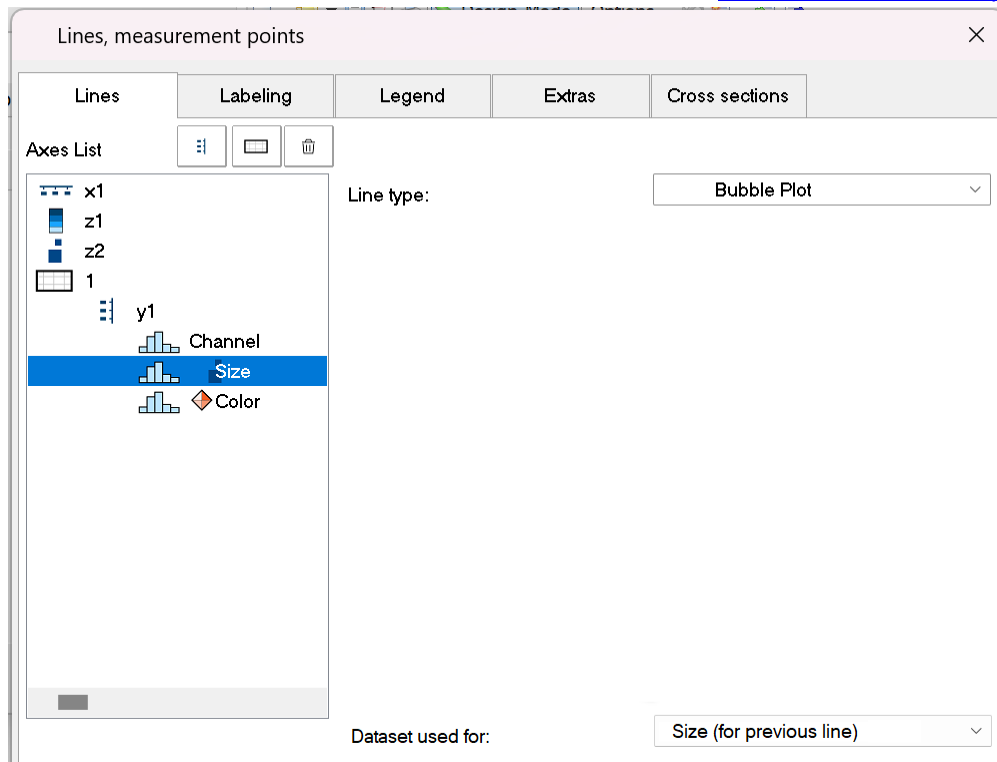
Bubble Plot with size channel and color palette



Matches "[Size of bubbles corresponding to values of another data set](#)" ⁷¹⁹ except for the following:

A third data set is used for the color palette.

Add the data set for the color and set it in accordance with [Bubble Plot with color palette](#) ⁷²².



In the dialog **Lines**, set all three data sets to *Line type: Bubble Plot*.

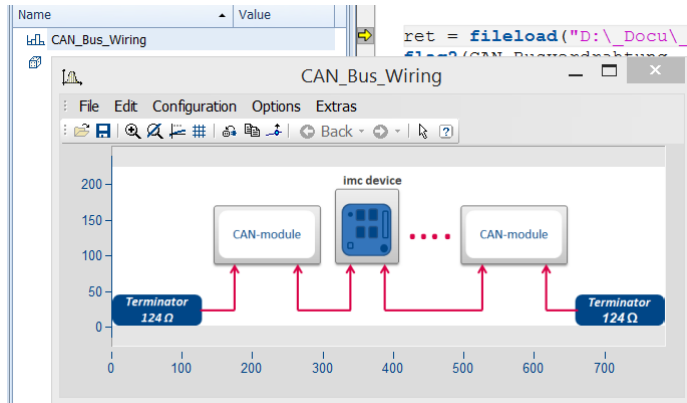
Make the following settings for the respective channels: *Datset used for: Color (for previous line) and Size (for previous line)*.

Expand the edge (see [here](#) ⁷²¹) and use the mouse to arrange the scaling axes for the variables Color and Size.

11.6.1.12 RGB-image

An RGB-image variable consists of a segmented data set in which each sample represents a pixel's RGB-code. When loading an image using `FileLoad( ..."#Picture.dll|Picture Format" ...)`, a color-flag is set in the properties of the variable, which can be queried by means of `Flag?` (Variable, 1). If the RGB-flag is set, when the curve window is opened, the property `Image from RGB-values`^[763] is automatically set under *Effect* in the dialog *Lines\Extras*.

The y-axis and x-axis must both have the same scaling so that the image will be displayed without distortion. This is ensured by means of the property *Resolution* in the dialog *Axes\Arrangement*^[747].



The color stages of the pixels are displayed exactly for the `Line type setting`^[756] "Steps". For the line type "Lines", the color gradients are interpolated.

If a data set has `multiple events`^[801] each having one image, these can also be displayed when the events have appropriate coordinates. With overlapping coordinates, only the last event may be visible.

The **X-offset** determines the offset in the graph, the event's trigger time is not taken into account.

If any settings are made which the system does not support, the image is not displayed. Settings which are not supported include invalid data formats (XY-data, TSA), XY-overlays over the curve window, selections of individual samples, skipping of segments, and period comparisons.

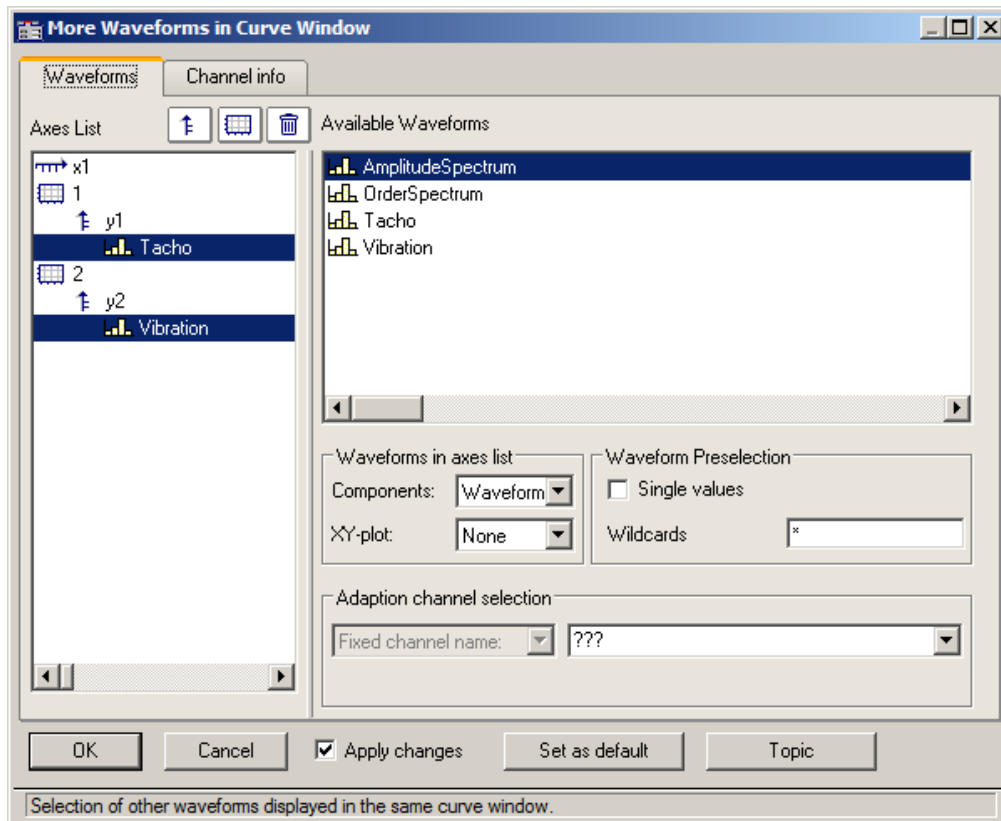
11.6.2 Showing data in curve window

A dialog is available for setting the waveforms to be displayed, the coordinate system and the y-axes in the curve window.

Up to 40 coordinate systems can be represented in a stacked display in the curve window, and each coordinate system can have y- and z-axes with different scaling. A coordinate system and a y-axis (and possibly a z-axis) from this coordinate system must be assigned to each waveform displayed in the window.

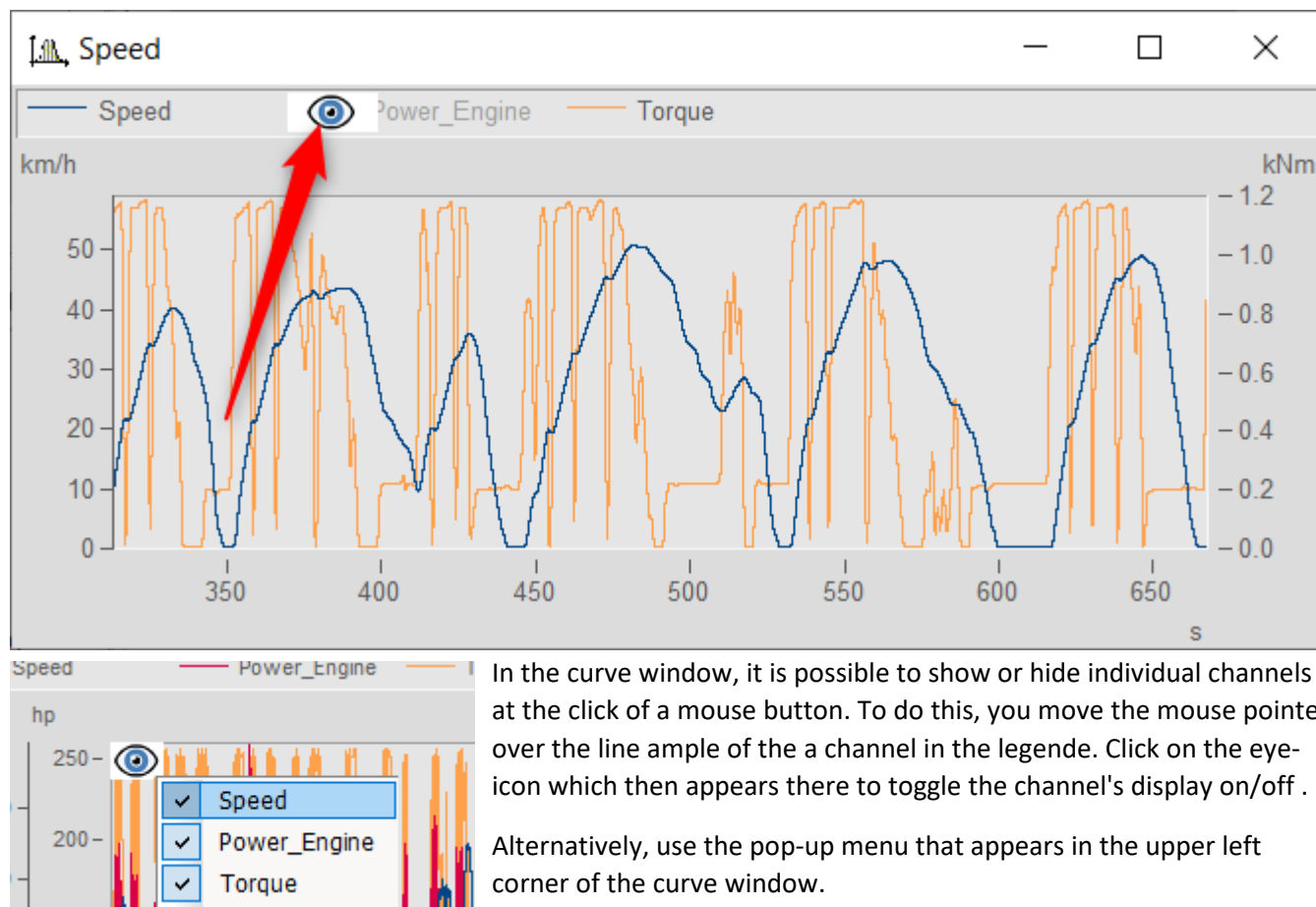
Open the dialog by right-clicking the mouse in the curve window and selecting *More Channels.....*

A short help message appears when the mouse pointer is held for a few seconds over a dialog object (provided the curve window help options are enabled).



The space allocated in the layout to the Axes list and the Available Channels respectively can be altered by dragging the border between them using the mouse.

The subsections below respectively describe the individual dialog elements.

Showing/hiding channels at the click of the mouse

11.6.2.1 Axes list

This *Axes list* reflects the current structure of the curve window. The symbols for the individual coordinate systems are found in the first column. The y-axes displayed in a coordinate system are indicated underneath, somewhat indented. Waveforms assigned to a y-axis are listed after the y-axis. Each waveform is specified by a symbol and its name.

Waveforms can be **added to Axes list** from the list of *Available Channels* using drag&drop.

Drag&drop applied to one or more selected entries **moves** these. When you hold down the CTRL key, a copy of the respective item is moved.

Buttons for editing the axes list

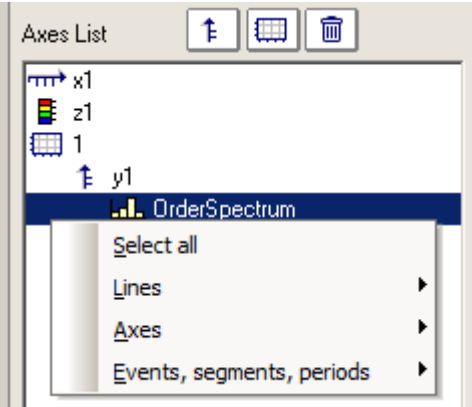
Icon	Description
	A new y-axis is created. To do this, click briefly on this button, which changes the mouse pointer to a suitcase symbol. Click on the target position desired. The new y-axis is inserted above the row over which the mouse pointer is located.
	A new coordinate system is created.
	The lines selected in the axes list (coordinate systems, axes, waveforms) are deleted. Use the <Del> button as an alternative.

Symbols used in the axes list

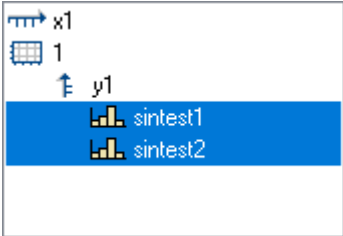
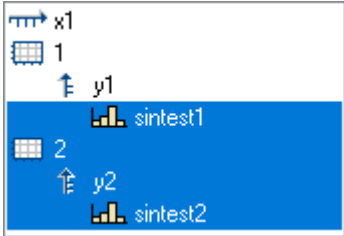
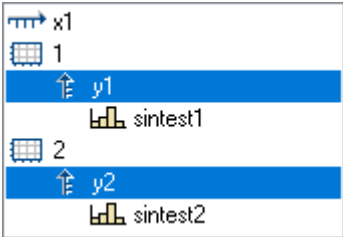
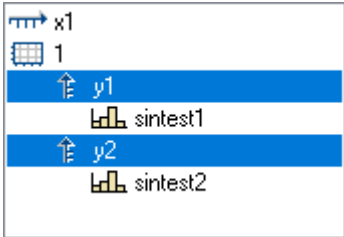
Icon	Description
	x-axis ; always at beginning of axes list. All following coordinate systems are scaled on the same x-axis
	z-axis ; only in <i>3D, color map and waterfall diagram</i> . The z-axis can be scaled only with a fixed z0 and delta-z .
	Indicates an independent coordinate system . All further axes and waveforms listed until the next such symbol appears or until the end of the list belong to this coordinate system.
	Indicates a y-axis . All further waveforms listed until the next y-axis belong to this Y-axis.
	Two consecutive waveforms in the list are superposed
	XY-Display : Two consecutive waveforms in the list are superposed; the data of the waveform denoted by a "y" are plotted over the values of the waveform denoted by a "x"
	3D-Display : Three consecutive waveforms in the list are joined in a multi-dimensional relationship. The data of a waveform designated y and the data of a waveform designated z are associated with the data of a waveform designated as x. This relationship is for the purpose of 3D display, but can also be used for a color map.
	Variable not present : The original waveform at this position has been deleted. Or: a loaded curve configuration expects a waveform at this position
	Second component missing : The waveform is already designated as one component of an XY-waveform, the other component is missing
	A normal waveform with equidistant x-scaling
	Empty waveform : A normal waveform with a length of 0
	Single value , normal waveform with a length of 1
	XY-waveform with monotonous x-track (time waveform)
	XY-waveform with NON-monotonous x-track
	A digital waveform

Icon	Description
	A complex waveform in magnitude/phase- or real/imaginary display

Context menu in Axes List



The following menu items are available:

Menu item	Description
Select all	Selects all lines in the list.
Lines	A <i>line</i> is the graphical representation of one or more data sets, in other words a <i>waveform</i>. Normally an entry in the axes list will indicate the simple representation of one data set as a line. The following items affect the selected lines.
Individual coordinate systems	This produces a <i>stacked</i> representation of curves.  before  after
All on 1 axis	All selected lines are displayed with only one axis. Any obsolete axes and coordinate systems are deleted.
Individual axes	A separate axis is displayed for each line.
Axes	The following items affect the selected axes.
All in 1 coordinate system	All axes are incorporated into only one coordinate system. Obsolete coordinate systems are deleted.  before  after

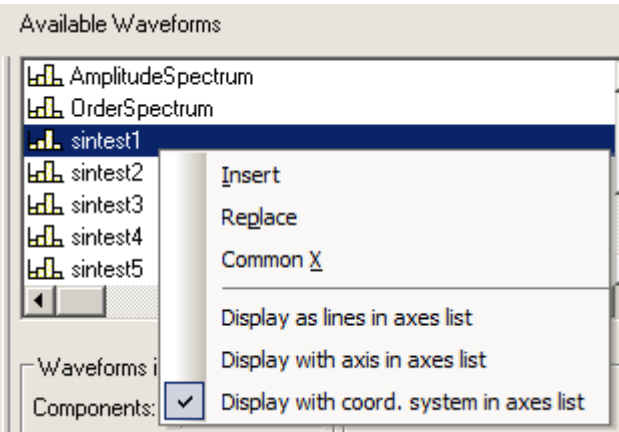
Menu item	Description
Individual coordinate systems	A separate coordinate system is displayed for every (y-) axis.
Events, Segments, Periods	The following items are relevant only to special data types (segmented and multi-shot waveforms)
Combine	Various lines are attached to each other, if the numbering permits. The range of possibilities is dictated by the exigencies of the corresponding dialogs for selecting structural elements. before after
Separate with same layout	One line consisting of several structural elements is dissolved into its components. The order of the elements is retained. If the original line was provided with, say, a coordinate system, then all the new lines will each receive one as well. before after
Separate into individual lines	This is the contrary of <i>Combine</i> . Multiple lines, which each amount to a structural element, are created.
Separate into individual coord. systems	Each structural element receives its own coordinate system.
Reverse order	Reverses the order in which the lines appear.
	before after

11.6.2.2 Available channels

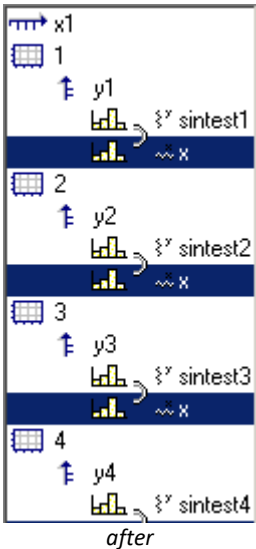
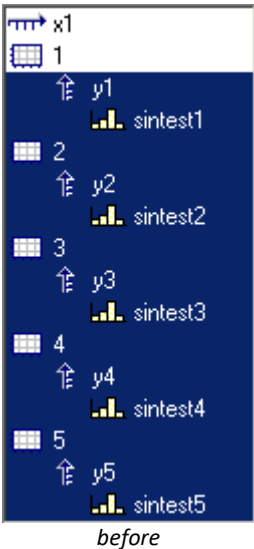
This list contains the waveforms which can be displayed in the curve window. Waveforms selected from this list can be fetched into the axes list using drag&drop (starting from the left edge of the list, the mouse pointer has the shape of a hand).

Context menu in available channels

Right-clicking mouse in the *Available Channels* list to open the context menu:



Menu item	Description
Insert	The waveform selected is added to the axes list. Depending on how the controls below are set, the waveforms are automatically created along with y-axes or coordinate systems: • Display as lines in axes list, Display as axis in axes list, Display with coord. system in axes list
Replace	The waveforms in the axes list are replaces with the waveform selected under <i>Available Channels</i> . If more than one waveform is selected under <i>Available Channels</i> , this menu item is not enabled.
Common X	The waveforms selected in the axes list contain an x-component.



11.6.2.3 Waveform preselection

Waveform Preselection: Single values

Single values are only indicated in the list of Available Channels if this option is selected. Frequently, single values are generated in FRAME programs or imc FAMOS sequences, for example, as control variables, parameters for functions or indexed variables in loops. The single values are not of interest for display in a curve window.

This name only

The user can specify here a filter for the name of the waveforms to be indicated in the list. Only those waveforms whose names match the specified filters are included. The program does **not differentiate** between **upper** and **lower case letters**. Use the wildcards '*' and '?' when specifying the filter.

The wildcard '*' indicates any number of any given characters, '?' indicates any one given character. The wildcard can also be placed at the beginning or end of the filter.

Examples for wildcards

*	All waveforms
a*	All waveforms whose names begin with 'a'
Channel?	All waveforms whose names consist of 'Channel', followed by any character
Channel	All waveforms in whose names the string 'Channel' appears
a*;t*	different filters are separated by ";"

11.6.2.4 Waveforms in axes list

Waveforms in axes list: Components

What components are selected affects the current selection in the axes list. For two-component waveforms, specify whether the complete waveform should be indicated (XY, polar plot) or only a particular component of the waveform.

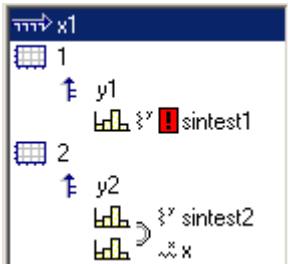
Component	Description
???	The settings for the selected waveforms do not match.
Waveform	Complete display of variables according to type. XY-Waveform: the entire waveform "Y over X" Complex data: polar plot
.x .y	The specified components of XY-waveforms are displayed
.r .i	Real or imaginary part of a complex waveform in real/imaginary display
.m .p	Magnitude or phase of a complex waveform in magnitude/phase or decibel/phase display

Waveforms in axes list: XY-plots

Superposition of normal single-component waveforms.

XY-plots	Description
???	The settings for the selected waveforms do not match
None	No XY-plot has been determined
x of XY	In an XY-plot, this waveform is the x-component
y of XY	In a XY-plot, this waveform is the y-component
x, y, x, y ..	The selected waveforms become x- and y-components in this order
y, x, y, x ..	The selected waveforms become y- and x-components in this order
z	In multi-dimensional displays, this waveform is the z-component.

An XY-plot always involves two or three consecutive waveforms in the axes list, which must both belong to the same y-axis. All must be single-component waveforms (a normal waveform or a specified component of a 2- or 3-component waveform). The two waveforms in an XY-plot are marked by a parenthesis in front of their names.



A small symbol indicates whether a waveform acts as the x-, y- or z-component in an XY-plot.
If a waveform has already been designated as part of an XY-plot, but it's counterpart is missing, this is indicated by a red exclamation mark in front of its name.

Dimensions for 3D displays

If you select the display type *3D*, the available dimensioning options for 3-component waveforms are presented here. Three waveforms as associated spatial dimensions are denoted by a parenthesis in front of the name.

XY-plots	Description
???	this means the waveforms selected have different settings
None	no multi-dimensional relationship defined
x	this waveform is the x-component in a multi-dimensional relationship
y	this waveform is the y-component in a multi-dimensional relationship
z	this waveform is the z-component in a multi-dimensional relationship
y, x, z	The selected waveforms are respectively the x-, y- and z-component of a 3D display

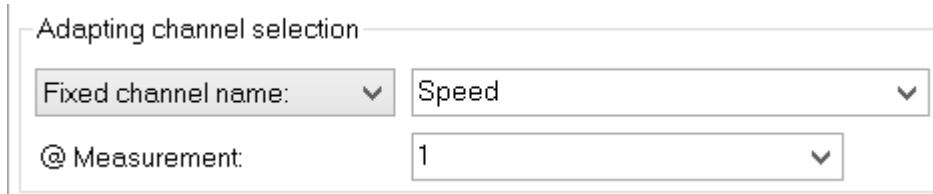
The multi-dimensional relationship always applies to three consecutive waveforms belonging to the same y-axis. All waveforms must be one-dimensional (so either normal waveforms or defined as one component of a 3-component waveform).

Also, there is a small symbol indicating whether a waveform is acting as the x-, y- or z-component. If a waveform is already defined as a dimension in a multi-dimensional relationship, but the complementing dimensions are still missing, a red exclamation mark will appear in front of the waveform's name.

The z component of the display is always the 3rd component. X, y or y, x come first. Please note that the purpose is to display a surface. This is a function $y = f(x, z)$. Compare this with the display of with normal time domain data, where $y = f(x)$. The difference is the same as with waterfall display. Y is the amplitude. X and Z are the independent coordinates. This corresponds to segmented waveforms where you have dx and dz for the two dimensions of the matrix. The values contained are y values. So a segmented waveform is a function as well: $y = f(x, z)$. Thus, a surface will be plotted like with segmented waveforms, but with segmented waveforms all x and z values are equidistant. With xyz display you can chose any pairs of x and z values.

11.6.2.5 Adapting channel selection

The dialog portion *Adapting channel selection* is needed in conjunction with the [Data Browser](#)⁵⁹⁷.



Adapting channel selection

Fixed channel name: Speed

@ Measurement: 1

Fixed channel name

Lets one specify that only a waveform having the channel name entered in the edit box at the right is displayed. Toward this end, "Unknown channel" from the window *Available Channels*: is added to the axes list and configured accordingly.

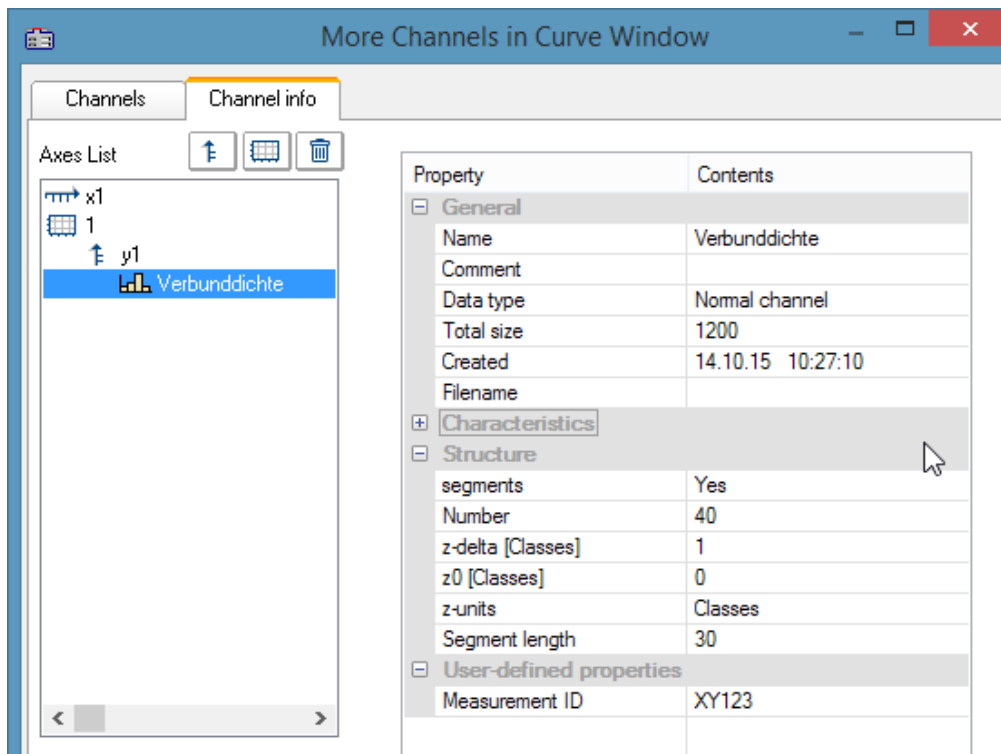
Selected channel

The curve window can be configured so that the waveform having the corresponding number in the Data-Browser is displayed at the this location. The pertinent number for the selected channel can be set in the dropdown list. Toward this end, "Unknown channel" from the window *Available Channels*: is added to the axes list and configured accordingly.

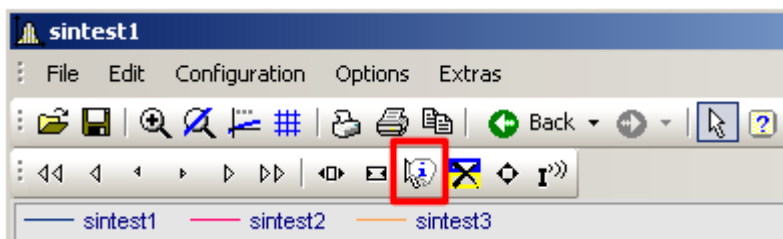
Additionally, the measurement selected in the Data-Browser can be set as a further display criterion in the dropdown-list *@Measurement*.

11.6.2.6 Channel info

The dialog "More Channels in Curve Window" contains a second page, *Channel Info*. Here you will find information about the properties or contents of any selected waveforms. The user-defined properties are also listed here.



In Select-mode, the dialog can also be reached via a line's context menu under the heading *Channel properties*, or in the menu ender [Configuration / Arrangement / Channel properties](#) ⁸⁰⁸ or via the corresponding symbol in the *Navigation Toolbar*.

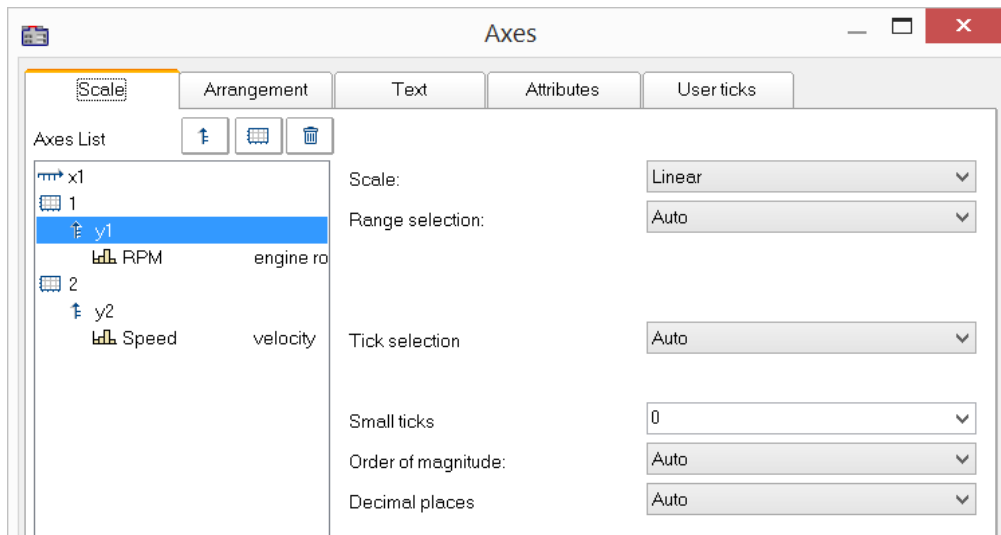


11.6.3 Axis configuration

Each axis (x, y, z) can be scaled manually, with a linear, logarithmic or dB scale. Linear scaling is useful for all time functions, whereas logarithmic scaling is appropriate for spectra. In logarithmic scaling, the waveform is stretched in small coordinate range and compressed at higher x-values.

Mouse Operation

- In the curve window's menu *Configuration*, select the item *Axes...* or to double-click in the region of the axis' scale labels. A dialog for scaling the axes then appears.



The Axes list shows the curve window's structure. Here, select the axis, which you wish to edit on the right side of the dialog. Selecting multiple axes is also possible.

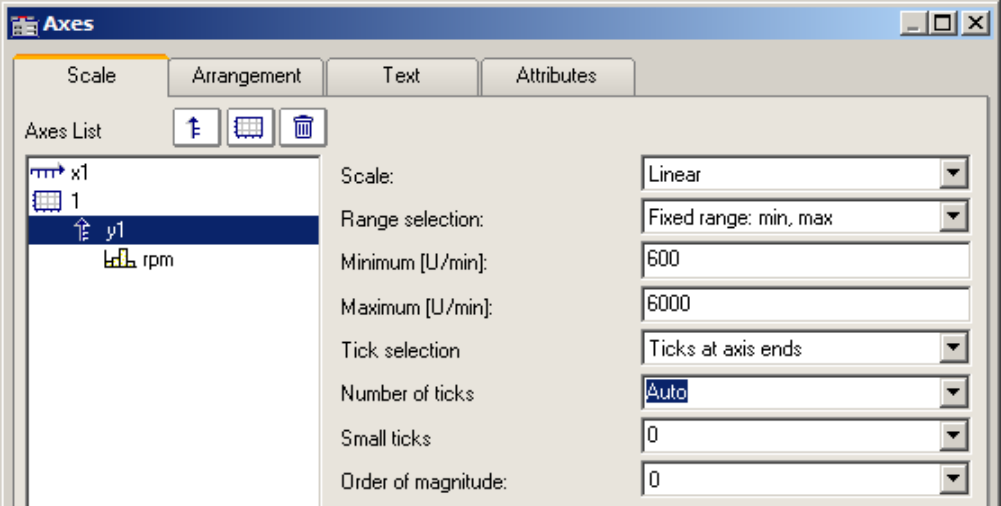
11.6.3.1 Scale

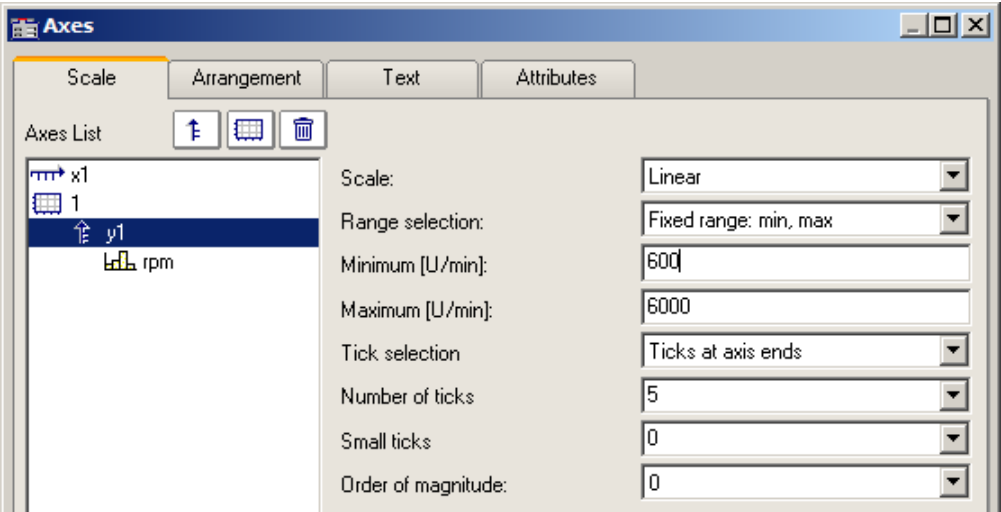
Scaling of the axes. An axis' domain can be defined in a number of ways.

Scale

The axis can be scaled as *linear*, *logarithmic* or in *dB*. In Date/time display, the scaling is always linear. When the mode *Rounding* is selected for logarithmic labeling, powers-of-ten will be written to the axis as long as the range displayed is large enough.

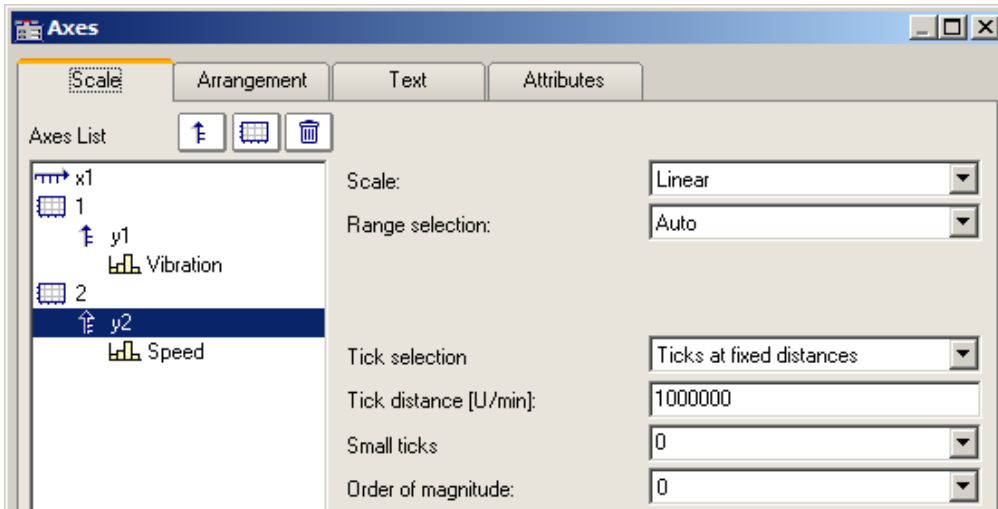
Range selection

Range selection	Description
Rounding: min, max	Specify the axis' domain by specifying the minimum and maximum. The specified values of the waveform are then rounded off in such a way that, taken together with the set number of markings on the x-axis, that axis appears labeled with nice, round numbers. Note that the maximum must always be greater than the minimum.  This menu option is not enabled when the option <i>Tick selection</i> is set to <i>Ticks at fixed distances</i>.

Range selection	Description
Fixed range: min, max	Here, the range limits and the number of markings are defined. For instance, a minimum specified as 10.0, a maximum of 40.0 and four markings on the axis result in the values 10.0, 20.0, 30 and 40.0 being marked along the axis.  In logarithmic display, the factor with which to multiply the individual tick points must be stated, and it must be more than 1. If you set the first value to 10, the factor to 2 and the number of ticks to 3, then the values 10, 20 and 40 are marked on the axis.
Auto	Determining the range is automatic. The entire curve is displayed in the curve window.
Automatic with zero	With this setting, the zero point is always visible. If the signal values lie within the range 2.0 ... 2.5, the range displayed is 0.0 ... 2.5. This corresponds to an oscilloscope's DC-setting. If the function values are distributed at only a small distance around the mean value, the deviations from the mean value will be interpreted as interference or noise. In that case, a display with visible zero intercept is always set automatically.
Like previous axis	This option is only available if more than one y-axis is displayed in the curve window and an axis other than the first one is selected. If you select this display style, the axis concerned appears in the same style as the preceding axis in the list. In this way you can make multiple curves be displayed with the same axis, and have this one axis state the scaling for all curves correctly.
0-Symmetric	The absolute value of the highest value determines the range, which is set as symmetric around zero. For instance, if the values lie within the range 2.0...2.5, the system sets the range as -2.5...2.5.

Tick selection

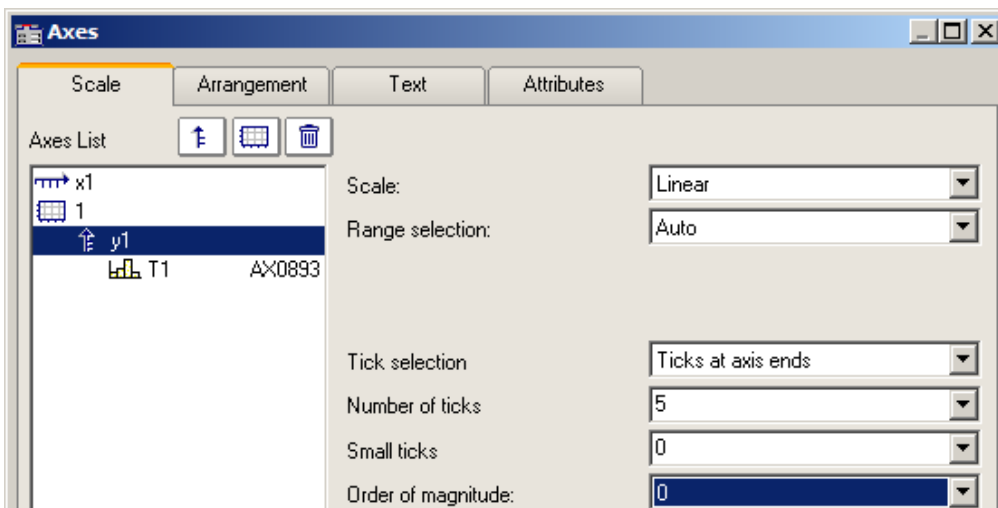
The main ticks, along with which the axes labeling appears in the curve window, can be placed anywhere on the axis. If you select the option *Ticks at fixed distances* in the *Tick selection* control, the additional control *Tick distance* becomes available. Here you have the ability to set the amount of ticks per unit. For instance, when the unit is ms, the ticks appear every 3ms, if 3 was entered in the text box *Ticks distance*.



Select the option *Automatic Labels*, if the ticks should be placed at the ends of the axis. The increment between ticks then depends on the total amount of ticks set.

Automatic tick spacing is recommended; for this, simply select *Auto* under *Tick selection*.

If the option *Automatic Labels* was selected in the *Tick selection* control, the number of ticks can be entered in a text box. The number must be greater than or equal to 2.



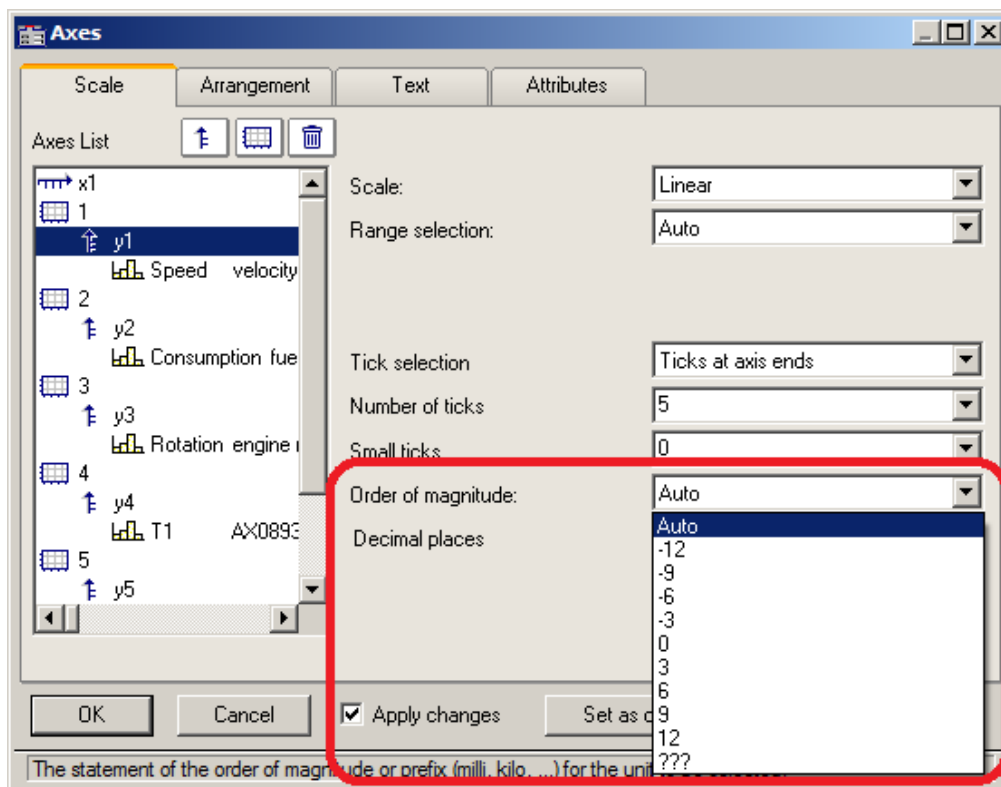
The number of ticks per unit is calculated by subtracting one (1) from the number of markings and dividing the difference by the displayed interval in units. For example, with an interval of 3ms and seven markings specified, two ticks per unit (ms) will be placed on the axis.

Remarks

- An error message is generated when clicking on *OK* if inapplicable values were entered in the text boxes. Error messages occur on account of inapplicable values (too large) or ranges (no positive values in logarithmic display, minimum not smaller than maximum, or inapplicable number of markings). Correct the relevant text boxes and then click on *OK* again.
- The distance (difference) between x_{min} and x_{max} may not be too small in relation to the maximum magnitude of x_{min} and x_{max} . For example, a range of 1.0000000000000001 1.0000000000000002 **CANNOT** be displayed. The maximum permissible factor between difference and maximum magnitude is 1E-13.
- When the display mode *1/3-octave/octave* labeling is selected, then the scaling of the x-axis is accomplished as detailed in the appropriate chapter. A different dialog appears for scaling the x-axis.

Order of magnitude

It is possible to arbitrarily determine the power-of-ten of the axis-scaling. If the axis is labeled with a unit, and appropriate prefix will be appended to the unit symbol, if applicable (e.g., mV or MV).

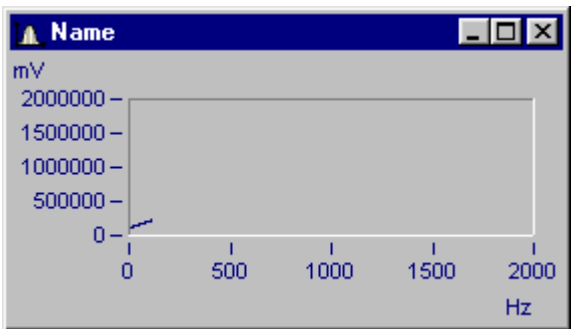


 Example

x-axis with a fixed

For an x-axis with a fixed scaling range of 0...2000V:

Order of magnitude	Display
Auto:	0..2 kV
+6	0..0.002 MV
0	0..2000 V
-	0..2000000 mV

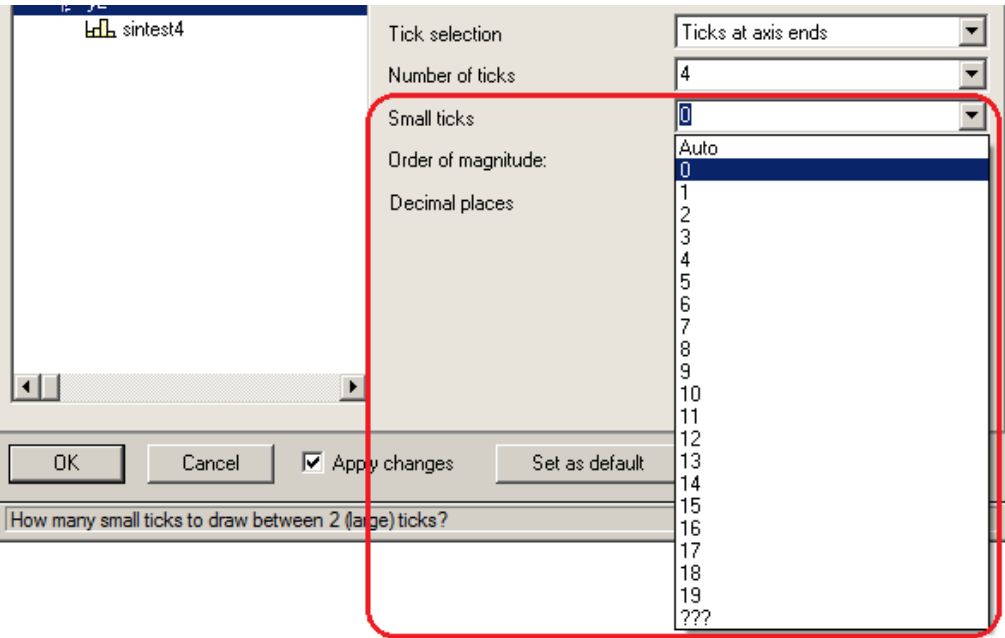


Curve window with inappropriate scaling

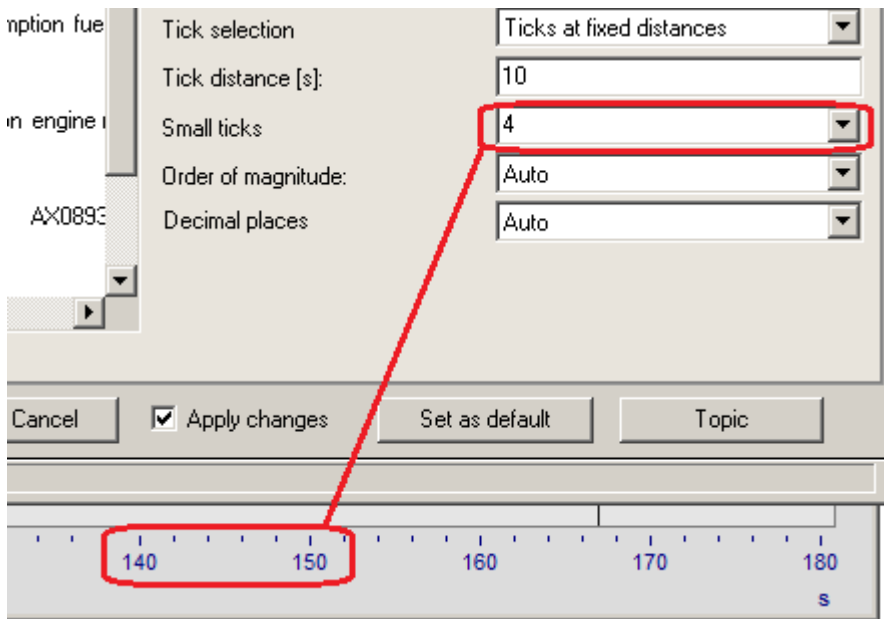
x-axis scaled with exponent 0, y-axis with exponent -3.

Small ticks

Small ticks can be positioned along the axis between the big ticks (main ticks):



To do this, select the amount desired from the pop-down list. If you don't want any, select 0.



The small ticks are not labeled. If the curve window featured a grid, then supplementary grid lines (which may appear thin in a printout) will appear at the small tick coordinates. For details, see the discussion of the menu item [Configuration/ Grid](#) ⁷⁹⁰.

Decimal places

With linear axes, the number of decimal digits can be set here.

Format (for Date/time absolute)

If the scaling of the x-axis is in abs./rel time, the format of the labeling can be specified:

- *Auto*, *Auto 1 line*, *Auto 2 lines*
- *fix 1 line* or *fix 2 lines*.

The display of time and date makes use of placeholders.

Placeholders in absolute time:

Time: h, hh for hours; m, mm for minutes; s through ss.ssssss for seconds

Date: D, DD for day, M, MM for month; YY, YYYY for year

Names: DDD for abbreviated weekday, DDDD weekday, MMM abbreviated month, MMMM month

A.M., a.m., AM, am for AM/PM format

The placeholders and special characters appear in angle brackets. Furthermore, other characters can be added.

Duplications << or >> for one < or > character in the output



Example

<hh:mm:ss.ss>

<hh:mm a.m.>

<DD.MM.YYYY, hh:mm>

<DDD, DD.MMM.YYYY>

date=<DD>.<MM>.<YY>

Placeholders with relative time:

h, hh for hours; m, mm for minutes; s to ss.ssssss for seconds

D to DDDDDD for days; o to oooooo for hours without days

The placeholders and special characters are expressed in angle brackets. Furthermore, other characters can be added.

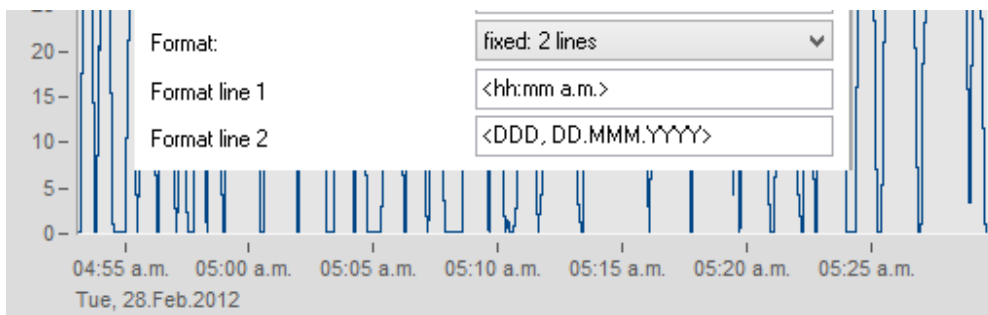
**Example**

<hh:mm:ss.ss>

<D> Tage

<o:mm:ss>

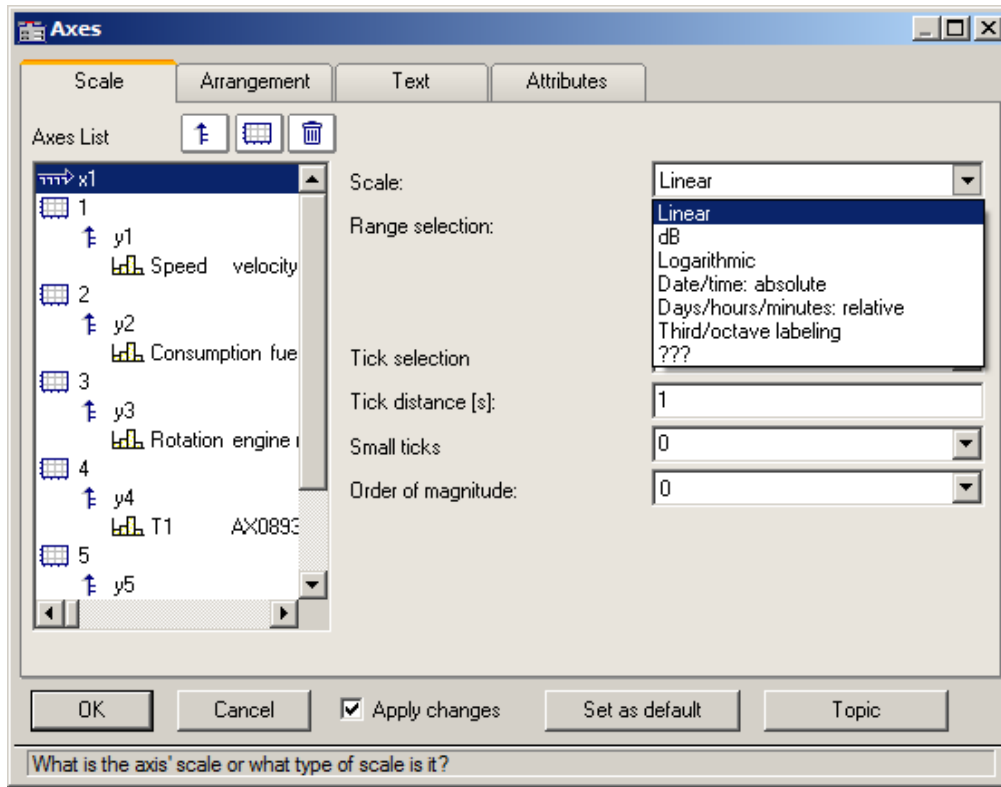
<o> Std, <mm:ss>



Example: line 1: <hh:mm a.m.> line 2: <DDD, DD.MMM.YYYY>

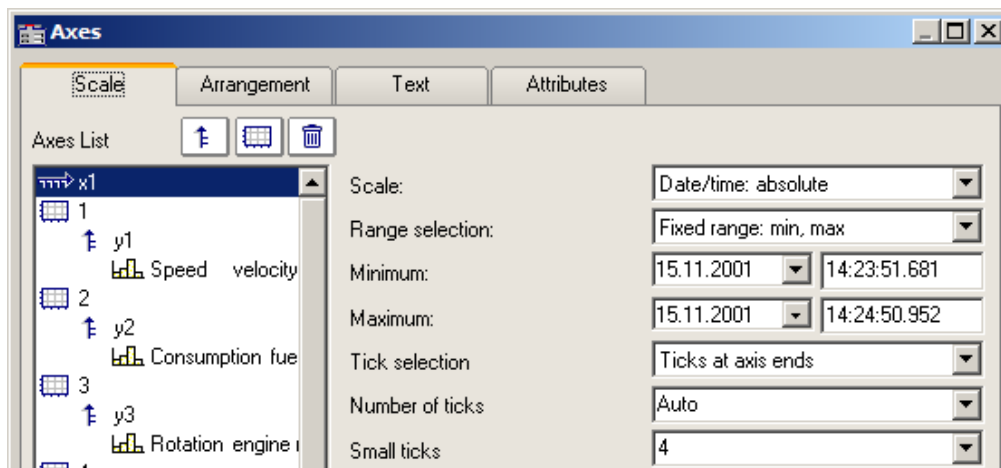
11.6.3.1.1 Particularities of the x-axes

The following concerns special display types only available for x-axes. The x-axis alone must be selected in order for these options to be enabled.



Scale

Absolute date, time: The minimum and maximum are each specified separately in terms of the time and date. Selection of the date is accomplished by means of a calendar. The time is entered into a text box in compact form. The seconds in the time can have decimal places.

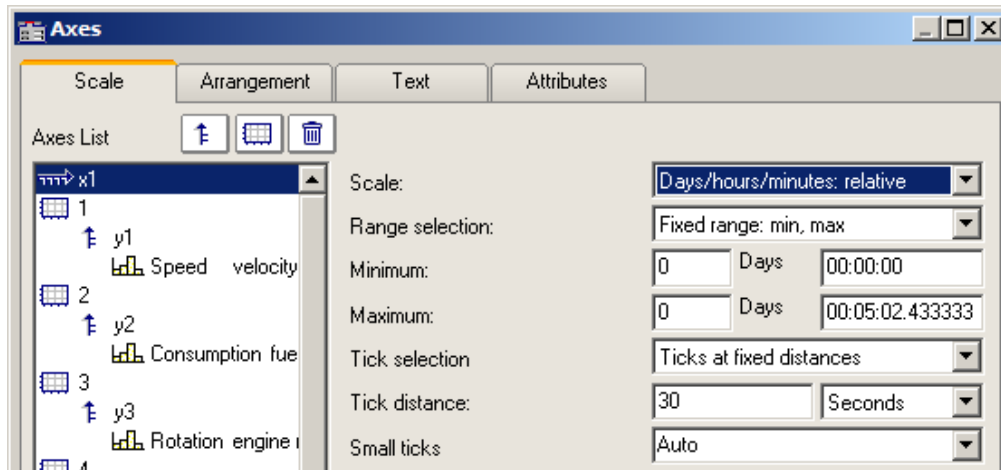


The distance between ticks can be set to be determined automatically or to be fixed. With a fixed setting, the unit can be anything from seconds to days.

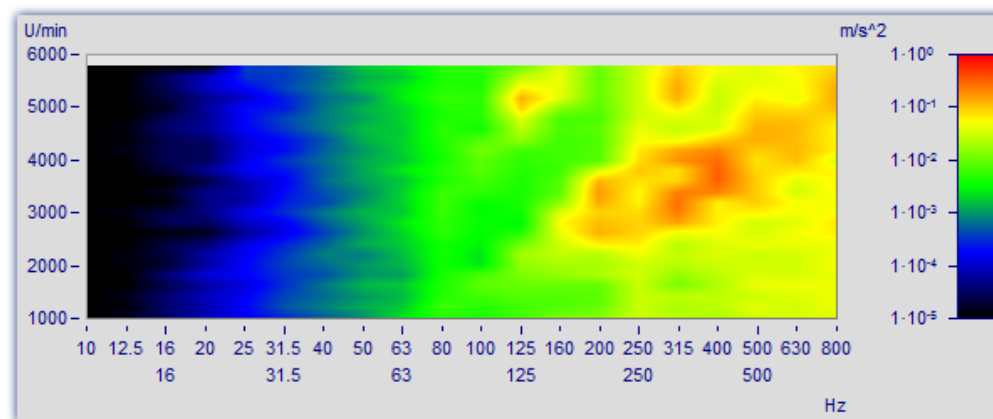
The channels are displayed in reference to absolute time. The absolute time of the measured points is usually derived from the sum of the specified absolute trigger time and the relative time for a point measured after the measurement's start.

Days, hours, minutes: relative: The amount of day and of hours, minutes and seconds (the latter may have decimal places) can be specified. As in linear display mode, the display of the measured data doesn't refer to the absolute triggering time. Thus, only the time intervals between the measured points and the trigger time are indicated.

It is also possible to specify a negative amount of days, in order to indicate points in time preceding the trigger.



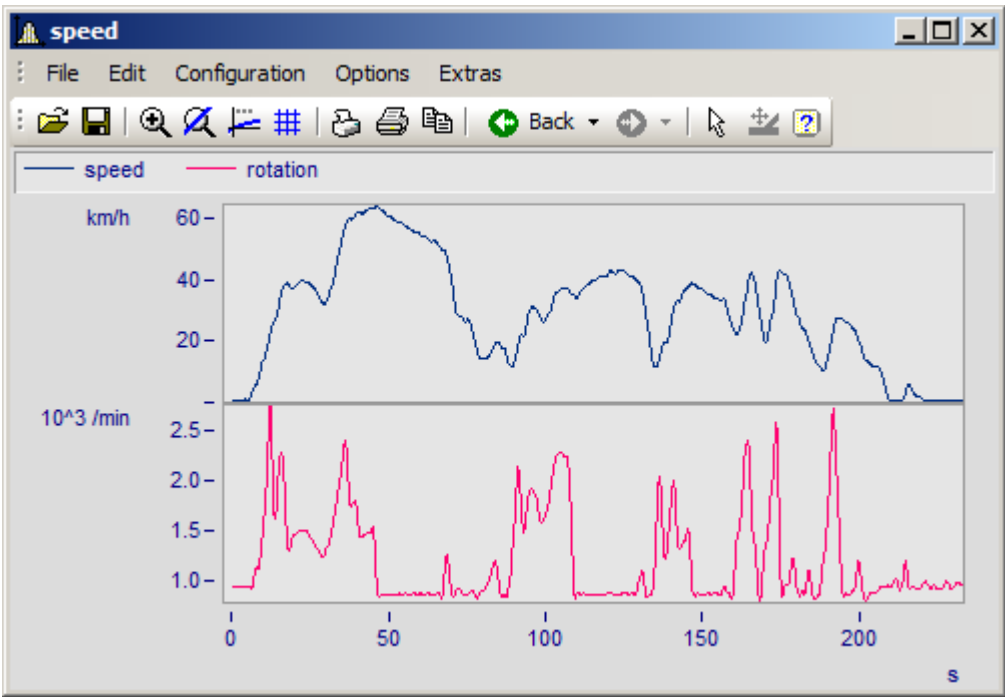
Third/ octave labeling: If a waveform contains a 1/3-octave or octave spectrum and the waveform's x-axis is scaled in 1/3-octaves, then the x-axis can be drawn with numerical values of the 1/3-octaves and octaves, in accordance with DIN.



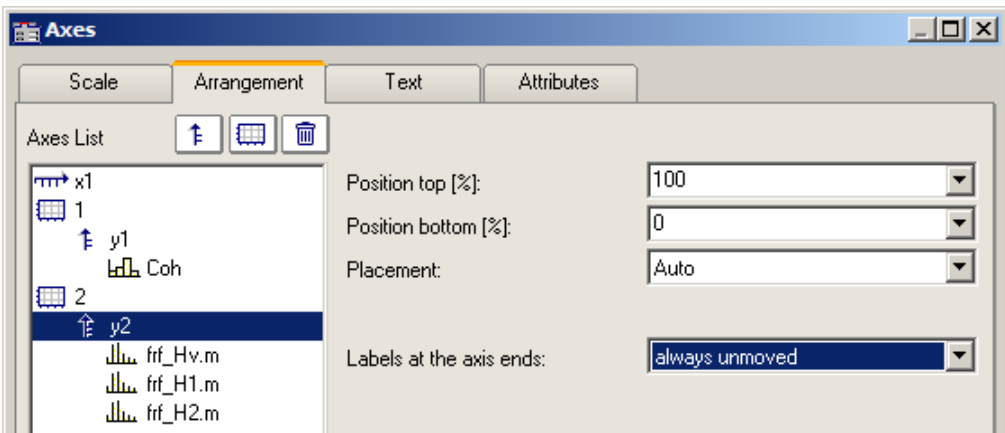
For details, see the dedicated section on "[Third / octave labeling](#)".

11.6.3.2 Arrangement

It is possible to stack multiple y-axes in a single coordinate system. It is also possible to set whether to layout the axes a the left or right of the coordinate system.



To do this, select the page *Arrangement* in the dialog *Axes*.



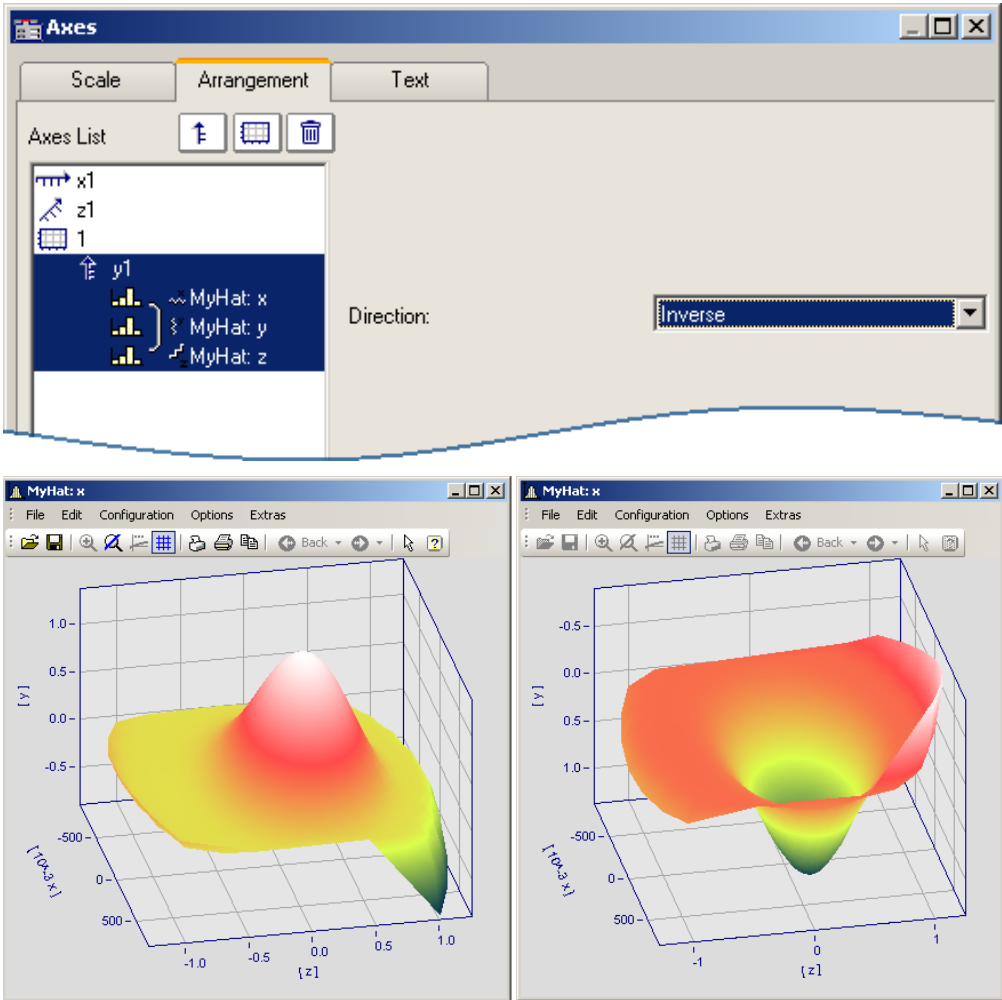
Settings	Description
Position	The relative position along the full height of the coordinate system can be specified for each axis. The setting 100% represents the position all the way at the top, 0% the very bottom. If the axes overlap each other, new "columns" are automatically set up.
Placement	The control <i>placement</i> sets whether to position the axis at the left or right of the coordinate system. Placement: Labels at the axis ends: Left Auto Left Right ???

Settings	Description
Labels at the axis ends:	Alongside automatic orientation of the labeling, this control also offers <i>always unmoved</i> and <i>move if necessary</i>. Labels at the axis ends: Auto Axis width [mm]: 20.0
Axis width	The width of the Y- and X-axis can be specified manually. In particular when using "User Ticks", this helps in providing sufficient room for the labeling.
Resolution	Setting the scaling of the y-axis <i>auto</i>: individually <i>same resolution as x-axis</i>: Generates a constant aspect ratio. In consequence, the ensures among other things that for an RGB-image the image is displayed without distortion. Resolution same resolution as x-axis This setting only works when the same setting (linear or logarithmic) applies to both the x-axis and y-axis, and only for either standard display or Y-axes stacked. Not available for abs./rel time, and 1/3-octaves.

Settings	Description

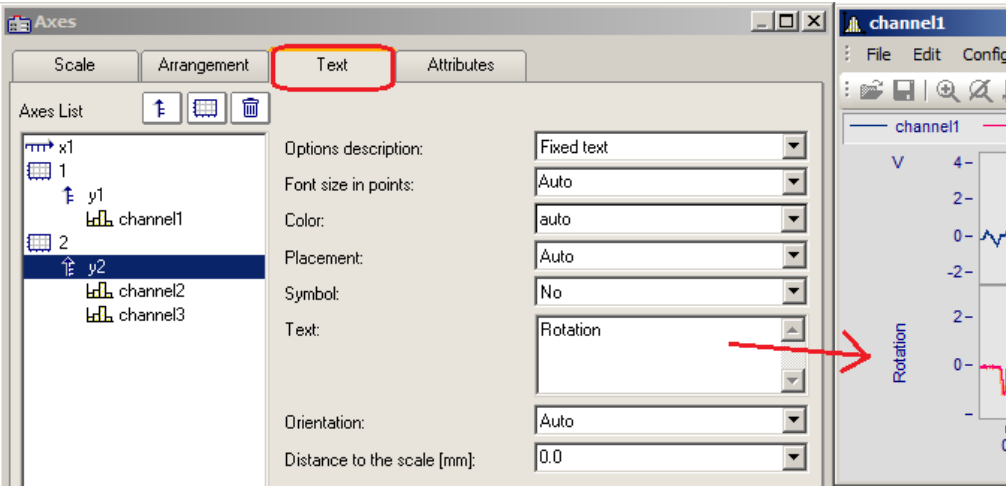
Inverting the axis direction With 3D display, you can invert the direction of the Y-axis.

Use the menu: *Configuration\Axes* and select the page *Arrangement*. In the dropdown list you will find the entry *Inverse*.



Example: Invert the axis direction
At left the normal orientation, at right the inverted Y-axis

11.6.3.3 Text



Along with the normal axis labeling in the respective units, there is also the possibility to use the page "Text" to personally define the axis labels and to set their parameters. If there is text for a y-axis, it will be printed in vertical alignment in the curve window. Text for the x-axis is aligned horizontally and placed in the center. The following settings are available for the text:

Settings - Options description	Description
No text	The default setting, in which the unit associated with the waveform is displayed horizontally at the upper end of the axis. For "No text", the additional option " <i>Unit indication</i> " is shown. This can also be hidden, for instance when using " User Ticks ".
Fixed text	Here you can enter any desired text, such as "Length [m]", as a permanent feature. See also the notes on Greek letters .
Unit	The unit saved with the waveform is displayed.
[Unit]	The unit saved with the waveform is displayed in angular brackets.
Name, Unit	The channel name is displayed in the first line of the coordinate system and the associated unit saved with the waveform are displayed.
Definable with placeholders and formatting instructions	The text can be given along with fixed components and placeholders. Available placeholders are: • <name> for the channel name • <unit> for the unit • <comment> for a comment on the channel • <e>exponent</e> for an exponent, as an alternative to a^b. • <s>index</s> formats the text as small index. Nesting of the index or exponent is no allowed. • <g*b> for Greek letters. Also in the exponent: A<e>-<g*a>t</e> => A^{αt}. • x<s>i</s><e>e</e> for exponent together with index.

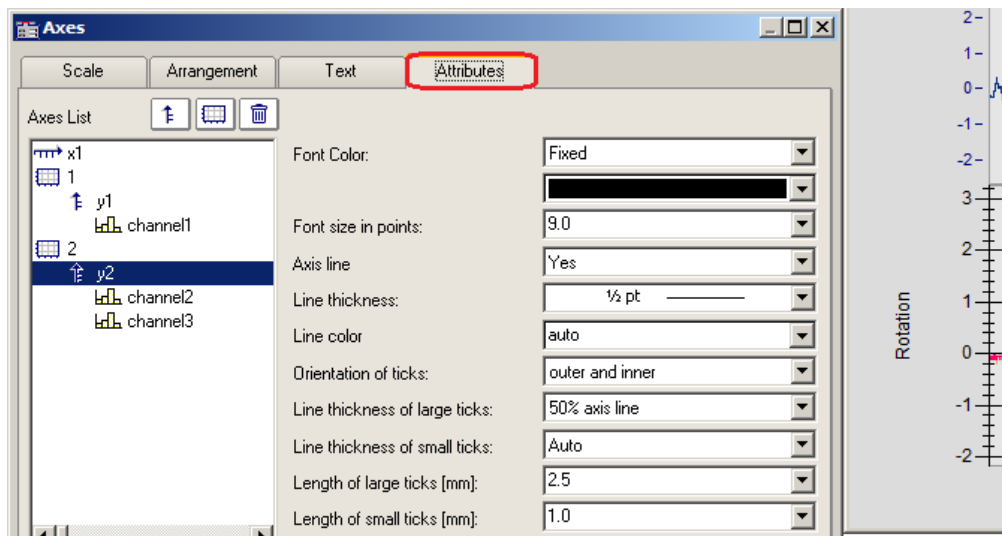
Additionally, it is possible to set the **font size**, font **color**, **placement** along the axis, and whether the **symbol** for the first line appears in front of the signal name, as it does in the legend. The symbol can only be set in the dialog for y-axes. The **Orientation** determines whether the text is displayed across the axis or parallel to it. Further, using **Distance to the scale**, it is possible to specify a minimum distance to the text.

Reference

Further texts

Further texts can be added on the tap [Display/Text](#) and in chapter [User ticks](#).

11.6.3.4 Attributes



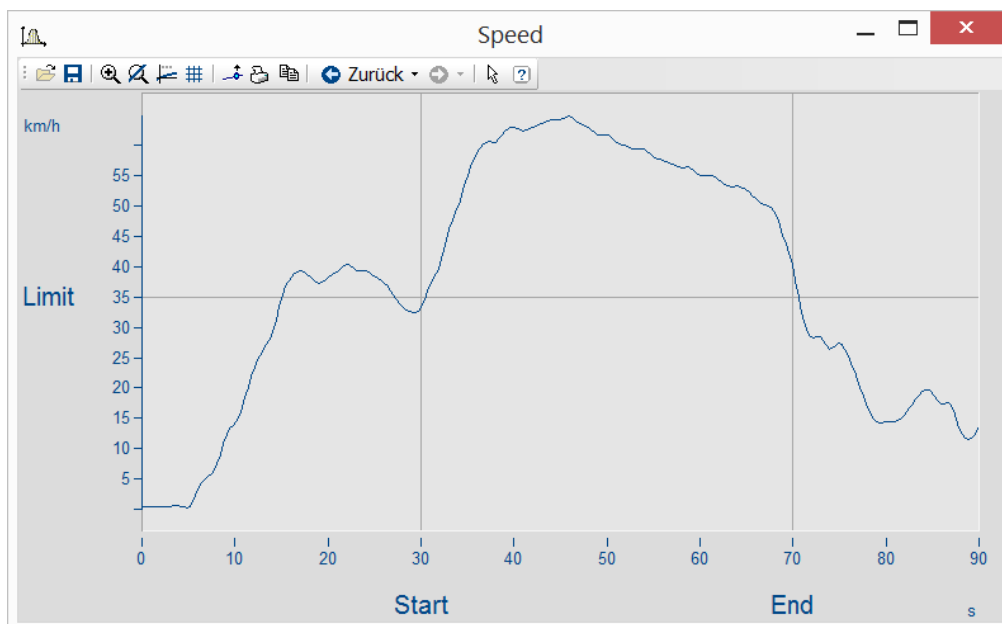
Settings for the axis

The control **Font Color** allows the color of the first data set to be applied, or a freely selected color to be set.

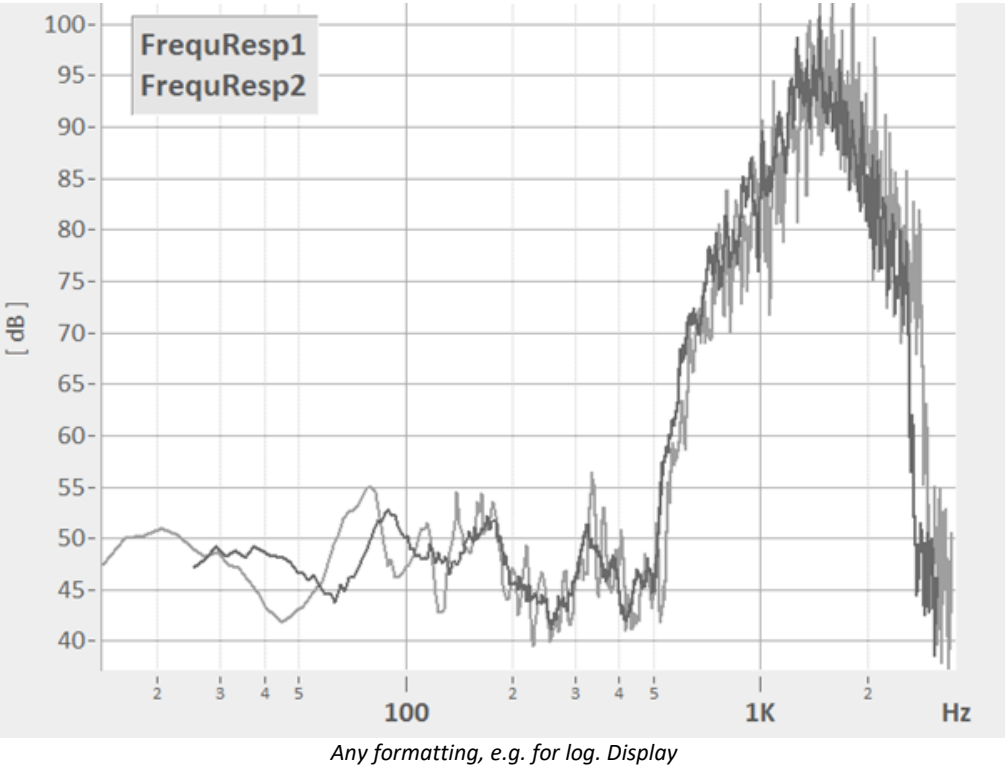
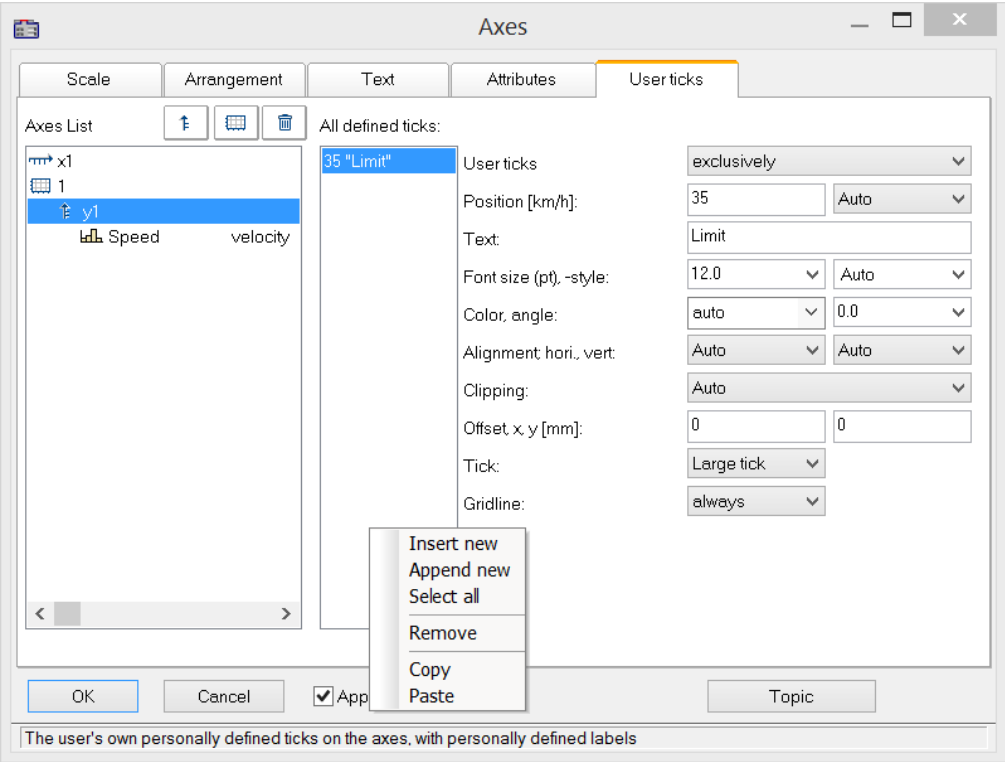
Additionally, it is possible to add an **Axis line** while specifying its **Line thickness** and individual **Line color**.

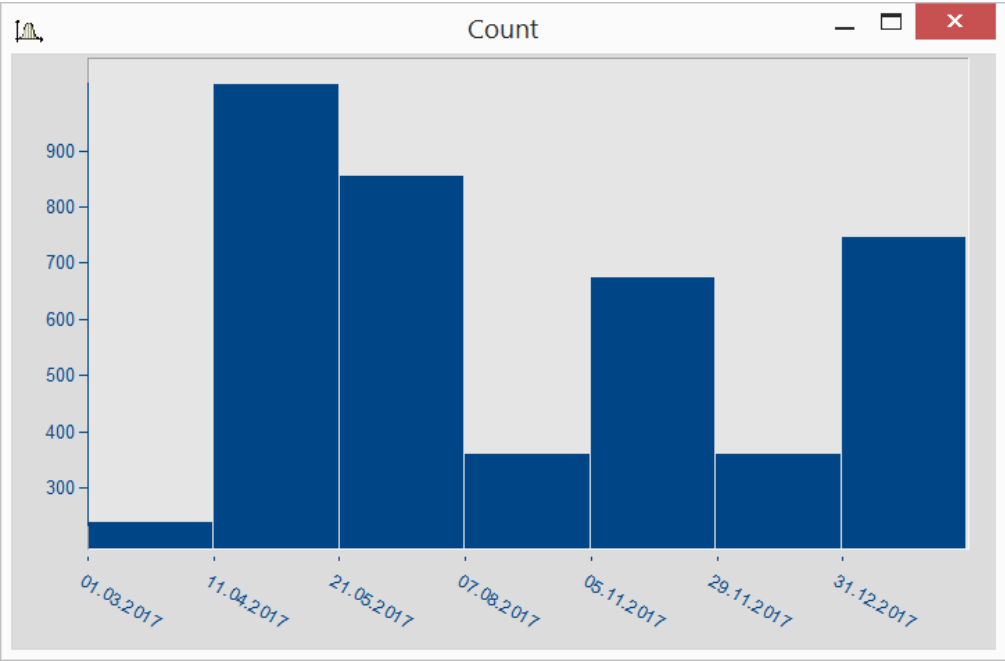
The **Line thickness** and **length** of the **ticks** can be set separately for both the small ticks (between numbers) and large ticks.

11.6.3.5 User ticks



On the page "User Ticks", there are controls for inserting extra ticks. Using the context menu for "All defined ticks", you can insert, copy and delete ticks.





Date display independent of x-delta

The properties of the respective ticks assigned are listed according to the axis selected.

Settings	Description
User ticks	Replaces(exclusively) or complements(Additionally) the existing scaling.
Position	Determines the position on the axis.
Font -size, -color, -alignment etc.	determine the appearance of the text. The necessary width can be set on the page " Arrangement " ^[747] .
Restriction	With <i>Restriction</i> active, the text is no longer displayed if it is moved out of the visible area by scrolling the axis. Without <i>Restriction</i> active, the text is still displayed.
Shift	Moves the user ticks to the left (<0) or to the right, for instance in order to avoid overlapping with the scaling in cases of additional ticks.
Tick	Selection of the tick representation, according to the settings on the tab Attributes ^[750] .
Gridline	Shows a line corresponding to the tick. There is a choice of whether or not to make its appearance depend on the presence of a grid.
Exclusively	Custom settings for the selected tick: <i>Yes</i> = only the user tick; <i>No</i> = in addition to the scaling
From "Label for raw data"	Only visible if <i>Insert from "Label for raw data"</i> was activated with the context menu. User-defined properties of the type imc30 are imported automatically.

 Reference

Further texts

Free texts can be set at [Display/Text](#)^[788] and on the tab [Axes/Text](#)^[748].

11.6.3.5.1 Label for raw data

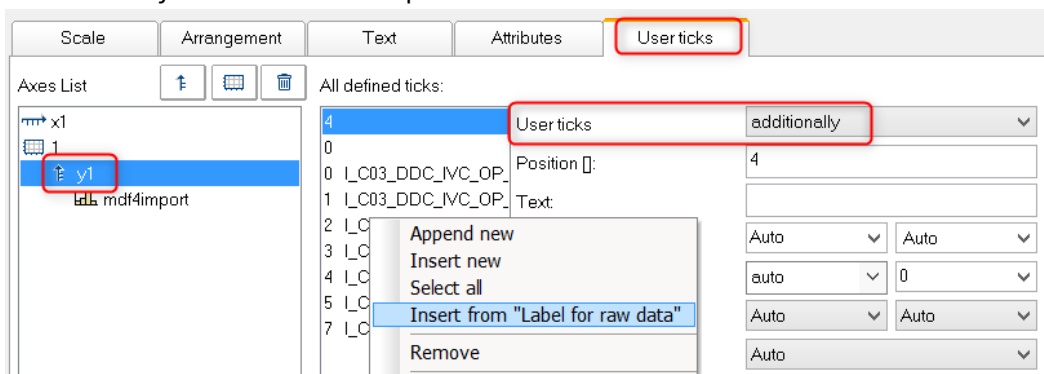
Some signals contain integers representing certain states or error codes. For instance, such a number can denote the gear in an automatic transmission (0=N, 1=D, 2=R, 3=P, etc.). Such information is typically recorded as an integer value.

imc's data format makes it possible to save such supplemental information along with the variable. One category of these "*user-defined properties*" is called *imc30* and lists texts which are provided by certain sources (MDF, CAN, etc.) as "Label for raw data[]".

When a data set contains such imc30-information, the values can be used as User Ticks.

Activating "Label for raw data"

- To import the data, open the dialog *Axes* -> Tab: *User ticks*.
- Select *additionally* or *exclusively* under *User ticks*.
- Select the affected Y-axis and right-click the mouse over the middle list, *All defined ticks*. Only when the signals has properties of the type *imc30*, the entry "*Insert from Label for raw data*" is visible.
- All "*Labels for raw data*" are imported.



Editing the presets for user ticks

As soon as *Insert from "Label for raw Data"* has been activated, the property "*Label for raw data*" is available with the following options:

no: as previously

yes: this tick is derived from a *Label for raw data[integer]*.

Template: Template, if new properties are associated with the channel.

FAMOS generates a tick without a text, at the position 0. This is used as a template. When new positions are added, new ticks are generated in accordance with these settings.

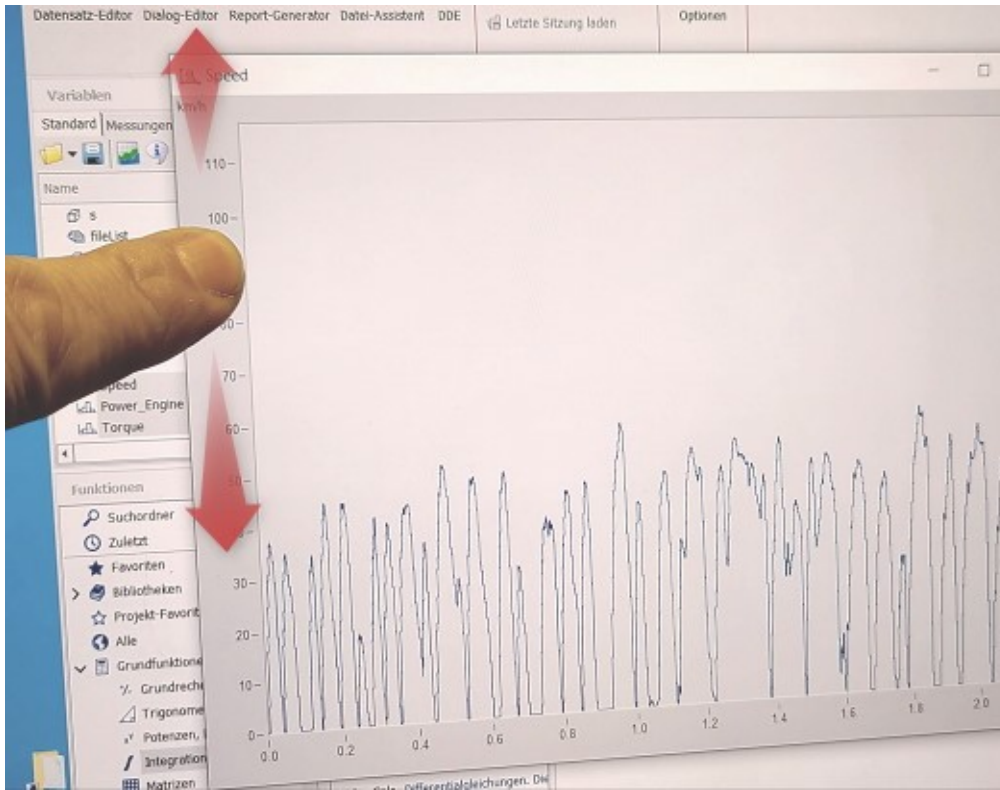
Remarks:

- When positions are discarded, the associated ticks are deleted.
- When you do not desire automatic updates, set the property *From "Label for raw data"* to "*no*".
- *Label for raw data[integer]*: The value in brackets is a whole number in the channel's raw data, which means unscaled (whole numbers). If there are real numbers, only whole numbers are given consideration.
- *Label for raw data[integer]* always only pertain to a channel's y-values and is only suitable for the y-axis. XY-display is the exception, in which the x-axis represents the y-values of an XY superposition.

11.6.3.6 Touchscreen operation

Enhanced touchscreen operation for the curve window

For the purpose of operating the curve window by touchscreen, certain regions of the curve window are designated for performing certain actions. Thus for instance, in the upper or lower region of the curve window, you can slide the curve in the y-direction, and in the left or right margins you can slide the curve in the x-direction.



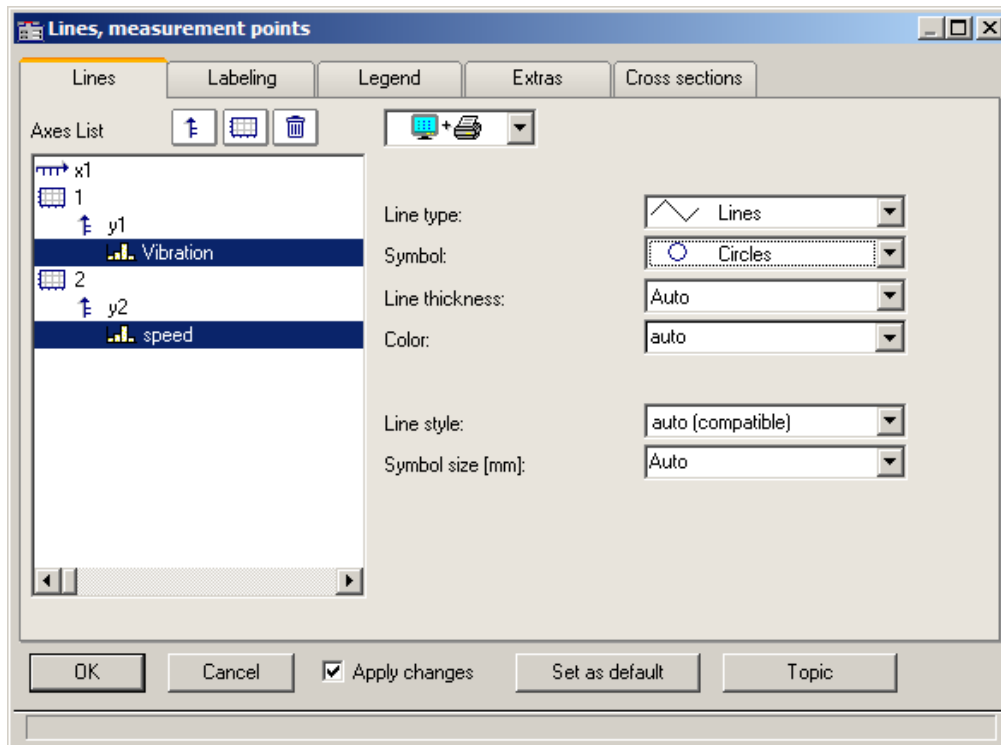
11.6.4 Line configuration

Function

Here, the type of line used to draw the curve and the symbols used to identify the sample values can be specified.

To make the settings for the lines, call the menu item *Configuration / Lines...*. Alternatively, you can switch directly from the dialog *Axes...* to the dialog *Lines* by using the button *Topic*.

The dialog for setting line properties then appears:



11.6.4.1 Lines

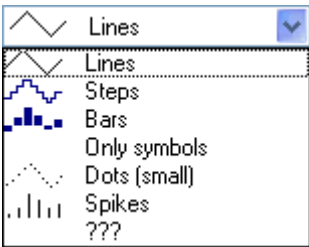
Appearance of lines in the screen display and in printouts

Effect on the printer and/or monitor screen



By making this selection, you determine whether the properties you set apply to the screen view and/or to the printer (or Clipboard). By this means, it is of course possible to make separate settings for the printer and for the screen. Not all properties can be set separately for the printer and the screen each.

Line type



Display of the samples. Normally, the samples are displayed by linearly interpolated solid *lines*. Alternatively you can select other display options from the list appearing further below.

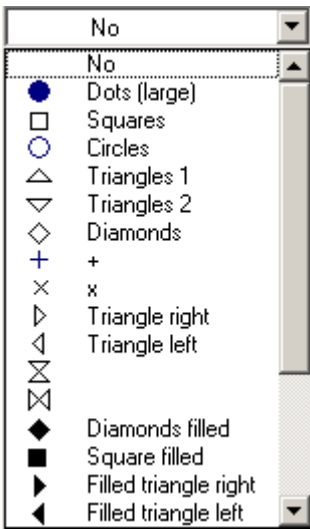
When displaying an [RGB-image](#)⁷²⁴, there is a distinction between *Lines* and *Steps*. For *Lines*, the pixels are interpolated between the samples (original image points) with color gradients. With *Steps*, the interpolated pixels remain constant. With any RGB-image, all line types other than *Lines* are treated as *Steps*. If multiple samples fall on one and the same pixel on the screen, the system takes an average.

The line type is always the same for the printer and the screen.

Line type	Description
Lines	The " <i>Lines</i> " style displays the curve as a polygon, i.e. as straight lines connecting the points of the waveform. In an RGB-image, the pixels are linearly interpolated between the original image points.
Dots	When the option " <i>Dots</i> " is selected, only the points of the waveform are displayed as dots of one pixel each.
Bars	The " <i>Bars</i> " display shows every point of the waveform as a bar rising from the horizontal axis.
Steps	When " <i>Steps</i> " is selected, all points of the waveform are connected with stair-steps, i.e. each sample value of the waveform is held until the next sample ('sample and hold' effect). In an RGB-image, the pixels are interpolated by a constant value between the original image points.
Only symbols	With the setting " <i>Only symbols</i> ", there is no continuous line. At each measured point, a symbol (e.g. a square) is drawn. The particular symbol desired must then be selected under the <i>Symbol</i> control.
Spikes	" <i>Spikes</i> " are vertical lines extending from the x-axis to the height of the measured value.

