

imc WAVE 2023

Manual

Doc. Rev.: 6.5 - 2024-02-29



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

© 2024 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 WAVE is available both as an e-book and in PDF format. 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 WAVE 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 WAVE 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 WAVE 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	7
1.1 Before you start	7
1.2 Tech support	7
1.3 Service and maintenance	8
1.4 Legal notices	8
1.5 imc Software License Agreement	9
2 Chapter overview	12
3 Setting Up - Software	13
3.1 System requirements	14
3.2 Installation - Preparation	16
3.3 Installation - Step by step	20
3.4 Product configuration / Licensing	28
3.5 Start	28
3.6 Info / Version information	33
3.7 Information and tips	34
4 Setting Up - Connect the device	39
4.1 Connecting via LAN in three steps	40
4.2 Special options for connecting to the device	42
4.3 Information and tips	55
4.4 The network	56
4.5 Firmware version	57
5 Documentation of devices - Help and Documentation	64
6 Demo Experiments	65
7 imc WAVE	66
7.1 Architecture	66
7.2 "What is..." - Terminology	67
7.3 Experiments, projects and the database	70
7.4 The User Interface	76
7.5 Spectrum Analyzer	94
7.6 Structural Analyzer	132
8 Setup - Device (general)	145
8.1 Device overview	146
8.2 Actions and Buttons	147
8.3 Operation	161
8.4 Performing a measurement: Procedure	171
8.5 Metadata and Comments on the measurement	173
8.6 Information and tips	178
8.7 Troubleshooting	182
9 Setup pages - Configuring device	184

9.1 Documentation (experiment description)	184
9.2 Configuring devices	185
9.3 Configuring channels and variables	237
9.4 Channel balance	275
9.5 Trigger and events	282
9.6 Storage options and directory structure	306
9.7 Storage media in the device	337
9.8 Measurement types	349
9.9 Fieldbusses	368
9.10 Special modules and Third party devices	460
9.11 Sensors, characteristic curves and TEDS - for imc DEVICES devices	465
9.12 Information and tips	471
10 Setup - Advanced device functions	490
11 Data analysis and signal processing	491
11.1 Overview of the tools	491
11.2 imc Online FAMOS	498
11.3 imc FAMOS dialog	721
12 Channel types, variable types, data types and data storage formats	730
12.1 Data types	730
12.2 Various variable types - Channel type	734
12.3 Data file format - Saving measured data and export/import formats	752
12.4 imc Data format	785
12.5 imc Format Converter	786
13 Panel	800
13.1 Tool windows	801
13.2 Context menu	817
13.3 Design mode	820
13.4 Widgets - operation and properties	820
13.5 Curve window	850
13.6 Special Widgets	1106
13.7 Pages	1120
13.8 Variable linkage	1133
13.9 Information and tips	1137
14 Sequencer, events and commands	1140
14.1 Sequence table	1141
14.2 Event Dialog	1142
14.3 Running and stopping	1144
14.4 Creating a sequence of commands	1145
14.5 Context menu	1147
14.6 Information and tips	1147
14.7 Command reference	1148
15 imc Format Converter	1200

- 15.1 Installation 1201
- 15.2 Settings 1202
- 15.3 Format converter as standalone-program 1207
- 15.4 Command line parameters 1209
- 15.5 Conversion via the Windows Explorer 1210
- 15.6 Format Converter as imc STUDIO command 1211
- 15.7 Converting via the Data saving assistant 1213
- 16 Shared Components 1214**
 - 16.1 Logbook 1214
 - 16.2 Placeholders 1217
 - 16.3 Options 1237
- 17 Miscellaneous 1242**
 - 17.1 Last Changes 1242
- Index 1254**

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](#) ^[7].
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](#) ^[9]. 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.

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](#) ^[7] will be happy to receive your feedback.

1.3 Service and maintenance

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

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



Management System
ISO 9001:2015
ISO 14001:2015

www.tuv.com
ID 0910085152

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 18, 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 carriers 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 carriers 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 inoperable one.

2 Chapter overview

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

Section	Synopsis
General introduction	Quality Management ^[8] , Tech support ^[7] , License agreement ^[9]
Setting Up ^[13]	Software setup operations <ul style="list-style-type: none"> • System requirements ^[14], Installation ^[16], Start ^[28]
 Setting Up - Connect the device ^[39]	Connecting via LAN ^[40] and Optimize the network ^[56]
imc WAVE ^[66]	NVH analysis
 Setup - Device (general) ^[145]	The device configuration <ul style="list-style-type: none"> • Performing a measurement ^[171] and Devices supported (Device Overview) ^[146]
 Setup pages - Configuring device ^[184]	The device configuration <ul style="list-style-type: none"> • Configuring device (Setup pages) ^[184] • Synchronization ^[188], Storage options ^[306]
 Setup - Advanced device functions ^[490]	Additional functions of the devices of firmware group A (imc DEVICES); including: <ul style="list-style-type: none"> • Fieldbuses ^[368]
 Data analysis and signal processing ^[491]	Various possibilities for processing and analysis of measured data <ul style="list-style-type: none"> • during a running measurement on the device: imc Online FAMOS ^[498] (function reference ^[542]) • subsequent analysis (post-processing): imc FAMOS ^[721]
 Channel types, variable types, data types and data storage formats ^[730]	In imc WAVE, 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. <ul style="list-style-type: none"> • Data types ^[730] • Various variable types - Channel type ^[734] such as User-defined variables ^[749] • Data file format - Saving measured data and export/import formats ^[752] such as Parameter set ^[759]
 Panel ^[800]	Display of measured data and operation via the user interface <ul style="list-style-type: none"> • Curve Window ^[850] and Widgets instruments ^[820] • Variable linkage ^[1133] and Data Browser ^[804]
 Sequencer, events and commands ^[1140]	Creating automated routines <ul style="list-style-type: none"> • Sequence table ^[1141] and Creating a sequence of commands ^[1145] • Command reference ^[1148]
Shared Components ^[1214]	Placeholder ^[1217] , Curve window ^[850]

Tech support

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

Questions or problems? Contact our [Tech support](#) ^[7].

3 Setting Up - Software

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

imc WAVE 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	<ul style="list-style-type: none"> • System requirements ¹⁴ • Installation / Uninstall ¹⁶ • Product Configuration / Licensing ²⁸
The first start, important settings	<ul style="list-style-type: none"> • The first start ²⁸ • Device connection / Network / Firewall ³⁰
List of components used and their versions	<ul style="list-style-type: none"> • 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: 1280 x 1024 (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.

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 WAVE setup if they are not already present:

Component	Version	Folder in "System" directory
Microsoft .NET Framework	4.8	DotNetFx4.8
Microsoft VC 2015-2019	14.28.29910	Microsoft Visual C++ Redistributable\2019
Microsoft VC 2010	10.0.402219.1	Microsoft Visual C++ Redistributable\2010
Microsoft VC 2005	6.0.3790.0	Microsoft Visual C++ Redistributable\2005
Microsoft Build Tools 2015	14.0.23107.10	MSBuildTools2015

Supported Measurement Devices

Which devices you can use in imc WAVE is described in the documentation on the "Setup" > "[Device Overview](#)"¹⁴⁶, as well as in the "Technical Data Sheet".

The imc WAVE analyzers can be used with any measurement systems belonging to the imc CRONOS device families or imc C-SERIES. In particular, the devices must be equipped with appropriate measurement amplifier modules for microphones/accelerometers (e.g. IEPE/ICP). Suitable module types: ICPU2-8, AUDIO2-4, QI-4 etc.

The device series imc CRONOS*flex* (CRFX) and imc CRONOS-XT (CRXT) are especially suitable and recommended. These device series digitize the captured signals at a resolution of 24 bits and are therefore a perfect fit for the high dynamic requirements in NVH testing.

Device groups

Firmware imc DEVICES - **firmware group A** (A4-A7)

- imc CRONOS*compact*
- imc CRONOS*flex*
- imc CRONOS-SL-N
- imc C-SERIES with sn. 14xxxx and higher
- imc CRONOS-XT
- imc BUSDAQ*flex*⁽¹⁾
- imc BUSDAQ⁽¹⁾ with sn. 13xxxx and higher
- imc SPARTAN⁽¹⁾ with sn. 13xxxx and higher
- other customized devices with sn. 13xxxx and higher

Firmware imc DEVICEScore - **firmware group B** (B10-B11)

- imc EOS
- imc ARGUS*fit*

¹: For additional measurement signals such as Fieldbusses (CAN), temperatures, strain gauges, etc.. Not suitable as audio inputs to the analysers.

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](#) ²⁸).

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](#) ¹⁹.

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 WAVE) 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. 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 WAVE 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](#) ¹⁸.



Reference

Update imc STUDIO

When using imc STUDIO, please observe the relevant notes about updating:

- Update with the help of the existing database,
- Update without using the existing database.

The subsequent procedure - an overview

Follow the installation program instructions (see: "[Installation step by step](#)" ²⁰).

- The installation process first checks whether the **required system software** is installed. If one 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 WAVE is ready for use.

Uninstalling

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



Reference

See also

- [Unattended Installation - Silent Installation](#) ³⁷
- [Installation - Installing projects](#) ³⁶
- [Recommended Virus Scanner Settings](#) ³⁵
- [Changing Languages and Installing Additional Languages](#) ³⁴

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 all experiments.

If you wish to run multiple versions of imc WAVE 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 WAVE database by both versions in parallel.

- If the same path is selected in the new imc WAVE 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

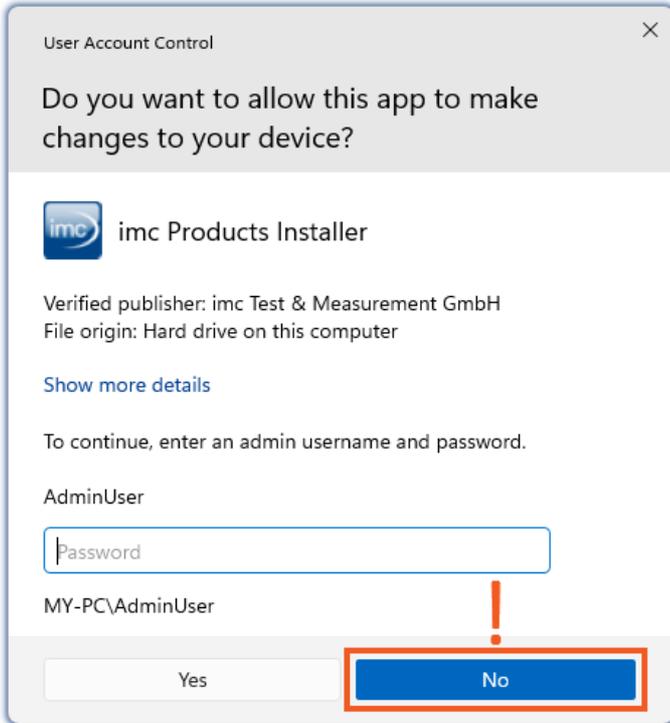
Using new pages

- Be aware that the new version provides new functions, such as new or enhanced pages and new menu items.
 - **Use of the new pages 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. You can re-insert these columns at the desired position (by means of the Column Chooser).
-

3.2.2 Notes and troubleshooting

Notes on Windows User Account Control

Do not change the user account

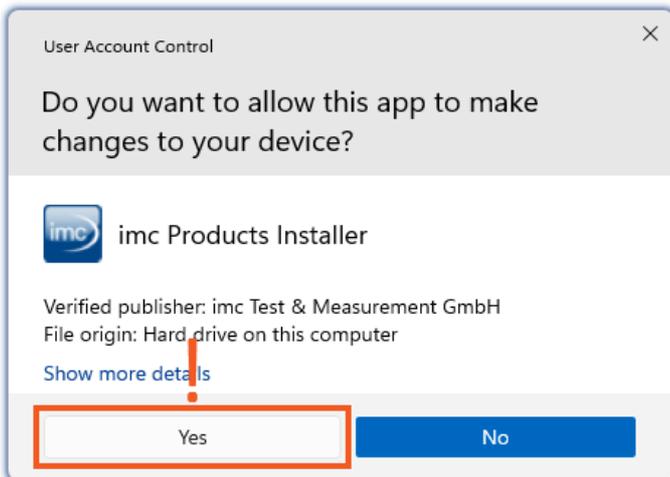


Example of a password request.
Select "No".

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 WAVE. For an exact listing of all of imc WAVE's functions, see the Technical Specs sheet.

If you activate components which are not available in imc WAVE, 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 WAVE*". Both a complete and a reduced installation package are offered. The reduced installation package is sufficient if you have only purchased imc WAVE. Which products are included is indicated next to the Download button.

Start the download first, and next the file (e.g. "*Installer_imc_STUDIO_2023_R4.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.

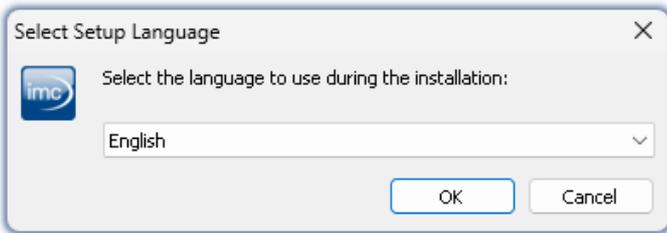


Hinweis

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.

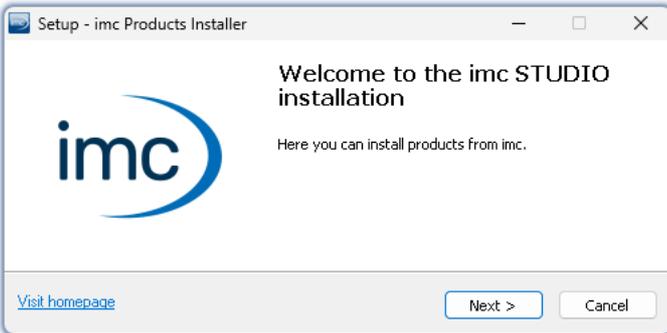
Installation language settings



Selecting the language during installation

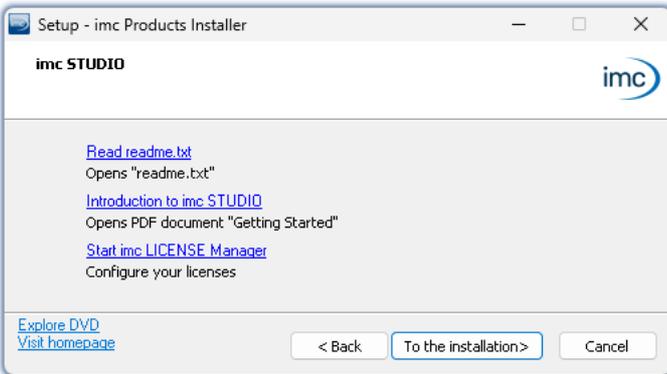
After starting installation, a window appears in which to select the installation language.

Performing installation



Installation setup welcome page

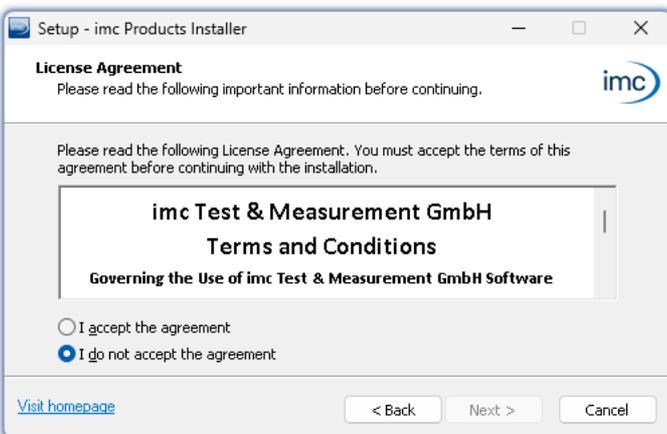
The installation setup then starts in the language selected.



Before starting installation

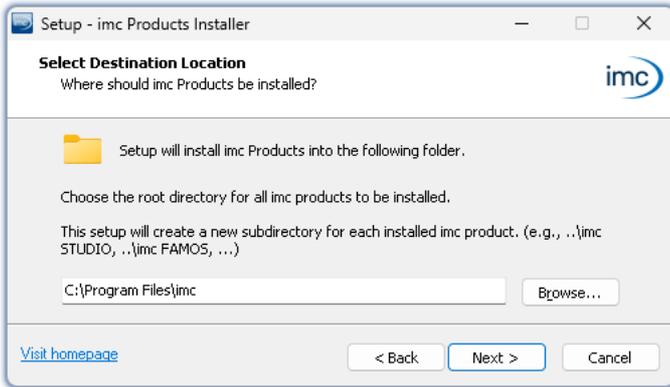
On the installation setup's next page, you are offered the following choices:

- 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 2023). 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*"²⁴

"*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*"²⁴ installation type.

Installation type	Beschreibung
Demo	Use the installation type " <i>Demo</i> " 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.
Typical	Use the installation type " <i>Typical</i> " 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 " <i>User-defined</i> " 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 WAVE, you must choose the installation type: "*User-defined*"²⁴ or "*Demo*"!

Note

imc WAVE reconfigure afterwards

Regardless of which variant of the installation is selected, when imc WAVE is installed, all components are included. This way, after successful installation it is possible to adapt the imc WAVE configuration at any time by means of the "*product configuration*"²⁸.