Symbol

Every measured point is indicated with a symbol.



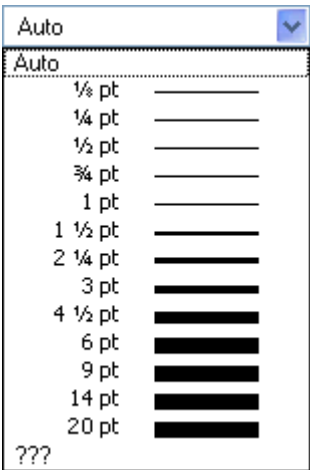
The symbol type desired is selected from this pop-down list. Symbols can be displayed either together with a line or all alone.

In special cases, not every measured point is represented by its own symbol, but rather the symbols are distributed evenly along the plot, for example, in order to differentiate between the plot lines of different channels. See the section on the menu item [Configuration/ Display](#)⁶⁴⁹, in connection with the property *Number of symbols*.

The symbol always appears the same on screen or printed out.

A fixed number of symbols can be specified on the "[Extras](#)"⁷⁶⁵ tab.

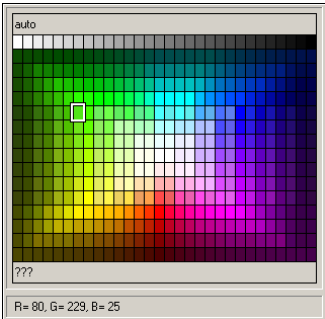
Line thickness



The thickness of lines can also be selected. It pertains not only to continuous plot lines, but also to other line types and certain symbols.

The line thickness can be different for the printout and for the screen display.

Color

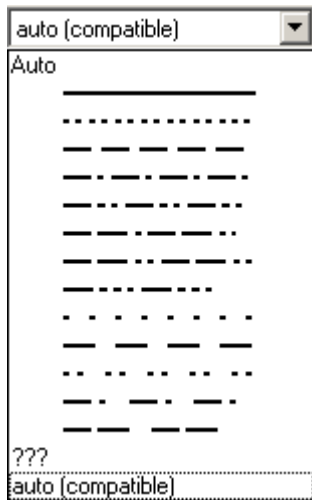


Here, the color of the line or symbol can be specified. If the color is set to *auto*, it is selected automatically by the system. In that case, a line's color depends on its order of appearance among the lines, and on the corresponding global settings for color assignments in the curve window; see the discussion of the menu item "[Options / Colors](#)"⁸⁸².

It is also possible to select a fixed color from the palette provided, which cancels all automatic settings.

The color can be different for the printout and for the on-screen view.

Line structure



This determines the line's structure. If the structure is set to *auto*, then different line structures are assigned in succession, i.e. the first line is solid, the second dotted, the third dashed etc. For *auto (compatible)*, the global setting *Curves in structure* is observed. This mode is compatible with imc FAMOS 5.0, where the line structure was set globally in the *Colors*-dialog.

Symbol size [mm]

If symbols were selected for display of the measured points, the diameter in mm can be defined here. For *auto*, the global setting *Symbol diameter* is observed.

Fill area

Fills the area below a line.

Fill area:

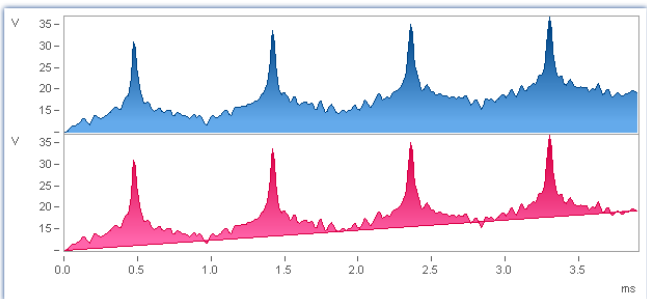
to y=zero

Area color gradient:

From top to bottom

Color gradient to :

auto



Fill area:

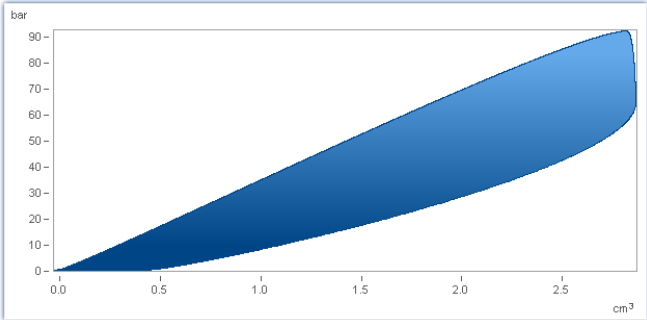
inward

Area color gradient:

From bottom to top

Color gradient to :

auto



For XY data sets, the hysteresis is closed if necessary so that a closed area can be filled.

Fill area:

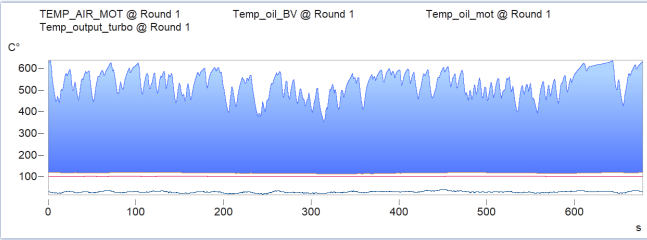
until previous line

Area color gradient:

From bottom to top

Color gradient to :

auto



For "Fill area: until previous line", the area between two consecutive lines in the curve window is filled.

Fill area

to y=zero

No

Area co

to y=zero

Color gr

to bottom

inward

???

no: as previously

to y=zero: Fills area below the graph up to the zero line.

to bottom: Fills area to the bottom edge of coordinate system.

inward: encircled area. The last sample of the data set displayed is connected to the first sample. An closed line results, and the area it encloses is filled.

Area color gradient

From top to bottom

No

Color gradient to :

From top to bottom

From bottom to top

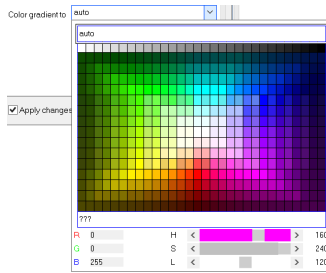
From left to right

From right to left

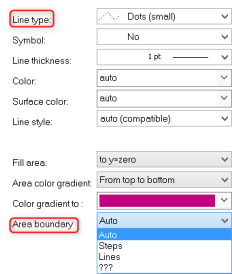
???

If **Fill area** is not **no**:

Area color gradient: Specifies the color gradient for the filled area. without color gradients, only the one color is applied evenly. Otherwise, the color transitions from one line color to another. The second color is specified under **Color gradient to**.



Color gradient to: *auto* (like the line itself) or permanently selected. For ***auto*** and color gradient, a lighter variant of the line color is drawn.



If **Line type = *Dots*** or **only symbols**, there is the additional option

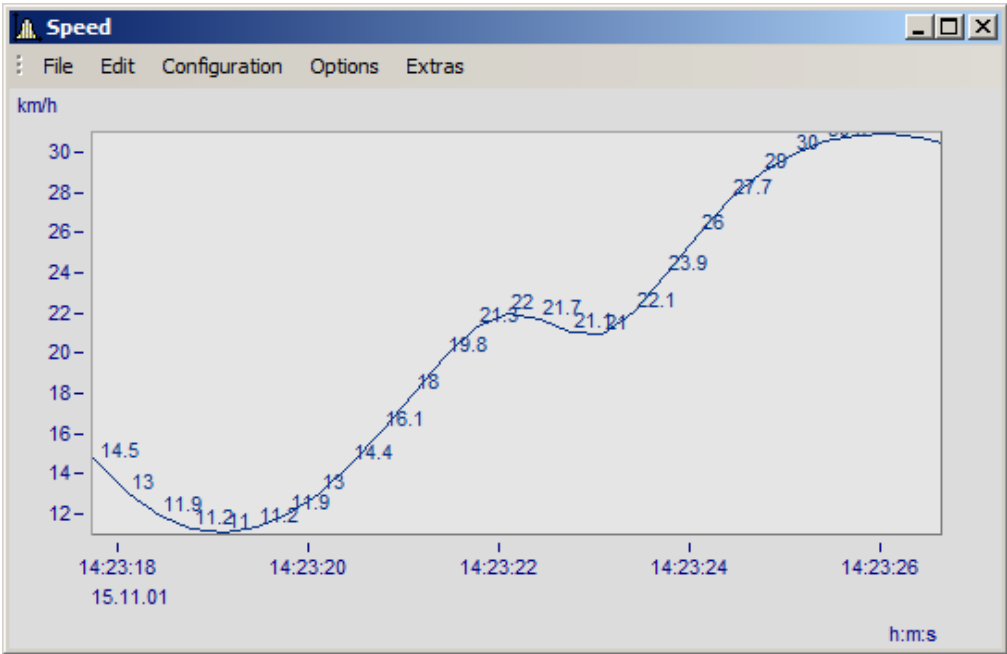
Area boundary: This determines whether the edge of the area is interpolated linearly or appears in discrete steps.

If multiple curves are displayed with overlap, the order for each curve is:

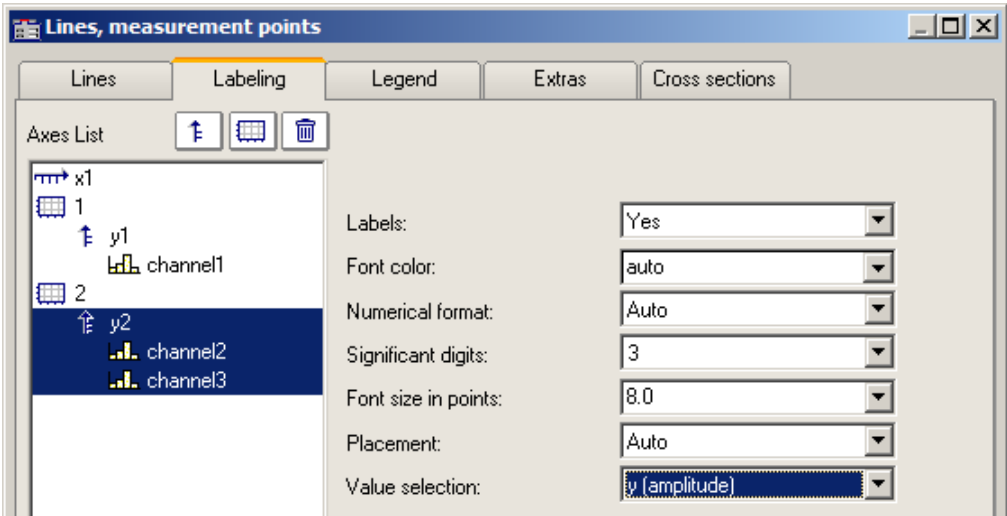
1. first the area
2. then measurement uncertainty
3. then lines

It is possible to generate a **pure** area without any borderline by setting **Line type = *Only symbols*** and **Symbol = *empty***.

11.6.4.2 Labeling

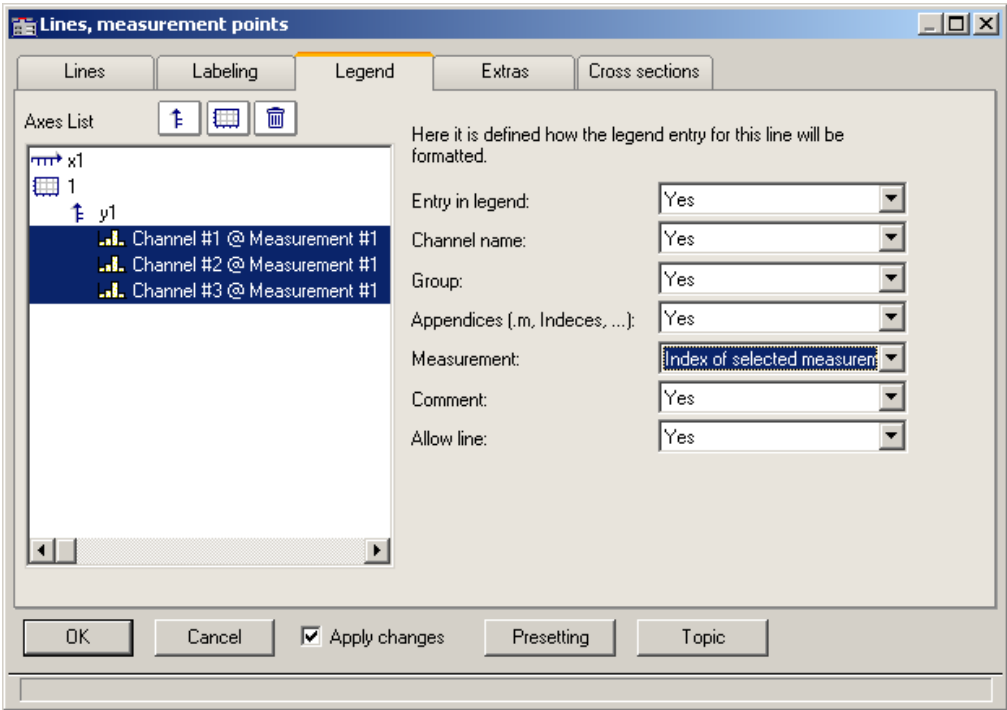


Here you can set whether or not the measurement points are labeled with text stating their numerical values. For the **labeling** of the numerical values, the **color**, **format**, **number of valid digits**, **font size** and **position** can be set.



Using **Value selection**, you can set which values are displayed: y, x, parameter, magnitude and phase

11.6.4.3 Legends



Options for displaying the legend

By selecting this tab, you can select various options for displaying the legend pertaining to the line selected.

Note

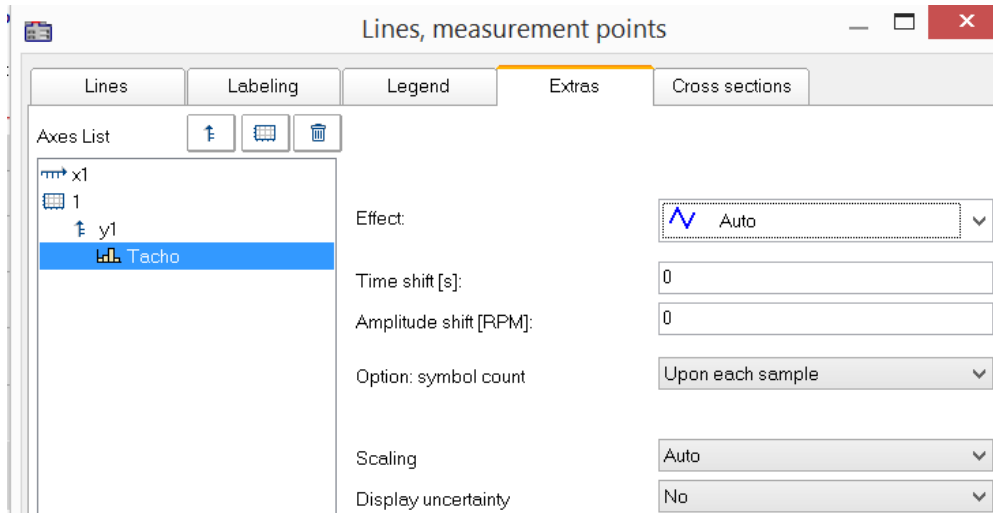
If you display only one data set in the curve window, the legend is only displayed if you have set the control for [Display\Legends\Display Legends](#)⁷⁶⁸ to *always*.

Settings	Description
Entry in legend	The option <i>No</i> means that no legend is displayed at all. Selecting <i>Yes</i> means all options are shown.
Channel name	Determines whether the channel name is displayed.
Group	Specifies the group for grouped data sets, e.g. Measurement1:Channel_01
Appendices (m, Indices, ..)	Indication of component or event number, e.g. Spectrum.b
Measurement	Here, the option <i>Number of the selected measurement</i> is especially interesting, which enables display of the measurement number instead of the measurement name.
Comment	This is the channel comment as described in the imc FAMOS manual, in the chapter Properties/ Charact. Dialog (Waveforms). For channels recorded using imc STUDIO, the comment can be entered as a channel property.
Allow line	This determines whether an example of the line's appearance is displayed to the left of the label.

Additional general legend settings are described here: [Menu Configuration / Legends](#)⁷⁶⁸.

11.6.4.4 Extras

On the page *Lines\Extras*, special display options are available:

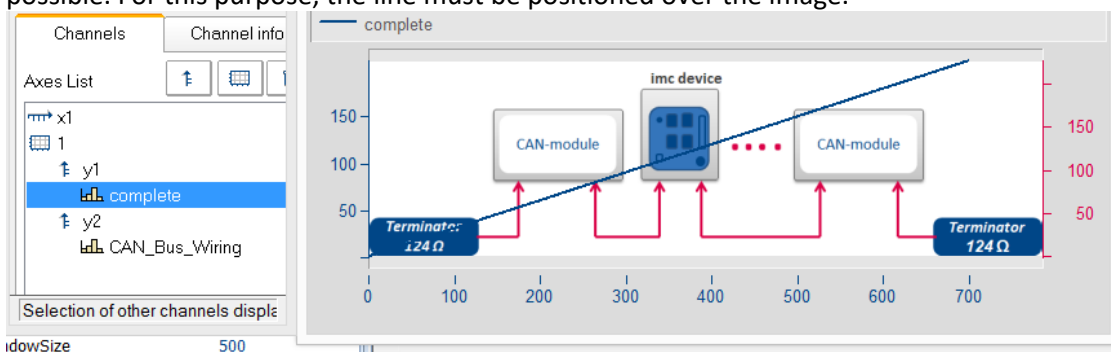


- [Time-/Amplitude-Shift](#)⁷⁶⁴: Shifts the line in the curve window without changing the values of the variable.
- [RGB-image](#)⁷⁶³: Display of an image made of segmented data.
- Special [color map display](#)⁷⁶⁴
- Reduction of the [symbol display](#)⁷⁶⁵
- [Measurement uncertainty](#)⁷⁶⁶

Effect: Image from RGB-values

When the color-flag is set, for RGB variables, the parameter **Effect** in the *Extras* dialog is automatically set to "Image from RGB-values". Otherwise, it can be set manually here.

The property applies to each line separately, which means that superpositioning of an image with a curve is possible. For this purpose, the line must be positioned over the image.

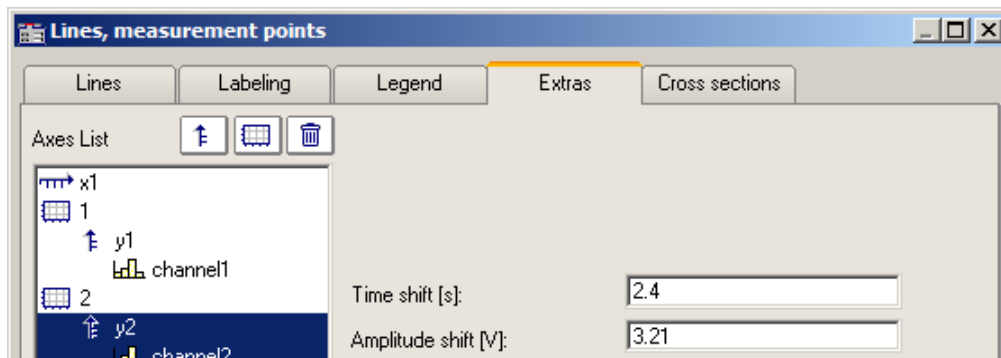


References

General info on RGB-images is presented in the chapter [Display variants of the curve window](#)⁷²⁴

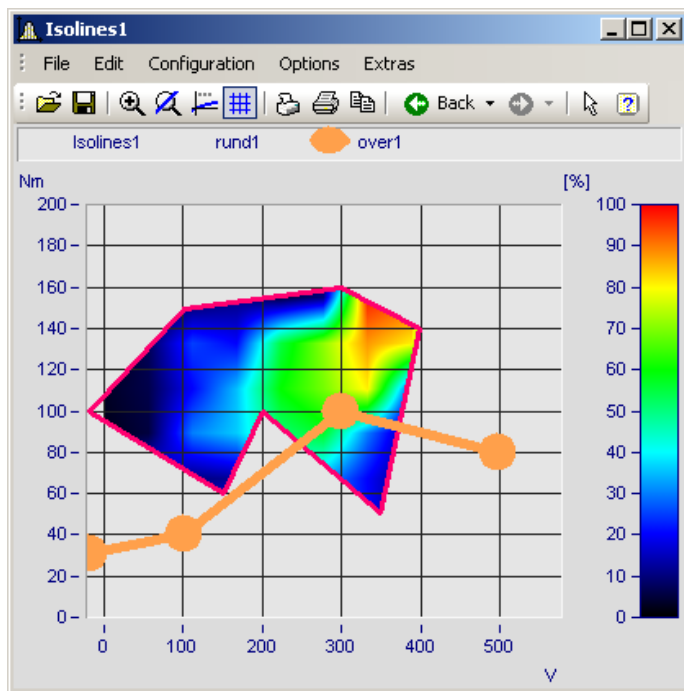
Info on aspect ratio and resolution is in the chapter [Axes configuration/Arrangement](#)⁷⁴⁷

In display mode *Default*, you can manually specify the parameters of the Line-Shift function here. For more info, see the section [Line-Shift](#) ⁸⁶⁵.



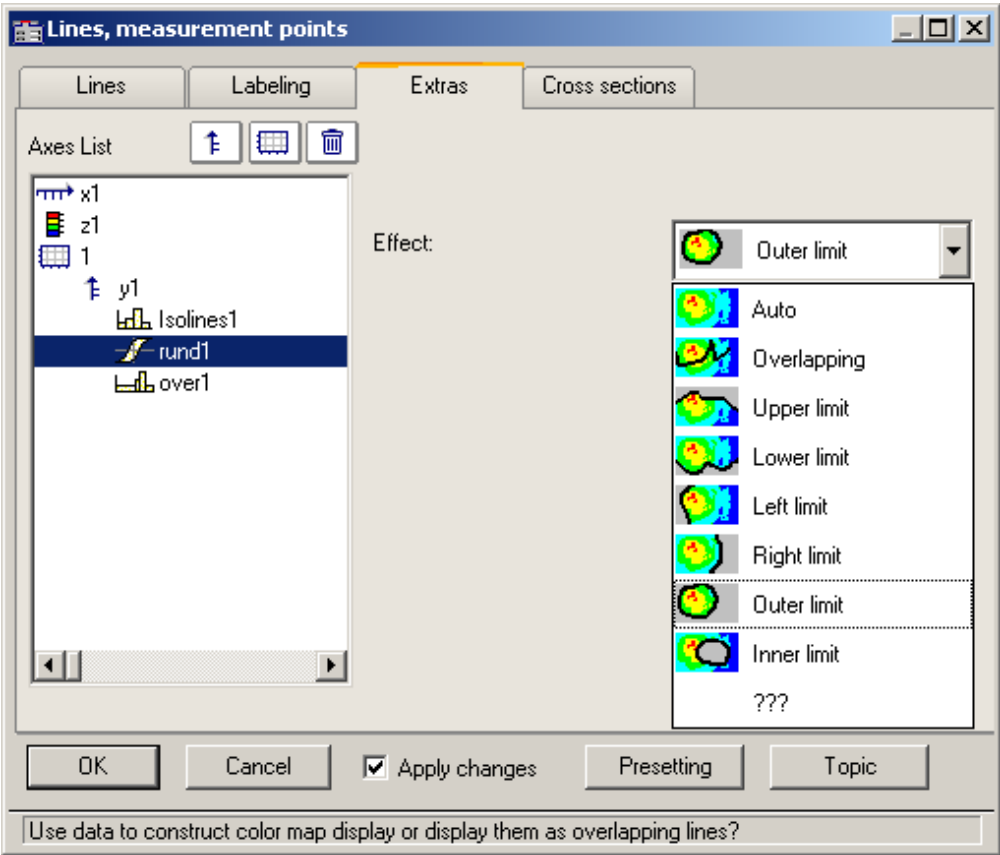
Data are moved in the picture in terms of the X- and Y-coordinates.

In the display style *Color map*, this menu contains the function *Effect*, with which you can use XY-data to construct the color map, or to superimpose as actual lines on the color map. This makes such displays as shown below possible.



In this picture you see a function for constructing the color map, and a function for superimposing a line over the color map.

Settings	Description						
Effect	The available <i>Effects</i> are 1) <i>Overlapping</i> for superimposing a line over the color map, and 2) various margin functions affecting the construction of the color map.						
	<table><tr><th>Effect</th><th>Description</th></tr><tr><td>Overlapping</td><td>An XY-waveform is superimposed as a line over the color map. For the purpose, the usual line properties are available.</td></tr><tr><td>... limit</td><td>An XY-waveform is inserted into the color map as a real line and frames the color map view as either the upper, lower, left, right, outer or inner margin, according to the setting for <i>Effect</i>.</td></tr></table>	Effect	Description	Overlapping	An XY-waveform is superimposed as a line over the color map. For the purpose, the usual line properties are available.	... limit	An XY-waveform is inserted into the color map as a real line and frames the color map view as either the upper, lower, left, right, outer or inner margin, according to the setting for <i>Effect</i> .
Effect	Description						
Overlapping	An XY-waveform is superimposed as a line over the color map. For the purpose, the usual line properties are available.						
... limit	An XY-waveform is inserted into the color map as a real line and frames the color map view as either the upper, lower, left, right, outer or inner margin, according to the setting for <i>Effect</i> .						



Symbol	The number of symbols can be defined with "Option: symbol count" and "Number of symbols". Symbols are activated on the line tab ⁷⁵⁷ .

Settings	Description
Scaling	Scaling Display uncertainty Color Auto Auto from 0 to 1 (relative) from 0 to 1 (with offset) in mm (relative) in mm (with offset) ???

Independently of the coordinate system, the data can be scaled in millimeters. By this means, it is possible to use a data set to define a graphical object, which retains a constant size regardless of the current zoom level.

Alternatively, the data can be displayed in a value rage from 0 to 1 across the entire coordinate system.

The position of the data can be set to a particular coordinate by means of an offset. This is accomplished using the user-defined property *Offset X display* or *Offset Y display*

Display uncertainty (tolerance range)	Scaling	Auto
	Display uncertainty	Filled area
	Color	auto
	Measurement uncertainty	Auto Auto Standard uncertainty Expanded uncertainty ???

The data set's measurement uncertainty can be indicated either as a *Line* or *Filled area*. The value of the measurement uncertainty is entered as a property of the variable already when the data are captured, and it appears in imc FAMOS as a *user defined channel property*. Alternatively, it can also be set in imc FAMOS as a *user defined property* in the *channel property* dialog, or by means of the function "UserPropSet(Data, "Uncertainty" ...).

Example: UserPropSet(Data, "imc33", 0.3, 0, 0) or UserPropSet(Data, "Uncertainty", 0.3, 0, 0)

Measurement uncertainty:

- Expanded uncertainty: In accordance with GUM, the expanded measurement uncertainty is a symmetrical interval around the measurement value.
- Standard uncertainty: Measurement uncertainty expressed as standard deviation.

11.6.4.5 Cross sections

Here you can set the default appearance of cross-sections for x-, y-, and z-data sets or segmented waveforms or segmented waveforms with the respective position. For more info see the section [Connect with 3D](#)

11.6.5 Additional display options

11.6.5.1 Display

Function

The display mode of a curve window can be defined in a variety of ways.

Calling the dialog

Select menu option "*Configuration*" > "*Display*" in the curve window. A dialog appears in which the curve window display can be defined.

Tap	Description
Display ⁶⁴⁹	Display variants of the curve window
Legends ⁷⁶⁸	All curve windows can include a legend, with the channel (variable) name, line color etc.
Map ⁷⁷¹	Map lets you place a picture in the background. It is additionally possible to layout the picture in specified coordinates.
Color palette ⁷⁸⁴	A channel's color can be governed by a reference signal's amplitude.

Display

Display

Legend

Map

Color palette

Text

Display type

☐ Default

☒ Stack y-axes

☐ Waterfall diagram

☐ Last value as number

☐ Table

☐ Bar meter

☐ Color map

☐ 3D

☐ Polar plot

Scale of x-axis

☒ Chosen x-unit

☐ Date/time: absolute

☐ Days/hours/minutes: relative

☐ Third/octave labeling

Reference for dB-display:

1.00000E+0

☒ Labeling

Coordinate system size:

Default

☐ Show x = 0 (trigger)

☐ Number trimming (useful when scrolling)

☐ Grid

OK

Cancel

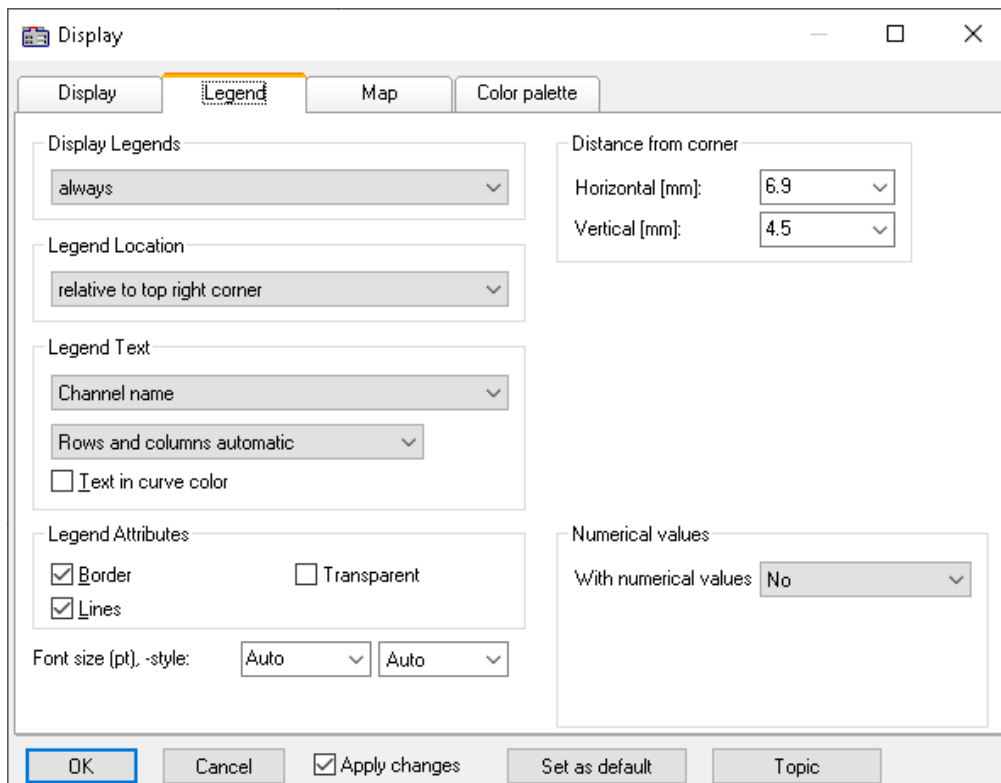
☒ Apply changes

Set as default

Topic

11.6.5.1.1 Legends

All curve windows can include a legend, with the channel (variable) name, line color etc. Select "*Configuration*" > "*Legends*" in order to make the desired settings.



Display Legends

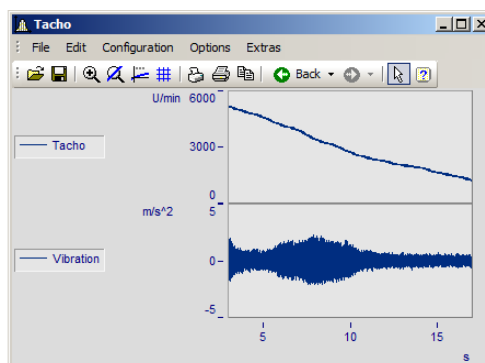
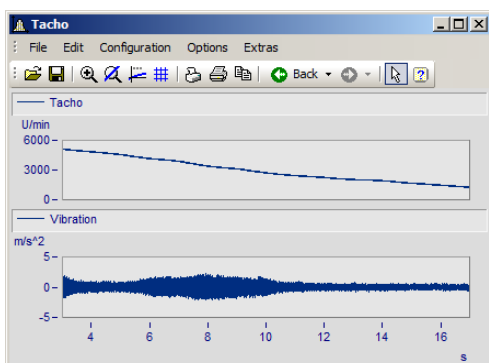
Specify here the conditions under which a legend is to appear. The following options are available:

- *Automatically*
- *Always*
- *Never*
- Only if more than one curve is present

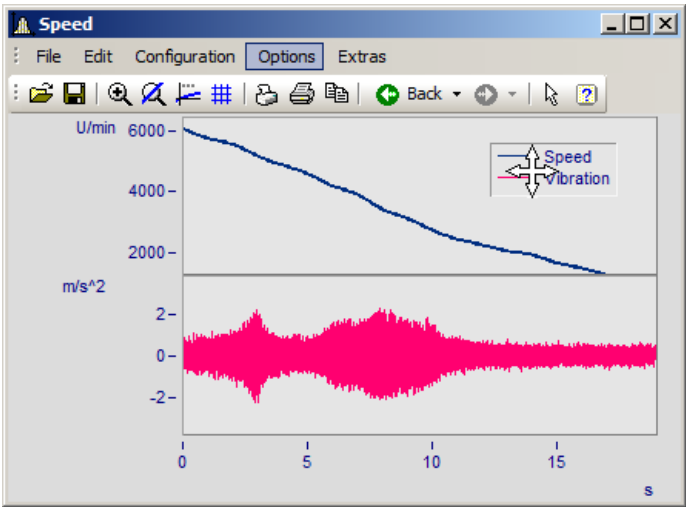
Legend Location

By default, the legend for all variables appears above the curve.

Alternatively, the combobox control *Legend Location* allows the legend to be placed above or to the left next to each coordinate system.



For making placements **within** the coordinate system, additional entries are available. It is possible to move the legend to anywhere within the coordinate system by means of the mouse.



Legend Text

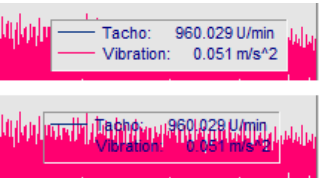
Specify here the text which is to appear in the legend, as well as the format and color of the text. The following options are available:

Settings	Description
Channel name	the variable name, with group name if available, is displayed. In the Data-Browser, affiliation to a membership is indicated by <i>Channel @ Measurement</i> ; if there is no such affiliation, only the channel name is displayed.
Channel name (without group name)	only the variable name is displayed.
Channel comment	the comment entered for the channel on the "Characteristics"-file card in the imc FAMOS dialog <i>Variable/Properties</i> is displayed. See also the notes on Greek letters ⁹⁰¹ .
Channel name and comment	the variable name, with group name and corresponding comment are displayed.
Channel name (without group name) and comment	only the variable name and the corresponding comment are displayed.
Channel name without measurement	only the channel name will be displayed. Even if the channel belongs to a measurement, the name of the measurement will not be displayed. - Only relevant in the Data-Browser.
Channel name with index of selected measurement	if the channel belongs to a measurement and if that measurement is currently selected, then the index (number) of the selected measurement will be displayed instead of the name of the measurement. If no index is available, then the name of the measurement will be used. If no measurement is available, then only the channel name itself will be displayed. - Only relevant in the Data-Browser.
Channel name (without group name and without measurement)	only the variable name is displayed.
Channel name (without group name and without measurement) and comment	only the variable name and the corresponding comment are displayed.

The legend text can be ordered in any of the following ways:

- Rows, columns automatic
- Always row
- Always 1 column
- Fixed number of rows (enter the number)
- Fixed number of columns (enter the number)

Legend Attributes



The option *Border* places the legend in a frame, which provides a 3D-effect.
Lines draws a line behind the variable's name in the color of the associated curve.
Transparent makes the curves visible behind the legend.

Numerical values

With running measurements, it is possible to display the updated values in the legend. To do this, in the box of controls "*Numerical values*", select the list box entry "*Last value as number*" for the control "*With numerical values*".

Numerical values

With numerical values: Last value as number

Separator: :

Units: Yes

Max. digits: 2

Settings	Description
Separator	<i>Separator</i> inserts a colon or equal sign.
Units	<i>Units</i> controls whether or not the units are indicated: <i>yes</i> or <i>no</i>
Max. digits	Maximum amount of digits possible. This determines the distance between the variable's name and the numerical value.

Font size (pt), -style

The font size and font style for the legend can be set manually here.

Font size (pt), -style: 12

Auto

Auto

Default

Bold

Italic

B+I

u

OK

Cancel

☒ Appl

OK, Cancel, Set as default

For details on operation, see the chapter [Confirmation bar](#)

11.6.5.1.2 Map

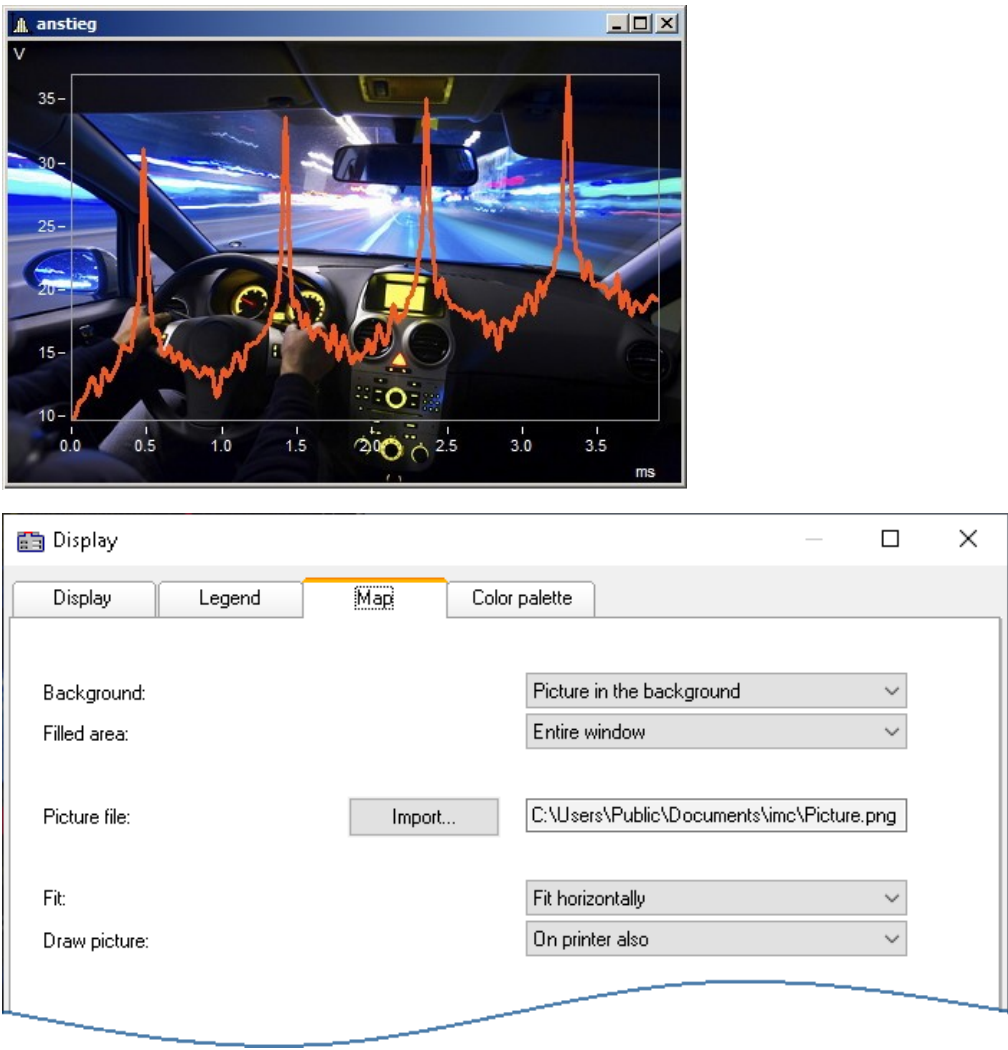
Function

Map lets you place a picture in the background. It is additionally possible to layout the picture in specified coordinates.

Mouse Operation

In the dialog called by the menu item *Configuration / Display*, select the option *Map*.

11.6.5.1.2.1 Background picture



Settings	Description
Background	Select Background to place a picture in the "Background"
Filled area	"Coordinate system" or "entire curve window"
Fit	"auto" adjusts both the picture's horizontal and vertical dimensions. "Retain size" does not adjust the size but centers the picture. "Fit horizontally" and "Fit vertically" each adjust the picture's respective dimension.
Draw picture	The option <i>Only on screen</i> is used to prevent the printer from printing the picture out.

Constraints

- *Background*: Only for *Default display*, *Stack Y-axes*, *Last value as number*, and *Table*. Applicable with limits to *Color map* and *Bar meter*, since the largest part is hidden. Not possible for *Waterfall* or *3D* display.
- The button *As presetting* is available in either the *Background picture* or *auto* modes. In the presettings, the filename and directory path of the picture are recorded. For any new curve window, the system first looks for this picture there.
- The background picture or the map is saved with a curve configuration file (CCV).
- .BMP files are very large, but can be adapted well to the display. JPG files normally are smaller, but are less capable of being adapted to the display.

11.6.5.1.2.2 Map

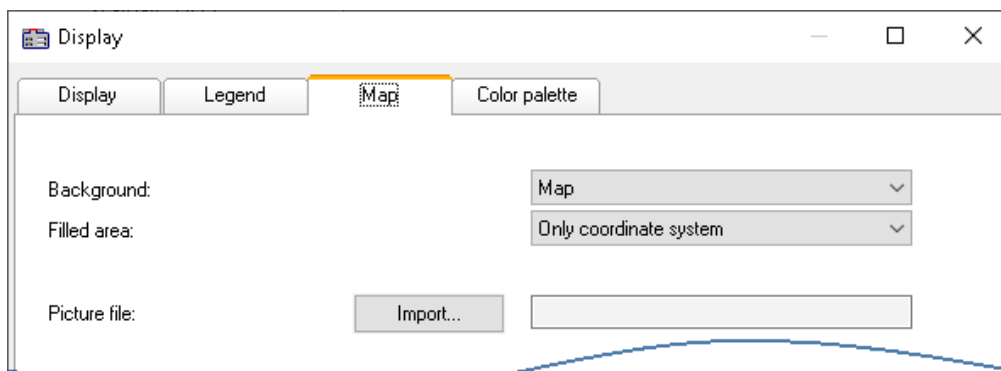
Positioning a map as the background picture

On the following conditions, a picture can be correctly displayed as a map:

- selected curve window configuration: *Display \ Default*
- linear axes
- only one coordinate system

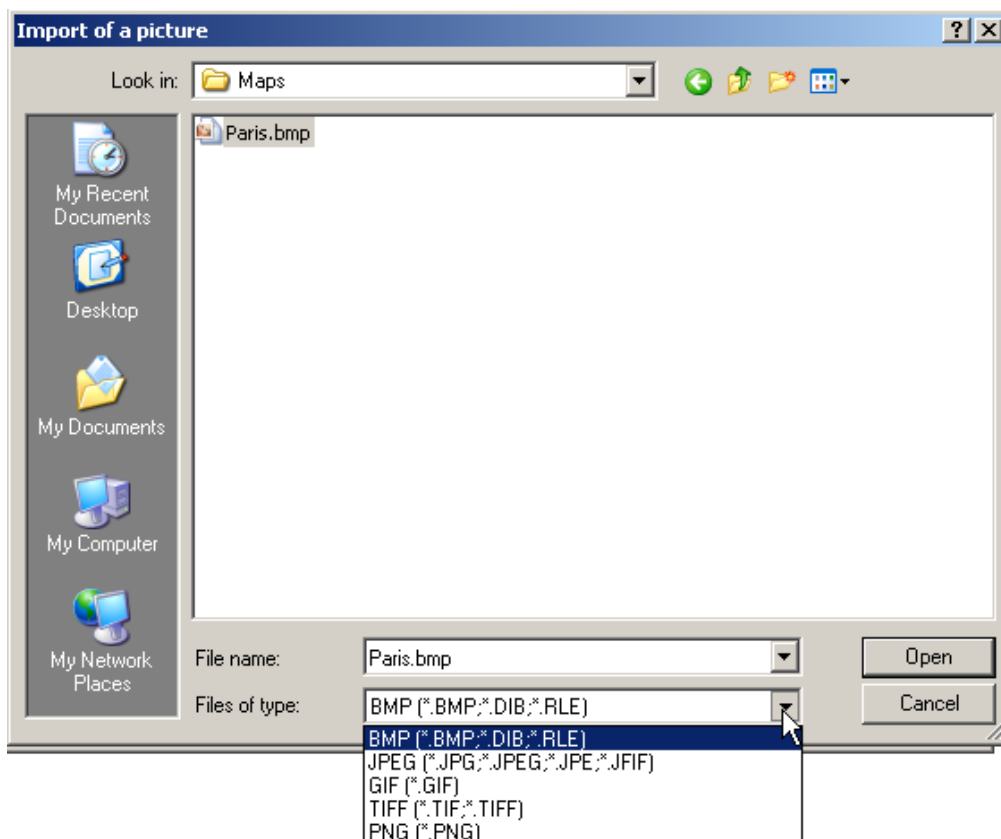
The following example illustrates how GPS data are linked with a map.

Import of a picture



Select the "Map" page and set the background to "Map"

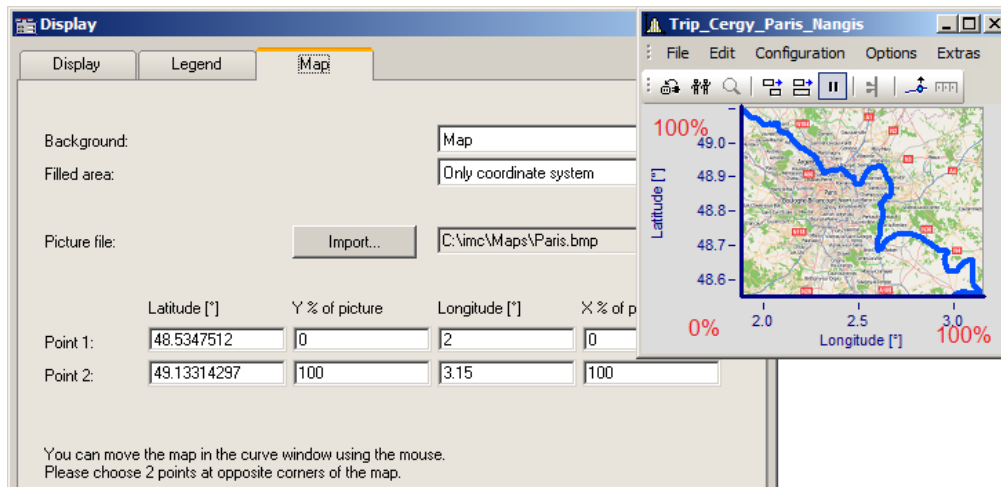
Click on the button *Import* and select a *picture file*. Specify its format.



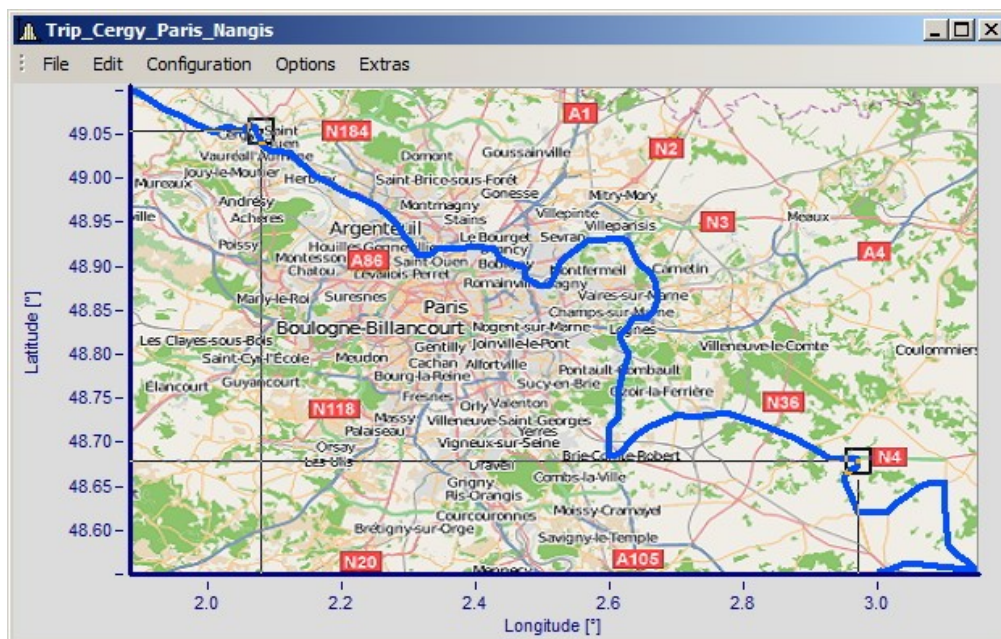
Selecting a picture file. All customary graphics formats are available.

Adapting maps and Map Mode

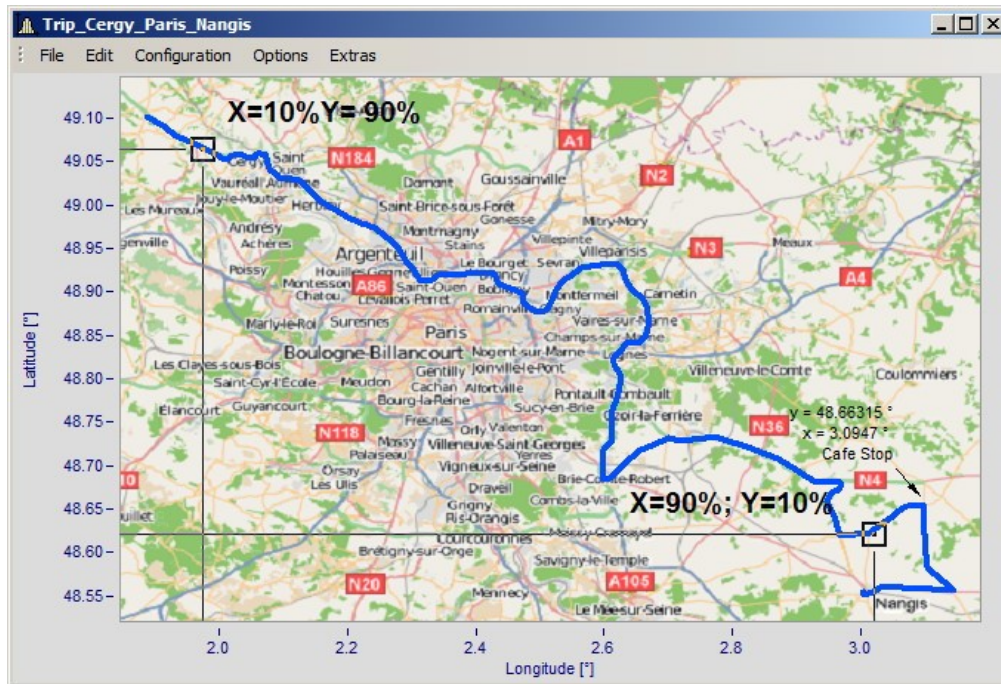
To position a map, you must specify two points. This accomplished by making entries in the input boxes. The curve window is then in Map Mode.



Specifying points: Point 1 at bottom left, Point 2 at top right



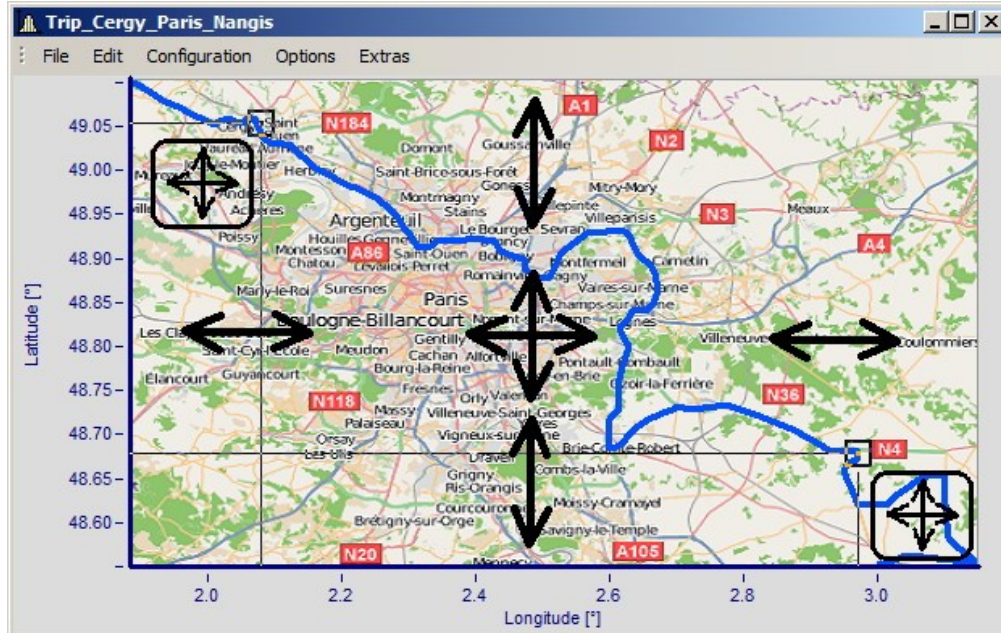
The points are entered into the curve window while the Map dialog is open. Move the points by clicking within the squares and dragging the points using to the mouse to the correct positions.



By moving, the points' X- and Y-positions are updated.

The mouse pointer changes appearance in response to the position in the curve window.

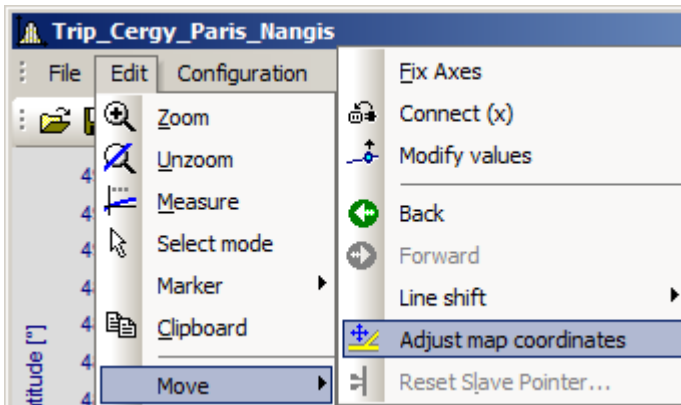
- Drag the entire map by clicking in the center of it.
- Stretch the map by clicking either at the right, left, top or bottom of the map.
- Move the position points by clicking within the squares.



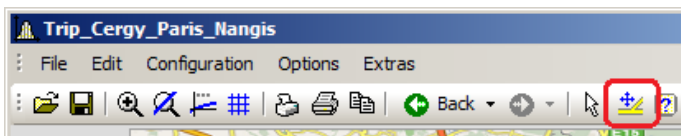
Adjusting the map and points in response to the mouse position

Activating Map mode retroactively

Once the map dialog has been closed, you can use menu commands to adjust the map: *Edit\Move\Map*.



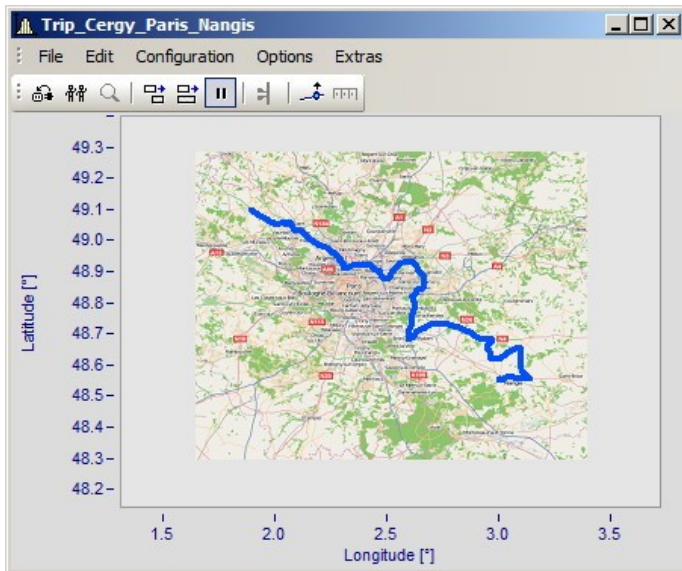
Alternatively, activate the mode by means of the toolbar. To do this, adapt the [Toolbar](#)⁸⁹⁴.



You can also drag this function into the toolbar.

Curve window and map

The map is dependent on the axis' coordinates:



This shows the effects of magnifying the axis width

Not that the picture is magnified, stretched or trimmed. Only the details of the original picture can be displayed. The maps used must have an appropriate projection. Constant latitude and longitude lines must be displayed as straight lines. The distances must be equidistant.

Naturally, a map extending from the north to the south pole cannot meet this requirement. The same applies to map material near the poles and the dateline.

Alternatively, you can use a [static picture in the background](#) ⁷⁷¹.

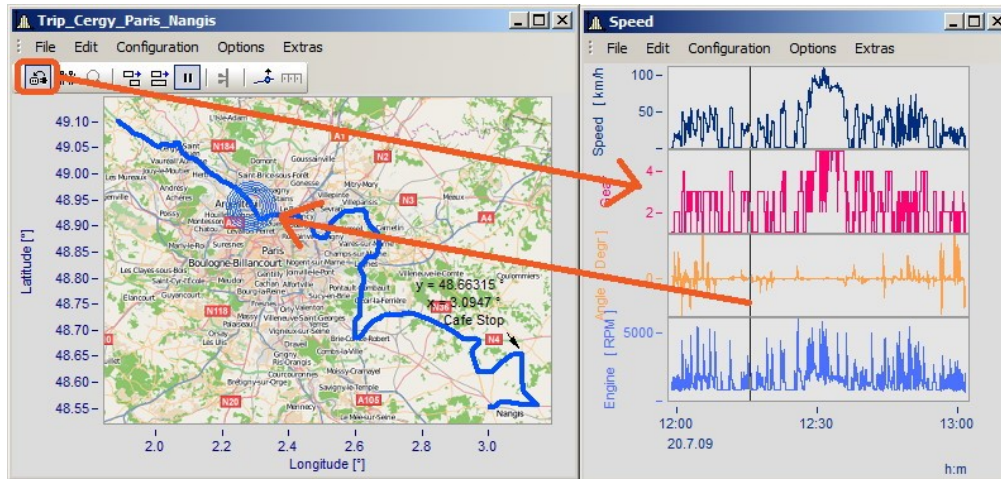
Constraints:

- **Map:** Only for Default display and Stacked Y-axes, but only for the first coordinate system and first Y-axis. All axes must be set to linear (no 1/3-octaves or absolute time).
- The button "As *presetting*" is not available in Map mode.
- The background picture or map is saved with a curve configuration file (CCV).
- .BMP files are very large, but can be adapted well in a display. JPG files are smaller but are not able to be adapted as well.

Linking a map with time data

It is possible to link a curve window displaying time data with a map display.

If the map contains an XY-channel, it is possible to link this channel with time data from a different curve window. To do this, drag the linking tool icon from the Communication toolbar to the desired curve window.



Linking the map with time data from a different curve window

You can move the position on the map in concert with the time marker line in the second curve window.

11.6.5.1.2.3 Map (from Internet)

Automatic loading of a map from the Internet

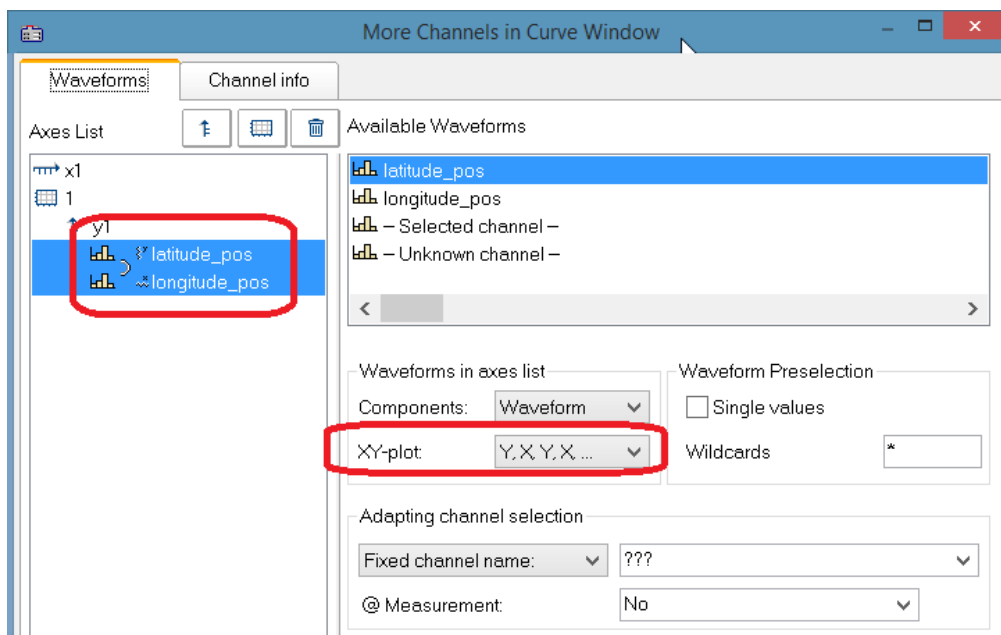
On the following conditions, a picture can be correctly displayed as a map:

- selected curve window configuration: *Display \ Default*
- linear axes
- only one coordinate system
- a data set with plausible position data is displayed.

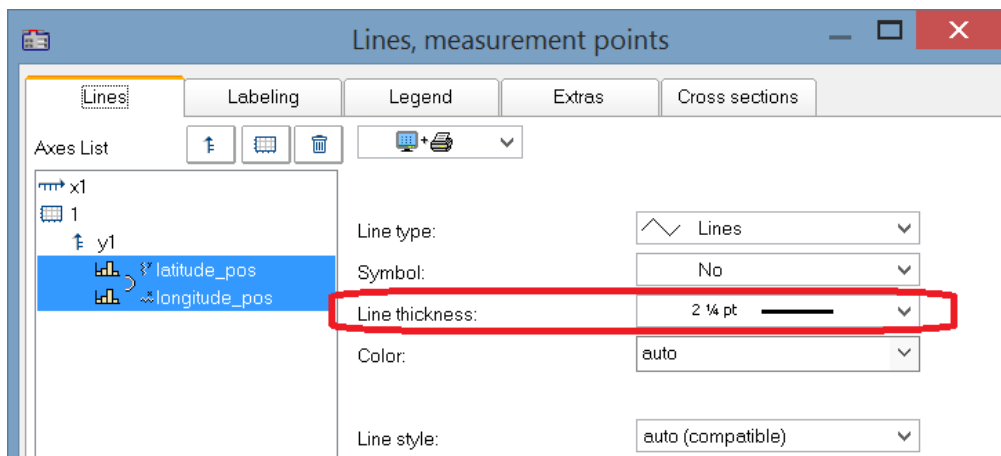
The following example shows how GPS data are automatically superimposed on a map from the Internet.

Selection and display of the GPS data as an XY-plot

Load the data sets for Longitude and Latitude and display them as an XY-plot. To do this, open a curve window with the two components and superimpose them by means of the dialog "More Channels in Curve Window":

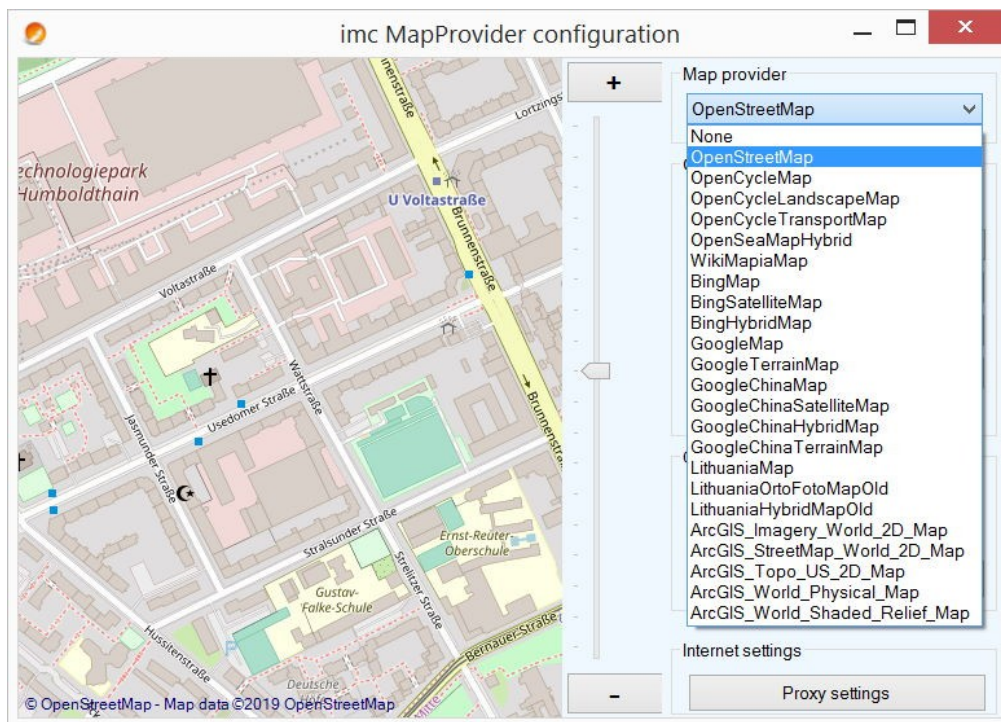
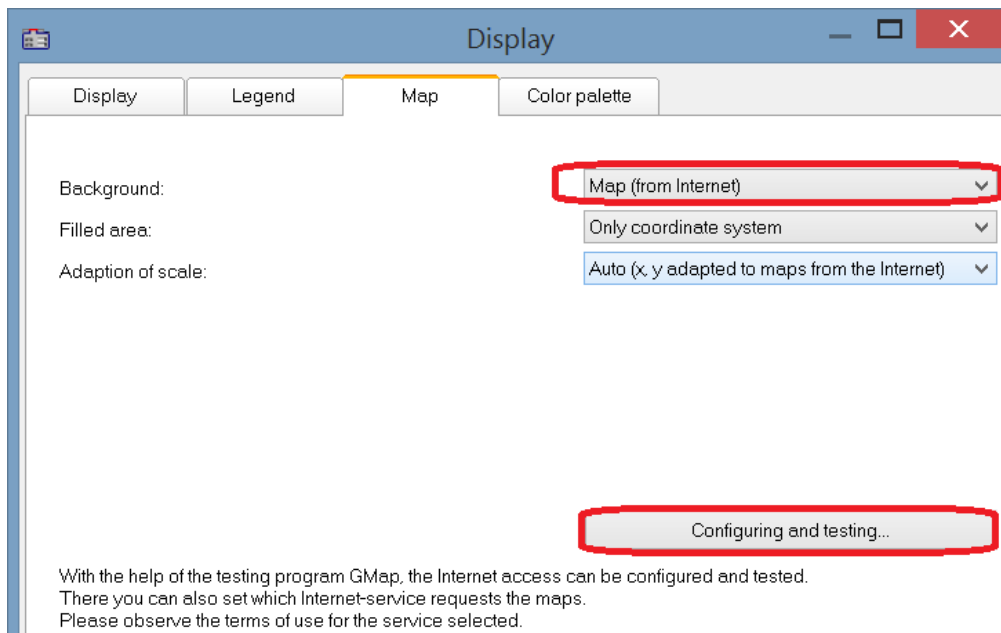


It is advisable to draw the lines more thickly:

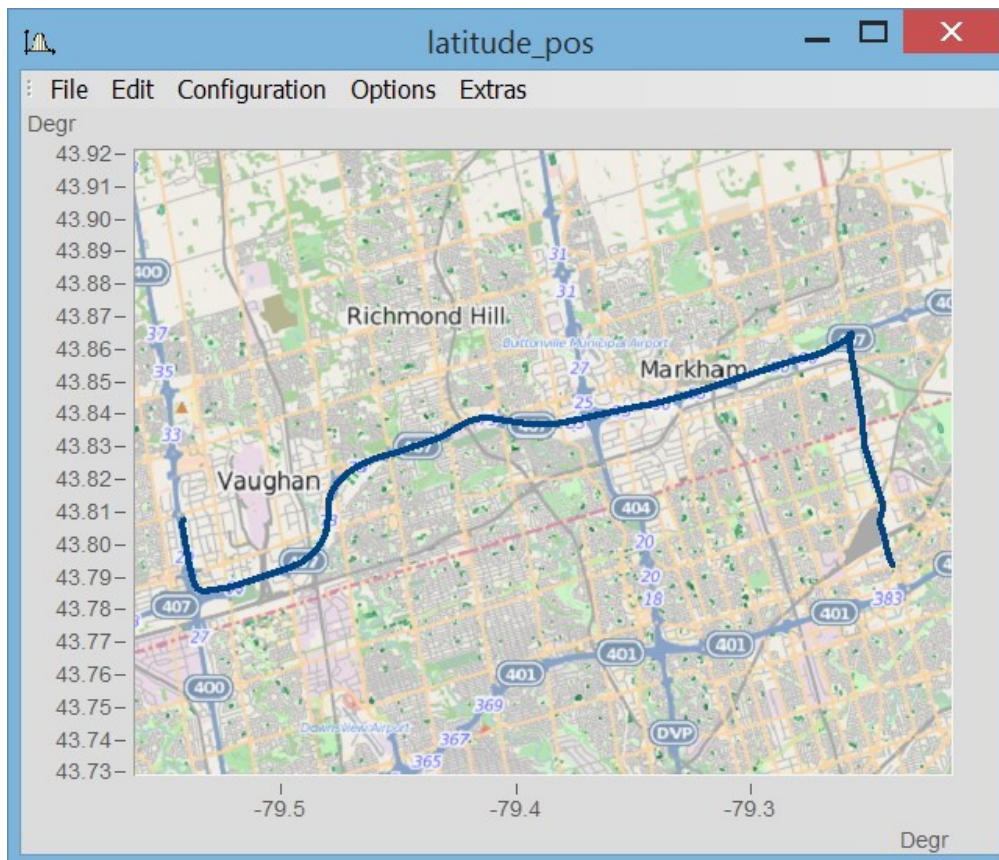


Selecting the map service

Select "*Map (from Internet)*" on the page "*Map*" in the dialog "*Display*". The setting "*Configuring and testing*" selects the Internet-service:



After selecting the map service, the map section loads immediately if an Internet connection already exists. In the curve window, this may appear as shown below, for example:



Note

Changing the map-provider

After changing the map-provider, the provider's maps are not automatically reloaded. To do this, the axes of the curve window must be actively changed, e.g. by changing the zoom factor with the mouse wheel or moving the image section.

Adding a map provider

The selection of map-providers can be supplemented with the file "*AdditionalMapProvider.config*" in the folder "*C:\ProgramData\imc\Common\Settings*". If this file doesn't exist on your system, create a text file having this name.

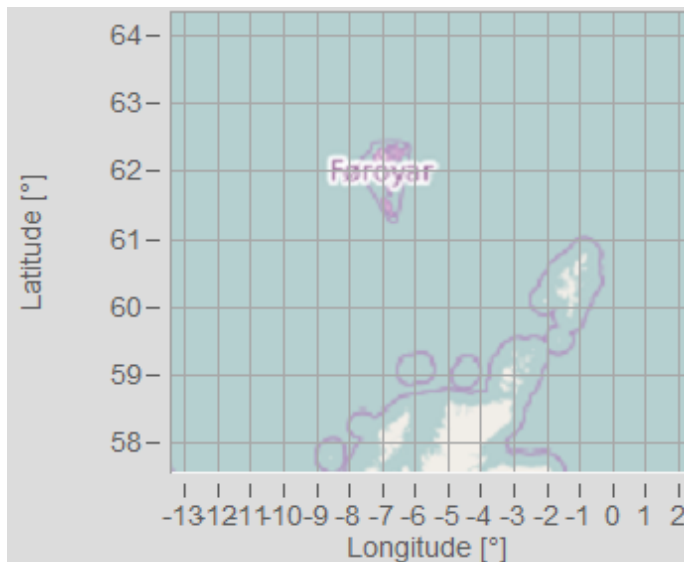
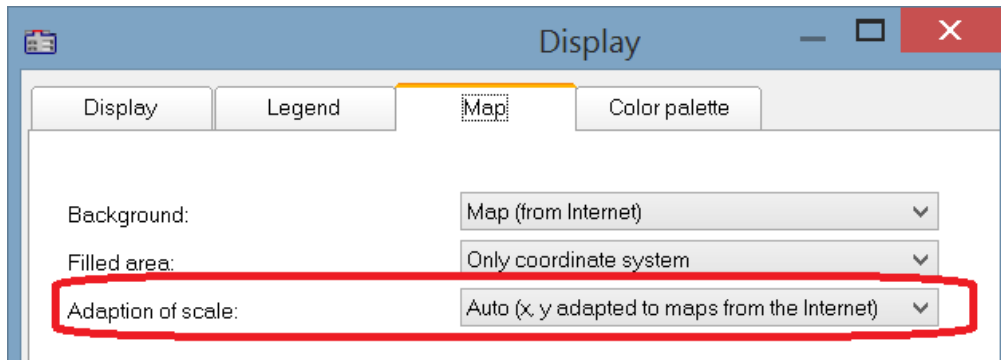
1. Open the folder. By means of *%programdata%*, Windows opens the folder *ProgramData* and then go to *imc\Common\Settings*.
2. The file *AdditionalMapProvider.config* is an XML file with the following structure:

```
<?xml version="1.0" encoding="utf-8"?>
<MapProviders>
  <MapProvider Name="OpenRailwayMap-Standard" Url="http://a.tiles.openrailwaymap.org/standard/{z}/{x}/{y}.png" User="" Pwd="" TileThreads="2"/>
  <MapProvider Name="OpenRailwayMap-Signals" Url="http://a.tiles.openrailwaymap.org/signals/{z}/{x}/{y}.png" User="" Pwd="" TileThreads="2"/>
  <MapProvider Name="OpenRailwayMap-MaxSpeed" Url="http://a.tiles.openrailwaymap.org/maxspeed/{z}/{x}/{y}.png" User="" Pwd=""
TileThreads="2"/>
</MapProviders>
```

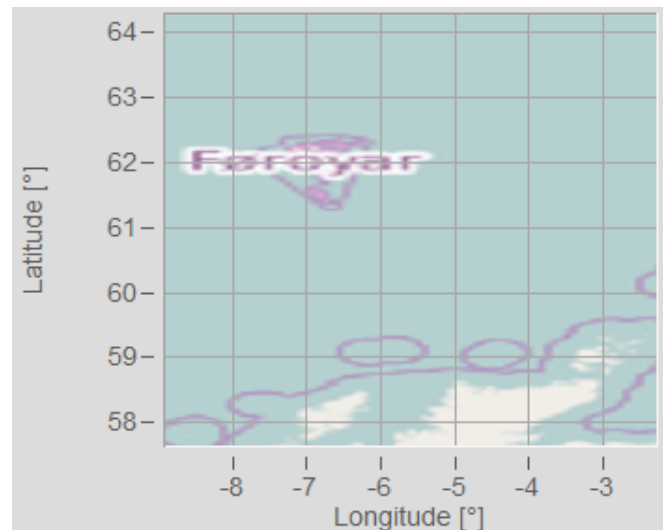
Replace the lines `<MapProvider Name="... TileThreads="2"/>` with the information pertaining to your map provider.

Adaptation of scale

Compensation of distortion due to Mercator projection. Maps typically display the coordinates with a distortion in the direction of the cylinder axis, in order to achieve correct depiction of angles between features on the Earth's surface. In the mode "Auto (x,y adapted to maps from the Internet)", the axis are stretched in proportion to the Mercator projection distortion.



Undistorted map accounting for the Mercator projection



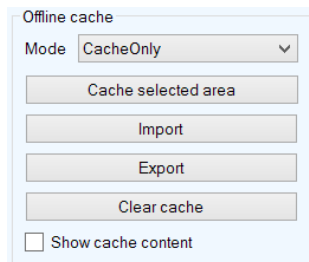
Coordinates equidistantly scaled -> Map distorted

Note

This projection named after Gerhard Mercator (1512-1592) causes imprecision in the curve window, since the coordinates are displayed equidistantly in it. The imprecision increases with the distance from the Equator and the size of the map.

More map service option

Offline cache



Offline cache

Mode: CacheOnly

Cache selected area

Import

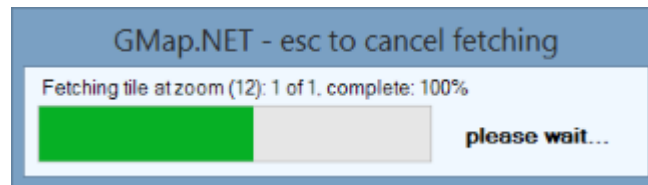
Export

Clear cache

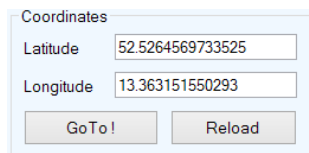
☐ Show cache content

You can export, import, or save a map section locally. To do this, select "*Mode*" *ServerAndCache*" or "*Cache only*".

To make the selection, drag open a rectangular region while holding down the mouse button. When you click on "Cache selected area", the region is downloaded and saved in various zoom levels.



Coordinates



Coordinates

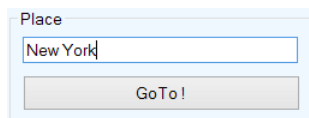
Latitude:

Longitude:

GoTo! Reload

Select the map section by its coordinates and zoom to the desired section.

Place



Place

GoTo!

Select the map section by entering a location.

Internet Settings

Here you can specify a proxy-server.

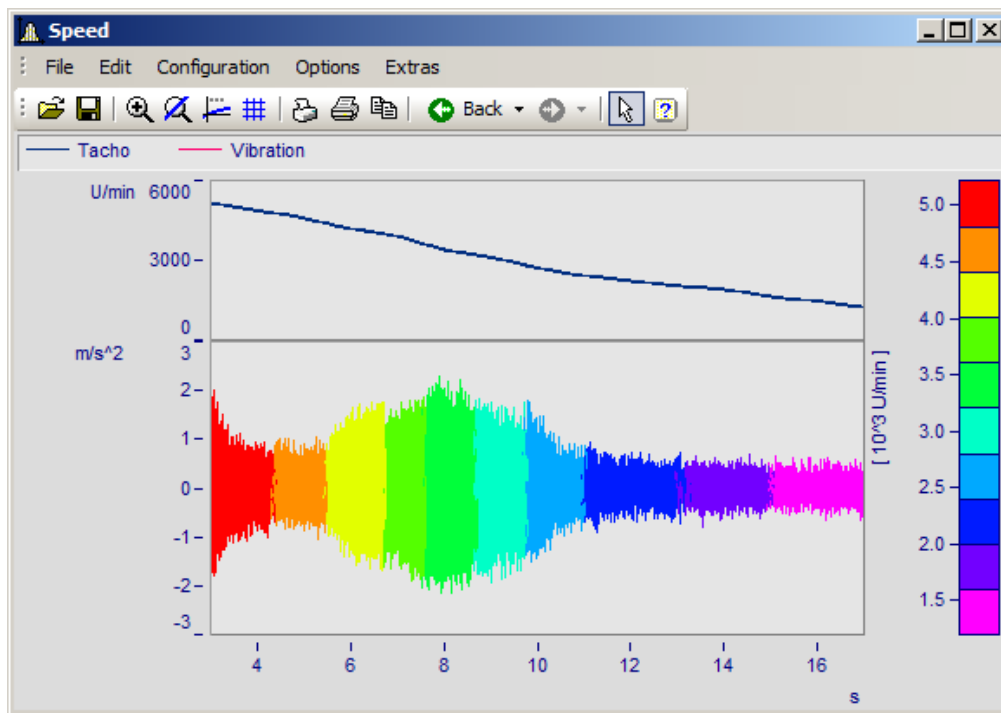
11.6.5.1.3 Color palette

A channel's color can be governed by a reference signal's amplitude.



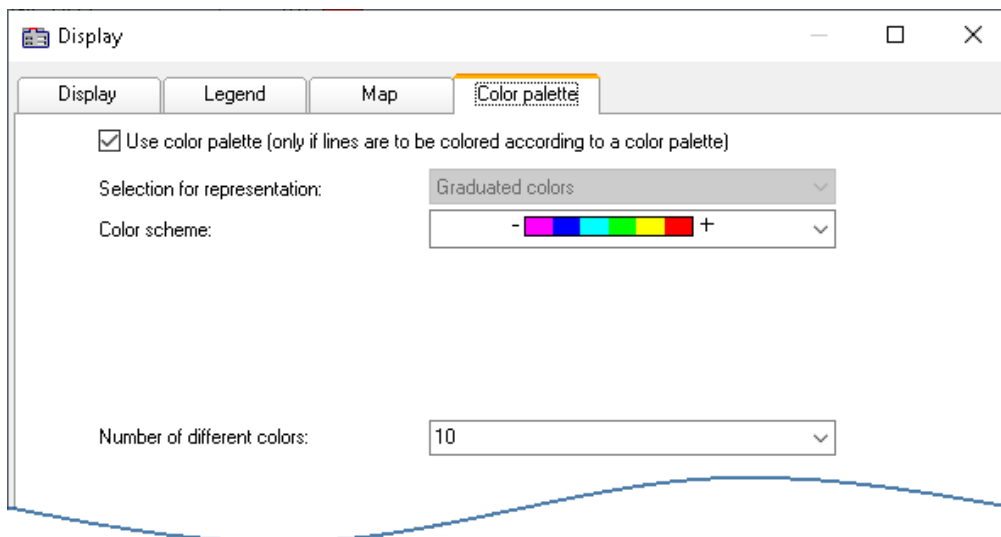
Note

This function is working only for channels using the same sampling rate (x-delta).



Line color of a graph of vibrations as a function of the RPMs

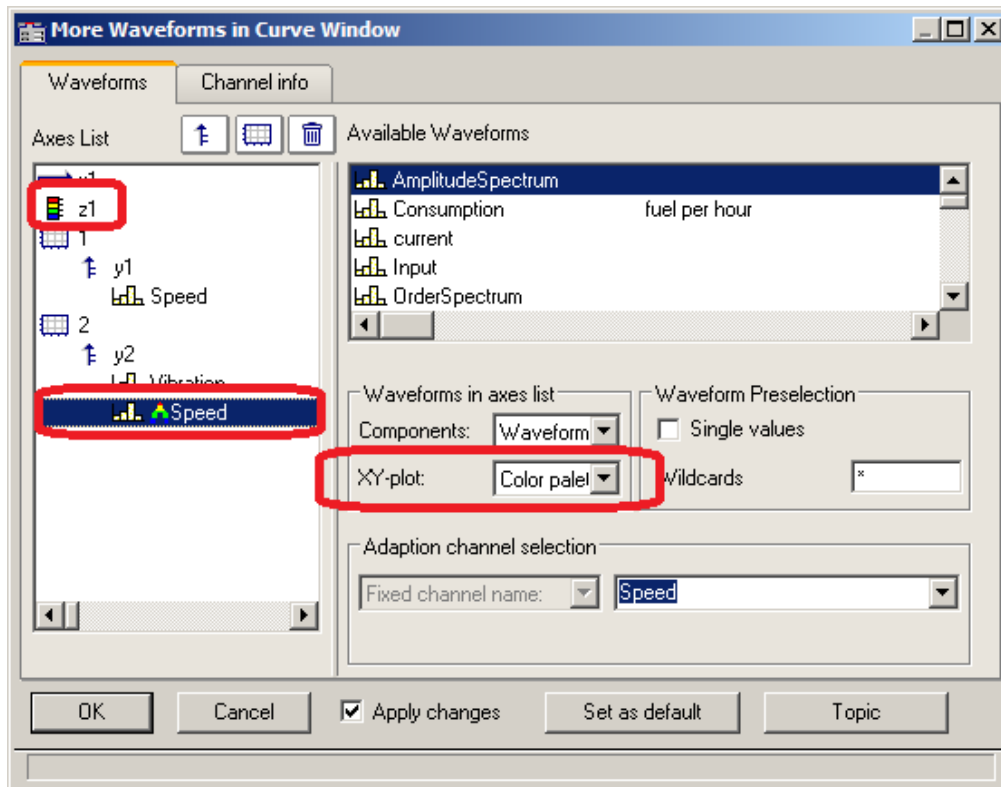
Select the page *Color palette* (not *Display type Color map*) in the dialog called by *Configuration\Display*.



First, activate the color palette by setting a check in its checkbox. In consequence, the controls for *Color scheme* and *Number of different colors*. The control *Selection for representation* is disabled, since at this moment only graduated colors are available.

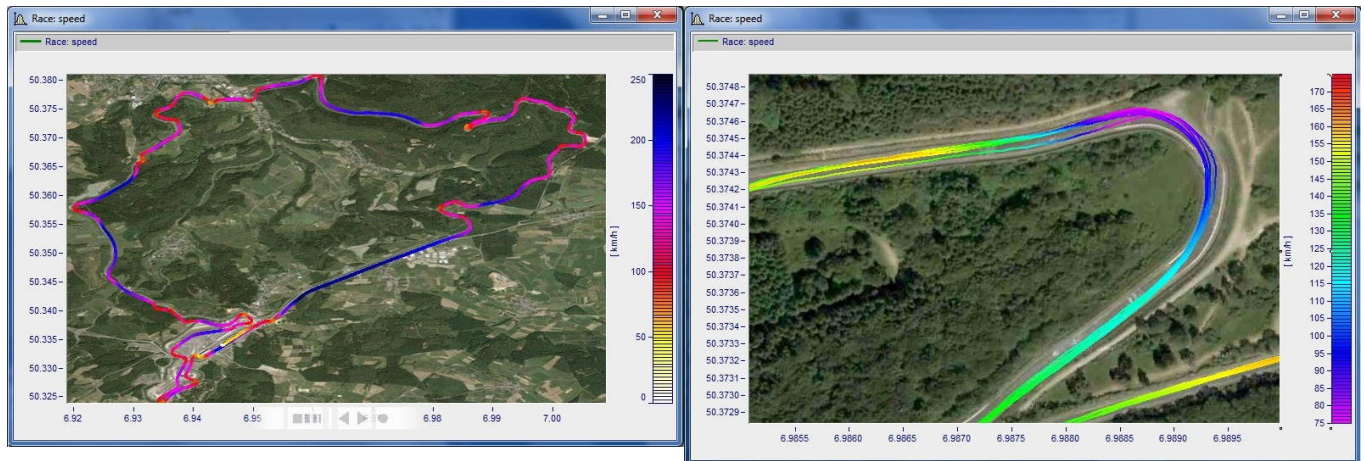
The same settings options for the color scheme and amount of colors are available as for [Color map](#) ⁶⁸⁸.

The assignment is performed in the dialog *More Channels...*, which is called from the curve window's context menu.



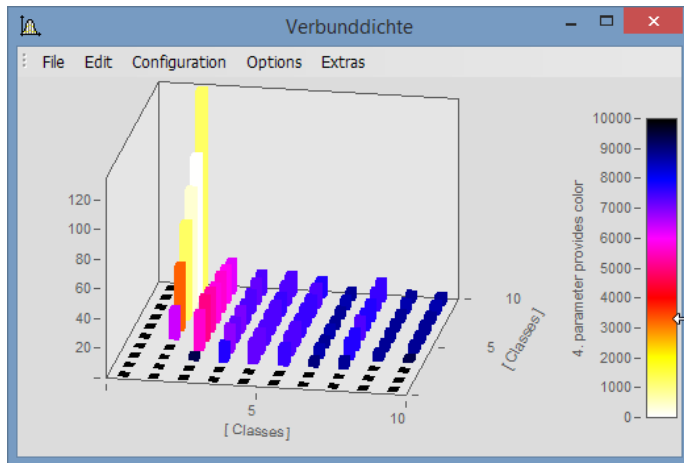
The reference channel is assigned to the same y-axis as the channel to be colored. Select a reference channel and set the XY-plot to *Color palette*.

Example:



Color palette as 4th parameter in 3D

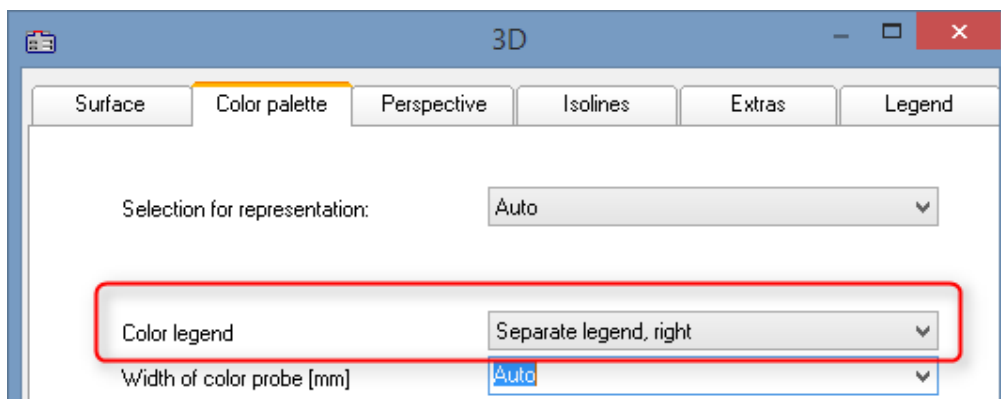
A 3D display can be colored according to a fourth parameter. Thus, 4 dimensions (4D= 3D + color) can be displayed. See also [here](#) ⁷⁹⁵.



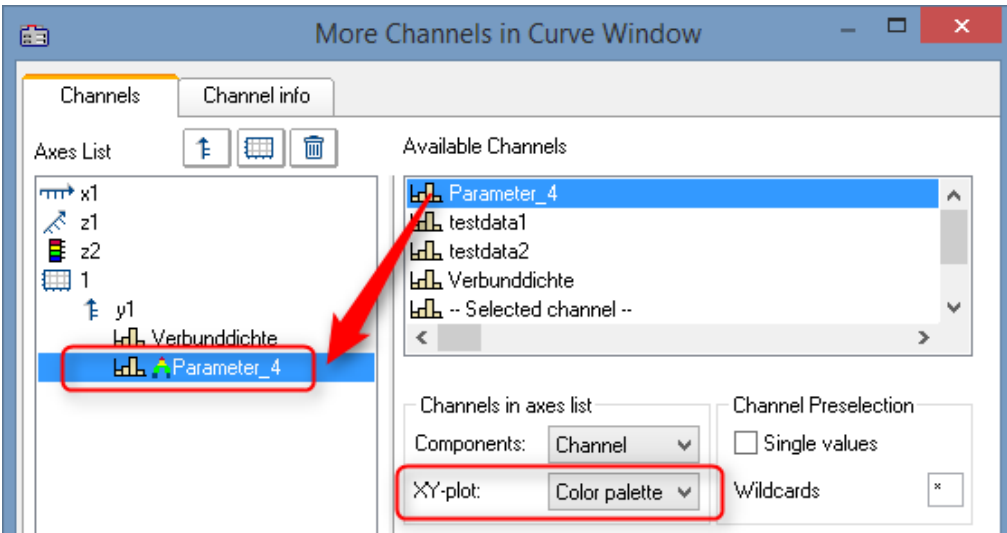
3D display with colors governed by a fourth data set

Procedure:

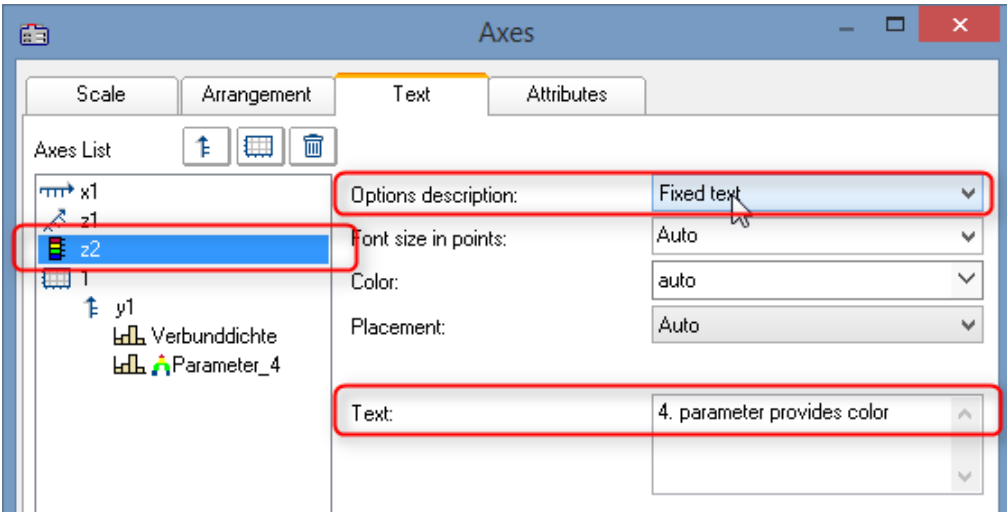
1. Set 3D display with **Separate legend right** or **left** in [Color palette](#) ⁷⁹⁴. With the settings *Auto*, *No* and *Integrated into y-Axis*, it is not possible to generate a colored overlay.
2. Under *More Channels...*, place the fourth variable directly below the 3D data set.
3. Under *XY-plot*, select *Color palette*.
4. Labeling of the color palette by means of *Axes\Text*. This is necessary since the legend does not provide an indication of the color scale's origin.



Activate *Separate color legend* at "Configuration\3D\Color palette"



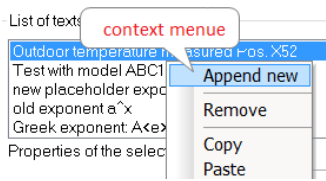
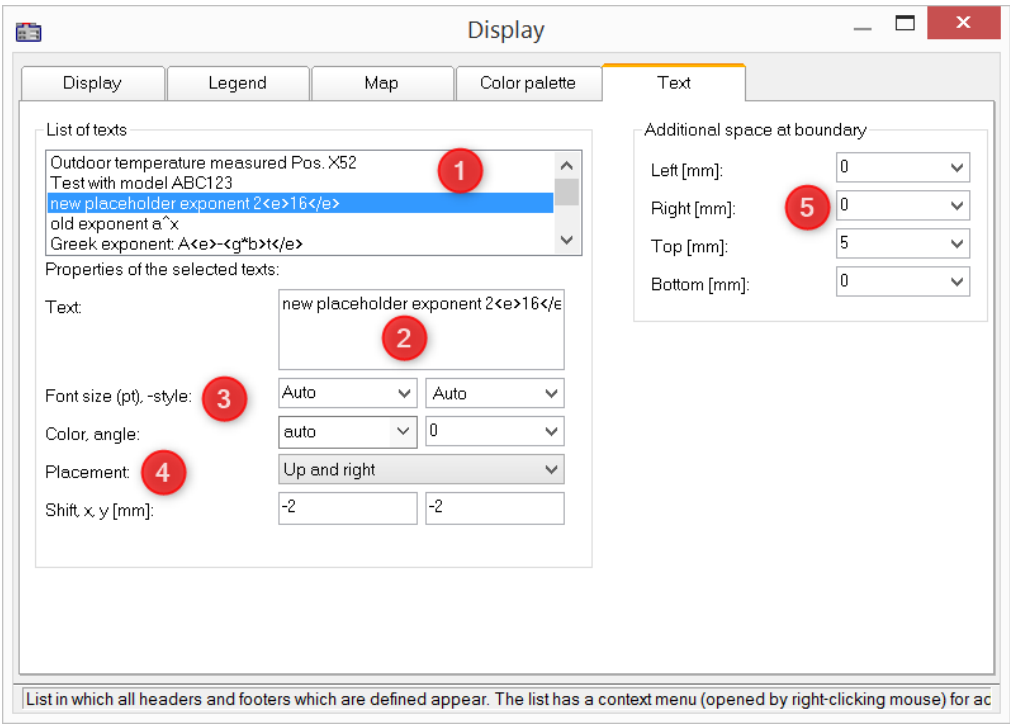
Under "Configuration\More Channels...", assign the color-coded variable



Under "Configuration\Axes\Text" with "Fixed text", state the origin of the coloring.

11.6.5.1.4 Text

The Text page of the Display dialog enables any desired texts to be positioned, for example as titles, headers, footers, or general comments.

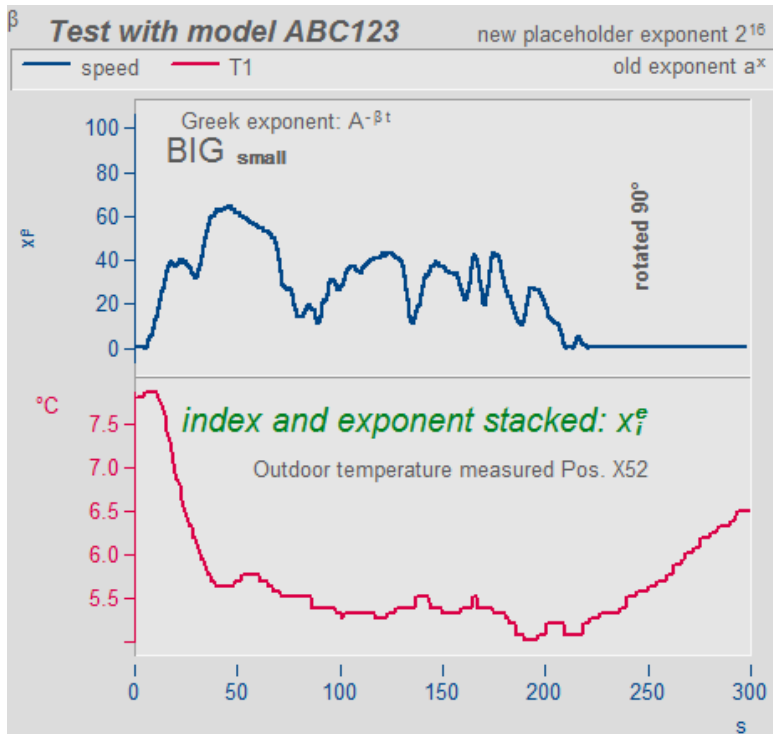


By **right-clicking** in the *List of texts* (1), you open a context menu for managing the entries.

In the *Text* box (2), up to 256 characters are allowed. Line breaks are placed by pressing the CTRL+ENTER keys.

The placeholders supported are described in the chapter [Axes configuration/Text](#) 748.

The *Font*, *font size*, *-color*, and alignment (3) can be set for each entry.



The *placement* (4) is initially determined from a list and subsequently fine tuned by means of the parameters for *Shifting* in the x and y directions. When a position is selected which references a **coordinate system**, an additional parameter for specifying the number of the Y-axis appears.

Beyond the space occupied by the actual curve graph, it is possible to make room for additional texts (5).

Remote control via sequences:

The function `CwDisplaySet` offers some functions "*header.x*" with which to texts can be placed.

```
CwDisplaySet("header.count", 5 )
CwSelectByIndex( "header", 1 )
CwDisplaySet("header.text", "TEXTMITTE" )
CwDisplaySet("header.position", 8 )
CwDisplaySet("header.text.color", 255 )
```

The additional space at the edge is adjusted using the "*legend.x*" functions.

```
CwDisplaySet("legend.space.left", 10.4 )
CwDisplaySet("legend.space.right", 4.7 )
CwDisplaySet("legend.space.top", 10 )
CwDisplaySet("legend.space.bottom", 10 )
```



Reference

Further texts

Further texts can be added via the [axis dialog](#) ⁷⁴⁸.

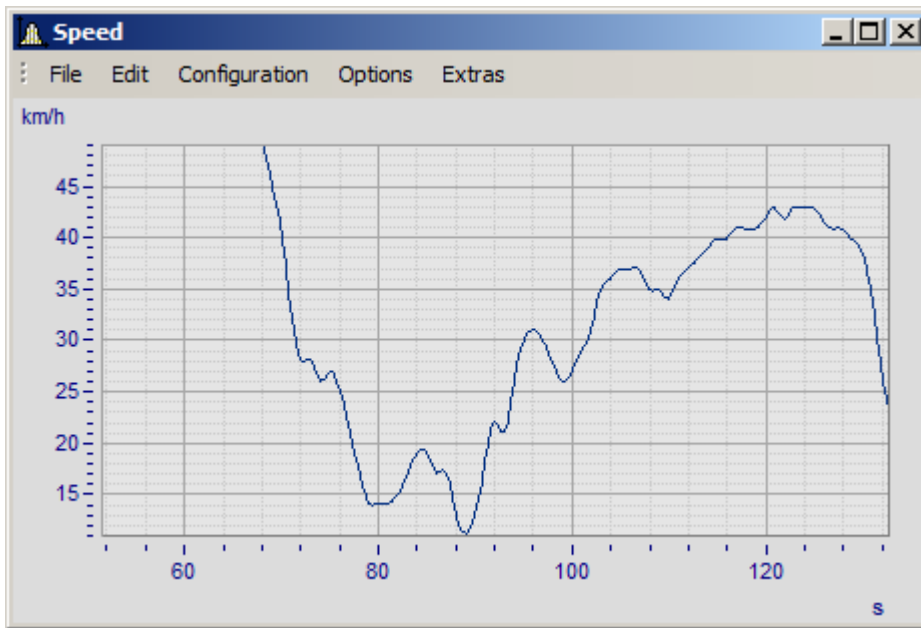
Texts can also be defined as [user ticks](#) ⁷⁵⁰. Here you can also find the function "[Label for raw data](#)" ⁷⁵³.

11.6.5.2 Grid

Function

A grid may be added to the coordinate system in the curve window. The grid is composed of vertical and horizontal lines running through the ticks on the axis, with additional lines between them.

The grid consists of a main and secondary grid. The main grid lines end at the main ticks of the coordinate system axes, where the axes are labeled. The secondary grid lines end at the secondary ticks, optionally inserted between the main ticks (see menu option [Configuration/Display](#)⁶⁴⁹). Settings for the secondary grid are made in the Axes dialog by means of the control [Small ticks](#)⁷⁴¹.



Various line widths and types can be specified for the clipboard or print-out, see menu [Opt./Clipboard Settings](#)⁸⁷⁸.

A grid is drawn for a logarithmically represented axis (but not in dB) so that eight lines appear between each set of markings on the axis. The eight lines are easy to interpret, since a factor of ten lies between each set of axis markings. The lines designate equidistant points, i.e. between the markings 1 and 10, the points 2, 3, 4, 5, 6, 7, 8, 9 and between the markings 10 and 100, the points 20, 30, 40,... . The lines are then spaced as on logarithmic graph paper.

Mouse Operation

- Select menu option *Grid* from the *Configuration* menu in the curve window.

Remarks

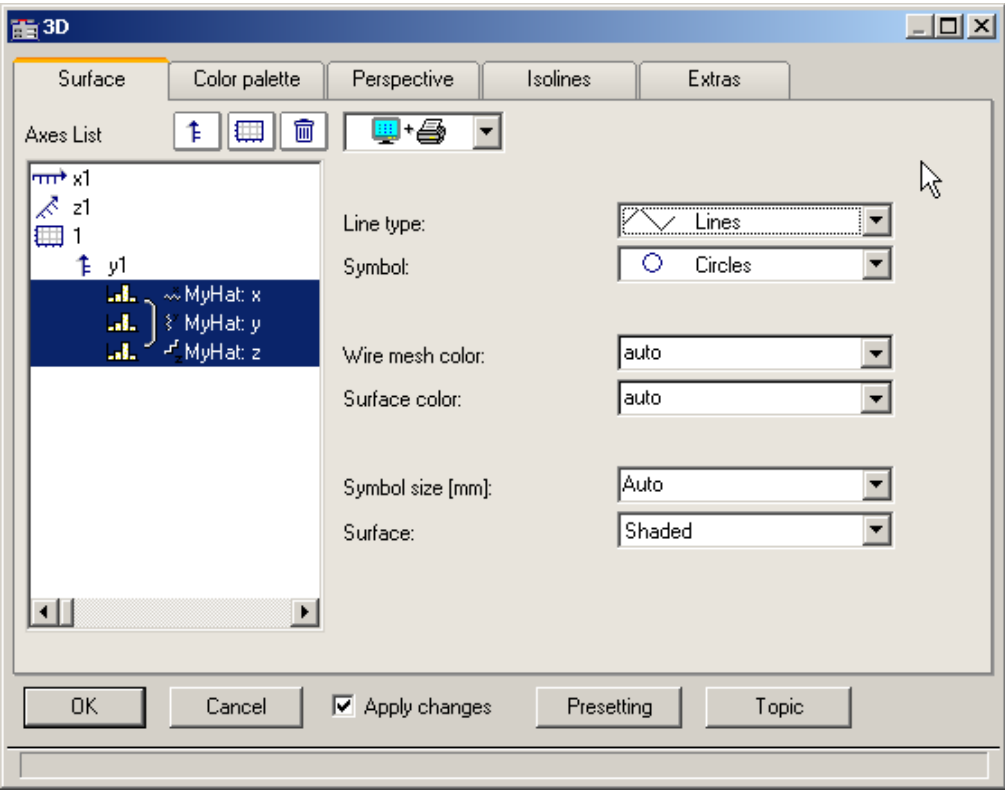
- The color of the grid lines can be determined in the *Colors...* dialog.
- A grid is not drawn if labeling is not visible.
- The menu option *Grid* is marked with a checkmark when grid display is selected.

11.6.5.3 3D

You have various options in the *3D* display, to modify and optimize the display of surfaces. If you have selected the *3D* option, you achieve the setting dialog for the 3D design *Configuration / 3D*. You can also achieve this setting dialog with the corresponding button in the toolbar.

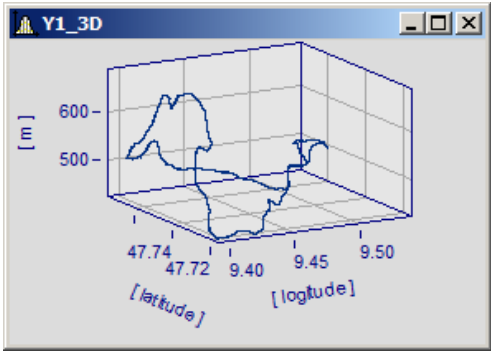
11.6.5.3.1 Surface

In the dialog page *Surface* there are additional settings for 3D-display besides the settings described under [Lines](#)⁷⁵⁶.

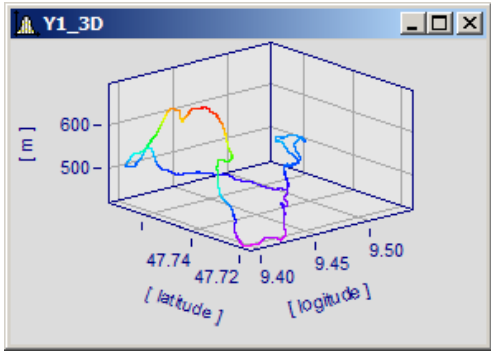


Settings	Description
Line type	In 3D-display, the only available line types are straight lines, no lines, or interpolated lines connecting measured points. For bars, see here⁷⁹³. Lines Lines Only symbols Interpolation, cubic poly ???
Symbol and Symbol size	See the section on lines ⁷⁵⁶ .
Wire mesh color	Here you can set the color of the grid, or of the connecting lines between measured points. If you set <i>auto</i> , then the color will be set according to the global color setting (color palette ⁷⁹⁴).
Surface color	Here you can set the color of the surface. If you set <i>auto</i> , then the color will be set according to the global color setting (color palette ⁷⁹⁴).

Settings	Description	
Surface	Surface	Description
	Wire mesh shaded	The individual measured points are connected with lines to a grid and are additionally displayed as a surface with the color/color palette set.
	Shaded	The individual measured points are displayed as a surface with the corresponding selected color/color palette.
	Wire mesh	The individual measured points are connected to lines in a grid.
	Wire grid with color tones	The individual measured points are connected with lines to a grid and this grid is additionally displayed as a surface with the color/color palette set.
	Samples	The individual measured points are displayed as points or, if selected, as symbols.
	Space curve	The individual measured points are all connected in sequence with lines.
	Space curve with color tones	In addition to the space curve, the line can be colored with different colors to correspond to different y-component value ranges.



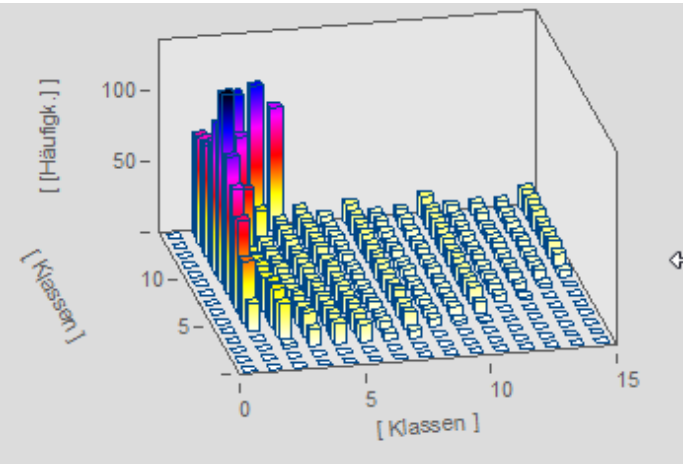
Monocolor space curve



Space curve with color palette

Options for 3D bars

With *Line type: Bars*, three-dimensional bars are displayed, for which additional display parameters are shown.



3D bars

Line type: Bars

Wire mesh color: auto

Surface color: auto

Color scheme Auto

Surface: Wire mesh shaded

Width [%] in x-direction 50 In between

Width [%] in z-direction 50 In between

Bar beginning in y-direction Auto

Settings for 3D-bars

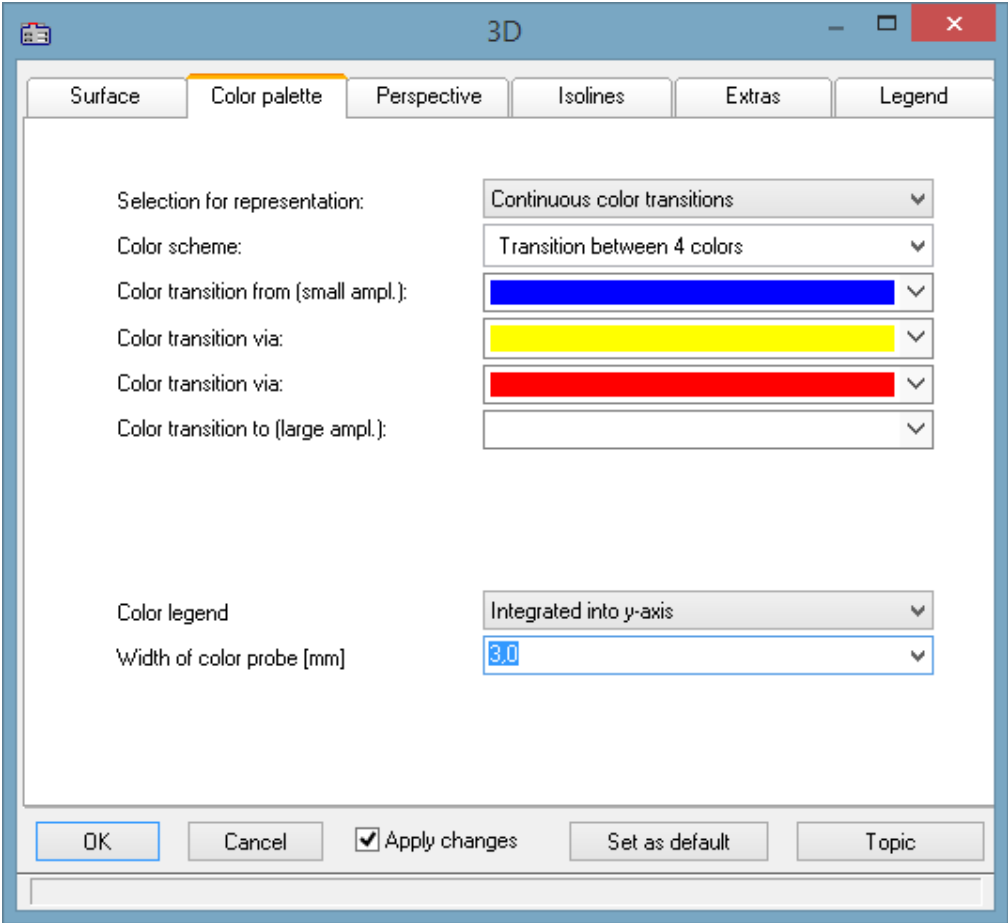
 **Note**

3D-bars can only be set for segmented data. XYZ-plots are not supported.

Settings	Description
Color scheme	The color of the column and the color of the wire mesh (edges) can be given fixed settings. With <i>Surface color</i> set to <i>auto</i> , a color gradient is drawn if the <i>Color scheme</i> is set to <i>auto</i> or <i>With color gradient</i> is set. With <i>Color scheme: One color</i> , the complete column is colored according to its peak value.
Surface	The surfaces of the columns can be displayed in the following ways: • <i>Wire mesh shaded</i> or <i>auto</i>: colored columns, with edges • <i>Shaded</i>: colored columns without edges • <i>Wire mesh</i>: only edges • <i>Wire mesh with color tones</i>: only colored edges
Width (%) in x/y-direction	The width and depth of the bars can be set in terms of a percentage. The position may be arranged as <i>In between</i> , <i>Centered</i> or <i>Aligned</i> .
Bar beginning in y-direction	This sets the height of the bottom surface.

11.6.5.3.2 Color palette

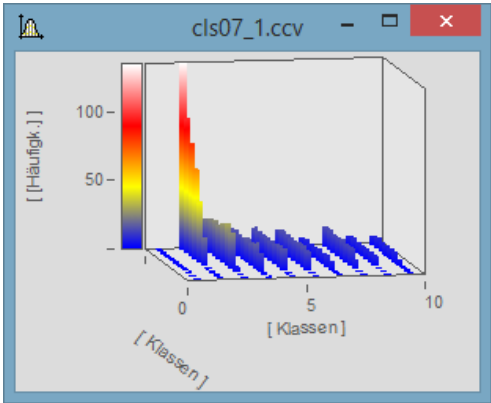
You can make a choice in the *color palette* tab to select a color for the surface. There are two illustration facilities like the continuous color transitions. You can find more information in the chapter [Color map dialog, "General"](#)



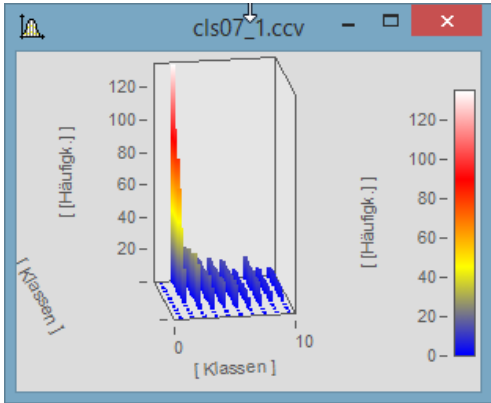
Color legend

The scaling of the colors can be indicated with a *color legend*. The following options are available:

Settings	Description
Auto, No	No color legend
Integrated into y-axis	The Y-axis is supplemented with color columns. Their width can be set.
Separate legend left, right	Separate color column to left or right of the data.



Color legend in y-axis

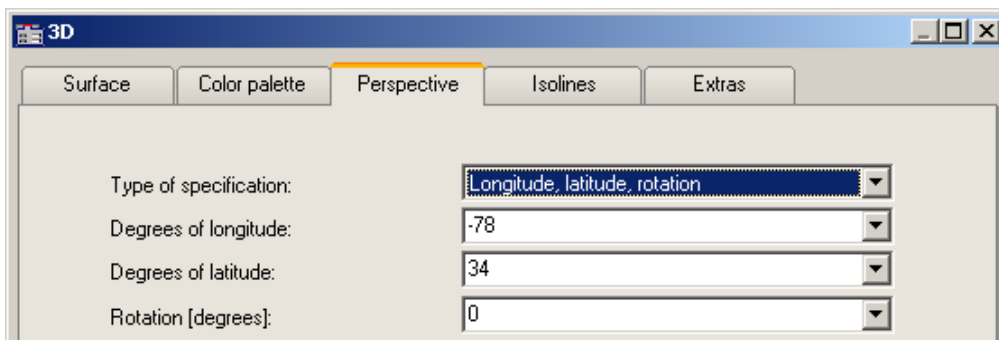


Color legend separate right

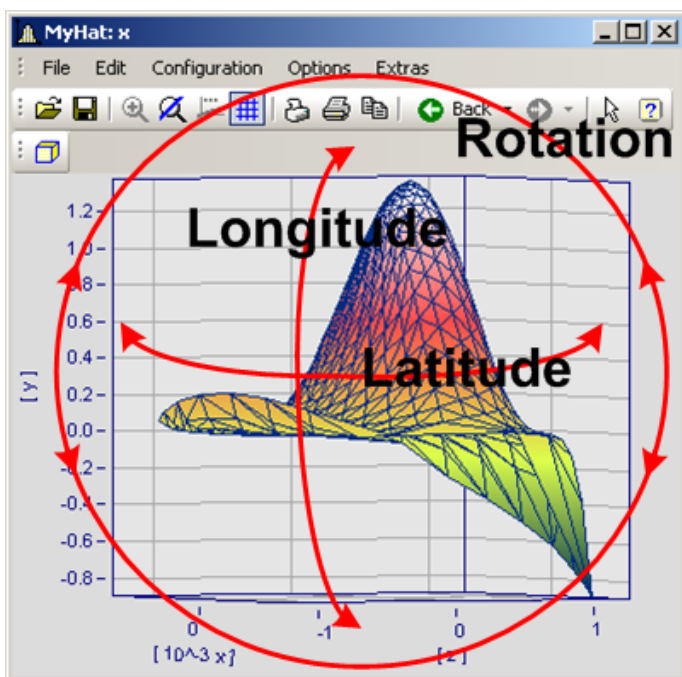
Reference

One way to color the 3D display according to a fourth variable is presented [here](#)⁷⁸⁶.

11.6.5.3.3 Perspective



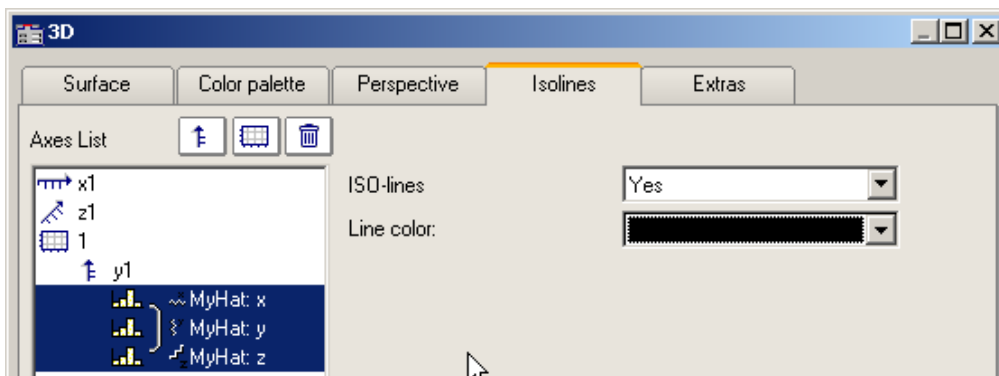
In the dialog page *Perspective*, you can set the viewing perspective as an angle between -180° and $+360^\circ$. There are three ways to make the specification: *Angle of z-axis*, *Longitude* and *latitude* and *Longitude, latitude* and *rotation*.



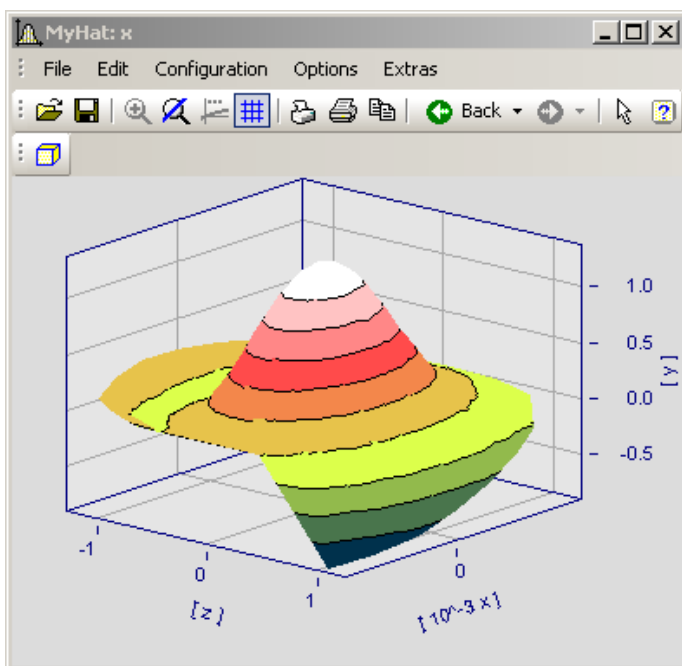
The perspective *Longitude* and *latitude* offers the best mouse control. This view behaves like a survey of the earth from outer space. This is the recommended perspective. If you wish you can additionally specify the rotation.

The three perspective parameters are related as shown below.

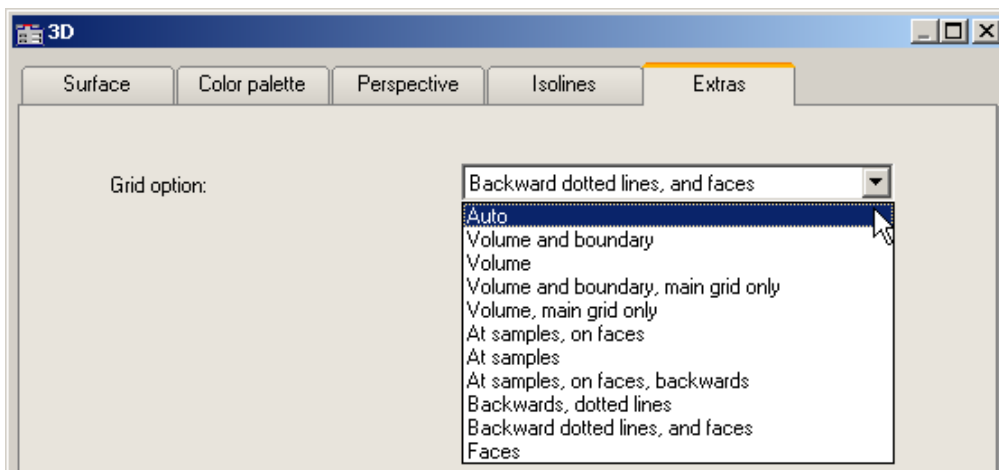
11.6.5.3.4 Isolines



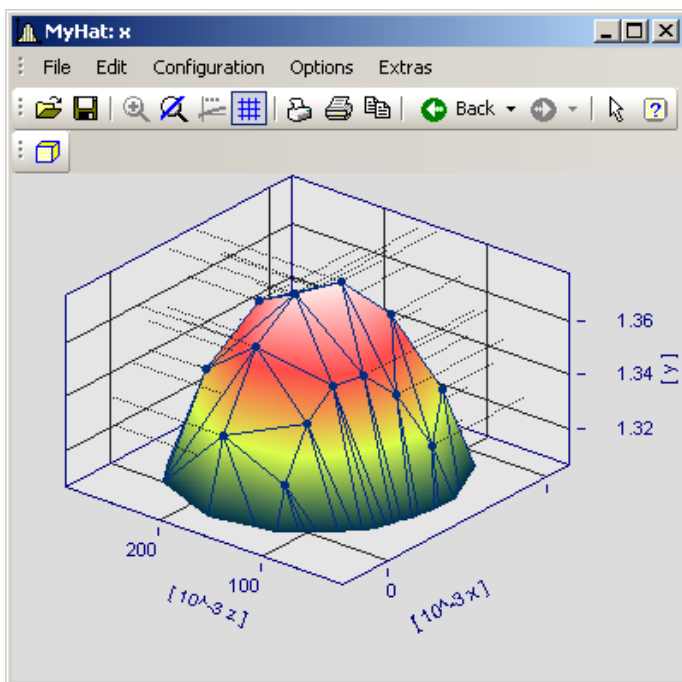
In the dialog page *Isolines* you can select whether to have isolines superimposed on the toned surface, like in color map display. This function can only be activated if graduated colors are set on the [color palette](#) ⁷⁹⁴ page.



11.6.5.3.5 Extras

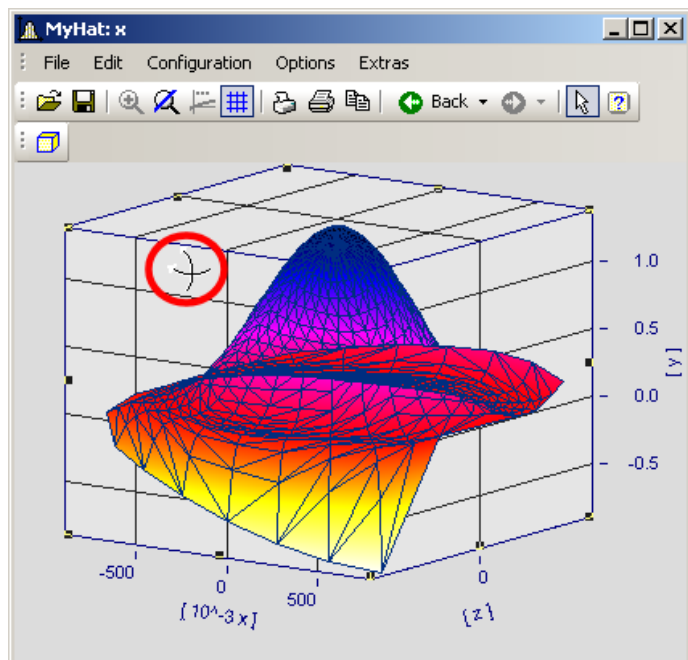


The dialog page *Extras* offers access to various grid options. This grid is then displayed in the space, in addition to the standard coordinate grid, for better legibility of a measurement point's axis values.



11.6.5.3.6 Rotate

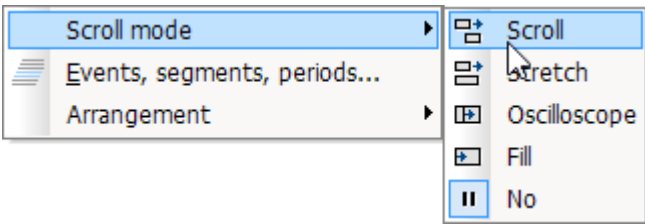
For spatial manipulation of a 3D-display, bring the mouse cursor in the coordinate system, which makes its shape change. If you then click on the mouse and start moving it, the cursor shape changes, as shown in the picture, and the 3D-display can be freely rotated by moving the mouse around.



Cursor for free rotation of a 3D-display

11.6.5.4 Scroll mode

This function is used during a running measurement, when the curve window is continually receiving new measured data to display.



Using the scroll mode, a particular segment of time-domain data can either (*scroll across*) the screen, or the segment itself, beginning from the measurement start, can continually (*grow*) or be stopped for purposes of performing analysis (*No*).

Menu item	Description
 Scroll Mode	The <i>Scroll mode</i> is set for the curve window for a running measurement. The extent of the displayed x-range remains constant; the offset of the x-axis is shifted with each new measurement value.
 Stretch Mode	Click on this button to have the curve window <i>stretch</i> during a measurement. The minimum value on the x-axis remains in effect, while the maximum is automatically set to the end of the waveform. Thus, the extent of the domain displayed increases with each new measured value.
 Oscilloscope-Mode	During a running measurement, the curve window displays a static image which, like an oscilloscope, is redrawn at intervals. The zoom width remains intact.
 Fill	Here, the curve window is filled similarly to Scroll mode. When the right edge is reached, the x-axis skips back by 75%, however. In consequence, the x-axis remains at rest most of the time, giving the user the opportunity to use a measure window ⁸¹⁴ during a running measurement.
 No: End Online mode	Click on this button to halt the <i>Scroll</i> or <i>Stretch</i> mode in a curve window in running measurement; this means the x-axis is frozen in its current state and the signal can then be viewed at rest as long as needed.

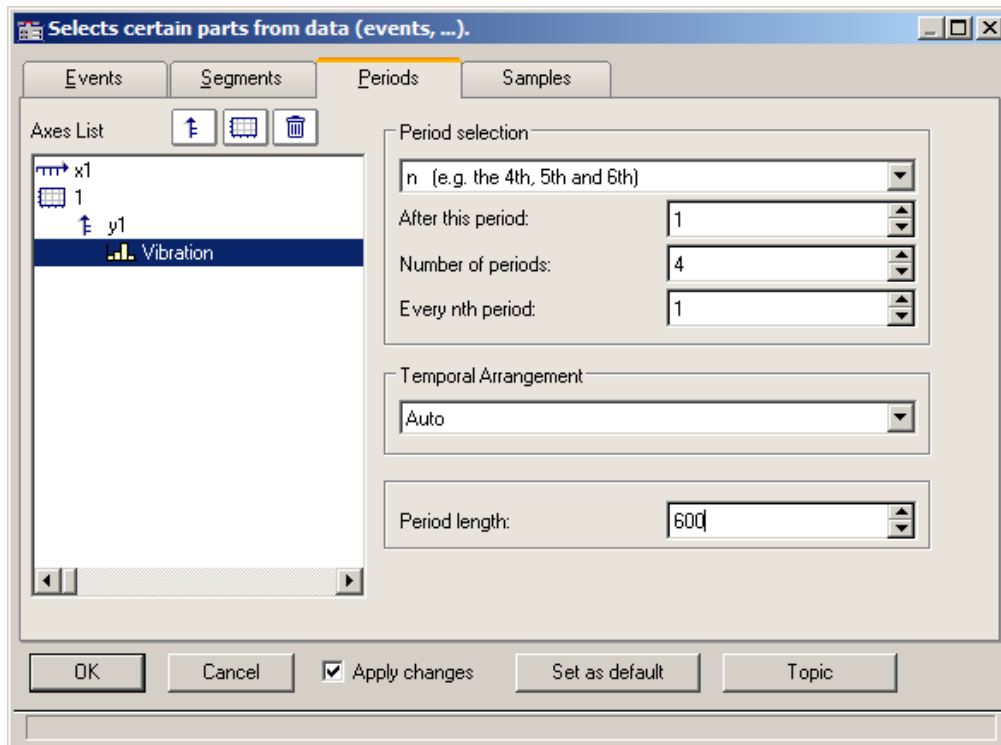
11.6.5.5 Events, segments, periods

Function

With normal waveforms, it is possible to perform a [comparison of signal periods](#)⁸⁰⁶ in a general way where an individual period is selected and compared with others. With **segmented** and **event-based** data, comparison of any desired portions of the data can also be carried out.

Operation

The dialog is called by selecting the menu item *Configuration / Events, Segments, Periods*.



Periods

First select the file card *Periods* from the stack (of three). Then select the channels, which are to be displayed in the period comparison, from the axes list on the left.

Next enter the *Period length* in the box on the bottom right; it is a amount of measurement values ≥ 1 .

Under **Period selection**, you specify in what way the periods are identified. The default entry is *No period comparison*. The option Auto shows all periods. Further options enable targeted display of specific periods. Select "N" periods from the list box, for example, then enter a starting period, an amount of periods, and the increment between successive periods to be compared. The first period in the waveform is denoted by the index 1.

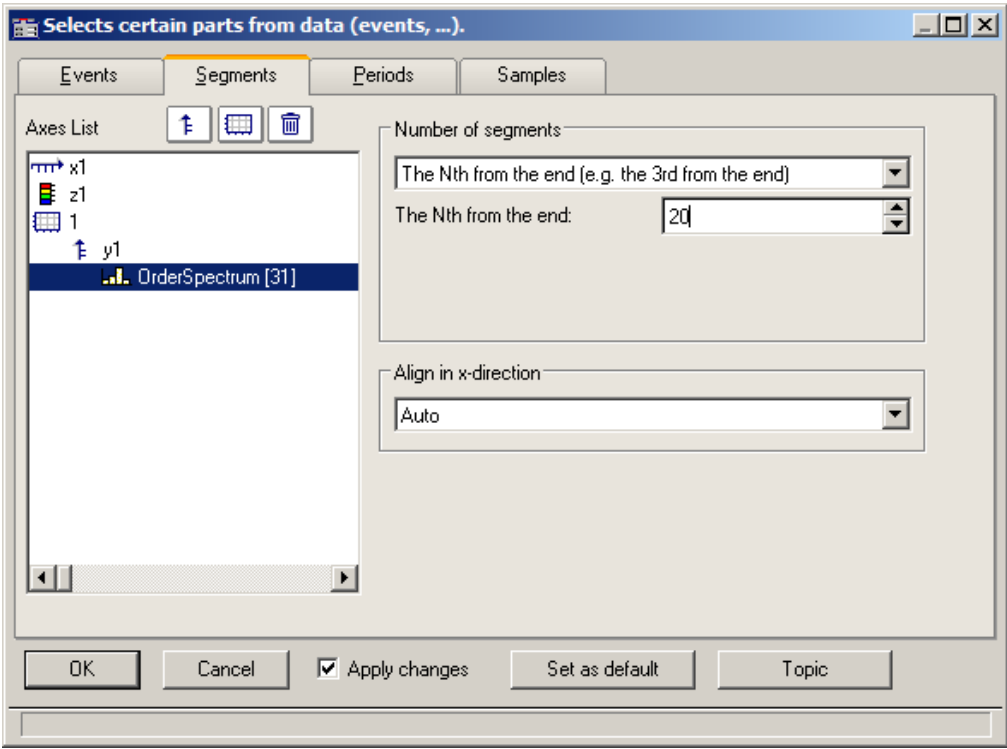
When you have specified the amount, then proceed to the **Temporal Arrangement**:

Settings	Description
Auto	The curve window determines the temporal arrangement. When in doubt, then, use this option.
x0 set to zero	The waveforms' property x0 (their relative trigger time) is set to 0 for the display.
x0 of the first period	The period comparison representation's initial x-coordinate is defined as the x-coordinate of the first measurement value belonging to the first period in the original data set.
x0 of the last period	The period comparison representation's initial x-coordinate is defined as the x-coordinate of the first measurement value belonging to the last period in the original data set.
Each period contains a unique x0	Each period has its own individual x0, which is identical to the x-coordinate of the first measurement value belonging to that period in the original data set.

Segmente

For the display of segmented data. Segmented waveforms are matrices, as generated, for example, by imc STUDIO spectral calculations or class-counting.

Select the *Segments* file card:



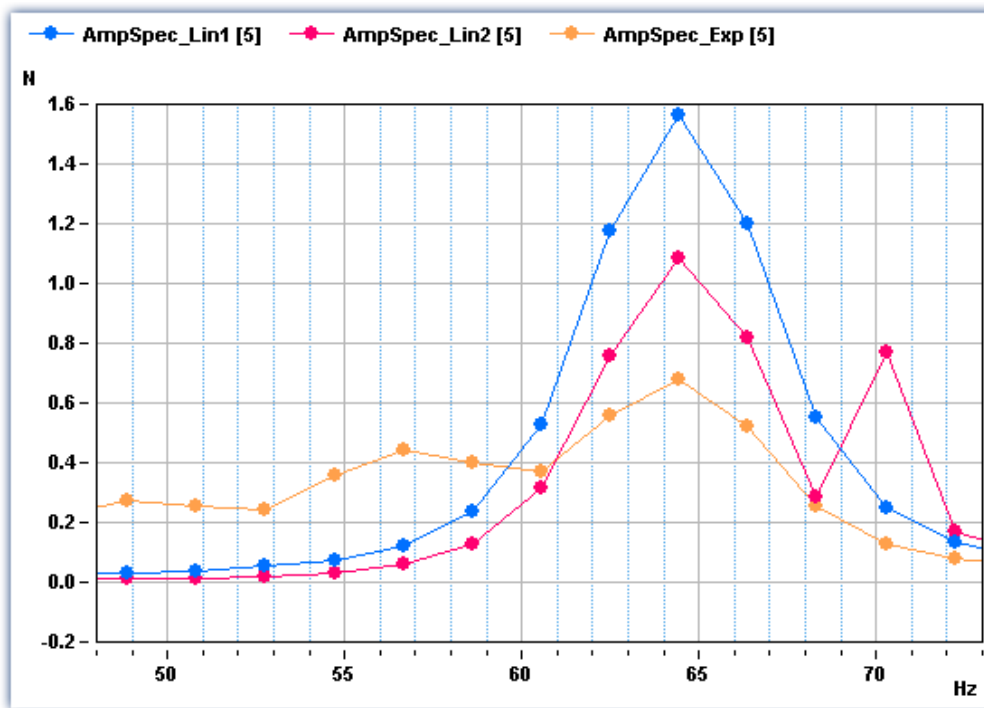
First you must make your selections in the Axes list on the left, as only the selected data indicated there are affected by settings performed with the instruments on the right.

As with *Number of segments*, select the segments to be displayed.

Then select ***Align in x-direction***:

Settings	Description
Auto	When in doubt, choose this option to at least get a view of the segments.
Retain x-coordinate	Each segment receives the same initial x-coordinate, namely, the x0 of the original data set. This is the most appropriate option for handling matrices (Rainflow,...) and multiple FFT's.
Add z to the x-coordinate	Segmented waveforms are provided with z-coordinates, which is determined from z0, dz, and a segment's index. The x-coordinate of a segment's first measurement value is in this mode the sum of x0 and the segment's z-coordinate

Example of segment indices:

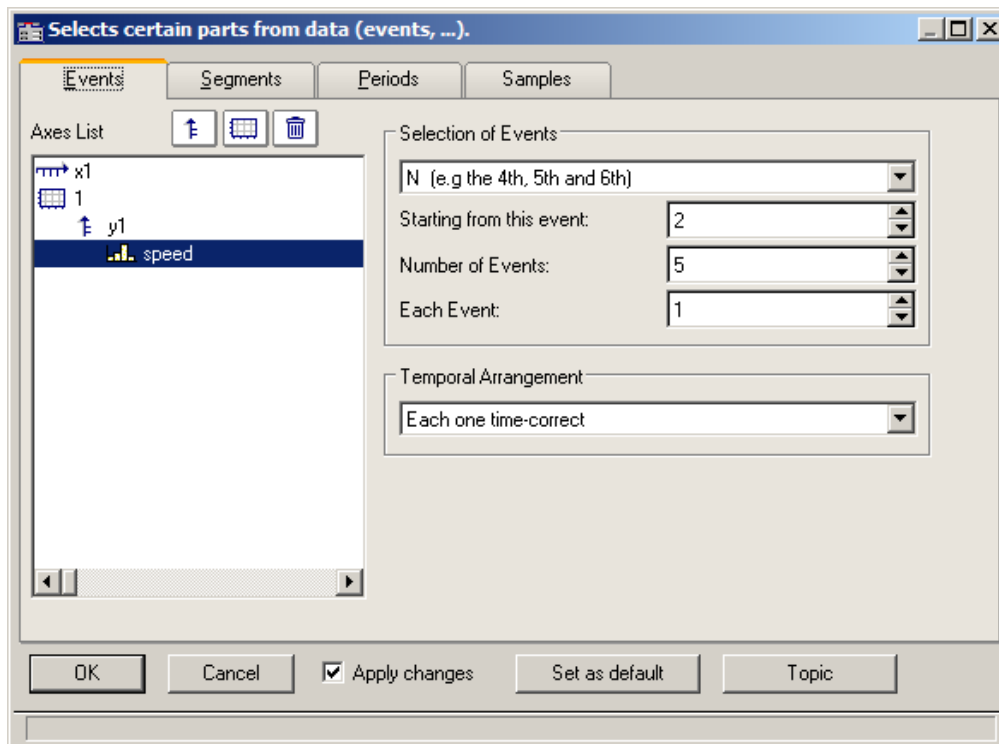


Legend with segment indices

Events

If you have recorded triggered waveforms, make the settings on the Events page. Events are triggered signal segments.

Select the page *Events*:



As before, the desired data sets are to be selected from the *Axes list* at left and the *Selection of Events* from the list box on the upper right.

Select the appropriate **Temporal Arrangement**:

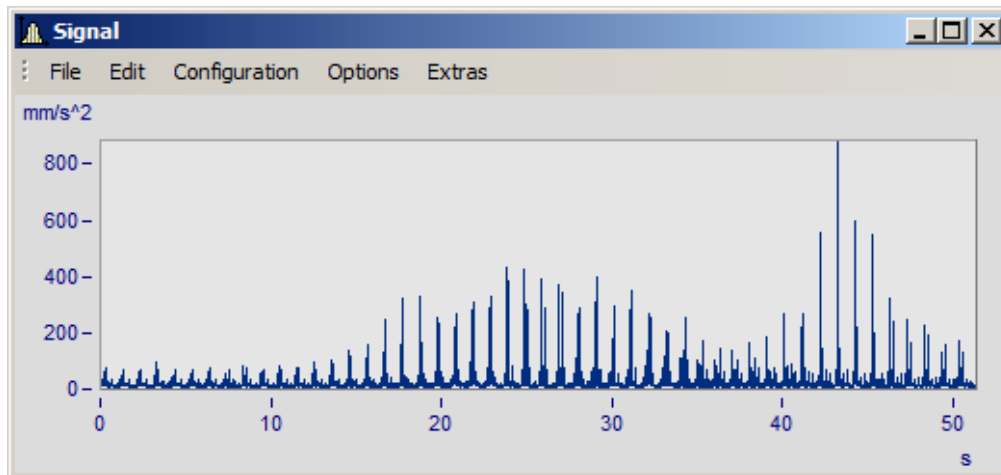
Settings	Description
Auto	This delivers an automatic representation, and is certain to achieve some result. When in doubt, then, use this option.
Each one time-correct	Each event is represented with its genuine time-coordinates. With the curve window's time representation set to "relative", the individual events receive time coordinates which correctly correlate to each other; in the "absolute" time representation the actual trigger instant for each event is indicated in absolute terms. This option is generally the most appropriate for multishot waveforms.
Trigger time of 1 st is valid for all	Caution! Options which rearrange the trigger time can lead to an incorrect temporal representation. The representation may, nonetheless, be of interest for purposes of comparison. This option places the initial time coordinate of every shot at the absolute trigger time of the first event. In relative time representation there is no effect (the time-accuracy is retained); but in absolute-time mode all events are <Shift>ed to reflect the first event's trigger instant.
Trigger time of last is valid for all	As above, but based on the last event's trigger instant.
Trigger time of 1 st displayed is valid for all	As above, but with reference to the first specifically selected event.
Trigger time of last displayed is valid for all	As above, but with reference to the last specifically selected event.
Trigger time difference to 1 st in x0	Caution! Options which rearrange the trigger time can lead to an incorrect temporal representation. The representation may, nonetheless, be of interest for purposes of comparison. This function uses the time difference between the trigger instant of the original data set and the trigger instant of the first selected event as the x0, the initial time coordinate of the period comparison representation. This affects a representation which is made in relative time mode. The event coordinates are <Shift>ed by their genuine trigger times. In absolute time representation no effect is achieved.
Trigger time difference to last in x0	As above, but based on the last event.
Trigger time difference to 1 st displayed in x0	As above, but based on the first specifically selected event.
Trigger time difference to last displayed in x0	As above, but based on the last specifically selected event.

11.6.5.5.1 Period comparison

Periodic signals occur in a variety of applications, such as rotating machinery, where the measured signal almost repeats itself at each revolution.

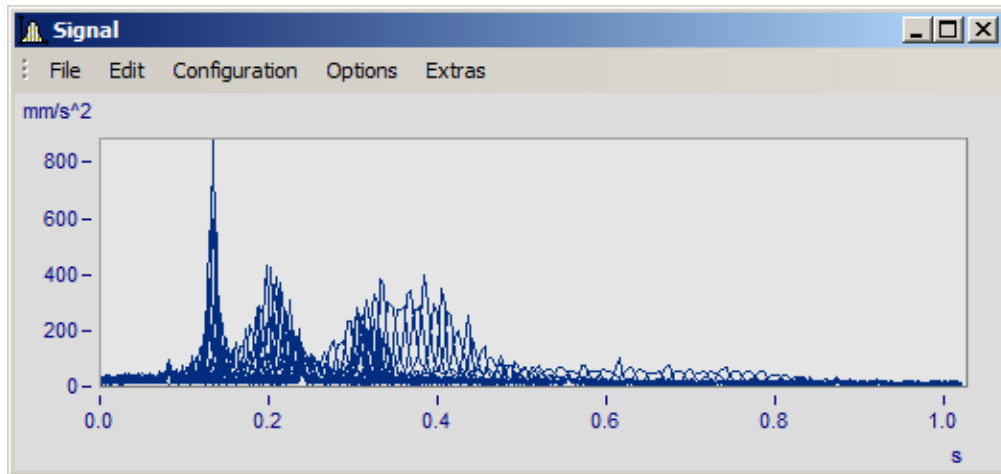
In the curve window's default display mode, entire waveforms appear plotted over time. This means that all periods are drawn in a series. In such a display, it is not always possible to compare individual periods in terms of their maximum values, for instance, or even other properties.

Example:

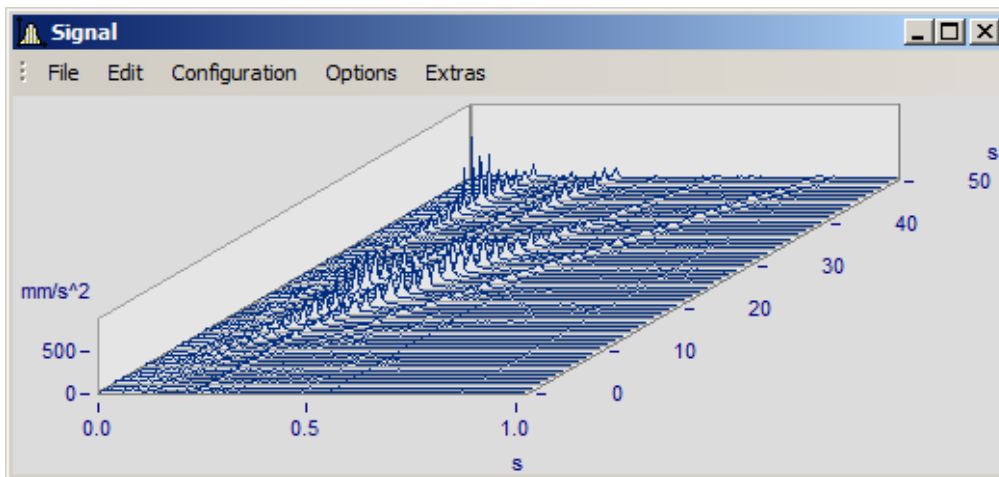


A waveform with 50 periods of 1s duration each is shown

In the period comparison mode, the waveform is divided into its individual periods. The periods are stacked over each other, enabling direct comparison. There are three ways of performing a period comparison:



Default mode. The curve plots of all the periods are drawn superimposed over each other. A dense region of coinciding signal points is visible.

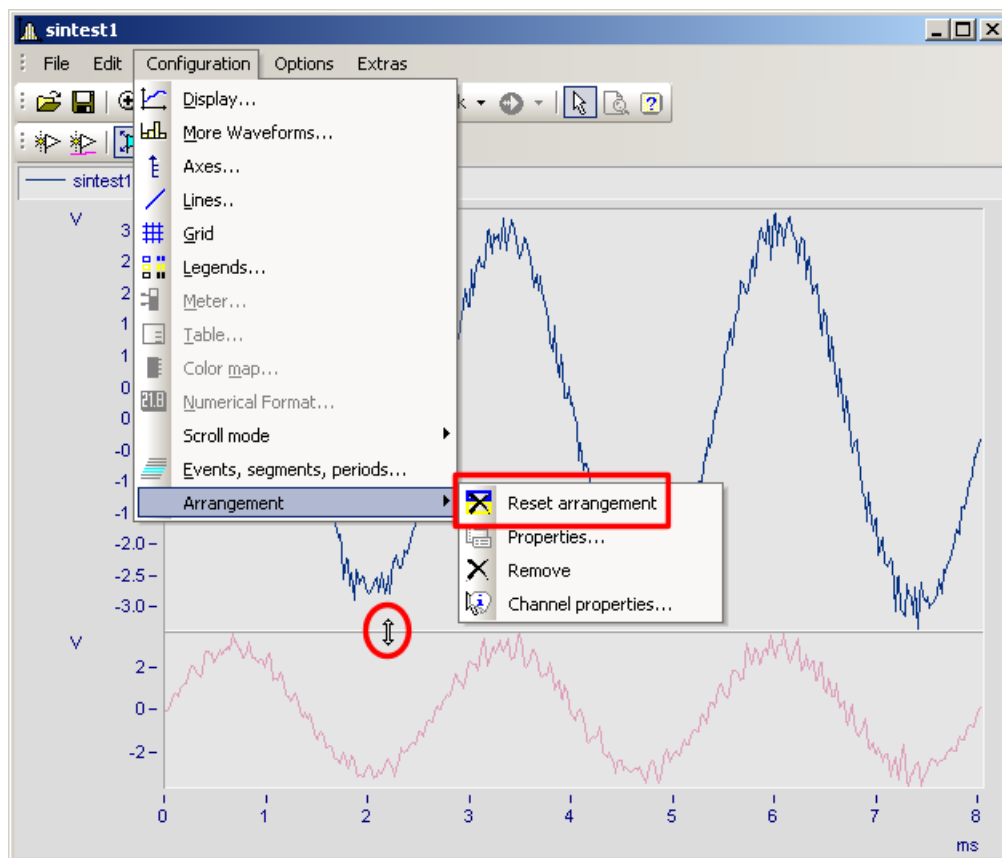


Waterfall diagram: the mountain-range-shaped surfaces clearly illustrate trends emerging across the accumulating signal periods.

In the period comparison mode, the length of a period -in data points- must be known. The amount of points reflect the amount of measured values in each period.

11.6.5.6 Arrangement

This function affects how the coordinate systems are laid out, in terms of the height of the y-axes.



In the menu *Configuration*, you have access to the same items as in the [Select-mode's](#) ⁸³⁰ context menu. Additionally, there is the function *Reset arrangement*, which lets you totally undo any previous changes to the coordinate system arrangement. The relative heights of coordinate systems within a curve window can be changed using the mouse. To do this, move the mouse cursor to the separator between two coordinate systems, so that its shape changes to a vertical double-arrow. Then drag the separator, by holding down the mouse button, up or down as desired to rearrange the coordinate systems' heights.

11.6.5.7 Confirmation bar

Apply changes

When this option is selected, all changes made in the axes list are displayed immediately in the curve window. Thus, the user is always informed about the current appearance of the curve window as resulting from the settings made in the axes list. On the other hand, redrawing the curve window, especially for long waveforms and several curves, can have an adverse effect due to delays.

OK

Click on this button to close the dialog. The current settings are implemented automatically. The same effect can be achieved by closing the window, as is customary in WINDOWS.

Cancel

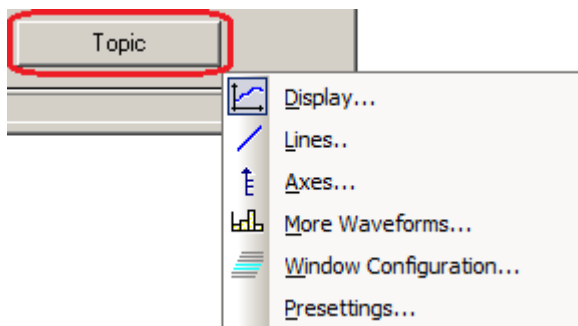
The dialog is closed and any changes made are discarded. This also applies to changes made and displayed in the curve window using the option *Apply changes*.

As presets or Set as default

This saves the currently effective settings as default settings for future curve windows. The settings are saved to the Windows Registry. If you wish to transfer the default settings to a different computer, use the program XConfig, which is located in imc FAMOS' BIN folder.

Topic

From here, you can go directly to the properties dialogs for other objects.



11.6.6 Working with the curve window

Menu item	Description
 Zoom 	Enlarges a section of the curve window.
 Unzoom 	Displays the entire curve.
 Measure 	A measurement value window and measurement cursors for measuring curves are displayed.
 Select-mode 	In this mode, it is possible to select and edit legends, coordinate systems, axes, curves and markers.
Marker 	Proceeds to the list of marker functions.
 Clipboard 	Copies the graphics in the curve window into the Windows clipboard.
 Graphic-export... 	The graphic of the curve window is saved as image or as a PDF file.
 To MS WORD 	The curve window is embedded as an OLE object in Microsoft WORD.
Move 	Proceeds to a submenu listing various curve viewing functions.

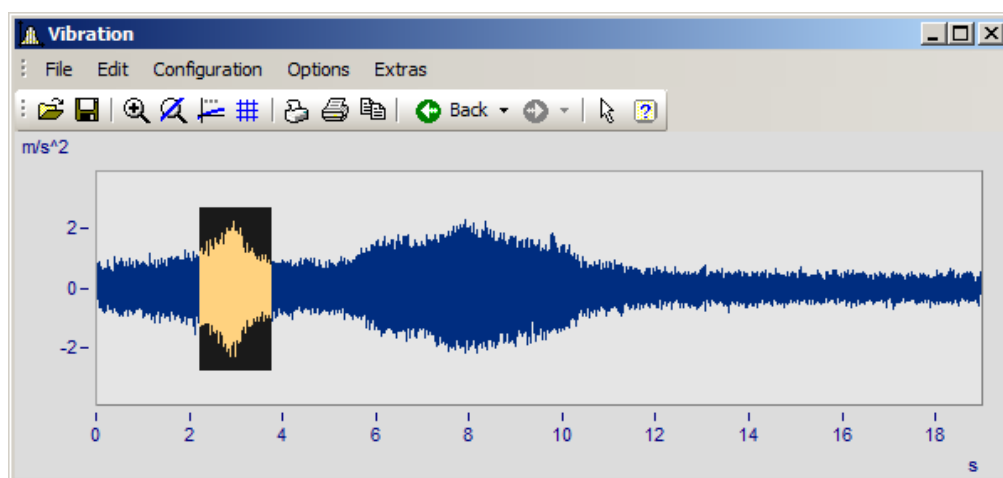
11.6.6.1 Zoom

Function

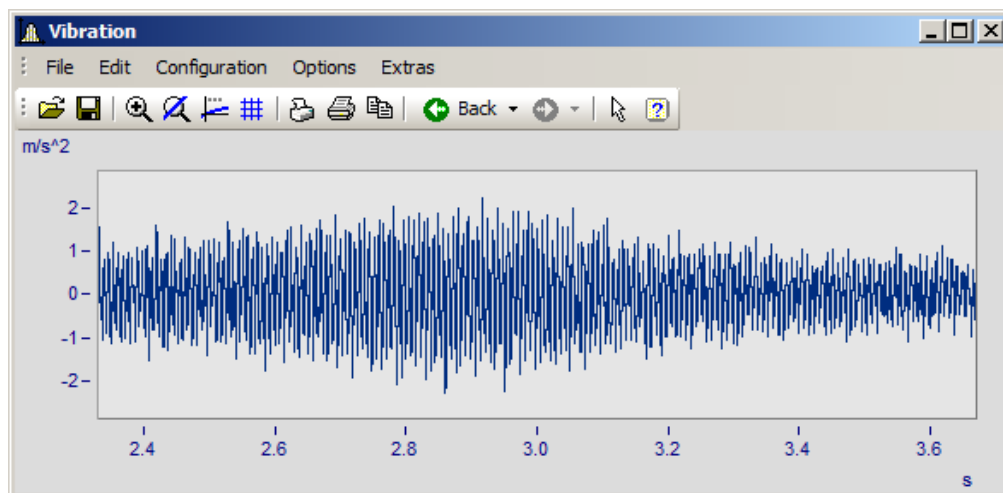
Any region of a displayed curve can be enlarged for viewing (zoomed). Zooming can be done in both x- and y-directions at the same time. The zoomed region is a rectangular area within the coordinate system, with size and position determined by the user.

Mouse Operation

- Select menu option *Edit / Zoom* from the *Scales* menu. The mouse pointer changes its shape to a vertical arrow.
- Draw a rectangle with the mouse around the area you want to see. The selected area is displayed with inverted colors.



- Release the mouse button to enlarge the area displayed in negative and end the zoom mode



Remarks

- The *Zoom* mode can be ended at any time by pressing ESCAPE.
- The mouse pointer must not necessarily be within the coordinate system when selecting the corners of the zoom range.
- A zoomed region can be zoomed further for greater detail.
- Use menu option *Unzoom* to view the entire curve.

- The zoom function can also be used when the *overview* or *measurement value windows* are active.
- After completed zooming, the axes can optionally be labeled with rounded values. In that case, a slightly larger region is usually displayed than was selected.
- Note that the option [use rounding](#)  can be switched off in the *Presettings* menu.
- The zoom function can be regarded as a short-cut for a series of manual axis scaling operations (see menu option [Axes](#) ). If the result is not satisfactory, scale the axes manually (without rounded values).
- Note that zooming in the x-direction often does not produce the desired results, especially for x-axes displayed logarithmically. In rounding, the values on the x-axis will be shown in powers-of-ten as long as the x-range is large enough. Try zooming even more in the x-direction or scaling the x-axis manually. Then you must give consideration to the presettings for the curve window.
- When several curves are displayed in a window, all are zoomed at the same time.
- The measurement value- and overview windows are updated automatically; the positions of the measurement cursors are kept, if feasible.
- Unlimited zooming is not possible:
 - The relative resolution is 10^{-13} .
 - Note that unlimited zooming doesn't make sense, since the representation of the numbers is only precise to approximately 15 decimal places.
 - The zoom region is determined by the resolution of the monitor screen. Enlarge the curve window to make zooming more precise.
 - When the display mode [y-axes stacked](#)  is selected, you can spread the color-negative rectangle of the zoom mode onto one single or several curves. If the rectangle covers only one curve, this curve is zoomed in the x- and y-directions; while for the other curves in the window, the y-range remains intact and the x-range is adapted to the zoomed region. If the rectangle extends over more than one curve, only the x-direction is zoomed; the y-range for each curve remains the same.
 - In waterfall diagrams, only the XY-level can be zoomed.

11.6.6.2 Unzoom

Function

Select the *Unzoom* function to return to the display of the entire waveform after having zoomed. If the curve window had customized scaling and parameters before zooming, those settings are lost after unzooming, since the scaling is then performed automatically.

Mouse Operation

Select menu option *Unzoom* in the *Edit* menu to restore the previous curve window settings.

Remarks

All y-axes present when the function *Unzoom* is selected are scaled automatically, except when they are scaled like the next axis to the left. The x-axis is scaled automatically so that the first curve in the window is displayed completely. Additional curves may extend beyond this range due to a different x-scaling.

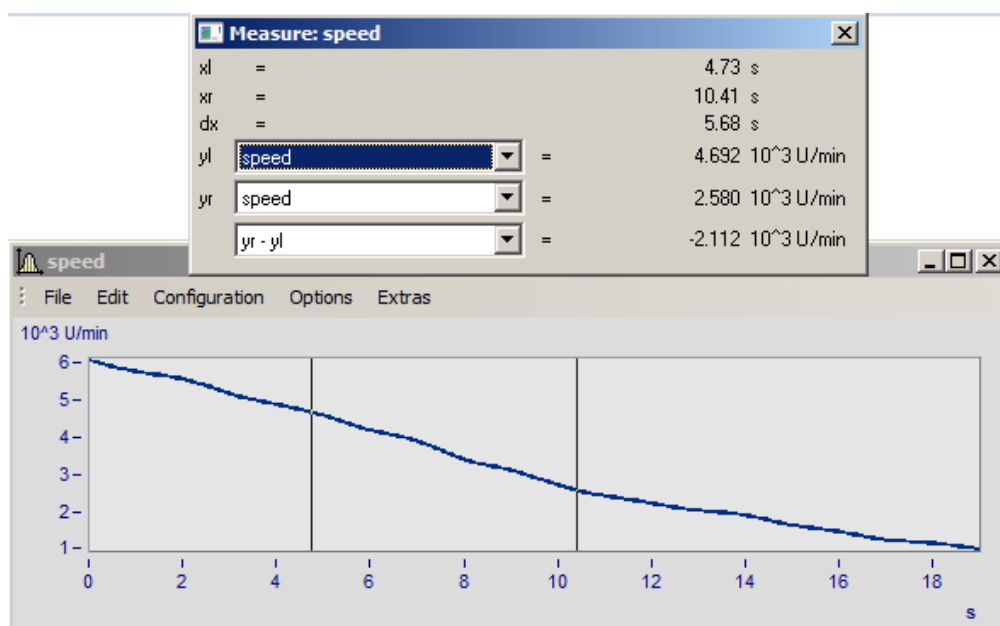
11.6.6.3 Measure

Function

Two independent measurement cursors are available for performing measurements. The x- and y-values of the intersections with the vertical cursor lines are displayed together with the curves in a measurement value window.

Mouse Operation

- The *Measure* window is opened via the *Edit/ Measure* menu or via the toolbar . A measurement value window appears and two measurement cursors are set up in the curve window.
- Move the mouse pointer to over the coordinate system and hold down the left or the right mouse button. The mouse pointer then jumps to the measurement cursor. On the other hand, if you wish to place the measurement cursor at the mouse pointer's position, hold down the keys CTRL or SHIFT while clicking on the mouse button.
- The **left mouse button** controls the **left measurement cursor**; the **right mouse button** the **right measurement cursor**. If you hold down both mouse buttons at the same time, you move both measurement cursors simultaneously.
- When you click in the curve while holding down the SHIFT key, the measurement cursor jumps to the mouse pointer's current x-coordinate.
- In order to place the measurement cursor at any desired location in XY-displays, click one of the mouse buttons while holding down the SHIFT key. The measurement cursor then jumps to the closest point on the curve to the current position.



Remarks

- The crosshairs may cross each other.
- In normal time displays, the horizontal line of a measurement cursor is visible only as long as the curve lies within the coordinate system and is defined for the x-coordinate of the measurement cursor.
- Zooming in XY or locus plots may result in the measurement cursors no longer being visible.
- If linear interpolation between the actual points of a curve is used to draw a graph (points connected by lines), the measurement cursors follow this path.

- To measure the actual points of the curve, select the display mode *Steps* in the *Style* submenu of the [Lines](#) ⁷⁵⁵ menu.
- The horizontal measurement cursor can be switched off in the [Options/ Presettings](#) ⁸⁸⁵ menu. This is sometimes useful when displaying slowly changing data or digital data consisting of steps.
- If no mouse is connected, the mouse pointer only appears whenever the curve window is active at the moment.

11.6.6.3.1 Measure window

Function

The measurement value window appears once menu option *Measure* has been activated in the curve window. Its position can be changed as desired. However, the measurement value window is always in front of the curve window to which it is assigned. The measurement value window shows the x- and y-values for both measurement points and those values' mutual differences; the display of other characteristic values is optional.

Symbol	Value
xl	10.00 s
xr	10.04 s
dx	3.5273E-2 s
yl	4.4876E-5 km/s ²
yr	4.5385E-5 km/s ²
yr - yl	5.0878E-7 km/s ²

With *Create variable from section*, a copy of the area marked with the cursor is copied to the variable list of FAMOS. This corresponds to the context function [Context menu in Measure window](#) ⁸¹⁷.

In the selection lists *yl* and *yr*", a waveform displayed in the curve window can be selected for measurement, which is accomplished using the measurement cursors. The list box at the bottom of the dialog contains the following entries as available choices of calculated values:

Symbol	Description
xl	x-coordinate of left measurement cursor
xr	x-coordinate of right measurement cursor
dx	Difference between the x-coordinates, xr-xl
xr/xl	Quotient of the x-coordinates of the measurement cursors
yl	y-coordinate of the specified curve at the intersection with the left measurement cursor
yr	y-coordinate of the specified curve at the intersection with the right measurement cursor

Calculations	Description
yr-yl	Difference between the y-coordinates
yr/yl	Quotient of the y-coordinates
Slope	Slope between the two measurement points, dy/dx
Slope/Decade	Slope is defined in units per decade, $dy/\lg(xr/xl)$; especially useful for logarithmic display. For example, the slope of a filter is often measured in dB/dec.
1/dx (frequency)	Difference between x-coordinates (xr-xl) in frequency units, e.g. 1/ms
Surface	Integral of the data between the measurement cursors = $\sum(Y_n \cdot x_{\Delta})$ Available only for equidistantly sample waveforms.
Min	Minimum between the measurement cursors. Available only for equidistantly sample waveforms.
Max	Maximum between the measurement cursors. Available only for equidistantly sample waveforms.
Mean	Mean value between the measurement cursors. Available only for equidistantly sample waveforms.
RMS	The root of the mean value of the squares of each value between the cursors ("root-mean-square") is calculated. Available only for equidistantly sample waveforms.
Std-Dev (Standard deviation)	For the data between the measurement cursors, the deviation of each value from the arithmetical mean is squared and then all of these squared values are summed. This value is then divided by the number of these values minus 1; finally, the square root is calculated. Available only for equidistantly sample waveforms.
reversal point x	Reversal point according to the following formula: $yl > yr \rightarrow$ x-position of minimum slope $yl < yr \rightarrow$ x-position of maximum slope Available only for equidistantly sample waveforms.

When using one of the display modes *Date/ time* or *Days/ hours/ minutes*, the time data appears as follows:

xl	=	05/05/2008 15:59:	16 s
xr	=	07/05/2008 06:37:	36 s
dx	=	+ 1 Days 14:38:	20 s
yl	Tacho @ 101	=	5.7498 10 ³ RPM
yr	Tacho @ 101	=	1.2142 10 ³ RPM
yr - yl		=	-4.5356 10 ³ RPM

Measure window with date and time view

xl	=	+ 1 Days 03:48:	35 s
xr	=	+ 1 Days 03:56:	17 s
dx	=	+ 0 Days 00:07:	41 s
yl	Tacho @ 101	=	3.0815 10 ³ RPM
yr	Tacho @ 101	=	3.0707 10 ³ RPM
yr - yl		=	-0.0108 10 ³ RPM

Measure window with relative time

A negative sign before data in *days/hours/minutes* and *seconds* pertains to the entire time declaration, not only to the days.

Positioning the measurement cursors

The left measurement cursor is positioned while holding down the left mouse button, and the right cursor while holding down the right mouse button.

In normal cases the cursors move along the data line; for this reason, the [line type](#)⁷⁵⁶ applies to it.

By means of the contextmenu command "[Cursors freely movable](#)"⁸¹⁸, it is possible to cancel the linkage.

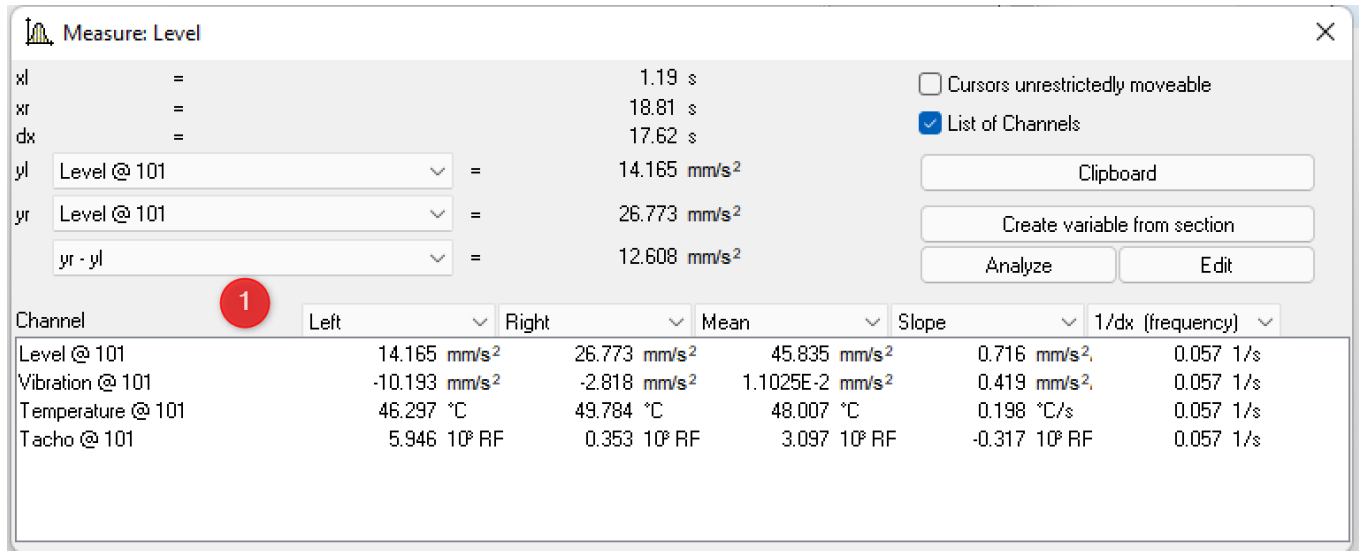
When you click in the curve window while holding down the Shift key, the cursor jumps to the selected x-position. Depending on the setting "[Cursors freely movable](#)"⁸¹⁸, the horizontal measurement line jumps either to the associated amplitude or to the selected position.

Cursors unrestrictedly movable

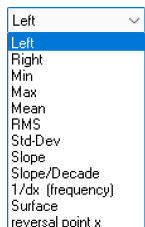
If you select this function, both measurement cursors can be moved freely in the curve window without any linkage to particular curves.

Channels List

When several curves in one window are compared with each other, it can be helpful to see simultaneously the measurement values of as many curves as possible. Do this by selecting the option **List of Channels**, which displays a list of all channels. The measurement value window is enlarged and a list of all y-coordinates at each of both measurement cursors appears, irrespective of which curve the cursors are currently set to move along.



When *List of all channels* is active, the menu item is marked with a check. The List mode is ended by removing the checkmark from the box. The last state remains in operation the next time the measurement window is opened. The channel list's size can be changed. You can highlight list entries to distinguish them visually.



By default, the columns display the content shown in the image 1. Additionally, the standard deviation and the RMS-value of the readings between the measurement cursors can be displayed.

Remarks

The column *right* and *left* above the list is assigned to the measurement cursor which is controlled by the corresponding mouse button.

Clipboard

All measurement values displayed in the measurement value window are copied to the clipboard in text format. This text is then available for use by other applications.

Create variable from section

The area of the dataset enclosed by the vertical lines of the measurement cursors is copied to the **variable list** of imc FAMOS. If the curve window is not opened by FAMOS, e.g. in imc STUDIO, then imc FAMOS starts automatically.

This new dataset is given a **variable name** that depends on the transfer options that are set in the [Options/Transfer options](#) ⁸⁹¹ menu of the curve window. Note that if the name is not changed, the original variable is overwritten if the curve window originates from FAMOS.

If **multiple curves** are displayed in the window, the sections of all curves are transferred to imc FAMOS if the corresponding transfer option is set.



Reference

More about [Transfer options](#) 

Analyse / Edit

This menu item allows signals to be reprocessed, for example, to undergo smoothing or peak clipping. For a detailed description, see the separate chapter [Analyse / Edit](#) .

Remarks

- Each curve window can be in the *Measure* mode, regardless of the status of all other opened curve windows. It is thus possible to conduct measurements with multiple measurement value windows independently of each other. The title bars in the *Measure* windows contain the names of the corresponding curve windows.
- When measuring in all XY-representations (or polar plot), the measurement cursors move somewhat differently than in normal time-dependent representation. The movement of the measurement cursors always follows the parameter of the XY-display. Moving the mouse to the right or upward moves the corresponding measurement cursor in the direction of the increasing parameter.
- The measurement cursor speed in XY and locus plots varies depending on the point density of the curve. The measurement cursors cannot be moved in arbitrarily fine increments between samples. Zooming too strongly, therefore, can cause the measurement cursors to jump between points.

11.6.6.3.2 Context menu in Measure window

Move the mouse cursor to the measurement value window and right-click the mouse to prompt the following context menu:

Cursors unrestrictedly movable

See [here](#) 

Send Curve Section to imc FAMOS

See [here](#) 

Clipboard

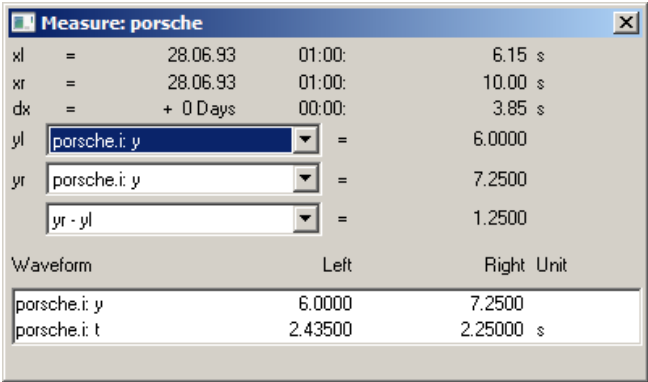
See [here](#) 

Channel List

See [here](#) 

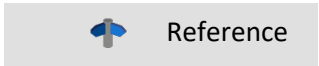
Expand List

When *List of all channels* is active, the menu item *Expand the list* appears. By selecting this option, separate measurement values for all XY-waveform components -if any such waveforms are present- are displayed in the list. In polar plots, the measurement values of the magnitude and phase are displayed. This option can also be activated by double-clicking on the pertinent entry in the list. Double-clicking on a component of the waveform also deactivates the option; and only the selected component is retained as a separate entry in the list.



Place marker with left click; Place marker with right click

Set a marker at the measurement cursor positions. Subsequently, a dialog for [marker settings](#)⁸⁴³ appears.



More about "[Markers](#)"⁸³² and "[Marker Definition](#)"⁸⁴³

Measurement value file

Append to measurement value file

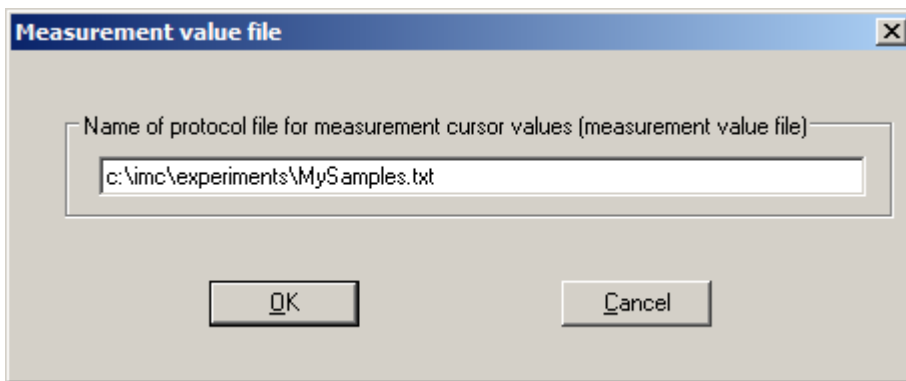
This menu item appends the measurement values to the file of measurement values as text. The name (along with the directory path) is set in the menu under *Measurement value file name*.

The file is a normal, text file in ASCII without formatting, so that it can be edited by any word processing program. By repeatedly calling this menu item for a variety of cursor positions, it is possible to generate a list of landmark points. If the file doesn't exist, it is first created.

Note that MS-Windows applications misread *umlauts* in these files. The OEM format is used for writing to these files. For this reason, use programs (such as word processors) not running under MS-Windows to analyze these files.

Measurement value filename

This menu item calls a little dialog in which the name of the measurement data file can be edited. The setting for this filename is valid globally for all measurement value windows.



Note

As of imc FAMOS 7.0 / imc STUDIO 5.0R3, these two entries for the measurement value file are no longer shown.

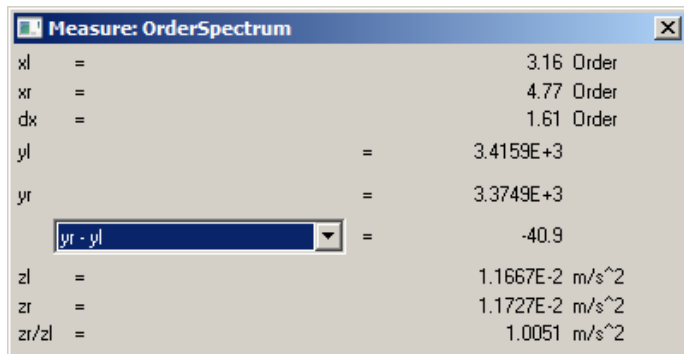
By means of an entry in the Registry, it can be made visible again:

`Computer\HKEY_CURRENT_USER\Software\imc Measurement and Control\Default\CurveDataManager\Curves`

If it is not already present, enter as a New String the text "EnableMeasFile". Set its content to 1.

11.6.6.3.3 Performing measurements on color maps

In this display mode, the measurement cursor can be moved freely over the entire plane of the coordinate system. x- and y-coordinates are arranged horizontally or vertically. The z-coordinate is indicated along the "color axis", which extends outwards from the plane of the monitor.



11.6.6.3.4 Analyze / Edit

Function

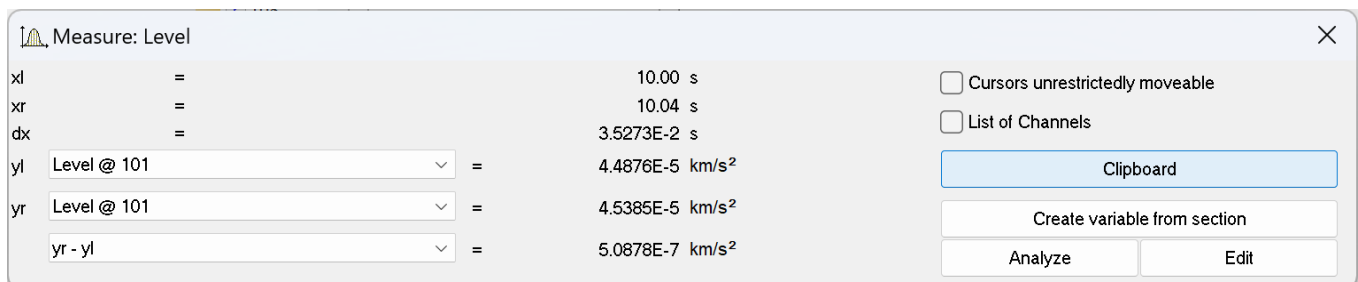
Here it is possible to edit, for instance, to smooth clip peaks from, simple signals. Thus, invalid readings can be removed from a signal and replaced with plausible ones, or distorted readings can be corrected, for example, by eliminating offsets or drifts.

These functionalities are only available off-line (in imc FAMOS).

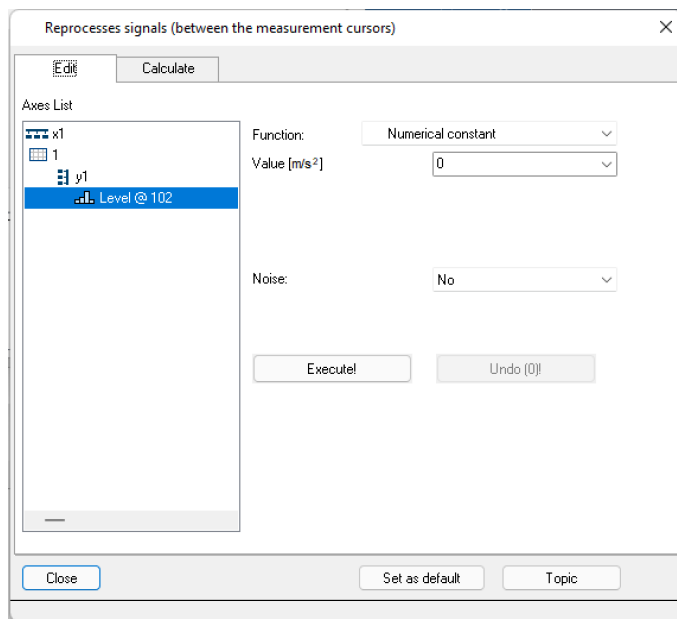
Operation

A channel is displayed in a curve window. An area is defined using the measurement cursors, e.g. the area with the interfering peak to be removed. There are two tabs in the Post-process signals dialogue: Edit and Calculate.

First, the Measurement value window is opened:



Select *Reprocessing signals....* via the button or the context menu. The following dialog is called:



Tab: Edit

In the dialog called from the menu item *Reprocessing signals...*, a mathematics function is selected for applying to the staked-out region. When this function is performed, the channel's measured value are replaced by newly calculated values.

Select in the left portion of the dialog the channels in the curve window to which the function is to be applied. On the right side, select the calculational function and the function parameter.

Use the measurement cursors to carefully delineate the desired signal segment.

When the *Execute!* button is pressed, the function is applied to the respective signal segment of the selected channels, which is delineated by the cursors.

The button *Undo* lets you reverse the last action taken.

Remarks

- Only simple data types can be handled in this way. These are equidistant data (constant sampling time, $dx = \text{constant}$) no having any additional structural pattern. In particular, it is not possible to process segmented waveforms, data with events, XY-data, data already subjected to Transitional Recording, Time-stamp-ASCII data, texts, and complex data.
- The data's original format is retained. This can mean that the value range is limited and stays that way, and that the math function can thus only be performed in approximation. For instance, if a channel's value range is $-10V \dots +10V$ with 2 Byte integer format, no value $>10V$ can be displayed. Thus, an overdriven spline can be truncated at $10V$. And it isn't possible to set the signal to $11V$, instead it stays limited to $10.0V$.
- **Undo:** Up to 30 actions can be reversed (undone).
- If invalid function parameters are specified, the function cannot be carried out. Instead, an error message appears.
- **The interval for determining boundary values may not be less than 0!**
- The dialog for reprocessing signals can only be opened for one curve window at a time.
- The dialog for reprocessing signals has no *Cancel* button. If the dialog is exited, changes made to the channels are retained. The *Undo* button is the only way to reverse changes that have been made.

- See the discussion of "[Confirmation bar](#)"⁸⁰⁹ for more information on the operation of the *Set as default* button.

Description of the functions

The following is a description of all functions in the list *Function*. The functions are ordered by subject.

The subjects appear aligned to the left edge, e.g. *Adapts curve*. The functions themselves appear in the list with an indent, e.g. *Constant level based on right boundary*. Only functions can be selected.

Subjects

- *Adapt curve*: A connecting line, either a straight line or a spline, is inserted.
- *Re-define curve*: The curve is re-defined; for instance, as a specified constant level.
- *Smoothing*: e.g. low-pass filtering
- *Edit signal*: e.g. value range limiting
- *Edit trend*: e.g. add offset, eliminate trends, high-pass filtering
- *Noise*: Add noise to produce readings which "look convincing".

Please note also the description of the parameters [Noise](#)⁸²⁴ and [Boundary interval](#)⁸²⁵.

Adapt curve	Description
Constant level based on right boundary	The mean value of the right boundary interval (which see below) is taken. This value replaces all original readings in the delineated region. If desired, noise (which see below) can be added.
Constant level based on left boundary	The mean value of the left boundary interval (which see below) is taken. This value replaces all original readings in the delineated region. If desired, noise (which see below) can be added.
Connecting line	The respective mean values of the right and left boundary intervals are taken. These mean values are used as the boundary values for a straight line replacing the signal curve between the cursors (linear interpolation). If desired, noise (which see below) can be added.
Cubic spline	The respective mean values and average slopes of the right and left boundary intervals are taken. The mean values are used as the boundary values of a replacement curve between the cursors, and the slopes are used to form a cubic spline, which is a polynomial whose slope at the boundaries matches the slope of the original curve.
Re-define curve	Description
Constant numerical value	The delimited region can be replaced with a fixed numerical value to be specified. If desired, noise (which see below) can be added.
Smoothing	Description
Low-pass	Smoothing: A digital filter is applied to the readings within the delimited region, a low-pass filter of Butterworth characteristic. The filter order and cutoff frequency can be specified. The order can be set to between 1 and 10. The pop-down list presents suggestions for the cutoff frequency. The cutoff frequency must be substantially less than half of the sampling frequency for the data. The function works in a manner similar to imc FAMOS' FILTTP() function. Note that the filter has a transient oscillation and always produces a delay.
Edit signal	Description
Clip signal above	All readings in the region delineated are clipped to keep the curve below a specified maximum value.
Clip signal below	All readings in the region delineated are clipped to keep the curve above a specified maximum value.

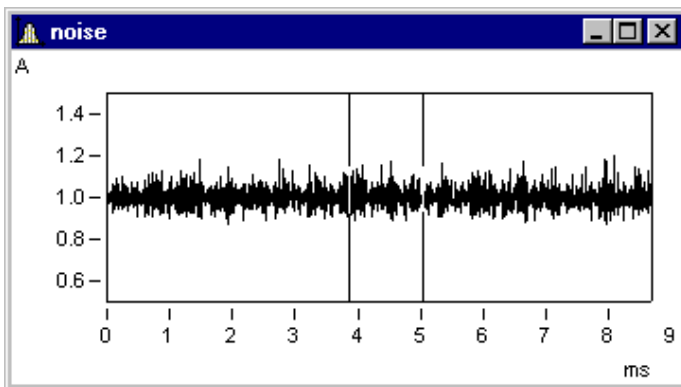
Edit trend	Description
Add a numerical constant	Adds an offset. A fixed numerical value is added to all readings in the delineated region. This means adding an offset. To subtract, enter a negative value.
Multiplication	A fixed numerical value is multiplied to all the measured values within the staked-out interval.
High-pass	Eliminates offsets and slow drifts: a digital filter is applied to the readings within the delineated region, a high-pass filter of Butterworth-characteristic. The order can be set to between 1 and 10. The pop-down list presents suggestions for the cutoff frequency. The cutoff frequency must be substantially less than half of the sampling frequency for the data. The function works in a manner similar to imc FAMOS' FILTHP() function. Note that the filter has a transient oscillation.
Add a slope	Trend correction: a slope is added to all readings within the delineated region. The slope is a linearly increasing function. To define the slope, and amplitude is specified for the initial point (y-coordinate at left boundary of delineated region) and the amplitude at the end (y-coordinate at right boundary of delineated region). The values specified define the slope, which is then added to the readings.
Subtract linear approximation	Trend correction: a linear approximation is formed from all the readings in the delineated region. The least-squares method is used for this. The approximation The approximation is later subtracted from the curve in the delineated region.
Fade in/out	The signal can be faded in to or out from zero within the delineated region. For this purpose, the signal is multiplied with a ramp. Fading in is the typical case, from 0% to 100%. Fading out from 100% to 0% is another typical action which is possible. Both percentages can be specified. An entry of 100% corresponds to a factor of 1.0. These parameters appear in the dialog as Start, component [%] and End, component [%] as the values at the left and right boundaries, respectively. In between, the values are interpolated linearly. The two percentages supplied may not be either 0.0 or 100.0, but can take any other real value in between.
Noise	Description
Add boundary noise	The respective trend is eliminated from each of the boundary interval's signals. This is accomplished by subtracting the linear approximation calculated. What remains is interpreted as noise. This noise, cyclically repeated, is added to the readings in the delineated region. In the process, the noise at both ends merges by means of linear weighting over the width of the delineated region. The values within the boundary interval naturally remain unchanged.
Add equally distributed noise	Equally distributed noise of specified amplitude is added to the readings within the delineated region. If the amplitude is 1.0, random numbers in the range -1.0 ... +1.0 are generated and added.
Add Gaussian-distributed noise	A noise signal with Gaussian distribution (normal distribution), having a specified RMS-value, is added to the readings within the delineated region.

Boundary interval

Many functions use the boundary values at the edges of the delineated region; for instance, all values within the delineated region are bridged by a connecting line or spline.

Now, it would be possible to simply use the boundary value itself for this purpose. However, this would not take the noise, which frequently is present, into consideration. The result of the function would be dependent on the noisy value at the region's particular edge, and this is largely a matter of chance. Therefore it would seem reasonable to take the mean of a small interval around the boundary, in order to reduce the influence of random noise. This interval should be, on the one hand, relatively small, but on the other hand still large enough to suppress the noise.

In the example below, the delineated region is to be replaced by a straight line (between the boundary values):



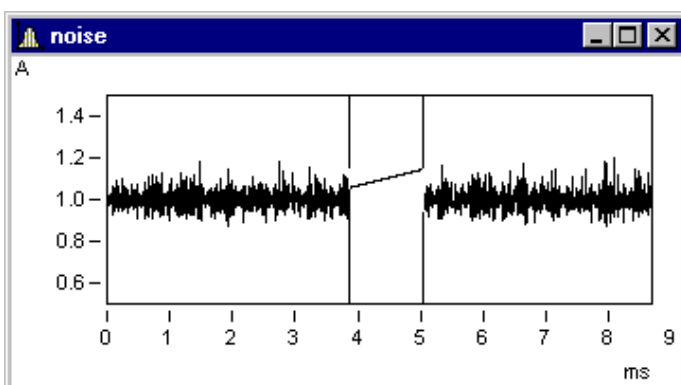
In the dialog *Reprocessing signals...*, the following settings are made.

Function:

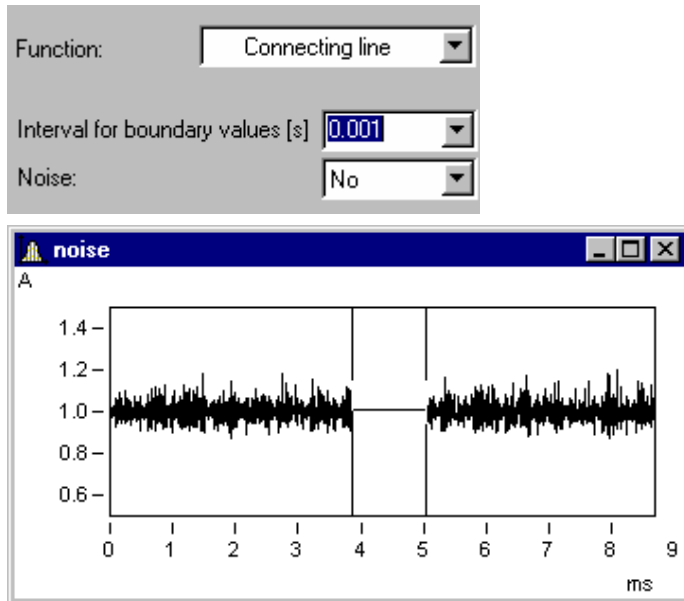
Interval for boundary values [s]

Noise:

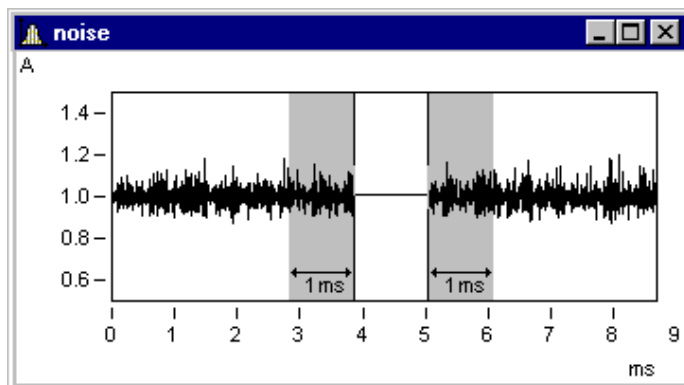
The following connecting line results:



This isn't what one would expect. By chance, the values at the edges (under the cursors) are random peaks. Setting a boundary interval of 1ms (0.001s !) produces proper results:



The boundary interval always is located outside of the delineated region:
Here, the boundary intervals are shaded in grey.