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 (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 (imc DEVICES - e.g. CRONOS family, C-SERIES, SPARTAN, BUSDAQ).
Firmware and driver package imc DEVICEcore	●	●	○	Package of device drivers and firmware required by imc STUDIO and imc WAVE for devices of the firmware group B (imc DEVICEcore - 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.

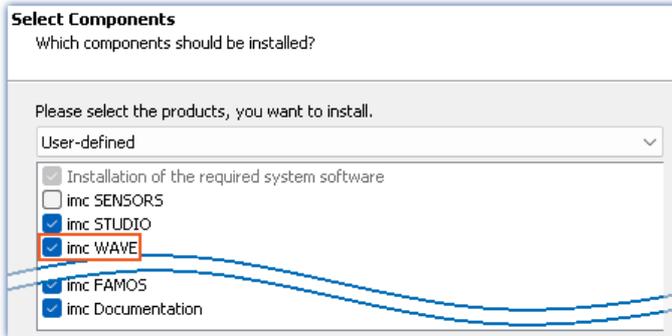
- 1: imc SENSORS requires a password for installation.
- 2: The associated Demo license must be activated after concluding installation by means of imc LICENSE Manager, otherwise it is not possible to start the product. Once the demo trial period has elapsed, use the [product configuration](#) to convert 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](#)"). By contrast, if you select "User-defined", the installation is first [configured](#).

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



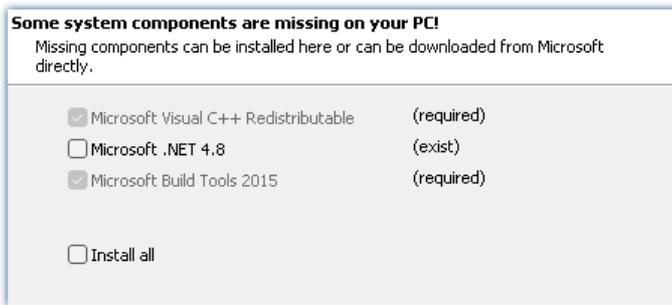
Selecting User-defined installation

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 WAVE here!

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

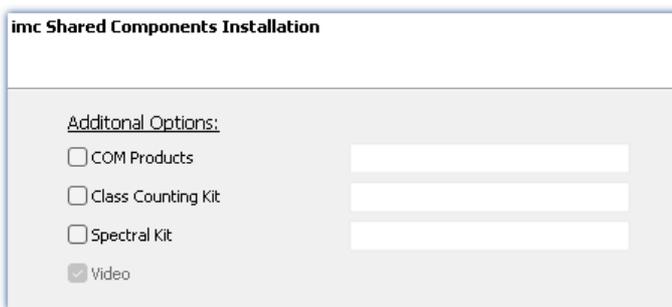
Required system components



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

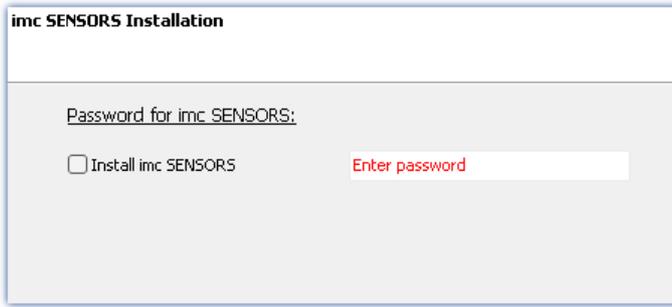


Configuring imc Shared Components

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

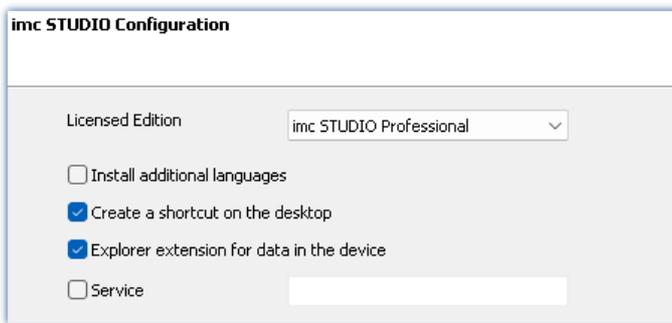
Password for imc SENSORS



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](#) ^[28] 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	<p>With this option, imc WAVE is installed in all available languages. (This will substantially prolong the installation procedure).</p> <p>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.</p> <p>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]).</p>
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 ^[146] (imc DEVICES)
Service	Service support. Install this component only if requested by our tech support. for devices of the firmware group A ^[146] (imc DEVICES) and group B ^[146] (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.



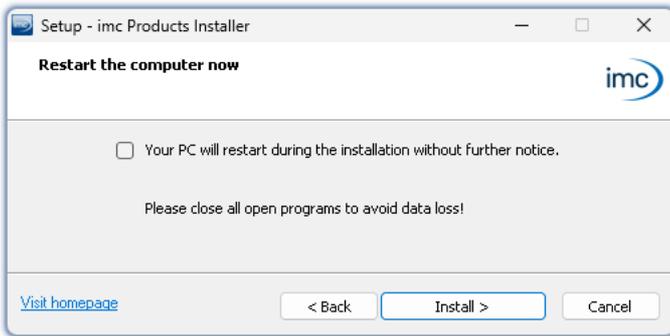
Reference

[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

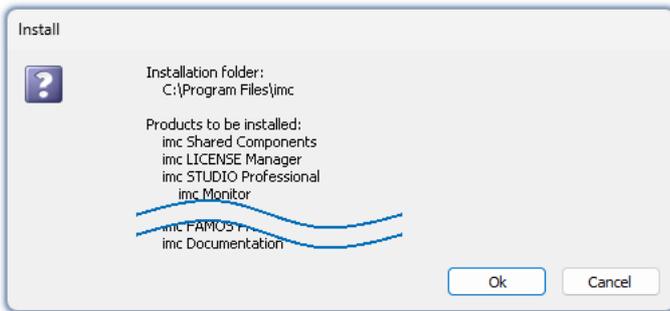
Before the installation, a **system reboot** is performed. For this reason, first **close any running programs** before continuing!

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*".

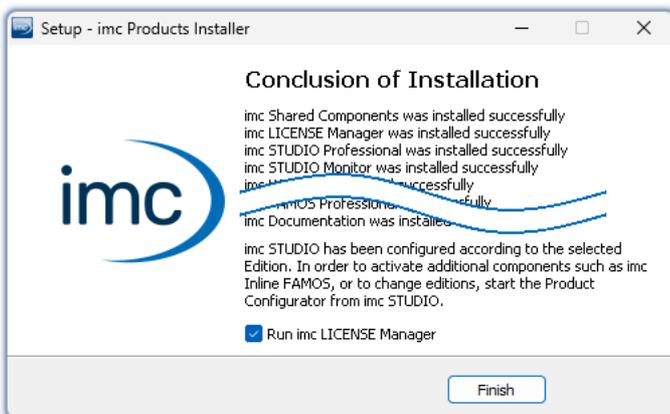
The figure shows an example; the actual dialog may appear differently depending on your respective previous selections.

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 WAVE 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 WAVE is a preconfigured product. The product configuration is set permanently to ensure optimum functioning. In **accordance with your respective license**, you may be able to **activate the analyzers**. Upon first starting, you are automatically guided to the product configurator by a dialog.

In order to **modify the configuration** at a later time, expand the title bar , open the context menu and click in the menu on the symbol .

The product configuration assistant now opens. You are now able to change the configuration. Follow the instructions and close the assistant.

After completing the configuration, you must close and restart imc WAVE 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](#) ²⁰.

imc WAVE 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](#) ²⁸ is missing, the Product configuration starts automatic.

Before you can use imc WAVE, you need to perform a one-time configuration of the software. The various license-based analyzers must be activated via the product configurator. Only after this step, imc WAVE can be started correctly.

Splash screen

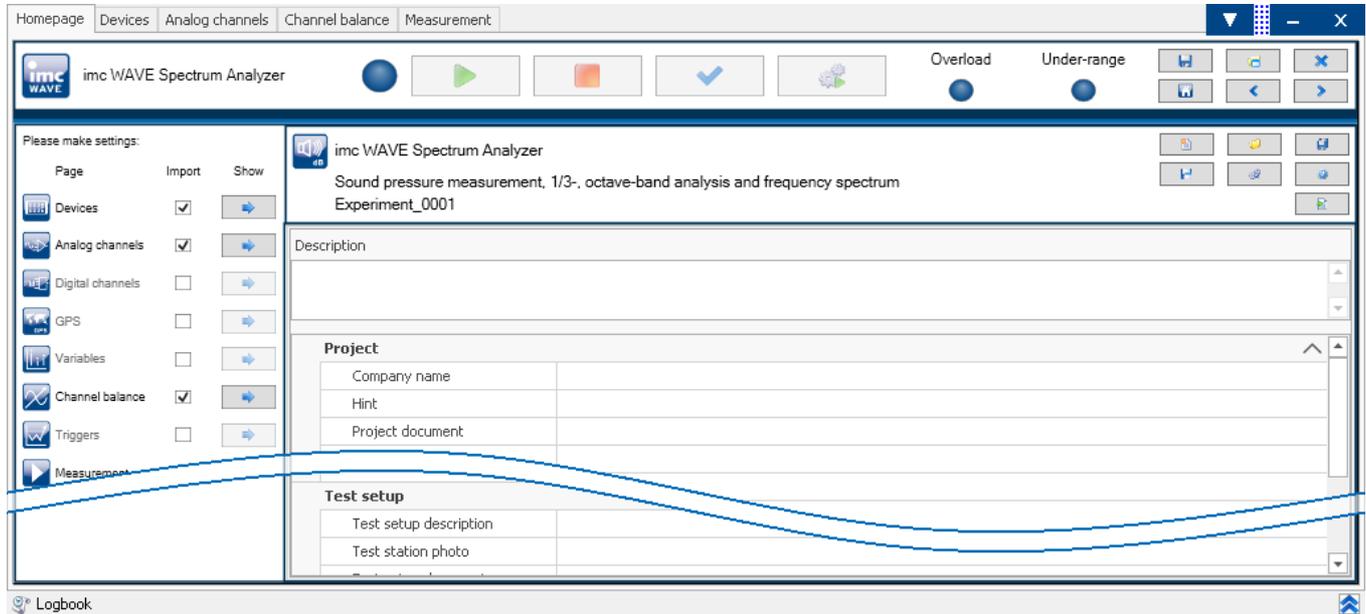


Next, a splash screen appears, indicating that software components are being loaded. Depending on the installation and the PC capacity, this could take some time.

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. After the first installation, the "**Homepage**" may appear as shown in the example below:



Start page (example Spectrum analyzer)

Changing analyzers

imc WAVE always starts in one of the available analyzers. The Homepage displays which analyzer is in effect. If you do not have a license for the started analyzer you will be notified via dialog.

To change analyzers, create an experiment in a different analyzer (📁), or open an existing experiment (📁).

3.5.1 Device Connection / Network / Firewall

To use an imc measurement device, imc WAVE must set up a **connection via a network** (mostly LAN). Details on the network settings are presented in the chapter "[Setting Up - Connect the device](#)³⁹".

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 Important settings

Options

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

You can change the folder path in the [options](#)¹²³⁷ (under "Project Management" > "HDD settings"):

Action	Page
Options (🔧)	Homepage

Note

Reading and writing access rights are required

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

Checking and modifying the time zone

Applicable to devices belonging to the [firmware group A](#)¹⁴⁶ (imc DEVICES): In order to assign a definitive point in time for all measured data, a specific time zone is set for all imc devices used. The factory setting is "**UTC+01:00**". If applicable, modify the time zone setting. For this purpose, the "[Devices properties](#)¹⁵³" dialog contains an entry "Device time and synchronization" > "Time zone".

Action	Page
Dialog: Device options > Device Properties	Devices

3.5.3 Command line parameters

Opening an Experiment file

An [experiment file](#) ⁷⁰ (e.g. MyExperiment.imcstudio) can be opened directly in the Windows Explorer by double-clicking. This starts a new instance of imc WAVE 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)
<p>In order to open an experiment file with a particular installation of imc STUDIO (imc WAVE), extend the link with the respective installation's name. For example:</p> <pre>"C:\Program Files\imc\imc STUDIO 2023\imc.WAVE.exe" "imcDB://DB_WAVE\SpectrumAnalyzer\MyExperiment"</pre> <p>The experiment file <code>MyExperiment.imcstudio</code> is started in the imc STUDIO installation (imc WAVE).</p>	

"Experiment file"	Experiment to be loaded
<p>Mandatory entry. Opens the experiment entered with the imc STUDIO installation (imc WAVE) 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.</p> <p>One possibility is to specify the path in absolute terms:</p> <pre>"C: \Users\Public\Documents\DB_WAVE\SpectrumAnalyzer\MyExperiment\config\MyExperiment.imcStudio"</pre> <p>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):</p> <pre>"imcDB://DB_WAVE\SpectrumAnalyzer\MyExperiment"</pre>	

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

3.5.4 Error causes upon starting

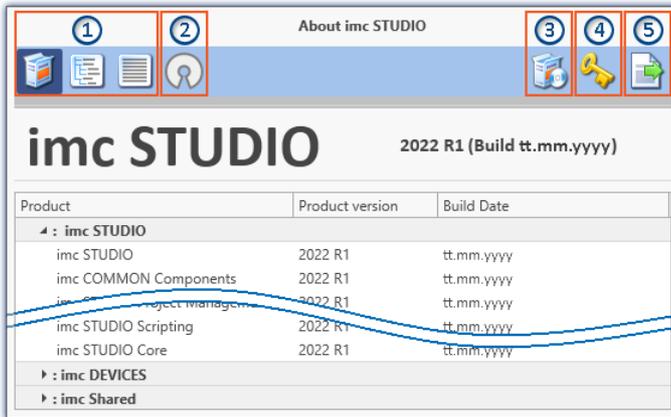
Corrective actions for when the logbook opens with error messages directly upon starting imc WAVE.

Message	Corrective action
<p>No matching DEVICecore found.</p>	<p>In the product configuration, the firmware imc DEVICecore is activated. However, no appropriate firmware has been found or installed.</p> <p>By default all firmware products are activated. You can take the following corrective actions:</p> <ul style="list-style-type: none"> install the appropriate imc DEVICecore-firmware. This is only necessary if you wish to use a device belonging to this firmware group. deactivate the component "<i>imc DEVICecore device adapter</i>" in the product configurator ²⁸.

Message	Corrective action
Error while initializing the plug-in "imcDevices V2.x Adapter" ...	<p>In the product configuration, the firmware imc DEVICES is activated. However, no appropriate firmware has been found or installed.</p> <p>By default all firmware products are activated. You can take the following corrective actions:</p> <ul style="list-style-type: none"> • 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 "<i>imc DEVICES V2.x Adapter</i>" in the product configurator ²⁸.
Could not load file or assembly 'imc.Studio.Plugins.DevSetup.Sensors.Stubs.dll' ...	<p>In the product configuration, the component imc STUDIO Sensors is activated. However, no appropriate imc DEVICES firmware was found or installed.</p> <p>You can take the following corrective actions:</p> <ul style="list-style-type: none"> • 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 "<i>imc STUDIO Sensors</i>" in the product configurator ²⁸.
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.	<p>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.</p> <p>Make sure that no older version of imc WAVE needs to access the database.</p>
More than one DEVICEcore installation found in STUDIO installation directory.	Solution: Uninstall the older imc DEVICEcore version.
Could not connect to the Guardian process. Some functions (e.g. delete measurements) will not work.	<p>The Guardian process is not running. Please start either the Guardian- or WatchDog-service.</p> <hr/> <p> For more info, see the section "<i>Guardian</i>"</p> <hr/>

3.6 Info / Version information

imc WAVE 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, expand the toolbar  and open a context menu out of the toolbar. Click in the menu 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](#)  ③, start the [imc LICENSE Manager](#)  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 chapter "[Tech support](#)" ).

3.7 Information and tips

3.7.1 Changing languages and installing additional languages

Following installation, imc WAVE 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](#)²⁴".



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](#)"²⁰

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" the de-installation of all previous versions is inhibited.

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 Setting Up - Connect the device

There are multiple ways to **connect the imc measurement devices with the PC**. In most cases, the **connection via LAN** (local area network, Ethernet) is implemented. See section "[Connecting via LAN in three steps](#)"^[40] for the **quickest way to connect** PC and measurement device.

But there are also other connection types:

- [WLAN](#)^[53]
- LTE, 4G, etc. (via appropriate routers)

These are described in a separate section: "[Special options for connecting to the device](#)"^[42].

The devices use the **TCP/IP protocol** exclusively. With this protocol, some settings and adaptations for your local network may be necessary. For this purpose, the support of your network administrator may be necessary.



Note

When **using multiple TCP/IP connections**, e.g. LAN, WLAN and modem, observe the remarks in the section "[Computers with multiple TCP/IP connections](#)"^[45].

Recommended network configuration

The latest and high-performance network technologies should be used to achieve the maximum transfer bandwidth. This means especially 1000BASE-T (GBit Ethernet). GBit Ethernet network devices (switches) are downward compatible, so that imc devices that only support 100 MBit Fast Ethernet can also be operated on them.

The cable length between the switch and a PC or a device should be less 100 m. Use a shielded cable. If the length of 100 m is exceeded, then you have to insert another switch.