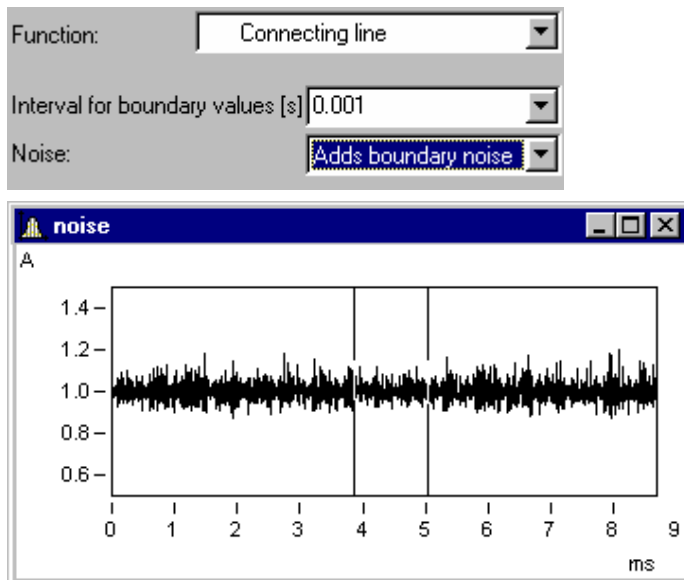
Remarks

If 0.0 is taken as the boundary interval width, the value at the exact edge is used. Thus, the minimum width of the boundary interval is 1 reading.

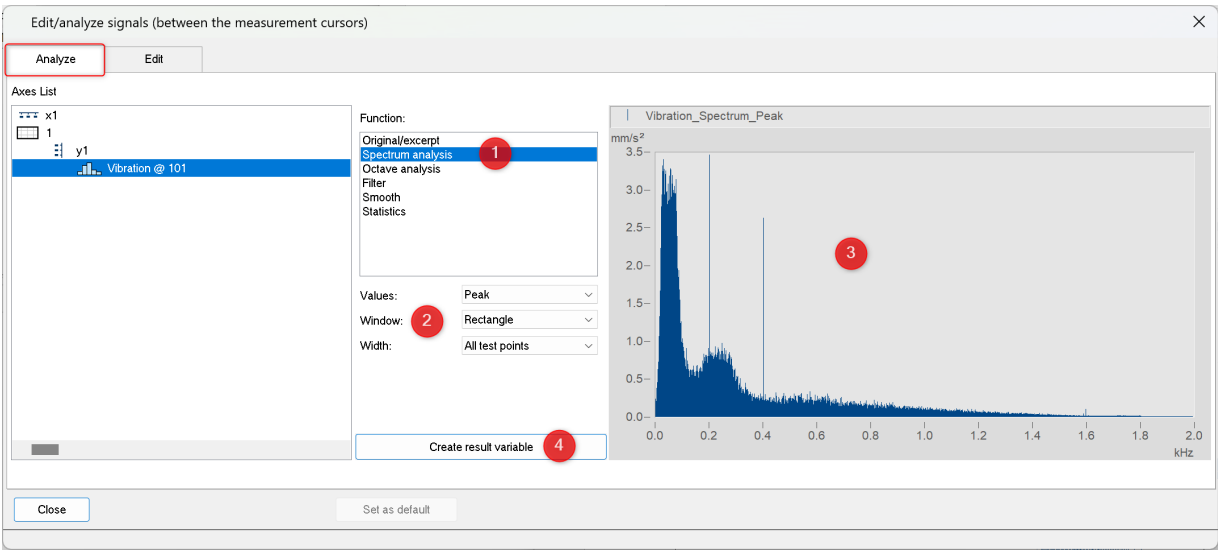
Adding noise

To the sample signal above, it is now possible to add noise. Toward this end, the noise in each of the boundary intervals is determined. This noise, cyclically repeated, is then added to the delineated region. Between the noise signal on the left and that on the right side, a smooth transition is accomplished by linearly increasing weighting along the width of the region.

With the following setting, noise from the boundary intervals is added to the delineated region. This produces a realistic signal appearance:



Tab: Analyze



Under the tab *Analyze* as well, a math function is to be selected, whose results are displayed directly in the dialog. The data to which it is applied are defined using the measurement cursors. The respective parameters pertaining to the function selected are displayed. The calculation is performed each time the parameters are changed.

➤ Description of the functions

Below are descriptions of all functions belonging to the *Function* list.

Original/excerpt	Description
no additional parameters	The range defined by the measurement cursors is displayed.
Spectrum analysis	Description
Values	Meaning of the lines' amplitudes: Peak values (<i>Peak</i>), root mean square (<i>RMS</i>), square of the RMS-spectrum (<i>Power</i>), power density = square of the RMS-spectrum divided by the frequency line distance (<i>PSD</i>)
Window	Window function for the FFT: <i>Rectangle</i> , <i>Hamming</i> , <i>Hanning</i> , <i>Blackman</i> , <i>BlackmanHarris</i> , <i>FlatTop</i>
Width	<i>All test points</i> of the region defined by the measurement cursors-> one spectrum or plot of multiple spectra displayed as a waterfall or color map graph. <i>Intervals (values)</i> : count of values for the FFT <i>Overlap</i> : percentual overlap of the values used in calculating the FFT <i>View</i> : <i>Color map</i> or <i>Waterfall</i>

Octave analysis	Description
Frequency width	Distance between frequency lines: <i>Octave, Third, 1/12 Octave, 1/24 Octave</i>
Preprocessing	<i>None, 1x integration, 2x integration, 1x derivative, 2x derivative</i>
Averaging interval	<i>None, FAST, SLOW, Impulse, Peak, Specified time</i>
Frequency weighting	Weighting: <i>Linear, A, B, C, D</i>
Lower/Higher Cutoff freq. [Hz]	Frequency range to be evaluated

Filter	Description
<i>Low-, High-, Band-pass, Band-stop</i>	Filter parameters with characteristics, cutoff frequency (or frequencies) and order

Smooth	Description
<i>Smooth</i>	Moving weighted average (triangle) with count of values in the averaging window (<i>Interval (values)</i>). Matches the FAMOS function Smo()
<i>Median</i>	Moving median with count of values (<i>Interval (values)</i>). Matches the FAMOS function Median()
<i>Moving mean value</i>	Moving average with count of values in the averaging window (<i>Interval (values)</i>) and reduction. Matches the FAMOS function MvMean()

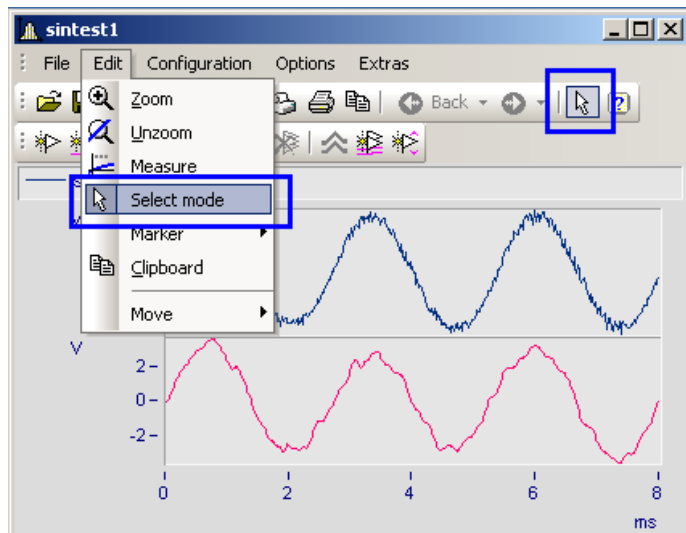
Statistics	Description
<i>Histogram</i>	Distribution of the amplitudes across the measurement range with the specified <i>Count</i> . <i>Width</i> in Y-units; 0 = (Max-Min)/ <i>Count</i>

Create result variable

Copies the calculated and displayed results to FAMOS' Variables list. See also "[Create variable from section](#)"

11.6.6.4 Select mode

The *Select mode* enables the user to select legends, coordinate systems, axes, lines and markers by means of the mouse. *Select mode* can be activated/deactivated either via the *Edit* menu or the toolbar. Alternatively, it is possible to activate/deactivate *Select mode* by double-clicking over a region where there are no objects, for instance in the middle of a coordinate system.



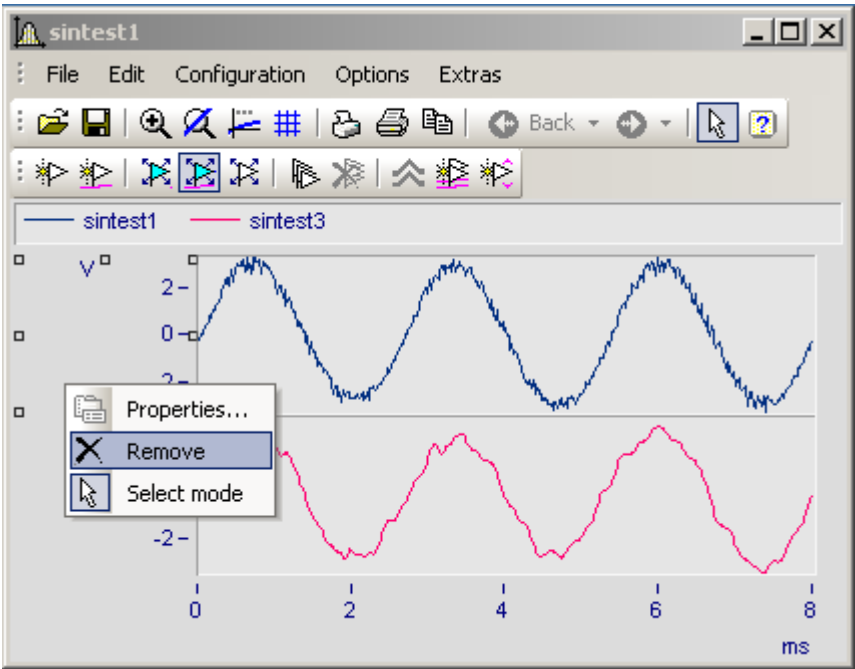
If you are in *Select mode*, the current object is highlighted. This is indicated by the presence of small empty squares at the object's edges. Any lines not selected appear dulled in color.

Multiple selection is possible by making a selection using the left mouse button while holding down the CTRL-key. However it is only possible to join up objects all of the same type (lines or axes) in multiple selection.

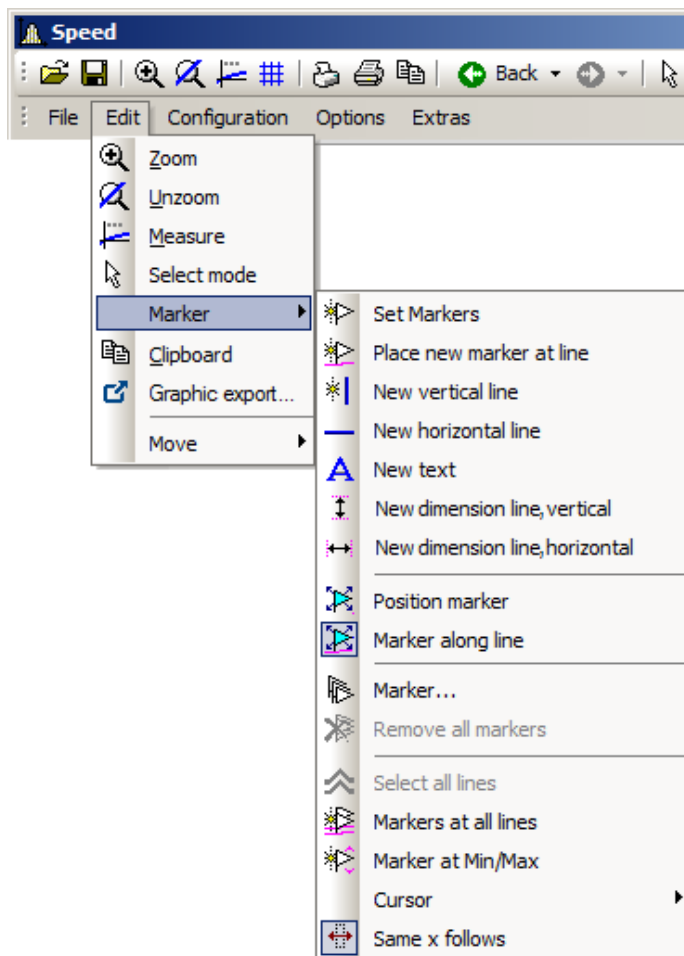
Among other things, the context menu contains the function *Delete*. This is only available if objects which can be deleted are available, such as a second y-axis. If such an object is deleted, any associated objects are also deleted. In the example, the lines in the deleted y-axis would also be deleted.

Example

You select the upper y-axis and deleted it. In the process, the sintest1 lines with which it is associated are also deleted.



11.6.6.5 Markers



Function

A marker is a designated point in a curve window, to which a text can be assigned. The point itself is not visible, but a line with an arrow may point from the text to the point. The text can be enclosed in a frame; various characteristics can be established for the lines and font for each marker, such as color and size.

Defaults

All marker types are described below. Each type has particular default settings which are applied when the marker is inserted.

You can customize these individually by modifying the parameters for the selected marker type.

Defaults

Standard marker

Vertical line

Horizontal line

Text

Vertical dimension line

Horizontal dimension line

Order line

Harmonics Cursor

Min-Marker

Max-Marker

Marker at all lines, y

Marker at all lines, x

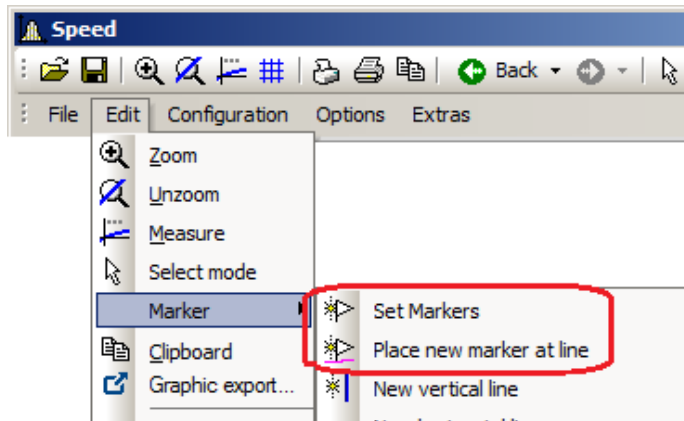
Marker for color map

x = 0.000 RPM

Property	Contents
Position	
How is x specified?	x-units
x =	
How is y specified?	y-units
y =	
Connecting Line	
Length	200
How is the length specified?	% of text height
Length [% of y-axis]	
Angle (0..360°)	45
Arrow style	Narrow, solid
Arrow size [mm]	Auto
Extension	

11.6.6.5.1 Set markers

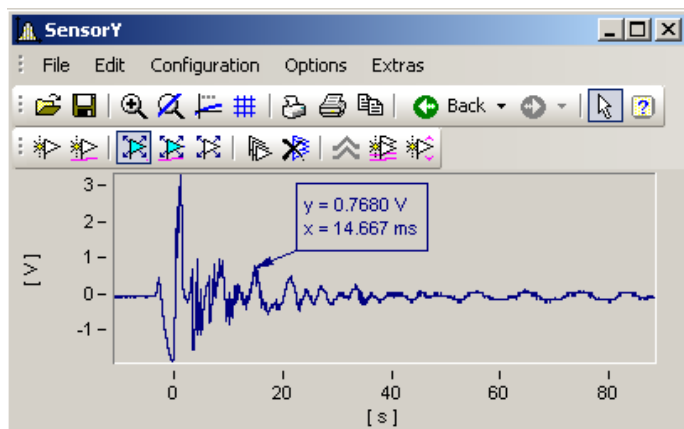
There are two ways of setting a marker, which are accessed either from the Marker toolbar or via the menu item *Edit/Marker*.



Set Markers (📍)

This function lets you generate markers at any desired position. If the function is selected, the cursor is represented by this symbol (📍), which disappears again after the marker is set.

Click to the point in the curve window to be marked. A dialog box is prompted in which marker characteristics can be established. If no settings are made and the dialog is ended with <OK>, the x- and y-values of the marked point are used as the marker text. More about [Marker Definition](#) ⁸⁴³.



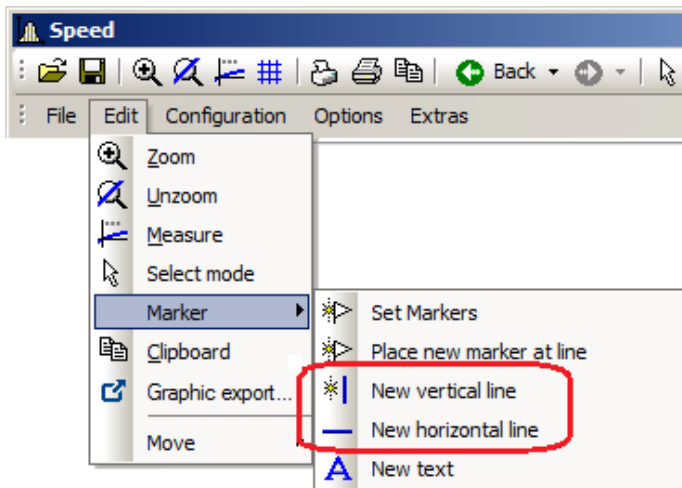
Markers can be placed anywhere in the curve window. Zoom more intensely if a marker should be placed near a line. Accuracy depends on the resolution. To place a marker exactly at a certain point in the curve, select the function *Place new marker at line*.

Place new marker at line (📍)

This function only helps to generate markers at lines. If the function is selected, the cursor is represented by this symbol (📍), which disappears again after the marker is set. The marker automatically snaps to the next closest line and then moves in response to cursor movements only along the line, until the cursor approaches another line.

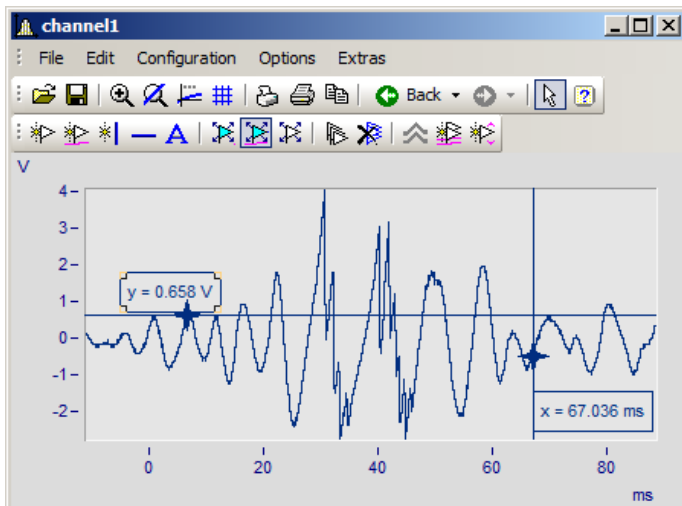
Both modes can be exited by a click of the right mouse button.

11.6.6.5.2 New line

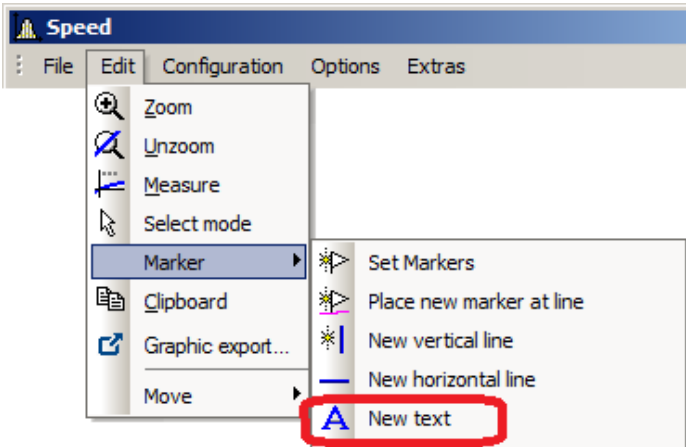


Vertical line new and Horizontal line new

Use these to create vertical or horizontal lines. The location is displayed as an amplitude or x-position in the text box. If the text box attaches directly to the line, it can be moved along the line. Drag the box further away from the line, it can be placed freely at any location.



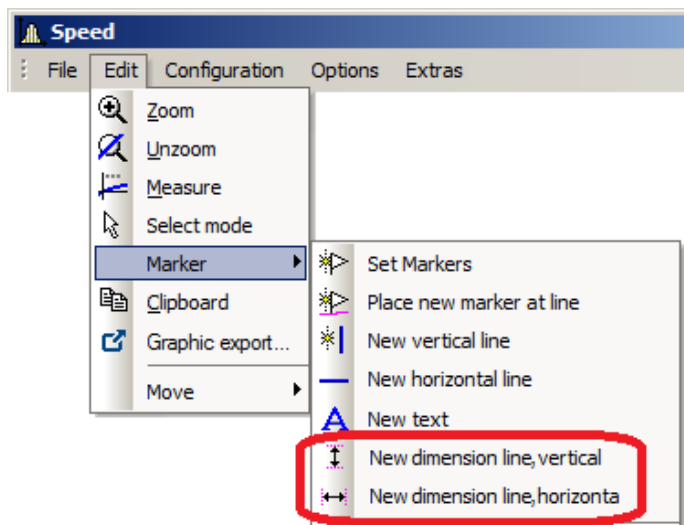
11.6.6.5.3 New text



New text

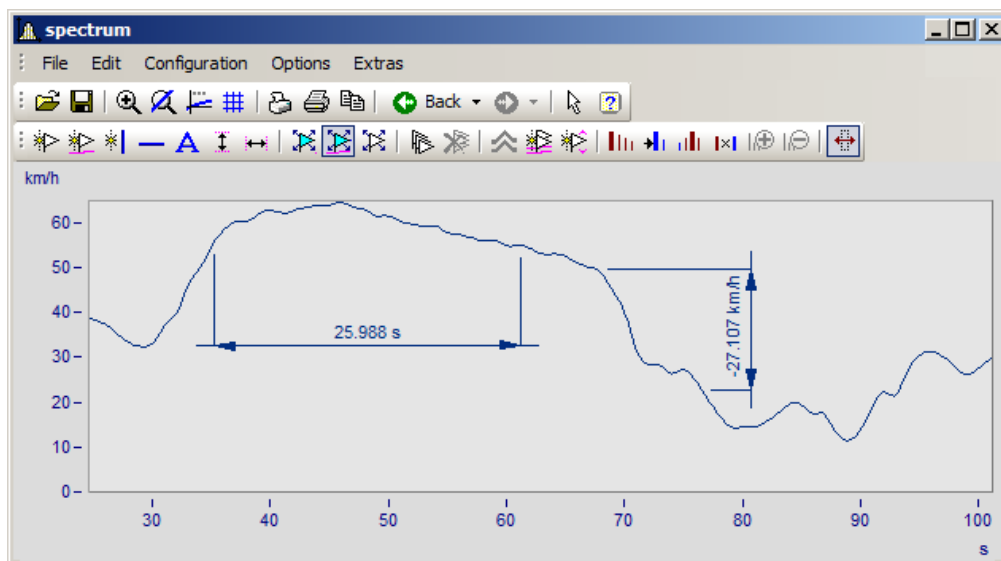
A text box without an arrow can be placed freely. Otherwise, the same settings are available as for markers.

11.6.6.5.4 New dimension lines

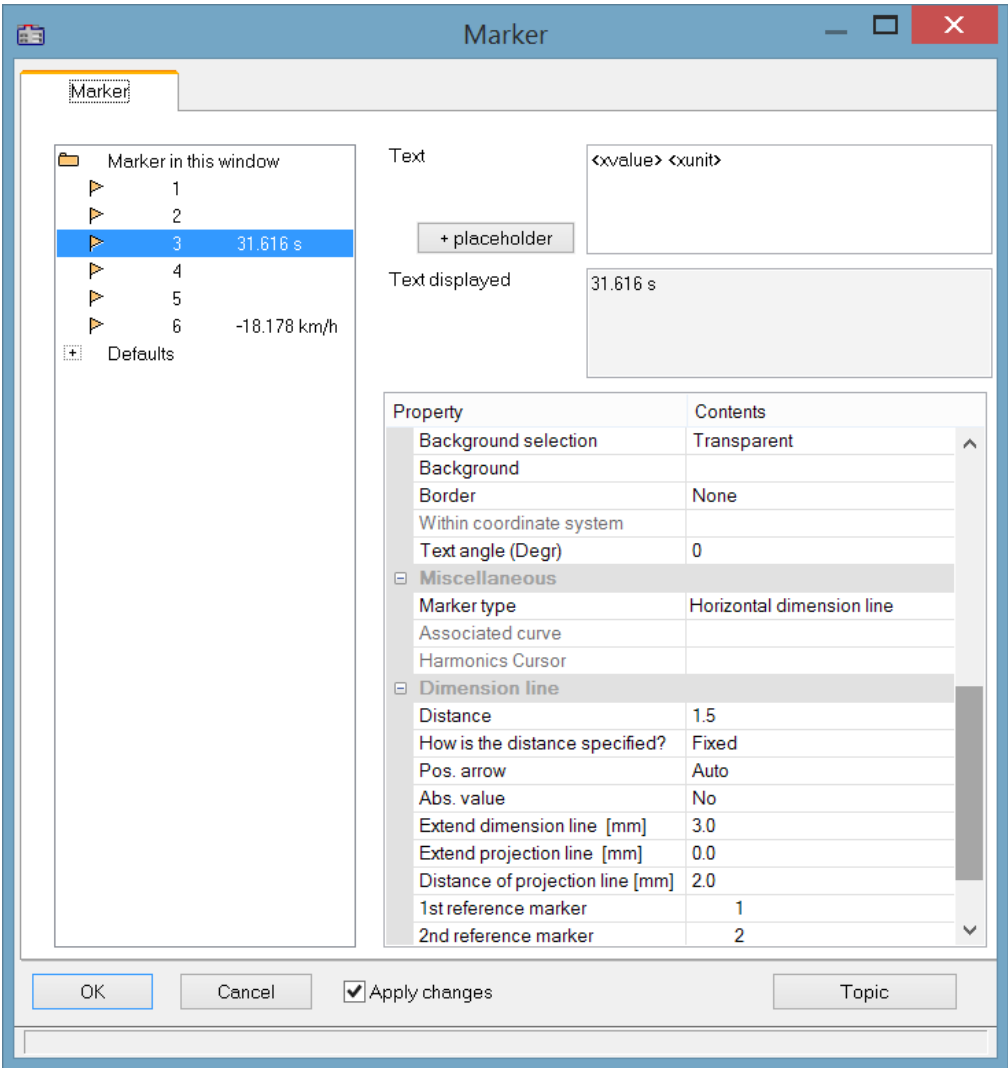


New Dimension Line: vertical and New Dimension Line: horizontal

Select this entry to create a vertical or horizontal dimension line between two points in the curve window where you click the mouse.



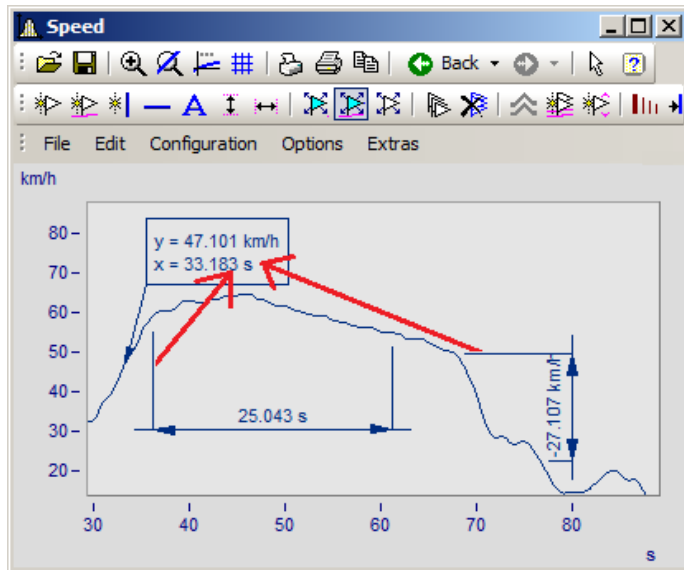
You can change the width of the dimension lines using drag&drop. The placement of the legend as well as of the line is positioned by drag&drop. Making settings for the markers is done as shown below:



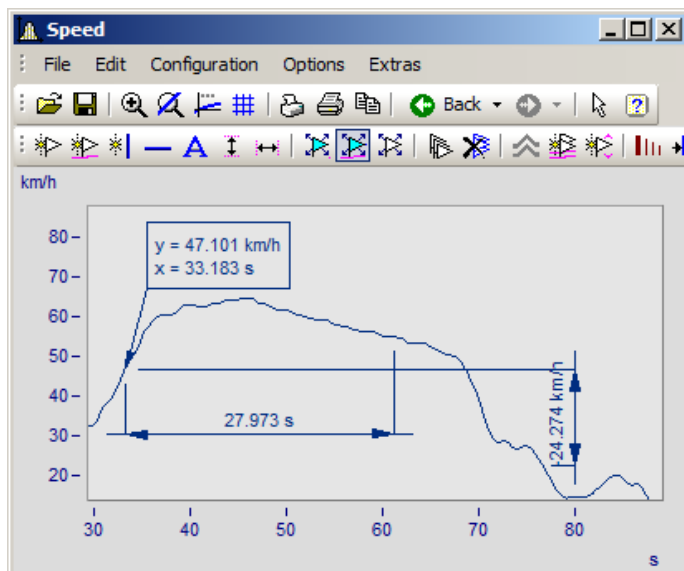
Position

Reference marker: A dimension line is defined by three entries for the marker: two reference markers (1, 2) and the actual line arrow and legend. If other markers are already in the curve window, these can also be used. Assignment of the markers is made by means of the list boxes for the 1st and 2nd reference markers. Alternatively, use drag&drop in the curve window to move a dimension line's extension lines to the existing marker.

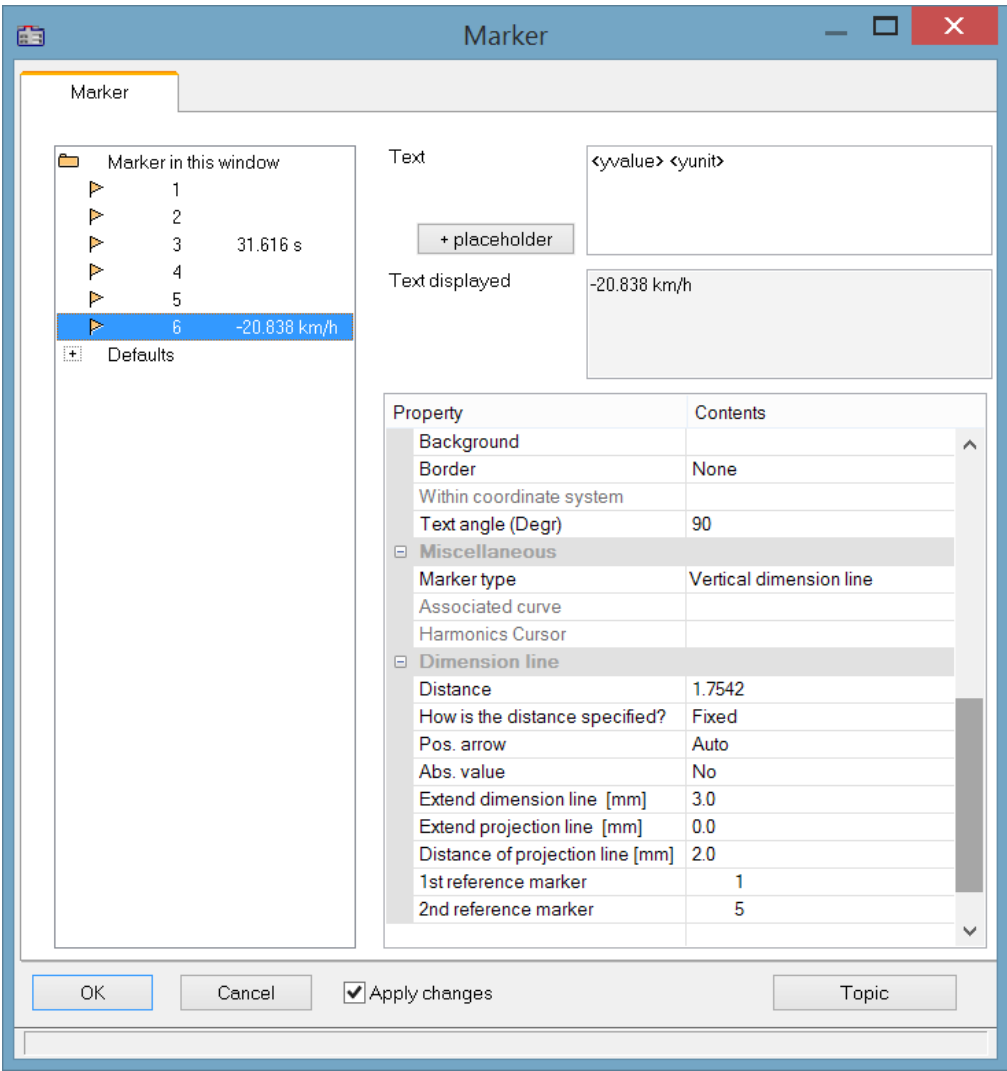
The example below shows how two dimension lines are linked to a marker. When the marker is subsequently moved, the dimension lines are adapted automatically.



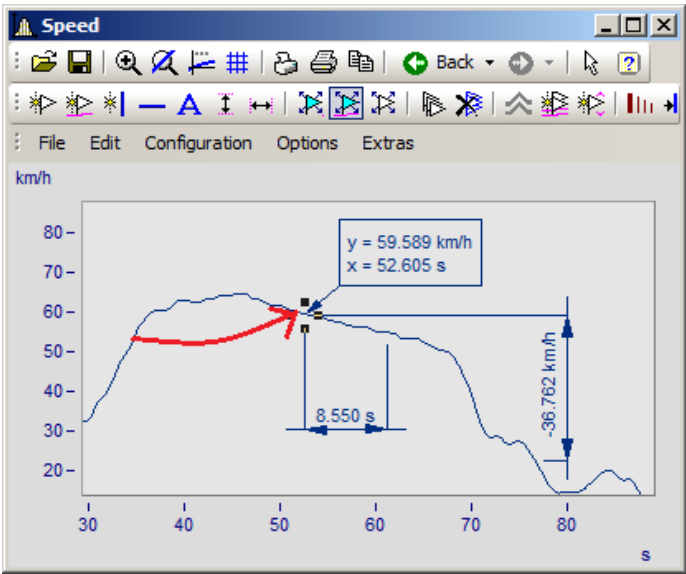
The marker and dimension lines are not yet linked. The extension lines are dragged to the marker.



Now the marker and the dimension lines are linked with the marker.

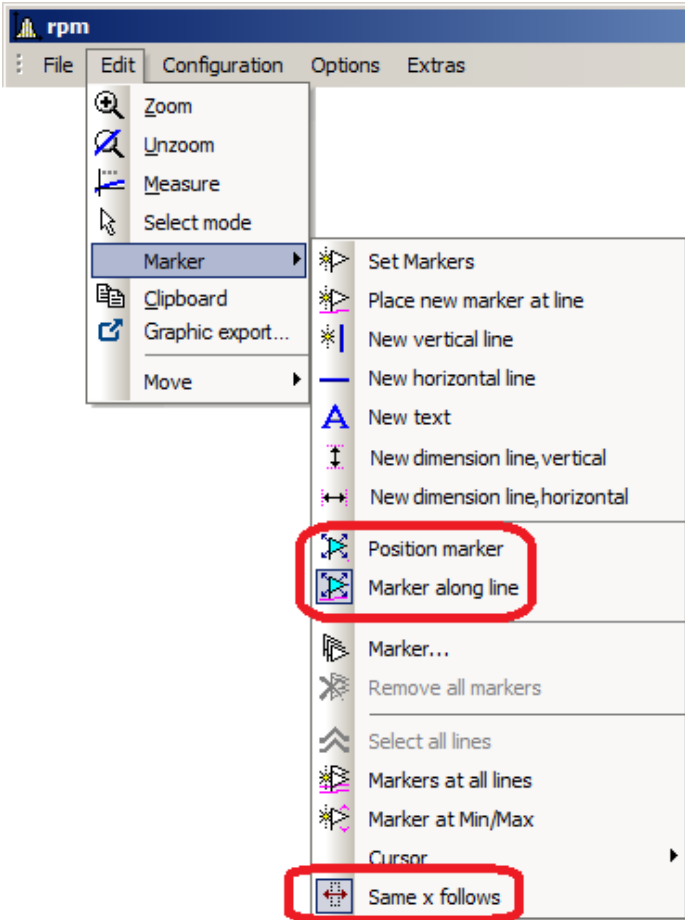


In the definitions dialog, the reference marker for the dimension lines is now linked with the marker.

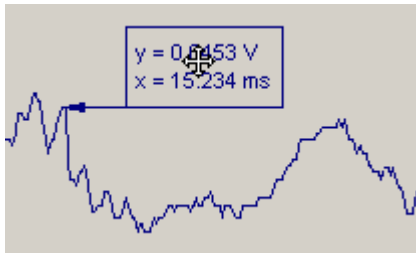
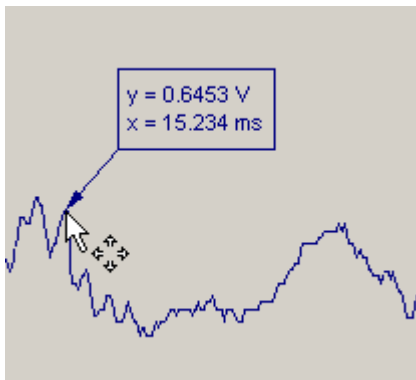


Now, when the marker is moved, the dimension lines are updated.

11.6.6.5.5 Move markers



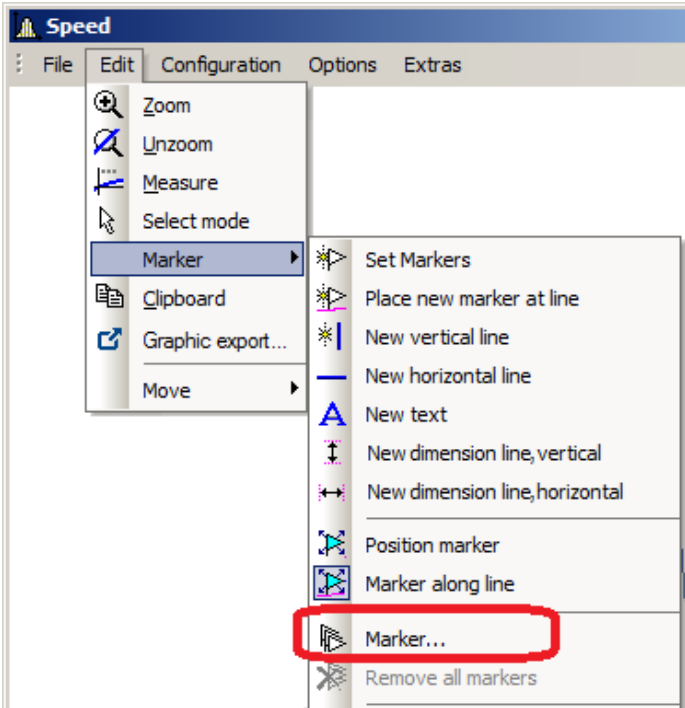
There are different ways to change the marker position:

Action	Description	
 Position Marker	This function lets you move a marker freely, while the marker text is automatically updated to reflect the respective new position.	
 Marker along line	This function lets you move a marker along a line, while the marker text is automatically updated to reflect the respective new position. If there are multiple lines in a curve window, the marker snaps to the line closes to the cursor's location.	
	In order to move a marker's label, bring the cursor near the label or the arrow, so that the cursor changes shape as shown below.	
Move label		
	Existing markers can be moved subsequently. To do this, use Drag&Drop to move the arrowhead point.	
Move marker		
 Same x follows	Markers assigned to the same x-position can be moved jointly by means of this function. To do this, move a marker. The markers having the same x-position are moved along with it automatically.	

Remarks

- Markers are assigned only to the curve windows, not the waveforms.
- In order to save markers, the curve configuration must be stored as a CCV file.
- Markers already established for a curve window may become meaningless if additional curves are modified or the XY-display is redefined. These markers can be deleted

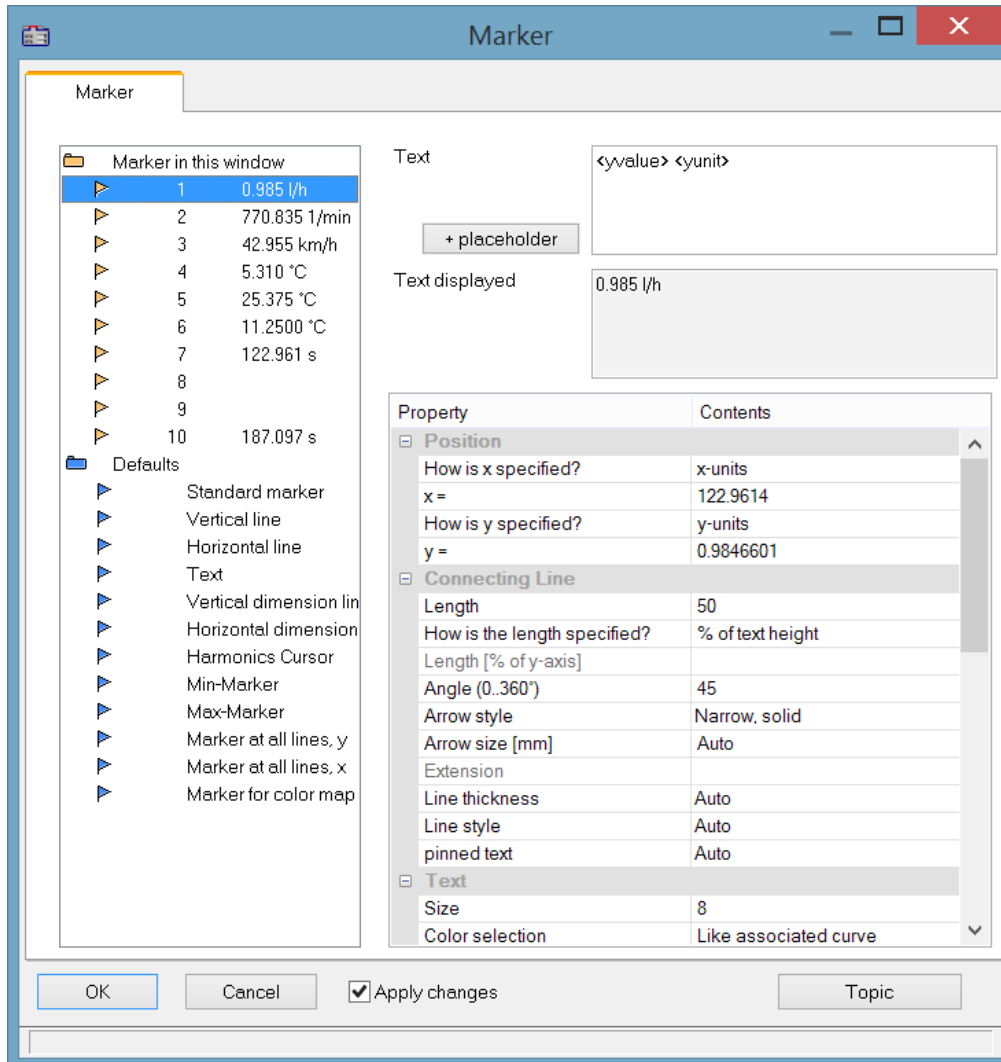
11.6.6.5.6 Marker definition



Function

Displays the properties of the markers created.

The following dialog appears if a marker was set in a curve window:



The list of markers, in which all markers in the curve window are listed, is found in the upper left box in the dialog.

Select an entry from the list to change the settings for the corresponding marker. Several markers can be selected by dragging the mouse across the entry or clicking several times on the entry while holding down the <CTRL> button.

The Marker-**properties** are listed at right in a table and are set using the column at right.

Marker type

Along with the usual markers having arrows and a text box, the following marker types are available:

- Vertical and horizontal lines
- Text
- Vertical and horizontal dimension lines

Normally, the type is determined already when the marker is set. By means of this combo-box, it can be changed subsequently.

Position

Determines the marker's position. By default, the position is specified in the coordinates of the curve to which the marker is assigned. In addition, the numerical value can also be specified in x- or y- units, or in percentage of the x- or y-axis. Select the corresponding entry in the combo box to the right of the number boxes. When specifying the numerical value in percentage of the axis, 0% is at the lower left, 100% at the upper right of the coordinate system. This allows markers to be defined so they are always visible in the curve window, regardless of the scaling. This is useful for comments, which needn't necessarily be assigned to a certain line in the coordinate system.

When changing from dB display to linear display, the axis coordinates change and the markers slip from their position. The scaling should be determined before the markers are specified.

Connecting line

The connecting line between text and marker is arranged in this group.

Settings	Description
Length	The length of the connecting line between the text and the marker. It is specified either in x-units, y-units, in percent of axis length or in percent of the text height. The last option is recommended. All value specifications should be in scientific notation with units, meaning that the exponent must be indicated. For a length of 0, no line is drawn. If an arrow is set, the length of the line is the entire length for line and arrow. If the line is shorter than the arrow, the arrow is drawn completely.
Angle: (0...360°)	The angle of the connecting line can be set between 0° and 180°. The line is horizontal if an angle of 0° is specified. This means that the line can be set to run in any direction.
Arrow	The line between marker and text may have an arrowhead at the end of the marker. The size and type can be selected: • none • wide • narrow • wide, solid • narrow, solid • large • large, solid • circle • Point • Diagonal dash • Star • Default The size of the arrow is oriented to the size of the symbol, see also "Clipboard settings".
Line thickness, Line style	Settings like for Lines. At printout, the line appears in the thickness of the cursor-line, see Clipboard Settings.
Text corner	Position of the text box (upper right, upper left, lower right, lower left)

Text

The text specified in the text window is assigned to the marker, and can consist of several lines. Press <CTRL> and ENTER to enter a new line. If a marker is created, this text is already defined as the x- and y-measurement values. This text can be overwritten.

Placeholder

The edit box initially contains the placeholder *<auto>*, which by default represents the y and x-value. This entry can be supplemented/replaced with the following placeholders:

Placeholder	Description
<xunit>, <yunit>, <zunit>	Display of the x-, y, or z-unit
<xvalue>, <yvalue>, <zvalue>	Display of the marker's x-, y, or z- component. For xvalue, mostly the time value in the x-axis' formatting.
<name>	Display of the variable's/channel's name
<comment>	Comment on the variable
<xtimeofday>"	Display of the marker's time. For this purpose, the x-axis must represent absolute time.
<xdate>	Display of the marker's date. For this purpose, the x-axis must represent absolute time.

Placeholders for values, with specified precision level

Placeholder	Description
value:fx e.g. <yvalue:f2>	Num#ber of digital positions (0..15); in the example 5.34211 -> 5.34
value:fxpy e.g. <yvalue:f2p3> <yunit>	Specified number of decimal places and order of magnitude as exponent of 10. The exponent of 10 is only displayed along with the unit. In this example, 3556.23 becomes 3.55 10^3 RPM.

Note

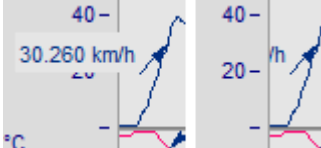
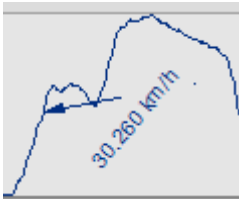
- [Greek letters](#) ⁹⁰¹ can also be displayed.
- Numbers without a unit sometimes come to a misleading representation:

Example:

31.000.000 represented with <yvalue: f0p6><yunit> is displayed as 3110⁶

A multiplication sign or blank space creates clarity:

<yvalue: f0p6>*<yunit> -> 31*10⁶

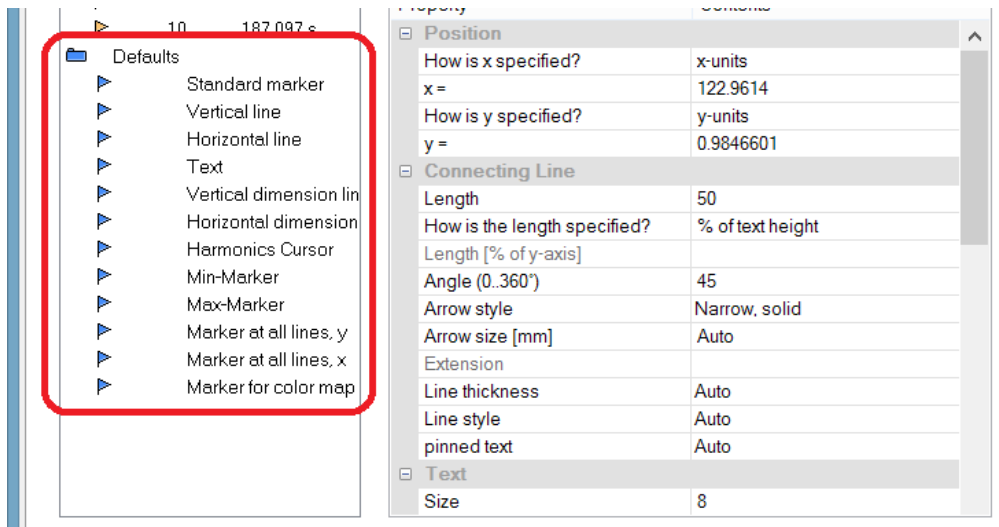
Settings	Description
Size	The size of the text can be defined in points. In general, a 12-point font can be read quite easily. The font specified in "Clipboard settings..." is the font used for printing. A true-type font should be selected. The standard font set for the curve window is also used for the monitor screen (see " Presettings " ⁸⁸⁵ ").
Color	The color can be selected as absolute (e.g. red) or relative. A relative color definition refers to a color already defined for the curve window. For example, if the color of the first curve in the window is selected, it is possible to select another color for the printout and the screen.
Background	The background of the text can be in the same manner as the text color. A transparent background is also possible, so that lines behind the background are not completely hidden.
Border	The text may have a border around it. Various options are available for the border joining the connecting line with the marker: • none, simple, pointed (trapezoid-shaped), double border
Within coordinate system	When the text box is located at the edge of the coordinate system, you can decide whether it overlaps the edge or whether it is clipped. 
Angle text (Degr)	Angle of the text box up to ±90 degrees. 

Delete

All lines selected in the marker list are deleted.

Edit defaults

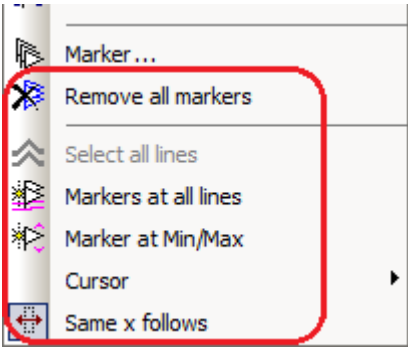
The **Defaults** for the *Marker types* are listed at left links and can be edited at any time using the Properties table.



Remarks

- If several markers are selected in the list, the characteristics for all of the selected markers are indicated. If a characteristic differs among the separate markers, this is indicated by "???" in the corresponding box. If a characteristic is changed, this change is effective for all selected markers.
- An inapplicable entry in a text box is ignored.
- When changes are being made in a dialog, they are implemented immediately and displayed. They can be seen immediately when the dialog is <Shift>ed next to the curve window. If the curves are very long and the on line-update is too slow, the curve window should be placed so that the graphics can no longer be seen when working in the dialog. It is recommended to use a transparent display for the representation in a waterfall diagram.
- Markers already defined for the curve window may lose their meaning if the curves are modified or XY-plots are redefined. These markers should then be deleted.
- The definition of markers is recorded when the curve window configuration is saved. The markers are also reloaded when a configuration is loaded. However, the markers may not be visible in the window because the settings do not fit the window or the curves.
- To design markers which are not device-specific, the colors should not be specified as absolute. Use instead a default or a color in the curve window (e.g. the color of the first curve). The default itself can be dependent on the device. The same is true of the font size and the line characteristics.

11.6.6.5.7 Extra marker functions



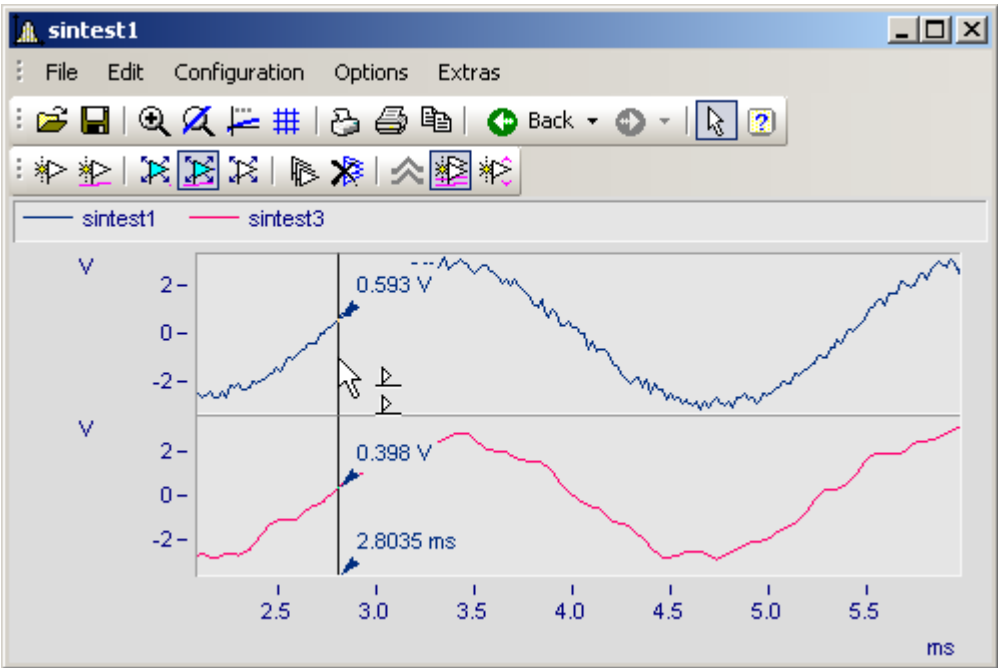
There are four additional marker functions available:

Menu item	Description
 Delete all markers	With this function, all markers set in the curve window can be deleted at once.
 Select al lines	With the Select mode active, this function selects all lines in the curve window.

Set markers at all lines or at all lines selected in the Select-mode. Select the lines desired an click on the symbol in the toolbar. A vertical line appears which extends across the entire curve window, even when multiple y-axes are stacked one above the other. The line denotes the y-value and the x-value on the y-axis. Simply move the line with the mouse and confirm setting of the marker at the desired position by clicking the mouse. The position of the markers as well as their labels can be moved as desired, just as described in the section "[Set Markers](#)".



Markers on all lines



 **Note**

Markers on all lines

If two lines are close together so that the positions of the markers almost match, only one marker is created for this position.

Menu item	Description

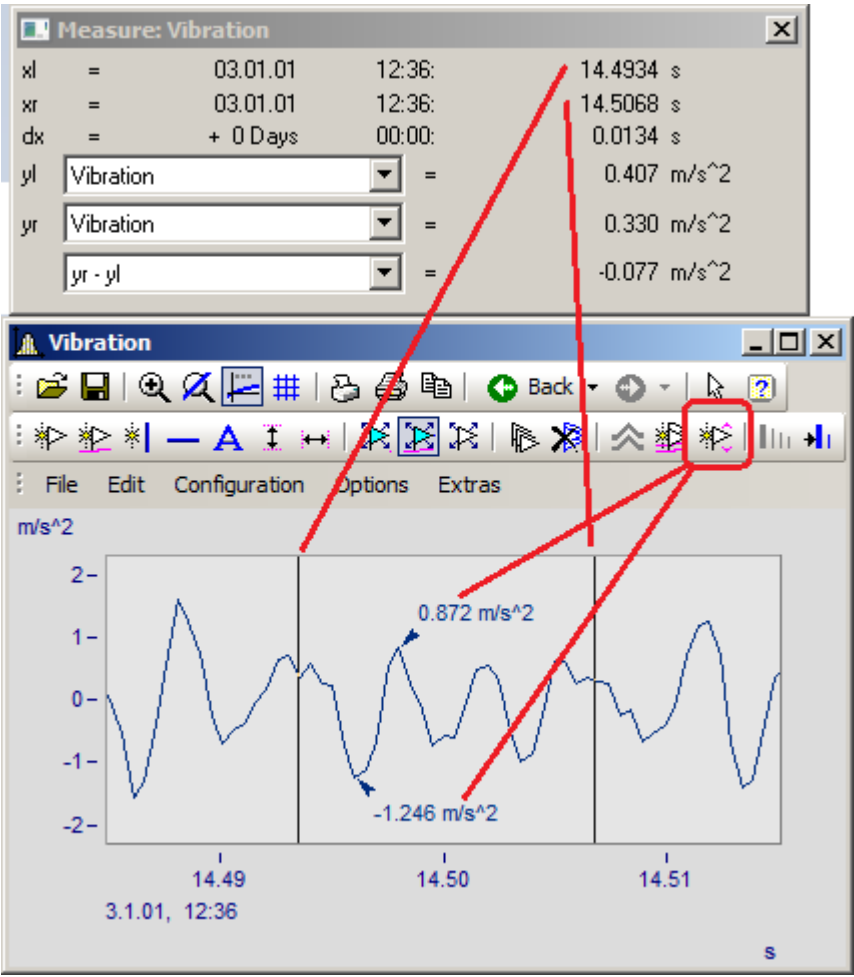
Markers are set for the maximum and the minimum of the line selected. If multiple curves are in a curve window, the function only applies to the line selected. As described in the section "[Move Markers](#)", the marker's position and its label can be moved as desired.

The function only affects the selected time segment. For this reason, zoom on to the region before setting the marker.

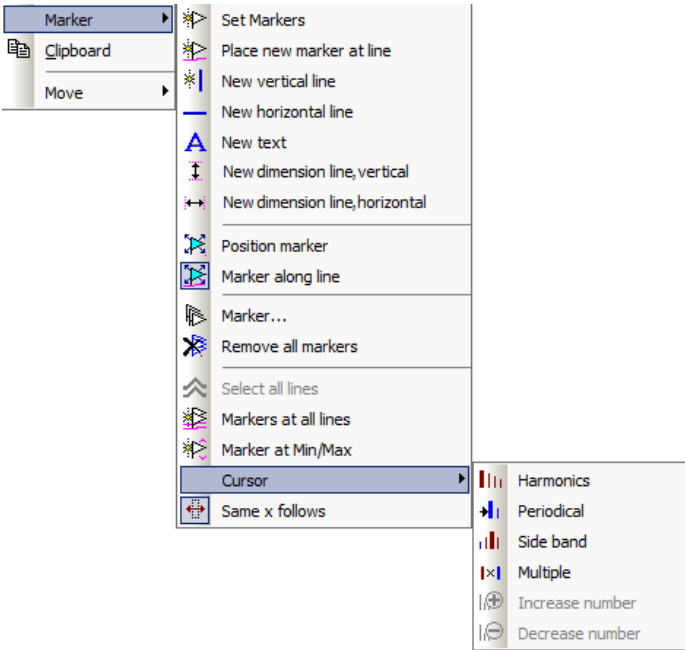
If the "Measure" function is also active, the system finds and denotes with markers the minimum and maximum just between the two measurement cursors.



Markers at Min/Max

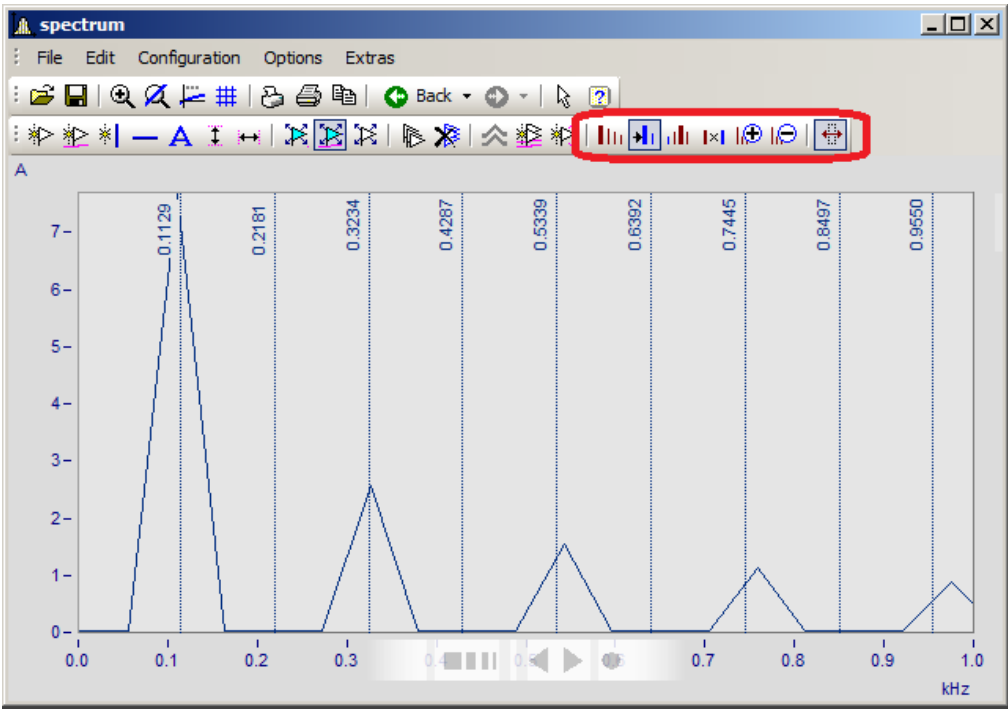


11.6.6.5.8 Harmonics cursor



Function

Harmonic cursors indicate periodic multiples of a fundamental frequency. Set the marker at the position of the fundamental oscillation. The position can be readjusted subsequently, if desired.



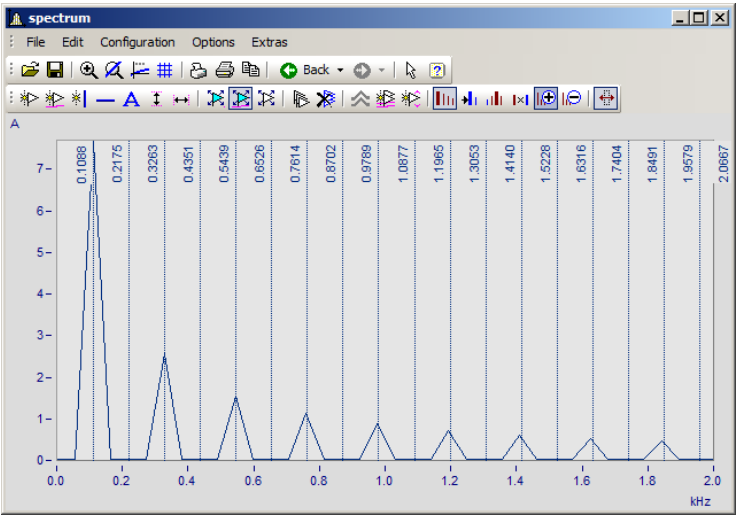
The following types are available:

Menu item	Description

Fundamental oscillations with harmonics. 10 equally spaced markers are inserted. Place the fundamental oscillation. The subsequent harmonics follow the fundamental oscillation by the respective multiple. The fine adjustment is performed by moving the upper harmonics.



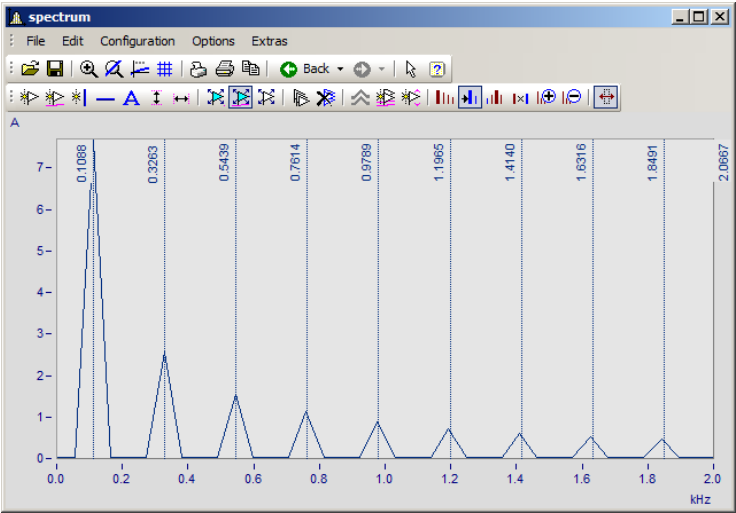
Harmonics



Periodic processes with arbitrary start time. The fundamental frequency can be freely placed. The distance between the harmonics is independent of the fundamental oscillation.



Periodic

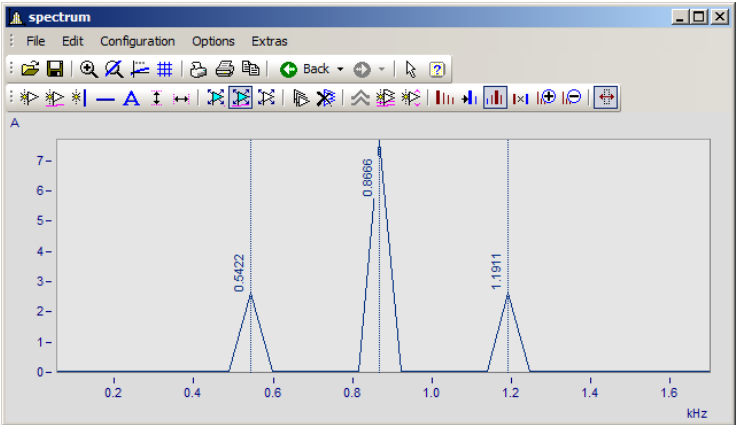


Menu item	Description

Fundamental oscillation with side bands. 10 equally spaced markers are inserted. Position the fundamental oscillation. The harmonics which follow the fundamental oscillation by the respective multiple. Fine adjustment is performed by moving the upper harmonics.



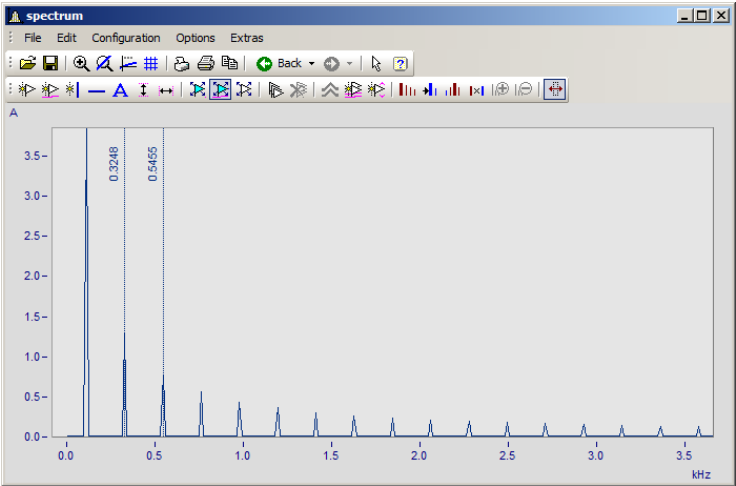
Side band



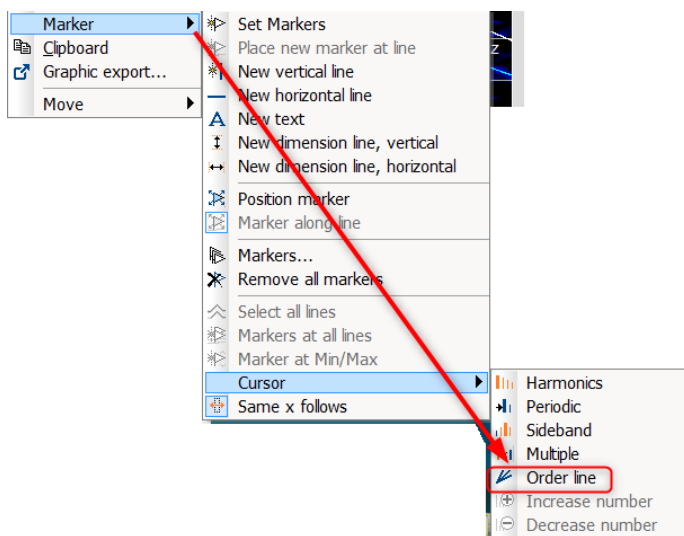
Distances at a fixed ratio. Two frequency lines which can be freely positioned.



Multiple



11.6.6.5.9 Order lines



Calling the Marker dialog for order lines

Function

The amplitudes of RPM-dependent vibrations can be displayed by means of various color diagrams.

When the amplitudes are plotted over the RPMs, the correlation becomes visible as straight **order lines**, which begin in the origin of the coordinate system. Frequencies which are independent of the RPMs appear in this diagram as horizontal lines of constant frequency.

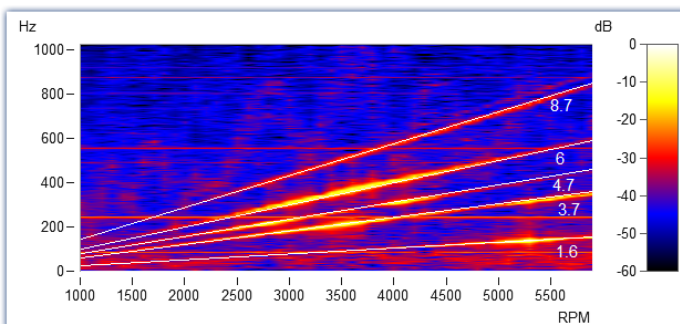
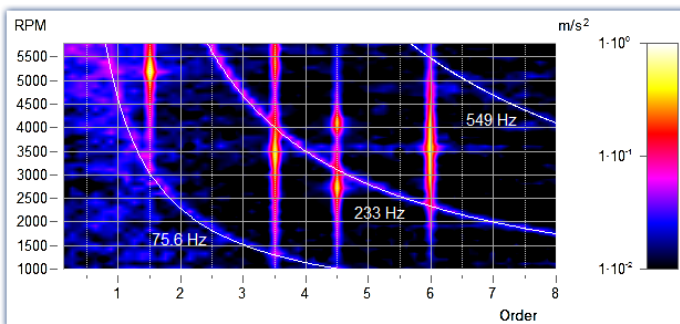


Fig. Amplitude spectrum with frequency over RPMs

In the Markers dialog, these order lines can be highlighted and labeled. Clicking on the marker symbol "Order line"  creates the first marker, applying the presets for calculating "Order line in the RPM-spectrum". Subsequently, you open the marker definition dialog by double-clicking and you can design the display.

Calculating the diagram over these orders, they appear as vertical lines. Fixed frequencies are distorted to **hyperbolas** by these conversions.

For the example shown here, setting for the calculation in the marker definitions must be "RPM and Order".



RPM and order with hyperbolas

As with all markers, the text belonging to the line can be designed with placeholders, freely defined texts and custom formatting.

The line's position can be moved manually using the mouse. As a snap grid, the parameter *Multiple of* in the section *Order line* is used.

New hyperbola markers get the value of the highest frequency line x120% as initial frequency - and this to the next grid *Multiple of*.

The most important properties of this *Marker-type: Order line* are:

Property	Contents
▢ Miscellaneous	
Marker type	Order line
Associated curve	At the first y-axis
Harmonics Cursor	
⊕ Dimension line	
▢ Order line	
Parameter	6
Calculation	Order line in the RPM-spectrum
Start of the line [%]	0
End of line [%]	100
Labeling position [%]	95.079
Multiple of	0.1

Order line

Calculation with unit over unit

Order line in the RPM-spectrum

RPM over frequency

Frequency over RPM (=default)

Angular frequency over frequency

Frequency over angular frequency

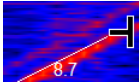
Hyperbola in the order spectrum

RPM and order (=default)

Angular frequency and frequency

Parameter is the value of the order or of the constant frequency which appears as a hyperbola. This value determines the position and is represented by the placeholder *<yvalue>* and formatted accordingly, e.g. *<yvalue:f1p0> Hz*

Start/End of line [%]: Determines whether the line is to be drawn up to the edge of the coordinate system (0-100%) or should keep some distance from it (e.g. 5-95%). The ends of the line can also be edited graphically using the mouse.



Labeling position [%]: This value specifies the position in percent of the visible order line length. The text can also be positioned manually using the mouse.

Multiple of: When the order line/hyperbola is moved, the position snaps to multiples of this value. When changing *Multiple of*, the parameter in the curve window is updated immediately, but not in the properties table. This allows further editing of the value in the properties. With Ok of the properties or click on the marker list the parameter is accepted.

Additionally, the section *Text* offers the usual formatting properties for texts, and in the section *Connecting Line* there are parameters such as *Line thickness*, *-style* and *Text angle*.

 Note

Order lines

- The first order line applies the [presettings for the marker-type Order Line](#)⁸³². You are able to format these according to your wishes. If any order lines are added, the settings for the previous order line are applied.
- It is possible for a marker to be inserted in the range which is not visible, e.g. if an order spectrum was calculated only for a range of 2000-4000 RPM. In this case, the marker must be given a parameter value in the visible range by means of the marker definition.

11.6.6.6 Export

11.6.6.6.1 Clipboard and Graphic export

The graph displayed in the curve window can be sent to the Windows clipboard. The clipboard is a Windows tool which allows applications to exchange data in any form, e.g. text or graphics.

With the menu item [Edit\Graphic-export...](#)^[876], the graphic can be saved as a file in an image or PDF format. The [default settings](#)^[890] are taken into account here. For example, it is possible not to overwrite an existing PDF file, but to add the new graphic to the document.

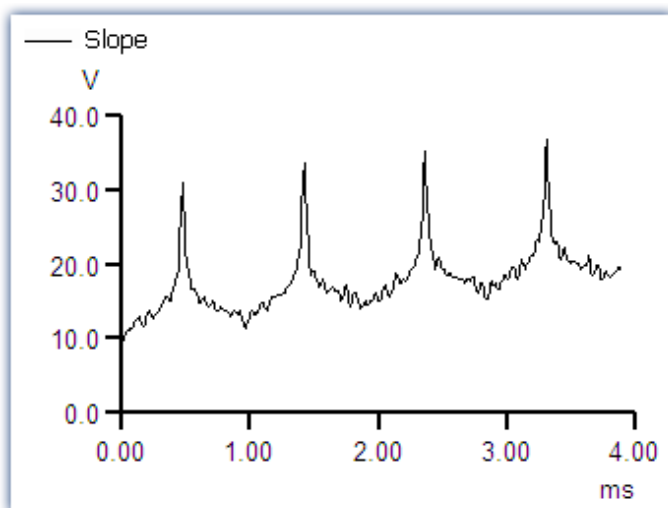
When one application sends graphics to the clipboard, another application can read and retrieve these graphics from the clipboard. In general, data are not damaged when they are read from the clipboard, which means that applications can retrieve the same graph as often as desired. Existing graphics are replaced when an application sends new graphics or text to the clipboard. Only the last information sent to the clipboard can be read.

In the case of curve windows, graphics are sent to the clipboard in the form of a memory-resident metafile. A meta file is a standard format defined in Windows and is supported by many applications. The meta file does not contain screen pixels; instead it describes the graphics using a multitude of vectors. This creates a high-resolution graphic display which can be arbitrarily scaled up or down.

Graphics sent to the clipboard can be read by text processing or desktop publishing programs. In that environment the graphics can be combined with text and other graphics to create documents to be printed by any printer or plotter supported by Windows. Graphics are printed in the best resolution possible for the output device.

Graphics are created in the fashion specified in the menu option "Options/ Clipboard settings...". The specified size, font and line width are used to format the graphics.

Graphics can be output to the clipboard in color or in black and white. Have regard for the color setting possibilities, especially the printer's. The curve window copied to the clipboard can, for example, have the following layout:



Remarks

- Letters may look distorted, lines may intersect and overlap and details may disappear when a meta file created from the curve when another program is loaded. The image can be improved by enlarging the display on the screen. Meta files are designed to optimize the image output by printers or plotters, and not for ideal display on the monitor.
- The fonts used for the meta files are selected from the available fonts for the printer (see "[Printer Setup](#)"⁸⁷³). Not all of these fonts are easily legible on the screen or available in the necessary size.
- A meta file can only be created when sufficient memory is available. The clipboard remains empty or is emptied if the available memory is insufficient.
- In some extreme cases, it may not be possible to create a suitable meta file. Such situations can occur when several curves are displayed, or in connection with dotted, bar-graph or XY-representations. It is often helpful to display a smaller section of the waveform, to change the display mode, to cut out a section of the waveform or reduce data using mathematical functions.
- When axes scaling is set to automatic, the axes labeling on the screen and on the printer will be calculated differently for the current dimensions. Specify a set scaling and a set number of markings to achieve clearly defined proportionality.
- In printing, texts may overlap if the selected font is too large.
- If a waveform does not have a "created.." - time, the current time is automatically designated as such.
- Images of curves sent to the clipboard can also be incorporated in reports using the imc Report Generator. However, a direct transfer of the curve window is recommended, whereby objects pose as curves. See chapter '[Report Generator](#)'⁸⁷¹ in this document.
- The time format (time and date) can be changed using menu item "Country settings..." in the Windows Control Panel.
- Even the best and most popular text processing and desktop publishing programs contain minor flaws. Although the graphics in most programs appear to be correct, one of these programs does not correctly interpret the offset of the curves, which is present as a command in the meta file. As a result, when graphics are inserted in this program, the curve lies next to the coordinate system. In another program, the clipping (fading-out) of curve sections extending beyond the coordinate system is ignored. Thus, the curve may extend well into the labeling or generally beyond the coordinate system. It is recommended to transfer the graphics to the Report Generator or to a text processing program directly, without modifying the graphics.

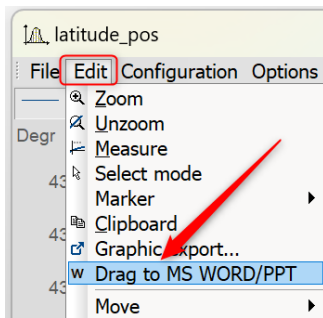


Reference

Clipboard Settings

Section '[Clipboard Settings](#)'⁸⁷⁸

11.6.6.6.2 To MS WORD (OLE)



Users with an **imc FAMOS Enterprise** or **Runtime** license can embed the curve window into a WORD or PowerPoint document.

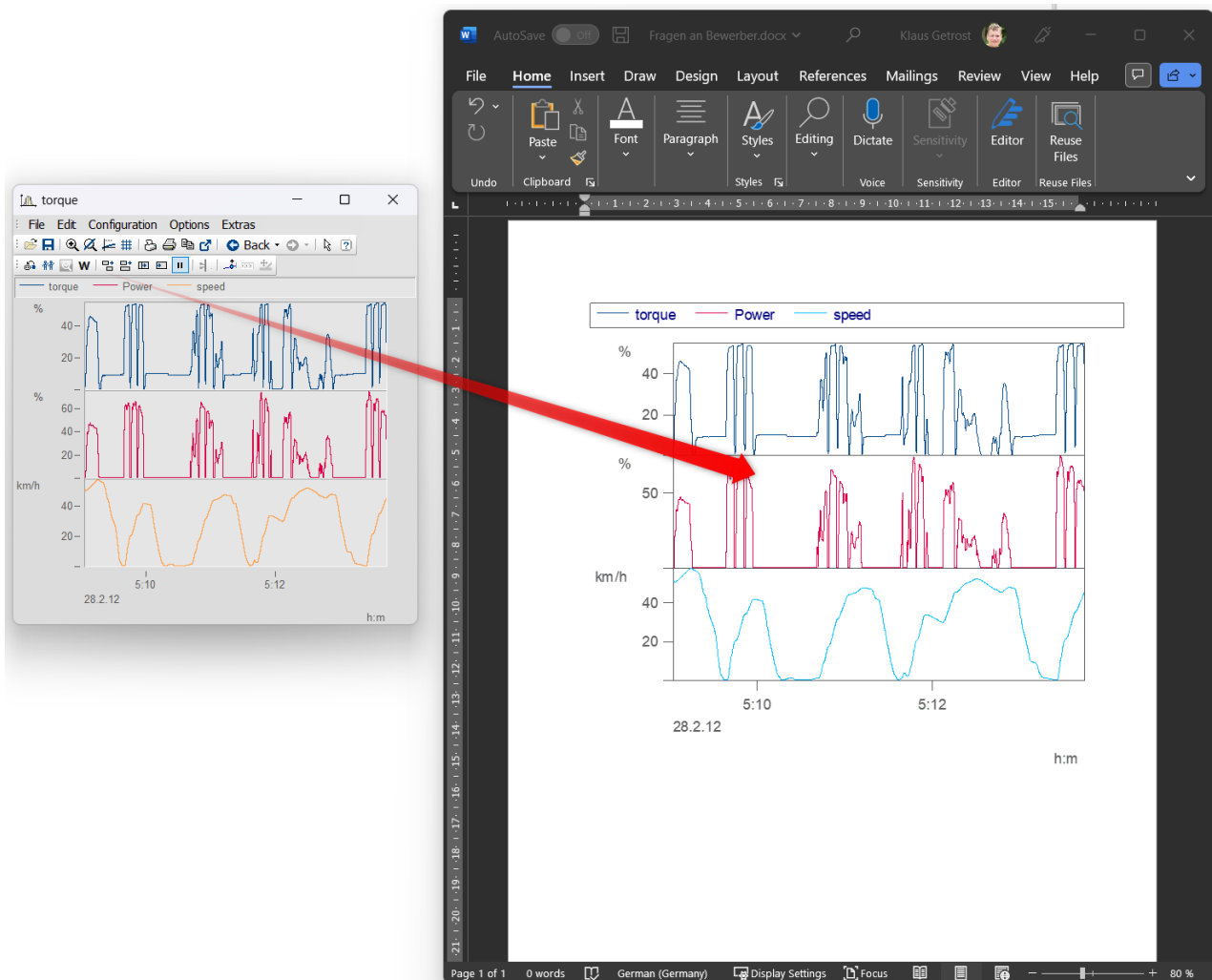
The **transfer** is done by **Drag&Drop** via the menu of the curve window or its toolbar.

Note

- With this action, the **data** is also transferred. This affects the file size of the Microsoft document, whereby the data is saved compressed. If the data exceeds a certain size, the transfer is prevented! This depends on the PC and the installed Microsoft version.
- **On the target system**, the curve window in the Microsoft document can only be **edited** if at least an **imc FAMOS reader** is installed. Otherwise, the curve window is displayed as an image and cannot be edited further.

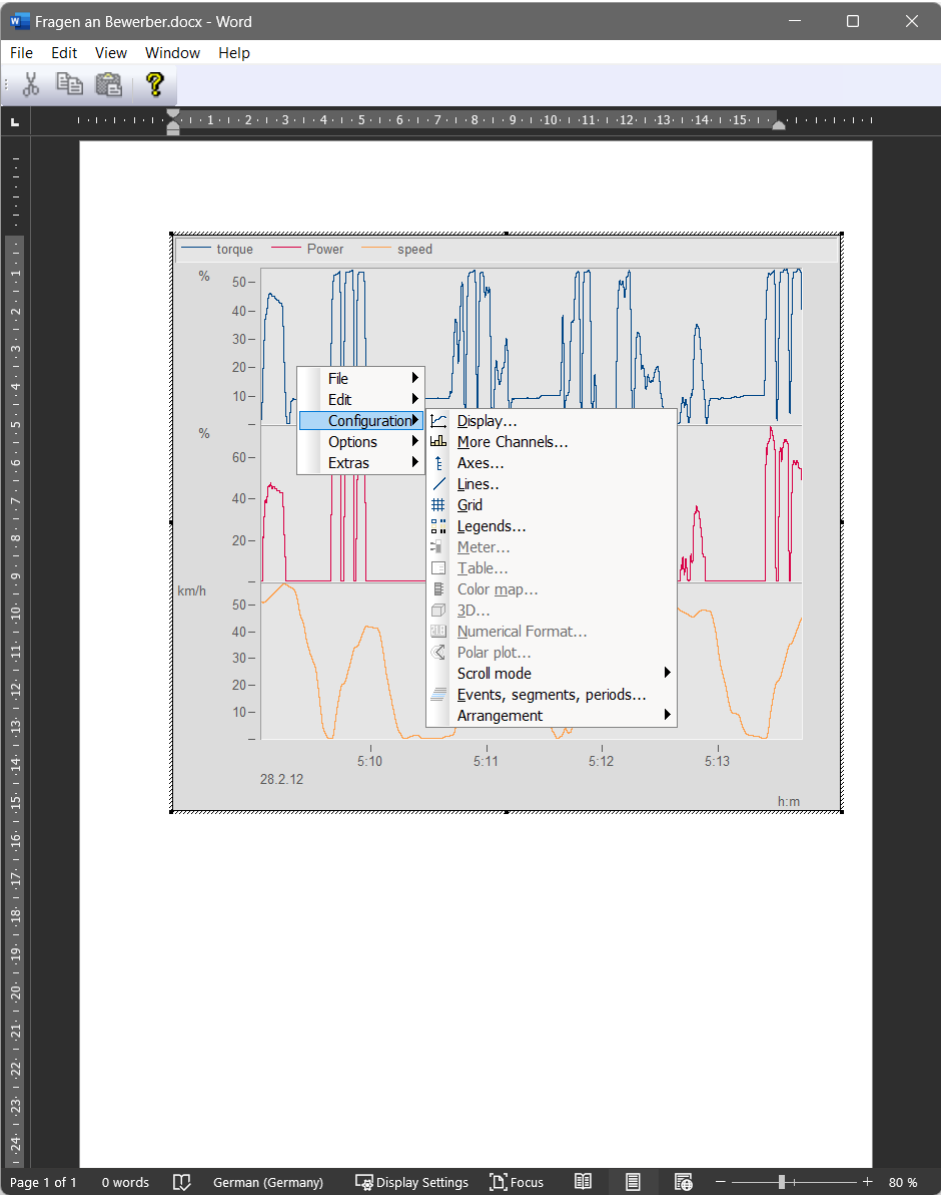
Transfer

The curve window is transferred to the Microsoft document by dragging and dropping from the *Edit\Drag to MS WORD/PPT*, or via the communication toolbar with the button **W**:



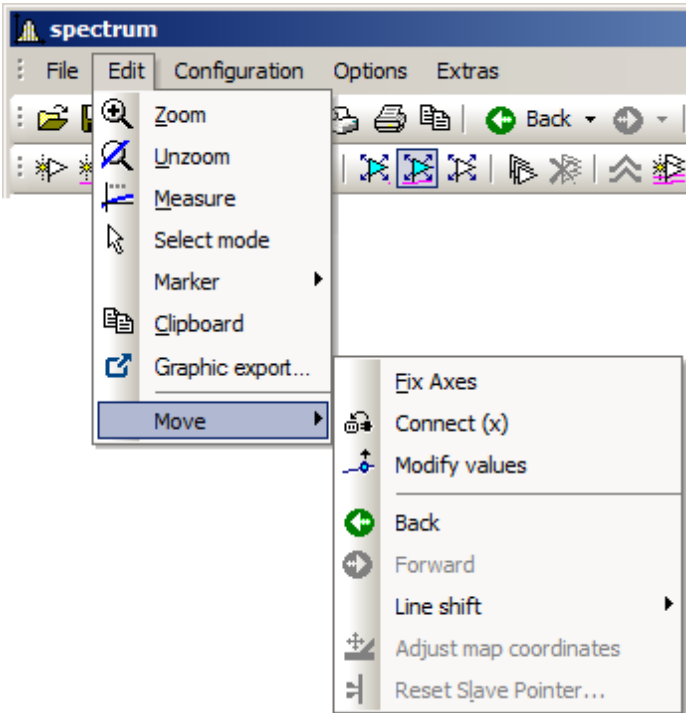
Edit

Double-click to set the curve window into edit mode. This is left again by clicking in the area outside the curve window. For editing, at least the imc FAMOS reader must be installed.



11.6.6.7 Move

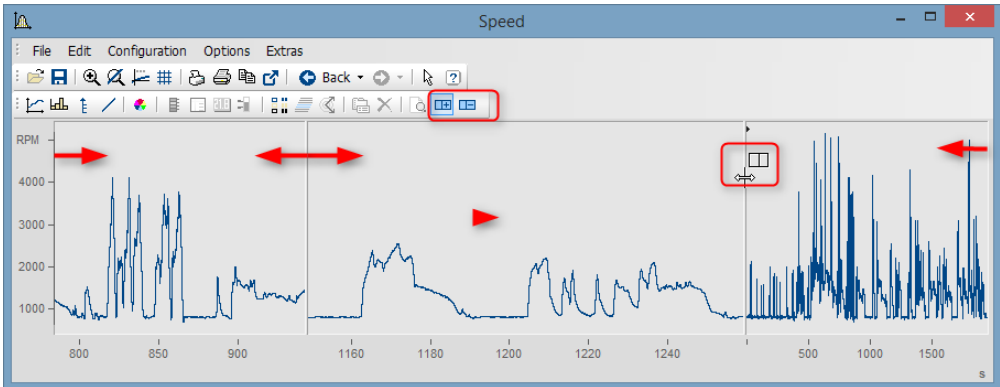
The menu item *Move* contains various curve window functions for working in the curve window:



Menu item	Description
Fix Axes	This lets you fix the current axes so that the curve window does not automatically adapt to any newly added waveform, but rather keeps the previously existing axis scaling.
Connect	For connecting curve windows. For more on this topic, see the section Connect ⁸⁶² .
Modify values	This function lets individual measured points from a data set be moved in the y-direction, thus changing their values. These changes are applied directly in the data set and stored. Caution: These changes can no longer be reversed by means of the <i>Back</i> -function. There will also not be any prompt for confirmation of the changes to the data set.
Back and Forwards	Step-by-step reversal of the most recent changes, or conversely repetition of the most recently reversed changes. With these two functions, changes in the curve window view can be reversed, or reversed changes can be restored. In the toolbar, the <i>Back</i> -function is additionally provided with a dropdown history, so that every change to the curve window view can be reversed separately. The exception is the function <i>Modify values</i> , since it directly changes the data set and not the curve window view.
Line-Shift	For moving selected lines in horizontal and vertical direction. More on this topic is presented in the section Line-Shift ⁸⁶⁵ .
Reset Slave Pointer...	For resetting the slave pointer. For more on this topic, see the section Bar meter ⁸⁷⁶ .

Split-Mode subdivides the curve window's X-axis. This makes it possible to observe the same data at different points in time and at different zoom levels. With *Split-Mode+*, the curve window can be split an arbitrary amount of times. With *Split-Mode-*, the subdivision is canceled.

Split-Mode

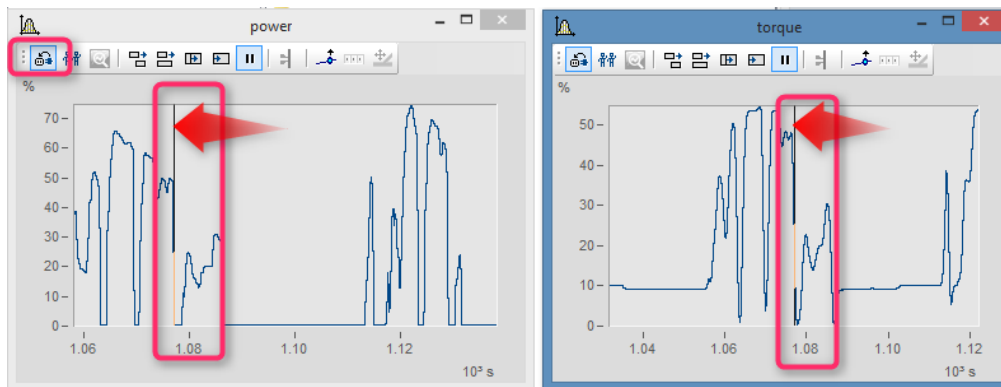


Split-Mode

11.6.6.7.1 Connect (Link)

Use this button to create a link between a curve window and either another curve window or a imc FAMOS table window. The connection is a x- (or scroll-) connection. Two windows connected in this manner have the same x- value at a defined position in their display ranges. Scrolling in one curve window prompts synchronous scrolling in the connected window. Several windows can be connected to form a linkage chain. No matter in which window the displayed x-range is changed, all connected table or curve windows are automatically updated to reflect the change.

To create a connection, click on the designated button and hold down the mouse button. The mouse pointer changes to a "blocked" symbol. If the mouse pointer is now moved over a curve or table window, it changes its shape back to a cursor. As soon as the mouse button is released, the windows are connected. The button appears to remain pressed and thus indicates the connection.



The x-axes of both curve windows are linked.

A vertical (curve window) or horizontal (table window) reference line indicating the x-position of the linkage is shown in the window. The connected windows each have the same scaling value in the x-direction at this line. The behavior desired can be specified using the *Options* dialog opened via Options\Presettings\Settings > [What is influenced in the Link and This window follows](#) ⁸⁸⁵.

This reference line first appears in curve windows in the middle of the x-axis, but can be moved to any position desired using the mouse. Move the mouse to this line; when the mouse pointer then changes its shape, hold down the mouse button and the line shifts as you move the mouse.

All connected windows should have the same x-axis mode (relative x-axis or display with absolute time).

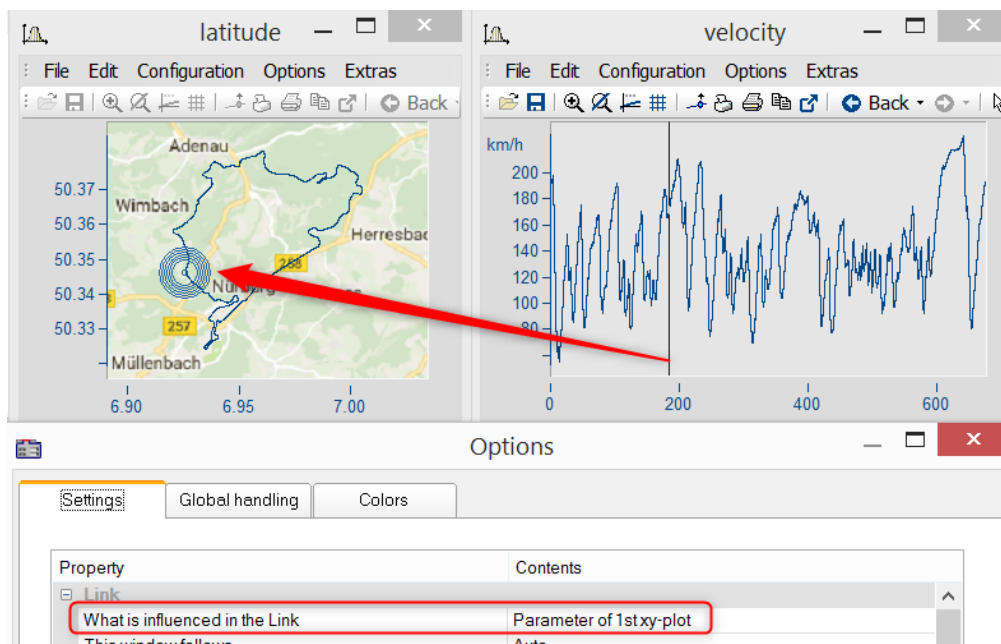
To undo an x-linkage, simply click on the button again.

Connect with XY

When linking XY and normal data sets, it may be necessary to specify which component of the XY data is to be linked.

This is configured under "Options"/"Presettings"/"Link"->"What is influenced in the link". In the following example the GPS channels Longitude and Latitude are used to display the speed with the position on a map.

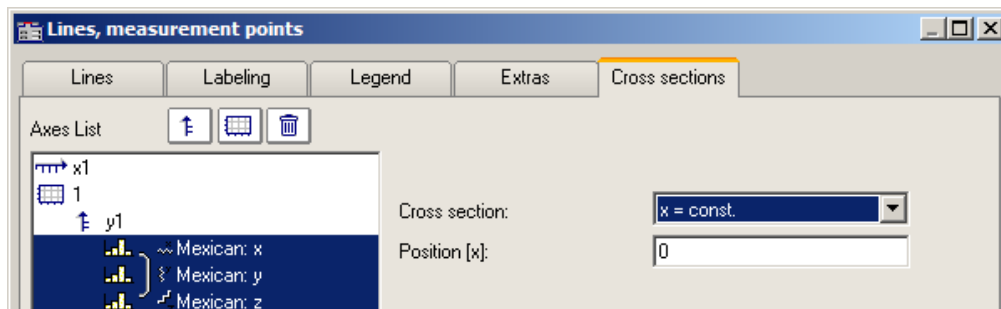
For this you have to set "[What is influenced in the link](#)" -> "[Parameter of 1st xy-plot](#)" ⁸⁸⁶.



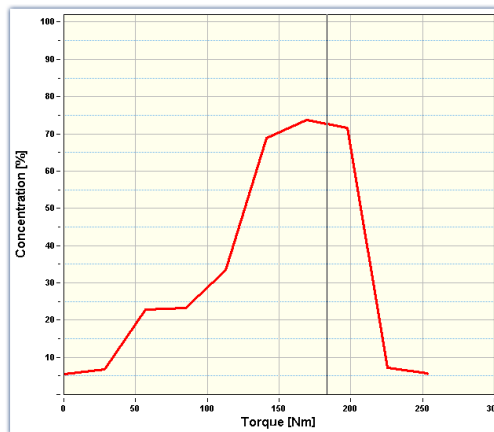
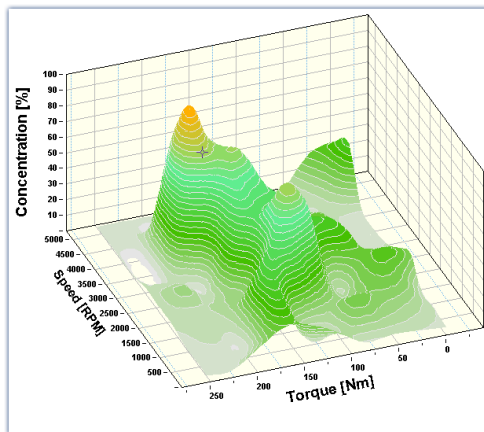
Connect with 3D

The ability to link curve windows together is also provided for 3D-displays. In this case, a 3D data set can be simultaneously displayed in 3D-display and as a cross-section with x or z as the constant, and both curve windows can be linked together.

To do this, load the 3D data set in a curve window with standard display style, parallel to the curve window with the 3D-display, and then open the *Lines*-dialog either from the coordinate system's context menu or as the menu item *Configuration / Lines*. On the page *Cross sections*, you can select from the dropdown-list whether to keep the x- or the z-coordinate constant. Then confirm by clicking *OK*.



In one of the two curve windows, activate the *Connect*-function, so that the cursor changes shape. Then move the changed cursor into the other curve window's coordinate system and click on the left mouse button. In the curve window with the 3D-display, a crosshairs appears at the upper left edge. Move the cursor to the crosshairs, click the mouse on it and move the mouse while holding down its button. In the curve window with the standard display and the cross section, there is a vertical line. The position of the vertical line changes in response to the position in the 3D-display.



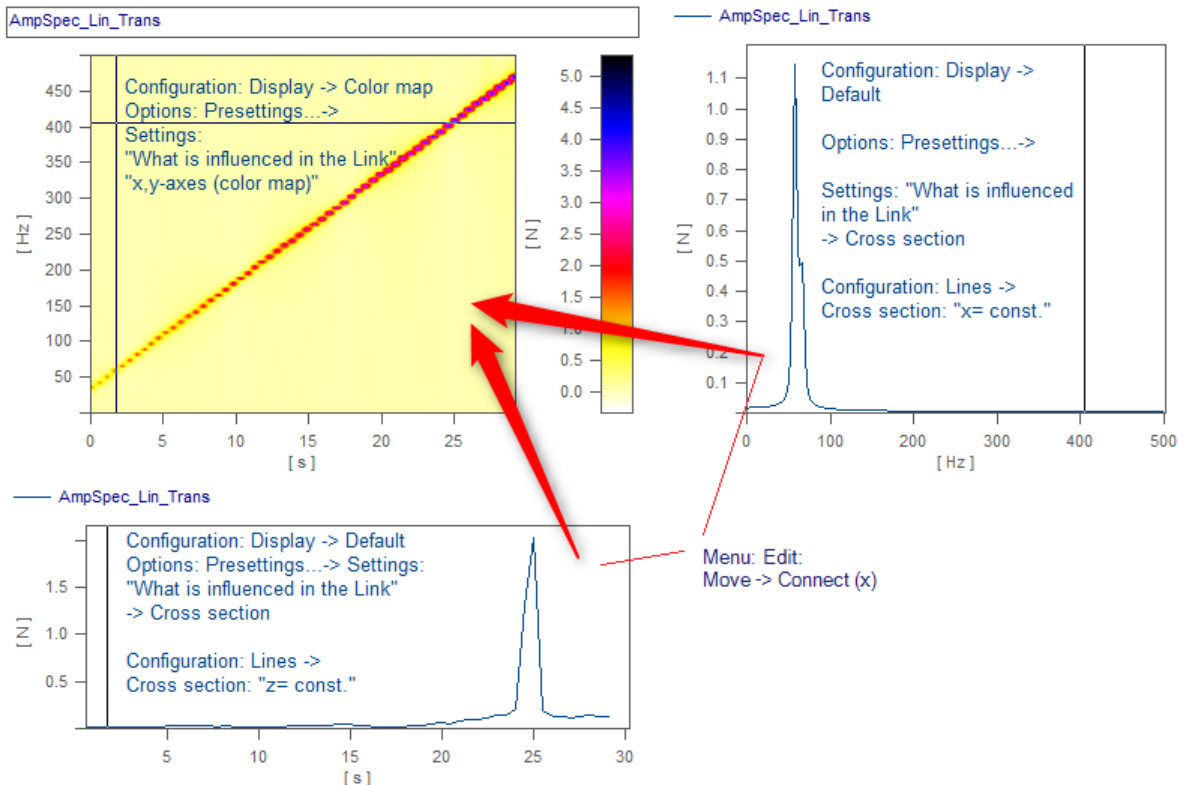


Example

Example with two cross sections

Color maps can also be linked in the y-direction. For this purpose, in the [Options for the curve window](#) ⁸⁸⁵, under *Settings*, it is necessary to set the *Link*-property "What is influenced in the Link" accordingly:

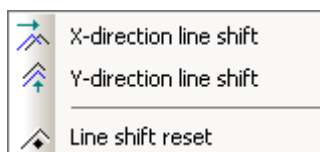
- On the color map: "x-, y-axis (color map)"
- On the right and bottom curve window: "Cross section"
- All curve windows show the same amplitude spectrum from example project "FA70 Spectral analysis"



The linkage is made by means of *Edit> Move> Connect (x)* by using *Drag&Drop* in succession from the lower right window into the color map.

11.6.6.7.2 Line Shift

Using this function, any lines selected in the Select-mode, or all lines at once can be moved in the either the x- or y-direction.



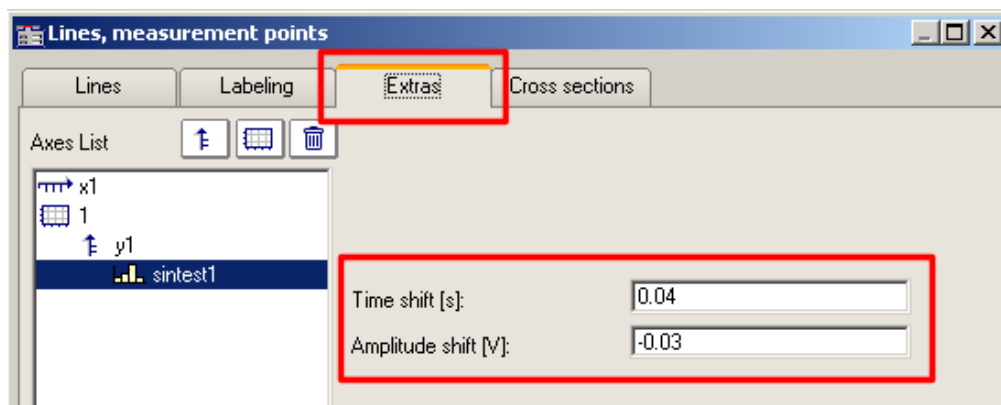
To do this, select one of the two functions either using the menu *Edit/Move/Line-Shift* or from the *Line-Shift toolbar*.

With the cursor in the pertinent coordinate system, and its shape changed to a double arrow, the lines can be moved either horizontally or vertically, depending on the function, by moving the mouse while holding down its button. With this function, only the view changes, i.e. the data set remains unchanged and any changes made with *Line-Shift* can be changed back either using the *Back* function or using *Line-Shift Reset*. With *Line-Shift Reset*, all changes are reversed at once, while *Back* reverses the actions one step at a time.



Line-Shift of a selected line in x-direction

The parameters for Line-Shift can also be changed directly in the Properties dialog for the lines, under *Extras*. There, too, the changes can apply either to individual lines or to all of them. With linear scaling display, the parameters are simply added. By contrast, with logarithmic scaling display, the parameter's value is interpreted as a factor, where the value 1.0 stands for no shift, a value of 10.0 stands shifting by tenfold, and a value of 0.1 shifts the lines downwards one-tenth.



Changing the Line-Shift parameters

Reference

The Time-Shift settings can be saved with the ccv file, if the curve window option "[Time-Shift in der CCV](#)"⁸⁸⁷ has been set to "yes".

11.6.7 Ribbon

11.6.7.1 "File" menu

Menu item		Description
 Load 		Loads a curve window configuration from a file.
 Save as 		The curve window configuration (e.g. the axis scaling) is saved in a file.
 Transfer to imc FAMOS! 		The waveform in the curve window is transferred to imc FAMOS.
 Report Generator 		Calls the Report Generator. The report's layout can have your own personal, creative design.
 Print 		Prints out the curve window.
Printer setup 		Sets up the printer used for printing out the curve window's contents.
Overview Window 		Displays an overview window.
 Twin Window 		Generates a new curve window with the same appearance.

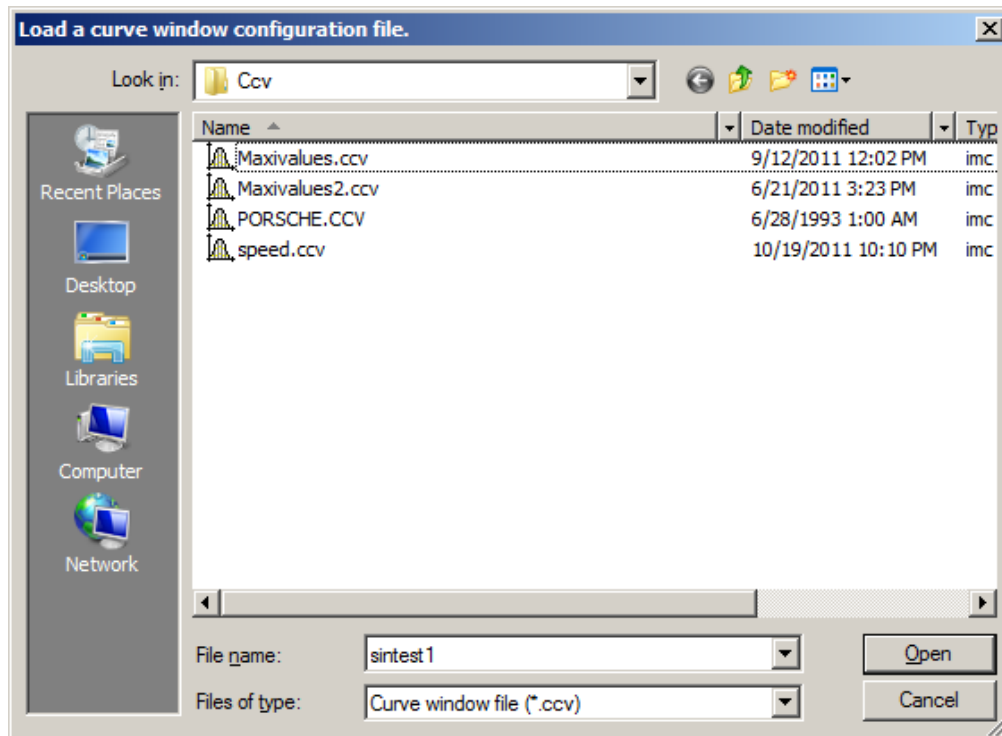
11.6.7.1.1 Load configuration

Function

You can load a curve window configuration directly from a file. A curve window configuration includes various attributes of the window such as size, axes scaling and names of additional curves in the window.

Mouse Operation

- Select *File/ Load...*
- A dialog appears in which the file can be specified whose configuration should be loaded.

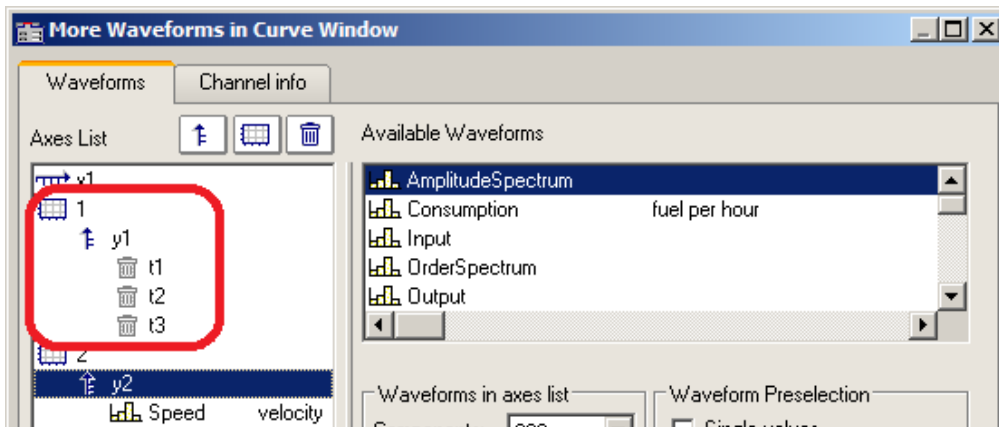


- Select file format "CCV".
- Select the desired file and end the dialog by clicking on *Open*.

Remarks

- The standard file name extension is "CCV".
- The directory used for loading configurations is read from the Windows System Registry when the program is started. In imc FAMOS, the folder can be handled via the menu item Extra / Options... .
- Once a file has been saved or loaded successfully, the directory involved is used for any subsequent loading or saving operation.
- The curves displayed in the window and their corresponding data sets are not contained in the configuration.

- When a curve configuration is loaded, an attempt is made to find all of the waveforms which were saved to the configuration. Any of these which are not found are marked as invalid:



- The names provide an indication of what waveforms are expected for the display. (When loading configurations from imc FAMOS 2.0, the system behavior of that time is reproduced and the waveform is deleted.) Configurations having multiple curves should be used only for stock tasks, where the names of the waveforms stay the same. For making changes, see Chapter 4, kit function *CvReplaceChannel()*.
- The name of the "basis waveform" (i.e. the first data set to be opened in the window) of a configuration is also saved.
- The position of the window does not change when a configuration is loaded manually. In automatic loading using a program or a sequence, the window is moved to the position saved in the configuration file. This does not apply to curve windows in dialogs.

Note

By means of [Copy & Paste](#) ^[904], you can directly transfer a configuration from one curve window to another.

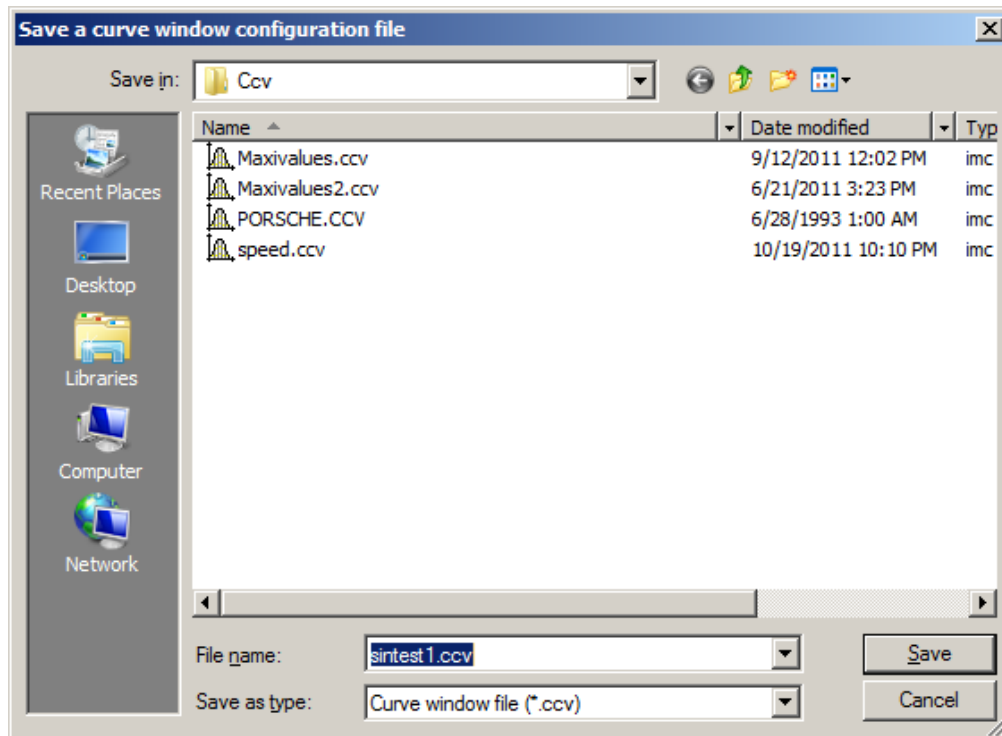
11.6.7.1.2 Save configuration

Function

The curve window configuration can be saved in a file. This configuration includes curve window attributes such as window size, axes scaling and the names of additional curves in the window.

Mouse Operation

- Select menu item *File / Save as...* in the curve window menu.
- A dialog appears for selecting the file in which the configuration should be saved.



- Enter the name of the file in the input box.
- Select the file format "CCV".
- End the dialog by clicking on *Save*.

Remarks

- If the file into which you wish to save the configuration already exists, the program asks if it should be overwritten.
- When a configuration is saved with automatic axes scaling set, different values may be written to the axes when this configuration is loaded for a new waveform. An appropriate scaling is defined automatically for the new waveform.
- It is useful for some applications to save configurations with automatic axes scaling. This is recommended especially when the expected signals fluctuate strongly in their range of values.
- The standard file name extension is "CCV".
- The directory path for projects is the project folder, or the default CCV folder from imc FAMOS Options.
- Once a file has been saved or loaded successfully, the directory involved is used for any subsequent loading or saving operation.
- The waveforms displayed in the window and their corresponding data sets are not contained in the configuration.

- The name of the "basis curve" of a configuration is also specified.

Note

By means of [Copy & Paste](#)^[904] you can directly transfer a configuration from one curve window to another.

11.6.7.1.3 Transfer to FAMOS!

"*Transfer to imc FAMOS*" creates a copy in imc FAMOS of the data sets displayed in this curve window. After the transfer, the data sets appear in the imc FAMOS Variables list, where they are available for working with. The transfer is normally only helpful if the curve window belongs to another imc program than imc FAMOS, such as imc STUDIO.

The settings made in the dialog "*Options*" > "[Transfer options](#)"^[891] for naming variables in imc FAMOS are also valid for this operation. A command can be specified to be executed after the variables have been transferred to imc FAMOS, for example, a command to call a sequence for analyzing the transferred waveform.

Note

Variables will be overwritten

Existing variables in the Variable list will be overwritten without notification.

Reference

- Transfer of waveforms or sections of waveforms can be made from a measurement value window. Refer to the section "[Context Menu in Measure Window](#)"^[818] for more information.
- Waveforms can also be transferred to imc FAMOS using [Drag&drop](#)^[904].

11.6.7.1.4 Report generator

This lets you open the Report Generator, which can be used to assemble your curve windows in a report. For more information, please see the Report Generator documentation.

11.6.7.1.5 Print

Function

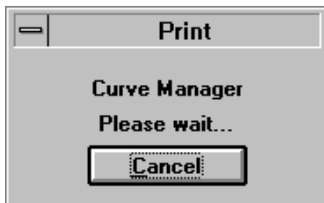
Hardcopy of a curve window can be created on any printer or plotter supported by Windows. The hardcopy is always made to an entire page. The graphics are centered on this page.

The graphics are sent to the printer which was specified in the dialog belonging to the curve window's menu item *File/ Printer Setup...* Various printing preferences can be specified in the *Opt./ Clipboard settings...* The size, font, width, etc. correspond to these settings.

Mouse Operation

To print the curve window, choose *File/ Print*.

In consequence, a small info-dialog appears which indicates that the printout is in progress. The process can be canceled by clicking on the *Cancel* button.



Normally, you have to allow for a delay before a print job is actually stopped.

Remarks

- Use the MS-Windows Control Panel to make settings for the printer.
- Additional parameters can also be specified for the printer here, e.g. printer resolution, paper size and format (portrait/landscape), printer memory allotment, etc. To do this, use the curve window's menu item *File / Printer setup...*
- The quality increases with the resolution. Graphics only look good, of course, when they are printed with high resolution. Note however that the higher the resolution, the longer it takes to print. In fact, the time to compute a printer job increases with the square of the resolution.
- Select a low printing resolution if you value speed of printout more than quality.
- The current time is used as the "Created/ Modified" time if the waveform does not have one already.
- To combine text and graphics in a desktop-publishing program, send output to the Windows Clipboard instead of using the Print option. See "[Clipboard](#)"⁸⁵⁶.
- If the printer is either not correctly defined in the MS-Windows system or switched off, or if it has no paper, an error message will appear.
- If you wish to create a more detailed layout, use the imc Report Generator. See chapter "[Report Generator](#)"⁸⁷¹ in this document for more information on this powerful program.
- The time format (date and time) can be changed in the Control Panel under *Regional Settings*.

Reference

See "[Clipboard Settings](#)"⁸⁷⁸, "[Printer Setup](#)"⁸⁷³

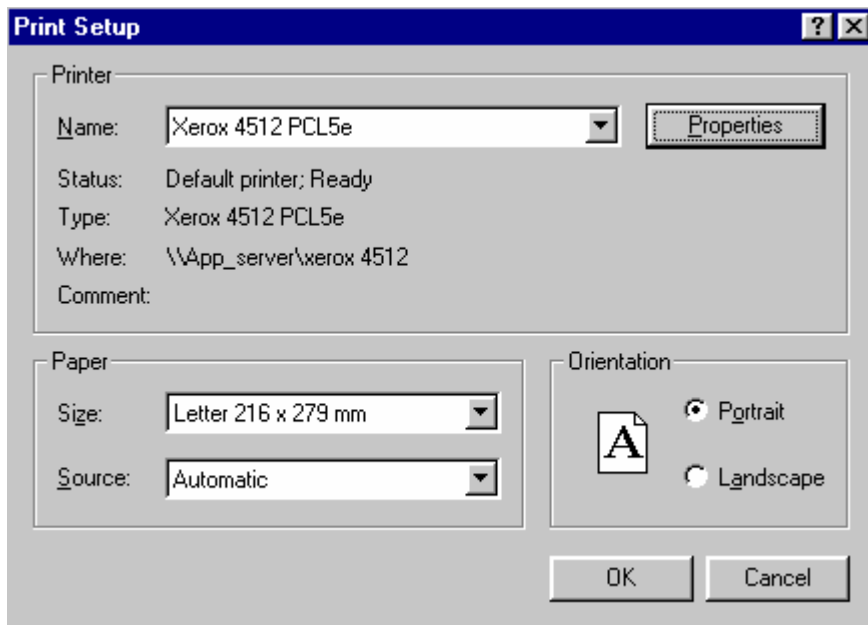
11.6.7.1.6 Printer setup

Function

This is where the printer is configured, which is used to produce the hard-copy output from the curve window.

Operation

Select menu item *File/ Printer setup...* in any curve window. The standard Windows dialog for selecting and configuring a printer.



This dialog's appearance varies according to what Windows version or printers are involved.

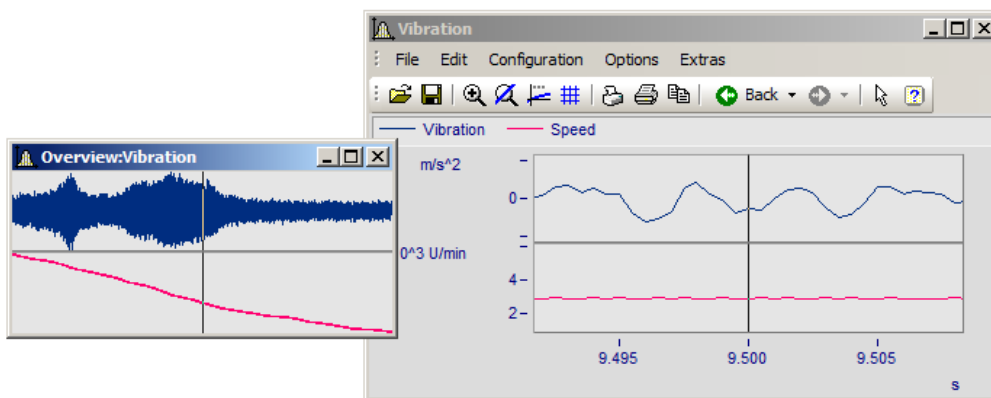
The settings apply to all of the computer's imc curve window's and remains operative upon the program's next start.

11.6.7.1.7 Overview window

Function

Whereas a curve window can display any portion of a waveform, an overview window, which is shown in conjunction with a curve window, will only display the entirety of any waveforms which are represented at all in the curve window. The curve- and overview window are linked with each other in the x-direction, in that the reference line of each window marks the same x-coordinate. The overview window is especially useful for:

- Keeping track of a position in convoluted or long curves (especially during zooming), or distinguishing between different axes lengths when several waveforms are displayed together.
- correctly associating a displayed section with its entire curve when scrolling in the curve window in the x-direction.



Mouse Operation

- To open an overview window, choose *File/ Overview Window*. A check mark appears beside the selected option.
- Zoom in on one region of the curve window and navigate through the signal using the reference line in the overview window.
- Click again on this option or select *Close* in the system menu of the *overview window* to close it.

Overview windows are independent windows which behave almost like curve windows. However, each overview window is assigned to a curve window and cannot exist without the curve window; the overview window also contains the same menu as the curve window. All settings for configuration and display of waveforms in curve windows can be made for the overview window. It is thus possible to display a curve together with variously strongly zoomed views by opening an additional overview window.

Remarks

- The overview window is only recommended if the range displayed in the curve window can be displayed fully in the overview window.
- When the curve window is reduced to an icon, the overview window is not visible.
- The display style used in the overview window can be changed and zoomed as desired.
- If an overview window and a measurement value window are displayed for the same curve window, the measurement value window should be closed when the zoom range is moved in the overview window. This reduces flickering while the curve window is being updated.
- The title of the overview window is composed of the string "Overview:" followed by the name of the corresponding curve window. This allows easy identification of the windows.

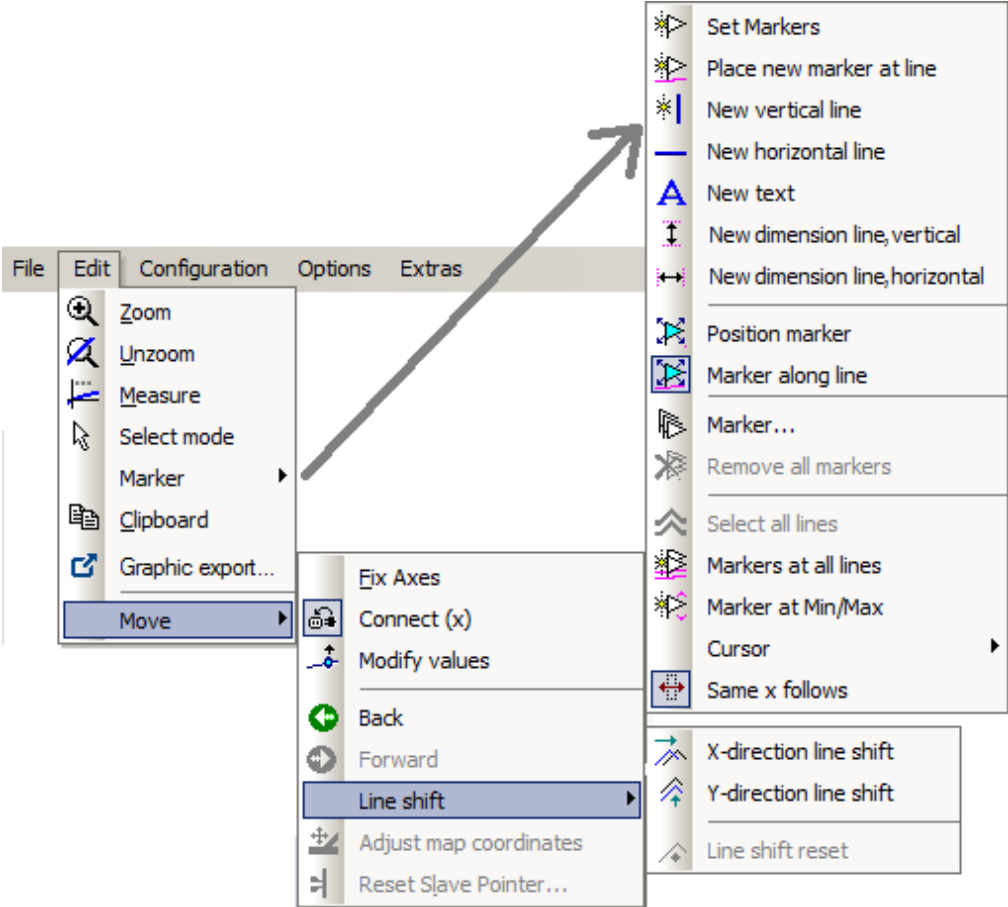
11.6.7.1.8 Twin window

Using this item you can create an identical copy of the curve window. Subsequently, both windows can be configured independently of each other.

Possible applications:

- During an online measurement, the measured data can be displayed both as a curve plot and as numerical values ([Last Value as Number](#)).⁶⁶⁵⁾
- Display of a Waterfall graph in 3D, or a color map.
- Simultaneous display of the exact same data both in an overview and zoomed.

11.6.7.2 "Edit" menu

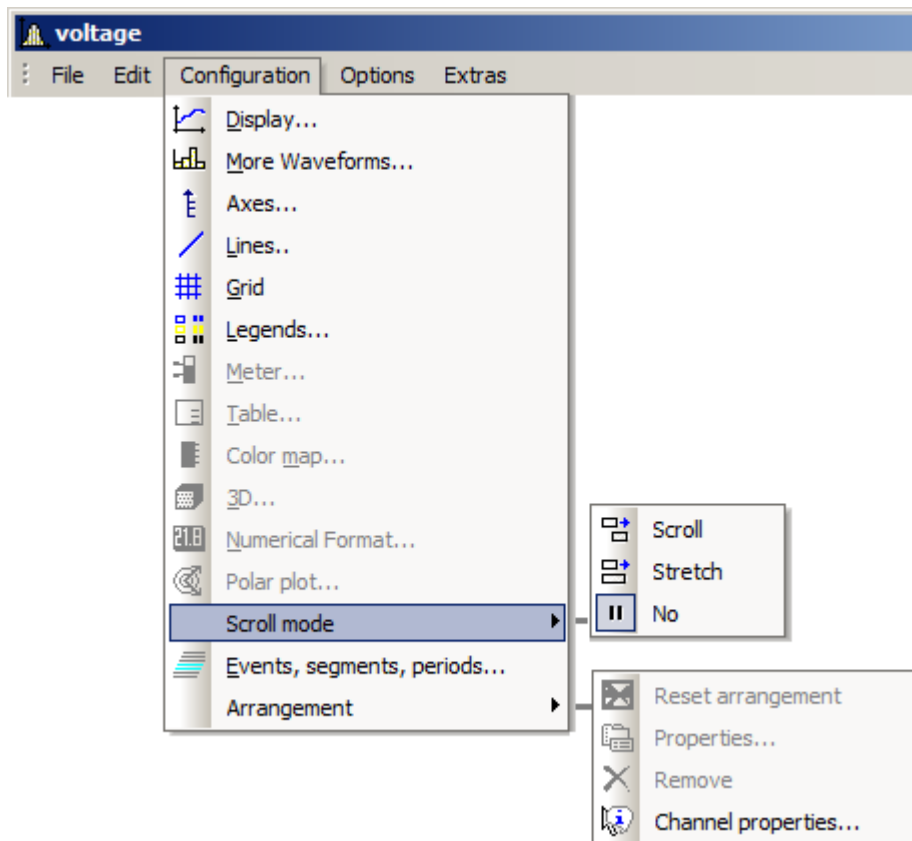


Menu item	Description
 Zoom ⁸¹¹	Enlarges a section of the curve window.
 Unzoom ⁸¹²	Displays the entire curve.
 Measure ⁸¹³	A measurement value window and measurement cursors for measuring curves are displayed.
 Select-mode ⁸³⁰	In this mode, it is possible to select and edit legends, coordinate systems, axes, curves and markers.
Marker ⁸³²	Proceeds to the list of marker functions.
 Clipboard ⁸⁵⁶	Copies the graphics in the curve window into the Windows clipboard.
 Graphic-export... ⁸⁵⁶	The graphic of the curve window is saved as image or as a PDF file.
 To MS WORD ⁸⁵⁸	The curve window is embedded as an OLE object in Microsoft WORD.
Move ⁸⁶¹	Proceeds to a submenu listing various curve viewing functions.

 [Reference](#)

Another possibility to navigate through the curve window is provided by the '[Axes Navigation Bar](#)' ⁹⁰⁵.

11.6.7.3 "Configuration" menu



Menu item	Description
 Display ⁶⁴⁹	The structure of the curve window and other attributes, such as the "Freeze" mode and time/date display can be selected here.
 More Channels ⁷²⁵	Selects additional waveforms to be displayed in the current curve window.
 Axes ⁷³⁶	Sets x- and y-axis parameters.
 Lines ⁷⁵⁵	Sets line (measurement cure) parameters.
 Grid ⁷⁹⁰	A grid can be added to the coordinate system.
 Legends ⁷⁶⁸	Sets the display of legends in the curve window.
 Meter ⁶⁷⁶	Sets properties of bar meter display.
 Table ⁶⁷⁰	Sets properties of tables display.
 Color map ⁶⁸⁵	Sets properties of color map display.
 3D ⁶⁹⁷	Sets properties of 3D display.
 Numerical format ⁶⁶⁵	Sets properties of the display mode "Last value as number".
 Polar plot ⁶⁹⁸	Sets properties of polar plot.
Scroll mode ⁸⁰⁰	In the curve window, a particular section of the data set can either "scroll past" or continuously "grow" from the beginning.
 Events, segments, periods ⁸⁰¹	Selects the display of individual events (in imc STUDIO: trigger events), individual segments (e.g. spectra or rows in a matrix) or even periods (period comparison).

Menu item	Description
Arrangement 	Restores the curve window arrangement, access to properties and deletion of selected objects.

11.6.7.4 "Options" menu

Menu item	Description
Clipboard settings 	Settings for graphics created for output or export with the clipboard are specified here.
 Colors 	Changes the colors in the curve window.
Presettings 	Presettings such as the standard configuration directory and drives for storage of temporary files are made here.
 Printer preview 	This function makes it possible to switch between the Print Layout view and Normal view. It is used exclusively in curve windows which are embedded in the Data-Browser's Report view.

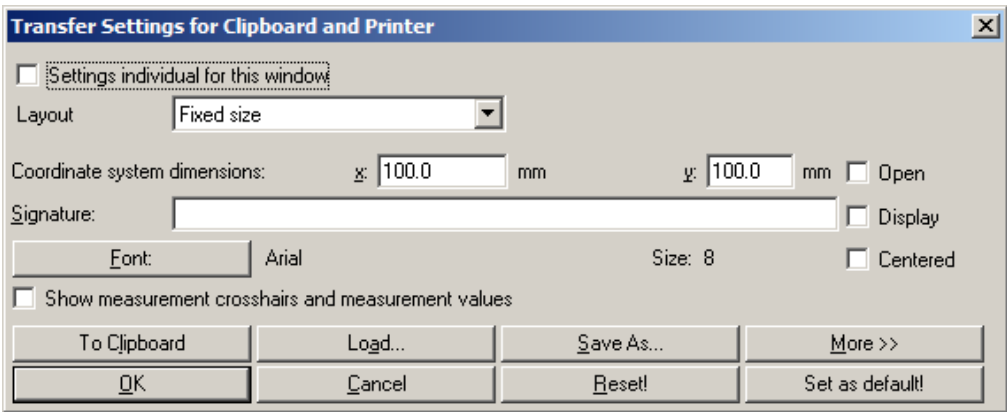
11.6.7.4.1 Clipboard settings

Function

The graphics in the curve window can be documented in professional quality. Graphics can be sent to the clipboard as vector graphics, for further processing by any text processing, graphics- or desktop publishing programs. The graphics can also be sent to the Report Generator for output in a report designed by the user. The user can specify how graphics are generated by selecting the font and line width in the clipboard settings.

Operation

Call menu option *Options/ Clipboard settings...* in the curve window menu. The following dialog appears:



The settings are applied when the item [Clipboard](#)  in the *Edit* menu is called.

Settings	Description
<i>Settings individual for this window</i>	Here you can set whether the clipboard settings only apply to the current curve window or for all curve windows.

Settings	Description
<i>Layout</i>	Proportions as on screen (the default setting!) saves the curve window to the clipboard with the proportions which accord with the settings currently applied on screen. The layout Coordinate system size creates a graph in which the coordinate system exactly applies the x and y values specified by the setting <i>Dimension</i>. Observe also the notes on the Table ⁸⁷⁹. Total size creates a graphic in which the outer edge of the curve window is determined by the x and y values.
<i>Coordinate system</i>	The width and height of the coordinate system, stated in mm, are set in the text boxes appearing in the dialog line <i>dimensions</i> . If the lines are particularly thick, the size is calculated from the middle of one line to the middle of the next line. Axis labeling of the coordinate system is not included in these dimensions, they are drawn outside of the actual coordinate system. Dimensions should be at least several mm, but no larger than the page.
<i>Open coordinate system</i>	This option causes curves in a standard display (y-axes not stacked) to be drawn in an open coordinate system, with no upper and right-hand borders. The curve is thus less crowded in the display. When this option is not selected, all lines in the coordinate system are drawn.
<i>Caption</i>	The user can specify a text to appear below the x-axis. This text may contain up to 60 characters. It will be displayed, along with the date and time, only when the option <i>Show</i> is selected.
<i>Centered</i>	When this option is selected, all labels (as much as possible) are centered under the ticks. The effect is most noticeable for the labels at the edges of the axes, at the far right and left of the x-axis and at the top and bottom edges of the y-axis. When the type is not centered, these labels are often drawn so that they end at the edges of the coordinate system.
<i>Show measurement cursors (crosshairs) and measurement values</i>	When a measurement value window is present when graphics are generated, and this option is active, the measurement cursors are included in the coordinate system display. The measurement values of the cursors are displayed below the x-axis.
<i>Font</i>	Select here the font for the axes scaling. The MS-Windows standard dialog appears when the "Font" button is selected. The dialog shows all fonts available for the printer currently set for the curve window. Select a font, the size in points and any other features desired, e.g. bold type. A font size of 10 to 12 points is generally very legible. TRUETYPE fonts are preferred since they can be scaled more easily.
<i>To clipboard</i>	Click on this button to copy the contents of the curve window with the current settings to the clipboard.
<i>Reset!</i>	When the <i>Reset!</i> button is selected, the controls of the dialogs are set to default values. Controls accessed with the <i>More >></i> button are also reset.
<i>Save as</i>	A dialog appears for saving the contents of the <i>Clipboard settings</i> in a file.
<i>Load</i>	Once the contents of the dialog have been saved in a file, this file can be loaded to initialize the dialog with the settings in the file.

Note

It is possible to achieve an exact scale of units/cm by means of the combination *Layout: Coordinate system size* and an appropriate count of [Ticks](#) ⁷³⁹ along the X- and Y-axes. Note that due to particular presets in the target program, the size of the graphic may not be equal to 100%. For example, after insertion into a WORD document, it is necessary to specify 100% for the *Size and Position* (in the graphic's context menu).

More >>

The dialog box is enlarged when this button is selected. Additional, less frequently used settings for the graphics configuration are made here. The dialog appears as follows:

Transfer Settings for Clipboard and Printer

☐ Settings individual for this window

Layout

Coordinate system size

Dimensions

Width [mm]: 100.0

Height [mm]: 100.0

☐ Coordinate system open

Signature:

☐ Display

Font:

Arial

Size: 8

☐ Centered

☐ Show measurement crosshairs and measurement values

To Clipboard

Load...

Save As...

OK

Cancel

Reset

Set as default!

Width of y-scale:

☒ Auto

Height of x-scale:

☒ Auto

Main tick size:

2.0

 mm

Inner: 0.0

 mm

Secondary tick size: 0.0

 mm

Inner: 0.0

 mm

Symbol size:

3.0

 mm

Line width:

Curves

3

Coord. System

5

Main grid

1

Sec. grid

1

Cursor

1

Line style:

3D z-axis:

Angle: 30.0

 Degrees

Length: 20.0

 mm

Small font:

Arial

Size: 7

Settings	Description
Width of y-scale	The width of the y-scale is set automatically, if the checkbox appearing after <i>y-scale width</i> is marked. To specify a different width manually, deactivate the check box <i>auto</i> . A text box appears in which the width should be specified in mm. Make sure that the specified width is large enough for the selected font. The width is calculated from the middle of the line enclosing the coordinate system.
Height of x-scale	The height of the x-scale is set automatically, if the checkbox appearing after <i>x-scale height</i> is marked. To specify a different height manually, deactivate the check box <i>auto</i> . A text box appears in which the height can be specified in mm. Make sure that the specified height is large enough for the selected font. The height is calculated from the middle of the line enclosing the coordinate system.
Size of ticks	Specify here the length of the <i>main</i> and <i>secondary ticks</i> . The length is composed of sections: length within and outside the coordinate system. Specify the total length and the portion <i>inside</i> for the main ticks and the secondary ticks, respectively. The lengths are specified in mm with up to one digit after the decimal; a length of zero is also possible.
Symbol size	Lines can also be distinguished by accompanying symbols, see the menu items Configuration/ Axes ⁷³⁶ and Configuration/ Display ⁶⁴⁹ . The size of the symbols is specified here. Enter the diameter of the symbols in mm, with up to one place after the decimal.

Settings	Description
<i>Line width</i>	The width can be specified for the following lines: curves, coordinate system, main grid, secondary grid, measurement cursor Any specified line thickness  for an individual waveform overwrite the settings saved in the Clipboard. Specify the line width with a setting between 1 and 100. A very fine line width can be achieved with the lowest settings. Line width depends on the resolution of the output device. Since the width is always entered in pixels (units of resolution), only a whole number of points of the output device can be specified. A line width of 1 may be a practically invisibly fine line on a high-resolution laser printer, but may appear like bold type on a simple matrix printer. Line widths are generally set to between 1 and 5.
<i>Line style</i>	Various line styles can be selected for the following: - coordinate system, main grid, secondary grid, measurement cursor The following line types can be selected: - broken, dense dots, coarse dots, dense dashes, coarse dashes, alternating dots and dashes The various line types help differentiated between main and secondary grids when both are drawn with thin lines, so as to avoid distracting from the displayed curves.
<i>z-axis angle</i>	The angle of the z-axis can be set to between 1 and 89 degrees for the waterfall diagram. The larger the angle, the steeper the z-axis climbs. An angle of 30 degrees is recommended.
<i>Length of z-axis</i>	The length of the z-axis can be specified in mm for the waterfall diagram, with up to one place after the decimal.
<i>Small font</i>	The font for the small print in the 1/3-octave/octave labeling of the x-axis can be specified. Click on the button <i>Small font</i> to prompt the MS-Windows standard dialog for selecting the font.

Remarks

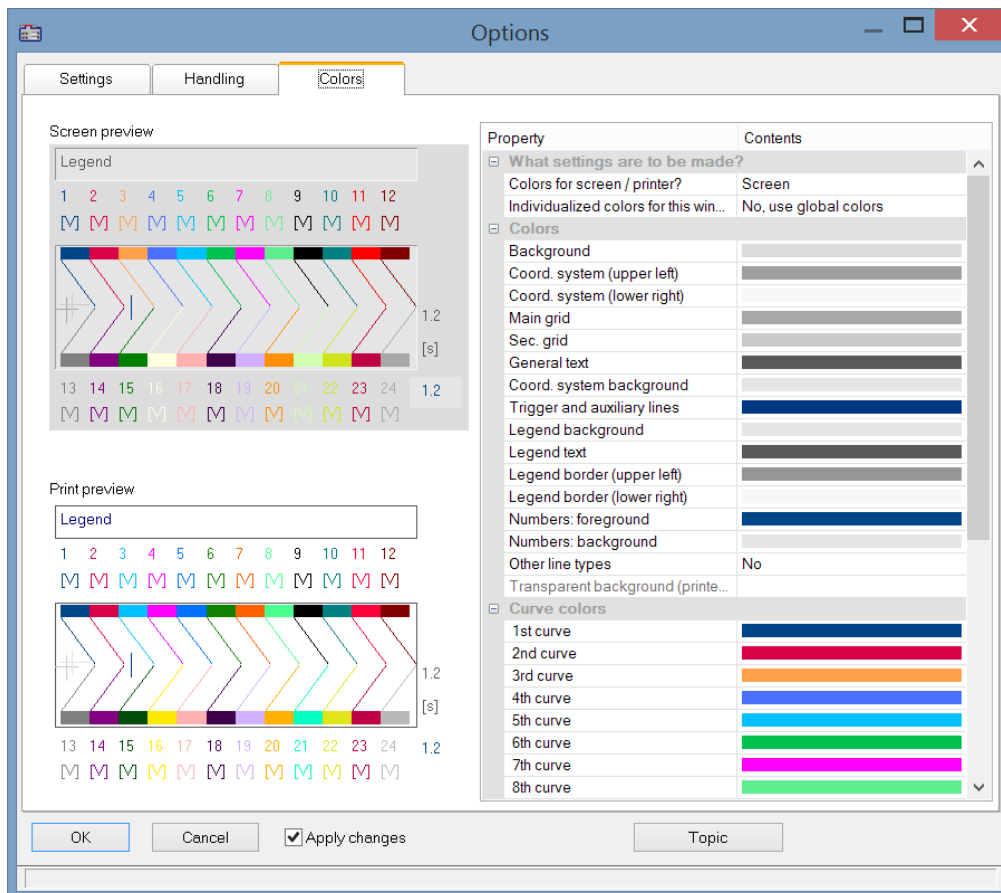
- The settings in this dialog apply to all curve windows.
- The fonts available for selection depend on the printer set for the curve window.
- When graphics are transferred to the clipboard, it is important that the printer set for the curve window is also the printer with which the graphics will later be printed. The meta file for the clipboard is created specially for the selected printer; if graphics are outputted on a different device, some fonts may not be available or scaling may change and then the graphics will appear in poor quality. Format settings such as landscape or portrait and the size of the page should also correspond. The metafiles are in some measure adapted to the output device.
- A metafile is created with the selected fonts and size for the curve window graphics when they are sent to the clipboard. If this metafile is later changed to a different size (e.g. because the graphics were shrunk by the text processing program after they were inserted in the clipboard), the font proportions may no longer be correct. The font may become too wide or too high. Use of truetype fonts may also have this result, since the Curve Manager only dictates the height of the font in points. Always create graphics in the size in which they will later be printed. If this is not possible, at least try to comply with the height-width proportions.

11.6.7.4.2 Colors

Function

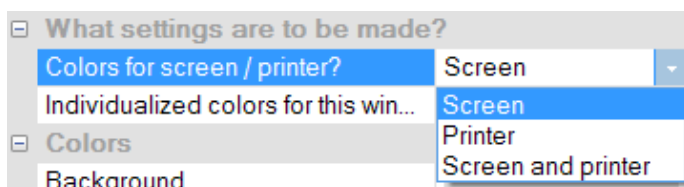
All colors used in the graphic interface of the curve window can be defined as desired in a special dialog. Single windows can use an individual color scheme.

The following dialog is used to define the colors:



Screen/Printer

A combobox is found at the top edge of the dialog, which contains the entries *Screen*, *Printer* and *Screen and printer*.



The colors can be changed from the default settings. The default colors for the printer are black/white. However, the screen colors can be applied for printouts, see below under [Copy..](#)

The printer settings additionally offer the option of leaving the background transparent. Besides saving ink, this has the advantage of keeping graphics objects located behind the curves visible.

Individualized colors for this window decides whether the colors are adopted as the default setting for all windows or not. This option can be set separately for the **screen** and **printer** selection. So it is possible to set colors for the screen individually but to use the global default settings for the printout.

Color graphic elements

A curve window contains the following graphic elements:

- background
- coordinate system and grid ...
- coordinate-system background
- general text
- legend...
- numbers...
- trigger line, auxiliary lines
- curves 1 to 12

Operation

Select menu option *Colors...* in the *Options* menu in the curve window.

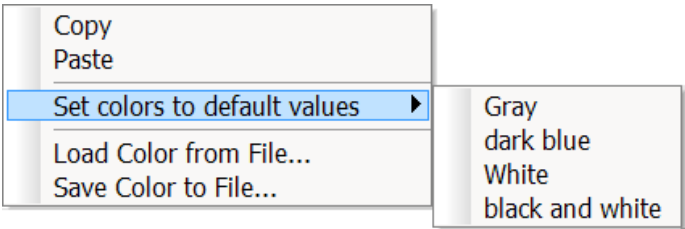
A dialog appears in which the colors can be defined. This dialog contains a list of the graphic elements in the curve window.

A display is found in the center of the dialog showing the colors currently defined for icons of the graphic elements. Colors are defined for one element at a time.

Select a graphical element from the list and set its color.

Context menu

Right-clicking the mouse in the dialog box calls the context menu:



Menu action	Description
Copy	The table including colors is copied to the clipboard.
Paste	The table values are pasted in the selected curve window in order to adjust the screen colors to the printer settings or for an individual window.
Set colors to default values	The colors are reset. There are several color schemes to choose from.
Load Color from File	The color settings for the monitor and for the printer are loaded from a file. This file must previously be created by using the menu option <i>Save Color to File</i> .
Save Color to File	Stores color settings in a file.

In the sequence, the colors can be loaded using the Curve Kit's function

`CwGlobalGet ("colors.printer.pattern")` or `CwGlobalGet ("colors.screen.pattern")`.

Remarks

- None of the colors produced by mixing and blending may be selected for lines and texts. If other color intensities are selected, the program will use the closest true color. This also applies to the coordinate system background color.
- It is recommended to select always colors which produce a good contrast. Yellow curves on a white background would be a very inappropriate combination.
- Background colors should not contain any noticeable pattern, to guarantee good contrast to the curves and lettering.
- Do not select a dark background color for printing. Laser printers are not designed to print large dark areas. A white background would be appropriate.
- Colors set in this dialog are effective for all curve windows.
- The colors for printing are also used when the graphics are transferred from curve windows to the MS-Windows clipboard, and when curves are sent to the Report Generator.
- The selected colors settings remain intact even after the program has been ended.
- Colors cannot be selected individually if more than twelve curves are displayed in the curve window. The colors are then repeated cyclically.



Reference

See also

"[Curve window colors on the Panel page](#)"  - Which colors are used for the on-screen display and which for the printout.

11.6.7.4.3 Presettings

Here you find more controls for the curve window's default settings, such as the font, axes, linkages to the curve window, etc.. By clicking on the column Contents, you can select from the possible options; see the arrow below.

Settings

Options

Settings

Global handling

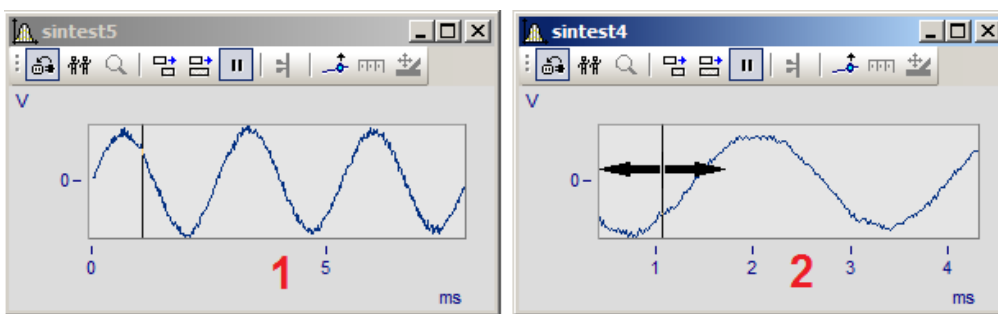
Colors

Property	Contents
Link	
What is influenced in the Link	Auto
This window follows	Auto
Upon changing the scale	Auto
Axis is adapted when the mouse is moved at the window edge	Auto
Axis is adapted if marker position gets close to border	Auto
Graphical shape of marker	Auto
Window	
Font selection mode	Auto
Font	Arial (8 pt)
Time-shift in the ccv	No
Optimize upon deleting data sets	Auto
Mouse	
Movement by means of mouse	Auto
Sound output	
Control	Auto
Start delay [ms]	0
Max. delay [ms]	0
History (moving back/forward through views)	
Maximum memory [MByte]	Auto

Presettings - Page: Settings

Link: The settings for the link affect how one curve window is linked to another one, for instance, one curve window's waveform in the time domain could be linked with second curve window showing GPS positions.

What is influenced by link	auto: depending on the display type, one of the following options is used x-axis: usually default Parameter of 1st XY-plot: e.g. for the position on a map⁷⁷⁸ x-, yAxis (color map): both the x-, and the y-direction are linked⁸⁶². y-axis: follows the y-direction in color maps Cross section: creates a cross-section referenced to the linked axis on the color map.
This window follows	auto: mostly Line follows Axis follows: The curve and the axis are moved, but the link pointer remains stationary. Line follows: The curve and axis remain stationary while the link pointer moves.
Upon changing the scale	auto: Stretching and compression of the x-axis; mostly: line remains at screen position Line follows: Line remains at screen position Line remains at position on screen: curve moves away behind it.
Axis is adapted when the mouse moved at the window edge	Affects curve windows where the line moves (in picture 2 below) auto: mostly "no", except with map⁷⁷⁸ yes: the axis of the curve window whose line is moved is compressed. no: no changes.
Axis is adapted if marker position gets close to border	Affects curve windows which move due to linkage (in picture 1 below) yes: Axis of linked curve window is compressed
Shape of position marker	auto: always line, except with map⁷⁷⁸ where a circle is used . Line: Vertical line



Link options

Window: In this section, set the font .	
Font selection mode	Auto: the font which is principally used whenever the curve window is opened. Individual for this window: Only affects the current curve window.
Font	All fonts used by Windows are available here.
Time-Shift in the ccv	Yes: Any shift made using the Time-Shift or Line-Shift function will be saved with a curve configuration file (ccv).
Optimize upon deleting data sets	With "<i>No: Lines and axes remain intact</i>", only a variable will only be removed from the curve window. The structure remains intact. This parameter can be set by the function <code>CwDisplaySet ("opt.on.delete", 0)</code> as well. With "<i>Auto</i>", the system behaves the same way as before, which means the associated axes are removed. If the deleted data set is the only variable in the curve window, the curve window closes.
Mouse: Behavior of the curve window when dragging the mouse while holding its button	
Movement by means of mouse	Auto: Moves the location of the data in the curve window in the X and Y directions, similarly to moving a map. Only in x-direction: Like <i>Auto</i> , only in the x-direction. No: The display is not moved. Instead, the data can be moved into a different coordinate system or curve window by means of Drag&Drop. (Default behavior up to imc FAMOS Version 7.0)
Sound output: Options for playback of a data set by means of the audio output.	
Control	Either the data is played from the player position (<i>auto</i>) or the last, currently recorded data is played back(e.g. during the measurement with imc STUDIO).
Start delay [ms]	Options for synchronization of online streaming data with imc STUDIO.
Max. delay [ms]	More info is presented here  .
History: Determines the maximum undo buffer depth for changed made.	
Max. memory [MByte]	Here you can determine the maximum memory allocated to the history of changes, which can be reversed by means of the undo  command.
Scroll mode: Options for automatic scrolling for streaming data with imc STUDIO	
Allow smart scrolling	<i>Yes/No</i>
Smart scrolling as of width	Stated in milliseconds. With this setting, you determine the scrolling behavior with streaming data during a running measurement.
Time lag with smart scrolling	In particular with the output of sound, these settings can be used to avoid lapses.
Jitter on the right edge	

Graphic export: Sets the maximum memory depth for the changes made.

Export optimization

Bitmap: A bitmap is created containing the **interior of the coordinate system**, with its curves and graph lines. This bitmap is then exported as a PDF. In consequence, when the graphic is complicated, having very many measurement points, graphics elements are not generated as a vector graphic, but only as a bitmap.

Markers, axis labels and legends are not affected and continue to be processed as text elements as vector graphics.

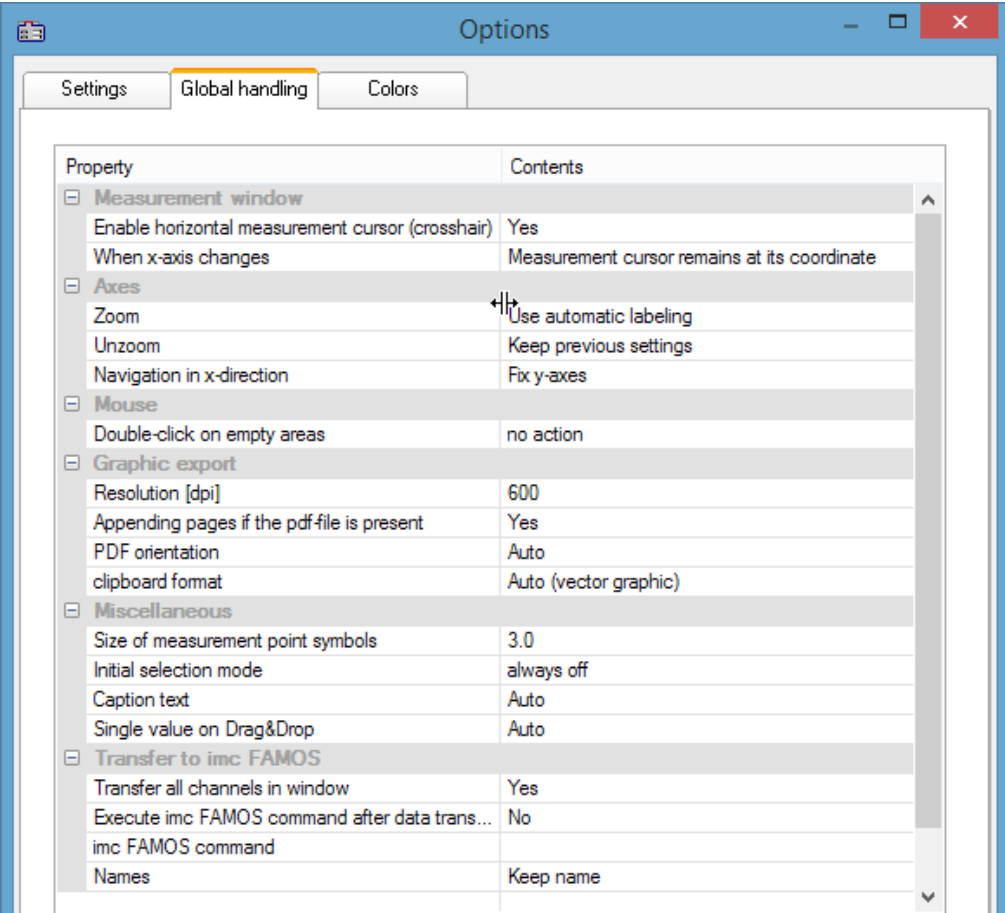
Vector graphic: The interior of the coordinate system generates a vector graphic.

Auto: Set to either "*Bitmap or Vector graphic*", depending on the implementation. This setting may change in future versions.

- The setting "*Export optimization*" is only useful if overall a vector graphic is to be generated in [Global handling](#)^[890]. If the setting there is for export of the curve window as a [Bitmap to PDF](#)^[890], this setting provides no benefit.
- This option is applied, besides when **exporting the curve window as PDF**, also when **printing out a Panel**, exporting the **Panel as a PDF** and when transferring to the **Report Generator**.
- **In a Report Generator printout file**, the setting Bitmap can cause very large DRB files, since bitmaps are not compressed there. But subsequently, compact and easily handled PDFs are generated from the DRB.
- This option applies **individually to each curve window** or ccv-file. It can thus be used to provide individualized modification of the behavior for particular curve windows, e.g. ones with complex graphics.
- When transferring a graphic to the **Report Generator**, any printer currently set for the Report Generator is taken into account. Thus, the coordinate system's bitmap is appears in shades of gray when a B/W printer is used. If this is not the desired behavior, because actually only a PDF is to be generated, a color printer (or if not available the XPDS printer) should be selected.



Global handling



Presettings - Handling

Measurement window	
Enable horizontal measurement cursor	yes/no: display of the horizontal measurement cursor (yes: crosshairs; no: vertical line)
When x-axis changes	Sets whether the measurement cursor remains at the data set coordinate or at the pixel position when the zoomed region is changed.
Mouse	
Double-click on empty areas	Enables Select-mode ^[830] to be activated by double-clicking on an empty space in the curve window.
Axes: Here, settings are made which affect the ticks on and at the ends of the axes, once zooming and or re-zooming has been performed.	
Zoom	Use automatic labeling; Place labels at end, use rounding;Place labels at end, use no
Rezoom	rounding; Keep previous settings
	See Axes Scale ^[737]
Navigation in x-direction	Determines whether after navigating ^[905] the Y-axis is permanent or is scaled automatically.

Graphic export: Settings for exporting the curve window as a graphic to the Clipboard	
Resolution [dpi]	Dots per Inch (150, 300, 600, 1200)
Appending pages if the pdf-file is present	If the PDF document already exists, it can be overwritten or supplemented with the curve window.
PDF orientation	Portrait or landscape
clipboard format	Vector- or pixel graphic
Create PDF files	Bitmap: The entire graphic is converted to PDF as a bitmap. Resolution in dpi. (Default before Version imc FAMOS 7.3) Vector graphic: All graphics elements which are not available as a bitmap are embedded in the vector graphic in the PDF. For color maps and 3D, the resulting bitmaps have 300 dpi, which are retained. However, the text elements in the curve window are processed to vector graphics. The system can do a search through the PDF for these by text search. The vector graphic as significantly better resolution at lower memory requirements for normal curve plots. BUT: for graphics with many vector elements (e.g. 10000 big solid dots), the PDF becomes unwieldy large an export is slowed down by an extreme amount. In this case, it is preferable to use the Bitmap setting. On an individual basis, the interior of the curve window can be generated as bitmap and the text elements as vector graphics. To do this, use vector graphics here and for Export optimization use "Bitmap" under Settings . For vector graphics, the Windows XPS printer is used. This is installed by the operating system and must be operational. Otherwise, the printer will need to be reinstalled via the PC's control panel settings. Here, a resolution of 600 dpi is always assumed; the stated resolution is ignored! Auto: When exporting from the Panel or Report Generator, imc FAMOS checks which method is more advantageous. For specific conditions, always select either Bitmap or Vector Graphic. From directly in the curve window, <i>auto</i> is either set to <i>Bitmap</i> or <i>Vector Graphic</i>, depending on the implementation. The settings are global and can also be set in imc FAMOS, under the menu item: "<i>Extra</i>" > "<i>Options</i>" > "<i>File - Save/Export: PDF</i>".
Miscellaneous	
Size of measurement point symbol	0.5 -10 mm. Size of the symbols  representing the measured points, e.g. squares, circles...
Initial selection mode	Default setting for Select-mode  .
Caption text	Auto: Name of first dataset. Filename: The filename of the loaded ccv file can be shown in the caption of a free-floating curve window.
Single value on Drag&Drop	With the option " <i>As horizontal line</i> ", single values are always displayed as a line, e.g. for display of value limits.

Transfer to imc FAMOS: A detailed description is provided [here](#).

Transfer all channels in window	You can transfer either all of the data sets or only the first one which has the window's title.
Execute imc FAMOS command after data transfer	The function/sequence set under " <i>imc FAMOS Command</i> " is executed.
imc FAMOS command	Command which is executed once the data have been transferred if the option " <i>imc FAMOS command after data transfer</i> " is activated.
Names	Variables names can be changed in order to avoid overwriting existing variables.

Remark: The presettings remain in effect after the program is closed.

Reference: [Colors](#).

Transfer Options

Waveforms displayed in the curve windows can be transferred directly to imc FAMOS. It is often useful to transfer data from imc STUDIO or user-made programs which use the curve window to imc FAMOS for special evaluation.

But even within imc FAMOS, the [measurement window](#) enables signal portions located between measurement cursors to be copied. Note that there is a way ([Name](#)) to automatically rename the portions, in order not to overwrite the original data.

Waveforms are transferred one by one to imc FAMOS. Once a waveform has been successfully transferred to imc FAMOS, it appears in the variables list.

For the purpose of transferring to imc FAMOS, it is also possible to select options via the [presettings](#).

Execute imc FAMOS Command after Data Transfer:

Immediately after all selected waveforms have been transferred, a imc FAMOS-command can be optionally executed. Any command which can be executed in imc FAMOS can be transferred. For example, a sequence call can be transferred, which imc FAMOS uses to execute the sequence. When the option "*Execute imc FAMOS command after data transfer*" is not selected, it is irrelevant what the user enters in the input box.

Names

Use this option to decide under which file names the data are stored in imc FAMOS after transfer. The following options are available:

Options	Description
Keep	The name of the waveform is accepted unchanged. This option is useful when data are transferred from another application to imc FAMOS and the same name should be used. Attention: Be careful when transferring measurement intervals of imc FAMOS curves!
'_' instead of first character	The first character in the name is replaced by '_' (underline).
'_' precede	A '_' is placed in front of the name. If the name is too long, it is truncated at the end.
'_' append	A '_' is added to the end of a name. If the name is too long, the last character of the original name is dropped.
Fixed Names	User-defined names can also be specified. This option is useful only when just one waveform is transferred. If the user attempts to transfer multiple waveforms with the same name, each waveform will be overwritten by the next one.

The option most appropriate depends on the application and the particular names. The options ' _ ' *precede* or ' _ ' *append* are recommended for transferring measurement intervals within imc FAMOS. Any significant characters in the name should not be changed. The transferred data should have distinct names.

Mouse Operation

- Select menu item *Options/Transfer options...* to set the options for transferring data. A dialog appears in which settings are made.
- Select menu option *File/ Transfer to imc FAMOS* in the curve window to transfer data to imc FAMOS using the specified options.
- In order to transfer a curve window section defined by the measurement cursors, select the context menu item [Send curve segment to imc FAMOS](#)⁸¹⁸.

Remarks

- If the imc FAMOS application hasn't been opened yet, it is automatically started when transfer to imc FAMOS is executed. The imc FAMOS.EXE file is searched for in the same directory as the IM7CUDAM.DLL file.
- If waveforms are to be transferred to imc FAMOS, it is recommended to end any sequences running in imc FAMOS before starting transfer.
- The transfer options remain intact after the program is ended.
- Transfer options are valid for all curve and measurement value windows.

11.6.7.4.4 Printer preview

This item is only enabled in the Data Browser's Report view.

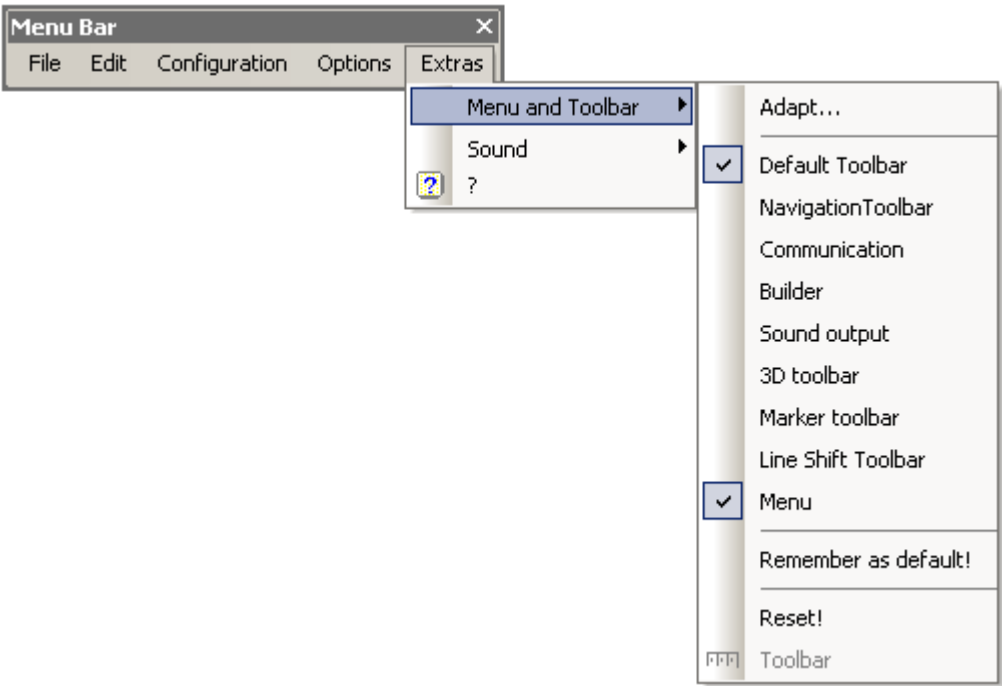
The *Print Preview* is activated by default for curve windows which are incorporated into a report in the Data-Browser. This means that the colors are displayed which the user set in the [Colors](#)⁸⁸²-dialog in the *Options* menu as the color scheme for the printer.

With this option, it is possible to switch to the monitor screen's color scheme.

11.6.7.5 "Extras" menu

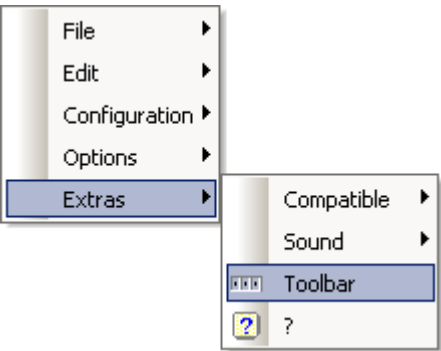
Menu item	Description
Menu and Toolbar ⁸⁹³	Options pertaining to the menu and toolbar.
Sound ⁸⁹⁶	Activates the sound output and access to the functions of the Sound-Toolbar.
?	Displays the curve window's Help.

11.6.7.5.1 Menu and Toolbar



Menu item	Description
Adapt... 	Dialog for adapting (making one's own arrangement of and settings for) the curve window's menu and toolbar.
Standard Toolbar, Navigation Toolbar, ...Menu	These toolbars or even the menu bar can each be activated separately.
Remember as default!	The arrangement of the menu and the toolbar is saved as the valid default arrangement. In this way, the same settings are used even for a different, new curve.
Reset!	The menu and toolbar are reset to their initial, factory-set state.

Toolbar

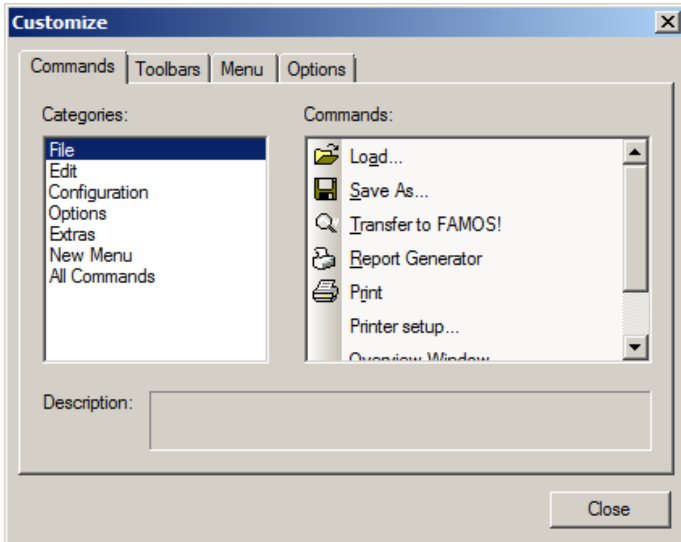


As a curve window integrated in the Data-Browser's Report view, the entire Toolbar only appears when a curve window is selected. To cancel this behavior so that the Toolbar does not appear, the function "Toolbar" can be selected. This is only available from within the Data-Browser. To make the toolbar visible again, select "Toolbar" under "Extras" in the curve window's context menu.

11.6.7.5.1.1 Adapt / Customize

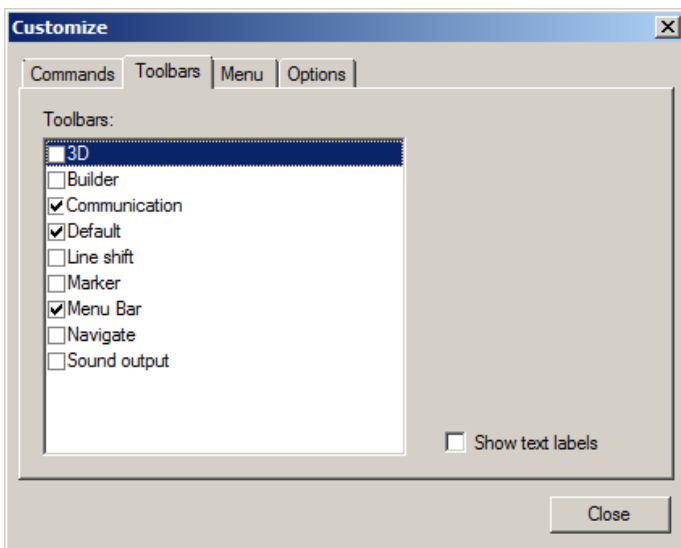
The curve window's menu and toolbar can be adapted. Right-clicking in either the menu or the toolbar opens the context menu. Alternatively, select the menu item *Extras / Menu and Toolbar*.

The menu item *Adapt...* opens the following dialog:

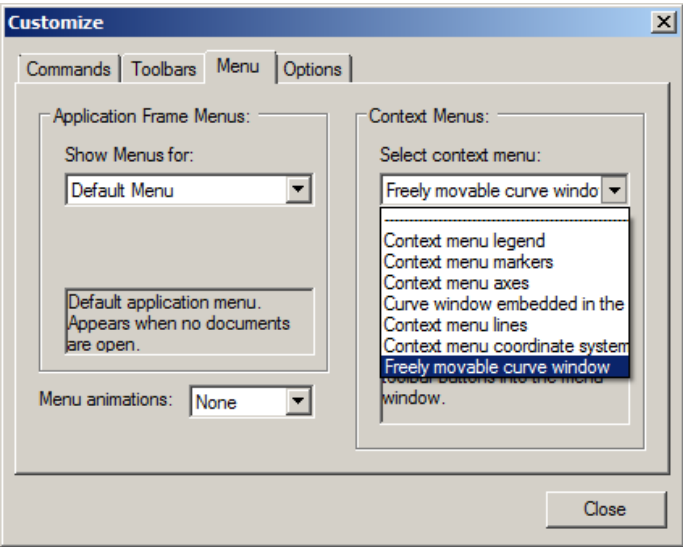


A command can be dragged from the list at right and dropped on a menu or toolbar. Pop-down menus open by themselves when the mouse is moved over them. The list at left contains the different command topics.

The dialog below offers the additional option of showing or hiding the various toolbars:



The curve window has a context menu. This, too can be edited:



Select the display type which you wish to process.

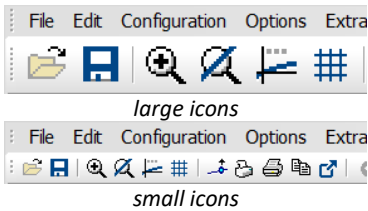
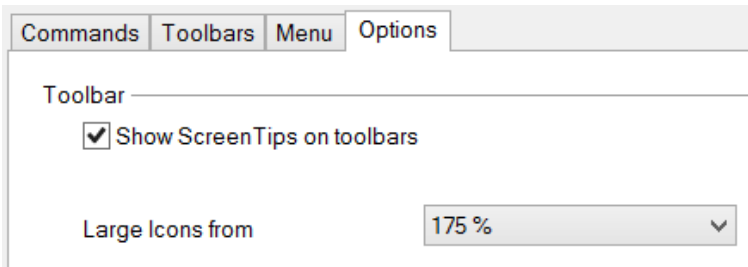
- normal curve window which is free-floating and freely movable and can also be maximized (popup-window)
- a curve window without title bar (child window) embedded in a dialog.

While editing menus, the curve window can't be operated.

While the dialog *Customize* is open, all menu items as well as toolbar elements can be moved around by the Drag&Drop technique. Note that moving an element to the "outside" means deleting the element.

On the page "**Options**", there are two options for the display of the *Toolbars*:

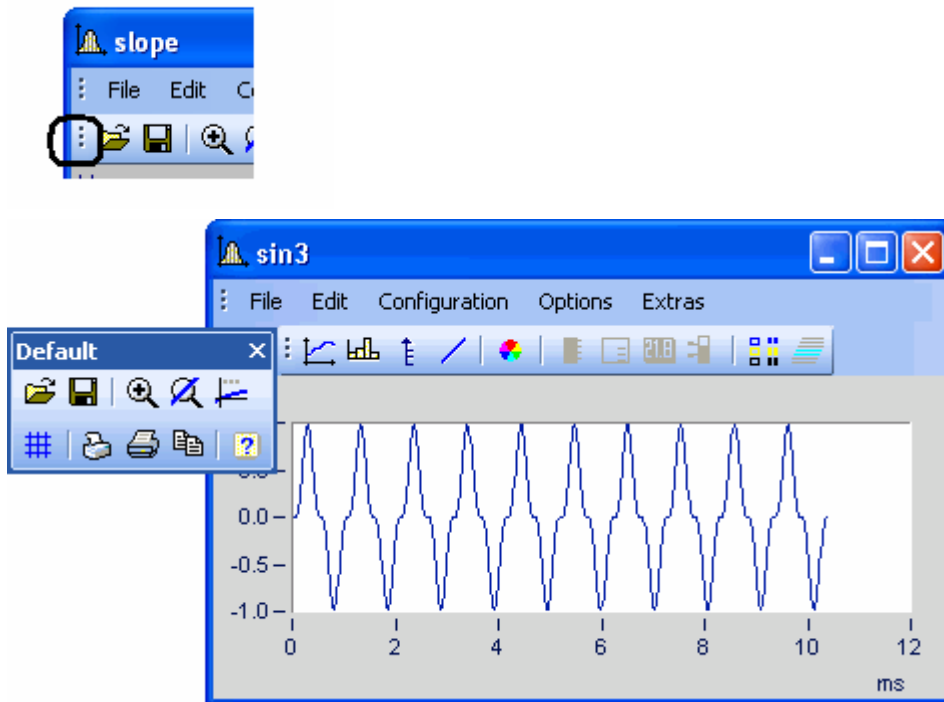
Options	Description
Show tips on toolbars	A description of the icon is displayed when the mouse is moved over the window.
Large buttons	For clearer display at fine screen resolutions such as 4K, the icons can be automatically enlarged from a certain scale.



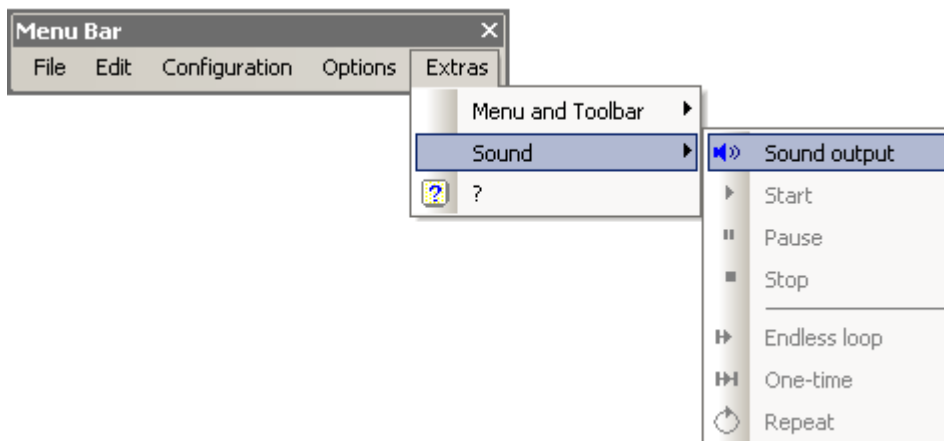
Changes of the menu are permanently saved when the dialog *Customize* is closed. The particular toolbar arrangement saved depends on the display type:

- Default
- Last value as number
- Overview window
- Table

The toolbars can also float freely away from the window. To accomplish this, drag the toolbar away by its left edge:



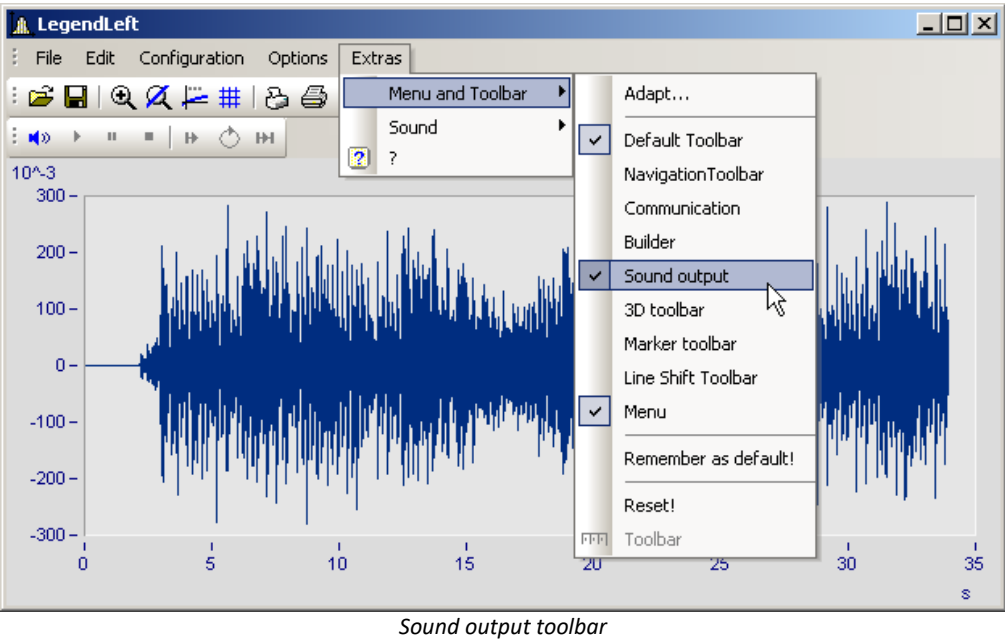
11.6.7.5.2 Sound



The Sound Toolbar lets you make measured data sets audible. For this purpose, a variety of functions such as cutting, endless looping and repeat are available. The sound output always only applies to the first data set in the first coordinate system and is only available in the Standard view, or with multiple y-axes.

11.6.7.5.2.1 Sound toolbar

For working with the sound output, you should select the associated toolbar from the menu *Extras / Menu and Toolbar / Sound output*.



Under Sound Output, the following functions are available:

Menu item	Description
 On/Off	This function lets you switch the sound output on and off. A vertical bar marks the current playback position within the curve window (playback marking), and a horizontal bar denotes the selected playback range.
 Start	With this function you start output of the sound, and the selected range of the data set is played back in real time. The playback's starting and ending points can be freely selected and the position of the resulting playback time window can be moved as desired (see Cut Sound ^[899]).
 Stop	This function stops playback of the data set. The playback marking is set back to the start of the selected playback region.
 Pause	This button pauses playback of the data set. The playback marking remains at its current position. By clicking on it again, playback can be resumed.
 Loop from loop start to end of waveform	When you choose this operation mode, the final position for the playback is automatically set to the end of the waveform; see Cut sound ^[899]
 Loop (repeat) from loop start to loop end	Repeated playback of loop (so-called Repeat-mode).
 Loop (single) from loop start to loop end	One-time playback of the loop

Volume

The volume is set by setting the scaling of the y-axis. The range between y-min and y-max is interpreted as the maximum volume of the sound card. Thus for example, if the signal extends from -5 to 5 (units) and the y-axis extends from -5 to 5, the maximum volume is achieved. If there is an additional signal extending only from -0.5 to + 0.5, and if it is displayed with the same y-axis settings, the amplitudes from the sound card will only be 1/10 as high. I.e. the signal will be played back correspondingly softly.

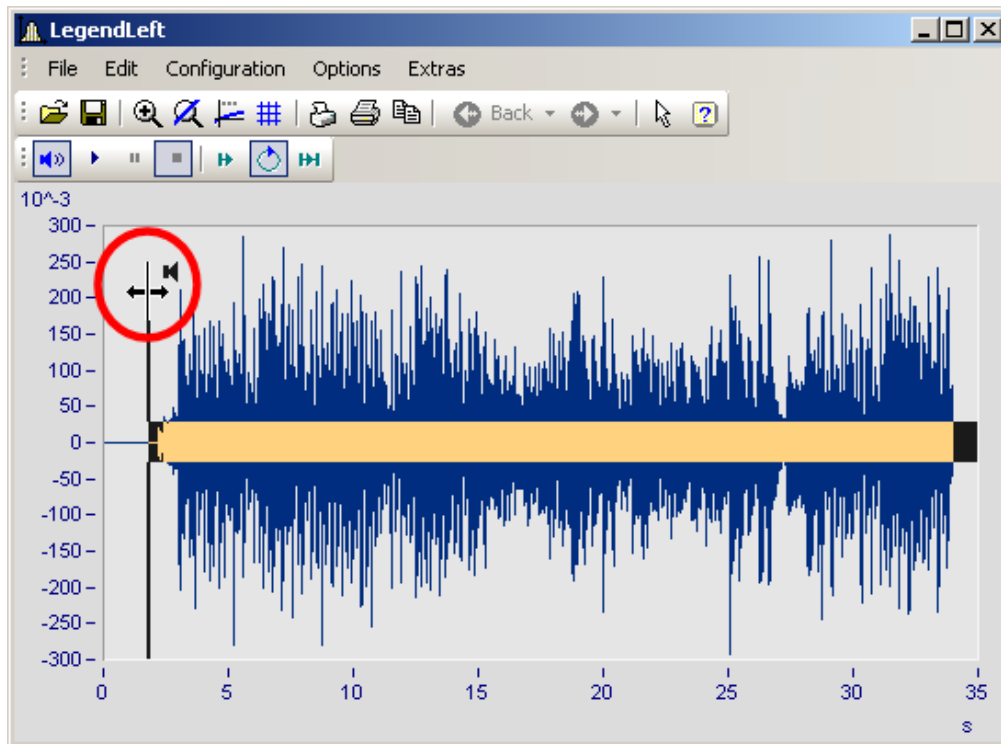
When the volumes of various signals are to be compared, all y-axes must be set to the same scaling. If the y-axes are scaled automatically, all signals have the sound card's maximum level.

The Sound Kit functions have not settings options for volume.

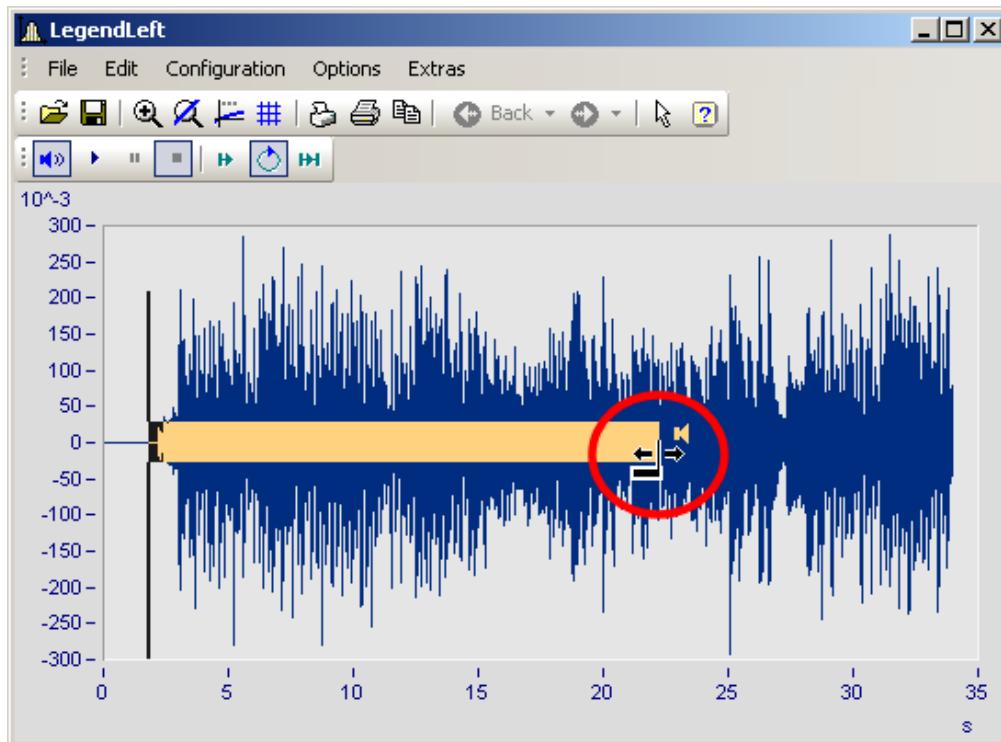
11.6.7.5.2.2 Sound clip

In sound output, it is possible to freely select (take a clip from) the playback region.

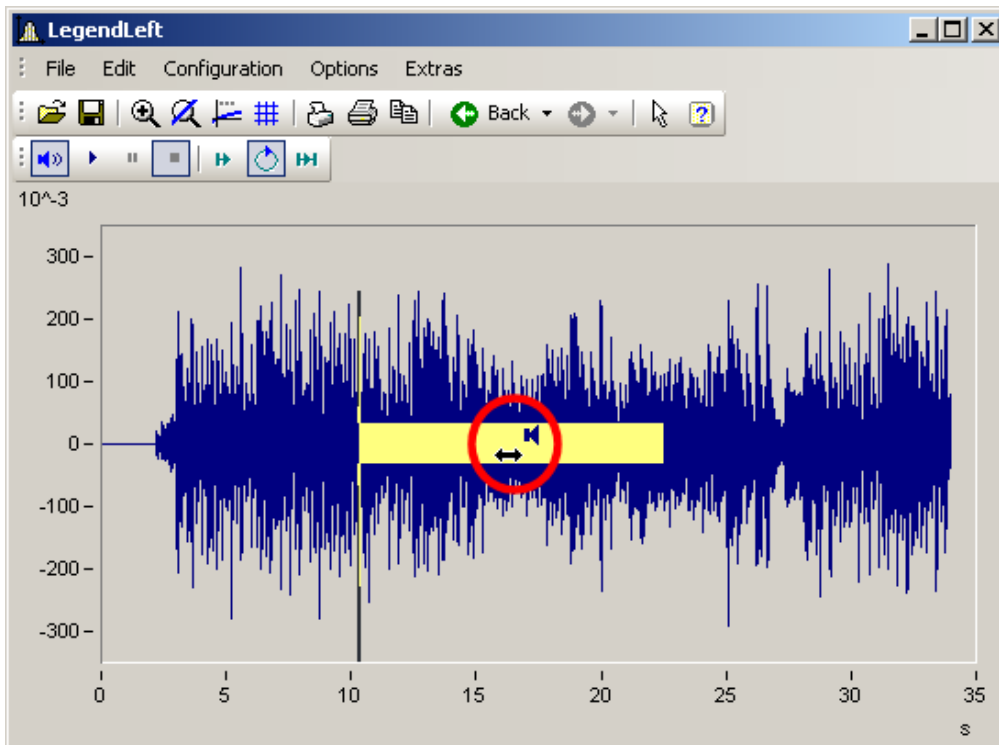
Grab the start position at the beginning of the Playback strip and move it to the position desired.



The final position for playback can also be changed. Grab the end of the playback strip and move it.



You can also move a selected playback region forwards or backwards along the data set. Click on the strip and move the mouse while holding down its button.



11.6.7.5.2.3 Audio output directly during measurement

During a measurement with imc STUDIO /DEVICES, it is possible to use the curve window to listen directly to the streaming data. Since the data are transferred block-by-block and sporadically, the output must be delayed in order for the playback to be seamless.

In the curve window [settings](#)⁸⁸⁷, you will find the two entries "Start delay" and "Max. delay".

At the beginning, as well as after losing synchronization, the "Start delay" takes effect. By means of "Max. delay", it is possible to compensate for asynchronization between the measurement device and the PC. Both values are stated in milliseconds.

11.6.8 Information and tips

11.6.8.1 Greek texts in comments, markers and axis labels

Axis labels and comments on the variables can be supplemented with Greek letters.

This is accomplished using the statement `<g*Placeholder>`.

Example

`<g*a> ^ <g*a>` is displayed as α^β .

The following table lists the available Greek letters with their respective placeholders. Here, the Unicode is only provided as a reference; it can't be used in the statement.