If the system is being integrated into an existing network, you must ensure that the minimum data rate can be guaranteed. Under some circumstances, this may require using switches to subdivide the network into separate segments in order to govern the data traffic in a targeted way and thus optimize the data rate.

In very demanding applications, you might consider grouping multiple GBit Ethernet devices via even higher-performance sections lines of the network (e.g. via 5 GBit Ethernet) and to connect these groups to NAS-components, for instance, via these lines.

When such imc devices are included which use network-based PTP-synchronization (e.g. CRXT or CRFX-2000GP), then it is necessary to use network switches which fully support this protocol on the hardware side. Appropriate network components are also available as imc accessories (e.g. CRFX/NET-SWITCH-5) and are then electrically and mechanically fully compatible with the imc systems.

4.1 Connecting via LAN in three steps

The most common case is described below: the PC and the device are connected via cable or network switch. The device's IP address must be set in the PC's address range. Subsequently, the device can be connected with the PC. If a connection has ever been established previously, the software recognizes the device's hardware configuration. In that case, experiment configurations can be prepared without any connection to the device.

Step 1: Connecting the measurement device

To connect via LAN there are two options:

1. The measurement device is connected to an **existing network**, e.g. via network switch. Only with a switch is it possible to run multiple devices.
2. The measurement device is connected directly to a network adapter on the PC (**point-to-point**).

In a LAN, the first case is typically implemented. Modern PCs and network switches are usually equipped with Auto-MDI(X) automatic crossover recognition, so that it is not necessary to distinguish between crossed and uncrossed connection cables. Thus both cable types can be used.

Step 2: IP-configuration

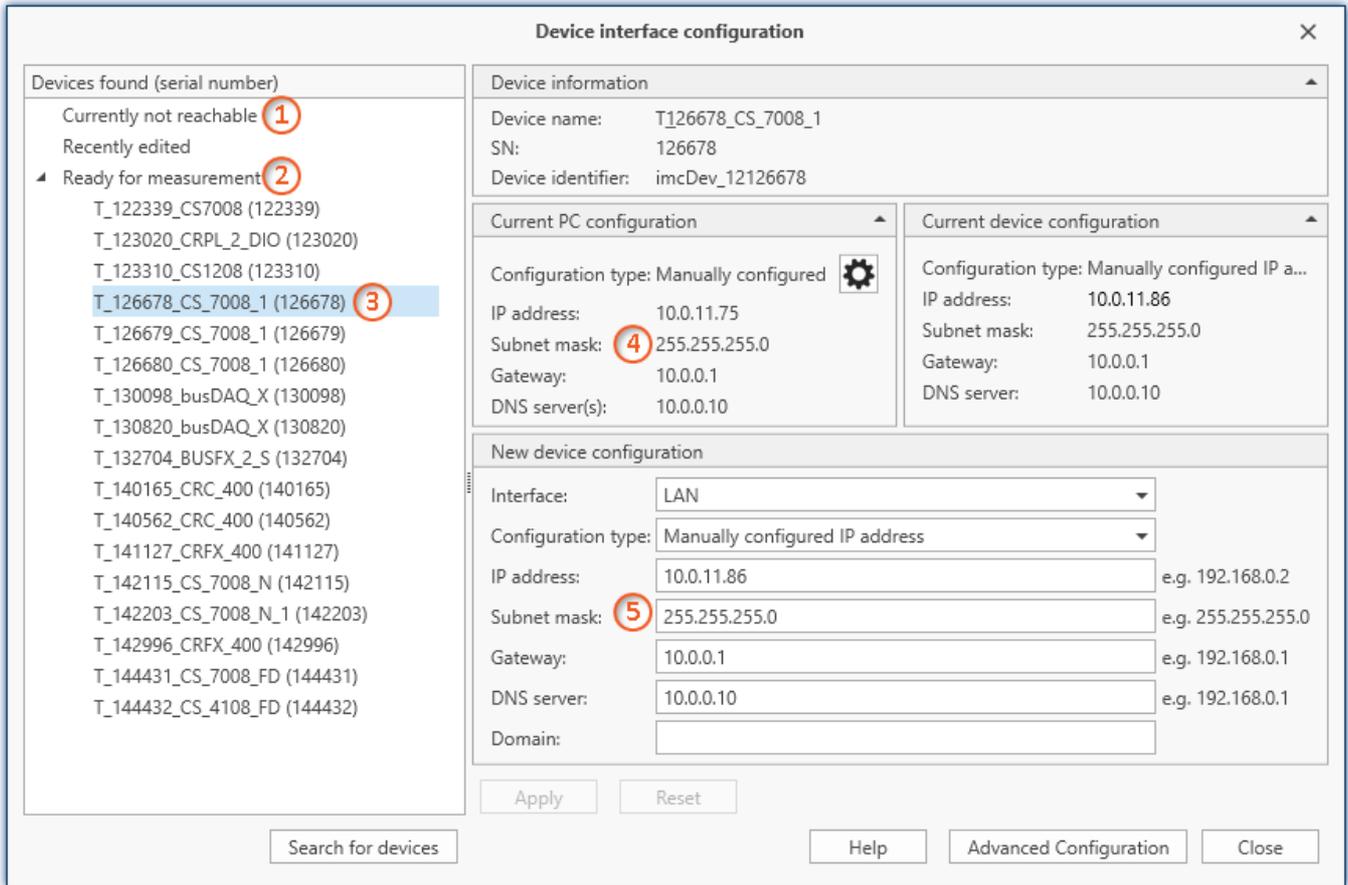
Start imc WAVE. Click the "*Device interfaces*" button to open the dialog for configuring the IP address of the device.

Action	Page
Device options > Device interfaces	Devices

Once the dialog starts, the system automatically searches for all devices in the network. In the tree diagram, all available devices are indicated. If the device appears among the group "Currently not reachable" ①, it is necessary to modify the device's LAN-settings. If the device appears among the group "Ready for measurement" ②, you can leave the settings as they are or review them.

Select the device for making modifications ③.

If there is any IP-conflict, devices affected will not be listed.



Display of measurement devices found and of the IP address

Set the **IP address manually** if you are not using DHCP. The device's IP address ⑤ must match with the PC's address ④. To conform to the network mask, only the device portion may be different (see example).

Example

In the example shown, the fixed IP 10.0.11.75 with subnet mask 255.255.255.0 is selected for the PC. For measurement devices, any numbers would be suitable which begin with 10.0.11. and then do not contain 0, 75, or 255. The 0 and the 255 should not be used, if possible, due to their special significance. The 75 is the computer's number.

Example for IP settings	PC	Device
IP address	10 . 0 . 11 . 75	10 . 0 . 11 . 86
Network mask	255 . 255 . 255 . 0	255 . 255 . 255 . 0

If the configuration type: "DHCP" is used, **the IP address is obtained automatically** from the DHCP-server. If it is **impossible to obtain any setting values** via DHCP, the **alternative values are used**. These could lead to errors in the connection (different networks, same IP addresses, etc.).

If there is a **direct connection** between the device and the PC by a cable, then **DHCP should not be used**.

In order to apply the changes, click on the button "Apply". Wait for the device to restart and then close the dialog.



Note

Connection via modem or WLAN

For devices of the firmware group A (imc DEVICES):

If the connection to the device is established via a modem or WLAN, start the program "*imc DEVICES Interface Configuration*" by clicking on the button: "*Advanced Configuration*" (see previous figure). An exact description is found in the software manual chapter: "*Setting Up - Connect the device*" > "[Special options for connecting to the device](#)"^[42].

Step 3: Integrating a device into an experiment

Now you are ready to add the device to the imc WAVE experiment. If your device is unknown to the system, first perform the "[device search](#)"^[151].

Action	Page
Dialog: Device options > Search for devices	Devices

Select the desired device: Once you click in the checkbox "*Selected*" for the desired device, it is ready to use in the experiment (see [Device Search - Known and Selected](#)"^[171]).

Selected	Device name	SN	Device specification
<input checked="" type="checkbox"/>	T_124835_C1_1_LEMO_ET	124835	imc C1-1 LEMO
<input type="checkbox"/>	T_130039_busDAQ_X	130039	busDAQ-X
<input type="checkbox"/>	T_130311_SPARTAN_U32_CAN	130311	imc SPARTAN

You can also select multiple devices for your experiment.

Now the device is "*known*". After the next program start it is available for selection. For further information, see the documentation on the component "[Setup](#)"^[145].



Reference

Time zone

Now check whether the correct time zone is set for the device: see "[Important settings](#)"^[30].

4.2 Special options for connecting to the device

Additional ways to connect with the device are presented below. For other special cases, please see the section on LAN connection.

4.2.1 Devices and TCP/IP

General preparations

A imc device with TCP/IP supports a classic network interface (LAN).

For a device having a LAN-interface, the following parameters must be set appropriately:

- IP address
- Subnet mask
- DNS-server-address (optional)
- Domain (optional)
- Router-address (optional)

If your network is equipped with a DHCP-Server, these data can be accessed automatically by the device upon switch-on. Have your network administrator make the necessary configuration of the DHCP-server and the DNS-server. You can skip the rest of this section, in that case, and proceed with [Configuration of the PC](#)⁴⁴.



Note

Notes for the Administrator

- We recommend using a fixed assignment of IP addresses to the devices and to set an unlimited lease-time. A device's host name cannot be changed. It is comprised of a prefix and the device serial number, e.g. " imcDev__99030143".
For a device with a PPP-interface, it makes sense to set the following parameters:
- local and remote IP address,
- Baud rate and protocol of the serial interface to modem or cable,
- DNS-server address (optional),
- user name and password (optional),
- scripts for connecting and disconnecting (optional).

If there is no DHCP-server, the necessary device settings must be made as described further below. Have your administrator issue IP addresses for your device units and make note of the address assignments. Have the administrator enter the device units in the DNS-server if there is one. In this case, also make note of the DNS-server's IP address and the name of your network domain.

Find out the subnets in which the devices are to be operated and make note of the corresponding subnet masks. You will also need the IP addresses of any routers present which your devices units are to use.

Once you have gathered all the above information, you can proceed with the next section.

4.2.1.1 Configuration of the PC

To operate TCP/IP devices you need a PC on which the TCP/IP-protocol is installed and ready. We assume here that your PC is already set up appropriately. If this isn't the case, follow the instructions in your network card's documentation and in the WINDOWS help texts in order to install TCP/IP on your PC.

Avoid changing your PC's IP address if the PC is on an existing network! Simply adjust the configuration of the measurement devices!

Firewall and Ports

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.

Note

Alternatively, you can enable the ports. To do this, each port must be entered for UDP and TCP. The following ports are used:

- UDP 1200
- TCP 1200 through 1202

4.2.1.2 Use of TCP/IP devices in various network situations

In some cases it's necessary to be able to run the PC together with the measurement device both in a network (e.g. company network) and, for mobile applications, with a point-to-point connection. In order to enable convenient and reliable operation in both settings, the following manner of proceeding is recommended.

In larger networks, the IP addresses for the network's PCs are usually assigned automatically ("*Network settings*" > "*Internet protocol (TCP/IP)*" > "*Properties*" > "*Obtain IP address automatically*"). For that purpose, the network comes with a DHCP-Server (DHCP: **D**ynamic **H**ost **C**onfiguration **P**rotocol).

If the Windows operating system is started on the PC in such a network (a physical connection between PC and the network (ethernet cable) exists!), then the PC is automatically assigned an appropriate IP address and the value for the subnet mask is set. The measurement devices can obtain appropriate IP addresses and subnet masks in the same way if the use of DHCP for the device is enabled. Thus, within a network with a DHCP-server, there aren't any address conflicts and no problems finding the devices in the network.

If the same PC is now started outside of the network environment (where there isn't any DHCP-server; e.g. on board a vehicle), then assignment of the IP address proceeds according to the APIPA-protocol (**A**utomatic **P**riate **I**P **A**ddressing). In such cases, an IP address is assigned by the Windows operating system from a reserved IP address range (169.254.0.1 ... 169.254.255.254). The subnet mask is set to the value 255.255.0.0. See also [AutoIP](#)⁴⁶.

In this case the measurement devices use the IP address set in the device.

If an address from the range reserved for APIPA is selected in the device as the IP address, and the subnet mask is set to the value 255.255.0.0 (the use of DHCP for the device remains enabled!), then it's also possible to set up a connection between the PC and the device within the company network without any problems.

Note

- The IP address actually used for the PC and the value for the subnet mask can be determined by entering the following command line in the "(MSDOS-) input prompt": `ipconfig /all`
- If an IP address for a device is to be set from the range reserved for the APIPA, it must first be checked which IP address has been assigned for the PC, if no DHCP server is available. To do this, the PC is deactivated and disconnected from the network. After activation of the PC, the IP address currently in use is determined as described above. Make a note of this IP address!
- Ensure that each device is assigned its own IP address. This also goes for assignment of IP addresses from the range reserved for the APIPA! This prevents address conflicts even in networks without a DHCP-server.
- Fundamentally, the physical connection (ethernet cable) between the device and the network should be set up before the device is activated, so that automatic assignment of the IP address can work. If the automatic IP address reference fails, the IP address already set remains valid from this moment until the device is deactivated. However, this address isn't generally suitable for the IP addresses used in the network and address conflicts can result, or locating the device in the network will fail!

4.2.1.2.1 Computers with multiple TCP/IP connections

If you have set up multiple TCP/IP interfaces in your PC system, be sure to enter different subnets for each one! Otherwise random conflicts will occur. Reasons for having multiple TCP/IP interfaces include:

- You separate your connection to the company network and your connection to the measurement device by means of two network boards.
- You occasionally connect your measurement device via modem (PPP)
- You have added a Bluetooth connection.

The settings for the addresses could look like this, for instance:

	Measurement device	Router-company	Network Board1	Network Board 2
IP address	192.168.1.3	192.168.0.1	192.168.1.26	192.168.0.13
Subnet mask	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0

In this example, your PC would communicate via Network Board 1 with the device and via Board 2 with the company network.

4.2.1.2.2 DHCP and APIPA

The settings presented here apply to the imc device, not the PC:

Setting	Description
Use DHCP	(Dynamic Host Configuration Protocol) The device tries to get its settings from a DHCP server. If the server is not available, the fixed settings are used! This corresponds to the alternative configuration in Windows PC.
DHCP ClientID	The <i>DHCP ClientID</i> can be entered if the DHCP server requires a special DHCP ClientID for IP assignment. If there is no entry, the MAC-address is used. Supposing measurement devices are to be assigned IP by the DHCP server from a particular range. In that case, 'MeasDev 1' etc. could be entered, for example.
DHCP Hostname	If your administrator makes the device accessible via a DHCP Hostname, enter it here. Otherwise, leave this box empty.
Auto-IP (DHCP + APIPA)	(<i>Automatic Private IP Addressing</i>) Use DHCP is automatically activated along with this option. If the DHCP server cannot be reached, the fixed configuration for the address is not used. Instead, the device uses a random number generator to get an address in the range from 169.254.1.0 to 169.254.254.255. Afterwards it tests whether the address generated is available. If it is already used for another device, the search is repeated until a free address is found.

Note

If the PC's network interface is configured to "Obtain IP address automatically" (e.g. DHCP), then there is a problem with point-to-point connections when the network connection is cut and then restored (e.g. restart of the device interface in the process of a firmware update).

In such a case, the restoration of the device connection is delayed until the PC's network interface is completely reactivated (with a permanently valid IP address).

4.2.2 Start the network configuration - Interface configuration

By means of the dialogs "*Device Interfaces*" and "*Interface-Configuration*", it is possible to find out whether connection to the device can even be established at all. You are able to review the device's network system and to adapt the settings to the local network.

All entries are checked for plausibility, thus preventing incorrect entries. Even so, making the configuration requires system expertise and you may need support from your network administrator.

With imc WAVE various options for interface configuration are supplied:

Programs	Description								
Device Interfaces ⁴⁷	<p>Offers a simple guided interface for modifying the devices' LAN-configuration.</p> <p>If the simple configuration is inadequate, it is possible to start the advanced Assistant via the dialog: Interface-Configuration ⁴⁷.</p> <p>The following connection types can/can't be configured via the dialog:</p> <table border="1"> <thead> <tr> <th>yes</th> <th>no</th> </tr> </thead> <tbody> <tr> <td>Device connection via LAN</td> <td>Device connection via Modem (TCP/IP with PPP)</td> </tr> <tr> <td>Device connection via WLAN</td> <td>Device connection via WLAN</td> </tr> <tr> <td>IP-configuration only</td> <td>advanced configurations, such as speed, encryption, ... but must be pre-configured via Interface-Configuration ⁴⁷.</td> </tr> </tbody> </table>	yes	no	Device connection via LAN	Device connection via Modem (TCP/IP with PPP)	Device connection via WLAN	Device connection via WLAN	IP-configuration only	advanced configurations, such as speed, encryption, ... but must be pre-configured via Interface-Configuration ⁴⁷ .
yes	no								
Device connection via LAN	Device connection via Modem (TCP/IP with PPP)								
Device connection via WLAN	Device connection via WLAN								
IP-configuration only	advanced configurations, such as speed, encryption, ... but must be pre-configured via Interface-Configuration ⁴⁷ .								

[Interface-Configuration](#) ⁴⁷ The program is used for the extended configuration of imc devices.

Start the configuration with "Device interfaces"

Start imc WAVE. Click the "Device interfaces" button to open the dialog for configuring the IP address of the device.

Action	Page
Device options > Device interfaces	Devices

 [Reference](#) [Make settings](#)

See further: "[Device connection via LAN](#)" ⁴⁹

Start the configuration with "Interface-Configuration"

Start the program "imc DEVICES Interface-Configuration".

Start via:	Description
imc STUDIO	Click the "Device interfaces" button to open the dialog for configuring the IP address of the device. (See " Start the configuration with "Device interfaces" " ⁴⁷)
	There, click on the button: "Advanced Configuration".
the Start menu	Start the program: imc DEVICES Interface Configuration, from the program group: "imc" .

At the left there is a tree labeled "PC" and below it one entry for each network adapter which uses TCP/IP. The name or IP address of your PC is usually displayed here. The respective entry can consist of either a domain name or the associated adapter's IP address. This includes the dial-up network's adapters.

 [Note](#) [Inactive entries](#)

If all the entries have been set as inactive or there is no entry, your PC is not configured correctly. Exit the program and check all network settings. Subsequently, repeat the configuration.

Reference

Make settings

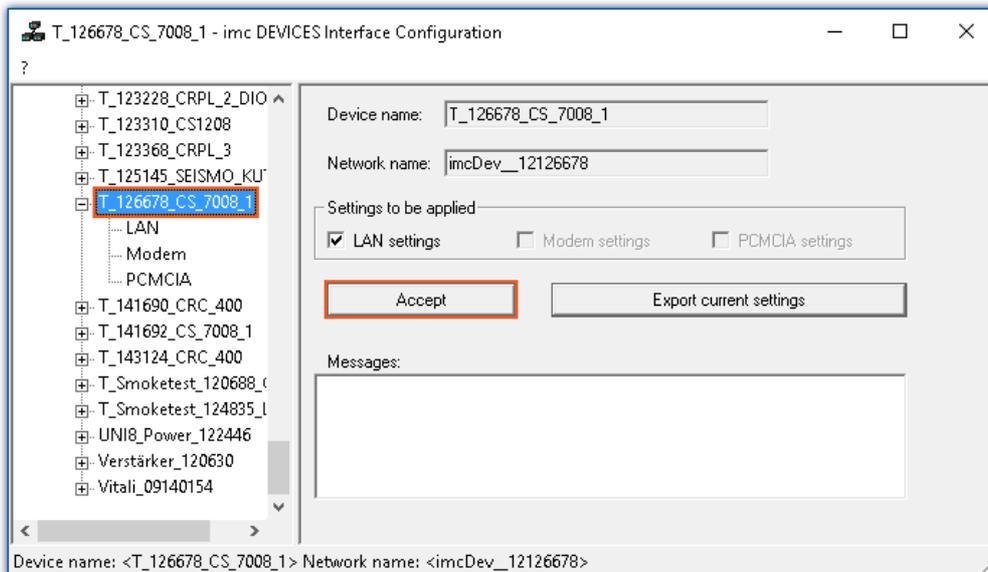
See further:

- [Device connection via LAN](#) ⁵¹
- [Device connection via WLAN](#) ⁵³

Accept changes

The configuration is only written to the respective measurement device once you click on the button "Accept".

- After the configuration, select the device again in order to return to the overview.



- In the group "Settings to be applied", select the device adapters for which the changes are to be applied.
- In order to write the changes to the measurement device, click on the button: "Accept"

Once the configuration has been written to the device, a warm start is carried out so that the new settings take effect. This process requires some time.

Reject changes

To reject all changes, simply exit Interface-Configuration and respond to the prompt "Reject all changes?" with "Yes". If you select "No" for this prompt, the entries you made on the right side of the dialog remain intact. But they will only be applied once you click on the button *Accept*.

Export current settings

For documentation purposes, it is possible to export the interface settings currently applicable in the device to a CFG-file. Import is not available. The device folder is automatically the destination and the suggested file name is the device name with a suffix.

Note

After confirmation "Interface-Configuration" allows, that the subnet mask has a format, that is not common for that network class.

Example: Networks starting with 192 are so called class C networks. They always use the subnet of the format 255.255.255.X. In special situation it is necessary to change the format to 255.255.X.X.

4.2.3 Device connection via LAN

Preparation

Connect the devices to the [LAN](#)⁴⁰ and switch the devices on.

Note

No devices found

If the devices described below fail to find any devices, check whether your PC and the devices are properly connected to the network and activated.

If finding devices still does not succeed, set up a direct connection to the PC. You should also check whether the [Firewall and port](#)⁴⁴ setting is preventing a network connection to the device.

DHCP

DHCP (Dynamic Host Configuration Protocol) in conjunction with an appropriate server enables dynamic assignment of an IP address to the devices (see "[DHCP and APIPA](#)⁴⁶").

The devices ship set for DHCP. If an DHCP server is activated in your network, no further manual settings are needed. See your network administrator if you have any questions.

Warning

Valid IP address

To set a valid and available IP address for the device in an existing network, be sure to contact your network administrator. If you set an IP address for the device which is already used by another device in the network, the result will be that one of the devices is no longer available in the network.

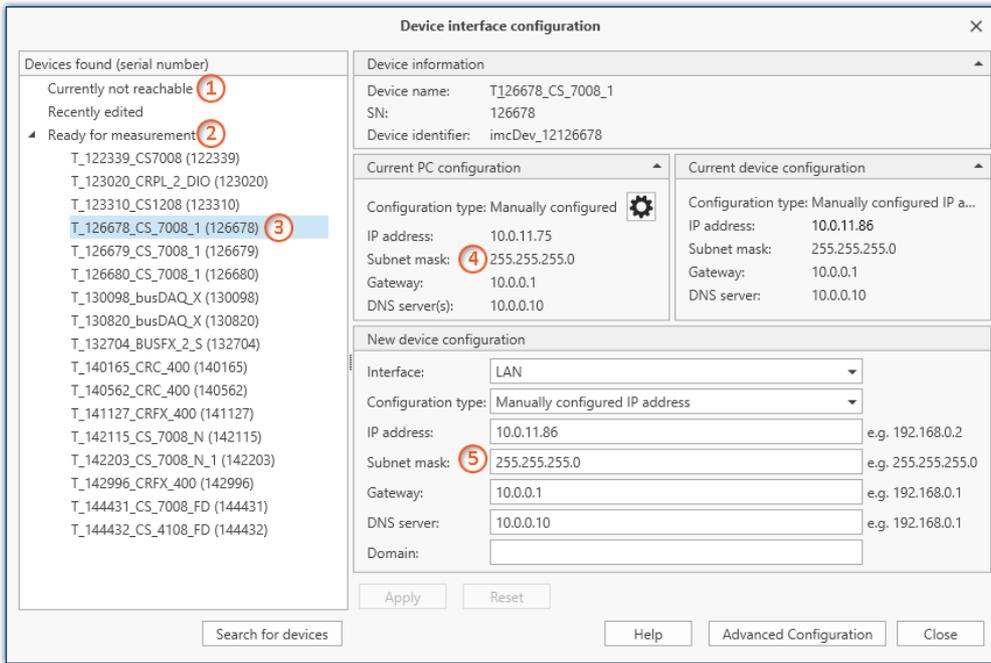
LAN-Configuration with Device Interfaces

Click the "[Device interfaces](#)" button to open the dialog for configuring the IP address of the device. See "[Start the network configuration - Interface configuration](#)⁴⁷".

Action	Page
Device options > Device interfaces	Devices

Once the dialog starts, the system automatically searches for all devices in the network. In the tree diagram, all available devices are indicated. If the device appears among the group "[Currently not reachable](#)" **①**, it is necessary to modify the device's LAN-settings. If the device appears among the group "[Ready for measurement](#)" **②**, you can leave the settings as they are or review them.

Select the device for making modifications **③**.



Display of measurement devices found and of the IP address

The right-side portion is subdivided into four regions:

Region	Description
Device information	Lists information for identifying the device.
Current PC configuration	Lists the current configuration of the PC-adapter via which the device was found. By means of the cogwheel-icon (⚙️) you can directly open the PC's adapter settings.
Current device configuration	Lists the current configuration of the device adapter.
New device configuration	Here, you can modify the device interface configuration.

Support by a configuration suggestion

This dialog offers support in configuring the interface. If the device's configuration is not appropriate for the PC, an appropriate configuration is found and suggested. Suggestions appear in "green" shade.

!

Note

Check the suggestion

Please check the suggestion exactly! Do not accept the suggestion without first verifying the IP address: Is it a valid and available IP address?

Subject to the PC-configuration, the next free IP address is offered as the suggestion. Free IP addresses might be occupied by devices which are currently switched off. In most cases, the suggested configuration can be used for a "point-to-point connection", since there are no other devices in the network in this case.

No suitable suggestion:

In some cases it may occur that no IP address is suggested. For IT reasons, the software is not able to conduct a search across all IP addresses and only a small number of IP addresses are checked. If all addresses checked are occupied, no result is returned. If the device is selected once again, a new search is started among other addresses.

If no address is suggested, it is best not to conduct any more searches but rather to manually enter a suitable address.

Procedure

- Select the device
- Check the settings and make the appropriate changes
- To write the changes to the measurement device, click on the button: "Apply"

Once the configuration has been written to the device, a warm start is carried out so that the new settings take effect. This process requires some time.

Advanced configuration

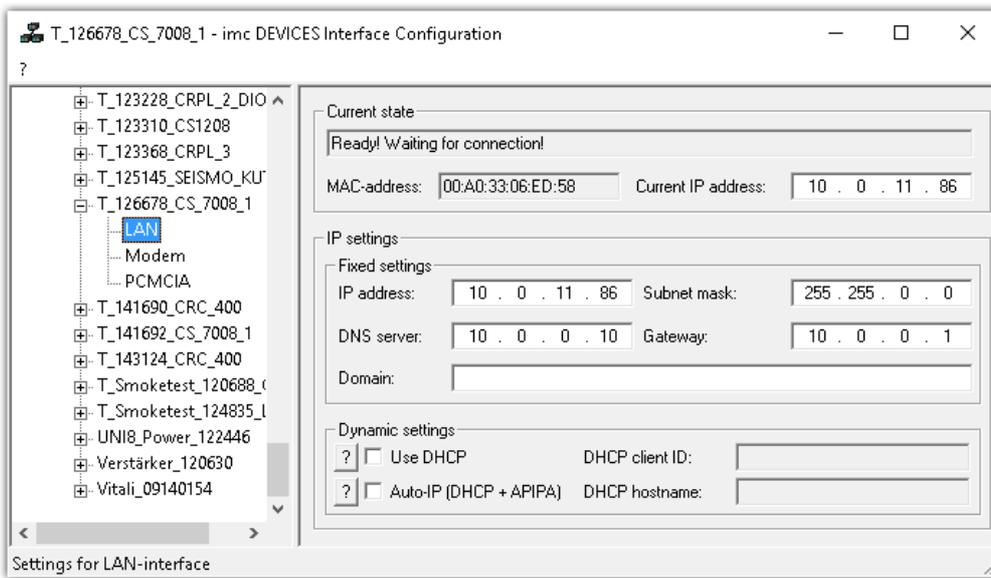
In some networks, configuration via the dialog is not adequate. For instance, it may be possible to activate the device connection via WLAN, but not to configure it. A device connection can not be established via modem (TCP/IP with PPP). In these cases, click on the button: "Advanced Configuration". The dialog: "[Interface-Configuration](#)" opens. Follow the instructions for configuring the device interface.

LAN-Configuration with Interface-Configuration

You can adjust the LAN configuration via the "[Interface-Configuration](#)" dialog.

Double-click on the network adapter's entry in order to start an automatic search. Depending on the network and the number of devices connected to it, this can take a while. Then the tree diagram will display all available devices under the adapter.

There, select the device desired and click on the sub-item "LAN". You can now see the measurement device's current configuration, as shown below:



TCP/IP configuration

Enter the IP settings in the right-side region.

Region	Description
Current state	Lists the device-adapter's current configuration.
IP settings	Here, you can modify the device interface configuration.

To write the changes to the measurement device, click on the button: "[Accept](#)"

Once the configuration has been written to the device, a warm start is carried out so that the new settings take effect. This process requires some time.

4.2.3.1 Connection via a direct address - User-defined device

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 "[Device Table](#)¹⁸⁶" and to establish a connection.

To do this, run a devices search by IP address or DNS name.

Action	Page
Device options > Search for devices by IP/DNS	Devices

Proceed as follows:

1. Open the dialog "Search for devices by IP/DNS".

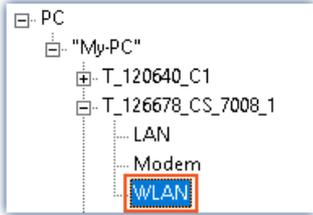


The screenshot shows a dialog box titled "User-defined device". It has a text input field for "Device DNS name or IP address" containing "10.0.11.86". Below this is a blue button labeled "Read device information". Underneath is a section titled "Device information" containing three text input fields: "Name" with "T_126678_CS_7008_1", "Serial no.:" with "126678", and "Type:" with "imc C Series". At the bottom are "OK" and "Cancel" buttons.

2. Enter the IP address or the DNS name.
3. Clicking on the button "*Read device information*" initiates an attempt to establish a connection. If it is successful, then the device's name, its serial number and its product type are displayed and the device can be integrated into the experiment.

4.2.4 Device connection via WLAN

You can adjust the WLAN configuration via the "[Interface-Configuration](#)" dialog. This can be reached via the dialog for "*Device interface-configuration*" -> "[Advanced Configuration](#)". There, select the device desired and click on the sub-item WLAN.



WLAN settings

Use integrated WLAN

Network

Network type: Accesspoint (Server) Network cell ID:

Network name:

Transfer

Channel: 5 Speed: auto

Encryption

Type: WPA2-PSK CCMP/AES (8 - 63 characters)

Key:

Parameter	Description
Network type	<ul style="list-style-type: none"> Managed (Client): In most cases, the measurement device is connected at an Accesspoint. For this purpose, the device must be operated in the mode "<i>Managed (Client)</i>" with the encryption specified (usually "<i>WPA2-PSK</i>"). <hr/> <p>It is crucial for the WLAN IP address range to be different from the LAN-IP address range.</p> <p> Example: LAN: IP 192.168.1.20 -> WLAN may not have an IP from the range 192.168.1.x!</p> <hr/> <ul style="list-style-type: none"> Accesspoint (Server): If no Accesspoint is available, it is possible to operated some imc devices as an "<i>Accesspoint (Server)</i>". Here, too, the use of "<i>WPA2-PSK</i>" encryption with a secure password is recommended. Both the measurement device and the PC must have a permanently set IP if there is no DHCP Server available. Ad-Hoc mode is no longer supported under Windows 10.
Network cell ID	This may be required in order to select an Accesspoint (4 hexadecimal characters); otherwise, leave this empty.
Network name	Name of the network to which you intend to connect the imc device: <ul style="list-style-type: none"> For the network type <i>Accesspoint (Server)</i>, the imc-device automatically provides the WLAN name via which the PC connects with it. For the network type <i>Managed(Client)</i>, you must enter here the WLAN-name via which the imc device is to be accessed.

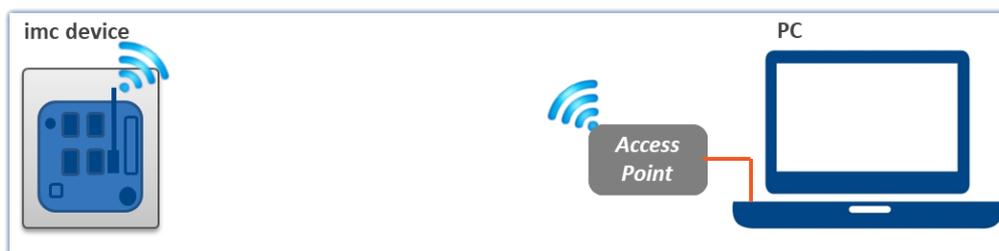
Parameter	Description
Channel	Channel 1-13 for 2.4 GHz. If the WLAN-module supports 5GHz, there are >36 channels. Observe all national regulations. For instance, in the USA, channels 12 and 13 may not be used.
Speed	The meaning of <i>auto</i> depends on the integrated WLAN-card! 11 Mbit/s max means that the system attempts to transfer data at 11 Mbit. If the connection is too bad, however, the transfer rate is reduced automatically. With the setting 11 Mbit/s fixed , the transfer rate is not adapted automatically.
Encryption	The key must be the same for all participants. WPA-PSK or WPA2-PSK; the latter is more secure. Radius-authentication is not supported. WEP has become obsolete and is not recommended.

Note

- Do not enable "**Use DHCP**" in the "*Dynamic settings*" section if the device operates as an **Access point (Server)**. However, the device does not assign IP addresses! These must be assigned in the same IP address range in the PC-WLAN settings. Make sure that the subnet mask on the PC corresponds to the device.
- If a maximum value has been set for the speed and the connection quality is bad, the speed is adjusted automatically. In consequence of frequent speed adjustments, there may be data overflows. In such a case, set a lower maximum speed.
- The device does not support data exchange between the WLAN connection and the LAN connection. Even if unauthorized access to the WLAN takes place, there is no danger of access to the device via the LAN.
- For devices as of group A4 and higher (SN>13xxxx), up to **54 Mbit/s** can be transmitted.
- For devices as of group A7 and higher (SN>19xxxx) up to **300 Mbit/s** can be transmitted. Therefore *Speed at Transfer* must be set to "*auto*". Furthermore the *Network type "Managed"* may be activated and please select as Encryption "**WPA2-PSK CCMP/AES (8-63 characters)**". If you select another encryption, then the speed will be reduced to 54 Mbit/s conforming to the 802.11n Standard.