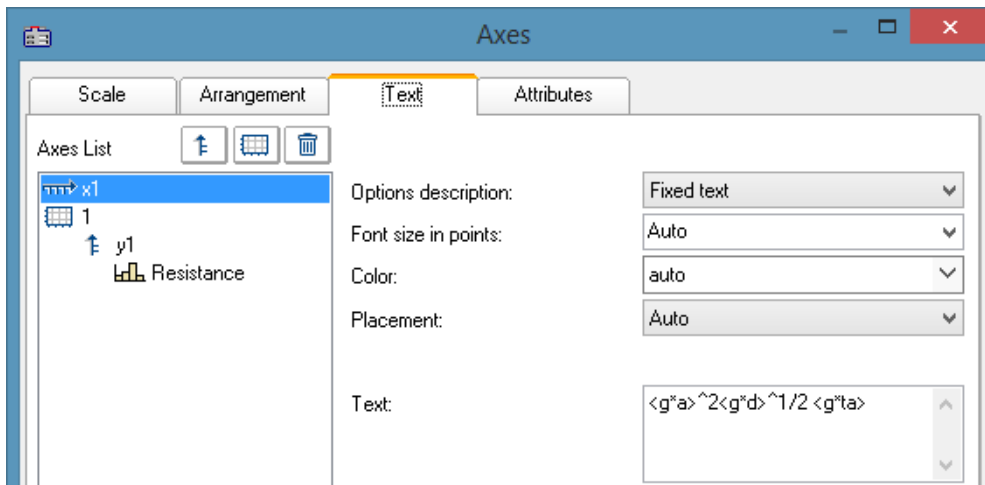
Placeholder			Placeholder		
Meaning			Meaning		
'a'	0x3b1	Alpha	'm'	0x3bc	My
'b'	0x3b2	Beta	'n'	0x3bd	Ny
'g'	0x3b3	Gamma	'x'	0x3be	Xi
'G'	0x393	uppercase Gamma	'X'	0x39e	uppercase Xi
'd'	0x3b4	Delta	'p'	0x3c0	Pi
'D'	0x394	uppercase Delta	'ph'	0x3c6	Phi
'e'	0x3b5	Epsilon	'ps'	0x3c8	Psi
'et'	0x3b7	Eta	'P'	0x3a0	uppercase Pi
'z'	0x3b6	Zeta	'Ph'	0x3a6	uppercase Phi
'th'	0x3b8	theta	'Ps'	0x3a8	uppercase Psi
'ta'	0x3d1	theta, (customary) notation	'r'	0x3c1	Rho
't'	0x3c4	tau	's'	0x3c3	Sigma
'Th'	0x398	uppercase Theta	'S'	0x3a3	uppercase Sigma
'k'	0x3ba	Kappa	'ch'	0x3c7	Chi
'l'	0x3bb	Lambda	'Ch'	0x3a7	uppercase Chi
'L'	0x39b	uppercase Lambda	'o'	0x3c9	Omega
			'O'	0x3a9	uppercase Omega



Example

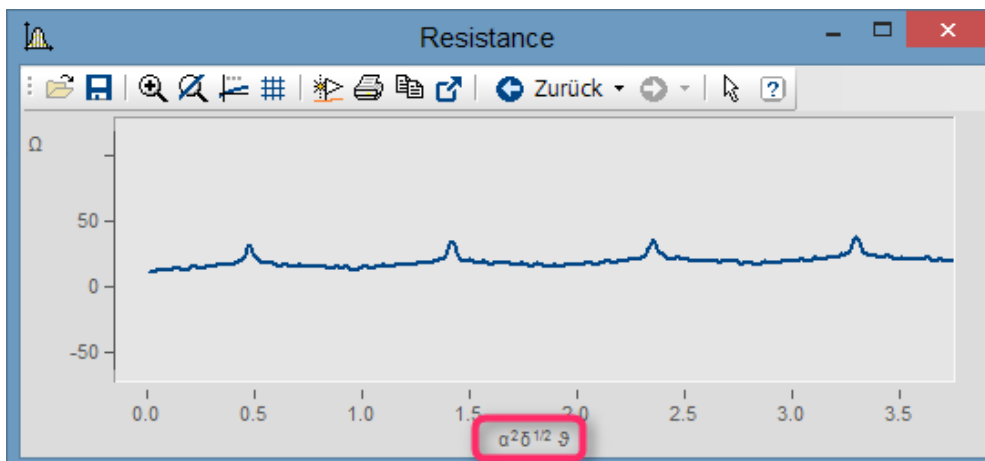
curve window

with



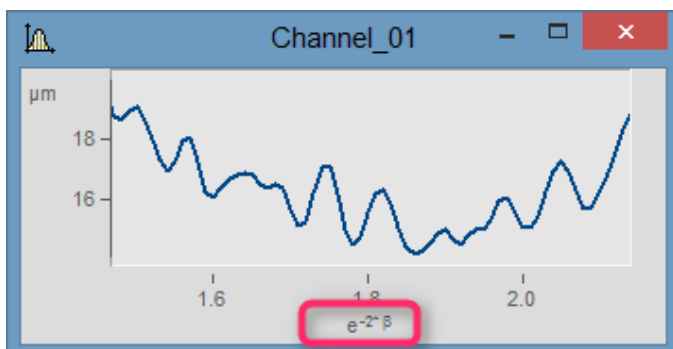
Greek characters in "Axes\Text": $\alpha^2 \delta^{1/2} \gamma$

the result is:



Greek characters as text at the X-axis

Exponent with multiple characters. From $e^{(-2 \cdot \gamma \cdot b)}$ we obtain:



Exponent of placeholder in parentheses: $e^{(-2 \cdot \gamma \cdot b)}$



Example

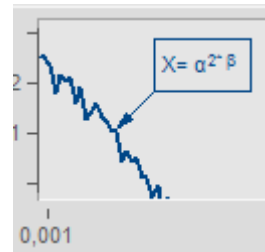
markers

with



Greek characters in marker-definition:

the result is:



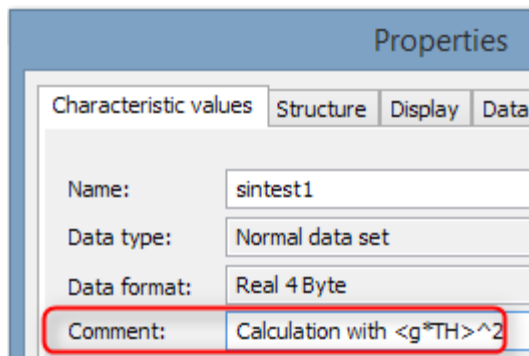
Greek characters in markers



Example

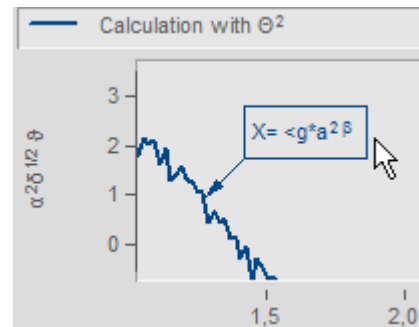
comment on variables

with



Greek characters in variable "Properties\Comment"

the result is:



Greek characters in "Axes/Legends"

Orders of magnitude and units

Specified orders of magnitude with units are recognized and displayed appropriately. For example, the unit 'mm' is corrected to the SI unit 'm'. 10.000 'mm' becomes 10 'm', 0.01 'mm' becomes 10 'μm'. The condition is that "auto" is set for "[Order of magnitude](#)" under "Scale".

11.6.8.2 Curve window context menu

Function

Click with the right mouse button in the curve window to prompt a context menu, which offers quick access to further functions in the curve window. The context menu depends on the cursor's position and on whether or not you are in Select-mode.

11.6.8.3 Copy&paste - Transfer configuration

With Copy & Paste, you can transfer once curve window's configuration to another. To do this, copy the configuration of the selected window using **CTRL-c** and transfer it to another one using **CTRL-v**.

With embedded curve windows, it is additionally necessary to hold down the **SHIFT**-key (curve window in the Panel).

11.6.8.4 Drag&drop, mouse wheel

Moving waveforms

Drag&drop allows the user to shift or copy waveforms in curve windows. To apply this technique within a coordinate system, hold down the CTRL-key. Use drag&drop for the following:

- **Drag** a waveform **from the imc FAMOS variable list to the curve window**. This allows the waveform to be displayed in the curve window as well as indicated in the list.
- Within a curve window: While the **CTRL-key is pressed, all data can be moved to an axis or to a coordinate system**.
Note: The old behavior without the CTRL-key is made available by means of the Registry entry:
`[HKEY_CURRENT_USER\Software\imc Measurement and Control\Default\CurveDataManager\Curves]; String value: "dd63"="1"`
- Transfer **all data** for an axis, a coordinate system or an entire curve window **to another curve window** for display.
- Move **all data** concerning an axis, a coordinate system or a curve window **to the imc FAMOS variable list**. If the data do not belong to imc FAMOS (i.e. are from another application, e.g. imc STUDIO), they are copied to imc FAMOS.
- With [maximized curve window display](#)  (coordinate system = outer frame), Drag&Drop also starts when an axis is moved to the bottom. In that case, the mouse pointer symbol changes.

Displayed region

- Using **drag&drop** within a curve window moves the axes.
- By using the **mouse wheel**, the region around the mouse cursor's current position is increased/reduced. The increments of size change are smaller/larger when pressing the SHIFT-/CTRL-key, respectively.
- If the **mouse wheel** is located over either the **X-** or the **Y-axis**, then only the respective axis is altered.
- Pressing the **CTRL-key** causes the **effect of the mouse wheel** to be **increased**, while pressing the **SHIFT-key** causes it to be **diminished**.

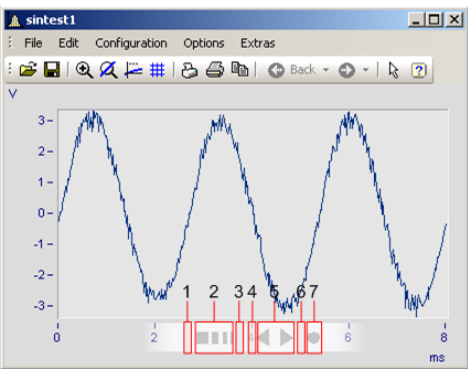
Remarks

- If the mouse is released over a y-axis, the data are displayed on this axis. Release the mouse over a coordinate system, and the data are displayed with a new y-axis for this coordinate system. Otherwise, a new coordinate system is opened.
- When data are moved to the variable list in imc FAMOS, the waveforms are actually copied. In all other cases, the same waveforms are simply relocated.
- The cursor changes its shape during "Drag". If a symbol appears indicating an invalid operation, "Drop" cannot be executed, only Cancel is possible.

11.6.8.5 Axes navigation bar

For quick navigation through the curve window, each axis has its own navigation bar. The controls appear when you move the mouse cursor square over the axis labeling. The navigation bar offers three basic functions:

- You can magnify or reduce the displayed region, while leaving the value in the middle of the x- or y-axis, or z-axis remains unchanged. This function is provided by the button denoted as "2" in the picture. Alternatively, you can change the axes' range using the mouse wheel.
- You can shift the region displayed to the right or the left. This function is provided by the button denoted as "5" in the picture.
- Any changes in the magnification, moving or zooming can be reversed and thus return to their original appearance.



Dragging		Description
1 <- [2] 3	Dragging from 2 to 1	Magnification
1 [2] -> 3	Dragging from 2 to 3	Shrinking
4 <- [5] 6	Dragging from 5 to 4	Move left
4 [5] -> 6	Dragging from 5 to 6	Move right
[7]	Click on 7	Rezoom the X-axis. Displays all data

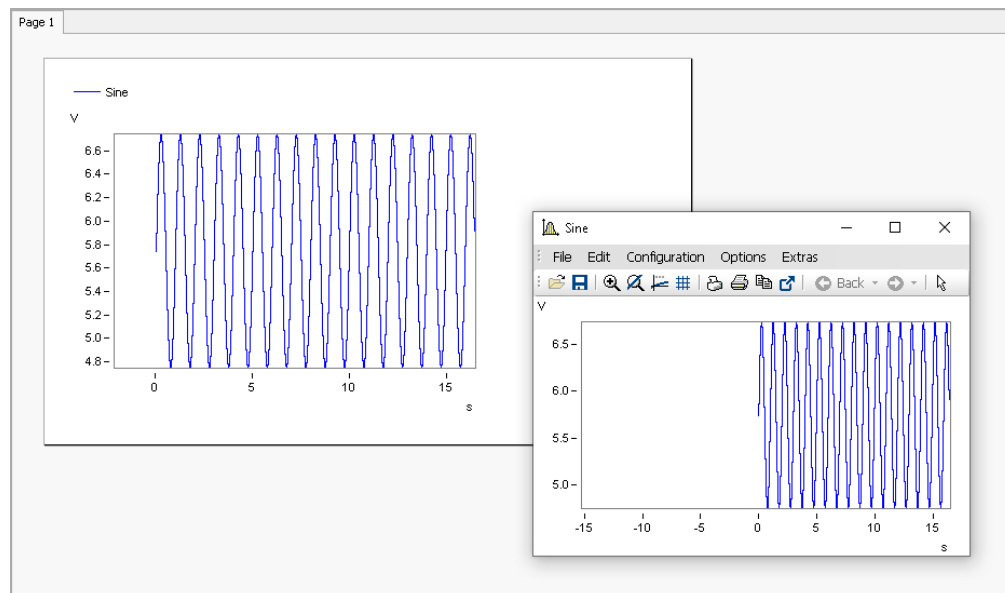
Use of the keyboard arrow keys

While holding down the left mouse button with the pointer in the Navigation pane, you can also move the region by using the keyboards arrow keys.

11.6.9 Curve windows in imc STUDIO

11.6.9.1 Curve window - "free-floating"

The **"free-floating"** curve windows are associated with the experiment. They remain open until closed (X) and are displayed even when the Panel is not open.



Example: curve window - embedded and "free-floating"

The curve window can be opened from different plug-ins:

Setup:

- Open the context menu for the desired channel in the Channels table
- Click on: *Show in curve/values window*

Panel:

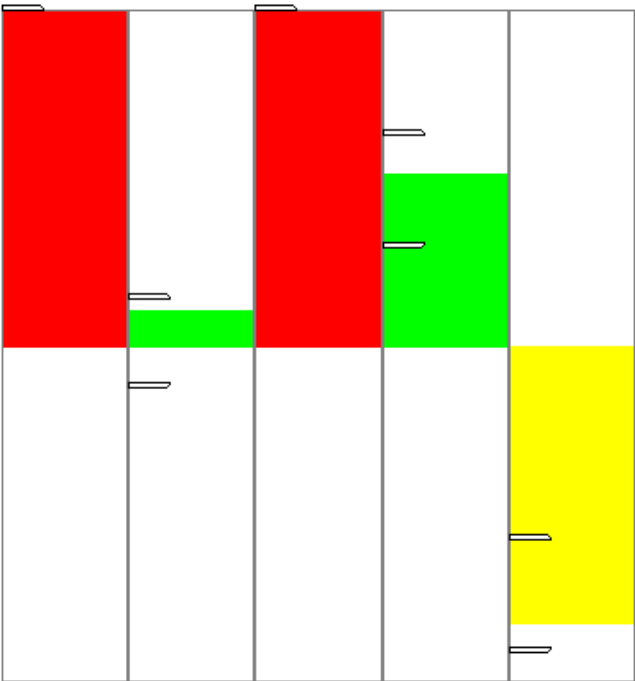
- Double-click in the tool window Data Browser on the desired channel
- or select the desired channels and click on the [Show](#) -button ( / ) in the symbol bar of the tool window Data Browser.

11.6.9.2 Level indicator

The Widget is a pre-configured curve window, suitable for channels having a measurement range; e.g. the analog inputs.

The level indicator displays a bar chart representation to indicate how far the current measurement value is from the measurement range. The channel's respective display range automatically adjusts itself to the measurement range set (this is applied upon performing the action "*Process configuration*").

The current signal level is represented by a three-state color display.



- The axes are configured as follows:
- **Range:** $\pm 110\%$ of the measurement range
 - **Warning color yellow:** 80-100% of the measurement range (for both upper and lower level)
 - **Warning color red:** beyond 100% of the measurement range (for both upper and lower level)

 **Note**

Taring and bridge balancing action before and during running measurement:

Upon starting the next measurement after a bridge balancing action, the level indicator automatically adopts the channel's value range boundaries. If the balancing is performed during the measurement, the Widget does not display the correct value level since it only knows the previous range boundaries. The indicated measurement range in the file also retains the previous state.

11.6.9.3 Curve window colors on the Panel page

The curve window colors can be configured differently for the display/operation pages and for the Report pages. The Panel pages have their own "skins", which specify the applicable colors. However, you are also able to personally specify the colors for each curve window.

As described in the chapter "[Colors](#)", there are colors for the on-screen display, and colors for the printout. Typically, the result should appear the same whether in the printout/PDF or in the monitor screen display. The colors are applied as follows:

Page	Colors
Dialog-page	• On the monitor screen: Screen color • In the printout: Screen color

Page	Colors
Report-page	- On the monitor screen: Printer color (The Report-page is designed for the printout or for the PDF. In consequence, the printer colors are displayed here.) - In the printout: Printer color

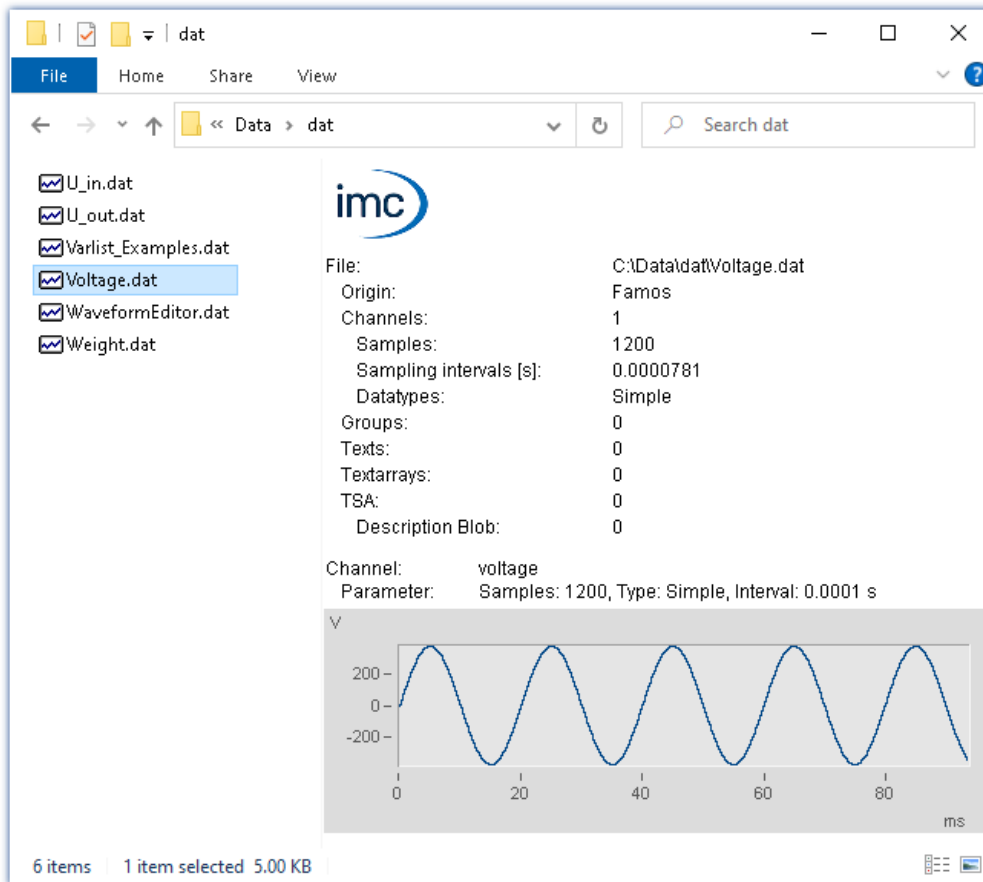
 **Note**

Curve window on Report pages

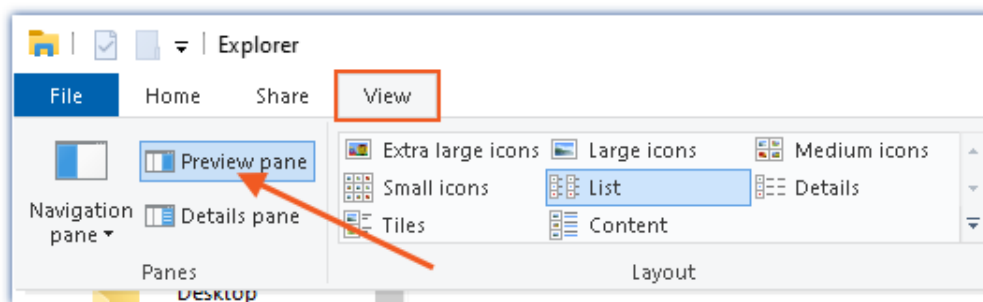
Curve windows on Report pages are set so that the global colors are used for the printout. Furthermore, the background is always white, even when a different color is set.

11.6.10 Preview in Windows Explorer

imc FAMOS files can be displayed using the Microsoft Windows Explorer's preview function.



Activating the Preview window



Activating the Preview window in the Windows Explorer's View menu

The following properties are displayed:

Area	Description
Upper view region	• A roundup of the data regarding the channel count, texts, text arrays, log channels, bus channel descriptions and groups. • For channels, the lowest and the highest channel counts, the sample interval and the data types present are additionally listed.
Lower view region	Infos: • Output of the displayed file set's name • Roundup of channel parameters, sample count, interval, data type • if applicable: additional channel properties group association
Curve window, text contents, array contents	Only one data set is always displayed. When multiple files are selected, only the first is displayed. The curve window comes with a limited scope of the imc curve window's functions, including the ability to zoom in on a specific region.
Scroll bar	When the file contains multiple channels , a scroll bar appears, by means of which it is possible to navigate through the various channels.

11.7 Special Widgets

Here, special Widgets are described.

11.7.1 Text input for report channel

This Widget currently has no function.

11.7.2 Clock

With its various design and formatting styles, the clock provides a variety of ways to visually illustrate a report. It can indicate how long a measurement has been running or when a measurement ended. The current time and date can be displayed. For these purposes, a variety of formats are available.

Here are a few examples:



Variants of the clock

Selecting the time display style

A variety of time readouts are available, depending on the linked variable.

Variable linkage	Description
SystemClock - PC time	Current PC clock time - Default option when the Widget is set up.
SystemClock - System time	Current clock time of imc STUDIO Monitor. This time is used for all components of imc STUDIO Monitor.
Linkage with a channel	Displays the channel's elapsed measurement time, meaning the amount of time which passed since the trigger fired, or the measurement was started. The final time display reflects when the measurement concluded or the triggering event ended. When the trigger fires again, the display time returns to "0 s" and restarts.
Linkage with a trigger-variable's "Trigger time"	Displays the moment in time at the last change of the state: Armed, Triggered, Stopped. I.e. when: • the measurement is started, • the trigger fires, • the trigger event ends and • the measurement is stopped. Thus, stopping the measurement also resets the clock after a previously concluded triggered measurement.

Clock properties

Properties	Description														
Appearance	Here you specify the appearance of the clock; e.g. with clockhands , or a digital readout .														
Mask	There are various ways to display the time or date. The output format depends on the system and on its setting for the Region (Windows "Clock, Language & Region"). E.g. in the format "Long date" in one region may contain the weekday, while in a different region it does not. <table> <tr> <th>Mask</th><th>Description</th></tr> <tr> <td>Long time</td><td>Example: 08:11:45</td></tr> <tr> <td>Short time</td><td>Example: 08:11</td></tr> <tr> <td>Long date</td><td>Example: Wednesday, April 17, 2019</td></tr> <tr> <td>Short date</td><td>Example: 17/04/2019</td></tr> <tr> <td>Date and time</td><td>Example: 17/04/2019 08:11</td></tr> <tr> <td>UTC (IRIG Format)</td><td> Example: 107:08:11:45:261 The first position indicates the current day's number within the calendar year. Thus the date 1/5/2019 is day "5" in this system. This format provides no indication of the current year's number. </td></tr> </table>	Mask	Description	Long time	Example: 08:11:45	Short time	Example: 08:11	Long date	Example: Wednesday, April 17, 2019	Short date	Example: 17/04/2019	Date and time	Example: 17/04/2019 08:11	UTC (IRIG Format)	Example: 107:08:11:45:261 The first position indicates the current day's number within the calendar year. Thus the date 1/5/2019 is day "5" in this system. This format provides no indication of the current year's number.
Mask	Description														
Long time	Example: 08:11:45														
Short time	Example: 08:11														
Long date	Example: Wednesday, April 17, 2019														
Short date	Example: 17/04/2019														
Date and time	Example: 17/04/2019 08:11														
UTC (IRIG Format)	Example: 107:08:11:45:261 The first position indicates the current day's number within the calendar year. Thus the date 1/5/2019 is day "5" in this system. This format provides no indication of the current year's number.														
Time zone	In imc STUDIO, it is possible for differing time zones and clock readings to coincide. • The external clock (e.g. GPS-mouse) may indicate different time than the PC. • The devices may be located in different countries. • On a business trip, the laptop's clock time may reflect a different time zone than the device's. The clock-Widget can be set for different time zones. For this purpose, there is a new selection available: "<i>imc STUDIO-Time zone</i>". With this selection, the clock automatically uses the device's time zone. With the selection: "<i>Local time zone</i>", the PC's time zone is used.														

11.7.3 Image

Using the Widget, you are able to display a logo or a project graphic, or a background which shapes the page. There are various ways to apply the Widget.

Reference

Background image

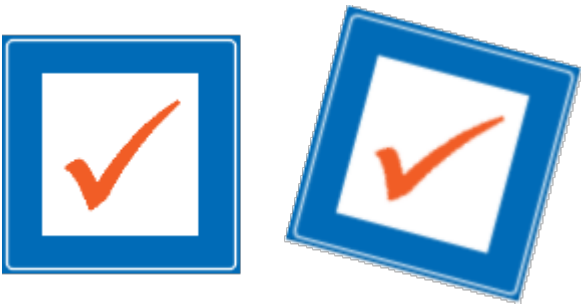
If you wish to display a background image over the entire page, you can use the page's property:
"[Background image](#)".

Properties

Properties	Description												
image	Available file types: png, jpg and bmp If an image is selected, the text: "(Bitmap)" appears under this property. In order to delete the image, delete the text. Be aware that large pictures place high demands on the PC's memory and performance capacity. So if possible, first modify the image to the desired size.												
Size mode	The image can be stretched or displayed without any zoom factor: <table><tr><th>Size mode</th><th>Description</th></tr><tr><td>Auto and Normal</td><td>The image is displayed in its original size within the Widget. The upper left corner is the reference position. If the Widget is too small, only a section is displayed.</td></tr><tr><td>Stretch image</td><td>The image is stretched to completely fill the Widget. The aspect ratio is not retained.</td></tr><tr><td>Automatic Widget size</td><td>The image is displayed in its original size. The Widget adjusts itself to the image's size.</td></tr><tr><td>Center image</td><td>The image is displayed in its original size within the Widget. The center point is the reference position. If the Widget is too small, only a section is displayed.</td></tr><tr><td>Zoom image</td><td>The image is stretched in such a way as to touch the edges of the Widget on at least two (opposing) sides. The aspect ratio is retained. At the other two edges, there are resulting empty strips ("Background color").</td></tr></table>	Size mode	Description	Auto and Normal	The image is displayed in its original size within the Widget. The upper left corner is the reference position. If the Widget is too small, only a section is displayed.	Stretch image	The image is stretched to completely fill the Widget. The aspect ratio is not retained.	Automatic Widget size	The image is displayed in its original size. The Widget adjusts itself to the image's size.	Center image	The image is displayed in its original size within the Widget. The center point is the reference position. If the Widget is too small, only a section is displayed.	Zoom image	The image is stretched in such a way as to touch the edges of the Widget on at least two (opposing) sides. The aspect ratio is retained. At the other two edges, there are resulting empty strips ("Background color").
Size mode	Description												
Auto and Normal	The image is displayed in its original size within the Widget. The upper left corner is the reference position. If the Widget is too small, only a section is displayed.												
Stretch image	The image is stretched to completely fill the Widget. The aspect ratio is not retained.												
Automatic Widget size	The image is displayed in its original size. The Widget adjusts itself to the image's size.												
Center image	The image is displayed in its original size within the Widget. The center point is the reference position. If the Widget is too small, only a section is displayed.												
Zoom image	The image is stretched in such a way as to touch the edges of the Widget on at least two (opposing) sides. The aspect ratio is retained. At the other two edges, there are resulting empty strips ("Background color").												
Background color	The color displayed wherever the Widget is not totally filled by the image.												

11.7.4 Graphical switch

The graphic switch can be used as an operator control and as a status indicator. Depending on the switch state, it can have **different designs** or **colors**. Thus for example, the **positions of movable elements** can be represented schematically. As well, **rotation of the image** can be achieved by means of an additional variable.



left: switch without any rotation
right: switch with 15° rotation

Properties	Description
Zones	In addition to the color, an image can be defined for each zone ⁶³⁴ . Thus, various states can be represented by a color, or movable objects by multiple images.
Rotation: Angle	The angle can be set to a fixed value or to a variable. The value is stated in degrees (0-360). Values outside of this range are converted accordingly: -90 = 270; 450 = 90.
Rotation: Center point	The center point around which the switch is rotated.
Rotation: Angle factor	This factor affects the angle. Example: Factor "90" → for angle values of 0, 1, 2, 3, the switch rotates by 0°, 90°, 180°, 270°.
Rotation: Zoom factor	Within the Widget, the image is shrunk by this factor so that it can be displayed completely even when rotated. As soon as any angle is entered, the Zoom factor is automatically set to 0.7. This value can be changed.

11.7.5 Execute menu action

This Widget lets you execute a menu action. A list contains the actions available in the menu ribbon. When this Widget is run, it replicates clicking the mouse over the corresponding button. This widget can be helpful when the Panel is running in fullscreen mode. The menu ribbon is not present then.

For touchscreen displays, the Widget comes in very handy. In order to make it easier to operate, it can be magnified for display. The icons adjust to the size of the Widget.



Various menu actions

This Widget adapts itself to the status of the menu action. E.g. when the action is currently not present, the button can not be clicked. Some menu actions are attached to specific user roles, device components or loaded imc STUDIO-components.



Reference

Command - Execute menu action

For Sequencer-routines, there is a [command of the same name](#)⁹⁹¹. This command is often linked with switches on the Panel.

Configuration: Execute menu action

Parameter	Description
Menu action	Here, select the desired menu action. Open the drop-down list. The menu actions appear in groups. Scroll to the appropriate group and select the desired action. Filtering: You can enter the first letter of the action's name in the input box. Then the list will be restricted to actions starting with that letter.
Error in execution	If this action returns a fault condition, there are various responses available. • Error: The action returns an error message in the logbook. • Warning: The action returns a warning message in the logbook. • Ignore: No info appears in the logbook.
Menu action description	The action's description text is displayed.
Additional information	The menu action's internal designator is displayed.

11.7.6 Table

The table has a wide variety of applications. It can contain single values as well as whole channels and segments. For each cell, you can define various editors such as lists, switches and sliders. You are able to edit a channel's individual points or [add or delete points](#)⁹¹⁷.

If a **value can't be displayed correctly**, the cell is filled with hash characters "###" as an immediately visible indication of the problem. A truncated number is not displayed.

Structure of the properties

The properties are divided among **three areas**: "Table", "Column" and "Cell". Depending on the selection made, the associated properties are displayed. This means that, for instance, if only the table is selected, only that one area is displayed. There is a fourth area when "[embedded Widget](#)⁹²⁰" is present.

Inherited properties: Some properties can be specified to apply throughout a higher area; for instance, you can define "Zones" for the table. In each column or cell, you can decide whether the zones are "*inherited*" from the level above, or whether to define specific ones.

Design and properties of the table

Most of the actions below are only possible in Design mode.

Action	Description
Linkage with variables	Either by means of Drag&Drop or of the Properties, you can link individual elements of the table with variables. • Linked with a cell Single values, channels, texts, ... can be linked with any cell. The respective momentary value is displayed. • Linked with a column You can link channels with a column. The cells are then filled with the variable's values. Using the property: "Column" > "Behaviour" > "Direction⁹¹⁷", you can specify whether the channel's first or last values are displayed. • Linked with the table Segmented channels, matrices can be linked with the entire table. The number of columns is determined by the number of segments (max. 20) Useful property: "Automatic table length⁹¹⁷"

Action	Description
Adding/removing columns and rows	By means of the properties, you can specify the column and row counts. If you wish to add/delete a targeted column, open the context menu at the desired position in the column header and select the respective action. If you wish to add/delete a targeted row, open the context menu at the desired position in the table and select the respective action.
Moving columns	Use Drag&Drop to move the columns to the desired position
Table length in relation to the channel length	By means of the property: "Table" > "Behaviour" > "Automatic table length", the table length is automatically assimilated to the count of points in the channel; e.g., you can display imc FAMOS results in the table. In response to each additional result in the channel, the number of rows increases.  When multiple channels are displayed simultaneously, please ensure that the point counts and update rates of the different channels are the same. Do not use another column to display single values.
Order/read-out direction	If the column is linked to a channel, use the property : "Column" > "Behaviour" > "Direction" to specify whether the first or last value is displayed at the top. By clicking the mouse button over the arrow icon next to the column header you can switch the read-out direction. Using the property: "Table" > "Behaviour" > "Allow a change of direction" you can deactivate the change at a click of the mouse.
Start only at x-th value	Using the property: "Column" > "Behaviour" > "Offset", you can set for a channel that, for instance, the values starting from the 5th value on. 1st value = 0; 2nd value = 1; ...
Insert/delete values for a channel	Add values to a channel or delete some values. By means of the context menu, you can insert a value either above or below the selection: "Insert sample before", "Insert sample after", "Delete sample"
Entry-editors	Instead of entering text and numbers from the keyboard, you also can use other methods to make entries. See " Entry-editors for the cells " ⁹¹⁸

Titles, captions and texts

Text-settings	Description
Change text of column header (title)	Change the title using the property: "Column" > "Appearance" > "Title". When the title is changed, the source^[628] is automatically set to "User-defined". If the column is linked with a variable (e.g. channel), the variable's properties such as its name, can be used as the title (see "Source^[628]").
Hide column header (title)	Set the visibility using the property: "Table" > "Appearance" > "Column captions". In this way, you can design your own headers with other elements.
Hide table header (title)	Set the visibility using the property: "Table" > "General" > "Title". By means of the position selection, you can hide the title by clicking on the box "Title".
Show title column	For each title, you can show a title column appearing to the left of the column. Property: "Column" > "Appearance" > "Title column". By means of the associated property: "Column title", you can change the title of the "Title column". Change the contents in the title column cell by means of the property: "Cell" > "Appearance" > "Cell title". When the title is changed, the source^[628] is automatically set to "User-defined". If the associated cell is linked with a variable (e.g. DisplayVar), some properties of the variable such as the name, can be used as the title (see "Source^[628]").
Alignment of the texts	The alignment of the texts (right/left/centered) can be set by means of various properties: • Column header (title) and the column cells: Property: "Column" > "Appearance" > "Value alignment" • Title column and its cells: Property: "Column" > "Appearance" > "Title alignment" • Table caption (title): "Table" > "General" > "Title". You can place the title using the position selection.

Entry-editors for the cells

To enter texts and numbers, you can also use other input techniques. For each cell, you can define an "Editor". Property: "Column/Cell" > "Behaviour" > "Editor".

In some cases, the cell's [zones](#)^[919] determine what selection possibilities exist and the scope of the editors.



The table's Editors

Editor	Description
Text	Entry of numbers and texts. When zones are used, no selection box is offered.
Combo Box	Entry of numbers and texts using a selection box. The zones determine the selection options available.
Spinner	 Entry of numbers. Techniques for making changes include the use of arrow keys in the Editor and the mouse wheel. In these ways, targeted increase/decrease of individual digits can be achieved.
Button	Changes the value while the mouse button is pressed.
Switch	Changes the value upon each click of the mouse button. Display of the value.
Check Box	Changes the value upon each click of the mouse button. Display of a checkbox.
Slider	Entry of a value using a slide control. The range boundaries are defined by the cell -range.

Zones in the table

On this topic please observe the description of the "[Zones-dialog](#)⁶³⁴". There are some additional settings which apply to the table and are described here.

Using the zones you can specify whether texts or numbers are to be set or displayed.

Option	Description
Text as set value	Activate this mode in order to write texts to text variables. Variables as " <i>Set values</i> " are not supported in this mode. There are no ranges.
Numerical zones	The variable's value is displayed numerically. The cell displays various colors depending on the zone-boundaries and the current value. If "<i>Text as set value</i>" is activated, the "<i>Set value</i>" can also be a text. In this case, the display output and the set value are the same.
Textual zones	The variable's value is displayed as a text. In this mode, there is a distinction between "<i>Set value</i>" and the displayed text ("<i>Text</i>"). In accordance with the zone boundaries and the current value, the "<i>Text</i>" of the active zone and the respective color are displayed in the cell.

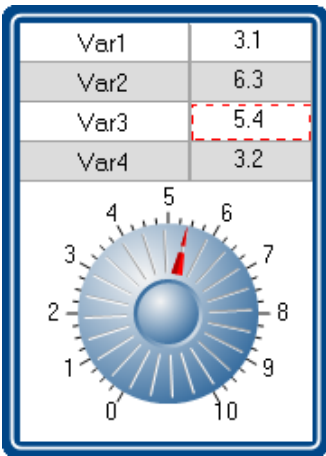
Accept only Zone-values as entries

"Accept only Zone-values as entries": While this option is activated, no entry can be made which does not match the "Set value". The way this is done is different from Editor to Editor:

Editor	Deactivated	Activated
Slider	Using the slider control, any value between the lowest and the highest "Set value" can be chosen.	Using the slider control, only the "Set values" can be chosen. The slider skips between positions as you move the mouse. If the variable's value is provided from outside, the value display and the slider control conform to each other.
Combo Box	Using the list, you can select the "Set values". You can also enter a different value in the input box.	Using the list, you can select the "Set values". There is no option for manual entry.
Button	The value toggles between the first and the second zones.	The value toggles between the first and the second zones.
Switch	The value toggles between the first and the second zones.	The value toggles between the first and the second zones.
Check Box	The value toggles between the first and the second zones.	The value toggles between the first and the second zones.
Spinner	You can enter a value into the input box.	You can enter a value into the input box. If the value does not match any "Set value", the system rejects it.
Text	You can enter a value into the input box.	You can enter a value into the input box. If the value does not match any "Set value", the system rejects it.
Auto	See Combo Box	See Combo Box

Embedding a Widget

Drag another Widget from the tool window to below the bottom cell of a column. The Widget is displayed as embedded there.



In this way, you can expand each column with its own Widget. Not all Widgets offer this capability.

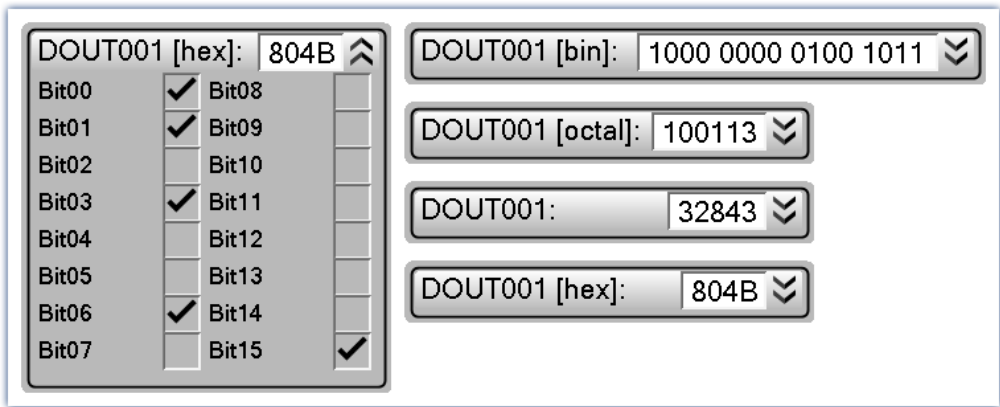
The Widget is linked with the selected cell of the column, which is highlighted by a **red frame**. The Widget displays the value of the selected cell. Additionally, the Widget can also be used as the entry for this cell.

11.7.7 DIO

The DIO-Widget is primarily designed for the DIO-ports and similar data types. But it can also be used with [other data types, within limitations](#) ⁹²³.

Visual display is provided of a port's (a variable's) individual bits. Additionally, the complete value can be outputted in various formats: binary, octal, decimal or hexadecimal.

Below some examples:



DIO-Widget: Examples of the various display varieties

When a variable is linked with the Widget, the display is automatically modified according to the respective data type. This means: an 8-bit port displays 8 bits and a 16-bit port accordingly displays 16 bits.

The Widget is subdivided into two regions. The **port-display** (top) and the **bit-display** (bottom). The bit-display can be expanded/collapsed by means of the Drop-Down button (at right). Using the option "Style", it is possible to hide the port-display, so that **only the bit** are still displayed.

Using the option "Format" you can define the **formatting of the port-display**: binary, octal, decimal or hexadecimal.

Changing the values

Type	Description
Changing bits individually	Place a check in the box next to the bit. The change is applied immediately in the variable
Changing the port or changing bits together	DOUT001 [hex]: A1 Bit00 Bit01 Bit02 Bit03 Bit04 Bit05 Bit06 Bit07 ☒ ☐ ☐ ☐ ☐ ☒ ☐ ☒ ☒ ☒ Select the port-value. The port can now be specified by user entry. Additionally, a list appears offering all bits which can be selected individually. The change is only applied once the input is confirmed.

Port-input with bit-list

Additional display options

Actions	Description										
Formatting the port-display	The formatting of the port-display is defined, by means of the option " <i>Format</i> ": binary, octal, decimal or hexadecimal.										
Hiding bits	By means of the context menu: "<i>Hide selected bits</i>" or "<i>Arrange bits</i>" Notes: Hidden bits no longer affect the port-value. Example: The second bit (Bit01) in a 4-bit-port is hidden. The bits have the following values: <table><tr><th>Bit</th><th>Value</th></tr><tr><td>Bit00</td><td>0</td></tr><tr><td>Bit01 (hidden)</td><td>1</td></tr><tr><td>Bit02</td><td>0</td></tr><tr><td>Bit03</td><td>1</td></tr></table> Since Bit01 is hidden, the following port-value is displayed: 4(hex) (although the complete port has a value of A(hex) - 100(bin) instead of 1010(bin). Even for the input, the bit has no significance. When a value of 2(hex) is entered, then Bit02 is set to 1. The others receive the value 0, except for the hidden bit. That obtains the value: 0110(bin).	Bit	Value	Bit00	0	Bit01 (hidden)	1	Bit02	0	Bit03	1
Bit	Value										
Bit00	0										
Bit01 (hidden)	1										
Bit02	0										
Bit03	1										
Moving bits	By means of Drag&Drop or via the context menu: " <i>Arrange bits</i> "										
Renaming bits	By double-clicking on the bit or via the context menu: " <i>Rename bit</i> "										
Dedicated icons	By means of the options " <i>On</i> " / " <i>Off</i> ", it is possible to specify dedicated icons for the checkboxes.										

Bit-collection

Displaying multiple individual-bit-variables together (e.g. multiple virtual bits)

If no complete port is available but only individual bits, these can still be displayed together. Each variable is moved to the Widget via Drag&Drop, where the mouse position determines the variable's location. The variable can **replace an existing bit** or be **inserted between two bits**.

Note: In this mode, the port-value usually has no significance and can be hidden using the option "*Style*".

Display of other data types

Type	Description														
32-bit Integer pv-variable	The display is limited to 32 bits (0-31). Thus all numbers can be displayed which the data type makes possible. Even negative numbers can be processed, since the integer-pv-variable is a signed integer. <table><tr><th>Decimal</th><th>Hexadecimal</th></tr><tr><td>2147483647</td><td>7FFF FFFF</td></tr><tr><td>1</td><td>0000 0001</td></tr><tr><td>0</td><td>0000 0000</td></tr><tr><td>-1</td><td>FFFF FFFF</td></tr><tr><td>-2</td><td>FFFF FFFE</td></tr><tr><td>-2147483648</td><td>8000 0000</td></tr></table>	Decimal	Hexadecimal	2147483647	7FFF FFFF	1	0000 0001	0	0000 0000	-1	FFFF FFFF	-2	FFFF FFFE	-2147483648	8000 0000
Decimal	Hexadecimal														
2147483647	7FFF FFFF														
1	0000 0001														
0	0000 0000														
-1	FFFF FFFF														
-2	FFFF FFFE														
-2147483648	8000 0000														
Float-variable e.g. display variable or float pv-variable	The display is limited to 23 bits (0-22). Negative numbers can not be precessed directly. Numbers which are too large can cause unexpected results. <table><tr><th>Decimal</th><th>Hexadecimal</th></tr><tr><td>8388607</td><td>7F FFFF</td></tr><tr><td>1</td><td>00 0001</td></tr><tr><td>0</td><td>00 0000</td></tr><tr><td>-1</td><td>7F FFFF <- overflow</td></tr><tr><td>-2</td><td>7F FFFE <- overflow</td></tr><tr><td>8388608</td><td>00 0000 <- overflow</td></tr></table>	Decimal	Hexadecimal	8388607	7F FFFF	1	00 0001	0	00 0000	-1	7F FFFF <- overflow	-2	7F FFFE <- overflow	8388608	00 0000 <- overflow
Decimal	Hexadecimal														
8388607	7F FFFF														
1	00 0001														
0	00 0000														
-1	7F FFFF <- overflow														
-2	7F FFFE <- overflow														
8388608	00 0000 <- overflow														

11.8 Pages

11.8.1 Insert - Dialog / Report

The following ways to add a page are available:

- via the page title's "[context menu](#)⁶¹³"
- via the menu ribbon
- via the tool window "[Page templates](#)⁶¹²" by means of Drag&Drop or double-clicking

Ribbon	View
Panel-Design > Insert Page ()	Complete
Panel-Design > Standard Dialog ()	Complete

Menu item	Description
 Insert Page	Selection window for creating a new Dialog- or Report-page is opened.
 Standard Dialog	Creates a new Standard dialog page.

 **Note**

Replacing a page

When you use Drag&Drop to move a template from the tool window "[Page templates](#)⁶¹²" to an empty page, it is replaced.

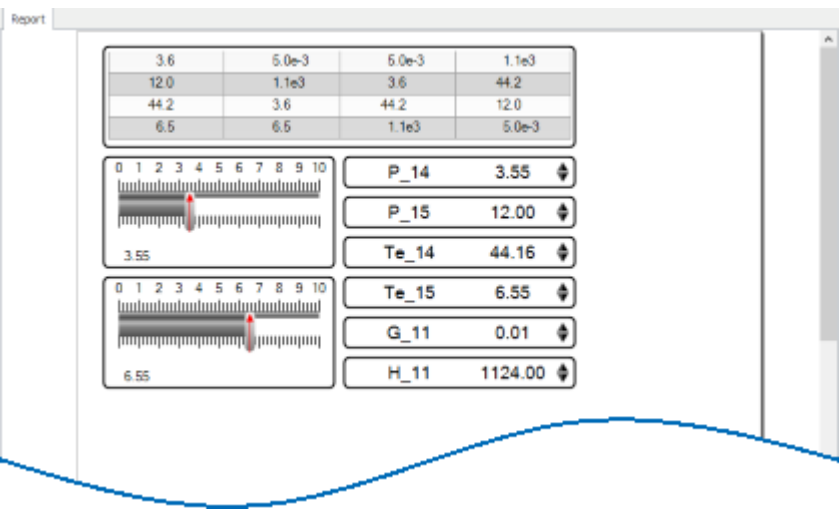
A page can be generated as a "Dialog-page" or as a "Report-page".

Report-page

The Report-page is optimized for printout. On the screen, one of its distinguishing features is that "margins" are displayed. By default, the pages are displayed in a fixed size.

Change the size and other page settings using the dialog: "Page setup" ("Page Layout for Printing" via the [context menu](#)⁶¹³ of the Panel page's tab).

Menu item	Description
	Opens the dialog: "Page Setup"
Page Layout for Printing	Here, you can change the setup and printout settings for Report-pages (including paper size, source, orientation, margin).



Example of a Report-page

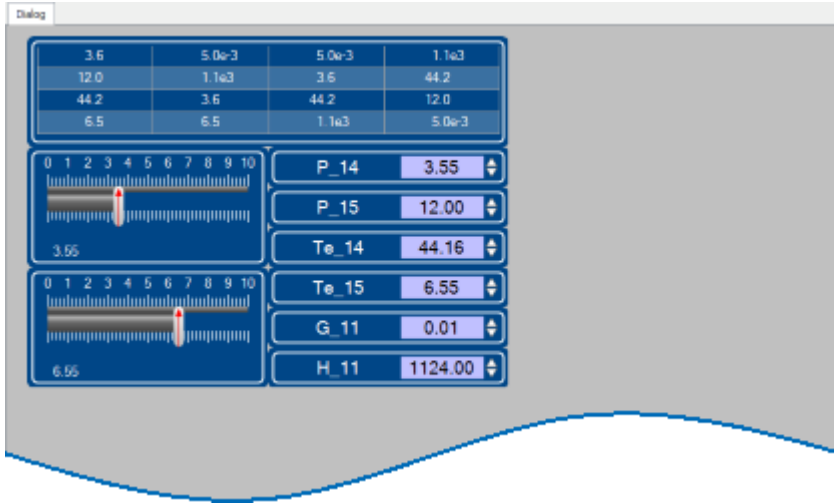
To **print out the page in the page format set**, select the menu item: "[Print](#)"⁵⁸⁹" or the command: "Print Panel page". By means of **PDF-export** (in the menu ribbon or the corresponding command), you can generate a corresponding PDF in that format.

Detailed information is presented in the chapter: "[Page printout or print to PDF](#)"⁹³⁷".

Dialog-page

The dialog-page is optimized for screen display. By default, the pages are **generated in maximum size**.

Change the size via the [Page Properties](#) ⁹²⁶.

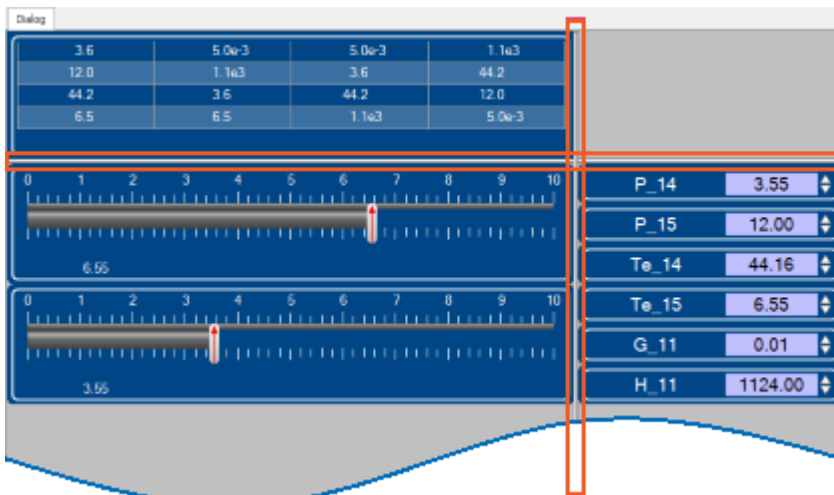


Example of a Dialog-page

After the program is started, the Panel contains a default Dialog-page.

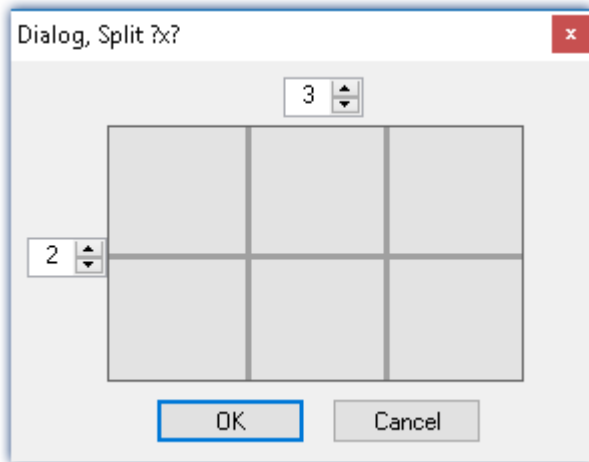
Divided dialogs, e.g. "Dialog 2x2"

These dialogs are partitioned into regions by movable sliders. The size of the [docked](#) ⁶²⁶ Widgets adjusts to the position of the sliders.



*Example of a partitioned Dialog-page
Sliders are framed in red*

Using the choice "*Dialog, ?x?*", you can personally define the partitioning:

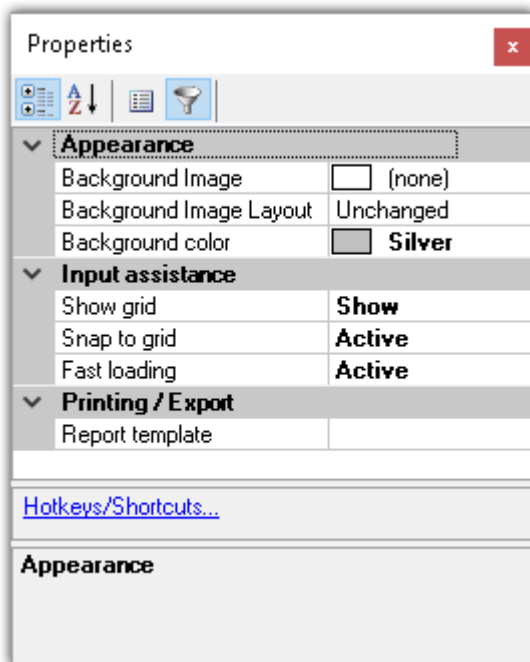


Partitioned format

11.8.2 Properties - Pages

With this tool window, you can set the page's properties. The Panel must be in [Design Mode](#)⁶¹⁶ in order that the properties can be seen/edited.

Right-click on an empty area of the page and select from the context menu the command **Properties** (to see all the properties, click on the symbol for "All properties" ):



General

Property	Description
Name	Name of the page. Each page has a unique name.
Title	The title displayed in the page's tab. Default: <empty>; in that case, the page name is used. If the title is changed, the page name is no longer used.
Multilingual title	The title can be pre-defined for multiple languages ^[944] .

Data linkage

Property	Description
Script name	Panel-scripts are associated with Panel pages. This can be defined directly by means of the script-properties. If no Panel page is selected, the script can be associated with the page by means of this property (Script name).

Layout

Property	Description
	Size of the Panel page (<Width>; <Height>)
Page size	The page size adjusts itself to the setting for the Zoom ^[933] and to the font size (dpi) in Windows. Example: Suppose the current size is "1200; 500". After zooming, or changing the Windows font size to 125 %, for instance, the new value is "1500; 625".  Advantage: When a larger text scaling value is set, the texts and widgets on the Panel page are thus larger and more legible. Disadvantage: When the text scaling is changed, the Panel page becomes larger/smaller and may no longer fit in the display area. If appropriate, have the page size adjusted automatically ^[930], or use the Zoom-function to compensate for the font size.
Width	Width of the Panel page
Height	Height of the Panel page
Page	Pages can be shown or hidden (not hidden in Design mode).

Appearance

Property	Description
Background image	An image can be displayed on the page. In order to select an image, click in the cell and next confirm by clicking on the button [...] on the right edge. In order to delete an image, delete the text "[Bitmap]".

Property	Description
Background image layout	Placement of the background image in reference to the page size: • Unchanged: The image is placed in the upper left corner in original-size. • Tile: multiple copies of the image are displayed side-by-side. The entire page is filled up. • Centered: The image is displayed centered and in original size. • Stretch: The image is displayed centered over the entire page (stretched) • Zoom: The image is displayed centered in optimum size (zoomed). The aspect ratio remains intact.
Background color	The page's displayed background color.
Background plain colored	Display of the background color: • Solid: The background color is displayed over the entire page. • Color gradient: The background color's shade becomes increasingly pale in the downward direction. The page's upper edge conforms to the background color. This progression is not visible for all colors.
Skin	You can choose among multiple pre-defined color schemes (Skin). The color scheme influences the foreground and background colors of the Widgets and pages.

Printing / Export

Property	Description
Report template	Define a Report-page as a print template. The print template must contain the Widget "Print Preview", which is filled in upon executing Print/Export. See: " Page printout or print to PDF ".

Input assistance

Property	Description
Grid size	The distance between the grid points.
Show grid	Hides/shows grid (see " Aligning to grid ")
Snap to grid	Activates/deactivates the alignment of the Widgets to the grid. • Activated: New Widgets or changes to the Widgets (size and position) are aligned automatically to the grid. • Deactivated: Widgets can be positioned freely.
Fast loading	• Activated: Going to pages for which the option "Fast loading" is activated happens quickly since the page's setup is retained in memory (default selection). • Deactivated: Going to pages for which the option "Fast loading" is deactivated happens more slowly since the page's setup needs to be loaded first. This option is recommended for elaborate pages in order to conserve memory resources.

11.8.3 Adapt/automatically adapting the size of pages and Widgets

You can adjust the size of the pages and Widget automatically.

- To do this, open the context menu of the Panel page
- Select *Adapt Page Size*:

Adapt Panel page to window size (or for all pages)	The Panel page takes up the space available.
Adapt Panel page to window size (keep aspect ratio) (or for all pages)	The Panel page retains the aspect ratio, but adjusts its height/width to the available space.
Adapt Widgets to the size of the Panel page (or for all pages)	The Panel page retains its page size. The Widgets take up the space on the page currently available.

 **Note**

Notes on automatically aligning to the grid

- If the Widgets are automatically oriented to the **grid**, this is taken into account when adjusting the size.
- Among other effects, this can lead to the current aspect ratio of the Widgets to no longer persist after the adjustment.
- On every page, there can be a resultant discrepancy of ± 1 grid increment. If this is not desired, first deactivate the [grid orientation](#).

all pages

The functions can be applied either to the current page or to all loaded pages simultaneously.
The following pages are not affected by the functions applied to all pages:

- all pages in which the property "fast loading" was deactivated, unless they are already open currently.

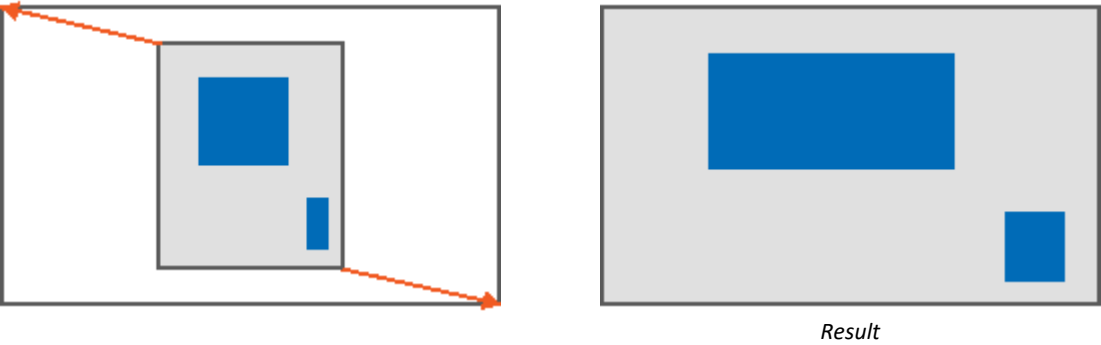
Use of the Zoom function

The Zoom-factor is deactivated whenever the Panel page is adjusted to the window size.

Adapt Panel page to window size

The Panel page takes up the space available. This page size remains constant, even if the program window size is changed.

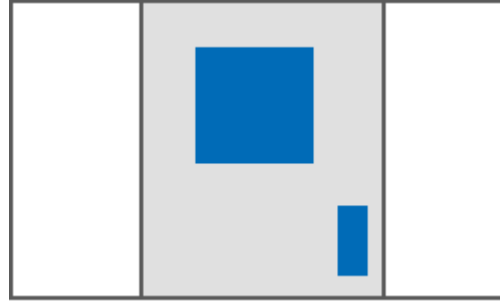
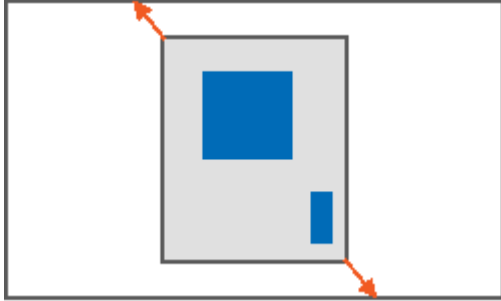
- The aspect ratio of the individual Widgets does not remain intact (observe the note above about the Grid)



Adapt Panel page to window size (keep aspect ratio)

The Panel page retains the aspect ratio, but adjusts its height/width to the available space.

- The aspect ratio of the individual Widgets remains intact (observe the note above about the Grid).

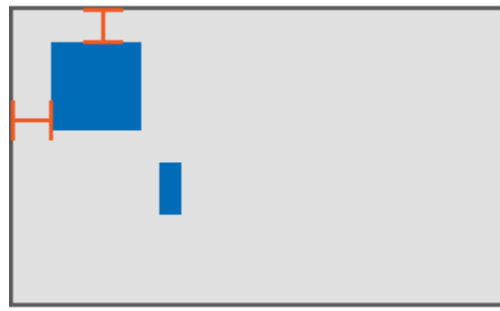
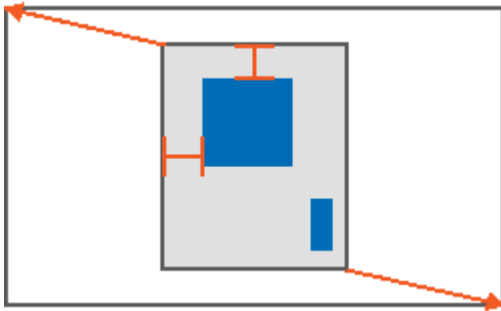


Result

Adapt Panel page to window size (keep Widget size)

The Panel page adapts to the size of the currently available space by only adjusting the page size. This is like when you adjust either the right or the bottom page margin.

- The sizes of the individual Widgets thus remain intact.

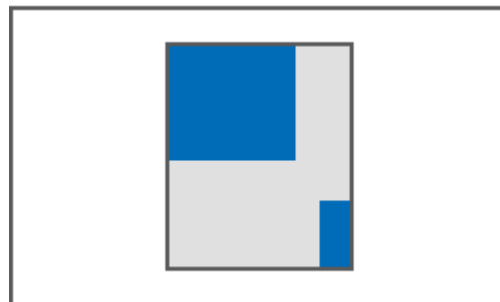
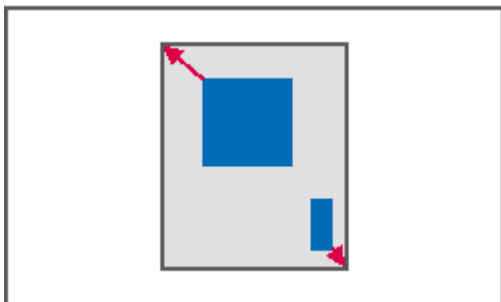


Result

Adapt Widgets to the size of the Panel page

The Panel page retains its page size. The Widgets take up the space on the page currently available.

- The aspect ratio of the individual Widgets is not retained (observe the note above about the Grid).



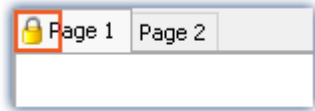
Result

11.8.4 Locking/unlocking a page

You are able to completely protect a page from being changed (also in "Design Mode"). Lock/Unlock the page via the associated menu item or via the context men of the Page-tab.

Ribbon	View
Panel-Design > Lock/Unlock Page (🔒)	Complete

The associated symbol appears next to the page-name.



The locking/unlocking **applies only to the current page**.

In contrast to the deactivated "[Design Mode](#)"⁶¹⁶, not even Widgets can be placed on a locked page by means of Drag&Drop.

 **Note**

No protection against deletion of the page

Locked pages can also be deleted. All that is protected against changes is the content.

11.8.5 Aligning to grid

To simplify the positioning and moving of Widgets on the page, the grid can be used. The Widgets can be aligned to the grid. The distance between grid points can be adjusted in the page's Properties (*Page Properties > Grid size*).

To activate/deactivate the grid or to align all Widgets to the grid, open the Panel page's context menu .

- Select *Grid* (📏):

Action	Description
Show Grid	Hides/shows grid (also adjustable via the page's settings)
 Snap to Grid	Activates/deactivates the alignment of the Widgets to the grid (also possible via the page's settings). • Activated: New Widgets or changes to the Widgets (size and position) are aligned automatically to the grid. • Deactivated: Widgets can be positioned freely.
Realign to Grid	All Widgets are aligned to the grid (even when <i>Align at Grid</i> is not activated)

11.8.6 Zooming Panel pages

For better visibility of Panel pages, for instance, you can zoom in on the display.

- To do this, open the context menu of the Panel page's tab.
- Select *Zoom* (🔍) or *Zoom (all Pages)* (🔍):

Action	Description
 50%	Panel page is zoomed to 50%. Image ratio: 1:2
 100%	Panel page is displayed in original size. Image ratio: 1:1
 >100%	Panel page is zoomed to 200% or 400%. Image ratio: 2:1 or 4:1
 Zoom	Arbitrary zoom-factor

 **Note**

all pages

The zoom can be applied either to the current page (*Zoom*), or to all loaded pages (*Zoom (all Pages)*) simultaneously.

The following pages are not zoomed with the function "*Zoom (all Pages)*":

- all pages in which the property "Fast loading" was deactivated, unless they are already open currently.

11.8.7 Fullscreen

Panel pages can be displayed on a monitor in a full-screen mode. The Fullscreen-mode offers certain advantages:

- **larger display area**, since the menu and tool windows are covered over
- **more protection** against unintended changes, since the fullscreen mode must be exited explicitly

Folgende Vollbild-Arten bietet imc STUDIO

- Display of a page on a monitor (no "genuine" fullscreen, without protection)
- Panel in fullscreen mode, so that the imc STUDIO user interface is no longer accessible (imc STUDIO Professional and higher)

Display of a page on a monitor (no "genuine" fullscreen, without protection)

Panel pages can be displayed in **fullscreen on individual monitors**. The imc STUDIO Monitor user interface can still be operated. However, the Panel page is in the foreground (so if there is only one monitor, it covers the user interface). Thus, you could **observe the Sequencer on the main monitor**, while the **second monitor shows measurement readings** on a Panel page in fullscreen mode.

When a Panel page is displayed on a **second monitor**, this page layout is **saved in the experiment** and restored upon subsequent loading. If the monitor is no longer present upon loading, the page appears again, embedded in the Panel.

- To do this, open the context menu of the Panel page tab.
- Select "Show Page on Monitor" ():

Action	Description
 Page on monitor	• embedded: The page is displayed in the plug-in's Panel main window (default selection) • <Monitor name>: The page is displayed in fullscreen mode on the monitor selected. Independent of the opened main window.

Uses	Description
Automatic assignment by command: Browse in workspace	When the command " Browse in workspace 

 **Note**

No comprehensive protection

This mode **offers no protection against changes** to the imc STUDIO user interface. The button for exiting the fullscreen can be hidden. However, it is possible to exit Panel by typing the key combination `alt+F4`. Or when using multiple screens, the program can be relocated to the second monitor.

Hiding the Exit-button in this fullscreen view

The button is hidden when the following conditions apply:

- The user who is logged on does not possess the right to close the fullscreen view.
- The Panel is already in fullscreen view (see the possibility 2).
- No tabs are shown in the fullscreen mode.

Panel in fullscreen mode

The Panel is displayed in fullscreen mode. The imc STUDIO **user interface is hidden** and can thus no longer be operated or accessed (excepting the Panel pages).

Ribbon	View
Start > Panel Fullscreen ()	Standard
Panel-Control > Panel Fullscreen ()	Complete
Panel-Design > Panel Fullscreen ()	Complete

 **Note**

Protection against changes

This mode **offers protection from changes** to the imc STUDIO user interface. The button for exiting the fullscreen mode can be hidden if one possesses the pertinent access rights. Without such rights, it is not possible to exit the fullscreen mode.

By means of the User administration, you can **prohibit exiting Fullscreen mode**.

[Right](#)⁸³: Panel fullscreen: Button "Exit fullscreen mode"

Navigation through the pages

In this case, show the tabs for the individual pages. Or use buttons and action, in order to navigate to the Panel pages. E.g., you can skip between the individual pages using the command: "[Browse in workspace](#)"⁹⁸⁵".

Moving the fullscreen view to a different monitor:

The fullscreen can only be moved to a different monitor via Windows-shortcuts. Use the keyboard combination `Windows + Shift + left/right`.

Uses	Description
Automatic launch in Fullscreen mode	Using the command-line parameters ³¹ , you can start an experiment directly in Fullscreen mode. Thus, you can start the desired experiment immediately . And you can prevent unauthorized modifications .
Automatic Fullscreen mode by means of the command: Browse in Workspace	When the command " Browse in Workspace " ⁹⁸⁵ is executed, imc STUDIO jumps to the selected page. If the Panel is defined as the target , it can be displayed in fullscreen mode .
Combination of the Fullscreen variants	If the Panel is in Fullscreen mode, you can additionally display more Panel pages on other monitors.

Title Bar - Overview of functions

The fullscreen window has its own title bar (menu). A **variety of functions** can be called from this menu (most of the functions are only available in the actual fullscreen mode).

	The menu includes various buttons. It can be expanded and moved .
	By means of this button, you can expand the menu to reach additional functions .
	Using this button, you can move the menu , in order to access elements located behind it. When you move the mouse pointer over this button, you obtain a special cursor (◀▶). While holding down the mouse button, you can change the menu's position within the top edge.
	By means of this button, you can exit Fullscreen mode . When the menu is expanded, this function is located at this button:  .
	This button minimizes the program .
	Thus button closes the program .
Show Tabs	These checkboxes control whether the tabs for the Panel-pages are hidden or shown.
Data Browser	This button  causes the free-floating Data Browser ⁵⁹⁷ to be shown.

Additional functions can be accessed via the context menu or via the button  in the expanded menu.

Menu item	Description
 Login ⁸⁰	Logs in a user
 Logout ⁸⁰	Logs a user out
 Design Mode	- Activated: the Panel page can be edited, the Widgets can not be operated. - Deactivated: The Panel page can not be edited. The Widgets can be operated.
 Info	Opens the dialog with version information
 Product Configurator	Opens the dialog for product configuration

 **Note**

Notes on modifying the fullscreen size

The full screen has **mode available space**, for which reason you can **increase the page size**. If you already have finished pages and you wish to **magnify these pages**, then in Fullscreen mode use the function: "Adapt Panel page to window size".

You may need to first deactivate the grid alignment, since there may otherwise be unintended displacements. For more information, please see the chapter: "[Adapt/automatically adapting the size of pages and Widgets](#)"⁹³⁰.

11.8.8 Page printout or print to PDF

Panel pages can be printout out as hard copy or to a PDF-file. imc STUDIO Monitor offers a variety of possibilities for this purpose: via the menu ribbon or via the commands ("[Print Panel page](#)⁹⁷⁷" / "[Export Panel page](#)⁹⁷⁷").

Ribbon	View
Panel-Control/Navigation > Print/Print Preview ( / )	Complete
Edit > Print/Print Preview ( / )	Complete
Home > Print ()	Standard

Ribbon	View
Panel-Control/Navigation > As PDF (Export) ()	Complete
Home > As PDF (Export) ()	Standard

The respective menu item allows you to **print/export only one page at a time**.
If you wish to keep **multiple pages** together, then it is best to use **the respective command**. You can, for example, place this on the Panel page and execute it from there. You can hide its button when printing the page out.

 **Note**

Font in the printout

"Font in the printout" is a topic which we can only improve step-by-step. imc STUDIO Monitor passes the information about your page to the default printer set up for your system. This then produces either the PDF or the printout. For this reason, it is hard for us to influence the results, which strongly depend on the printer driver.
If you experience problems, please send our [tech support](#)⁶ the **exact identification** of your system's **default printer**. Switching the default printer temporarily may be helpful.

PDF with vector-elements

In response to the menu item "*As PDF (export)*", a PDF is generated on which graphics are displayed. If the graphics quality is not sufficient, it is possible in some cases to generate vector elements in the PDF by means of a PDF-printer. This, too, depends on the printer driver and not necessarily every PDF-printer is able to do it. Additionally, not all Widget support this function. Some will still embed the image as a graphic.
Therefore, please test this technique with your printer driver before choosing it.

Report-pages - Adjusting the size

The Report-page is optimized for printout. You can make changes to the size and other page settings by means of the dialog: "*Page Setup*" ("*Page Layout for Printing*" via the [context menu](#)⁶¹³ over the tab for the respective Panel page)

Menu item	Description
Page Layout for Printing	Opens the dialog: " <i>Page Setup</i> " Here, you can change the setup and printout settings for Report-pages (including paper size, source, orientation, margin).

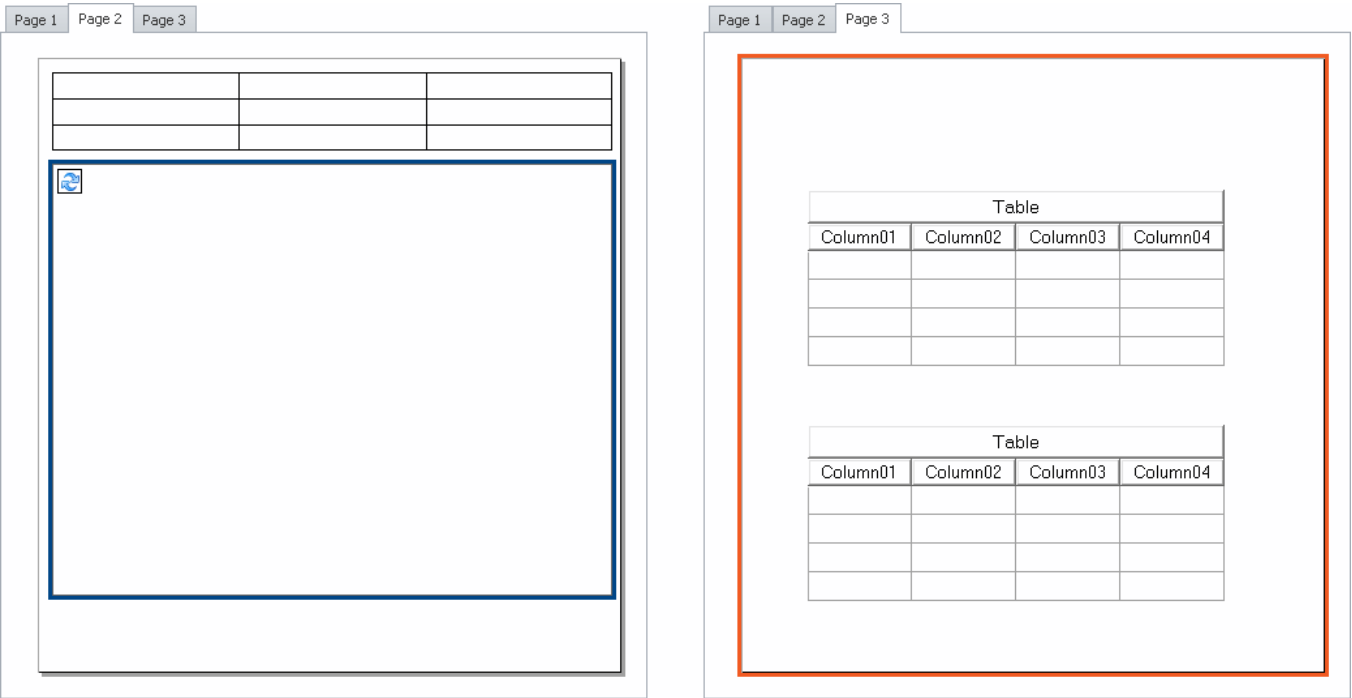
Report template - Header

For purposes of export or printing a dialog page as a report, it is possible to define another report page as the template. This "template page" can be set up as a kind of "page header" and be filled with multiple report pages upon printout. E.g. texts and logos which are to appear on every page.

On this report page the Widget: "Print preview" appears.

In the properties of the dialog page, select the report page under "Report template".

If the dialog page is now printed, the Widget on the report page is filled with the page to be printed.



Example: When it is printed out, the Panel page on the right side will appear inside of the frame appearing within the page shown on the left side.

11.9 Variable linkage

To link a Widget with a variable, there are two methods:

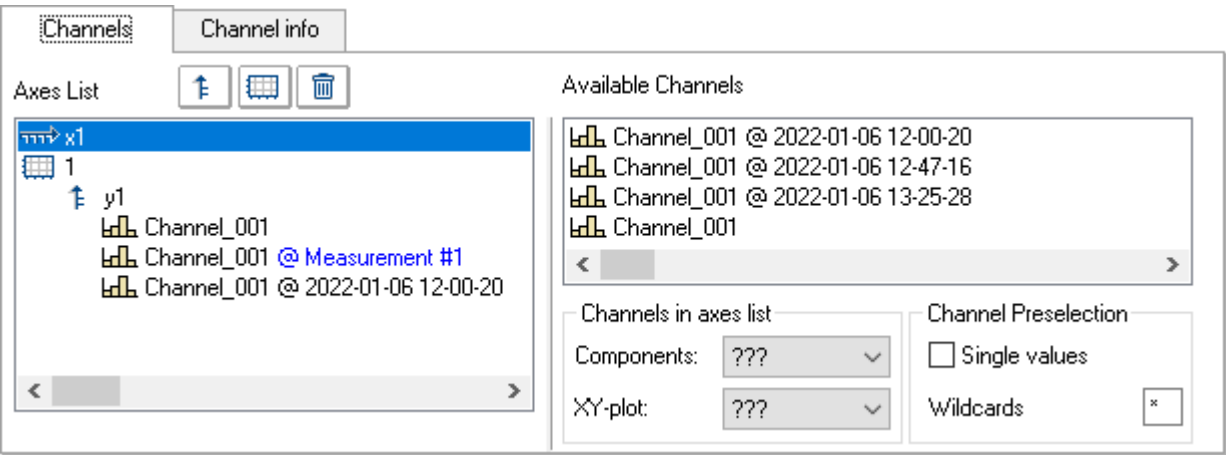
Variable linkage by Drag&Drop ^[940]	Move the variable using Drag&Drop from the Data Browser ^[597] to the Widget or the Panel page.
Variable linkage by Widget Properties ^[940]	Open the " Properties " ^[627] window of the Widget and there click on " Variable ".

Even **without existing variables**, it is possible to set up a linkage (see: "[Variable linkage to not yet existing variables](#)"^[942]).

Directly connecting with a measurement, or changing measurements at the click of the mouse (measurement number)

You can link a variable either by its **fixed name** or its **symbolic name** with a Widget.

Fixed name	<VariableName>@<Measurement Name> e.g. Channel_001 In this way, the Widget always indicates the current measurement. or Channel_001@2022-01-06 12-00-20 in this way, the Widget is always linked with this concrete measurement
Symbolic name	<VariableName>@Measurement#<Measurement number> e.g. Channel_001@Measurement#1 Contains the name of the variable followed by a measurement number. The measurement number can be assigned variably with the Data Browser. This enables you to compare measurements . After measurement, you can select the saved measurement (it is then assigned the the number "1"). Subsequently, all Widgets show the variables belonging to the saved measurement.



Properties of the curve window. Example: three times the Channel_001.

1. Current measurement
2. Selected measurement (Measurmnt number 1)
3. Saved measurement with the defined name

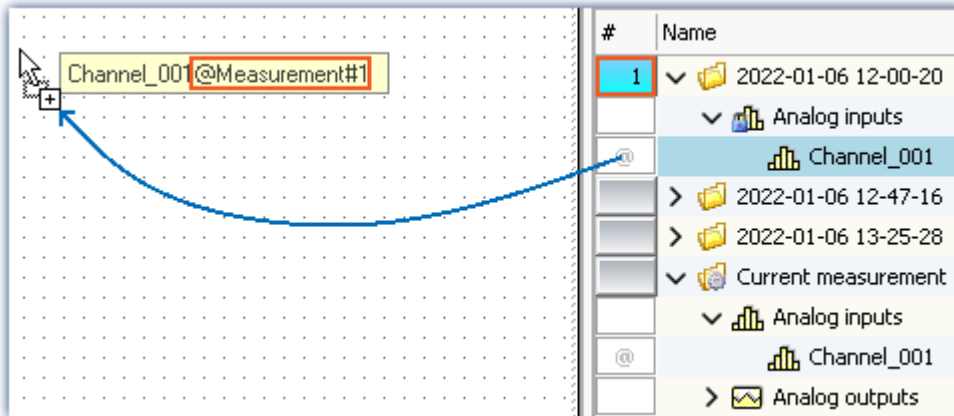
Browsing data

Once you have linked a Widget with a **symbolic name**, you can view the various measurements by simply clicking in the Data Browser (see: "[assigning a measurement number](#)"^[603]).

Variable linkage by Drag&Drop

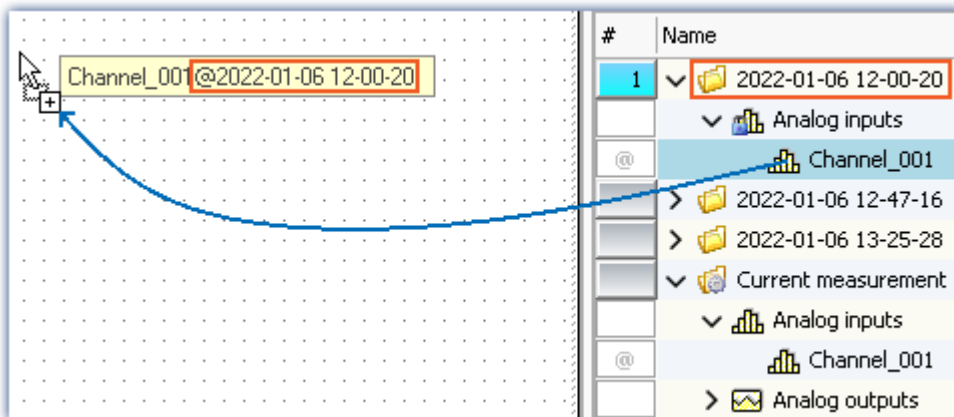
If you have opened a measurement in the Data Browser, the "@" Symbol () appears in the **Numbering** column.

- To link a variable with a Widget **via a measurement number** (symbolic name), use **Drag&Drop** to move the variable **from the "@"-symbol** to the page.



Variable linkage with symbolic name.
Example: "Channel_001@Measurement#1"

- To link a variable with a Widget by its **fixed name**, move the variable to the page **from the name column** using **Drag&Drop**.



Variable linkage with fixed name.
Example: "Channel_001@2022-01-06 12-00-20"

- Release the mouse button over the Panel page and select the Widget with which the variable is to be displayed,
- or release the mouse button over a Widget. The Widget is linked to the variable.

Variable linkage by Widget Properties

Note

Only for Widgets having the property "Variable". For the curve window, please see the separate [documentation](#) ⁶⁴⁷.

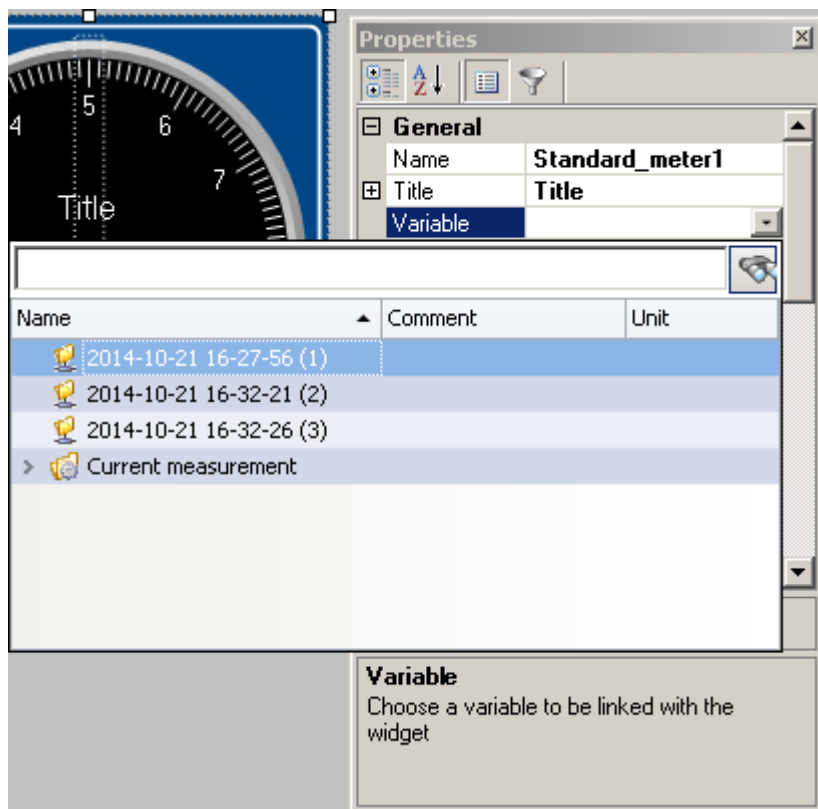
You can establish the link with the variable by means of the Widgets's properties.

- Open the properties of the Widget
- There, click on the entry **Variable**

You can use the dropdown list or enter a name for the variable.

Variable linkage by dropdown list

- Click on the dropdown button (▼)
- A Data Browser opens

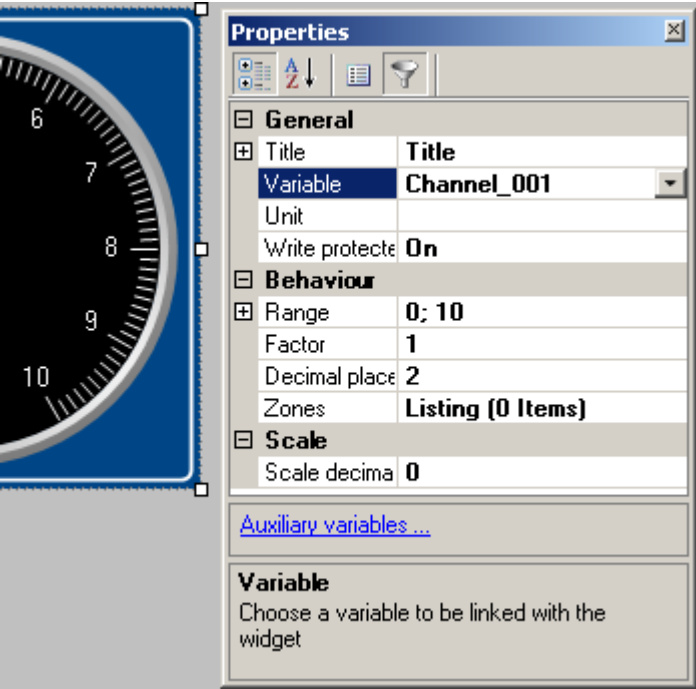


- Select the desired variable

The Widget is now linked with the variable.

Entry of the variable's name

- Enter the variable's name in the text box.



The Widget is now linked with the variable.

By means of the property "Variable", you can also set up a link using a **fixed name** or a **symbolic name**:

Fixed name	e.g. Channel_001
Current measurement	
Fixed name	e.g. Channel_001@2022-01-06 12-00-20
Symbolic name	e.g. Channel_001@Measurement#1
Measurement#x	

Variable linkage to not yet existing variables

You can even design a Panel page before any variables exist.

Using the property "Variable", you can establish a future variable linkage. *

- Enter the variable's name in the text box

The Widget is linked with the non-existing variable.



Since the variable doesn't exist, a yellow warning triangle appears on the Widget.

As soon as the variable is created, the Widget displays the variable's value and the warning triangle disappears.

* (Only for Widgets having the property "Variable". See [Variable linkage by Widget Properties](#) ⁽⁹⁴⁰⁾)

11.10 Information and tips

11.10.1 Tips for the curve window

Curve window - Multiple-triggered signals

Signals which cause multiple trigger releases generate "Events" which can accumulate to a data volume which overloads the system.

Tip	Description
Last Event	Use the curve window menu item "Configuration" > "Events, Segments, Periods". In the dialog which appears, select the page Events. Enter how many of the most recent events you really wish to have displayed. In that case, the curve window plot will always be drawn at the same speed.

Curve window Updating

Many curve updates per second during a running measurement make it easy to view the data plotting.

Tip	Description
Fast interface	The Ethernet connection needs to be as free of interference as possible.
Small segment	The curve window requires longer to plot the data if the number of data displayed is large or if the curve window itself is simply very large . If the number of channels displayed is reduced, the speed is increased. The time needed for drawing the plot also depends on your graphics card's power.
Fewer channels transferred to the PC	If data from fewer channels are transferred to the PC, the performance improves. The other channels are saved to the internal device hard drive, for instance.
Monitor channels	For viewing purposes, channels whose data points have already been averaged or simply more slowly sampled channels (monitor channels) can be used. Data storage can then be performed on the fast-scanning channels.

11.10.2 Jiggling display and heavy burdening of the processor

Virus scanner

The presence of many channels can burden the PC if a virus protection program scans the data captured. We strongly recommend removing imc STUDIO Monitor from the virus scan.

See: Recommended virus scanner [settings](#) 

Refresh rate of Widgets

Placement of many Widgets burdens the PC's capacity. You may need to adjust the refresh rates of some Widgets if, for instance, the curve in the curve window shakes.

See "[Properties - Widget](#)" 

11.10.3 Multi-lingual text input

In many cases you can pre-define titles and zone-texts for a variety of languages. Depending on what the operating system's language is, the respective language variant is displayed. If any text is not defined for any language, the respective English text is displayed (fallback-language). The English text is always automatically filled first.

 Example

You define the title of the Panel page in German, English and French. If you start the experiment on a German system, the German text is displayed; on an English-language system the English text, etc. When you start the experiment on a system not set to any of the three languages, the English text is displayed.

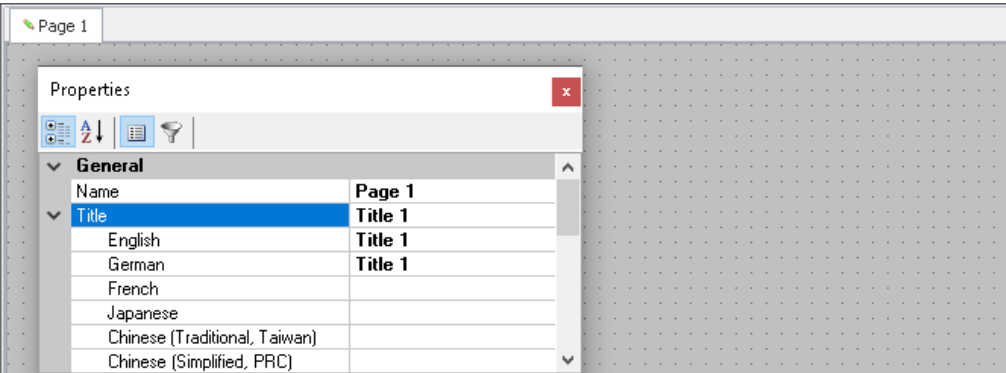
Option: Deactivating multi-language text input

You can, if desired, activate multi-language input: *"Panel" > "General Options"*.

Option	Description
Multi-lingual text input	Enables textual input for computers with other language settings. For example, in this way, Panel pages can be pre-configured for different language settings. If the language set is configured for the respective text, it is displayed. Otherwise, the default language: "English" is displayed.

Defining titles and automatic adaptation

The various languages are usually only shown once the **Properties-filter** is deactivated. Display all of the properties (). Expand the Title-property. In the main branch, you see the **currently displayed title**. Below that are all available **languages and the content**.

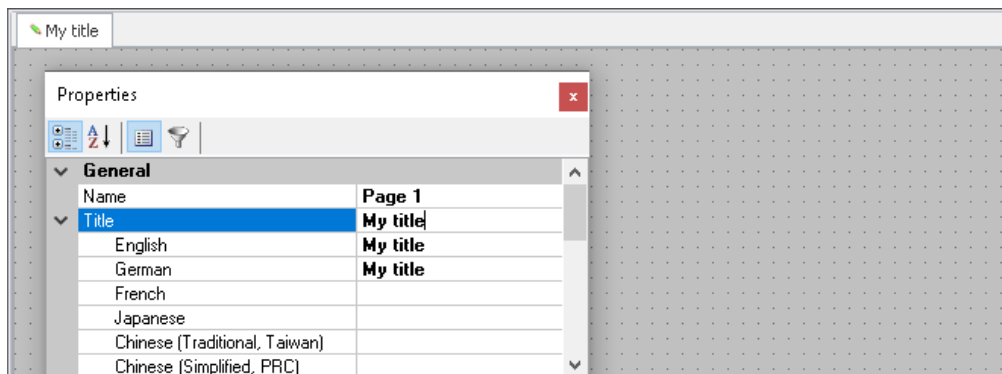


Example - Title of the Panel page

In most cases, the operating system's language and English are defined, unless English is already the operating system's language.
For the following example cases, the operating system language is (1): *"English"* (2): *"German"*.

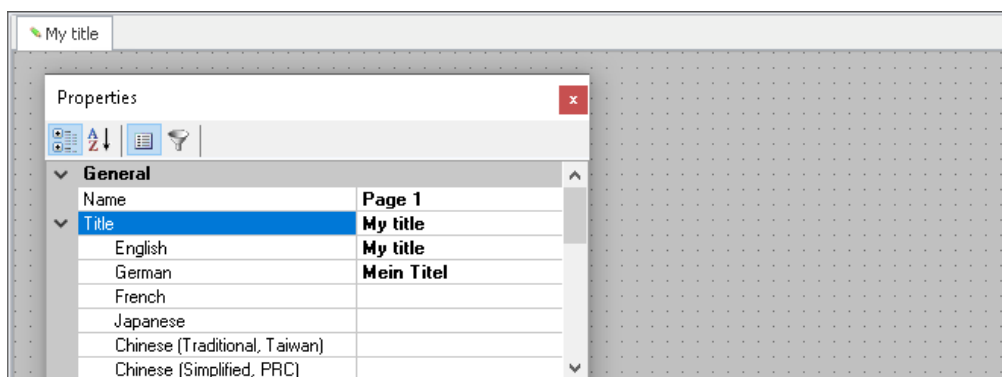
Case 1: You change the title in the main branch (all existing titles are the same):

The following is implemented: (1) the English entry is set to the new title | (2) English and German entries are set to the new title.



Case 2: You change the title in one language (e.g. German):

The following is implemented: German systems now display the new title.



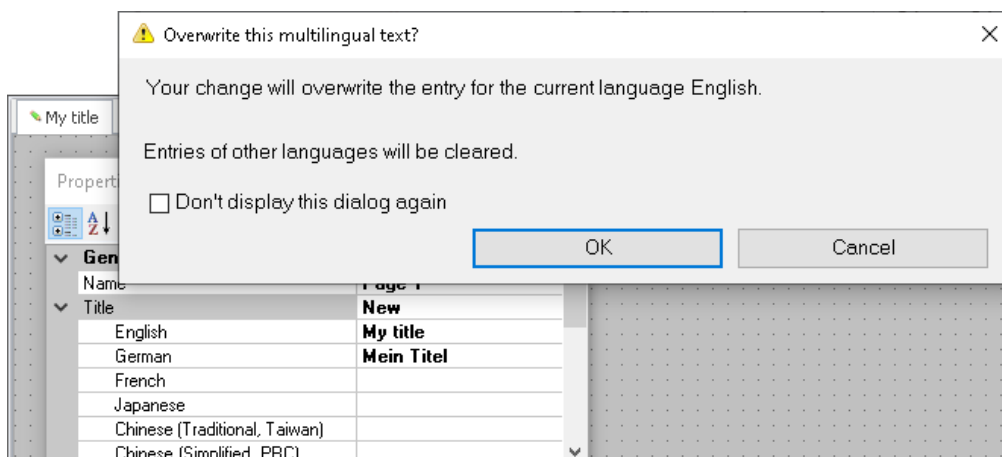
Fall 3: You change the title in the main branch while the titles in the other languages are all different:

The following is implemented: An error message is posted.