4.2.4.1 WLAN connection setup possibilities

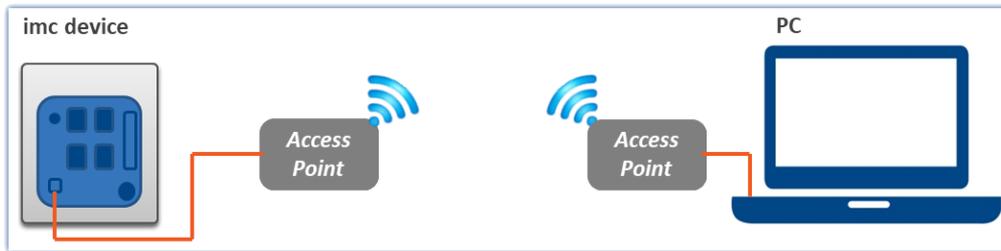
PC via Access Point - Device via integrated WLAN



PC via Access Point - Device via integrated WLAN

The PC is connected with an Access Point. This connection is made either with a cable or a stable WLAN connection. The Access Point can be chosen in such a way to ensure adequate power supply for a strong antenna.

PC via Access Point - Device via Access Point



PC and device via Access Points

This option is well suited for wireless transmission across especially long and problematic expanses.

At distances of over 30m, we recommend the use of Access Points at both ends. This means that strong, directed antennas can be used whose power supply can be ensured by the choice of appropriate Access Points.

The connection between the Access Point and the PC or the device is made by a cable or stable WLAN hookup.

In this case, the interface from the PC and the measurement device is configured like a normal [LAN connection](#)⁴⁹. The cable is replaced by the wireless connection and administered by the Access Points. These must be configured in accordance with the instructions provided for them.

4.3 Information and tips

4.3.1 Determining the MAC-address of the measurement device

In order to find your imc measurement device's MAC-address, start the "input prompt" (Keyboard combination [`<Win> + R`]). Type `cmd` and confirm by pressing Return).

There, begin by accessing the device with a ping and the IP address. The IP address can be found with the help of the program [Interface-Configuration](#)⁴⁷.

Example for the ping command:

```
ping 10.0.9.30
```

```

C:\>ping 10.0.10.30

Pinging 10.0.10.30 with 32 bytes of data:

Reply from 10.0.10.30: bytes=32 time=2ms TTL=64
Reply from 10.0.10.30: bytes=32 time<1ms TTL=64
Reply from 10.0.10.30: bytes=32 time<1ms TTL=64
Reply from 10.0.10.30: bytes=32 time<1ms TTL=64

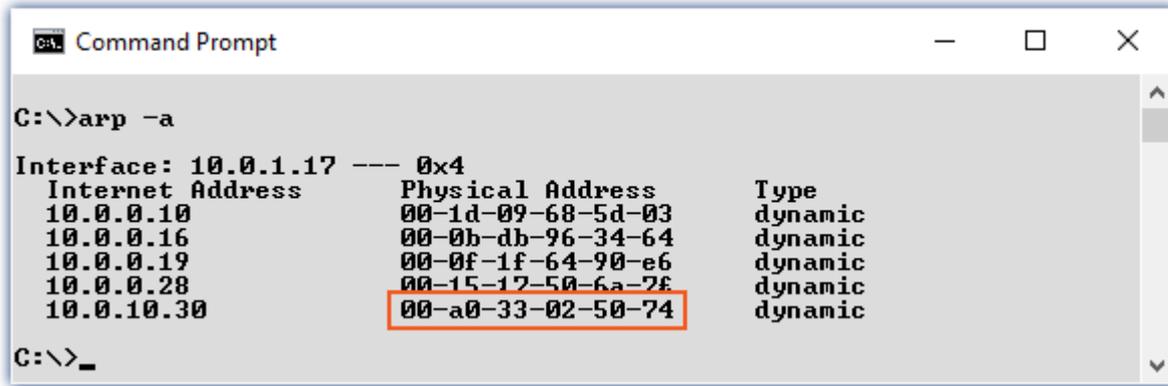
Ping statistics for 10.0.10.30:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 2ms, Average = 0ms

C:\>_

```

Next, enter the following command:

```
arp -a
```



```

C:\>arp -a

Interface: 10.0.1.17 --- 0x4
Internet Address      Physical Address      Type
10.0.0.10             00-1d-09-68-5d-03    dynamic
10.0.0.16             00-0b-db-96-34-64    dynamic
10.0.0.19             00-0f-1f-64-90-e6    dynamic
10.0.0.28             00-15-17-50-6a-7f    dynamic
10.0.10.30            00-a0-33-02-50-74    dynamic
C:\>_

```

In the second column ("Physical Address") you will find the MAC-address.

4.4 The network

Optimize the network

You should observe the following notes for networks with high data rate, to reduce the risk of a data overflow.

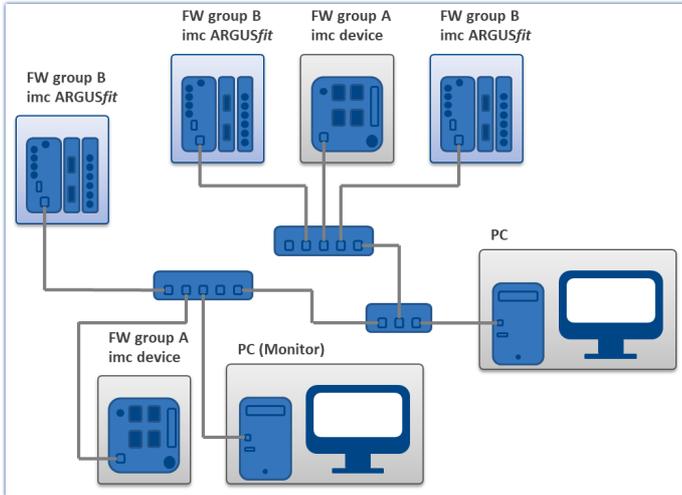
1. The network between switches and PC should be 1Gbit/s
2. The actual data rate is decreasing with the number of channels. Avoid unnecessary channels.
3. The RAM buffer should be as big as possible, not "Auto". See also [RAM buffer time](#)³²².
4. Save only channels you need to the internal disk.
5. Avoid curve window setting that need a lot of performance. Choose scroll mode. Avoid the rezoom function after a long measurement duration. Don't set the sample points to symbols. Try to find a curve window configuration, that shows no bucking.
6. Observe the [recommended virus scanner settings](#)³⁵! It would check every sample written to the hard drive. That takes most of the performance of a PC and the hard drive.
7. Check the processor and network load with the PC task manager

Using a second network

To avoid disturbing data transfer between devices units and PC's, a second network using a second network card per PC can be set up. This second network can be used for transferring data between the PC's.

Example of a Complex Network

Until now, we have only considered networks consisting of a PC and *n* imc devices units. However, any arbitrary combinations are possible.



Monitoring

The PC on which imc WAVE is installed is declared as the Master-PC, which configures the measurement device.

For devices belonging to the [firmware group A](#)¹⁴⁶ - e.g. CRONOS family, C-SERIES, SPARTAN, BUSDAQ: Various other clients can access the measuring device via the network (e.g. via imc STUDIO Monitor or imc LINK or, it is set up accordingly, using a browser via imc REMOTE WebServer).

The well-known network restrictions are also valid here. In particular, it is impossible for more than one person to write to the same file. Furthermore, it is impossible for one person to open a file while another writes to it. In general, all network restrictions that apply to files also apply to the individual Devices units.

In consequence, the software can prevent multiple users from configuring the same device.

4.5 Firmware version

imc DEVICEcore and imc DEVICES as a driver package for imc WAVE

The drivers and firmware components for imc devices are provided in "imc DEVICEcore" and "imc DEVICES". Thus, at least one of the two driver packages is a necessary requirement for operating imc devices under imc WAVE.

- imc DEVICEcore: For devices belonging to the [firmware group B](#)¹⁴⁶ - e.g. ARGUSfit, EOS
- imc DEVICES: For devices belonging to the [firmware group A](#)¹⁴⁶ - e.g. CRONOS family, C-SERIES, SPARTAN, BUSDAQ

Both packages can be installed and operated in parallel. The software and the devices automatically apply the appropriate package.

The software only works with devices having the right firmware. Upon the connection/preparation, the versions are compared. If they don't match, the [firmware update](#)⁵⁸ is carried out.

Note

Parallel firmware versions

imc DEVICEcore: The associated firmware- and driver package ships with the software used, and is installed in the software's installation folder. Any other installations are not recognized and thus cannot be used.

imc DEVICES: It is possible to work with multiple groups of imc DEVICES simultaneously and in parallel. In special cases, this may be relevant or even recommendable, particularly in conjunction with the use of multiple devices. Additional notes on this subject are provided in the chapter: "[Associated Firmware \(imc DEVICES\)](#)"⁶²

4.5.1 Firmware update

Every software version comes with matching firmware for the hardware. The software only works with devices having the right firmware.

Once the program connects up with the unit, the device's firmware is checked. If the software version doesn't match the device's firmware version, you are asked if you want to perform a firmware-update.

Note

The firmware update is only required if the software was obtained as an update. If you obtained your hardware equipment together with the software, no firmware update is necessary.

Warning

Do not interrupt the firmware update!

Be absolutely certain to observe the following:

1. Under no circumstances should the device or its power supply be deactivated during the firmware update!
2. The network connection may not be interrupted. Use a cable connection, not WLAN!

Firmware update - imc DEVICES

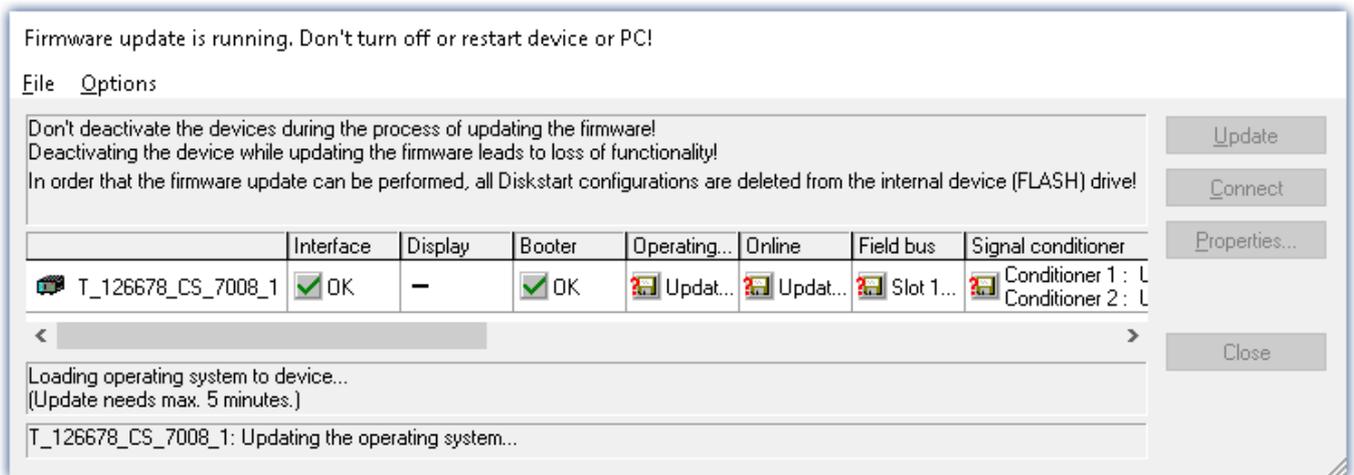
Note

Firmware and driver package

The following description applies to devices of firmware group A (imc DEVICES) - e.g. CRONOS family, C-SERIES, SPARTAN, BUSDAQ.

Depending on the device type, the following components are loaded automatically: Interface-firmware (Ethernet, modem, ...), booting program, amplifier firmware, firmware for the signal processors.

The dialog for the firmware-update looks like this:



*Start of the firmware update (example of a single device)
The state of the components of the firmware is displayed in the list.*

Component	Description
Interface	Interface-Firmware (Ethernet)
Booter	Start-up program for the device upon switching-on
Operating system	Device operating system
Online	Online-functions and hard drive controller
Display	Operating system of the connected displays
Fieldbus	Fieldbus interfaces (e.g. CAN etc.)
Signal conditioners	Amplifiers

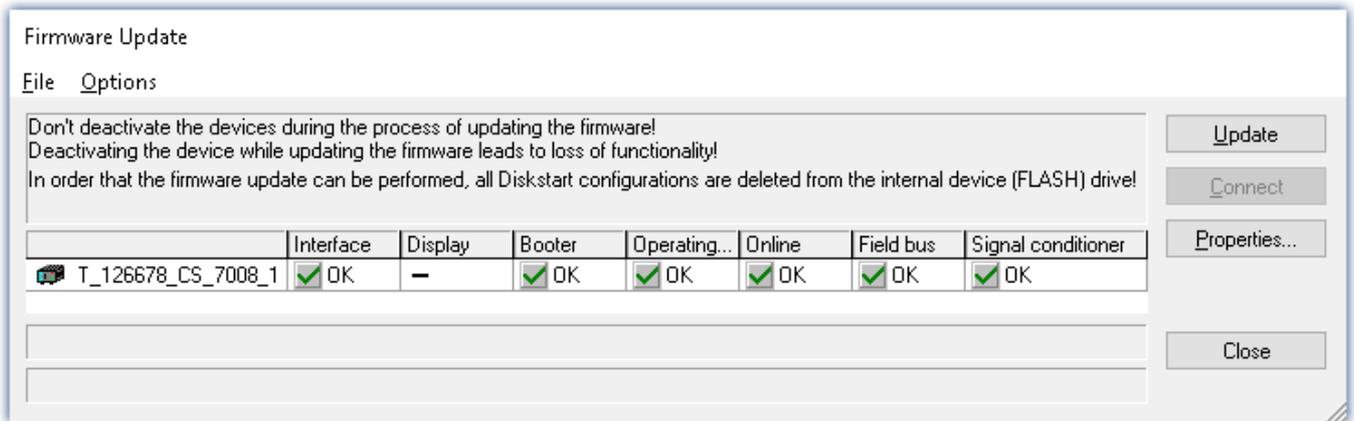
The following symbols for the individual firmware components appear in the list:

Symbols	
	not current
	firmware conforms to current standards
	error occurred during update procedure
	this option is not available on the device

If no status indicators are displayed, no connection could be made to the corresponding device.

The duration of the update depends on the number of amplifiers (can last up to several minutes). You will be informed on the progress.

You are notified when the firmware setup concludes successfully, as shown below:



Conclusion of the firmware update (example of a single device)

Choose "Close". The device can now be used with the product software.

 **Warning** Be sure to observe in case of error

- For a variety of reasons, the firmware update sometimes does not conclude properly, for example due to interruption of the power supply. For instance, the "handshake signal" at the end of the procedure may be missing. In this case, no measurement channels would be displayed initially. However, restarting the device and its software and performing the firmware update again usually restores everything to normal. It may be necessary to call the menu function "Update all components" in the Firmware update dialog's Options menu. This scenario only results in permanent damage in the most rare cases, and it is very worthwhile to repeat the procedure before sending a device in for repair.
- Behaviour under error condition, Windows cuts off the network connection without the user's knowledge; but this can be prevented using the PC's Control Panel.
Background: During the firmware updates there is no data transfer for a few minutes and thus no network activity; Windows detects inactivity of the connection and the following mechanisms are set in motion:
 - a) Windows' energy saving mode switches the LAN adapter off, consequently interrupting the network connection!
 - b) Windows switches to the next LAN adapter if there is one (some PCs have multiple adapters in order to, for instance, to access services in parallel that are accessible via separate networks.)
 - c) Other scenarios are feasible, e.g. if switches are activated, which can also respond to missing data traffic.

If an error message is posted during the firmware update, leave the device on and contact our [tech support](#). The firmware update may be continued with guidance from the tech support.

 **Note**

Firmware logbook

The "File" menu offers a function for working with the firmware log file. Every action taken during a firmware update plus any errors which may occur are recorded in a log file. This log file can be displayed with menu "File" > "Show log file".

Update all components

The "Options" menu offers the option to "Update all components". This makes it possible to earmark all the components of the selected device for an update. The function is only to be used in compliance with instructions from the tech support.

Firmware update - Password locking

It is possible to prevent an unintentional firmware update. For this purpose, the "Access protection" > "Password" and "Use for firmware update" control is provided in the "[Device properties dialog](#)".

Action	Page
Dialog: Device options > Device Properties	Devices

No default password is supplied in the manufacturing. When a firmware update is started, the system checks whether firmware updating is enabled in the device configuration; otherwise the firmware update is canceled with an error message.

+	Device	
-	Access protection	
	Password	not active=0
	Use for firmware update	don't use=0
	Use for establishing connection	don't use=0
	FTP-login name	imc
	FTP-password	not active=0
	Anonymous FTP access	enabled=1
+	UPS	

Blocking / Unblocking of the firmware update

Blocking and unblocking

The firmware update is blocked with a freely defined password. To unblock, you must enter the password which was used to install the blockage. It is only removed if the password is identical with the password saved in the device.



Note

A general password is available

Resetting of the password can also be accomplished by a **General password**. This general password is derived from the device's serial number and can be provided by imc if necessary.

Firmware update - imc DEVICEcore

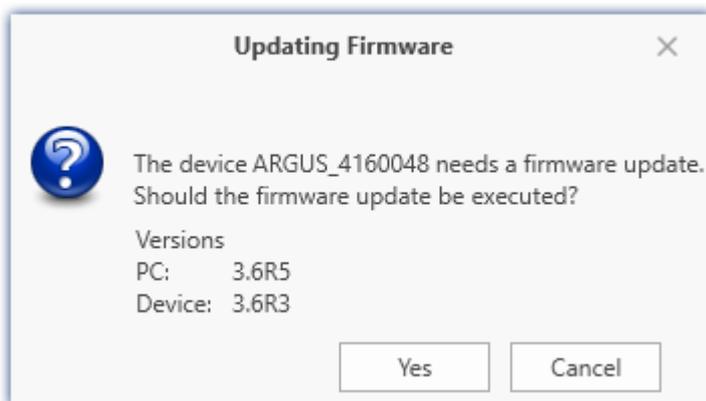


Note

Firmware and driver package

The following description applies to devices of firmware group B (imc DEVICEcore) - e.g. ARGUSfit, EOS.

Upon the connection/preparation, the versions are compared. If they do not match, the next dialog notifies you about this, and indicates both which version is being used and which version is required.



When you click "Yes", the firmware update starts. The duration of the update depends on the number of amplifiers (can last up to several minutes). You will be informed on the progress.

The system notifies you when the firmware update has been completed successfully. Close the dialog. Subsequently the device can be used with the device software.



What to do in case of an error occurring in a firmware update?

Answer: When a firmware update fails to run correctly, the last active firmware is restored. The device remains operational by means of the previous installed firmware. Repeat the procedure, and if the repeat procedure also fails, please contact our tech support.

What is an optional update?

Answer: In some cases, a firmware update is not mandatory. In such cases, the update is available to you as "optional". However, we still recommend performing the update, since every update not only implements improvements, but also fixes any bugs. Nevertheless, with the new version of imc WAVE, you are able to work with the device running the old firmware edition, if you choose.

4.5.2 Associated firmware (imc DEVICES)

If multiple firmware versions are installed on the PC, then a defined version must be selected for each device. The selection strategy is set in the Options: "Setup" > "Device options" > "Selection of the imc DEVICES version".

Action	Page
Options (🔗)	Homepage

Option	Description
Selection of the imc DEVICES version	<p>If multiple imc DEVICES versions are installed on the PC, a specific version must be selected for the operation of each device. This option controls the selection strategy.</p> <p>If only one imc DEVICES version is installed, this setting has no effect.</p> <ul style="list-style-type: none"> • Manual: With "Manual", the system always asks which firmware version to use when device are selected or an experiment is loaded. • Automatic: "Automatic" avoids firmware updates. The version currently running on the device is used preferentially. • Always use latest: With this selection, the most current firmware version compatible with this imc WAVE version is always used.



Note

Use of older firmware

Please note that a new firmware version not only contains new functions, but also reflects resolved issues. These changes only take effect if the device is also using the new firmware.



FAQ

Question: Which firmware versions support my imc WAVE version?

Answer: In regard to the use of imc DEVICES as the device driver, there is a clear distinction determining the compatibility of versions/version groups:

imc WAVE version	assigned firmware / version group	additionally compatible
5.2	2.10, 2.11, 2.13	
2022	2.15	2.14
2023	2.16	2.14, 2.15
...	...	

For version 5.2, the applicable rule is: the assigned version group and all **predecessor groups going back to 2.10** are compatible with the imc WAVE-version used.

As of version 2022, the applicable rule is: the assigned version group and all **predecessor groups going back to 2.14** are compatible with the imc WAVE-version used.

Question: Can devices in the same experiment use different firmware versions?

Answer: Yes. If multiple devices are used, each one may use a different firmware version. Prerequisite: The imc WAVE-version used supports the firmware versions.

Question: Can I install a new firmware version without changing the imc WAVE version?

Answer: Yes. Prerequisite: The imc WAVE version used supports the firmware version.

Question: Is there a way to determine what firmware the device is using?

Answer: Yes.

- On the Setup page: "[Devices](#)¹⁸⁵", a column can be added: Device firmware
- In the [Device properties](#)¹⁵³.

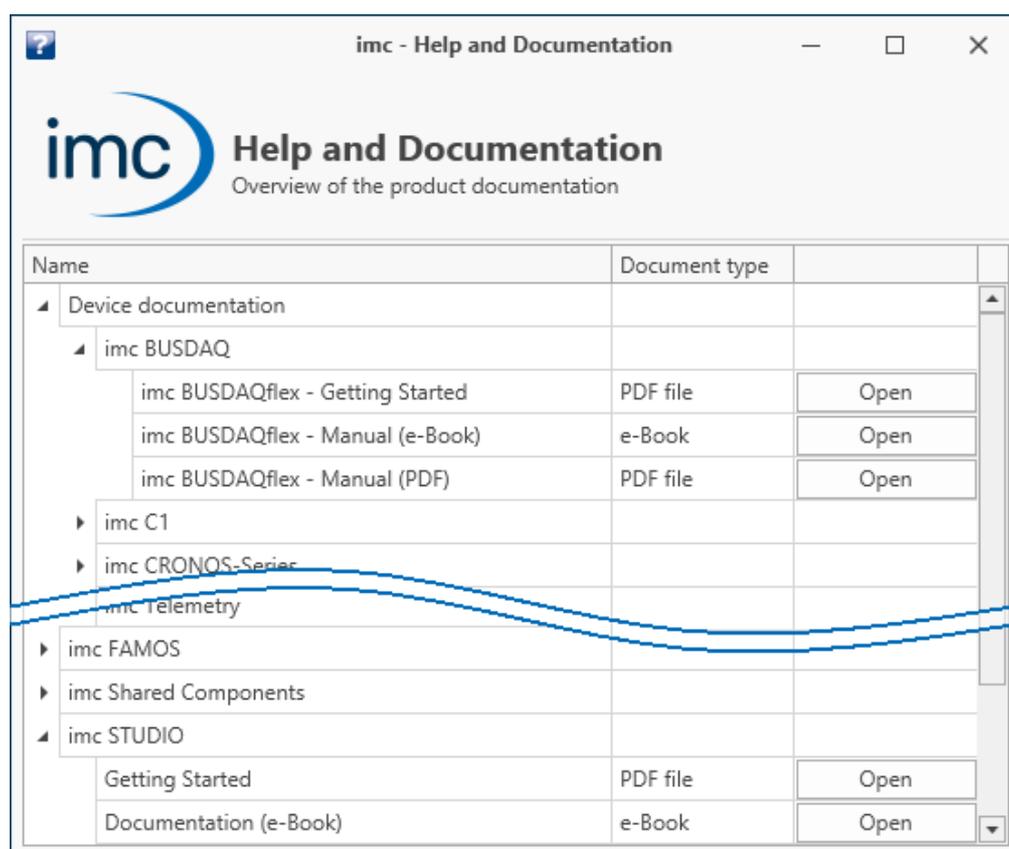
5 Documentation of devices - Help and Documentation

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.

6 Demo Experiments

Getting a feel for imc WAVE

It is possible to download demo experiments from the imc-homepage (Spectrum Analyzer). The experiment contains measurement data which provide a sense of how imc WAVE can be used.

You can view the pages and play back the measured data without needing to add any additional device. And you may use the experiment for your own purposes, if desired.

Notes on the Demo-experiments are available on our webpage in the Download area for imc WAVE:

<https://www.imc-tm.com/imc-wave/demo-experiments/>

7 imc WAVE

imc WAVE (**W**orkstation for **A**coustic and **V**ibration **E**ngineering) 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.

The software enables the user to perform measurements, evaluations, data visualization, and to compose reports specifically for acoustic tests. The device families imc CRONOS und imc C-SERIES in particular are used for this purpose.

One distinctive feature of the imc WAVE analyzers is their capability to process not only the primary acoustics signals from microphones and accelerometers, but also additional measurement data channels that an imc device can provide. Such as temperature, mechanical strain (strain gauges) etc., or signals and other information from CAN-busses on board vehicles. This means that operating conditions of the test object and its environment can be measured synchronously along with the acoustic variables, and that the results can be correlated and analyzed accordingly.

imc WAVE comprises comfortable user guidance along the typical workflow, including menus for configuration of the applied measurement systems and calibration routines for the sensors.

The captured measurement data are processed according to comprehensive analysis procedures that are fully compliant with all relevant standards. All data are saved and administered and the results are visually displayed in a selection of pre-defined pages on the PC and as printable reports.

Chapter overview

Synopsis	Section
Overview of the most important terms and links to advanced topics	<ul style="list-style-type: none"> • "What is..." - Terminology ⁷⁶
Operation and description of the user interface	<ul style="list-style-type: none"> • The User Interface ⁷⁶ • Setup panel pages for device configuration ⁷⁹
Analyzers - Configuration and measurement	<ul style="list-style-type: none"> • imc WAVE Spectrum Analyzer ⁹⁴ • imc WAVE Structural Analyzer ¹³²

7.1 Architecture

imc WAVE provides a platform for various analyzers which are each optimized for specific applications, e.g

- **Spectrum Analyzer**
 - Noise
 - Rotation
 - Vibration
 - Structure
- **Structural Analyzer**



7.2 "What is..." - Terminology

imc WAVE is experiment-oriented and is equipped with a simple operating interface. A experiment is begun by selecting an analyzer. The analyzers provide appropriate settings options and automated analysis capabilities. There are additional pages used by all the analyzers in common, such as channel and device settings, trigger settings, etc.

Below, an overview of the most important points is presented. See the subsequent chapters of this manual for more detailed explanations.

Data acquisition

A variety of imc measurement devices are available for the purpose of data acquisition. A list of all available devices appears in the technical data sheet. The devices are operated and configured by means of the user interface.

Online data-processing

The result data are computed whenever possible during a running measurement.

Offline data-processing / Further evaluation (only Spectrum Analyzer)

In order to subject the signal data to further evaluation, it is possible to perform offline-analysis. The saved measurement readings are the raw data for the re-calculations. In this way, data already recorded can be re-evaluated by changing the analyzer settings and running the function "Play back measured data" in the Data Browser.

Display on the Panel pages

Data analysis is already under way during data capture. For display and control purposes, various Panel pages are available. Each evaluation page contains various Widgets for display of measured data, such as the curve window. In addition, a variety of graphic analysis techniques is available, from comparisons all the way to 3D waterfall and color map representation.

Thus you have the ability to set up additional pages with curve windows.

Experiment: How to create a experiment?

imc WAVE saves the settings in so-called "[experiments](#)". The first step towards performing a measurement is **to create a new experiment** (Homepage: Button ).

In the subsequent dialog,

- select the analyzer desired and
- enter a name under which to save the experiment.

Analyzers are available for selection in the list on the left side. You can use whichever analyzers you have enabled by obtaining a license.

Once all information has been entered and confirmed by pressing the "OK" button, the experiment is saved under the selected name. You can now proceed to configure the experiment and perform measurements.

What pages are available?

Each analyzer has its own special pages for configuration and analysis. Some pages exist in all of the analyzers, e.g. those for configuration of the devices and channels. The pages are displayed with tabs. Which tabs are present can be set on the homepage.

Page	Description
 Homepage ⁷⁶	Sets which pages are visible. Configuration of the meta-information.
 Devices ⁷⁹	Configuration of the measurement device: device selection, data storage, synchronization, timed start, ...
 Analog Channels ⁸⁰	Configuration of the analog input- and output channels , as well as of the calculated channels (virtual channels). The analyzer's settings (Spectrum Analyzer).
 Digital Channels ⁸⁰	Configuration of the digital input- and output channels and incremental counters .
 GPS ⁸⁰	Configuration of the GPS channels .
 Variables ⁸⁰	Configuration of various existing variables: Display variables, Ethernet bits, Virtual bits, Process vector variables, ...
 Channel balance ⁸¹	The balancing settings for the analog input channels.
 Triggers ⁸⁰	Configuration of the triggers in order to perform targeted starting and stopping of channels.
 WAVE Analyzer	The analyzer's settings (Structural Analyzer).
 Measurement	The analysis options and display settings for the measured data.
Optional pages	Additional customized pages.

The individual items are described below. Each of the items pertains to a particular experiment. When you switch to a different experiment, the data for the new experiment are opened.

What is an analyzer?

Formally, an analyzer transforms the input signals into output signals by means of an algorithm. The output signals are then available for display and evaluation.

An analyzer is selected when creating a experiment and is permanent for the duration of the experiment. It is a fixed property of a experiment; it cannot be changed within a experiment. Depending on the analyzer selected, different settings and processing steps are available. The description of the analyzers will appear further below.

In the context of imc WAVE/imc STUDIO, analyzers are administered as so-called "*Projects*".

How is a new page created?

To create a new page, click on the button  on the "*Homepage*". Under "*Import Panel page*", select the checkbox for the desired page. Enter a name and click on "OK". Subsequently, a new tab will appear for the new page, having the name you entered.

On the page "**Command batches**", you can create and run your own command sequences. There are a few choices of commands available for the sequence.

The sequences can be run manually or be appended to events.

Event	Description
Experiment_loaded	After loading an experiment
Devices_AfterCheckConfiguration	After checking all device configurations
Devices_Stopped	After the measurement has stopped for all devices
WAVE_Results_Saved	After the saving of WAVE channels
OnError	After occurrence of an error

Using the page "**Customized page**" you can select how many curve windows you wish to see within the page ("Number of windows"). Depending on how the page is partitioned, you now have one to four curve windows on your page. The partitioning and size of the curve windows is permanent.

Each of these curve windows can be configured individually.

By means of the dialog called by the button: , you can manage your own pages, e.g. change the name or delete pages. The pages are associated with a particular experiment; in order to be able to use the pages also in other experiments, you can export and import them.

Curve windows

A curve window is where the measured data are graphically displayed. There are numerous options for modifying the appearance and content of curve windows. An overview of the various functions and settings options is presented in the chapter: "Curve Window". All of the curve window's settings can be reached via the context menu.

The curve windows of the pre-defined pages are configured according to their associated analyzer in such a way as to display the most important data. However, you are able to customize the curve windows according to your requirements.

Displaying measured data in the curve window:

If you have not yet performed any measurement, perform "Process configuration" () . This provides empty measurement data sets which you can display.

Select the curve window's menu item "*Configuration*" > "*More Channels*". In the dialog which then appears, you can add data from the right-hand list by means of drag&drop into the left-hand list. All waveforms listed at left are displayed in the curve window.

How to start a measurement?

After creating a experiment, selecting and configuring the display, and configuring the analyzers (see below), measurement can be prepared and started.

The analyzer, which is the heart of imc WAVE, computes the desired output channel signals from the data provided by the input channels. These channels can be displayed and stored along with the input channels. To activate the output channels, select the analyzer's configuration dialog and make the appropriate settings.

Once you have made all settings, you can start the measurement by clicking on the button  .

7.3 Experiments, projects and the database

This chapter describes the interdependencies among the "*database*", "*projects*" 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 WAVE 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](#)⁷³.

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](#)¹⁸⁷" (Setup page: "*Devices*" > "*Storage*").

Project

In the context of imc WAVE/imc STUDIO, analyzers are administered as so-called "*Projects*".

A project is primarily a collection of various experiments.

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.

Database

The database is the data repository for imc WAVE. 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 	<p>Here you set where the "Database" is to be saved.</p> <p> 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.</p>

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	<p>Select the appropriate database and click on the button "Apply".</p> <p>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:</p> <ul style="list-style-type: none"> • 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 "Create new". 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	<p>Click on the button "..." next to where the folder is indicated. Select an appropriate folder.</p> <p> Here, select a folder where the database folder is to be created later. Not the database itself. E.g. the default folder: "C:\Users\Public\Documents". The database will be created in this folder, e.g. "DB".</p>

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.

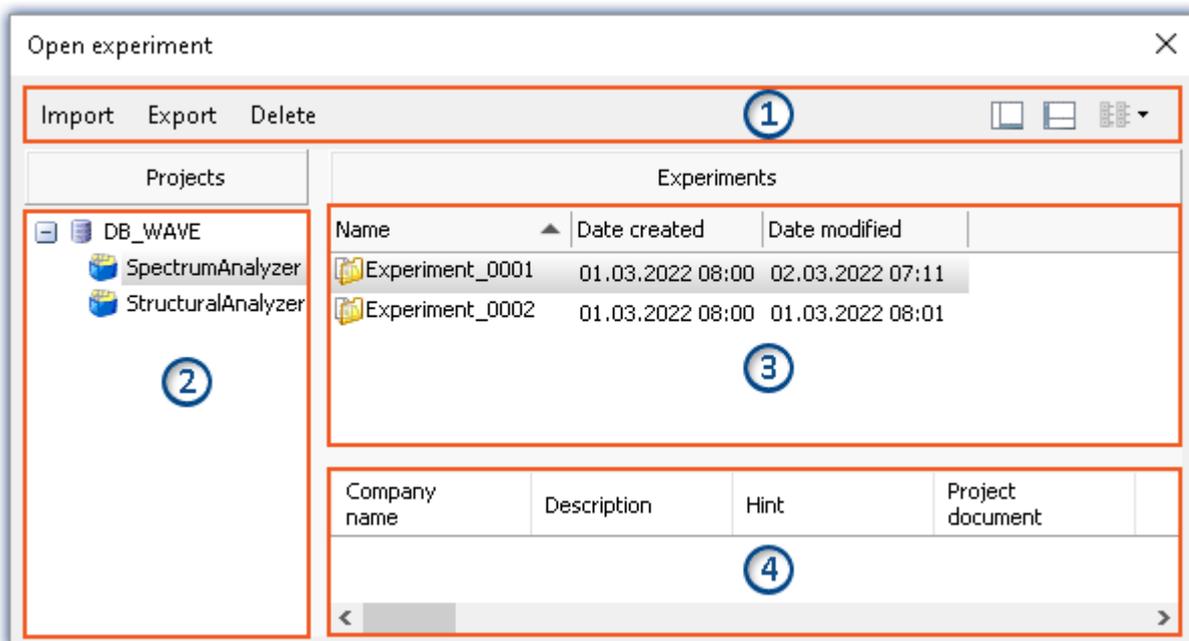
The folder "Meta" is now ignored in all components.