(1): "Your change will overwrite the entry for the current language English. Entries of other languages will be deleted."

(2): "Your change will overwrite the entry for the current language German and the entry of the default language English. Entries of other languages will be deleted."

The languages stated are assigned the new title; all others are deleted.



12 Sequencer, events and commands

Using imc STUDIO Monitor, you are able to create sequences, which carry out various commands in sequence either manually or automatically.

Sequences can be created in a variety of imc STUDIO Monitor components:

Component	Description
Panel	You can link events with commands by means of the " Event Dialog ". The commands are executed when the event occurs. Here, you can use selected commands.

The sequence

Name	Hint
- Switched Off #01 Execute menu action: Stop (Stop measureme... #02 Run imc FAMOS sequence #03 Browse in workspace - Panel: Panel/Report - Switched On #01 Execute menu action: Start (Start measurem ... #02 Browse in workspace - Panel: Panel/Messung	Occurs when the switch has been switched off Execute menu action: Stop (Stop measurem... Occurs when the switch has been switched on Execute menu action: Start (Start measurem... ...

Sequence table: Example having multiple commands

A sequence consists of a table (referred to below as the "*Sequence table*"). The commands appear there in a tree diagram. The structure and complexity of the Sequence table depends on the component used.

 Note

Not all commands have any functionality in imc STUDIO Monitor

For the Panel, certain buttons are provided, by means of which various commands can be run. The list of commands is comprehensive and provides access to all kinds of actions in imc STUDIO. But be aware that not all of these are appropriate or functional for purposes of imc STUDIO Monitor.

The descriptions of commands can contain references to chapters of the imc STUDIO documentation. In such cases, the pertinent function has no appropriate use in imc STUDIO Monitor.

Many application cases are described in the context of imc STUDIO. Some of these are also applicable to imc STUDIO Monitor.

Chapter overview

Synopsis	Section
Creation and configuration of a sequence of commands. Adding, moving and making settings for commands.	Creating a sequence of commands  951
Events relating to Panel-switches	Event Dialog  948
Description of all commands.	Command Reference  954

12.1 Sequence table

The sequence table contains multiple columns.

Status	Command	Comment	Enable	Stop on error
 Done	Set variables		☒	☒

Columns - Command sequence table

Column	Description										
Status	In this column, the commands' status is indicated. <table><tr><th>Status</th><th>Command</th></tr><tr><td> Done</td><td>Set variables</td></tr><tr><td> Error</td><td>Browse in workspace - Panel: Panel/Page 1</td></tr><tr><td> Running</td><td>Starts the current experiment's measurement</td></tr><tr><td></td><td>Run imc FAMOS sequence</td></tr></table> Status in the sequence table • Empty: The initial state before the sequence's first start. • Running: The command is still running and has not been concluded. • Done: The commando was concluded successfully. • Error: The command was concluded by an error.	Status	Command	 Done	Set variables	 Error	Browse in workspace - Panel: Panel/Page 1	 Running	Starts the current experiment's measurement		Run imc FAMOS sequence
Status	Command										
 Done	Set variables										
 Error	Browse in workspace - Panel: Panel/Page 1										
 Running	Starts the current experiment's measurement										
	Run imc FAMOS sequence										
Command	The command's name. The text entry can not be changed. The command's settings are accessible via the button at the cell's right edge.										
Comment	Click in this cell and enter a comment.										
Enable ⁹⁴⁷	Indicates whether a command is run or ignored. By default, this checkbox has a mark in it, which means that the respective command will be performed during runtime. However, you can also deactivate a command by de-selecting this option.										
Stop on error ⁹⁴⁷	With this option, the sequence is halted ⁹⁵⁰ when an error occurs while the command is running.										

"Enable" and "Stop on error"

In the sequences, you are able to activate/deactivate individual commands. Additionally, you are able to make a setting there which governs whether the [sequence should be interrupted](#) ⁹⁵⁰ if an error occurs.

Link events with commands					
Status	Name	Comment	Enable	Stop on error	
▼ 	Button1				
▼ 	Clicked	Occurs when the ...	☒	☒	
	#01 Run imc FAMOS sequence: ...		☒	☒	
	#02 Export parameters		☒	☒	
	#03 Show Panel page as dialog:		☒	☒	
	#04 Export Panel page		☒	☒	

Commands associated with a button's event

You are able to deactivate complete events along with their commands, or only individual commands.

Example of "Enabled": A Timer-event is to start only once its configuration has been completed. You are able to deactivate it and to later re-activate it at the time desired.

Events and their commands

Status	Name	Comment	Enable	Stop on error
▼	Panel	Experiment-specific events of the Panel		
▼	Page 1	Experiment-specific events of the Panel		
▼	Button1	Experiment-specific events of the Panel		
▼	Clicked	Occurs when the button has been clicked.	☒	☒
	#01 Run imc FAMOS sequence	...	☒	☒
	#02 Export parameters		☒	☒
	#03 Show Panel page as dialog:		☒	☒
	#04 Save Panel page as PDF		☒	☒
>	Project	Project-specific events		
▼	Sequencer	Experiment-specific events of the Sequencer		
	Device_AfterCheckConfiguration	After checking the configuration of a device	☒	☒
	Device_AfterRequestConnect	After trying to connect to a device	☒	☒
	Device_AfterRequestDisconnect	After trying to disconnect from a device	☒	☒
▼	Device_BeforeCheckConfiguration	Before checking the configuration of a device	☒	☒
	#01 Import variable		☒	☒
	Device_BeforeCreateDiskStart		☒	☒

Commands in the Sequencer associated with the events
1: Applies to the respective command
2: Deactivation applies to all commands associated with the event

If the option is **deactivated for the event**, this applies to **all associated commands**. If the option is activated for the event, the setting is applied to the respective command.

12.2 Event Dialog

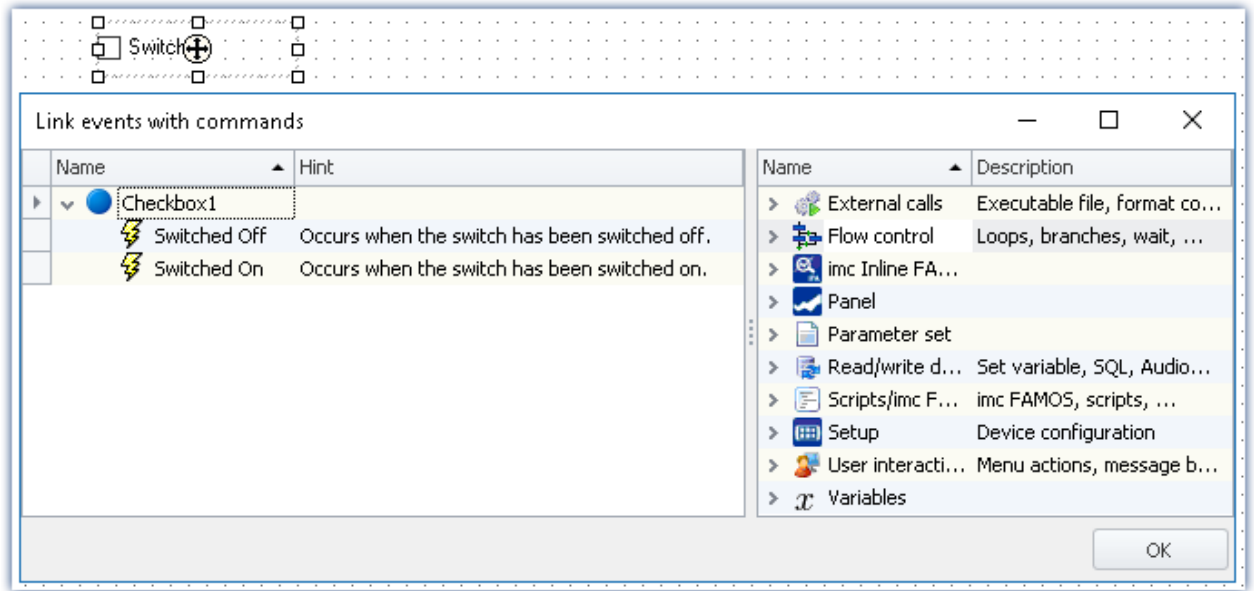
Certain **events** can be **linked with commands**. The commands are performed **upon occurrence of the event**. For configuring the event, the "event dialog". Events can, for example, occur in the Panel when a switch is activated.



Example

As an example, the two events (⚡) belonging to a "Switch" from the group "Standard":

Event	Description
⚡ Switched off	from On to Off
⚡ Switched on	from Off to On



"Event Dialog":
Events belonging to a switch from the group Standard

12.3 Running and stopping

The sequence is read, analyzed and implemented step by step. Analysis of the commands occurs at runtime. The top command in the sequence table is the first to be run.

Except in cases of intentional intervention, a **running sequence only stops** when the **last command** has been run.

Targeted running of commands and events via the context menu

Action	Description
Start	Starts the sequence from the beginning of the table.
From selected command to end of group	Starts the sequence from the location where the context menu had been opened.
Only selected command	Only the command selected is run.

Stopping due to an "Error"

When there is an **error notification**, it usually means that something is interfering with the processing of the sequence. For this reason, **all sequences are immediately stopped** (including events). This is independent of the error notification's source. The notification may not have anything to do with the sequence.



Note

Deactivating "Stop on error"

If you wish that a **command is not stopped by an error notification**, then remove the check mark from the "Stop on error" box for the respective command. If an error is reported while the command is running, the sequence is not interrupted. This can be necessary in cases where, at certain points in the sequence, errors are possible but should not interrupt the sequence because they are handled in a different way.

Treat error as warning

Some commands offer the ability to **convert "personal" error messages to warnings**. E.g. the "[Load/Refill variable](#)"⁹⁹⁸ command with the option "Treat error as warning".

The command can return an error if, for instance, a variable can not be loaded because it is already present. So if you wish to load multiple variables from a file, of which one of these might already be present, you can put a checkmark in this option's box. Then the message indicating that the variable can not be loaded will be outputted as a warning.

12.4 Creating a sequence of commands

The following examples are illustrated in the sequence table. However they also apply to the "[Event dialog](#)"⁹⁴⁸.

To add a command to the sequence table, there are two options:

by Drag&Drop	• Use Drag&Drop to move the command from the "Commands" tool window to the desired position. • Release the mouse button.
by the context menu ⁹⁵³	• Open the context menu of the desired position and select a command.

You can place a command at different positions.

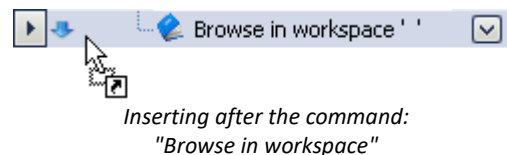
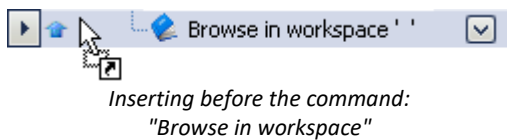
	Before the current position
	Below or within the current position: events (following also called group)
	After the current position

Examples of Drag&Drop

Use Drag&Drop to move the command from the right side of the dialog (in the "*event dialog*") to the desired position.

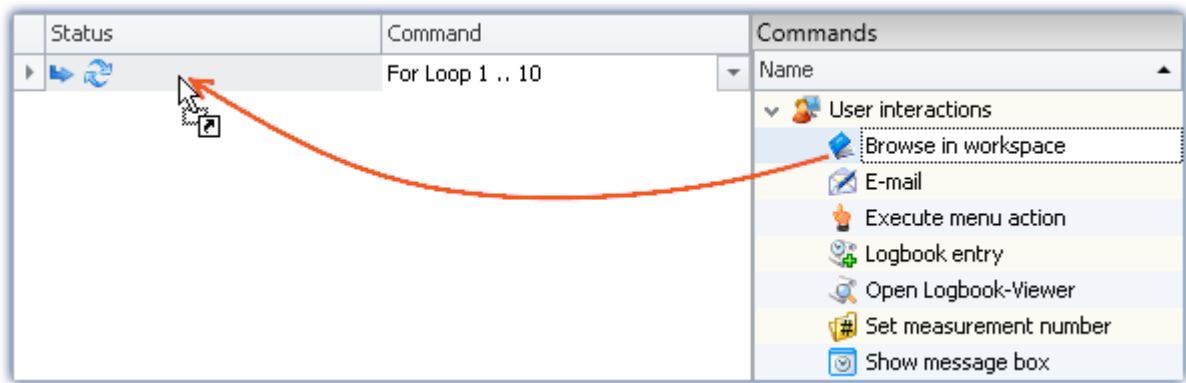
Adding commands before or after existing commands

Commands can be added either **before** or **after** an existing command. To do this, move the mouse either a little upward/downward.



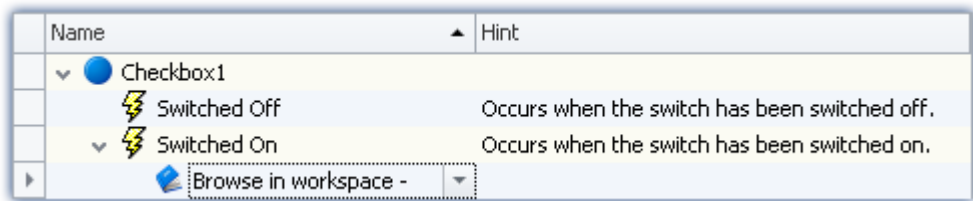
Adding commands to a group

Use Drag&Drop to place the command in a group, as seen below:



Example: Using Drag&Drop to add the command:
"Browse in workspace" to the loop

To the left of the even the position is specified (). Release the mouse button. The command is added to the group:



Result: Event dialog

Deleting a command

To delete selected commands, highlight them and select the context menu item "Delete".

Action	Description
Delete	Deletes the selection
Delete all	Deletes all commands in the Sequence table

Configuring a command

Most commands need to first be configured appropriately before they will work. E.g., when using "Export", a setting must be made to specify what to export and to where, etc.

To make the configuration, select the command in the sequence table. **Click on the button** to the right of the command's name. In most case, a configuration dialog will open.

The **various configuration possibilities** are presented in the description of the commands: "[Command Reference](#)".

12.5 Context menu

With the context menu, you can control all of the Sequencer's functions. In order to open the context menu, click on the right mouse button in the free area of the sequence table.

Menu item	Description
 Add or New ⁹⁵¹ before/after/into	Sets up the selected command at the specified location.
 Cut	Clips out the selection and moves them to the Clipboard.
 Copy	Copies the selection to the Clipboard.
 Paste	Inserts the content of the Clipboard at the selected position.
 Delete	Deletes the selection
 Delete all	Deletes all commands in the Sequence table or in the events.
 Start	
Start	Starts the sequence from the beginning of the table.
From selected command to end of group	Starts the sequence from the location where the context menu had been opened.
Only selected command	Only the command selected is run.
 Expand/ Collapse	Expands/collapses either all groups or the selected group.

12.6 Information and tips



FAQ

The command-sequence is interrupted upon every error

The command-sequence is interrupted upon every error. What is the best way to prevent this?

Example: While the command: "Wait" is being executed, another action causes an error. In consequence, the sequence is cancelled.

Solution: For this command, deactivate the associated checkbox "[Stop on error](#)"⁹⁴⁷ in the Sequences table. With this setting, any error messages posted at that position will be ignored during runtime. As a result, the subsequent command will be executed even if an error occurs. Please be aware that this means even the running command's errors will be ignored.

If it is possible, verify later that the action was performed correctly.

Here is an example involving the command "Execute menu action" > "Start measurement". The sequence is not intended to end even if the command fails. Using the IF-command, you can check the value of the variable "Measurement status" and thus verify whether the measurement is actually running. This gives you the ability to respond to the result in a targeted way and close the Sequencer properly, if appropriate.

12.7 Command reference

Commands allow the execution of targeted actions such as jumping to a Panel page, running a imc FAMOS sequence, or printing out a Panel page as a report.

Commands are available in various areas of imc STUDIO, imc WAVE and imc STUDIO Monitor:

- imc STUDIO Sequencer
- imc STUDIO Panel
- imc STUDIO Automation

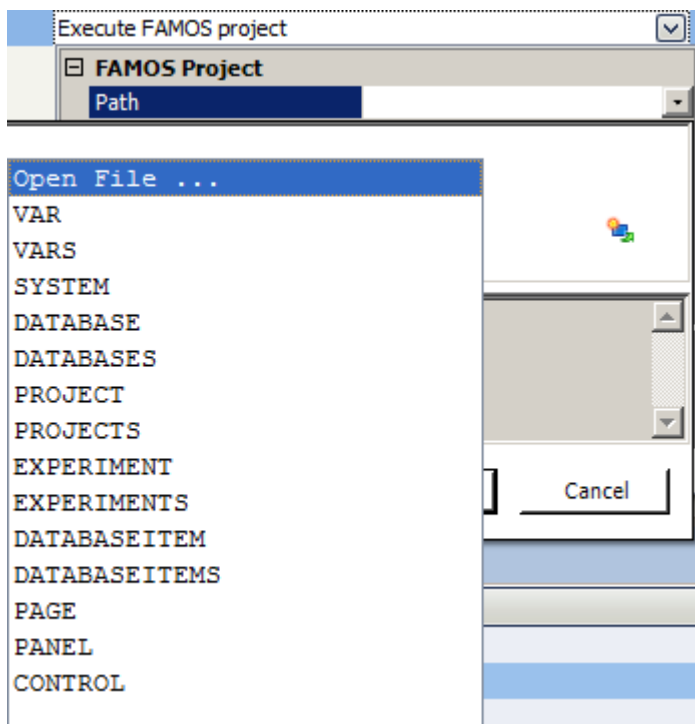
Note

Not all commands described here are available in all areas mentioned, since not all of them are relevant to every area. The commands respectively available are displayed in the associated command list.

12.7.1 Data analysis and Scripting

12.7.1.1 Execute imc FAMOS project

A imc FAMOS project is selected and started:



Command: *Execute imc FAMOS project*

Click on the symbol  and select *Open File...*

In the project folder, select the file *Default.FamosPrj*.

12.7.1.2 Run imc FAMOS sequence

Status	Command
	Execute FAMOS sequence

From within imc STUDIO it is possible to transfer variable currently being measured to imc FAMOS and to process them with a imc FAMOS sequence there. The results of the imc FAMOS sequence can be transferred back to imc STUDIO and displayed in a Panel page, for instance.



Note

Prerequisite

The prerequisite is installation of imc FAMOS on the same (see "*Technical data sheet*" > "*Additional imc Software Products*").



Reference

Functions

Information on imc FAMOS' scope of functions is presented in the separate imc FAMOS manual

To **open the imc FAMOS Dialog**, click on the button (symbol with three dots) in the sequence table.



Reference

imc FAMOS dialog

More information on the imc FAMOS dialog is provided in the section:
 "Data analysis and signal processing" > "[imc FAMOS dialog](#)".

12.7.1.3 Run Python code file

The Python command supplies functions which provide a bridge to the Python programming language. imc STUDIO Monitor generates an embedded instance of the Python runtime environment, which provides an interpreter for the Python programming language.

For the purpose of analyzing the measured data, a Python code file is selected. The variables to be analyzed with Python can be selected, and the results can be returned in variables.

System requirements

Software	Compatible version	Installation	Notes
Python 64-bit	3.12, 3.11, 3.10, 3.9, 3.8	required	Support is provided exclusively for the Python reference implementation from the " <i>Python Software Foundation</i> " (CPython) in one of the versions listed below; these can be downloaded and installed from https://www.python.org/ .
NumPy 64-bit	2.1, 2.0, 1.23, 1.20, 1.19	optional	The Python bridge provides special support for data types defined in the expansion library "NumPy" (https://numpy.org).



Note

Installation

- Python and NumPy are not included on the installation medium.
- In order to facilitate subsequent installation of packets, it is recommended that you activated the option "*Add python.exe to PATH*".

Configuration

In the command configuration, the Python code file is selected and the transfer of variables to and from Python is configured.

Run Python code file

Path to Python file <EXPERIMENT.PATH>\Meta\PythonTest.py ...

Input variables

▶ Channel_001

☀

Output variables

▶ MeanValue

☀

☒ Synchronous event

OK

Cancel

Properties of the command "Run Python code file"

Parameter	Description
Path to Python file	Path to the executable Python code file (*.py). Ideally, the file is saved in the Meta-folder. The content of this folder is also included when performing Save As/Export. <i>Note:</i> For the file selection, the current experiment-path is replaced by the experiment-placeholder. In consequence, no modifications are necessary when the experiment is saved under a new name.
Input variables	The variables entered are passed to the Python-sequence. The variable names must match those in the Python-sequence. The spelling is case sensitive. Transfer and processing of the variables is performed on the basis of the "Current Measurement" and not on the basis of any saved (earlier) measurements.
Output variables	The variables calculated by Python are assigned to imc STUDIO Monitor via the table. The variable names must match those in the Python-sequence. The spelling is case sensitive. If the variable does not yet exist in imc STUDIO Monitor, it will be created. <i>Note:</i> A channel can not be transferred back to a "User-defined Variable" of the type "Numeric" (single value). For this purpose, please modify the type in Python if you wish to obtain a single value. If you require a channel, use as the target a variable created using Python. This will then always be assigned the appropriate data type.

Data type conversion to Python

The data type of the Python variables generated is automatically determined. The default Container datatype is "*numpy.ndarray*".

Here, the following rules apply:

imc STUDIO Monitor data type	Python data type
Single values	
8-Byte Integer (full-scale)	Integer ('int')
Miscellaneous numerical data formats	Real number, 8-Byte ('float')
Normal data sets	
8-Byte Real with length = 0 (constant 'EMPTY')	None ('NoneType')
8-Byte Integer (full-scale)	NumPy array ('numpy.ndarray' [..'int64'..])
1-Byte Integer unsigned (full-scale)	Byte array
Miscellaneous numerical data formats	NumPy array ('numpy.ndarray' [..'float64'..])
TimeStampASCII and other special formats	not supported
2-component data sets	
Complex, Cartesian, Length = 1	Complex number ('complex')
Complex, Cartesian, Length <> 1	NumPy array ('numpy.ndarray' [..'complex128'..])
Complex, Magnitude/Phase	not supported
XY	not supported
Structured data (Events/Segments)	
Segments ... Integer 1 Byte unsigned (full-scale)	2-dimensional NumPy-array of Bytes ('numpy.ndarray' [..'uint8'..]). Each segment forms one row of the matrix.
Segments ... miscellaneous formats	2-dimensional NumPy-Array ('numpy.ndarray'). Each segment forms one row of the matrix. The data type of list/array elements is determined as described above.
Events	not supported
Miscellaneous data types	
Text	String ('str')
Text array	List of strings ('list' [..'str'..])

 Note
Channel properties

The properties transferred include: sampling rate and unit, as well as all metadata saved with the channel (see "[Traceability of data](#)"^[144])

Event-based data sets

It is not possible to apply calculations to event-based data sets. A channel automatically becomes an event-based data set when the parameter "*Available events*" under "*Data transfer*" is set to "*all*". Set this parameter to "*last*" in order to obtain a normal data set.

Special characters

The rules for special characters in Python-variable names must be observed. Some characters which are allowed for imc STUDIO-variables are not allowed. Such variables which have a period in their name, for instance, pv-variables, can not be transferred.

Notes on the type 'numpy.ndarray'

This data type is defined in the very prevalent 'NumPy'-package. It is optimized for efficient data storage and high-speed processing of multi-dimensional arrays. The NumPy-arrays generated are saved in "C-contiguous" (C_CONTIGUOUS) style.

Transfer format from and to Python

For each variable, a dictionary of values and properties is transferred in the following form:

<Variable name>	Name of dictionary in Python	Remarks
"Values"	<Data of the variables>	"Values" = fixed name for accessing the data
<Property name 1>	<Property value 1>	<Property name> = Name of the variable's property
...	...	
<Property name N>	<Property value N>	

**Example****Transferring: Channel_001**

```
Channel_001{
'Values': array([ -1.40389428,  17.09,  35.64, 317.28]),
'eDisplayName': 'Channel_001',
'eXFormat': 1,
'eDataType': 1,
'eCategory': 'Analog',
'eSampleTime': 0.0002
}
```

Accessing Python:

```
data = Channel_001['Values']
SamplingInterval = Channel_001['eSampleTime']
```

The output variables can be returned with or without a dictionary. If the dictionary contains no data or no "Values" key, a variable without data is generated.



Example

Event variable for imc STUDIO

Transferring to imc STUDIO: result1

```
result1 = {  
    'eXFormat': 1,  
    'eSampleTime': 0.0001,  
    'eChannelComment': "calculation of..",  
    'Values': data  
}
```

Variable: imcVariables

Initially, the system generates and transfers a variable "*imcVariables*", which contains all names of the variables transferred to Python. By means of this variable, it is possible, for instance, to run a particular sequence in the Python script according to which variable is transferred.



Example

imcVariables

Transferring: Channel_001 and Channel_002

```
imcVariables = ["Channel_001", "Channel_002"]
```

Accessing in Python:

```
imcVariables[0] -> 'Channel_001'
```

Data type conversion to imc STUDIO Monitor

The data type of the imc STUDIO Monitor-variable generated is determined automatically from the data type of the Python-variable.

Here, the following rules apply:

Python data type	imc STUDIO Monitor data type
Standard types	
'NoneType'	Data set of length 0; data format: 8-Byte Real (like constant 'EMPTY')
'int'	Single value; data format: 8-Byte signed Integer
'float'	Single value; data format: 8-Byte Real
'complex'	Complex data set (RI) of length 1; data format: 8-Byte Real
'bool'	Single value; data format: 8-Byte Real
'str'	Text
Container types 'tuple', 'list', 'range', 'set', 'frozenset'	
with element number 0	Data set of length 0; data format: 8-Byte Real
all elements of type 'float'	Data set; data format: 8-Byte Real
all elements of type 'int'	Data set; data format: 8-Byte signed Integer
all elements of the type 'boole'	Data set; data format: 8-Byte Real
all elements of the type 'complex'	Complex data set (RI) of length 1; data format: 8-Byte Real
all elements of the type 'str'	Text array
all elements of a 1-dimensional container, where all containers have the same length N and all elements the same numerical type (float, int, complex)	Segmented data set; segment length: N
other element-types or various types	not supported
Miscellaneous container-types	
'dictionary'	not supported
'bytes', 'bytearray'	Data set; 1-Byte unsigned Integer

NumPy-Arrays ('numpy.ndarray')

This data type is defined in the widely prevalent 'NumPy'-package. It is optimized for efficient data storage and high-speed processing of multi-dimensional arrays.

Homogeneous arrays of Dimension 1 or 2 are supported. For Dimension 2, a segmented data set is generated, where the rows of the NumPy-array form the segments. The storage layout must be either "C-" or "F-contiguous" (C_CONTIGUOUS, F_CONTIGUOUS) and the data must be saved in "Little-Endian-Byte order".

NumPy-Array-element type	imc STUDIO Monitor data type
'float64', 'float32'	Data set (8 Byte Real)
'float128', 'float16'	not supported
'int8', 'int16', 'int32'	Data set (8 Byte Real)
'int64'	Data set (8-Byte signed Integer)
'uint8'	Data set (1-Byte unsigned Integer)
'uint16', 'uint32'	Data set (8 Byte Real)
'uint64'	not supported
'bool'	Data set (8 Byte Real)
'complex64', 'complex128'	Complex data set (RI, 8 Byte Real)
'str'	Text array
'datetime64', 'timedelta64'	not supported
'bytes' and all other types	not supported



Notes

User-defined variable as the target

These variables are automatically converted to an appropriate data set. If the target variable had previously been defined in imc STUDIO Monitor as a user-defined numerical variable, then a single variable can be assigned to this variable. The variable remains a single value.

Behavior

- If an input variable does not exist, an "empty channel" is transferred to Python. This will not appear in the Data Browser, but is only available in the Python-environment.
- When a variable which Python requires is not transferred, an error is thrown upon execution.
- If any output variable is defined which Python does not use, a warning appears in the logbook.
- Upon starting the sequence, the lists of source- and results-variables are checked for illegal names. If any name is illegal, the program aborts and posts a message accordingly. An example of an illegal name would be "imcVariables".

Additional notes

- The asynchronous mode is not yet available at this time, for which reason the option "*Synchronous event*" is disabled (appears in gray).

Examples

 **Example**

Calculating the sum of all samples

```
import math
pySummeSine = math.fsum(Sine)
```

 **Example**

Adding all data sets transferred

Transferring:

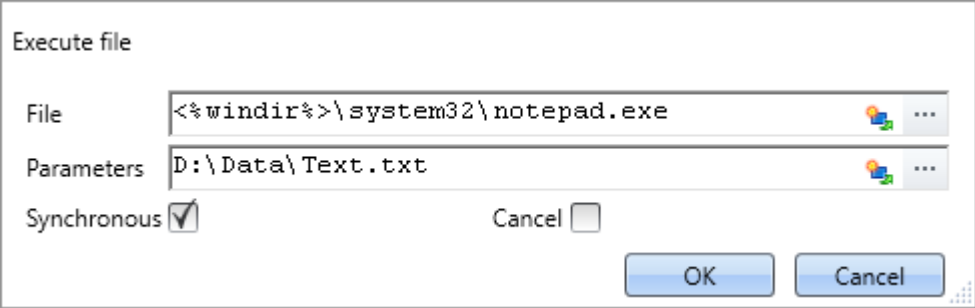
```
imcVariables = ["Channel_001","Channel_002","Channel_003"]

allData = locals()[imcVariables[0]]
allData = allData['Values']
arraysToAdd = len(imcVariables)
for i in range(1, arraysToAdd):
    varData = locals()[imcVariables[i]]
    allData = np.add(allData, varData['Values'])
```

12.7.2 External calls

12.7.2.1 Execute file

The command starts a Windows program.



Execute: In the example, Notepad is started

Parameter	Description
File	Display text in the sequence table (without effect on the dialog).
Parameters	The dialog's header.
Synchronous	When activated, the Sequencer remains on this command until either the program or the file is closed.
Cancel	Activates the Timeout function. At the latest, after a specified timeout time, the Sequencer runs the next line.
Timeout	Duration in seconds [s] until the timeout occurs.

12.7.2.2 Format converter

The imc Format Converter provides the ability to convert measured data to a different format, e.g. to EXCEL data format. For instance, the imc Format Converter can operate as a command performed automatically at the end of a measurement for the purpose of converting the data to the desired format.

Reference

You can also use the imc Format Converter as a **standalone** program, if imc STUDIO or imc FAMOS has been installed; see the chapter "[imc Format Converter](#)"⁵⁷⁵

Export formats

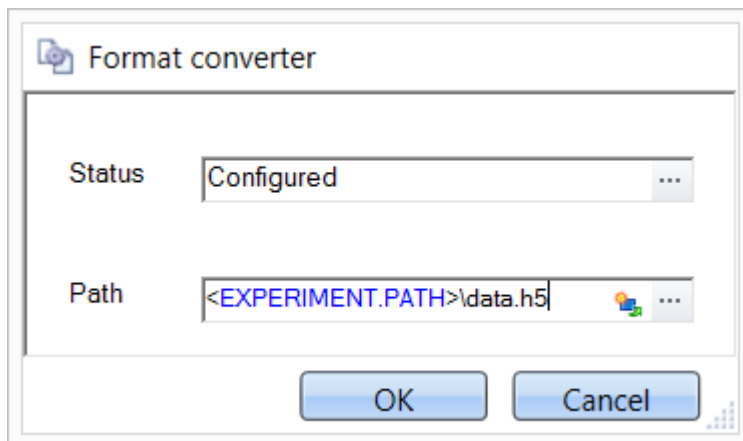
The same export formats are supported which are also offered by imc FAMOS. These include:

- user-defined ASCII export-formats
- user-defined EXCEL export-formats
- MDF2.0, MDF3.0, MDF4, ASAM ATFX, ASAM ATFX NVH, Catman 5.0
- Google Earth Export, HEAD acoustics (also 4.5 compatible)
- Matlab 4 and 5, nSoft-DAC, RPC-3, Somat SIF (nCode output)
- TEAC TAFMat and TEDAM, DIAdem TDM and TDMS
- Binary Universal File Format
- Universal File Format (UFF)

12.7.2.2.1 Format converter as command

Calling

Via the Sequencer: "External calls" > "Format converter":



Configuration of the command: Format Converter

Status

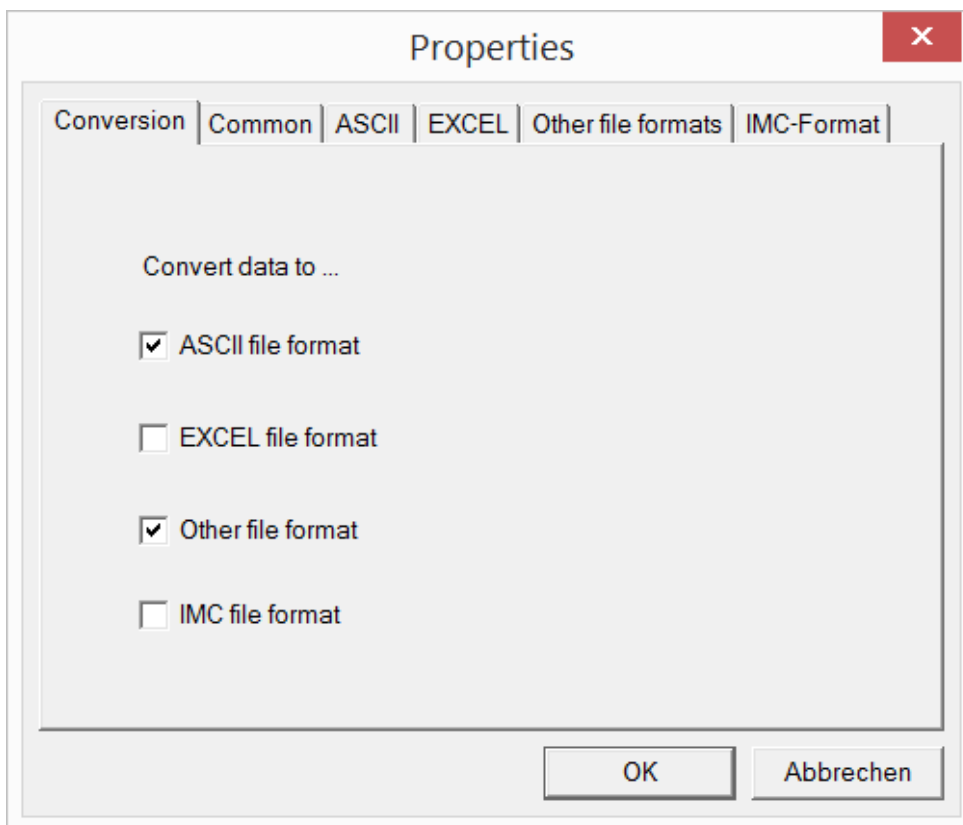
Immediately upon opening the command's configuration, the status is set to "not configured".

Path

- If just **one** file (*.dat, *.raw) of measured data is specified here, you can select whether to **convert either only the single file, or all files of measured data** which the **folder** contains.
- If a **folder** is specified here, then **all files of measured data in the folder** are converted:
 - If the option "Save all files of a folder in one file" among the settings on the tab page "Common" (see the segment on "Status") is selected, then **all files of measured data** present in the specified folder are converted to a **single target file**.
 - If the option is **not selected**, then each file of measured data which the folder contains, the data are converted to a **separate file**.

Using the button  in the box "Status" you can open a dialog via which you can make all the settings for the conversion:

- Conversion: Target format. It is also possible to select multiple formats.
- Common: Storage location and whether individual files or a multi-file is to be created.
- ASCII: Selection of the export template and specifying of the file extension.
- EXCEL: Selection of the export template and specifying of the file extension xls or.xlsx.
- Other file format: Selection of the format and if necessary its formatting options.
- IMC-file format: Save the data to a file in the imc format, or as a copy to a different memory location.



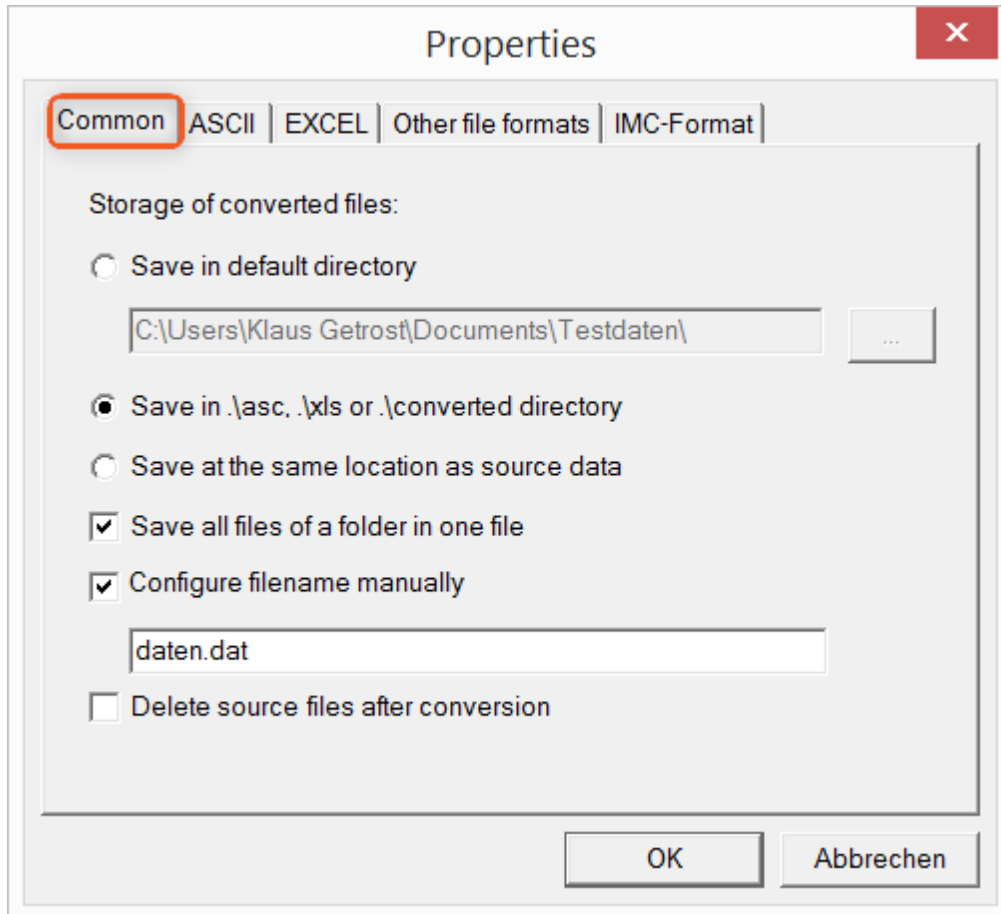
Setting the Format Converter

12.7.2.2.2 Settings

In the settings you specify whether the data are saved either individually or together, which templates are used for ASCII and EXCEL, or what other formats are used.

The settings remain intact, so that under normal circumstances you can immediately select data and convert them.

Data saving



Storage location for converted files:

Save in default directory: Free choice of a **target folder**. Even network drives are possible.

Save in .\asc, .\xls oder .\converted directory: The data are saved in their source folder but additionally in a **subfolder**. For ASCII in the *asc* folder, *xls* for EXCEL and *converted* for any other export formats.

Save at same location as source data: The data are saved on the same **level of the tree diagram as the source data, without subfolders**.

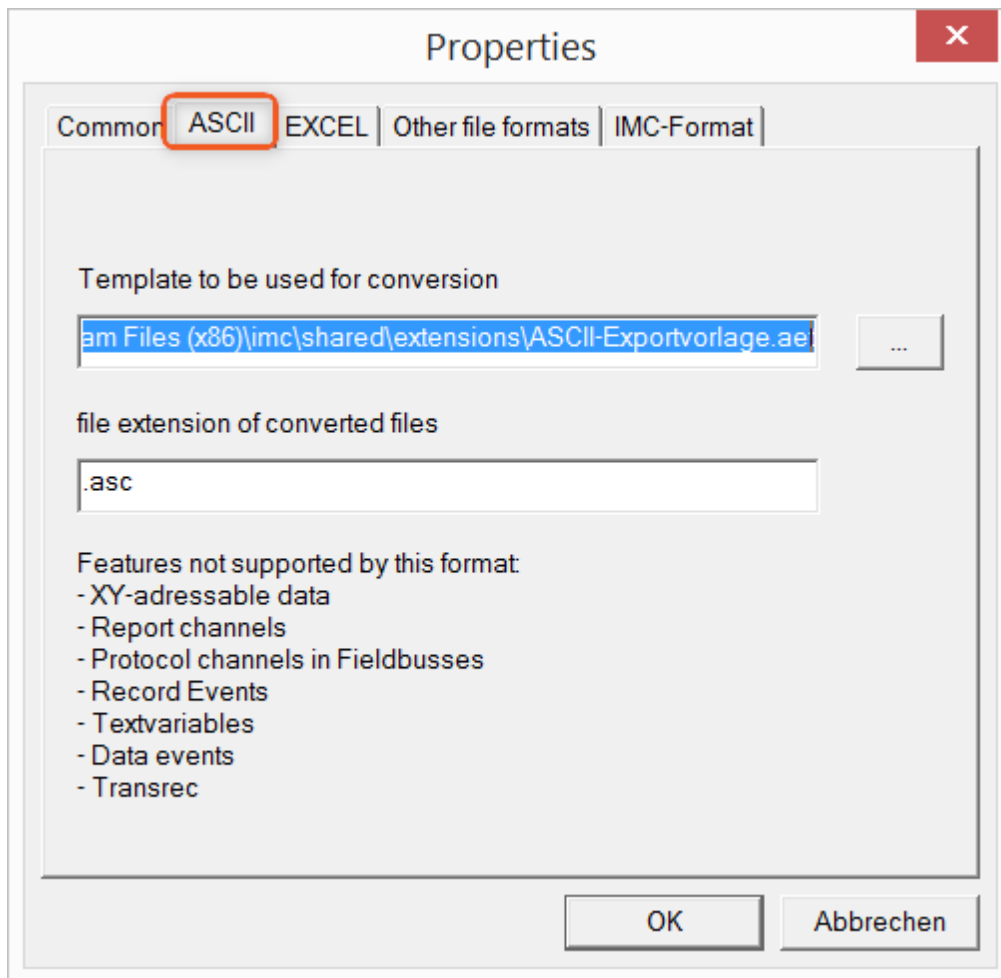
Manner of data saving:

Save all files of a folder in one file: All of a channel's files are saved **together in one file**.

Configure file name manually: **Name of the target file** in which all channel files are saved together.

Delete source files after conversion: Once the data have been converted, the **source files can be deleted automatically**, in order to make memory space free. This setting can make sense when all data are saved together in the imc format.

ASCII



Format template to be used for conversion:

By default, ASCII export templates are located in the folder "*C:\Program Files (x86)\imc\Shared\Extensions*" and use the file extension **.aet*. Owners of imc FAMOS are also able to personally create or modify these templates. In this case, the aet-files are saved in the folder "*C:\ProgramData\imc\Common\Def*".

By default, the file extension is "*asc*", but any arbitrary one can be specified here.

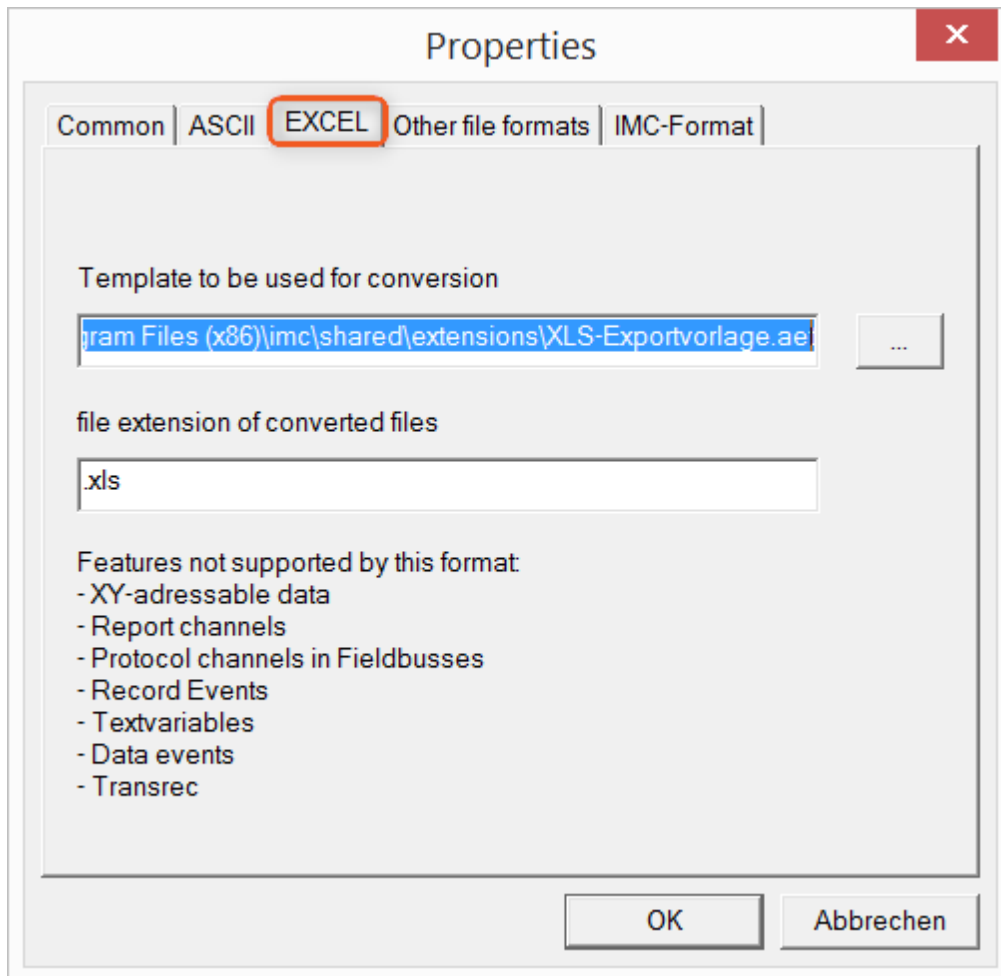
Special variable types which can **not** be exported as ASCII data are listed on this tab page.

! Notes

File extensions

In imc FAMOS the file extension is specified with the ASCII template. Within imc FAMOS this file extension is added to the ASCII file. With imc Format Converter the file extension specified here is used instead.

EXCEL



Format template to be used for conversion:

For EXCEL files, a similar technique is used as for [ASCII](#)⁹⁶⁶. By default, the file extension is "XLS", but can be arbitrarily specified here.

Variable types which are **not** exported correspond to those for [ASCII-export](#)⁹⁶⁶.

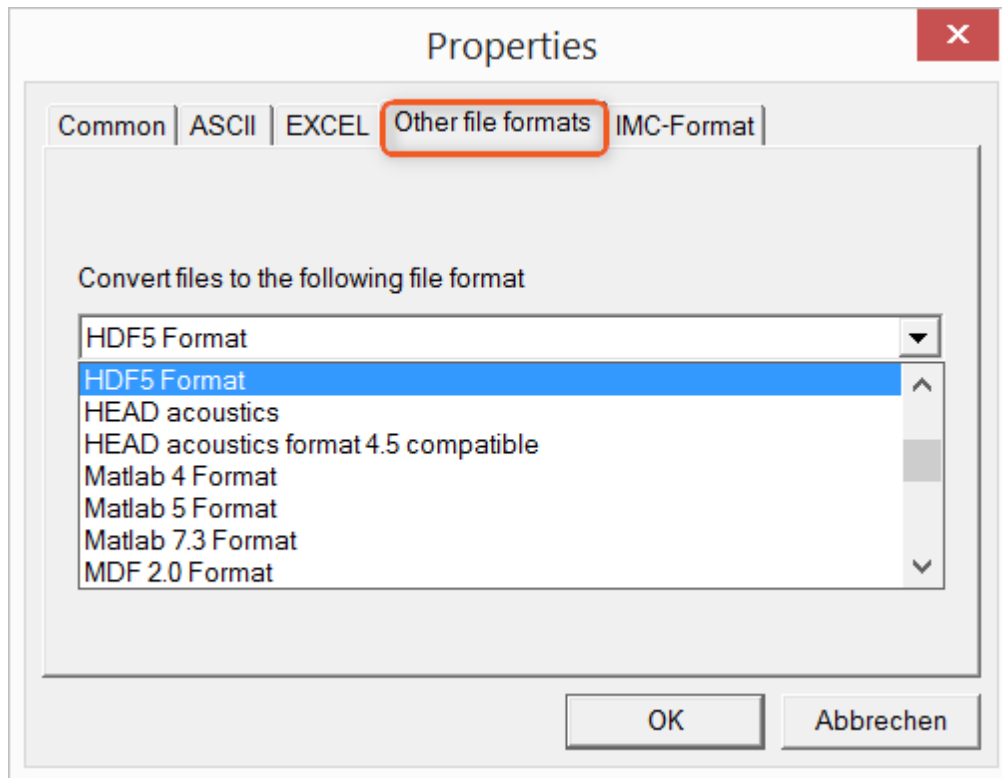


Notes

File extensions

If the XLSx format was selected as the export template, it must be here as the file extension. The imc Format Converter does not import the specified file extension from the AET file.

Other file formats



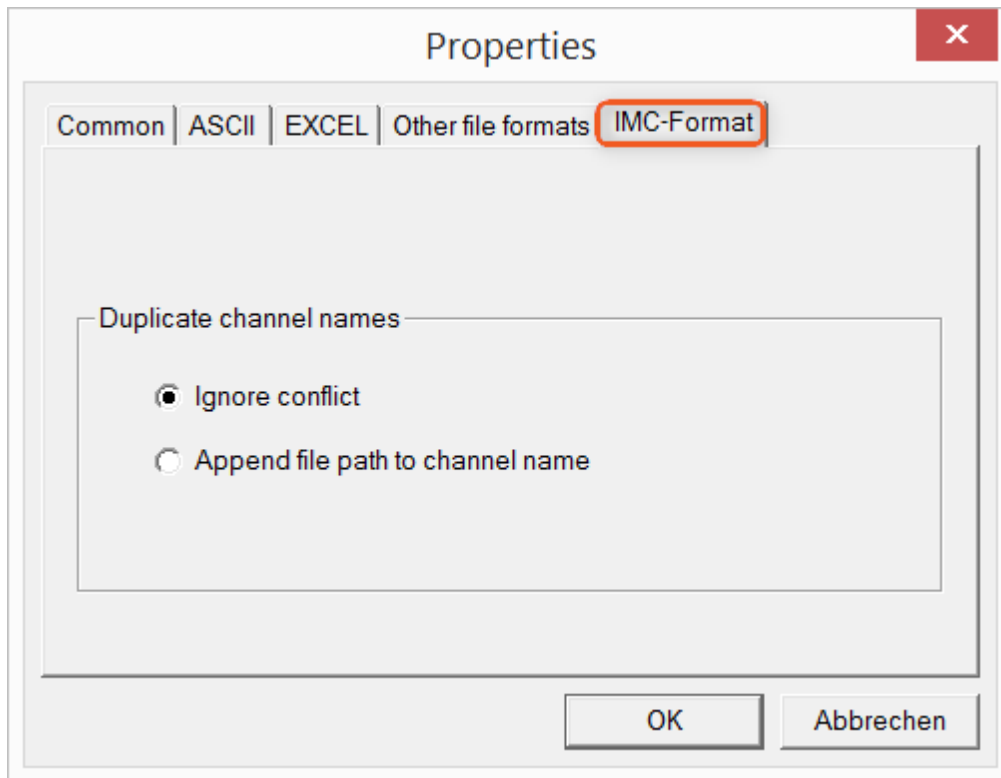
Convert files to the following file format:

Selection of the format in which the data are to be saved. Installation of the imc Format Converter makes certain export formats available.

Options

If the format selected comes with additional export options, these can be set by means of the dialog opened by clicking on the button "*Options*". Descriptions of the options appear in the document *ImportExportFilter.pdf*, of which a copy is imported along with the imc FAMOS installation. Alternatively, the document is available from the [imc homepage](#).

imc-Format



Handling duplicate channel names

Ignore conflict: If files are overwritten due to duplicate channel names, the conflict is not reported.

Append file path to channel name: A unique channel name is generated from the file path.

! Notes

Why save in the imc-format?

The imc Format Converter is designed for imc data, so why is there an imc-Format option? imc devices allow complex data structures in which channels can be recorded at different moments in time (using triggers). For this reason, the channels are saved as individual files. Using the imc Format Converter, you are able to join the data sets accumulated into one single file after the measurement. One appropriate group of settings on the page "[Common](#)" to accomplish this would be:

Save at same location as source data = active

Save all files of a folder in one file = active

Delete source files after conversion = active

12.7.2.3 Run Managed DLL method

This command enables you to use methods from external **managed** DLLs in imc STUDIO.



Note

Notes on using this command

- Whether external DLLs and their methods **work correctly** or on the other hand even cause damage **can not be ascertained by** imc STUDIO, and thus also **not be prevented**!
- For the purpose of obtaining a good understanding of error messages and the necessary settings, skill in distinguishing between **unmanaged** and **.NET DLLs**, as well as basic programming experience, are **recommended**.
- Only primitive data types can be used.
- If you wish to use only one **WinAPI-DLL**, be **absolutely certain** that the **parameter signature** is correct. Otherwise, the imc STUDIO functionality could be affected.

Example of a configured command

Parameter	Description
Assembly	Clicking on the button  causes a .NET Framework Assembly to be selected which usually is of the type .dll.
Filter	You can set up a filter which hides any unnecessary method names from a list of available ones. The filters listed here are permanently programmed and cannot be changed.
Method	Here, methods offered in the assembly are listed, naturally only the ones designated as public.
Parameter - Return value	Description
Type	The data type's complete .NET name , which is returned by the method. This name must be outputted in its entirety in order to distinguish between any multiple declarations in different locations (in the example shown: "System").
Return value	The name of the imc STUDIO variable in which the value is to be saved; this box can also remain empty, if the return value is not needed.

Parameter - Parameters	Description
Type	The data type's complete .NET name , which is expected by the method. This name must also be specified completely in order to distinguish between any multiple declarations.
Parameter	The parameter's name in the method declaration.
Value	The value of the parameter to be transferred; this box comes with the same functions as Set Variables. If a variable from imc STUDIO is used here, it must necessarily be convertible to the target data types . If this rule is violated, then an error message appears in the log book at the moment the command is run.

12.7.3 Flow control

12.7.3.1 Stop Sequencer

The command is not used in imc STUDIO Monitor.

12.7.3.2 Wait

Using the command *Wait* you can halt the Sequencer. The waiting time can be set in the mode *Timeout* or be made to depend on a *Condition*.

Status	Command	Comment	Enable
	Wait: Timeout [s] 5 Wait Timeout Timeout [s] 5 Show Cancel dialog after delay: Immediately OK Cancel		

Flow control: Wait

Parameter	Description
Wait	Sets the mode to either Timeout or Condition .
Timeout [s]	In the <i>Timeout</i> mode, enter the waiting time here, in seconds. This can be achieved by use of a variable, or it can be a fixed value.
Show Cancel dialog after delay	If you wish to cancel the timeout early, you can have a "Cancel dialog" after a specified time.
Condition	In the mode <i>Condition</i> , set the condition here, e.g. <code>Virt_Bit01==1</code>

Note that a comparison is performed using the operator "==".

12.7.4 imc Inline FAMOS

12.7.4.1 Import imc Inline FAMOS source code

Using this command, you can import imc Inline FAMOS source texts. You can choose whether a text is to be immediately applied and run after the import, or initially only imported.

Import imc Inline FAMOS source code: imc Inline FAMOS

Task

Task nameimc Inline FAMOS

Path

Import path<EXPERIMENT.PATH>\config\Code.ifa.xml

Show dialog☐

Action

Task actionApply

OK

Cancel

Configuration: Import imc Inline FAMOS source code

Parameter	Description
Task name	Target name of the Task. If a Task of that name exists already, it is overwritten without confirmation prompt. If no Task of that name is present, a new Task of that name is created.
Import path	Specification of the file to import, including the pathname.
Show dialog	If this option is activated, the file selection dialog appears when the command is executed. Select the file desired. If a path has been defined, the selection dialog starts there. The selection does not take effect on the configuration of the command in the command list, but only on the import currently performed.
Task action	The imported Task can either be applied and run immediately, or initially only imported. • Apply: The Task is applied and the calculation started immediately (see also the information on the "Apply" Ribbon in the Data Processing) • Edit: The source text is imported, but the Task is not applied immediately. Thus, this does not cause the running calculation to be concluded. If the measurement is re-started, the Task is applied automatically.

12.7.5 Measurement

12.7.5.1 Set measurement number

The command *Set measurement number* assigns a measurement number to a measurement.

The available settings are:

- Set measurement number **to a fixed measurement name**
- Set measurement number **to a fixed index**
- Set measurement number **to the last completed measurement**
- Set measurement number **to a fixed offset before the last completed measurement**
- Remove measurement number

The numbers **1 through 99** can be used for assigning measurement numbers.



Reference

The last measurement

More detailed information on the "last measurement" is presented in the chapter:

"Storage options and directory structure" > "[The last measurement](#)"²¹⁹

set - to a fixed measurement name

The dialog box titled "Set measurement number" contains the following fields:

- Measurement number:** A dropdown menu showing "set - to a fixed measurement name".
- Symbolic measurement name:** A text field containing "Measurement# 1".
- Measurement name:** An empty text field.
- Buttons:** "OK" and "Cancel" buttons at the bottom right.

Command "Set measurement number to a fixed measurement name"

With this command, you can assign a number to a measurement which has a **particular measurement name** at the time the command is run.

In the box *Symbolic measurement name*, you specify the measurement number to be assigned. In the box *Measurement name* you can assign a name to the measurement receiving that number.

set - to a fixed index

The dialog box titled "Set measurement number" contains the following fields:

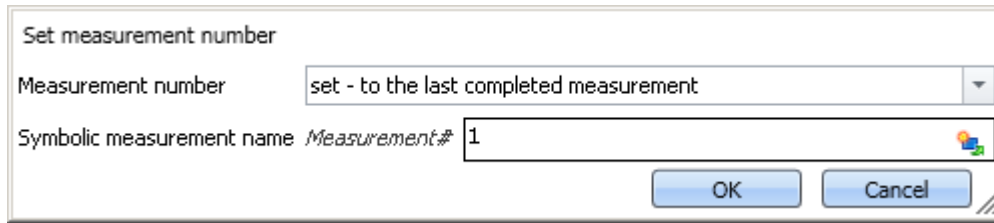
- Measurement number:** A dropdown menu showing "set - to a fixed index".
- Symbolic measurement name:** A text field containing "Measurement# 1".
- Sorting of measurements:** A dropdown menu showing "Time - ascending".
- Index:** A text field containing "1".
- Buttons:** "OK" and "Cancel" buttons at the bottom right.

Command "Set measurement number to a fixed index"

With this command, you can assign a number to a measurement which has a **particular index** at the time the command is run.

In the box *Symbolic measurement name*, you specify the measurement number to be assigned. In the box *Sorting of measurements*, you can specify what kind of sorting to apply to the value entered in the box *Index*.

set - to the last completed measurement

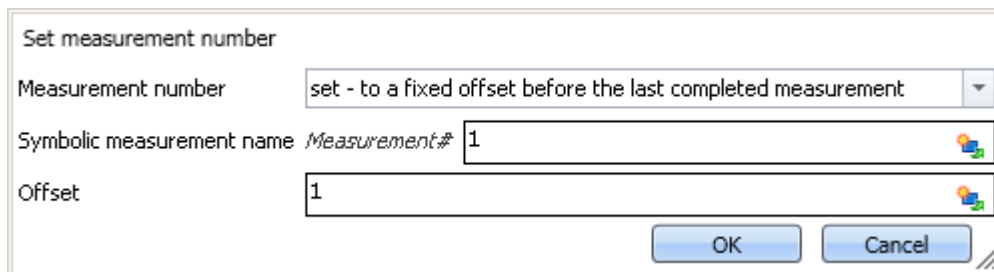


Command "Set measurement number to the last completed measurement"

With this command, you can assign a number to a measurement which is the **last completed measurement** at the time the command is run.

In the box *Symbolic measurement name*, you specify the measurement number to be assigned.

set - to fixed offset before the last completed measurement

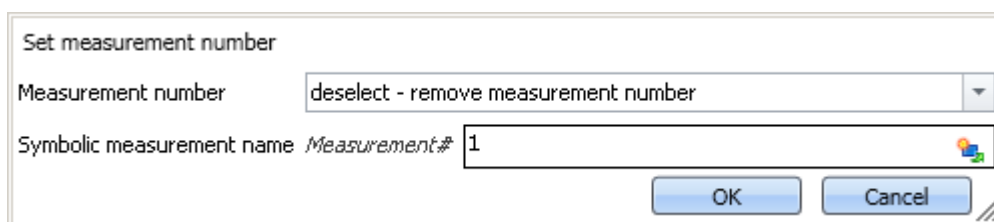


Command "Set measurement number to a fixed offset before the last measurement"

With this command, you can assign a number to a measurement which has a certain **offset to the last completed measurement** at the time the command is run.

In the box *Symbolic measurement name*, you specify the measurement number to be assigned. In the box *Offset*, you can specify how many completed measurements going back the measurement to be numbered is. For instance, an offset of 1 means that the next-to-last completed measurement is assigned the number.

deselect - remove measurement number



Command "Set measurement number - remove measurement number from a measurement"

With this command, you can delete an assigned measurement number from any measurement.

In the box *Symbolic measurement name*, you specify which number to delete. It does not matter to which measurement this number is assigned.

12.7.6 Panel

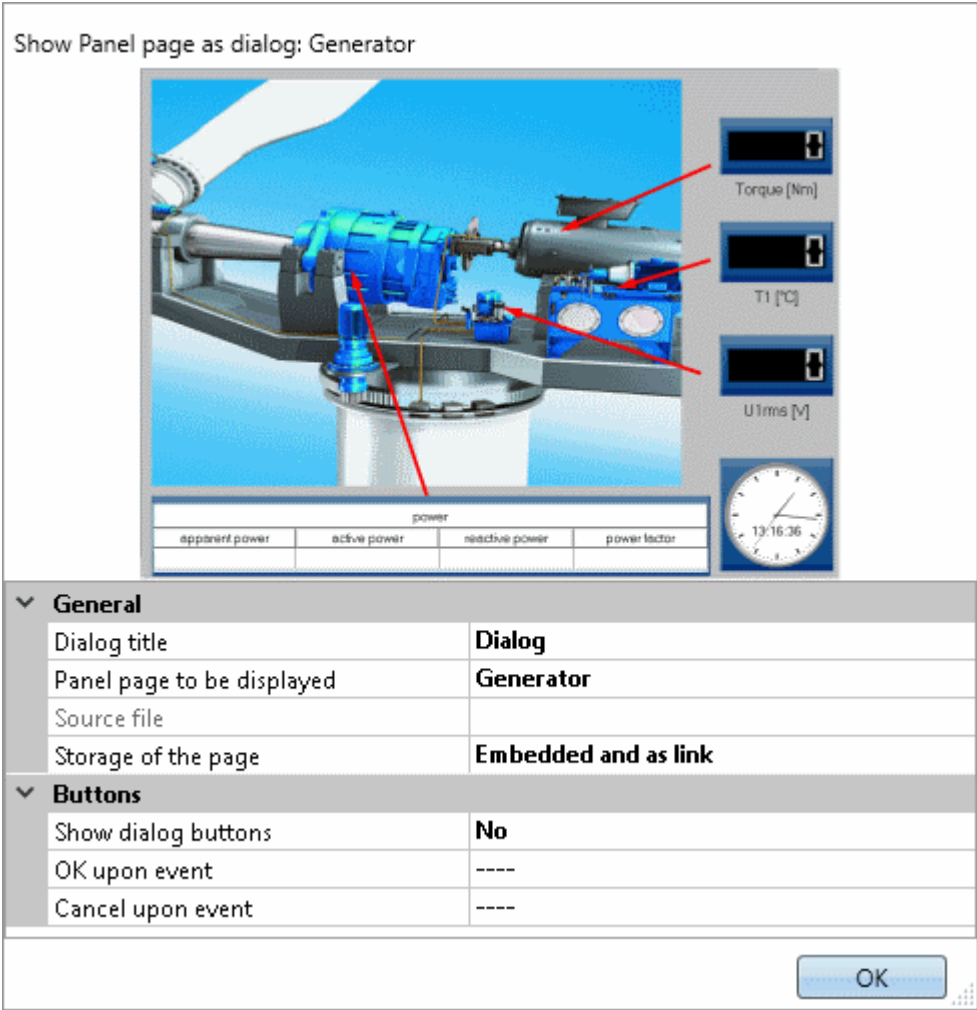
12.7.6.1 Show Panel page as a dialog

A Panel page can be displayed as a dialog.

The ways to close the dialog are:

- use existing buttons on the Panel page, or
- actuate the dialog's separate "OK" and "Cancel"-buttons.

If Setup pages are displayed on the Panel, the buttons can be hidden until all important boxes are filled.

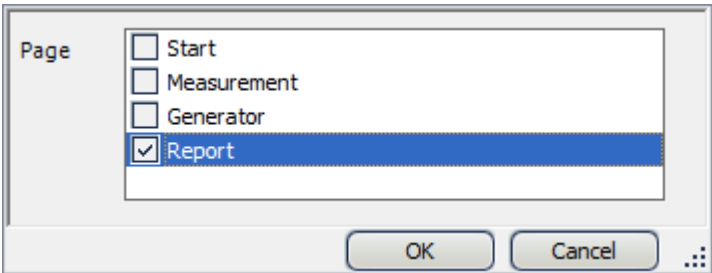


Parameter: General	Description
Dialog title	Title of the dialog to be displayed
Panel page	Select the page to be displayed. The possible selections are existing pages and exported pages in the dbv format. Using "..." it is possible to select any arbitrary *.dbv file.
Source file	If this box is empty , the Panel pages of the current experiment are used. If you have selected a *.dbv-file in the box "Panel page", its name including its path is displayed here.

Parameter: General	Description
Storage of the page	Here you can define to which changes to the Panel pages the system responds. Embedded: This page is embedded into the command. There is no connection to the page in the Panel. • If this page is deleted, the command continues to function. • Changes to the page are not applied for the command. As link: Only a reference to the existing Panel page is saved. • If the page is deleted, the command no longer functions. If it is run, an empty page is displayed. • Changes to the page are applied for the command. Embedded and as link: The page is embedded into the command. Additionally, there is a linkage to the existing Panel page. As long as a page having this name is present, it is displayed. If the page is not present, the last state is displayed. • If the page is deleted, the command continues to function. • Changes to the page are applied for the command. • If a new page having the same name is created, then it is used.
Parameter: Buttons	Description
Show dialog buttons	Separate "OK" and "Cancel"-buttons in the dialog can be shown or hidden. Yes: The buttons are displayed upon execution. The parameters for the OK- and Cancel-buttons are shown for the purpose of configuring the button. No: The buttons are not displayed. Instead, use certain buttons on the Panel page ("OK and Cancel upon event"). If neither of the two actions is defined upon event occurrence, the Close-button (X) is displayed. Then it is only possible to close the window using the "X". Closing this way is interpreted as "Cancel".
OK upon event	Select an event on the page, which closes the dialog with "OK". E.g. pushing on a switch. Some switches have two events, while others have only one. For this purpose, the switch must be present on the page.
Cancel upon event	Analogous to "OK upon Event". In this case, the dialog is closed with "Cancel".
OK button	This field is only enabled if "Show dialog buttons" is set to "Yes".
Visible	Shows/hides the button.
Active	The button can be deactivated in particular situations. Always: The button can be actuated at all times. If all inputs are valid: The button can only be actuated if all inputs are valid. For this purpose, embedded Setup pages are required. For details on this topic, see the chapter: "Setup-Layout" > Using Mandatory Boxes  169".
Label	The button's caption.
Cancel button	This field is only available if "Show dialog buttons" is set to "Yes". For the Cancel-button, the same parameters are available as for the OK-button (Visible, Active, Label).

12.7.6.2 Print Panel page

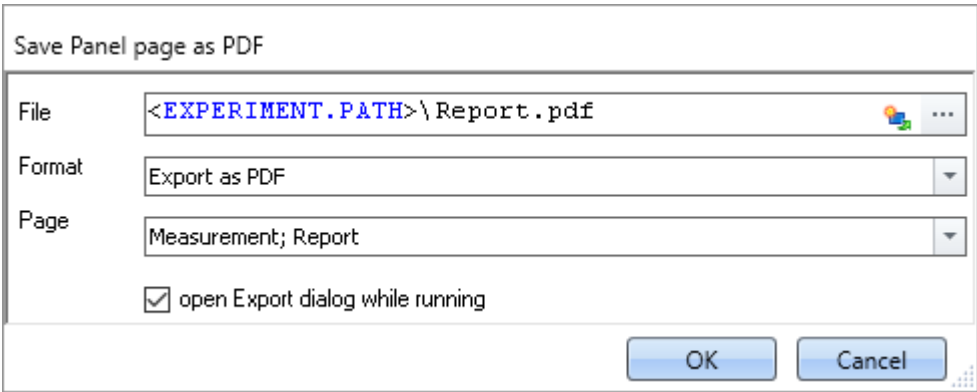
You can print out Panel pages. By this means, it is easy to compose reports. To do this, create Panel pages which are displayed as a report.



Example for printing out a Panel page

12.7.6.3 Export Panel page

Export Panel pages to a specified folder. As a graphic, PDF or in the form of Panel export.



Example of exporting Panel page

Parameter	Description								
File	Specification of target file including the path.								
Format	Target format for export. Please always select the appropriate format! Possible options: <table><tr><th>Format</th><th>Description</th></tr><tr><td>Export Panel page</td><td>Exports the page in dbv-format. The file can be imported back into the Panel at another position as a page. For each page, a separate page is generated.</td></tr><tr><td>Export as PDF</td><td>Exports the pages in PDF. All pages are exported to one file.</td></tr><tr><td>Export as graphic</td><td>Exports the page as a graphics file in png-format. For each page, a separate page is generated.</td></tr></table>	Format	Description	Export Panel page	Exports the page in dbv-format. The file can be imported back into the Panel at another position as a page. For each page, a separate page is generated.	Export as PDF	Exports the pages in PDF. All pages are exported to one file.	Export as graphic	Exports the page as a graphics file in png-format. For each page, a separate page is generated.
Format	Description								
Export Panel page	Exports the page in dbv-format. The file can be imported back into the Panel at another position as a page. For each page, a separate page is generated.								
Export as PDF	Exports the pages in PDF. All pages are exported to one file.								
Export as graphic	Exports the page as a graphics file in png-format. For each page, a separate page is generated.								
Page	Selection of the pages to be exported. Depending on the parameter "File" and "Format", either one or more pages can be selected.								
Open Export dialog while running	If this option is activated, then when the command is run the "Save as" dialog appears. The change has no effect on the configuration of the command in the commands list, but only affects the current export.								

**Note****Name of the results file depending on the setting**

For varying formats, there are different respective results unless there are clear specifications available for the parameter "File".

Panel page:

Parameter	Description
C:\tmp\Report.dbv	Generates the file: Report.dbv in the path C:\tmp Only possible when an individual page is selected.
C:\tmp\Report	Generates a separate dbv-file in the path C:\tmp\Report for each selected page
C:\tmp\Report\	The files are named for their respective page.

PDF:

Parameter	Description
C:\tmp\Report.pdf	Generates the file: Report.pdf in the path C:\tmp.
C:\tmp\Report	
C:\tmp\Report\	Generates the file .pdf in the path C:\tmp\Report (<-- <i>not recommended!</i>)

Graphic:

Parameter	Description
C:\tmp\Report.png	Generates the file Report.png in the path C:\tmp Only possible when an individual page is selected.
C:\tmp\Report	Generates a separate png-file for each selected page in the path C:\tmp\Report
C:\tmp\Report\	The files are named for their respective page.

12.7.6.4 Import Panel page

You can import a Panel page which was previously exported in DBV format.

12.7.6.5 Delete page

If the command is executed, a defined Panel page is deleted, without waiting for confirmation.

By means of the command's settings, you determine which Panel page is to be deleted. To do this, select an existing page, or enter the page name, if the page does not yet exist.

It is possible to use the list to select multiple pages. To select multiple pages by making an input, use the following syntax:

```
Site 1; Site 2; Site 3.
```

12.7.7 Parameter set

A parameter set is a group of parameters which can be used for various purposes. For instance, parameters may be **settings for the device and channel configuration**. They can also be **metadata** and **variables values**.

The command: "*Export parameters*" allows automated export of the parameters at desired times and according to previously adjusted settings.

The command: "*Import parameters*" allows automated import of the parameters at desired times and according to previously adjusted settings.

Reference

The description is provided in the section: "*Channel types, variable types, data types and data storage formats*" > "[Parameter set](#)"

- "[Export parameters](#)"
- "[Import parameters](#)"

Note

Import/Export of variables preferably via the variable commands

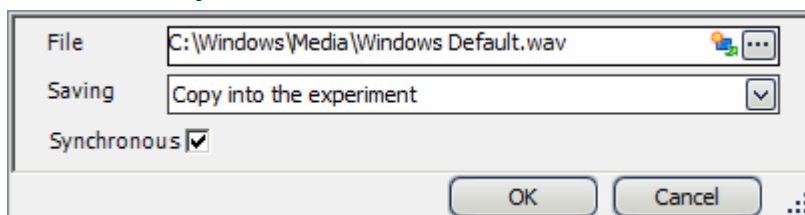
For **importing/exporting variables' values**, the command "[Load/Refill variable](#)" or "[Export variable](#)" should be used. These variable commands are specially designed for variable values.

Using the command: "*Import parameter*" can cause unintended alteration of the device control.

Certain variable types can only be exported by means of the command "*Export variables*"; this included text-variables.

12.7.8 Read/Write data

12.7.8.1 Play soundfile



Properties of Play soundfile

Parameter	Description
File	Select an audio file in WAV format.
Saving	Sets whether the audio file is permanently saved in the experiment file (<i>As copy in the experiment</i>) or only the path to the file is saved (<i>Only as link</i>).
Synchronous	When activated, the Sequencer remains on this command until audio output was completed.

12.7.8.2 Data saving assistant

If data saving to the PC was selected for at least one active channel, then in the default case, the measurement results are saved in the database. With the help of the **Data Saving Assistant** command, one can continue to specifically process these measurements. This command must be started either through a Panel event (Widget) or by means of a sequencer event.

The Data Saving Assistant lists all new measurement folders which were created since the start of the last measurement. The number of folders depends on for instance the interval data saving or the interrupt/resume of data storage. Thus, these measurements, for example, can then be exported with or without deleting the original files.

Data saving assistant

☒ Documentation

☐ Experiment

☐ Annotation after measurement

☐ Annotation before measurement

☐ Use imc Format Converter

Format converter

☒ Allow data export

☐ Delete original files

☐ Export measurement settings

☐ Allow selection of the export folder

Default path D:\Data

☒ Allow discarding of all measurements

☒ Show prompt for confirming discard

Default button

Save

☐ Execute without acknowledge

OK

Data Saving Assistant

Setup pages

It is possible to save **measurement comments** (metadata) which pertain to the measurement. In the list at the top, the Setup pages are listed. When you select one of the pages, this page's parameter set as saved with the measured data as a .csv file.

If the assistant is running, a new window appears. For each page selected, a tab is added at the top. There, you can fill the fields.

imc Format Converter

In addition to the saved or exported measurement results, you can transfer the results to other formats. In the options for the imc Format Converter, you can additionally specify the format, e.g. ASCII or EXCEL.

Data export

This settings allows you to export saved measurements.

Allow data export

When this option is active, the measurement results can be exported to a different location. When the command is executed, an additional button appears, by means of which the export is initiated. I.e. in that case the buttons *Export* and *Save* become available.

- If you click on *Save*, the measurement result only goes to the database.
- If you click on *Export*, the data are exported according to the settings for the other options.

Delete original files

The original files in the database can be deleted upon exporting. This means that the database only contains the explicitly saved measurements which were not exported. The exported measurements are only located in the export path.

Example: Supposing you don't need the measurement results to be in the database. If the measurements were performed correctly, the results are exported and the original files are deleted right away. In all other cases, you can delete the measurements right away by means of the button "[Discard measurement](#)"  (described further below).

Export measurement settings

No function

Allow selection of the export folder

- If a default path is specified, you can allow/forbid changes by the user.
- If this option is activated, a folder selection dialog always appears. If a default path is specified, it is the suggested target folder in the dialog.
- If this option is deactivated, export is always to the specified default path.

Default path

Here you set the target folder for export.

Discard measurement

You have the choice of discarding measurement results.

Allow discarding of all measurements

If this option is activated, it is possible to delete the measured results. When the command is executed, an additional button by means of which the results can be deleted appears. I.e., the buttons *Discard measurement* and *Save measurement* become available. If you click on *Save*, the measurement results remain in the database. If you click on *Discard measurement*, the measurement results are deleted.

Prompt to confirm discard

Here you set whether to post a confirmation prompt when *Discard measurement* is clicked.

Default button

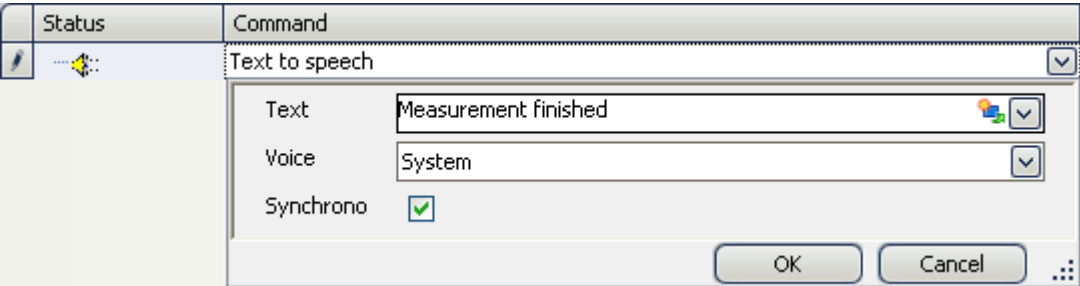
Here you set which button is "pre-selected". If the commando is executed, the selected action can be run by clicking on the Enter-key.

Execute without acknowledge

If the box "Execute without acknowledge" is checkmarked, the selected default action is executed automatically, without a delay for the purpose of confirmation. In this case, it is not possible to change the action if the command is executed.

You can only checkmark this box if the other settings do not require any acknowledgement. For instance, there would have to be a pre-set path for exporting, and no confirmation prompt would be required for deleting.

12.7.8.3 Text to speech



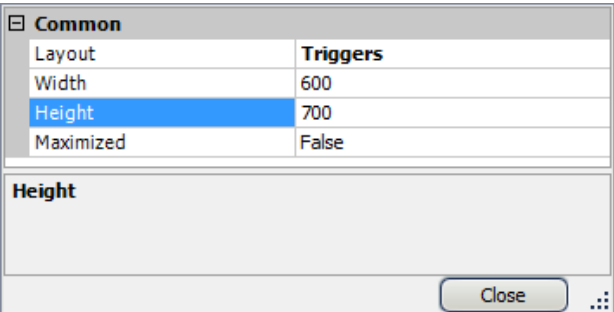
Properties of Text to speech

Parameter	Description
Text	Enter here the text to be spoken, e.g. "Measurement finished".
Saving	Character of the voice output. The available selection depends on the Windows system and usually only offers English voices.
Synchronous	When activated, the Sequencer remains on this command until audio output was completed.

12.7.9 Setup

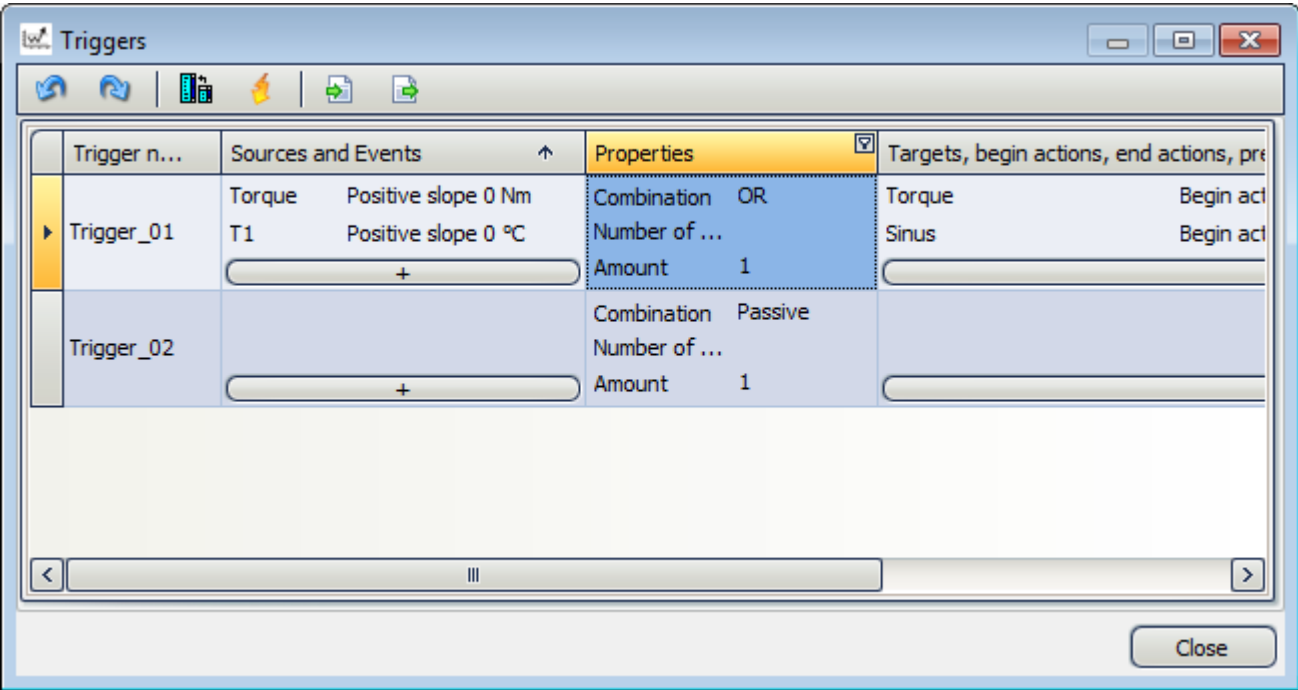
12.7.9.1 Display Setup dialog

The command group *Setup* offers all setup settings in dialog form.



Setup properties

Select *Layout* and set the *Width* and *Height*. Alternatively, the dialog's display can be *maximized*.



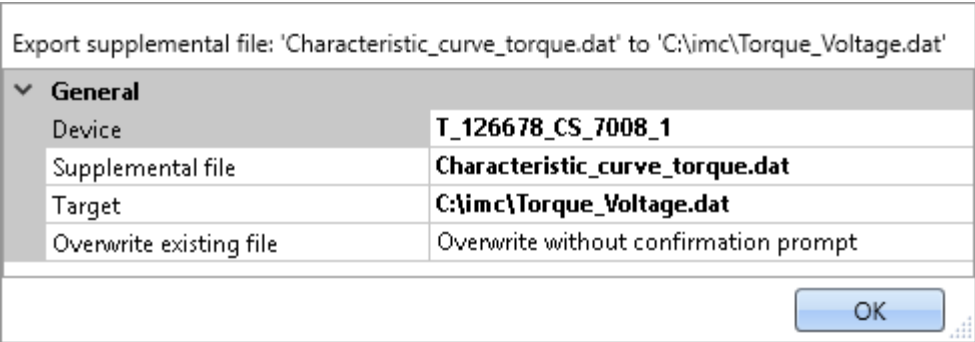
Example Trigger dialog, started by means of the command Setup

12.7.9.2 Export supplemental file

This command enables export of a supplemental file. The files can, for example, be exported from the device.

 Reference	Description of the supplemental files

Supplemental files exist in various file types. For an exact description of the supplemental files, see the section: "Setup - Device Configuration" > Ribbon: "[Supplemental Files](#)".



Configuration: Export supplemental file

Parameter	Description
Device	Here, select the source where the file is located. The source may be a device. It is always only possible to make one selection.
Supplemental file	Specifies the file to be exported.
Target	Target file including the path.

Parameter	Description
Overwrite existing file	Here, define the system's behavior for handling the files if the target file already exists. • Overwrite without confirmation prompt: Files in the target location which have the same name are overwritten without any confirmation prompt. • Do not overwrite: Files in the target location which have the same name are retained. The file is not exported and no notification is posted. • Prompt for action: A dialog concerning the subsequent proceedings is displayed. Available options are: "Overwrite", "Cancel". Each of these actions concludes the command with "Finished". This option is relevant for automated routines.

12.7.9.3 Import supplemental file

This command enables import of one supplemental file. The files can be used by imc Online FAMOS, for example.



Reference

Description of the supplemental files

Supplemental files exist in various file types. For an exact description of the supplemental files, see the section: "Setup - Device Configuration" > Ribbon: "[Supplemental Files](#)".

Import supplemental file: Torque_Voltage.dat

General	
Import file	c:\imc\Torque_Voltage.dat
Target name	Characteristic_curve_torque
Overwrite existing file	Overwrite without confirmation prompt
Mapping table	

OK

Example of a characteristic curve used in imc Online FAMOS with the Charact() function

Parameter	Description
Import file	Specification of the file to import, including the pathname.
Target name	Optional entry. Here, specify a deviating name if the filename in the device is not supposed to match the original filename.
Overwrite existing file	Here, define the system's behavior for handling the files if the target file already exists. • Overwrite without confirmation prompt: Files in the target location which have the same name are overwritten without any confirmation prompt. • Do not overwrite: Files in the target location which have the same name are retained. The file is not imported and no notification is posted. • Prompt for action: A dialog concerning the subsequent proceedings is displayed. Available options are: "Overwrite", "Rename", "Cancel". Each of these actions concludes the command with "Finished". This option is relevant for automated routines.
Mapping table	Here, select the import target. The target may be a device . it is always only possible to make one selection.

12.7.9.4 Delete supplemental file

This command enables deletion of a supplemental file. The files can be deleted from the device, for example.

 **Reference**

Description of the supplemental files

Supplemental files exist in various file types. For an exact description of the supplemental files, see the section: "Setup - Device Configuration" > Ribbon: "[Supplemental Files](#)¹²⁰".

Delete supplemental file: Characteristic_curve_torque.dat

General

Filename	Characteristic_curve_torque.dat
Device selection	 T_126678_CS_7008_1
Delete	Without confirmation prompt

OK

Configuration: Delete supplemental file

Parameter	Description
Filename	Specifies the file to be deleted.
Device selection	Here, select the source where the file is located. The source can be a device. It is always only possible to make one selection.
Delete	Define here whether a confirmation prompt is to appear. Without confirmation prompt: The supplemental file is deleted without any confirmation. With confirmation prompt: A dialog concerning the subsequent proceedings is displayed. Available options are: "Delete", "Don't delete". Each of these actions concludes the command with "Finished". This option is relevant for automated routines.

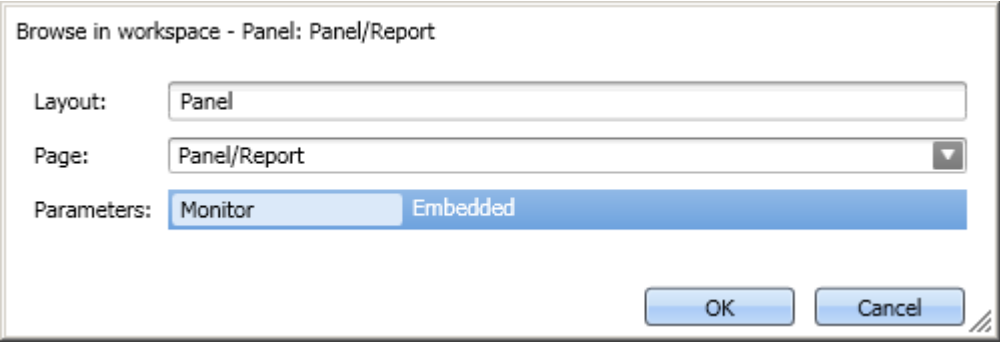
12.7.10 User interaction

12.7.10.1 Browse in workspace

If the command **Browse in workspace** is executed, imc STUDIO jumps to the selected page. Every main window can be selected as the target. With main windows which have their own pages, it is also possible to jump directly to a special page (e.g. Panel or Setup).

If a Panel page is defined as the target, the page can also be displayed on a monitor of the user's choice in fullscreen mode. In this case, imc STUDIO does not change the main window. Thus, for instance, the Sequencer can be viewed on the main monitor, while on the second monitor, the measured data can be viewed on a Panel page in fullscreen mode.

Using this command, it is always possible to open the appropriate window via the Sequencer, or to jump to, for example, the balancing dialog, by means of a button on a Panel page.



Selection of a specific page

Parameter	Description
Layout	Display of the main page's name. Adjusts itself according to the target page. Can not be edited.
Page	Here, you define the target page.
Parameters	If target page: Panel • View: Panel Embedded Mode: Jumps to the Panel. If necessary, exits fullscreen view. • View: Panel Fullscreen Mode: Jumps to the Panel in fullscreen mode (see Panel menu ribbon: <i>Control</i> > Panel Fullscreen^[586]). If target page: Panel page • Monitor: Embedded: Jumps to the Panel and opens the selected page • Monitor: <Monitor name>: The page is displayed in fullscreen mode on the monitor selected. Independent of the opened main window (see Panel Context menu: The Panel page's tab: Show Page on Monitor^[613]).

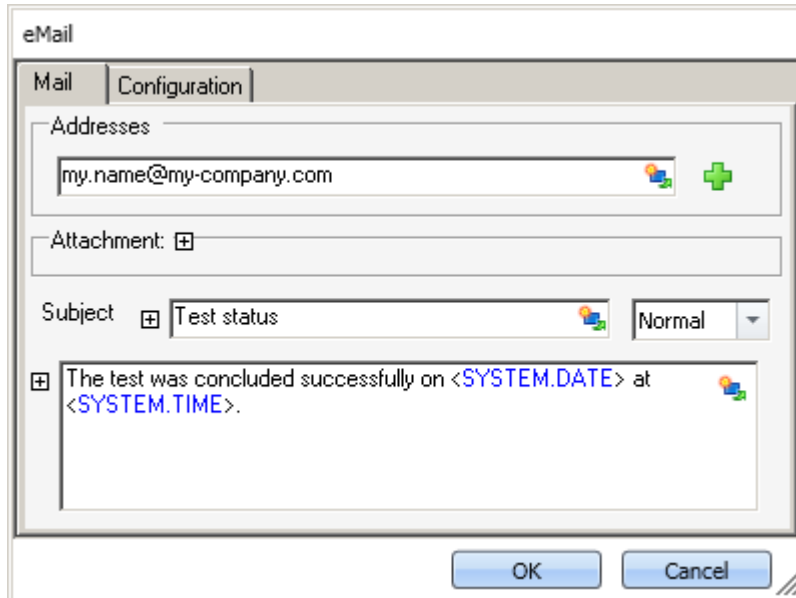
12.7.10.2 E-mail

The command **E-mail** enables automated sending of an e-mail in important situations. For this purpose, no external e-mail program is required.

The E-Mail-provider requires the login data of the sender's address.

Mail

The typical dialog boxes for an e-mail program are found on the page *Mail*.



Defining an E-Mail

By right-clicking, or by clicking on the placeholder symbol (), you can apply a variety of placeholders.

Under *Addresses*, you can define one or more recipients. Additionally, an *Attachment* can be added and the *Priority* of your message specified.

The box *Subject* and the text to be sent can be defined in various languages.

Configuration

On the page *Configuration* you set the access data for the E-Mail provider.

eMail

Mail

Configuration

Sender address

test.bench@my-company.com

Sender name

imc STUDIO Test bench

User

Test bench

Password

xxxxxxxx

Mail server (SMTP)

smtp.my-company.com

Port

25

Use presetsings

☐ [Configure presetsings](#)

Secure connection

☐

Postfix

This E-mail has been sent automatically
by imc STUDIO.



OK

Cancel

Defining the E-Mail configuration

Parameter	Description
Sender address	Sender address (your e-mail address).
Sender name	The name of the sender of the e-mail (your name).
User	User name for logging in.
Password	Password for authentication.
Mail server (SMTP)	Outgoing mail server of the E-Mail provider used.
Port	SMTP server port used.
Use presetsings	In the imc STUDIO option, it is possible to define a presetting for the E-Mail command (Ribbon <i>Extra</i> > <i>Options</i>). If "Use presetsings" is activated, these settings are applied. This means that the setting can be defined a single time in the Options and does not need to be entered again for each E-Mail command.
Encoded connection	Encryption protocol for secure data transmission via Internet.
Postfix	This postfix will be appended to every e-mail.

 FAQ

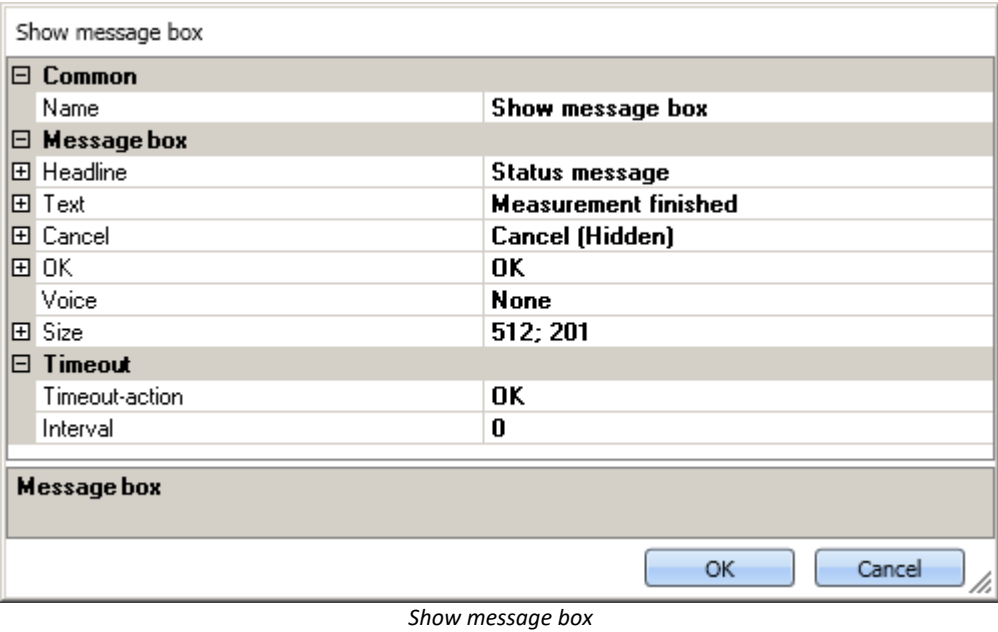
Why do e-mails not get sent, even though the settings are correct?