- 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.

7.3.1 Dialogs: Project and Experiment

The Project Management dialogs are described below.

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



Example: Open experiment

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

1. Menu bar
2. List of all projects
3. List of all experiments 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
	Imports elements 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.
	Exports the selected entries to a file. Individual or multiple experiments 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 ".
Delete	Deletes the selected entries. Individual or multiple experiments 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 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 region is shown (via the menu button: ).

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.

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

7.3.2 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, or when you save an experiment under a new name, you are asked for a name. In this case, enter a unique name.

Action	Description
New Experiment	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. <ul style="list-style-type: none"> 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" are copied and are available in the new experiment.

7.3.3 Exporting and importing experiments and projects

Using the experiment-dialogs, you can export/import experiments along with associated measurement data.

Exporting

The elements selected are exported to a file.

Action	Page
Open (📁)	Homepage
Save as (💾)	Homepage

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
Experiment-settings	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](#)⁷²".

Importing

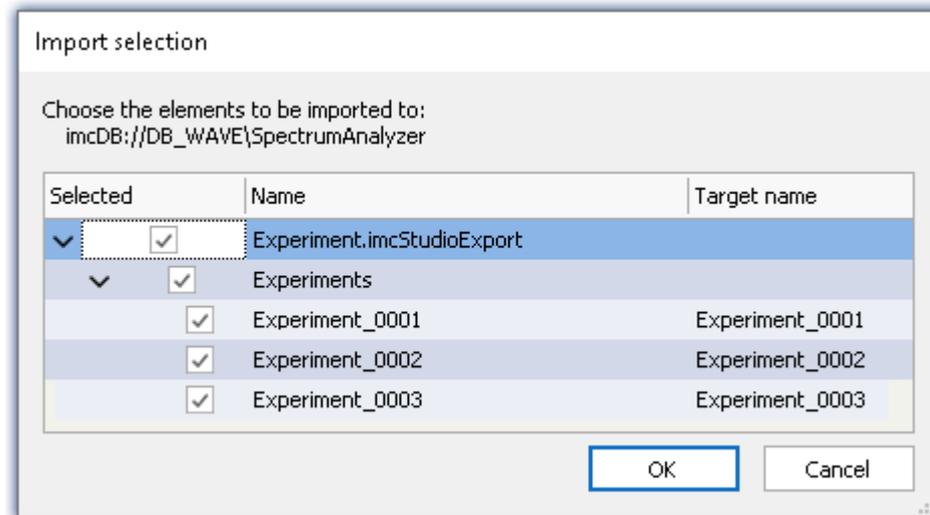
You can import previously exported experiments by means of the experiment-dialog.

Action	Page
Open (📁)	Homepage

Procedure: First select the location to which to import the content - select a **suitable project** (analyzer). 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 experiments). 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)



Import multiple experiments from a single file

Be sure that you previously select the target correctly. You can **import** a experiment if a **suitable project (analyzer)** is selected.

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
Experiment-settings	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?

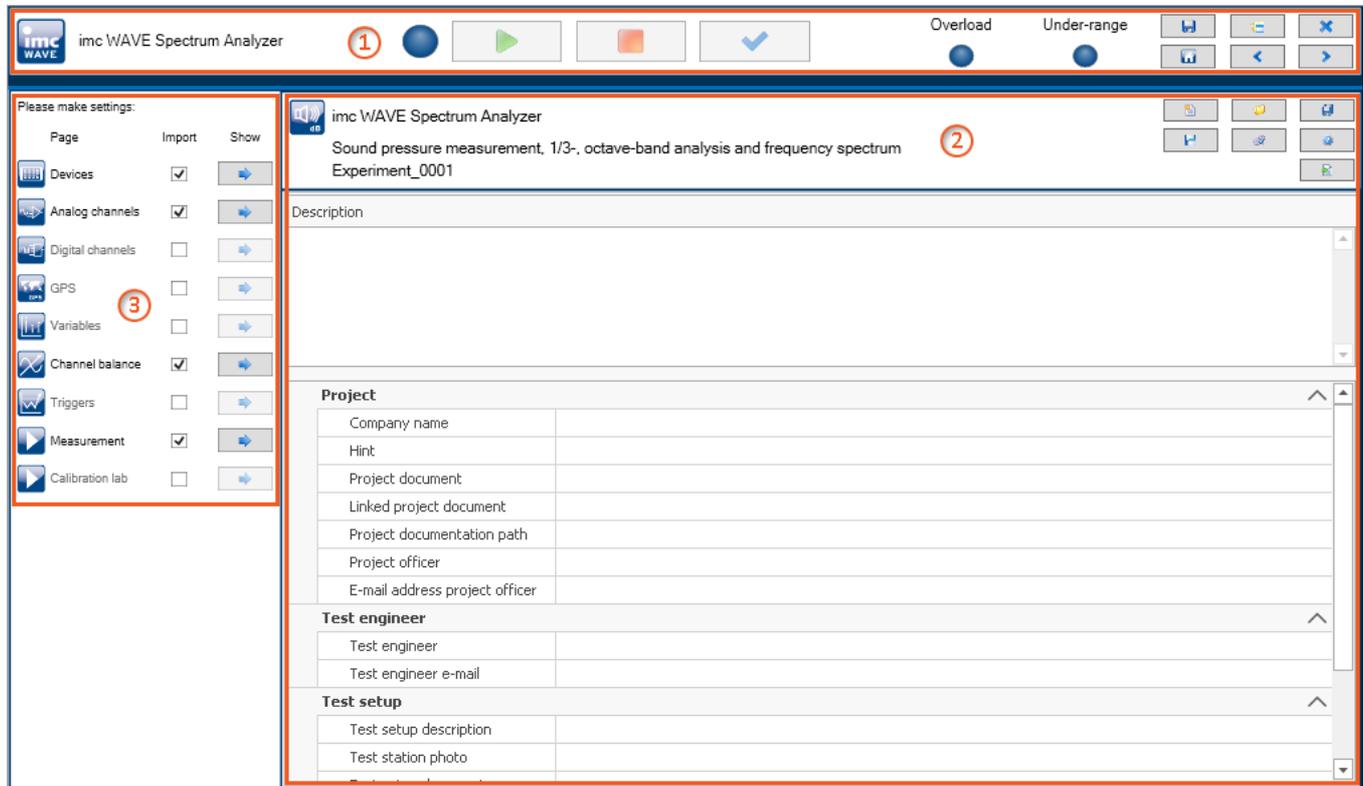
You can import experiment-settings without deleting the underlying elements.

Only elements having the same name are overwritten.

7.4 The User Interface

Homepage

After a successful start of the software the homepage will appear. The following screenshot displays the homepage of the Structure Analyzer:



Homepage

The window is subdivided into three regions:

Region		Description
1	Header	All pages have a header. It provides information and buttons affecting the measurement process.
2	Information and experiment-control	Here you can enter meta-information for documentation purposes. By means of the buttons, you can, among other things, create new experiments, or load others, and open the online help.
3	Page display and selection	Here you can activate pages and browse through them.

Region 1 is displayed by default. The others (2 and 3) are structured differently on each page. The respective structure is described separately.

Brief description of buttons in the header

Button	Description
	Navigates to the homepage.
	Saves the current experiment.
	Opens a dialog for creating and managing additional pages.
	Exits imc WAVE.
	The measurement is started ^[150] for all devices.
	The measurement is stopped ^[150] for all devices. Displayed while the measurement is running.
	<p> Process configuration ^[147]</p> <p>The current configuration is inspected for errors and made available to all components. For this purpose, no connection to the device is required.</p> <p>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 (see chapter Logbook ^[1214]).</p> <p> The symbol for the experiment's Configuration Download ^[149] to the device appears when the CTRL-key is pressed while the focus is on the header or the mouse is positioned over the buttons. This process downloads all settings affecting the device to the measurement system. The measurement starts immediately afterwards.</p>
	Plays back ^[83] a recorded measurement.
	Navigation: Browses one activated page backward.
	Navigation: Browses one activated page forward.

Brief description of buttons in the Homepage

Button	Description
<input checked="" type="checkbox"/> 	Browses to the respective page, e.g. Analog Channels ^[237] . Displayed if the page is activated.
<input checked="" type="checkbox"/> 	Activates/deactivates a page. Pages which are not needed or not to be displayed can be hidden in this way. Thus, unintended changes can be prevented.
	Creates a new experiment ^[73]
	Opens an existing experiment
	Saves the current experiment under ^[73] a new name
	Saves the current project
	General options ^[1237]

Button	Description
	Open user's manual
	Meta-information which has been entered is transferred. The information is saved in variables in the Data Browser.

Title bar- Function overview

The fullscreen window has its own title bar (menu). A **variety of functions** can be called from this menu.

	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.
	This button minimizes the program .
	Thus button closes the program .
Data Browser	This button  causes the free-floating Data Browser  to be shown.

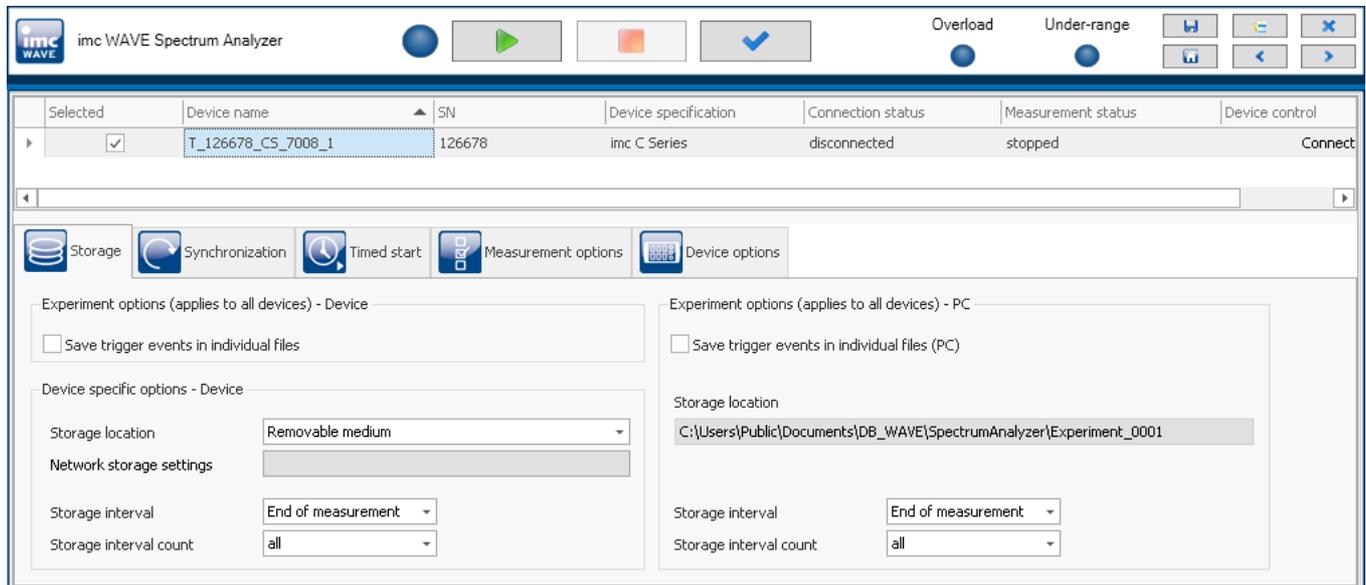
Additional functions can be reached via the context menu or via the button  in the expanded menu.

Menu item	Description
 Login	Logs in a user
 Logout	Logs a user out
 Info	Opens the dialog with version information
 Product Configurator	Opens the dialog for product configuration

7.4.1 Setup panel pages for device configuration

Devices

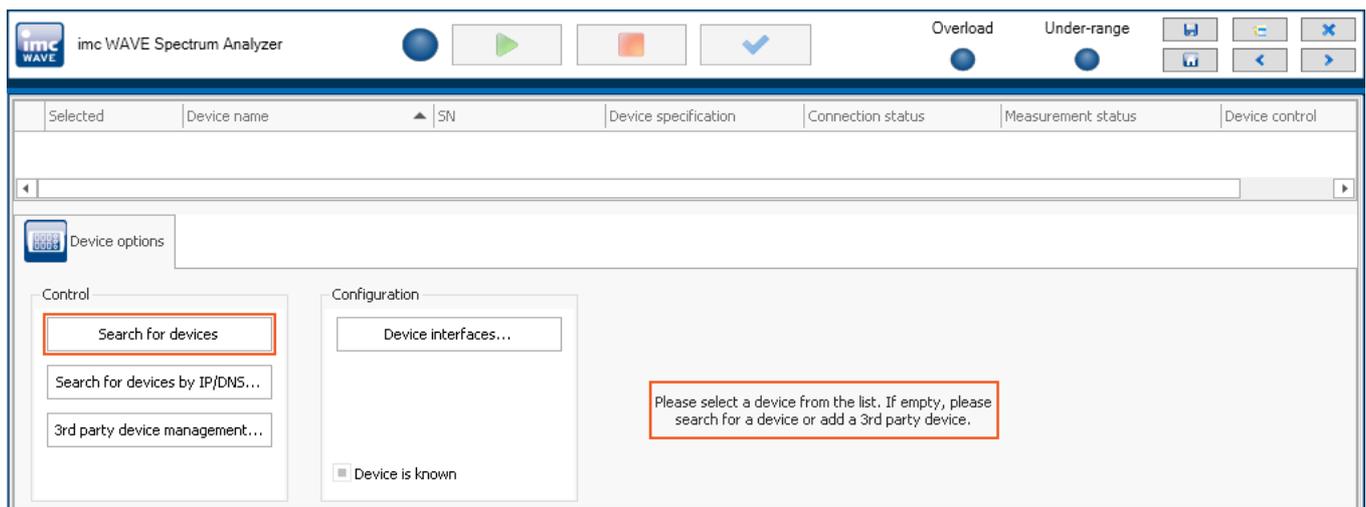
The page "Devices" enables you to perform the first steps toward selecting, configuring, and making settings for your device.



Device search

Once the software has started, all dialog pages on the Devices page, except the "Device options", are hidden unless a device is selected. Thus you can continue directly with the device search. An introductory notification appears.

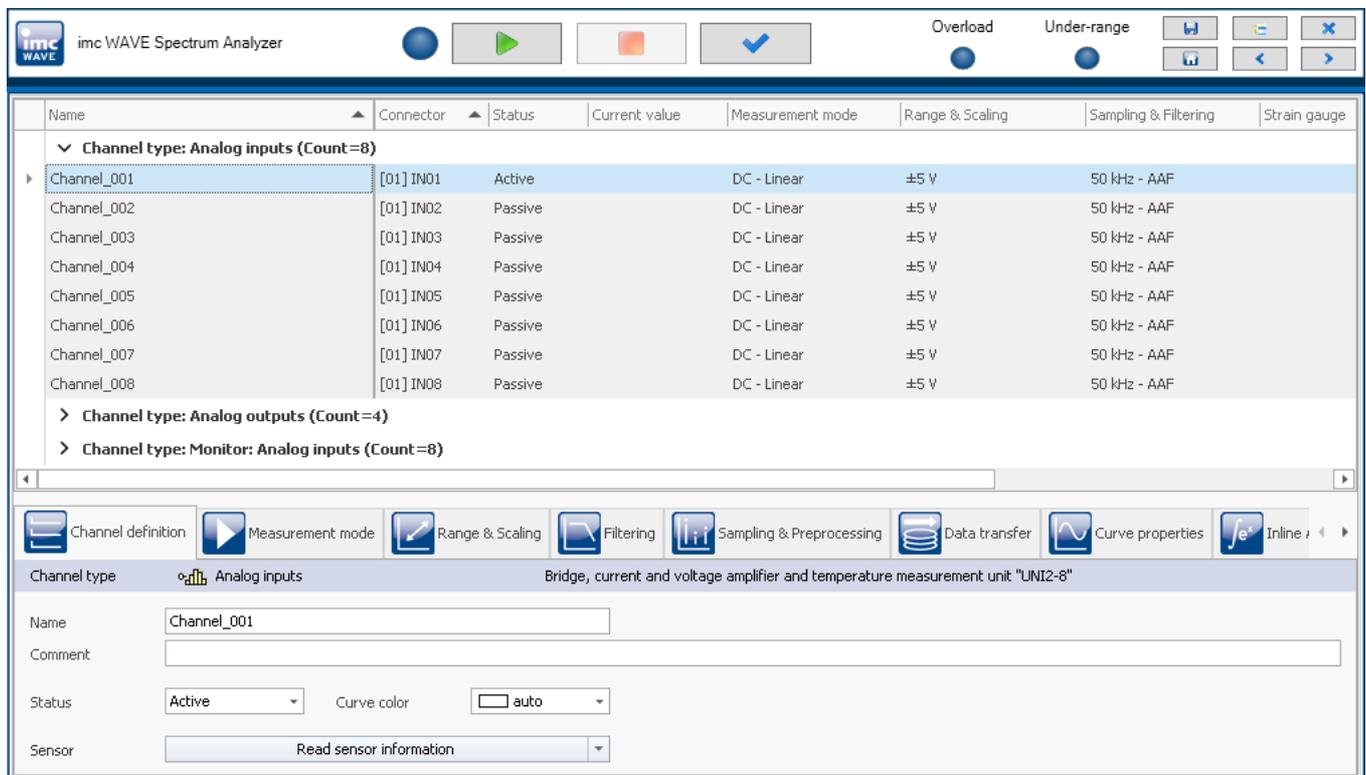
Button: "Search for devices". In a device search, the system searches through the network for all appropriate devices. Subsequently, all devices found are listed.