Answer: For reasons of security, some firewall programs prohibit the sending of e-mails when the firewall is not familiar with the program. Please check your firewall and configure it if necessary. **The process "imc.studio.exe" must be permitted to send e-mails if imc STUDIO is intended to send mails.**

12.7.10.3 Show message box

When the command *Show message box* was executed, a user-defined dialog was displayed. This can be used to display information, or to ask the user what to do next.

Along with a header and the information text, the dialog can also be displayed with a voice output.



Parameter	Description
Name	Display text in the sequence table (without effect on the dialog).
Headline	The dialog's header.
Text	Text displayed in the dialog. With "\r\n", it is possible to force a line break.
Cancel OK	Settings for the buttons: "OK" and "Cancel" Text: Button caption Visible: The button can be either shown or hidden. • Show: The button is shown. In this way, the dialog can be closed with the corresponding dialog response. • Hide: The button is hidden In this way, the dialog can not be closed with the corresponding dialog response.
Voice	Character of the voice output. The available selection depends on the Windows system and usually only offers English voices.
Size	Size of the dialog.
Timeout-action	The selected action is performed upon elapse of the timeout time. The dialog is closed with "OK"/"Cancel".
Interval	Duration in seconds [s] until the timeout occurs. "0" switches the timeout off.

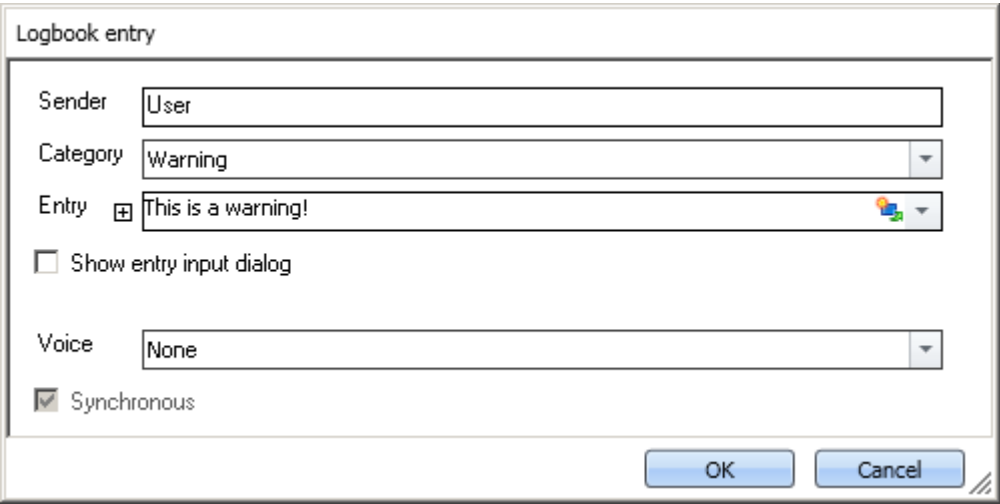
12.7.10.4 Open logbook-viewer

If the command is executed, the Logbook-Viewer is opened. How to operate the Logbook-Viewer is described in context of the tool window: [Logbook](#)^[71].

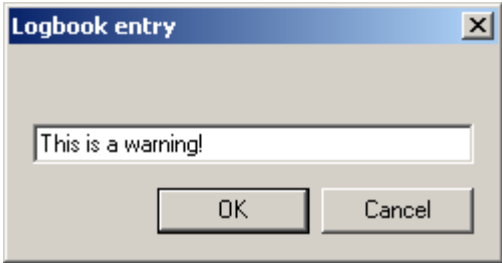
The Logbook-Viewer can be started with defined filter settings. For example, messages of the category: [Information](#)^[72] and all [Duplicates](#)^[73] can be hidden.

12.7.10.5 Logbook entry

Using this command, you can create an entry in the logbook and stop the Sequencer if appropriate.



Properties of the log book entry

Parameter	Description
Sender	Entry of information on the message's origin.
Category	There are four different categories: • Information: Informative messages • Warning: Warning text entry in the log book, otherwise no other effects. • Error: Error message, which interrupts the running Sequencer if appropriate. • Fatal: Fatal error message, which interrupts the running Sequencer if appropriate.
Entry	Text entered as message in the log book.
Shows entry input dialog	Enables a text input during runtime. 
Voice	The message text can be spoken by the computer voice.
Synchronous	When activated, the Sequencer remains on this command until audio output (Voice) was completed.

Output in the Logbook:

The output is logged to the logbook.

	Time	Code	Message	Sender
!	13.02.2015 16:42:48	0	Sequencer ...done	imc STUDIO
✖	13.02.2015 16:42:48	0	This is a fatal error!	User
!	13.02.2015 16:42:48	0	This is an error!	User
!	13.02.2015 16:42:47	0	This is a warning!	User
!	13.02.2015 16:42:47	0	This is some information	User
!	13.02.2015 16:42:47	0	Sequencer ...starting	imc STUDIO

Logbook with generated entries

12.7.10.6 Execute menu action

This command lets you execute a menu action. A list contains the actions available in the menu ribbon. When this command is run, it replicates clicking the mouse over the corresponding button.

This command is often linked with switches on the Panel. In particular when the Panel is run in fullscreen. The menu ribbon is not present then.



Reference

Widget: "Execute menu action"

The action can be directly linked with the synonymous Widget: "[Execute menu action](#)". This Widget adapts itself to the status of the menu action. E.g. when the action is currently not present, the button can not be clicked.



Note

When is the command "Done"

The sequence does not wait for the function behind the action to be concluded. For the sequence, the action is "done" when the action is acknowledged.

Examples:

- Menu action: Connect
The command does not wait until the connection can be established.
The command does not care whether the connection even can be established.
- Menu action: Start Measurement
The command does not wait until the measurement has been started.
The command does not care whether the measurement even can be started.

Once the action has been initiated, the command is "done". Even if error messages are posted afterward.

The menu action can not be executed if it is not available or not enabled. Some menu actions are attached to specific user roles, device components or loaded imc STUDIO-components. When such actions are initiated, a corresponding error message is posted.

Execute menu action: Start (Start measurement for all devices.)

Menu action

Start

Error in execution

 Warning

Menu action description Start measurement for all devices.

Additional information

Menu action name acDeviceStart

OK

Cancel

Configuration: Execute menu action

Parameter	Description
Menu action	Here, select the desired menu action. Open the drop-down list. The menu actions appear in groups. Scroll to the appropriate group and select the desired action. Filtering: You can enter the first letter of the action's name in the input box. Then the list will be restricted to actions starting with that letter.
Error in execution	If the action is not enabled or not available, the command may respond in various ways. • Error: The command returns an error message in the logbook. If "Stop on error" is activated, the sequence is stopped. • Warning: The command returns a warning message in the logbook. The sequence continues to run. • Ignore: The command is concluded with "Done". No info appears in the logbook.
Menu action description	The action's description text is displayed.
Additional information	The menu action's internal designator is displayed.

12.7.11 Variables

12.7.11.1 Export variable

Description of the command "*Export variables*" and of the corresponding [function accessed via the Data Browser](#) .

The function **enables variables** and their respective values **to be saved** at any desired location. It is possible to select one or more variables to be exported.

The following applies to the command: All options can be either specified, or selected manually while the Sequencer is running.

Parameter - File options	Description
Folder	Specifies the target folder. If no target folder is specified and the option: " <i>Show Dialog</i> " (only applicable to command) is not activated, a folder selection dialog appears when the function is executed.
Save to one file per file format (Save all in a single file)	If this option is activated, all variables are saved to a single (multi-channel) file. The filename can be specified. The file format for the file does not need to be specified. However, if no filename is specified, all variables are automatically saved to the file <i>data.dat</i>. Furthermore, for each file format a separate file is created. If the option is not activated, each variable is saved in a separate file. In that case, the filename can be specified separately for each variable.
Parameter - Mapping instruction	Description
Mapping instruction	Here you can select pre-defined export settings. • No selection: The user-defined settings are used (e.g. selection of variables) • Default Export Mapping: It is not possible to select variables. Instead, all variables (but not single-value variables) are exported. Thus, the same functionality is implemented as for the menu action: "Save current Data". Additionally, it is possible to select a measurement whose variables are to be exported.
Measurement	Here you specify which measurement's variables to export. The option can only be used in conjunction with a mapping instruction. • No selection: Variables from the current measurement are exported. • Selection or entry possibilities: Measurement name or Symbolic measurement name (with measurement number)
Parameter - Variables	Description
Add variables	Here, the variables to be exported are selected and entered in the Variables list. • If the variables exist in the Data Browser, they can be selected by means of the button . • If a variable does not yet exist, it is still possible to add it. Enter its future name and confirm by clicking on Enter. Selected and entered variables appear in the Variables List.

Parameter - Variables list	Description
By means of the checkbox, it is possible to de-select for export a variable already entered. This means that it still exists in the list, but is not exported. It can later be activated for export again at any time.	
Variable name	Name of the variable as it is shown in the Data Browser
Measurement name	Measurement from which the variable is to be exported • No selection: The variable is selected from the current measurement • Selection or entry possibilities: Measurement name or Symbolic measurement name (with measurement number)
File format	Target format of the file to export. The system selects the file format automatically on the basis of the variable type. Alternatively, you can specify the file format in the settings under Options . imc Format Converter provides a variety of export format templates. Storage location for the templates: "<i>C:\ProgramData\imc\Common\Def</i>" Note: With the help of imc FAMOS, you can create your own export format templates (*.aet) and use them in imc STUDIO Monitor. The use of one's own templates is also possible on other PCs. No imc FAMOS installation is required. For information on this topic, see the imc FAMOS user's manual.
Filename	(not for: "Save all in a single file") By default, the file name is formed from the variable's name. You can change this here.
Parameter - Other options	Description
Treat error as warning	If the function is executed even though not all selected variables exist, error messages are posted. In normal cases, the function and the Sequencer is exited when an error message appears. This also causes not all variables to be exported. The variables in the list are exported in succession and all variables below the non-existent variable are thus not exported. In order that always exports all existing variables, it is possible to activate this option. In this case, no error messages appear in the logbook, but warnings instead. Warnings do not cause the function or the Sequencer to close.
Show file options and Show variables options (Only applicable to command)	If one of the two options is activated, then when the command is executed, the settings dialog opens for the respective scope: • Show file options: File options • Show variable options: Mapping instruction, Variables and Variables list. All settings can be checked and edited. Any changes do not take effect on the configuration of the command in the command list, but only on the export currently performed.
Always overwrite existing files	If this option is activated, files having the same name in the destination folder will be overwritten without a confirmation prompt. This option is useful for automated routines.
File comment	A file comment can be added. In imc FAMOS, the file comment is called as shown below: <pre>path = FileName?(Channel_001) id = FileOpenDSF(path, 0) comment = FileComm?(id) FileClose(id)</pre> Supported: • Export in RAW/DAT-format, as NO Key • Export to *.aet files for the placeholder %FILECOMMENT%

12.7.11.2 Load/Refill variable

Load Variable: The command enables **creation or replace** of user-defined **variables with pre-defined values** from saved files. The user-defined variable inherits all of the properties of the variable from the saved file.

(Possible targets: User-defined variables)

- If the variable doesn't exist yet, it is created.
- If the variable exists already, the existing variable is replaced with the new one.

Refill variable: The command enables **setting the values of an existing variable** from saved files. The variable must already exist before the command is run. Only the values are adopted. (Possible targets: among others device or user defined variables)

The values must be saved in a suitable file, for example one which was created using the command: "[Export Variable](#)". It is possible to select one or more files. If you wish to use multiple files, they must all be located in one folder.

All options can be either specified, or selected manually while the Sequencer is running.



Note

Background information

A function "**Load variable**" is now implemented, which resembles "Load Data" in imc FAMOS. Overwriting existing variables is allowed. A confirmation prompt appears for each conflict of variables. If any of the confirmation prompts is canceled, the entire import automatically cancels. In this way, the system prevents the loading of an incomplete set of controller parameters, for example.

When an existing variable is overwritten in the process of loading, then it is effectively completely replaced, including all of its properties. In consequence, the old variable is neither deleted nor is any new one created, so that no "events" are triggered by the deletion or creation. This is important for scripts, curve windows, etc.

In the process, the user-defined variables' scope is also overwritten. But when a variable for which no scope is defined is imported, the variable becomes "*temporary*".

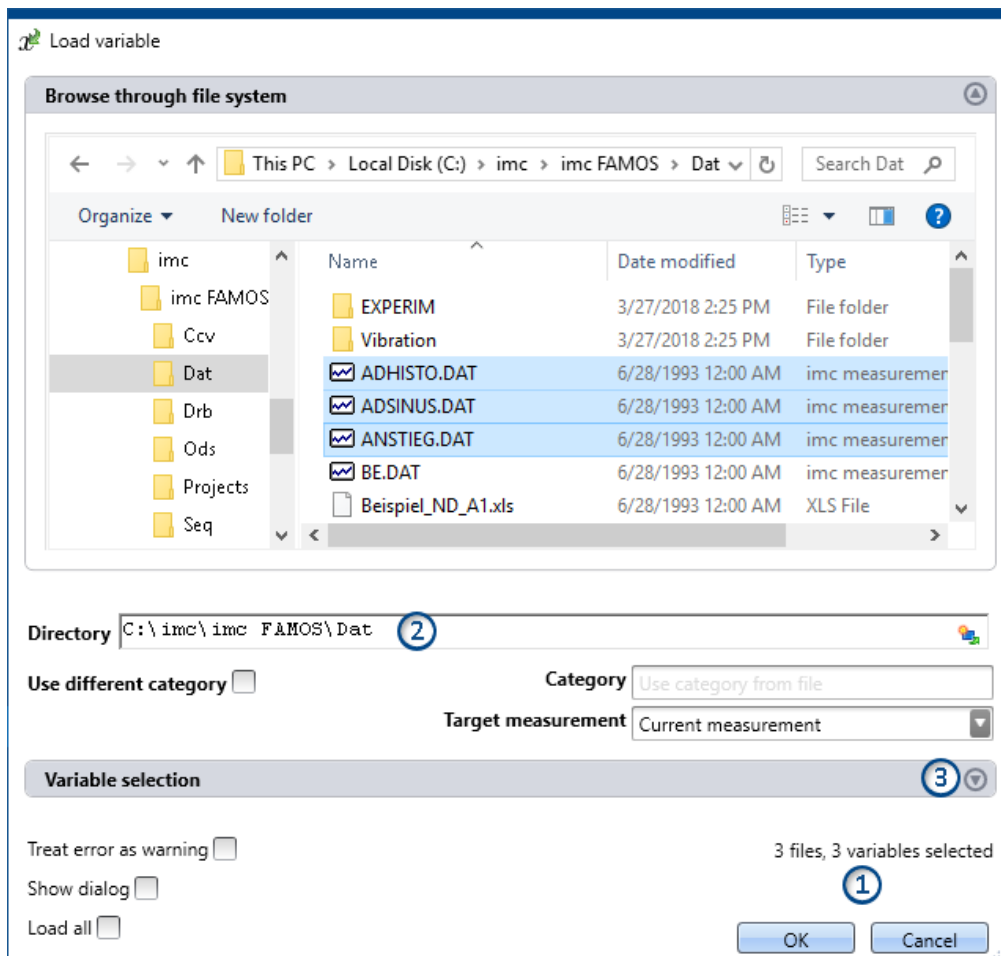
In contrast to imc FAMOS, there are variables which cannot be overwritten, e.g. channels or pv-variables. The reason is that, for instance, these variables belong to the specific device, or possess specific data formats and other properties which must be retained. If the user attempts to overwrite them, an error message is posted indicating that they can't be overwritten.

With the "**Load variable**" command, there is an option for determining whether existing variables can be overwritten with or without a confirmation prompt. By default, this option is activated, meaning that there is no confirmation prompt for overwriting.

By using "**Refill variable**", it is possible to change the content of existing variables. In this way it is possible, for instance, to change the content of pv-variables or user-defined variables. Since this only changes the variables' content and not their type, the target variable and the variable to be loaded must have the same properties, including, for example, the unit. Thus, only values and the sample count may be different. Therefore, please always use "**Load variable**" in cases where changes are possible.

Brief description

In the upper region, there is an Explorer for selecting files:



Here you can use the "Explorer"-functions to **navigate to the files and select them**. After making a selection, you can close the dialog by clicking "OK". **All data selected are loaded** (or, when using the command, selected for loading).

Exception: If you don't select any file in the Explorer, all importable files belonging to the current folder are automatically selected.

Note

Adding variables subsequently

Please be aware that the **variables are no longer in the selected state** in the Explorer once the command has been concluded by clicking "OK". If you open the configuration again and **select an additional file**, then **only this file is selected**; all others are de-selected.

Add more files by means of either the <CTRL>- or <SHIFT>-key. In this case, the files are added to the list.

Remove the variables via the "Variable selection" area.

File count (1): At the bottom right there is a readout of how many files are selected and how many variables belonging to these files are selected.

Directory (2): Below the file selection window, there is an input box for the folder path, which can be used as an alternative for specifying the file(s) desired. Here, you can enter placeholders () or paths which only exist once the command is run.

 **Note**

Utilizing a placeholder

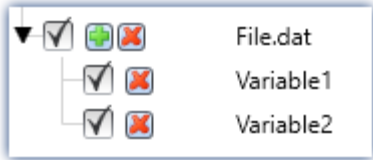
The placeholder can be used at all positions (start, middle, end). E.g. "<EXPERIMENT . PATH>\Data\"

When possible, the path is resolved in the "Explorer". There, you can also select files.

However, if you open a subfolder or a different folder in the Explorer, the field "Directory" is filled with the current value. The placeholder is no longer present.

Variables-box (Variable selection) (3): Using the buttons (📁) near the right edge, you can, for example, expand the Variables area. You can also make additional settings such as

- **de-/selecting** specific files or **variables** using the checkbox (☐), for cases where not all variables belonging to a file are to be loaded. This means that the are present in the list but will not be loaded. They can be activated again at any time.
- **adding more elements**, for the case where the variable only exists once at the command's runtime. Use the "+"-symbol (➕) to add files and variables. Be aware that a file must always contain at least one variable.



Example: a file having two variables

Variables list	Description
Variable name	Name of the file/variable as found in the file.
Target variable name	Name of the variable as it is to be found in the Data Browser following load.

The element-specific configuration is discarded once the file/variable is deleted from the list.

Other options

Category	Description
(Only for the command: Load Variable)	
Use different category	Put a checkmark in the box if you wish to create the variables in a deviating category. An input box for the new category appears. This category is applicable for all variables loaded. • deactivated: The variable is displayed in the variable's category, or without any category under "no category". • activated: The variable is displayed in the specified category or without any specification under "User-defined variable". If you wish to create a category within the category "User-defined variables", do the following: write a "\" before the category name.  Category: "MeasurementPoint_1". A category called "MeasurementPoint_1" is created in parallel with the other categories, such as "Analog Inputs". It contains the variable. Category "\"MeasurementPoint_1". This creates a category "MeasurementPoint_1" within the category "User-defined variables". It contains the variable. This applies, for example, to variables created by means of the command: "Load variable" or to generating a user-defined variable⁵³⁶ by means of the Data Browser.

Target measurement	Description
Current measurement	Load: The variable appears in the Data Browser under " <i>Current measurement</i> ". Refill: The variable overwrites the value of any variable existing under " <i>Current measurement</i> ".
Last completed measurement	Load: The variable appears in the measurement which is in the Data Browser and is saved in that measurement's folder. The variable remains always available once the measurement has been loaded.
Measurement number Measurement#<No>	
Fixed measurement name	
Other options	Description
Treat error as warning	If the command is executed even though not all selected variables exist, error messages are posted. In normal cases, the Sequencer is exited when an error message appears. This also causes no variable to be imported. In order that the command always imports all variables, it is possible to activate this option. In this case, no error messages appear in the logbook, but warnings instead. Warnings do not cause the Sequencer to close.
Show dialog	If this option is activated, the settings dialog appears when the command is executed. All settings can be checked and edited. Any changes do not take effect on the configuration of the command in the command list, but only on the import currently performed.
Always overwrite existing variables	Load: If this option is activated, variables (and files, if applicable) having the same name in the destination folder will be overwritten without a confirmation prompt. This option is useful for automated routines.
Load/Import all	When this option is activated, then when the command is run, all variables are imported in accordance with the file selection: • if files are selected, then all variables belonging to those files are imported at runtime. • if no file is selected, then all files and their variables are imported at runtime.

**Note****Changes to files are not applied**

If configuration of the command is completed, the Variable list's configuration is permanent. This means that if the file is subsequently changed, the changes will be ignored.

Example: A file contains two variables: Var_1 and Var_2.

The command is configured in such a way that the file is selected. When the command is run, both variables are imported.

Later the file is replaced. The two variables still exist in it, but now with an additional variable: Var_3. If the command is executed, only Var_1 and Var_2 are imported. The new variable is not imported.

If you wish to make changes to the files actually applicable, please activate the setting: "*Import/Load all*".

12.7.11.3 Delete variable

The command **Delete Variable** enables existing variables to be deleted. When the command is executed, all selected variables are deleted. It is possible to select one or more variables to be deleted. All options can be either specified, or selected manually while the Sequencer is running.

Variables list

To add a variable to the list, first click in the empty box and then

- enter a variable's name or
- click on the button  and select an already existing variable from the Data Browser.

 **Note**

Fault condition - Unable to delete variable

If a variable cannot be deleted during running of the command, as is the case with device-variables and system-variables, a pertinent warning is posted. The same applies when a certain variable does not exist. However, all variables which can be deleted are deleted.

Parameter - Other options	Description
Show dialog	If this option is activated, the settings dialog appears when the command is executed. All settings can be checked and edited. Any changes do not take effect on the configuration of the command in the command list, but only on the deletion currently performed.

12.7.11.4 Set variables

The command **Set variables** assigns values to variables.

Set variables

Please select variables and define the values!

Name	Value	Reset
MyVariable	5	☐
MyText	My project 123	☐
pv.Controller	0.5	☐
DisplayVar_01	2	☐
		☐

OK

Cancel

Command: Set variables

It is possible to set pv-variables, Display variables, virtual bits, or user-defined variables. You can also set triggers to 0 or 1, as well as assign data to whole channels.

To select an **existing** variable, click on , the Data Browser opens. All variables used in this command must be present at the time the command is run.

In the column "Value", you can use placeholders; see the chapter for more [Placeholders](#)⁹⁵ information.

**Note****Automatic resolution**

As described below, this command automatically resolves a number of expressions. To prevent this automatic resolution, place the assigned value in quotation marks.

It is recommended to write texts in the box *Value* in quotation marks, as a rule, in order to prevent unintended resolution.

The quotation marks are automatically deleted before the assignment, so that they do not appear during subsequent processing and display of the variables.

If you wish to include quotation marks in a variable, use \".

Resolving existing variables

If the name of an existing variable is used in the "*Value*" box, then at runtime it is replaced by its current value. In order to prevent a new name being replaced by a value, place the text in quotation marks. If you wish to resolve a targeted variable within a text which is in quotation marks, please use the following placeholder: `<VARS["My_Variable"].VALUE>`.

Mathematical operators

In the "*Value*" box, it is possible to use mathematical operators such as +, -, / and *. Thus, you could set the variable `a`, for instance, to the value `a+1`.

Logical operators

In the "*Value*" box, it is possible to use logical operators such as && and ||.

Comparisons

It is possible to perform comparisons in the "*Value*" box such as <, >, <=, >=, ==, !=. In order for the comparison operators to be recognized as comparison operators, there must be numerical values (either a number or a resolved variable) to their right and left.



Example: In order to toggle a variable between the values 1 and 0, you can use either of the two following methods:

```
MyVar == 0  
or  
MyVar != 1
```

Reset

If the checkmark: "*Reset*" is set, the variable is set to the initialization value. The value entered in the box "*Value*" is ignored.

User-defined variables have an initialization value which is set when the variable is created.

Device variables have no initialization value. Here, the value is set to "0".

13 Miscellaneous

13.1 Last Changes

13.1.1 in Doc. Rev. 8.1

Setup pages - Configuring device

Section	Amendments
CAN-bus assistant (firmware group B)	When multiple selection was activated, the selection of messages/channels could not be deleted. Now the messages or channels can be deleted when multiple selection is activated. The imc CAN bus assistant (firmware group B) now also supports digital bits that can be used for communication between control units.
imc SIMPLEX	Discontinued

13.1.2 in Doc. Rev. 7.4

Setup pages - Configuring device

Section	Amendments
Pretrigger	The following no longer applies to group B devices from 2024 R4: To be able to view a signal's plot including the pretrigger region, the first measurement (after prepare) needs to have been started before the trigger event by at least as much as the pretrigger time.
Configuring channels and variables	The setting parameters via the "Range & Scaling" dialogue have been expanded. In addition to the calculated measuring range, the "non-calculated" measuring range, i.e. the electrical measuring range, is now also displayed in this dialogue.
CAN-bus assistant (firmware group B)	When multiple selection was activated, neither the scaling nor the error handling could be opened. Now, with multiple channel selection, both the scaling and the error handling of these channels can be opened.

Sequencer, events and commands

Section	Amendments
System requirements	Support of NumPy 2.1.1 and 2.0.0

13.1.3 in Doc. Rev. 7.3

Setup pages - Configuring device

Section	Amendments
CAN-bus assistant (firmware group B)	The assistant supports multiple selection.

13.1.4 in Doc. Rev. 7.2

Setting Up - Software

Section	Amendments
Operating system components	The following component is supported: Microsoft VC 2015-2022 14.36.32532 Microsoft Visual C++ Redistributable\2022

Data analysis and signal processing

Section	Amendments
WFT Assistant	New: WFT-Plug-in

Setup pages - Configuring device

Section	Amendments
Filtering	The default setting is AAF, where a Cauer filter is calculated internally. The displayed characteristic (in the picture Butterworth) has no influence when the AAF filter is set.

imc Inline FAMOS function reference

Section	Amendments
Function reference: Miscellaneous	The following functions are now also verified for imc Inline FAMOS: Red, Max, Min, Mean, RMS, StDev, NumberOfPulses, HighLowRatio, Sum, Sum2, ExpoRMS, PulseDuration, PulseFrequency, PulsePhase

Sequencer, events and commands

Section	Amendments
System requirements	Support of NumPy 2.0.0 is in preparation
Transfer format from and to Python	Example added: event variables for imc STUDIO

Scripting

Section	Amendments
Events	Example was changed

13.1.5 in Doc. Rev. 7.1

Setup pages - Configuring device

Section	Amendments
Bridge measurement with strain gauge	• The strain gauge measurement modes have been updated: The descriptions and illustrations have been revised. The illustrations now show, among other things, what is present in the device and what must be connected externally to the plug. • New strain gauge measurement modes for the new amplifier "ARGFT/B-4"

13.1.6 in Doc. Rev. 7.0

imc STUDIO Monitor (general)

Section	Amendments
Import / Export	Description removed: Import Panel pages from experiment.

Monitor

Section	Amendments
Monitor - First steps	Limitation added: When the configuration is changed in imc STUDIO, the device must be re-selected in imc STUDIO Monitor.

Setup - Device (general)

Section	Amendments
Saving metadata with the channel	No storage of port metadata

13.1.7 in Doc. Rev. 6.5

Setup - Device (general)

Section	Amendments
Device overview ¹¹⁶	Device overview supplemented with imc Online FAMOS and the pre-processing.
Saving metadata with the channel	- The metadata on the channel are now also saved with the channels upon saving to the device (firmware group A). - Using the placeholder "VARS.PROPS", metadata saved with the channel can be displayed again. - The metadata saved with the channel can now again be inserted as a column in the Data Browser.

Setup pages - Configuring device

Section	Amendments
For devices of the firmware group B (imc DEVICEcore) ²²¹	- Notes on Hot-Plug capability ²²¹ have been removed. - Information on the file format ²²¹ of the internal hard drive of the imc EOS-devices. - Notes on protecting the memory card by means of routine formatting ²²². - Corrected specification of the file system: "FAT32" -> "Large FAT32"

Sequencer, events and commands

Command Reference

Section	Amendments
Run Python code file	The Python command now uses the NumPy-extension library and requires "NumPy" to be installed.
Export variable	Notes on the export-format templates added.

13.1.8 in Doc. Rev. 6.4

Setting Up - Software

Section	Amendments
Select the products / Installation type	imc SENSORS is no longer pre-selected in any installation variety.
Command line parameters	The exe-file of any program (imc STUDIO, imc WAVE, imc STUDIO Monitor) is now located in the respective installation folder and no longer in the folder "ProgramData". Furthermore, the file is provided with a signature.
Recommended virus scanner settings	

imc STUDIO Monitor (general)

Section	Amendments
Data management	"*.Prepare*" -files in the folder "config" now included in the description.

Section	Amendments
Logbook	New function: Export logbook-files Function eliminated: Send logbook via E-Mail
Options > Multi-lingual text input 	The description text was missing.

Monitor

Section	Amendments
Monitor - First steps	Transfer of the channels to imc STUDIO Monitor can be activated on a per-channel basis.

Setup - Device (general)

Section	Amendments
Ribbon - Control 	The buttons " <i>Connect</i> " and " <i>Disconnect</i> " have been removed from the menu ribbon since they had no useful function in the context of monitoring.
Metadata	Saving metadata with the channel Abgleich-Informationen können nun auch als Metadaten im Kanal gespeichert werden. Über die Option " <i>Setup</i> " > " <i>Traceability of data</i> " steht die neue Auswahl " <i>Balance information</i> " zur Verfügung.

Data analysis and signal processing

Section	Amendments
Bus Decoder - Decoding fieldbus log channels	Firmware groups added to "Supported fieldbus types" table.
Inline Analysis - Monitoring the overmodulation and under range 	New: Monitoring the under range

Channel types, variable types, data types and data storage formats

Section	Amendments
Process vector variables (pv-variables)	Info added: Deactivation of pv-variables for data recording channels is not possible for devices belonging to the firmware group A.
Data types	Data type: Text arrays from imc FAMOS
Parameter set > Mapping Import parameters > Mapping	"<i>Table description</i>" from the mapping declaration "<i>by connector</i>" has been removed: The table description is not relevant to the mapping. - The mappings have been renamed: "<i>Import of channels according to connector and device serial number</i>" → "<i>Multiple-device operation (Firmware group A)</i>" "<i>Import of channels according to connector</i>" → "<i>Single-device operation (Firmware group A)</i>" - New mapping: "<i>Uniformly per module type</i>"

Panel

Section	Amendments
Page printout or print to PDF	More detailed description of the " <i>Report template</i> "
Fullscreen	Note added: Moving the fullscreen view to a different monitor:
Load/Refill variable 	Note revised: Previously, there was a mention that with " <i>Load Variable</i> ", only user-defined variables having the scope " <i>temporary</i> " can be overwritten. However, all other user-defined variables can also be overwritten.
Graphical switch	The description of the widget " <i>Graphic switch</i> " is now available.
Preview in Windows Explorer 	The preview in the Microsoft Explorer is now independent of the default program linked. For this reason, RAW-data are now displayed correctly in the preview, even if RAW is linked with an image processing program.
Multi-lingual text input	The option is now deactivated by default and can be activated as needed.

Sequencer, events and commands

Command Reference

Section	Amendments
Set variables	• Example of toggling variables added. • The Unequal-operator has been included in the description: !=
Load/Refill variable	Note revised: Previously, there was a mention that with " <i>Load Variable</i> ", only user-defined variables having the scope " <i>temporary</i> " can be overwritten. However, all other user-defined variables can also be overwritten.

13.1.9 in Doc. Rev. 6.3

Setting Up - Software

Section	Amendments
Device Connection / Network / Firewall	The firmware programs are automatically allowed in the " <i>Windows Defender Firewall</i> " upon installation. For this reason, popup dialogs about the firewall no longer appear upon first connecting to the device.
Installation - Step by step	Description of how to download and start the Installer-file.

imc STUDIO Monitor (general)

Section	Amendments
Setting access rights for actions	Note on the right: "Save experiment"
Logbook	Copying entries, including supplemental information
Options 	New option described: Traceability of data

Setup - Device (general)

Section	Amendments
Metadata	New function described: Saving metadata with the channel

Setup pages - Configuring device

Section	Amendments
Events and circular buffer 	New section summarizing the settings for Events and Circular buffer.

Channel types, variable types, data types and data storage formats

Section	Amendments
Parameter set	The selection options for the export of the configuration for the parameter sets have been made more precise.

Panel

Section	Amendments
Preview in Windows Explorer 	Preview function in Microsoft Windows Explorer This function is now also available without having to install imc FAMOS. Activation of these functions for *.raw without imc FAMOS is not possible at this time.

Sequencer, events and commands

Command Reference

Section	Amendments
Run Python code file	New command with which to perform Python-analyses

Index

(

- () Opening Closing Bracket 283
- (%) Modulus 281
- (-) Subtraction 280
- (*) Multiplication 280
- (/) Division 280
- (;) Semicolon 283
- (^) Power 280
- (+) Addition 280
- (<) Less than operator 282
- (<=) Less than or equal operator 282
- (<>) Inequality operator 282
- (=) Equal Operator 281
- (=) Equal Sign 281
- (>) Greater than operator 283
- (>=) Greater than or equal operator 283

<

- <auto> (marker) 843

1

- 1/3-octave, octave labeling 649

2

- 2-point controller 448

3

- 32 bit integer
 - pv-variable 531
- 3D 649
 - Color palette 794
 - Display 697
 - Extras 798
 - Isolines 797
 - options 790
 - Perspective 796
 - Rotate 799
 - surface 791
- 3D-bars 791
- 3rd Party Device Management 117

4

- 4D
 - 3D + color 784

A

- ABCRating 284
- Abs 285
- Absolute date, time (curve window) 744

- Access rights of imc user group 83
- Accessing the data carrier 224
 - Error 232
- AccuLength 286
- AccuMax 287
- AccuMean 287
- AccuMin 288
- AccuRMS 288
- AccuStDev 289
- Acos 289
- Acoustics 703
- activating map mode retroactively 776
- Adapt
 - Menu (curve window) 894
 - toolbar (curve window) 894
- Adapt Panel page to window size 930
- Adapt Panel page to window size (keep aspect ratio) 930
- Adapt Panel page to window size (keep Widget size) 930
- Adapt Widgets to the size of the Panel page 930
- adapting maps 774
- Adaption channel selection
 - More Channels in Curve Window 734
- Add device interface 227
- Add file
 - Command Load/Refill variable 997
- Add variable
 - Command Load/Refill variable 997
- Adding a device (Add) 227
- Addition 280
- Additional Columns 151
 - Combined column 154
 - export 86
 - import 86
 - Metadata column 160
 - Parameter set column 162
 - Parameter set import column 167
 - save 84
- Additional Documents 67
- adjustment 7
- Advanced Diagnosis 503
- Advanced display
 - Zone 637, 641
- Aligning Widgets 624
- All available 601
- Allow a change of direction 917
 - Prevent a change 917
- Allow data export 981
- Allow discarding of all measurements 981
- Allow login of anonymous users 76
- Allow selection of the export folder 981

- Amplitude-Shift 764
- Analog channels 182
- Analog inputs 182
- Analog outputs 182, 260
- AND 290
- Anonymous users
 - Logging allow 76
 - Logging prohibit 76
- Appearance
 - Property - Widget 629
- Application option 61
- Arrangement
 - Widgets 624, 625
- ARON measuring circuit 275
- Arrangement
 - Axes 746
 - Configuration (curve window) 808
- Arrangement (Widgets)
 - Align Bottom 624
 - Align left 624
 - Align Right 624
 - Align top 624
 - Cascade 624
 - To Background 623
 - To Foreground 623
- As presettings
 - Curve Window 809
- ASCII
 - imc Format Converter 576, 965
 - imc Format Converter Command 965
- ASCII display (curve window) 670
- Asin 290
- Aspect ratio for RGB image 747
- Assignment 187
- Atan2 290
- auto
 - Buffer time 206, 207
- Auto filename 196
- Automatic table length 917
- Autosize
 - Property - Widget 630
- Available Channels
 - Context menu 730
 - More Channels in Curve Window 730
- Available events 188, 197, 201
 - Display and calculations 201
- Axes
 - Arrangement 746
 - labeling 649
 - Scaling 736
 - Text 748

- Axes List 727
 - Symbols 727
- Axes Navigation Bar (curve window) 905
- B**
- Back (curve window) 861
- Background
 - Property - Widget 629
- Background color
 - Property - Pages 928
- Background image
 - Property - Pages 928
- Background image layout
 - Property - Pages 928
- Background plain colored
 - Property - Pages 928
- background picture (curve window) 771
- Backups of experiments 92
- Balance 117
- Balance values import 562
- Bar meter (curve window) 649, 676
- Base Diagnosis 496
- Base path for measurement data storage 62, 213
- Beeper 260
 - Overflow 246
- Binary 921
- BitAnd 291
- Bitmask
 - Property - Widget 630
- BitNot 291
- BitOr 291
- Bits 523
- BitXor 292
- Block comment
 - imc Inline FAMOS 262
- Bookmarks
 - imc FAMOS 484
- boolean variables
 - from files 268
 - IF conditions (OFA/IFA) 268
- Border
 - Background 629
- boundaries (curve window) 825
- boundary intervals (curve window) 825
- Boxplot (curve window) 715
- Breakpoints
 - imc FAMOS 484
- Browse in workspace
 - Layout 985
 - Page 985
 - Parameters 985

- Bubble Chart 717
- Bubble Plot
 - change edge 717
 - mit Farbkanal 722
 - Standard 717
- Buffer time 205
 - auto 206, 207
- Bus Decoder 487
 - Opening the Editor 488
 - User interface 488
 - Which fieldbus 488
- Button 918
 - create 59
- C**
- CAEMAX 516
- Calibrating the amplifiers 185
- calibration 7
- Calibration date 185
- CAN protocol
 - CANalyser-MDF format 541
- CANalyser-MDF format 541
- CAN-Bus
 - Send with imc Online FAMOS 266
- CAN-Bus-Data (curve window) 670
- Cancel Button
 - Show message box 989
 - Text (Button caption) 989
 - Visible (Hide) 989
 - Visible (Show) 989
- CanMsg 447
- Caption in tables 918
- Cardioid characteristic 703
- Cardioid pattern 703
- Cascade Widgets 625
- ccv file
 - loading 868
 - saving 870
- CE Certification 7
- Cell title 918
- Centered
 - Background image 928
- Certain parts from data (curve window) 801
- Certificates 7
- CF card (Compact Flash) 225
- CFast Storage medium 225
- Change requests 7
- changing the scale 885
- Channel comment 185
- Channel count 184
- Channel definition (Setup page) 185
- Channel file name 196
- Channel info
 - More Channels in Curve Window 735
- Channel name 185
 - imc Inline FAMOS 262
- Channel number 191
- Channel status 185
- Channel table 182
- Channels 182
- Charact 293
- Characteristic curves
 - export 86
- Check - Syntax 253
- Check Box 918
- Choose Tool Windows 58
- Circular buffer time 188, 197, 202
 - Display and calculations 202
- Class Learning Time 497
- Class-counting 278
- CIHistogram 293
- Clipboard 878
 - curves 856
- clipboard format (curve window) 889
- Clipboard settings 878
 - Font 878
 - Line style 878
 - Line width 878
 - plot 878
 - Print 878
 - printout 878
 - Ticks 878
- CILevelCrossing 294
- CIMarkov 295
- Clock 911
- CloseSaveInterval 296
- CIRainFlow 297
- CIRainFlowRes 298
- CIRainFlowTM 299
- CIRainFlowTMRes 300
- CIRangePairCount 301
- CIRevolutionsHistogram 301
- CIRevolutionsMatrix 302
- CITrueMax 302
- CITwoChannelHistogram 302
- Cluster size 543
 - FAT32 222
- CodeRange 303
- Collection Editor 151
- Color 185, 190
 - Title 628

- Color components 882
- Color grandient
 - Background color 928
- color map (curve window) 649, 784
 - Display 685
 - general 688
 - ISO-lines 694
 - measure mode 821
 - Waveforms with associated x-,y-,z-variables 696
- color of Y-axis 750
- Color palette
 - 3D 794
- color palette (curve window) 784
- Color spectrum 685
- Coloring
 - Property - Widget 629
- Colors (curve window)
 - Copy 882
 - Curves 882
 - Screen/Printer 882
- Column captions 169, 918
- Column description
 - save 84
- Column descriptions
 - export 86, 173
 - import 86, 173
- Column ID
 - CurrentValueDisplay 192
 - eAllocation 187
 - eChannelComment 185
 - eChannelName 185
 - eCurveColor 185, 190
 - eCurveYAxisMax 190
 - eCurveYAxisMin 190
 - eCurveYAxisOption 190
 - eDuration 186
 - eEnumeratedChannelNumber 191
 - eHistogramSaveInterval 189
 - eHistogramUpdateInterval 189
 - ePluginName 191
 - eStatus 185
 - eXFormatVariable 187
 - SampleCount 186
- Column title 918
- Columns
 - Showing and moving 148
- Combined column 154
- Combined Parameters 154, 164
- Combo Box 918
- Command
 - Delete supplemental file 985
 - Delete Variable 999
 - Execute file 962
 - Execute imc FAMOS sequence 955
 - Export Panel page 977
 - Export supplemental file 983
 - Export Variable 993
 - imc FAMOS project 954
 - imc FAMOS sequence 955
 - Import imc Inline FAMOS source code 972
 - Import supplemental file 984
 - Load variable 995
 - Open Logbook viewer 990
 - Panel page - Export 977
 - Panel page as dialog 975
 - Panel page entfernen 978
 - Refill variable 995
 - Set measurement number 973
 - Set variables 999
 - Show Panel page as dialog 975
 - Stop Sequencer 971
 - Supplemental file - Delete 985
 - Supplemental file - Export 983
 - Supplemental file - Import 984
- Command line parameters 31
- Command Reference 954
- Comment 185
 - Property 272
 - Title - Default Source 628
- Comment a line
 - imc Inline FAMOS 262
- Comment several lines
 - imc Inline FAMOS 262
- Communicator
 - create scroll link 862
 - Transfer to imc FAMOS 871
- Compact Flash 225
- Comparison operators 248, 265
- Complete
 - View 28
- Complete layout 174
- Complete layouts 131
 - export 86
 - import 86
- Components 29
 - Product configuration 27
- Configuration (curve window)
 - Arrangement 808
 - loading 868
 - Menu 877
 - saving 870
- Configuration transfer

- Configuration transfer
 - Curve window 904
- Connect 117
- Connect (curve window) 861, 862
 - cross-section with 3D 863, 864
- Connect XY with 2nd curve window 862
- Connection 179
 - to Device 29
- Connection failed - transfer configuration 62
- Connection status 179
 - Variables 530
- Connector 191
- Context menu
 - Available Channels 730
 - Curve window 904
- CONTROL 97
- Control commands 247, 248, 263
 - activate 248
 - Array 264
 - Comparison operators 265
 - Create a process vector variable 264
 - Create a variable 264
 - Data array 264
 - deactivate 248
 - Single value 264
 - SingleValueChannel 264
- Control functions 248
- Controller functions 448
- Conversion 45
 - Database 16
- Coordinate system size 649
- Coordinates (map) 783
- Copy
 - Logbook 73
- Copy & Paste
 - Curve window 904
- Copy data to the PC 222, 226
- Copy with details
 - Logbook 73
- Cos 303
- Create a process vector variable 264
- CreateVChannel 303
- CreateVChannelInt 304
- Creating the columns with the Collection Editor 151
- Cross sections
 - Lines 766
- CrossCorrelation 304
- Cross-section with 3D
 - connect 863, 864
- CtPID 448
- CtPID.Calc 449
- CtTwoPos 450
- CtTwoPos.Calc 451
- Current measurement 939
- Current value 192
- CurrentValue 305
- Cursor
 - Measure window 816
 - Messfenster 816
- Curve color 185, 190
- Curve configuration
 - loading 868
 - saving 870
- Curve properties (Setup page) 189
- Curve window
 - changing the scale 885
 - clipboard 856
 - clipboard format 889
 - construction 725
 - context menu 904
 - Date/time absolute 649
 - Days/hours/minutes relative 649
 - Double-click on empty areas 889
 - Enable horizontal measurement cursor 889
 - Fill mode 800
 - Font 885
 - Global Handling 889
 - Graphic export 889
 - graphic export as Vector graphic 885
 - hiding data 725
 - history 885
 - import a picture or map as background 773
 - Introduction 647
 - legend 768
 - Line-Shift 885
 - link 885
 - Max. delay [ms] 885
 - Measure 814
 - Measure window 814
 - More Channels 725
 - Navigation in x-direction 889
 - New coordinate systems 727
 - New x-axis 727
 - New y-axis 727
 - Oscilloscope mode 800
 - PDF orientation 889
 - Pointer 707
 - Rezoom 889
 - scroll links to tables 862
 - Scroll mode 800
 - selection mode 885
 - settings 885
 - Size of measurement point 889

- Curve window
 - sound output 885
 - Start delay [ms] 885
 - Stretch mode 800
 - Time-Shift in the ccv 885
 - touchscreen operation 754
 - Transfer to imc FAMOS 889
 - Undo 885
 - Updating 943
 - XY plots 732
 - Zoom 889
- Custom base path for measurement data storage 62, 213
- Custom folder naming for measurement data storage 62, 213
- Customer support
 - Tech support 6
- Customize Ribbon Menu 58, 88
- Cut sound
 - sound output (curve window) 899
- 
- DAC 260
 - Scaling 260
- Data format
 - imc2 574
 - imc3 574
- Data loss 202
- Data overflow 198, 205, 236
- Data recording with sampling 233
- Data recording with time stamp 233
- Data saving 194
- Data Saving Assistant 979
 - Allow data export 981
 - Allow discarding of all measurements 981
 - Allow selection of the export folder 981
 - Default button 982
 - Default path 981
 - Delete original files 981
 - Deleting a measurement 981
 - Discard measurement 981
 - Export measurement settings 981
 - imc Format Converter 980
 - Measurement comments 980
 - Parameter set 980
 - Prompt to confirm discard 981
 - Setup pages 980
- Data security mechanism for experiments 92
- Data storage 211
 - Resume 118, 218
 - Suspend 118, 218
- Data stream 194
- Data transfer 196, 222, 226
 - FTP access 228
 - Password 222
 - Storage medium 222, 226
- Data transfer (Setup page) 188
- Data types 523
 - Memory requirement 543
- Data Browser 597
 - All available 601
 - current measurement in the Data Browser 599
 - Event time 601
 - Load variable 605
 - Measurement number 603
 - measurements in the Data Browser 599
 - Refill variable 605
 - renaming measurements 599
 - saved measurements 599
- Data Browser columns
 - Category 598
 - Comment 598
 - Event time 598
 - Metadata columns (optional) 598
 - Name 598
 - Numbering column (#) 598
 - Unit 598
- Database
 - conversion 45
- Database conversion 16
- Database folder 45
- DATABASE/DATABASES 98
- Date and time 911
- Days/hours/minutes relative (curve window) 744
- dB 306
- Deactivating the pv-variables 531
- Decimal 921
- Decimal places
 - Property - Widget 630
 - Zone 645
- Default 445
- Default button
 - Data Saving Assistant 982
- Default path
 - for export 981
- Default printer (curve window) 873
- Default Source
 - Title 628
- Default value 161
- Default values 121
 - Applied condition 121
 - Default value 121
 - Element type 121

- Default values 121
 - Enumerator class 121
 - Group name 121
 - Parameter 121
 - Parameter identifier 121
 - Row scope 121
- DelayBuffer 306
 - Fill 308
 - Next 309
 - SetSize 310
- DelayBuffer.Fill 308
- DelayBuffer.Next 309
- DelayBuffer.SetSize 310
- DelayLine 311
- Delete
 - experiment 48
 - project 48
 - Set variables 999
 - Supplemental file 985
 - views 85
- Delete original files 981
- Delete sample 917
- Delete supplemental file 985
- Delete Variable 999
 - Show dialog 999
 - Treat error as warning 999
 - Variable name 999
 - Variables list 999
- Deleting a measurement
 - Data Saving Assistant 981
- Delta-connection 276
- Delta-x 272
- Delta-z 272
- Description 177
- Design Mode 616
- Designation data folders 211
- Destination location
 - Installation 18
- Deviation of system time from NTP 531
- Device
 - connection 29
- Device control 179
 - Starting measurement 118, 179
- Device group 116
- Device is known 142
- Device is selected for measurement 178
- Device name 179
- Device overview 116
- Device selection 142
- Device SN 179
- Device specification 179
- Device system variables 528
- Device Table 178
- Devices - Documentation 39
- Devices deselected 62
- DFilt 311
- Diff 312
- Digital channels 182
- Digital inputs 259
- Digital inputs / outputs (bits and ports) 182
- Dimension Lines 837
- DIN-EN-ISO-9001 7
- DIO
 - Widget 921
- DirClosed 198, 220
- Direction 917
- Directory structure 211
 - Trigger 212
- Disable if this column is empty 154
- Discard measurement
 - Data Saving Assistant 981
- Disconnect 117
- Disconnect from running measurement 62
- DiskFreeSpace 312
- DiskFreeTime 313
- DiskRunDir 314
- Display
 - 3D 697
 - bar meter 676
 - color map 685
 - Number (curve window) 665
 - waterfall diagram 659
- Display (curve window) 649
- Display format 154
- Display format parameter 154
- display options
 - third-octave axes 707
- Display Setup dialog 982
- Display text
 - Zone 919
- display uncertainty 763
- Display update interval
 - Histogram 189
- Display variables 183
- Display, calculations of measured data 188
- DisplaySetButton 314
- DisplaySetPage 314
- Division 280
- Dock
 - Property - Widget 629
- Dock (Widgets)

- Dock (Widgets)
 - Below 626
 - Fill 626
 - Free 626
 - Left 626
 - Right 626
 - Top 626
- Documentation 39
- Documentation (Setup page) 177
- Documentation of devices 39
- DotNET (.NET) 970
- drag to WORD/PPT 858
- Drag&Drop
 - curve window 904
- Drive Element 511
- Drive elements
 - Belt Gearing 504
 - Clutch 504
 - Drive Shaft 504
 - Gear Drive 504
 - Planetary Gear 504
 - Rolling Bearing 504
 - Shaft 504
 - Steady Frequency 504
 - Steady Order 504
 - Three-Phase Motor 504
- Duration 186

E

- Entry format 161
- Enumerated channel number 191
- Enumeration classes 147
- Environment variables of the OS 114
- Equal 317
- Equal Operator 281
- Equal Sign 281
- Error (Logbook) 71
- Error causes upon starting 33
- Ethernet bits 183, 527
- Event (curve window) 801
- Event time
 - All available 601
 - Data Browser 601
- Event-based data set 201, 524
- Events 188, 201
 - Available events 197, 201
 - Property - Widget 630
- EXCEL
 - imc Format Converter 576, 965
 - imc Format Converter Command 965
- Excel-Display (curve window) 670
- Execute device action
 - Starting measurement 118
- Execute file 962
- Execute imc FAMOS sequence 955
- Execute menu action
 - Command 991
 - Widget 915
- Execute program 962
- Exit fullscreen mode 935
- Experiment 43, 48
 - copy 49
 - delete 48
 - export 48
 - import 48
 - new 47, 49, 54
 - open 54
 - save (as) 49, 54
- Experiment from template 49
- Experiment has changed 62
- Experiment option 61
- Experiment template 45, 48, 52
 - export 48
 - import 48
 - new 47
 - preferred 53
- Experiment, open
 - Automatically upon starting imc STUDIO 31
 - With a specific imc STUDIO installation 31
- EXPERIMENT/EXPERIMENTS 99

- Experiment/Project
 - Difference 91
 - What is saved where? 91
- Expert mode 557
- ExpoRMS 318
- Export
 - Column descriptions 173
 - experiment 48, 50
 - experiment template 48
 - Logbook 73
 - Measure window 818
 - project 48, 50
 - Table description 173
- Export all channels
 - Mapping 564
- Export all channels and settings
 - Mapping 564
- Export current data 217
 - Options 218
- Export measurement settings 981
- Export Panel page 977
- Export parameters 554
- Export supplemental file 983
- Export Variable 605, 993
 - Add variables 993
 - Always overwrite existing files 994
 - File format 994
 - File options 993
 - Filename 994
 - Folder 993
 - Mapping instruction 993
 - Measurement 993
 - Measurement name 994
 - Save all in a single file 993
 - Save to one file per file format 993
 - Show file options 994
 - Show variables options 994
 - Treat error as warning 994
 - Variable name 994
 - Variables 993
 - Variables list 994
- Export views, meta data columns, sensors, ... 86
- ExpressCard Storage medium 225
- Extension for Windows-Explorer 226
- Extra menu
 - Administration 60
 - Login 60
 - Logout 60
 - Options 60
 - Start (Sequencer) 60
 - Startup behavior 60
 - Stop (Sequencer) 60
 - User Access Rights 60
- Extras (curve window)
 - 3D 798
 - Lines 763
 - Menu 892
 - Menu and Toolbar 893
- F**
- Factor
 - Property - Widget 630
- FAMOS-ZIP format
 - Limitations 540
- Farbkarte
 - Wirkung (Extras) 764
- Fast loading
 - Property - Pages 929
- FAT16 543
- FAT32 543
- Fatal (Logbook) 71
- FFT 320
- FFTAplitudePhase 321
- FFTAverage 322
- FFTInverse 323
- FFTReallmaginary 323
- fieldbus
 - Analog inputs 182, 187
 - data recording 233
 - Digital inputs / outputs 187
 - Digital inputs / outputs (bits and ports) 182
- File
 - Menu (curve window) 867
- File comment 994
- File format 197, 539, 540
 - CANAlyser-MDF 541
 - Header 542
 - imc FAMOS 540
 - imc FAMOS ZIP 540
 - Key 542
- File header 542
- File name for channel 196
- File System
 - FAT32 222
- File system FAT16/FAT32 229
- Filesize (maximum) 229
- Fill mode
 - Curve window 800
- FiltBP 324
- FiltBS 324
- Filter
 - instable state 324

Filter

Setup table 137

filtering (curve window) 823

FiltHP 325

FiltLP 325

Firewall

Security alert 29

Fix Axes 861

Fixed-point (curve window) 665

Flags 31

Float

pv-variable 531

floating-point notation 665

Floor 325

Flush right (curve window) 665

Folder

Installation 18

Folder naming for measurement data storage 62, 213

Font

Property - Widget 629

Title 628

Font (curve window) 885

Last value as number 665

For 444

format marker (curve window) 843

Formatkonverter

Export formats 963

formatting 223, 229

Formatting (x-axis absolute)

Format line 1, 2 737

Forward (curve window) 861

frequency bands

third-octave display 707

From default settings 52

From existing experiment 52

From present settings 52

From Variable

Range 630

FTP access

Data transfer 228

Full functionality for 30-day demo 21

Fullscreen 933

Automatically upon starting imc STUDIO 31

Exit fullscreen mode 935

Launch 933

Panel 589, 593

Function Reference

imc Inline FAMOS 279

imc Online FAMOS 279

G

GearRatio 326

General terms and conditions 7

Generate default values from selection 121

GetDateTime 326

GetDuration 328

GetHistoValue 328

GetHistoValue2 329

GetLastError 330

GetSampleCount 333

GetSamplingTime 333

Global Handling (curve window) 889

GPS

process vector variables 533

RS232 settings 534

GPS data as XY-plot 779

GPS-Channels 182

Graphic export

Bitmap 885

Vectorgraphic 885

Graphic export (curve window)

Bitmap 889

Vektorgraphic 889

Graphical switch

Widget 913

Greater 333

Greater than operator 283

Greater than or equal operator 283

GreaterEqual 333

Greek characters (curve window) 901

Grid

Curve Window 790

Grid (curve window) 649

small x-ticks (-1 = auto) 649

small y-ticks (-1 = auto) 649

Grid (Panel)

Realign to Grid 932

Snap to Grid 932

Grid size

Property - Pages 929

Group

Device overview 116

Group (Widgets)

Create Group 627

Dissolve Group 627

Enter Group 627

Leave Group 627

Group By Box

Setup table 136

Grouping

Grouping
 Setup table 136
Guarantee 7
Guardian 94
GuardianService 94
GuardianWDServices 94

H

Hard drive 198, 221
harmonic cursor 851
Header (file header) 542
Headline
 Show message box 989
Heavy burdening of the processor 943
Height
 Property - Pages 928
Help and Documentation 39
Help menu
 Additional documents 67
 imc website 67
 Product configuration 67
Hexadecimal 921
Hide disabled columns from caption 154
Hide Passive Channels 117
hiding data
 curve window 725
HighLowRatio 334
high-pass (curve window) 823
Histogram / Rainflow (Setup page) 189
Histogram display update interval 189
Histogram storage interval 189
history (curve window) 885
Homepage 28
Hotline
 Tech support 6
Hot-Plug
 FTP access 228
 Storage medium 224
Hyst 335

I

Identifier 153
Identity 76
iDiv 336
IEEE Float
 pv-variable 531
If 442
IF conditions
 boolean variables 266
IFA 245
Image

Widget 913
imc Data format
 imc2 574
 imc3 574
imc Help and Documentation 39
imc Language Selector 34
imc Software License Agreement 8
imc Systems 226, 227
 formatting 229
imc WFT Assistant 516
imc FAMOS
 Always overwrite existing files 475
 Bookmarks 484
 Breakpoints 484
 congestion 484
 Current measurement 478
 Debug Mode 475
 From imc FAMOS 478
 Last concluded measurement 478
 many data cutting interaction 484
 Measurement 478
 Measurement number 478
 Open 475
 Options 478
 parallel data cutting interaction 484
 parallel processing of sequences 484
 Synchronous event 475
 Timeout 475
 To imc FAMOS 478
 Transfer table 478
imc FAMOS project 954
imc FAMOS sequence 955
imc Format Converter
 ASCII 576, 965
 Command line parameters 583
 Data Saving Assistant 587, 980
 EXCEL 576, 965
 Export formats 575, 963
 Save in one file 576, 965
 Sequencer Command 585, 963
 Settings 576, 965
 Stand alone (Operation) 581
 Storage location 576, 965
 Windows-Explorer 584
imc Format Converter Command
 ASCII 965
 EXCEL 965
 Save in one file 965
 Sequencer Command 963
 Settings 965
 Storage location 965
imc Inline FAMOS 245

- imc Inline FAMOS 245
 - Class-counting 278
 - Comparison 239
 - Data flow schematic 239
 - Import source code 972
 - maximaum stack 279
 - Nesting depth 279
 - pv-variable 531
 - Ribbon 486
 - Supplemental Files 277
 - System requirements 244
- imc LICENSE Manager 27
- imc Online FAMOS 245
 - Local channels 259
 - Local single value variables 259
 - Local variable 259
 - Single value variables 259
- imc STUDIO DataProcessing 485
 - Ribbon 486
- imc WAVE Noise 455
- imc WAVE Rotation 461
- imc WAVE Structure 471
- imc WAVE Vibration 465
- imc2
 - Data format 574
- imc3
 - Data format 574
- imcSyslog 198
- Import
 - Column descriptions 173
 - experiment 48, 51
 - experiment template 48
 - project 48, 51
 - Source code 972
 - Table description 173
- Import / Export
 - menu 54
- import a picture or map as background (curve window) 773
- Import imc Inline FAMOS source code 972
- Import mapping by name
 - Mapping 564
- Import of channels according to connector
 - Mapping 564
- Import of channels according to connector and device serial number
 - Mapping 564
- Import Panel page 978
- Import parameters 559
- Import supplemental file
 - Mapping table 984
- Import views, meta data columns, ... 86
- ImportRoot 169
- In experiment
 - Metadata 161
- Increment
 - Property - Widget 630
- Incremental counter inputs 182
- Individualized colors for this window 882
- Inequality operator 282
- Information (Logbook) 71
- Inline Analysis 451
- Inline FAMOS 245
 - Apply 253
 - Block comment 262
 - Channel name 262
 - Check - Syntax 253
 - Comment a line 262
 - Context menu 256
 - Digital inputs 259
 - Line comment 262
 - Menu 255
 - Schematic 245
 - Special characters 262
 - Starting 249
 - Syntax check 253
 - Time Base for Channels 262
 - Virtual channels 258
- Inline Function 452
- Input variables
 - Python 955
- Insert sample after 917
- Insert sample before 917
- Insert Widget 617
- instable filter functions 324
- Installation 14
 - imc DEVICEcore 23
 - imc DEVICES 23
 - imc FAMOS 25
 - imc SENSORS 24
 - imc Shared Components 23
 - imc STUDIO 24
 - Projects 36
 - Security ofware 18
 - Silent 37
 - Step by step 18
 - Unattended 37
 - Update 14
 - User account control 17
 - Views 36
 - Virus scanner 18
 - Your own personal programs 36

- Installation type
 - Full functionality for 30-day demo 21
 - Typical functionality incl. imc STUDIO Professional 21
 - User-defined 21, 23
- Integral 336
- Integral2 336
- IntegralFFT 336
- IntegralP 337
- IntegralP2 337
- Internal storage media 221
- Internet Settings for maps 783
- Interval
 - Count 202
 - Property - Widget 633
 - Saving 202
- Interval (Timeout)
 - Show message box 989
- Interval funktions 337
- IntervalFrom1Level 339
- IntervalFromLevels 340
- IntervalFromPulse 341
- IntervalMax 341
- IntervalMean 342
- IntervalMin 342
- IntervalMult 343
- IntervalResample 344
- IntervalRMS 344
- IRIG Format 911
- ISO-9001 7
- Isolines
 - 3D 797
- Issues
 - Storage medium 224, 232
- IsSynchronized 345
- J**
- Jiggling display 35, 943
- JKFlipFlop 345
- K**
- Kurvenfenster
 - freie Texte 788
 - Text 788
- L**
- Label for raw data 753
- Labeling
 - Lines 761
- Language
 - Installation 18
 - Language change 34
 - Language installing 35
- Last measurement 219
- Last value as number (curve window) 649
 - Display 665
 - Font 665
- Launching in Fullscreen mode 933
- Layout 58
 - Browse in workspace 985
- Layout Repository 131
- Layout-Repository
 - export 86
 - import 86
- LED 260
- LED6 118
 - Overflow 246
- legend
 - curve window 768
- legends (line) 762
- LEQ 346
- Less 347
- Less than operator 282
- Less than or equal operator 282
- LessEqual 347
- Level Indicator 907
- Liability restrictions 7
- Licensable components 27
- License 27
- Limitations
 - Storage medium 224, 232
- Limited Warranty 7
- line color
 - depend on magnitude 784
- Line comment
 - imc Inline FAMOS 262
- Lines (curve window) 755
 - Color 756
 - Cross sections 766
 - Extras 763
 - Labeling 761
 - Line structure 756
 - Line thickness 756
 - Line type 756
 - Lines 756
 - Printer / Screen: 756
 - Symbol 756
 - Symbol size 756
- Line-Shift 764, 861, 865
 - in the ccv 885
 - Reset 865
- Link
 - Metadata 161

- link (curve window) 885
 - Link + in experiment
 - Metadata 161
 - Link with several curve windows 863, 864
 - Link XY with 2nd curve window 862
 - linking a map with time data 778
 - Lissajous figures
 - curve window 714
 - Ln 347
 - Load
 - views 85
 - Load dialog (curve window) 868
 - Load measurement settings 610
 - Load variable 605
 - Add file 997
 - Add variable 997
 - Category 608, 997
 - Command 995
 - Data Browser 605
 - Load all 998
 - Show dialog 998
 - Target measurement 608, 998
 - Target variable name 607, 997
 - Treat error as warning 608, 998
 - Variable name 607, 997
 - Variables list 607, 997
 - Local channels 259
 - Local single value variables 259
 - Local system information
 - Variables 528
 - Local variable 259
 - Lock page 932
 - locus curve 701
 - Log off
 - Logging a user off 80
 - Log on
 - anonymous user 80
 - registered user 80
 - LogAnd 347
 - Logbook 71
 - Autoscroll 73
 - Category 71
 - Copy 73
 - Copy with details 73
 - Deleting 73
 - Duplicates 73
 - Export 73
 - Filtering 72
 - Location 71
 - Logbook-Viewer 73
 - Options for the Logbook 73
 - Search 72
 - Zip file 73
 - Logbook (curve window) 670
 - Logbook category
 - Error 71
 - Fatal 71
 - Information 71
 - Warning 71
 - Logbook viewer
 - Opening by command 990
 - LogNot 347
 - LogOr 348
 - LogXor 348
 - Long date 911
 - Long name
 - Title - Default Source 628
 - Long time 911
 - Lower 348
 - low-pass (curve window) 823
- ## M
- Magnitude (curve window) 665
 - Main window 29
 - maintenance 7
 - Making a device known 142
 - Manage Projects 47, 54
 - Managed DLL 970
 - Mandatory box 161, 164, 169
 - Map 773
 - cache 783
 - curve window 777
 - from Internet 779, 780
 - link 778
 - Map provider, Adding- 781
 - mode 774
 - provider 780
 - settings 783
 - stretched 782
 - map coordinates adjust 773
 - Map provider, Adding-
 - Map in curve window 781
 - Mapping 564
 - Export all channels 564
 - Export all channels and settings 564
 - Import mapping by name 564
 - Import of channels according to connector 564
 - Import of channels according to connector and device serial number 564
 - Multiple-device operation (Firmware group A) 564
 - Single-device operation (Firmware group A) 564
 - Mapping table

- Mapping table
 - Export supplemental file 983
 - Import supplemental file 984
- Marker
 - Voreinstellungen 832
- Marker (curve window) 832, 843
 - Delete all markers 849
 - extra functions 849
 - Markers at Min/Max 849
 - Markers on all lines 849
 - move 841
 - new line 835
 - Select all lines 849
 - set 834
- marker precision 843
- Matrix (curve window) 806
- Max 190, 348
- max. delay (sound) 900
- Max. delay [ms] (curve window) 885
- Maximum stack
 - imc Inline FAMOS 279
- Maximum
 - Range 630
- Maximum amount of variables in imc Inline FAMOS 244
- MDF format 541
- Mean 350
- Measure (curve window)
 - 1/dx (frequency) 814
 - dx 814
 - Slope 814
 - Slope per decade 814
 - xl 814
 - xr 814
 - xr/xl 814
 - yl 814
 - yr 814
 - yr/yl 814
 - yr-yl 814
- Measure context menu
 - Channels List 818
 - Clipboard 818
 - Expand List 818
 - Export Curve Section 818
 - Place marker with left click 818
 - Send Curve Section 818
- Measure mode
 - Curve 813
- Measure window (curve window) 813, 814
 - color maps 821
 - context menu 818, 821
 - cursor positioning 816
 - Date/ time display 814
 - Days/ hours/ minutes display 814
- Measurement 179
 - loading 602
 - Start 118
 - Stop 118
 - unloading 602
- Measurement comments
 - about Data Saving Assistant 980
- Measurement cursor (curve window) 813
- Measurement data 43
 - Memory requirements 542
- Measurement folder structure 213
- Measurement number 939
 - Assigning 603
 - Locking 603
- Measurement process states 436
- Measurement running 62
- Measurement running - reconnect not possible 62
- Measurement status 179
 - Variables 530
- Measurement storage area 213
- MEASUREMENT.SQL 102
- Median3 351
- Median5 351
- Memory card 221
- Memory cards 116
- Memory requirements 542
- Menu
 - Edit (curve window) 876
 - Extras (curve window) 892
 - File (curve window) 867
 - Options (curve window) 878
- Menu (curve window)
 - Adapt 894
 - Configuration 877
 - Toolbar 894
- Menu and Toolbar (curve window) 893
- Mercator-projection 782
- Messfenster (Kurvenfenster)
 - Cursor positionieren 816
- Messunsicherheit 766
- Meta 46
- Metadata 48, 144
- Metadata column 160
- Metadata readout 144
- Metadata save in the channel 144
- Metadata type
 - Date 160
 - Directory 160
 - Document 160

- Metadata type
 - Logical value 160
 - Multiline text 160
 - Picture 160
 - Single-line text 160
 - Text from list 160
 - Time 160
 - Metadata-column
 - export 86
 - import 86
 - Metafile
 - transferring curve windows 856
 - Microphone polar pattern 703
 - microSD Storage medium 221
 - Min 190, 351
 - Minimum
 - Range 630
 - Minimum length of
 - Password 76
 - User name 76
 - Modify values (curve window) 861
 - Modulus 281
 - Monitor 933
 - Analog inputs 182
 - Digital inputs / outputs (bits and ports) 182
 - Incremental counter inputs 182
 - System requirements and limitations 40
 - Monitor channels 182, 527
 - Monoflop 352
 - MonoflopRT 353
 - More Channels in Curve Window 725
 - Adaption channel selection 734
 - Available Channels 730
 - Channel info 735
 - Waveform Preselection 731
 - Waveforms in axes list 732
 - Most recently use values 161
 - mouse wheel
 - curve window 904
 - move area in curve window 904
 - Move markers
 - curve window 841
 - Moving and showing columns 148
 - Multi-Event
 - Curve window 943
 - Multiline comment
 - imc Inline FAMOS 262
 - Multi-lingual text input 63, 944
 - Multilingual title
 - Title 628
 - Multimeter (curve window) 665
 - Multiple triggering
 - Memory requirements 544
 - Multiple-device operation (Firmware group A)
 - Mapping 564
 - Multiplication 280
- N**
- Name 179, 185
 - Property 272
 - Property - Pages 928
 - Property - Widget 628
 - Show message box 989
 - Title - Default Source 628
 - Names (curve window) 665
 - Navigation bar (curve window)
 - Axes 905
 - Navigation in x-direction (curve window) 889
 - Navigation Pane 29, 68
 - Necessary system component
 - Installation 23
 - Nesting depth
 - imc Inline FAMOS 279
 - Net bits 183
 - Network
 - Firewall 29
 - Network drive 198
 - New
 - experiment 49, 54
 - experiment from template 49
 - experiment template 47
 - project 47, 54
 - New experiment 47, 49
 - from template 49
 - NMEA 533
 - NMEA Talker IDs
 - GA, GB, GI, GL 535
 - GN, GP, GQ 535
 - No new devices found 62
 - Noise 455
 - noise (curve window) 823
 - Normal data set 523
 - NorthCorrection 354
 - NOT 355
 - Number (curve window)
 - Display 665
 - Number of symbols (0 = every sample) 649
 - Number trimming (useful when scrolling) 649
 - NumberOfPulses 355
 - Numeric Zone
 - Zone 644
 - Numerical zones 919

O

Octal 921
octave labeling
 display options 707
OFA 245
Offline cache (map) 783
Offset 917
Offset (curve window) 823
OK Button
 Show message box 989
 Text (Button caption) 989
 Visible 989
 Visible (Hide) 989
 Visible (Show) 989
OLE 858
OnAlways 437
OnECUCmdReturn 316
OnECUCmdReturn_ECU_001 357
OnInitAll 266, 437
Online FAMOS 245
 Apply 253
 Block comment 262
 Channel name 262
 Check - Syntax 253
 Comment a line 262
 Context menu 256
 Digital inputs 259
 Line comment 262
 Menu 255
 Schematic 245
 Special characters 262
 Starting 249
 Syntax check 253
 Time Base for Channels 262
 Virtual channels 258
OnMeasureEnd 438
OnMeasureStart 438
OnPowerOff 441
OnSyncTask 439
OnTimer 441
OnTriggerEnd 266, 440
OnTriggerMeasure 440
OnTriggerStart 266, 439
Open 47
 experiment 54
 imc FAMOS 475
Open source software licenses 33
Opening Closing Bracket 283
Opening the Bus Decoder Editor 488
Operable

Property - Widget 629
Operating system
 System requirements 13
Optional components 27
Options 61
 3D 790
 colors 882
 Export current Data 218
 Menu (curve window) 878
 Presettings (curve window) 885
 Save current Data 218
OR 357
Orientation
 Property - Widget 629
Oscilloscope mode
 Curve window 800
OtrAngleAdd 358
OtrEncoderPulsesToRpm 358
OtrFrequLine 361
OtrFrequLine2 362
OtrFrequLine3 362
OtrOrderSpectrum 363
OtrOrderSpectrumP 365
OtrPulseDuration 367
OtrResample 368
OtrResampleAngle 370
OtrResampleFromRpm 371
OtrRpmComplexOrder 372
OtrRpmOrder 375
OtrRpmPresentation 377
OtrRpmPresentVector 378
OtrRpmSpectrum 379
OtrSynthSin 379
OtrTrackingLowPass 380
Output variables
 Python 955
Overflow
 Beeper 246
 imc Inline FAMOS 246
 imc Online FAMOS 246
 LED6 246
overload 474
overmodulation 474
Overview Drive Elements 511
Overview window 874
Overwrite autostart configurations 62
Overwrite existing file 62
Overwrite this multilingual text? 944

P

Page 105

- Page 105
 - Browse in workspace 985
 - Property - Pages 928
 - Repository 612
- Page setup 925
- Page size
 - Property - Pages 928
- PAGE.NUMBER
 - Formatting 111
- Panel 105
 - Repository 612
- Panel Fullscreen 589, 593
- Panel page
 - Adapt Page Size 930
 - Deleting by command 978
 - Dialog 975
 - Export 977
 - Repository 612
 - Zoom 933
- Panel page as dialog 975
- PANEL.PAGECOUNT
 - Formatting 111
- Paper size 925
- Parameter set 546
 - Mapping 564
- Parameter set column 162
- Parameter set columns
 - export 86
 - import 86
- Parameter set import column 167
- Parameters
 - Browse in workspace 985
 - Showing and moving 148
- partition 223, 229
- Password 76, 80
 - Data transfer 222
 - Minimum length 76
- Path to Python file 955
- PC time 911
- PDF orientation (curve window) 889
- Peakhold (curve window) 665
- Peaks (curve window) 821
- Percentage
 - Zone 642
- Period comparison (curve window) 649, 801, 806
- Periodic (curve window) 806
- Perspective
 - 3D 796
- Pictogram
 - Property - Widget 629
- PIDcontroller 448
- Place (map) 783
- Placeholder
 - CONTROL 97
 - DATABASE/DATABASES 98
 - EXPERIMENT/EXPERIMENTS 99
 - PAGE 105
 - PANEL 105
 - PROJECT/PROJECTS 105
 - SELCONTROL 107
 - SETUP 108
 - SYSTEM 109
 - VARS 110
- Placeholder Formatting
 - PAGE.NUMBER 111
 - PANEL.PAGECOUNT 111
 - SYSTEM.DATE 112
 - SYSTEM.TIME 112
 - VALUE 113
- Placeholders 95
- Placeholders for folder naming
 - SETUP.SQL 214
 - STORAGE.FOLDERNAME 214
 - STORAGE.MEASUREMENT 214
 - VARS.VALUE 214
- Places left of point (curve window) 665
- Places right of point (curve window) 665
- Platzhalter
 - MEASUREMENT 102
- Plug-in
 - Info 33
 - Version information 33
- Pointer
 - Online display 707
- Pointer diagram 699
- Polar plot
 - orientation 705
- Polardiagram 698
- Poll 381
- Polygon needle
 - Property - Widget 629
- Position
 - Property - Widget 629
 - Title 628
- Power 280
- Power measurements 275
- Power1 275, 381
- Power2 275, 382
- Power3 276, 383
- Powertrain Monitoring 492
- PPT 858
- Preferred experiment template 53

- Prepare 117
 - Presettings
 - Curve Window 809
 - Presettings (curve window) 649
 - options 885
 - Preview in Windows-Explorer 909
 - Print 925
 - Curve Window 871
 - Print Panel page 977
 - Print preview
 - Widget 938
 - Printer
 - Setup (curve window) 873
 - Printer preview 892
 - printer settings 878
 - Problems
 - Storage medium 224, 232
 - Process configuration 117
 - Process vector
 - pv-variable 531
 - Process vector variable 531
 - Create 264
 - process vector variables 183
 - GPS 533
 - Product
 - Configuration 27
 - Edition 27
 - Info 33
 - Licensing 27
 - Version information 33
 - Product configuration 27
 - Changing 27
 - Product improvement 7
 - Program
 - execute 962
 - Program add
 - Installation 36
 - Project 44, 48
 - delete 48
 - export 48
 - import 48
 - new 47
 - Project has changed 62
 - Project Menu
 - Import / Export 54
 - Manage Projects 54
 - New experiment 54
 - Open experiment 54
 - Save / Export current Data 54
 - Save as template 54
 - Save experiment (as) 54
 - Save project 54
 - Project option 61
 - Project/Experiment
 - Difference 91
 - What is saved where? 91
 - PROJECT/PROJECTS 105
 - Project-internal imc STUDIO user 76
 - Prompt to confirm discard 981
 - Properties of virtual channels
 - Comment 272
 - Delta-x 272
 - Delta-z 272
 - Name 272
 - x-offset 272
 - x-unit 272
 - y-unit 272
 - z-offset 272
 - z-unit 272
 - PROPS
 - Metadata readout 144
 - Protection against changing 935
 - Protocol display (curve window) 670
 - PulseDuration 383
 - PulseFrequency 385
 - PulsePhase 386
 - Push-button
 - Switching behavior 630
 - pv.State.ExternalPower 531
 - pv.State.SyncTimeDeviation 531
 - pv-variable 531
 - Create 264
 - Deactivating 531
 - Name 531
 - Process vector 531
 - pv-variables 183
 - Python 955
- 
- Quality Management 7
 - Quick access toolbar 68
- 
- Radius
 - Scale position 633
 - RAM buffer time 205
 - Range 190
 - Property - Widget 630
 - RangeMax 388
 - RangeMin 388
 - raw 540
 - Reaction Time 497

- ReadyForPowerOff 389
 - Realign to Grid 932
 - Receiver
 - GPS 533
 - Reconfigure 117
 - RecordEvent 390
 - RecordText 391
 - Red 392
 - Reduce logbook entries 65
 - ReduceDataRate 393
 - Refill variable 605
 - Add file 997
 - Add variable 997
 - Command 995
 - Data Browser 605
 - Import all 998
 - Show dialog 998
 - Target measurement 608, 998
 - Target variable name 607, 997
 - Treat error as warning 608, 998
 - Variable name 607, 997
 - Variables list 607, 997
 - Refresh rate
 - Property - Widget 628
 - Removable data carrier 198
 - repair 7
 - ReplaceFirstValues0 393
 - ReplaceFirstValuesN 394
 - Report Generator 871
 - Report template 938
 - Property - Pages 929
 - Report-page 925
 - Repository
 - Page 612
 - Panel 612
 - Panel page 612
 - Widget 612
 - Reprocessing signals (curve window) 821
 - ReSample 394
 - Reset
 - Set variables 999
 - Slave Pointer 676
 - Reset Slave Pointer... 861
 - Reset Window Arrangement 58
 - Restart
 - Installation 26
 - Restore
 - views 85
 - Resume data storage 117, 118, 218
 - Rezoom (curve window) 889
 - RGB image
 - Aspect ratio 747
 - Resolution 747
 - RGB-image 724
 - Ribbon
 - Customize 88
 - Rights
 - denied 83
 - disabled 83
 - enabled 83
 - full access 83
 - hidden 83
 - write protected 83
 - Ring (circular) buffer storage 197, 202
 - RMS 273, 395
 - Rosette representation
 - Wind force 706
 - Rosette1 396
 - Rosette2 397
 - Rotate
 - 3D 799
 - Rotation 461
 - Round 398
 - RS232 settings
 - GPS 534
 - RSFlipflop 398
 - Run Python code file 955
 - RunAutoBalance 399
 - RunAutoShuntCalibration 399
- 
- Sample count 186
 - SamplesGate 400
 - sampling
 - data recording 233
 - Sampling & Preprocessing (Setup page) 186
 - Save
 - experiment 54
 - project 54
 - views 85
 - Save / Export current Data 54
 - Save as 47
 - experiment 49, 54
 - Save as template 52
 - Save current data 217
 - Options 218
 - Save data 188
 - Save dialog (curve window) 870
 - Save measurement data
 - Saving subsequent to measurement 217
 - Targeted saving 217
 - Save trigger events in individual files 212

- Save/load view settings
 - additional columns 84
 - column descriptions 84
 - metadata columns 84
 - parameter set columns 84
 - table descriptions 84
 - window arrangements 84
- Saving 194
 - Cluster size 543
 - Data types 543
 - to the internal drive 198
- Saving interval 202
- Saving subsequent to measurement 217
- Sawtooth 400
- Scale angle
 - Property - Widget 633
- Scale center
 - Range 630
- Scale decimal places
 - Property - Widget 633
- Scale position
 - Property - Widget 633
- scaling
 - lines 763
 - x-axis 737
- Script name
 - Property - Pages 928
- Scroll mode (curve window) 800
 - Curve window 800
 - End online mode 800
- Search for devices 117
- Search for devices by IP/DNS 117
- Security alert 29
- Security software 18, 35
- Segment (curve window) 801
- Segmented data set 524
- SELCONTROL 107
- Select device with explorer 222, 226
- Select mode (curve window) 830
- Selected 142, 178
- Selected x-unit (curve window) 649
- Semicolon 283
- Send Curve Section to FAMOS!:context menu 818
- SendMessage 400
- Sensors
 - export 86
- Sequence table 947
- Sequencer 946
 - Start 950
 - Stop 950
- Sequencer stop
 - about commands 971
- Sequencer: Context menu 953
- Service
 - Tech support 6
- service and maintenance 7
- service check 7
- Set as default
 - Curve Window 809
- Set as preferred experiment template 53
- Set markers (curve window) 834
- Set measurement number 973
- Set the root directory 612
- Set value
 - Zone 919
- Set variables
 - Delete 999
 - Reset 999
- Settings (curve window) 885
- Setup
 - Printer (curve window) 873
- Setup data in Panel 108
- Setup data in Sequencer 108
- SETUP.SQL 108
- Shell extension 226
- Short date 911
- Short time 911
- Show Grid 932
 - Property - Pages 929
- Show message box
 - Cancel 989
 - Headline 989
 - Interval (Timeout) 989
 - Name 989
 - OK 989
 - Size 989
 - Text 989
 - Timeout-action 989
 - Voice 989
- Show most recently use values 161
- Show Panel page as dialog 975
- Show placeholder 161
- Show Start Trigger 117
- Show Tabs 933
- Show x=0 (trigger) 649
- Showing and moving columns 148
- Silent installation 37
- Simplified view 28
- Sin 401
- single value (curve window)
 - window 669
- Single value variables 259

- Single values 523
- Single-device operation (Firmware group A)
 - Mapping 564
- Single-phase power measurement 275
- SingleValueChannel 401
- Size
 - Show message box 989
- Size automatic (curve window) 665
- Size of measurement point (curve window) 889
- Skin
 - Property - Pages 928
- SkipFirstValues 404
- Slave Pointer
 - Reset 676
- Slider 918
- SlopeClip 404
- Small Ticks (curve window) 737
- Smo3 404
- Smo5 405
- smoothing (curve window) 823
- SN 179
- Snap to Grid 932
 - Property - Pages 929
- Software
 - Installation 14
 - Uninstall 14
 - Update 14
- Solid
 - Background color 928
- Sorting
 - Setup table 136
- sound output (curve window) 885, 896
 - cut sound 899
 - directly 900
 - Toolbar 897
 - volume 897
- SoundPressureLevel 405
- Source code
 - Import 972
- space curve 791
- Special characters
 - imc Inline FAMOS 262
- Specific Values 511
- Specification 179
- SpecThirds 405
- Spectra 511
- Spinner 918
- Spline (curve window) 823
- Split up pretrigger data into intervals 66
- Spread angle
 - Scale angle 633
- Sqrt 407
- SSD Storage medium 221, 225
- Standard
 - View 28
- Standard display (curve window) 649
- Start 118
 - Measurement 118
- Start angle
 - Scale angle 633
- start delay (sound) 900
- Start delay [ms] (curve window) 885
- Start measurement 117
 - Automatically upon starting imc STUDIO 31
- Start page 28
- Start Sequencer
 - Automatically upon starting imc STUDIO 31
- Start with shortcut 31
- Start: Sequencer 950
- Starting
 - imc Inline FAMOS 249
 - imc STUDIO Monitor 27
- StartTimerPeriodic 446
- StartTimerSingle 447
- Startup behavior 79
- Startup parameters 31
- State of the external power 531
- Status 185, 947
 - Property - Widget 628
- StDev 407
- Stop 118
 - Measurement 118
- Stop measurement 117
- Stop on error 947, 950
- Stop Sequencer
 - Command 971
- Stop: Sequencer 950
- StopTimer 447
- Storage
 - Memory requirements 542
 - Setup page 180
- storage device
 - formatting 223, 229
 - partition 223, 229
- Storage interval 202
 - Histogram 189
- Storage interval count 202
- Storage location (PC) 200
- Storage location measurement data 62, 213
- Storage media 221
- Storage medium
 - CF card 225

- Storage medium
 - CFast 225
 - Cluster size 222
 - Compact Flash 225
 - Data transfer 222, 226
 - ExpressCard 225
 - FAT16/FAT32 229
 - FAT32 222
 - File system 222, 229
 - Filesize (maximum) 222, 229
 - Hot-Plug 224
 - Issues 224, 232
 - Limitations 224, 232
 - microSD 221
 - Problems 224, 232
 - SSD 221, 225
 - USB 225
- Storage options 196
- Storage type 161
- Stretch
 - Background image 928
- Stretch mode
 - Curve window 800
- STri 408
- Structure 471
- StudioWDSservice 94
- Subtraction 280
- Sum 409
- Sum2 410
- Summary 193
- Supplemental file
 - Delete 985
 - Export 983
 - Import 984
- Supplemental files
 - Characteristic curves 120
 - Filter data 120
 - imc Online FAMOS source code 120
 - Messaging configurations 120
 - Synthesizer directory structures 120
- Surface
 - 3D 791
- Suspend data storage 117, 118, 218
- Swapping the storage medium 221, 224
- Switch 443, 918
 - Switching behavior 630
- Switching behavior
 - Property - Widget 630
- Symbols
 - Fixed count at line 763
 - for lines in curve window 756

- synchron (sound) 900
- Synchronization status
 - Variables 530
- Synchronous event
 - Python 955
- SyncOverload 412
- Syntax-Check 253
- Syslog 198
- SYSTEM 109
- System information
 - Variables 528
- System reboot
 - Installation 26
- System requirements
 - imc Inline FAMOS 244
 - Operating system 13
 - Windows 13
- System requirements and limitations
 - Monitor 40
- System time 911
- SYSTEM.DATE
 - Formatting 112
- SYSTEM.TIME
 - Formatting 112
- SystemClock 911

T

- Table
 - Direction - Prevent a change 917
 - Widget 916
- Table (curve window) 649
- Table description 174
 - export 173
 - import 173
- Table descriptions
 - export 86
 - import 86
 - save 84
- Table display (curve window) 670
- Tables
 - Title 918
- Tables (curve window) 670
- Tan 413
- Target measurement 608, 998
- Targeted saving 217
- Tech support 6
- Telephone numbers
 - Tech support 6
- Text 918
 - Axes 748
 - Show message box 989

- Text 918
 - Zone 919
- Text array 523
- Text as set value
 - Zone 644, 919
- Text display (curve window) 670
- Text format
 - Property - Widget 630
- Text input for report channel 910
- Text list 161
- Text Off/On
 - Property - Widget 629
- TextAdd 413
- TextFormatE 414
- TextFormatF 415
- TextFormatH 416
- TextFormatI 417
- Textual Zone
 - Zone 644
- Textual zones 919
- Text-variables 523
- Third/ octave labeling (curve window) 744
- third-octave axes
 - display options 707
- Three-phase power measurement 276
- Tick distance
 - Property - Widget 633
- Ticks (curve window)
 - small 737
- TI-float
 - pv-variable 531
- Tile
 - Background image 928
- Time Base
 - imc Inline FAMOS 262
- Time Base for Channels in a Formula 262
- Time counter
 - GPS 533
- time stamp
 - data recording 233
- Time Stamp ASCII (curve window) 670, 672
- Time zone 911
- Timeout-action
 - Show message box 989
- Timer functions 446
- Time-Shift 764, 865
 - in the ccv 885
- TimeStamp-ASCII (TSA) 523
- time-stamped numeric channels 525
- time-stamped text channels 525
- Title 153
 - Property - Pages 928
 - Property - Widget 628
- Title column 918
- Title in tables 918
- Tolerance 417
- Tool window 29, 69
 - docking 71
 - freely positioning 71
 - hiding 70
 - operation 69
 - pinning 71
 - showing 70
- Tool window chooser 70
- Toolbar
 - Sound output (curve window) 896, 897
- Toolbar (curve window) 893
- touchscreen operation
 - in curve window 754
- Traceability of data 66, 144
- Transfer function 701
- Transfer Options
 - Transfer to imc FAMOS 891
- Transfer to imc FAMOS 889
 - from curve window 871
 - Transfer Options 891
- Transfer to PC 196
- Transferring curves
 - using clipboard 856
- TransRec 417
- Treat error as warning
 - Load variable 608
 - Refill variable 608
- Trigger
 - Multiple-triggered signals 943
 - Variables 528
- Trigger events
 - in individual files 212
- Trigger time 911
- TSA 523, 525
- TSAGate 422
- TSA-Kit 672
- Twin Window (curve window) 875
- Two-phase power measurement (ARON) 275
- Two-Point controller 448
- Types of Variables 270
- Typical functionality incl. imc STUDIO Professional 21

U

- UAC 17
- Unattended installation 37

- uncertainty (display) 763
- Unchanged
 - Background image 928
- Undo (curve window) 821, 885
- UnEqual 423
- Uninstall 14
- Unit
 - Property - Widget 628
- Unlock page 932
- Unzoom (curve window) 812
- Update
 - Backup View settings 16
 - Database 16
 - Restore View settings 16
- Updating
 - Curve window 943
- Upper 423
- USB 221
 - supply of storage medium 225
- USB storage medium 225
- User
 - adding 76
 - deleting 76
 - log off 80
 - log on 80
 - removeing 76
 - switch 80
- User account control 17
- User administration 75, 76
 - activating 76
 - deactivating 76
 - log on 80
 - Startup behavior 79
 - User access rights 83
- User group 75, 76, 83
 - imc Administrators 75, 76
 - imc Advanced Users 75, 76
 - imc Developers 75, 76
 - imc Standard Users 75, 76
- User name
 - Minimum length 76
- User Ticks 750
- User type
 - Active Directory domain accounts or groups 76
 - Local computer accounts or groups 76
 - Project-internal imc STUDIO user 76
- User-defined
 - Installation type 21, 23
 - Title - Default Source 628
- User-defined buttons 58, 59
- User-defined variables 536
- UTC (IRIG Format) 911
- 
- VALUE
 - Formatting 113
- Value display 154
- Value representation
 - Property - Widget 629
- Variable
 - loading 602
 - Property - Widget 628
 - unloading 602
- Variable display format 154
- Variable linkage 939
- Variables
 - Types 270
- VARS 110
- VectorChannel 425
- VectorChannelSet 426
- VectorFromFile 426
- VectorizeAndSkip 426
- VectorizeOverlapped 427
- Vectors 523
- VectorStatic 427
- Version information 33
- Vibration 465
- VibrationFilter 429
- View
 - Complete 28
 - Standard 28
- View menu
 - Choose Tool Windows 58
 - Customize Ribbon Menu 58
 - delete view 58
 - load 58
 - Reset Window Arrangement 58
 - restore 58
 - ribbon 58
 - save view (as) 58
 - User-defined buttons 58
- View settings export/import 86
- Views 58, 84
 - delete 85
 - load 85
 - restore 85
 - save (as) 85
- Views, metacolumns, sensors, ... 86
- Virtual channels 182, 258
 - Properties 272
 - Trigger 277
- Virtual channels and the trigger machine 277

Virus scanner 18, 35, 943
VisAnyGreater 431
Visible 153
 Property - Widget 629
Visible in printout
 Property - Widget 628
VMax 432
VMaxV 432
VMean 432
VMeanV 432
VMin 433
VMinV 433
Voice
 Show message box 989
volume sound output 897
VRedV 433
VRMS 433
VSum 433
VValueAtXValue 434
VXValueOfMax 434
VXValueOfMin 434
VXValueWithYValue 434

W

Wait
 Command 971
Warning (Logbook) 71
Warranty 7
WatchDog 94
Waterfall diagram (3D) 649
 display 659
Waterfalls 649
WAVE_OverloadVariable 474
Waveform Preselection
 More Channels in Curve Window 731
Waveforms in axes list
 More Channels in Curve Window 732
WFT Assistant 516
While 445
Widget
 DIO 921
 Repository 612
 Table 916
Widget linkage 939
Widgets 596, 910
 Aligning 624
 Arrangement 623, 624
 Cascade 625
 Dock 626
 Grouping 627
 insert 617

 new 617
 Realign to Grid 932
 Snap to Grid 932
Width
 Property - Pages 928
Wind force
 Rosette 706
Window arrangements
 export 86
 import 86
 load 84
 save 84
Window size
 Panel 930
Windows
 Firewall 29
 Security Alert 29
 Shell extension 226
 System requirements 13
 User account control 17
Windows-Explorer
 Preview 909
WindRoseCorr 435
WORD 858
Write protected
 Property - Widget 628
Write-protected 153

X

X-axis 187
 automatic 737
 changes 889
 fixed range 737
 lin/log 737
 markings 737
 rounding 737
 scaling 737
 Scaling the Axes 736
X-link with curve windows 862
x-offset 272
x-unit 272
XY data
 complex data 714
 different sampling rates 714
 Lissajous figures 714
 measurement cursors 714
 polar plots 714
XY-data sets 523, 525

Y

Y-axes stacked 649
Y-axis

Y-axis

- color 750
- line thickness 750
- small ticks 750
- y-axis max 190
- y-axis min 190
- y-axis range 190
- Y-connection 276
- y-unit 272

Z

- Z-axis 659, 697
- Zip - Logbook 73
- ZIP format
 - Limitations 540
- z-offset 272
- Zone
 - Advanced display 641
- Zone dialog 634
- Zone representation
 - Property - Widget 630
- Zone ring
 - Zone 645
- Zones
 - Property - Widget 630
- Zoom
 - Background image 928
- Zoom (curve window) 811, 889
- Zooming Panel pages 933
- z-unit 272



An Axiometrix Solutions Brand

Contact imc

Address

imc Test & Measurement GmbH
Voltastr. 5
13355 Berlin

Phone: (Germany): +49 30 467090-0

E-Mail: info@imc-tm.de

Internet: <https://www.imc-tm.com>

Tech support

If you have problems or questions, please contact our tech support:

Phone: (Germany): +49 30 467090-26

E-Mail: hotline@imc-tm.de

Internet: <https://www.imc-tm.com/service-training/>

Service and maintenance

Our service team is at your disposal for service and maintenance inquiries:

Phone: (Germany): +49 30 629396-333

E-Mail: service@imc-tm.de

Internet: <https://www.imc-tm.com/service>

imc ACADEMY - Training center

The safe handling of measurement devices requires a good knowledge of the system. At our training center, experienced specialists are here to share their knowledge.

E-Mail: schulung@imc-tm.de

Internet: <https://www.imc-tm.com/service-training/imc-academy>

International partners

You will find the contact person responsible for you in our overview list of imc partners:

Internet: <https://www.imc-tm.com/imc-worldwide/>

imc @ Social Media

<https://www.facebook.com/imcTestMeasurement>

<https://www.youtube.com/c/imcTestMeasurementGmbH>

https://x.com/imc_de

<https://www.linkedin.com/company/imc-test-&-measurement-gmbh>