In the chapter "[Configuring Devices](#)^[185]", you will find a detailed description of the settings: [Storage](#)^[187], [Synchronization](#)^[188], [Timed start](#)^[233], [Measurement options](#)^[236] and many more.

Analog/Digital channels, GPS, Variables, Triggers, ...

On these pages, you can make channel-specific parameter settings and configure the triggers.



The screenshot displays the imc WAVE Spectrum Analyzer interface. At the top, there are control buttons for power, play, stop, and check, along with indicators for Overload and Under-range. Below this is a table listing channels. The table has columns for Name, Connector, Status, Current value, Measurement mode, Range & Scaling, Sampling & Filtering, and Strain gauge. The channels are grouped by type: Analog inputs (Count=8), Analog outputs (Count=4), and Monitor: Analog inputs (Count=8). Channel_001 is selected, and its configuration is shown in a detailed view below the table. This view includes fields for Name (Channel_001), Comment, Status (Active), Curve color (auto), and Sensor (Read sensor information). A toolbar at the bottom of the interface contains icons for Channel definition, Measurement mode, Range & Scaling, Filtering, Sampling & Preprocessing, Data transfer, Curve properties, and Inline.

Name	Connector	Status	Current value	Measurement mode	Range & Scaling	Sampling & Filtering	Strain gauge
Channel type: Analog inputs (Count=8)							
Channel_001	[01] IN01	Active		DC - Linear	±5 V	50 kHz - AAF	
Channel_002	[01] IN02	Passive		DC - Linear	±5 V	50 kHz - AAF	
Channel_003	[01] IN03	Passive		DC - Linear	±5 V	50 kHz - AAF	
Channel_004	[01] IN04	Passive		DC - Linear	±5 V	50 kHz - AAF	
Channel_005	[01] IN05	Passive		DC - Linear	±5 V	50 kHz - AAF	
Channel_006	[01] IN06	Passive		DC - Linear	±5 V	50 kHz - AAF	
Channel_007	[01] IN07	Passive		DC - Linear	±5 V	50 kHz - AAF	
Channel_008	[01] IN08	Passive		DC - Linear	±5 V	50 kHz - AAF	
Channel type: Analog outputs (Count=4)							
Channel type: Monitor: Analog inputs (Count=8)							

Channel type: Analog inputs
Bridge, current and voltage amplifier and temperature measurement unit "UNI2-8"

Name: Channel_001
Comment:
Status: Active
Curve color: auto
Sensor: Read sensor information

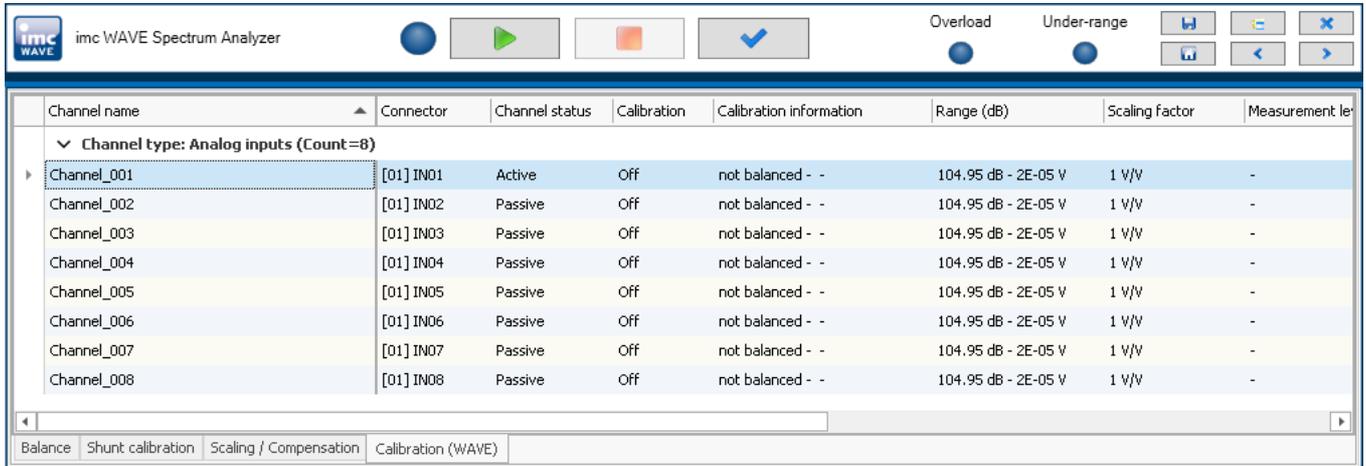
Example: Analog channels

Reference

In the chapter "[Setup - Device configuration](#)¹⁴⁵¹", you will find the detailed description of the individual Setup pages which are displayed here.

Channel balance

On the page "*Channel balance*", channels can be balanced.

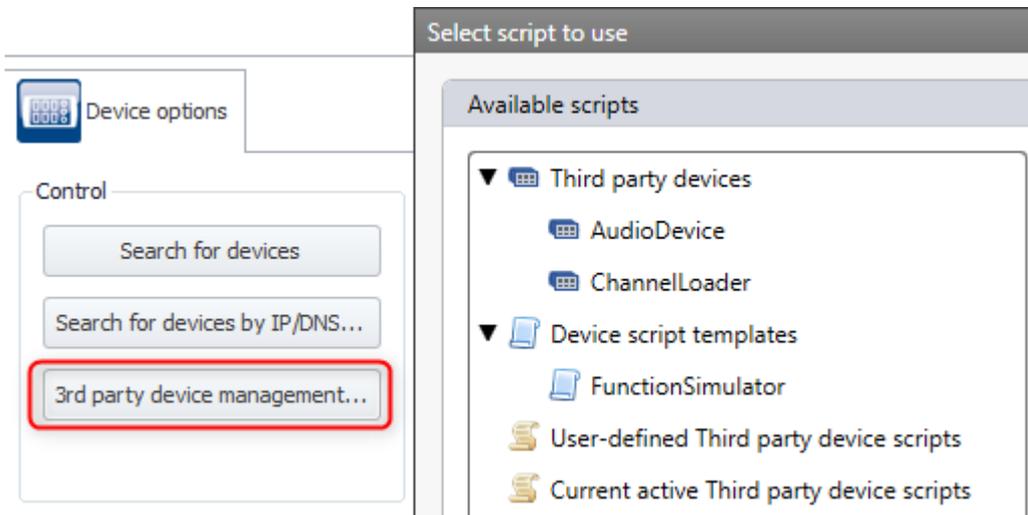


Reference

In the chapter "[Channel balance](#)²⁷⁵", you will find the detailed description of the Balancing-pages.

7.4.1.1 3rd party devices

Along with such measurement devices as imc CRONOSflex or -compact, imc WAVE also allows additional signal sources to be included.



These 3rd-party devices (3PDI) enable certain limited measurements and tests to be performed. The channels of these 3rd-party devices can be played back later and calculations can be performed on them. As well, the results can be re-calculated. However, functions performed by the imc hardware such as triggers, balancing, imc Online FAMOS, and data saving to the device are not available for these 3rd-party devices.

The button for opening "*3rd-party device management*" is located below that for the "[device search](#)⁷⁹".

The following 3rd-party devices are available:

Device	Description
AudioDevice ⁴⁶¹	With the 3rd-party "AudioDevice", it is possible to use the computer's audio devices (such as the microphone input) as a data source.
ChannelLoader ⁴⁶²	The 3rd-party "ChannelLoader" enables files in the imc-format to be played back as a signal during a measurement.
FunctionSimulator ⁴⁶³	The 3rd-party "FunctionSimulator" simulates various signal types (sine, cosine, trapezoid, square wave, ...).

It is not possible to add more of one's own 3rd-party devices.

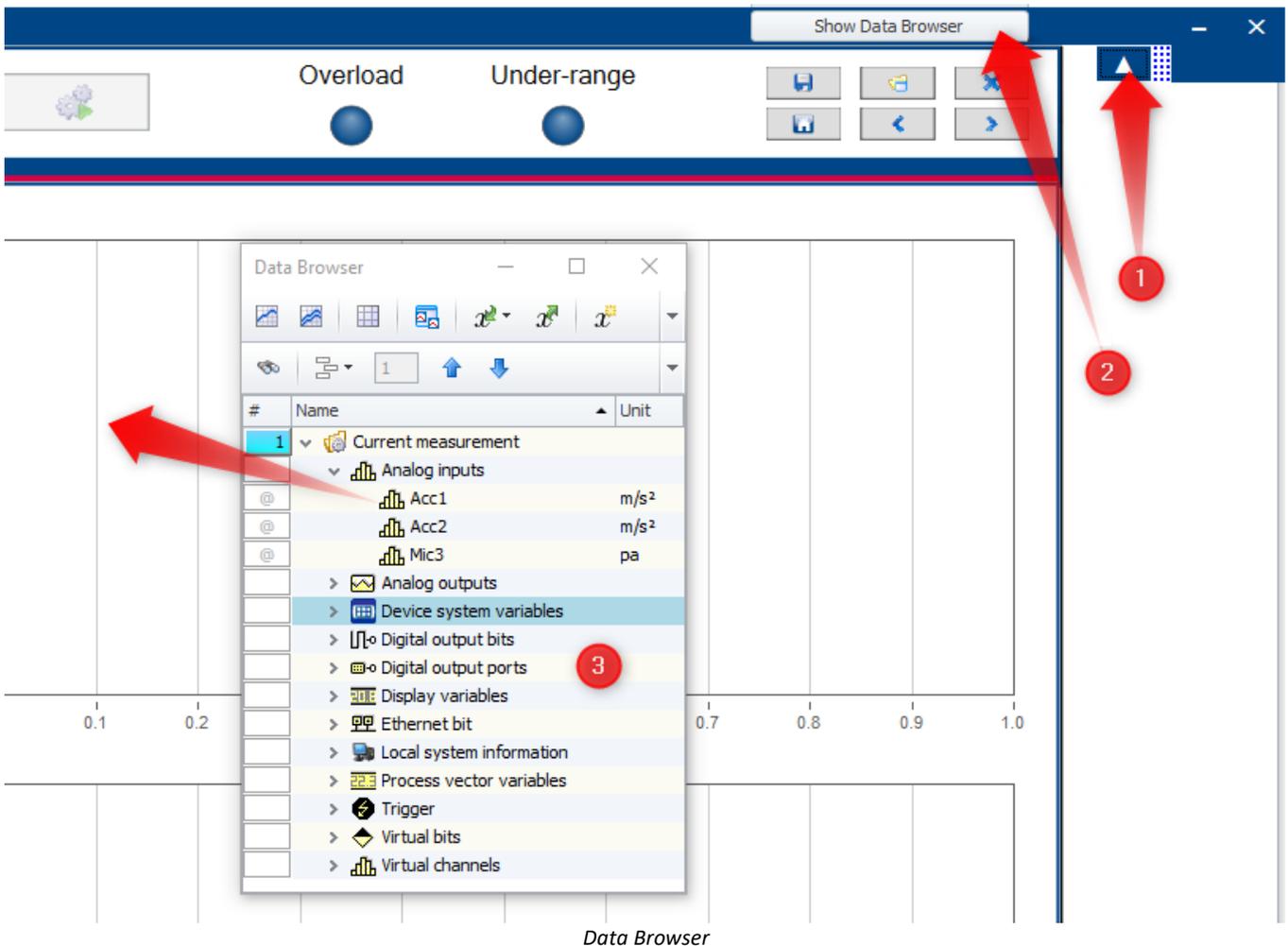
! **Note** For existing experiments

The button for opening the "3rd-party device management" is found with the "Device search". For this purpose, the Devices-page has been revised. In existing experiments, please replace this page manually in order to obtain the latest edition.

To do this, temporarily hide the Devices-page on the "homepage", and then open it again.

Attention: This will undo any manual modifications of the column alignments on the page.

7.4.2 Data Browser



The Data Browser ③ presents all active channels and variables. As a free-floating dialog, it is independent of the page selected. The window can be closed by clicking on the "X" at the top right. To open, click on the triangle ①, and then on the button "Show Data Browser" ②.

By double-clicking on the triangle to the left of a channel's list entry, or by using its associated context menu (opened by right-clicking the mouse over it), you can open the channel in a free-floating curve window.

It is possible for settings which have just recently been changed to not yet be updated in the Data Browser. You can force the system to refresh the display of settings by clicking on the button "Process".



Reference

Playing back recorded measurements

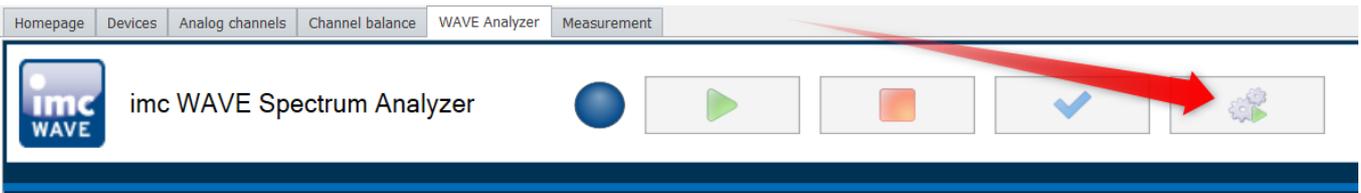
Recorded measurements can also be played back from the browser with a right click. Further information can be found in the section "[Playback](#)".

7.4.3 Playback

The Spectrum Analyzer offers a playback function for recorded measurements.

- One use for this is to **generate calculated channel data streams based on different parameters**. When the playback speed is set to "Maximum", the duration is limited only by the computer's performance.
- Additionally, it is possible to listen to signals using the computer's sound card.

The function is actuated by a button appearing along with the Start and Stop (measurement) buttons.



Click on the "Playback" button



Notes

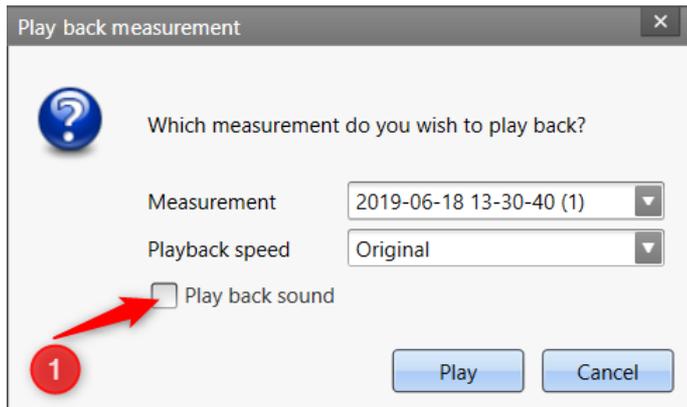
Limitations

The following limitations currently apply:

- supported devices: devices of the [firmware group A](#) (imc DEVICES) and the 3rd-party devices (however not devices belonging to the [firmware group B](#))
- channel-data type: double/float-channels or int16-channels
- equidistant data (no TSA-data, no DI-ports)
- no triggered data (no data containing more than one event or which start at a different time than the "BaseTrigger")
- for evaluation purposes, the analyzer Inline Analysis is supported

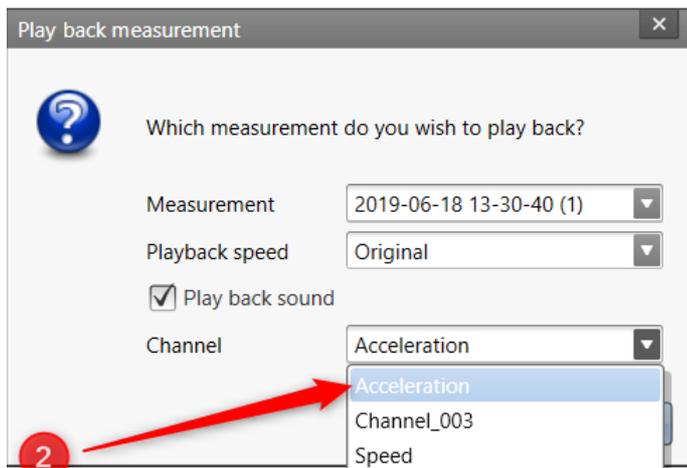
7.4.3.1 Listening to the recorded data

If any data are already recorded for the experiment, you can click on the button Playback. The dialog entitled "Play back Measurement". For signals in the audible range, activate the checkbox "Play back sound" ①.

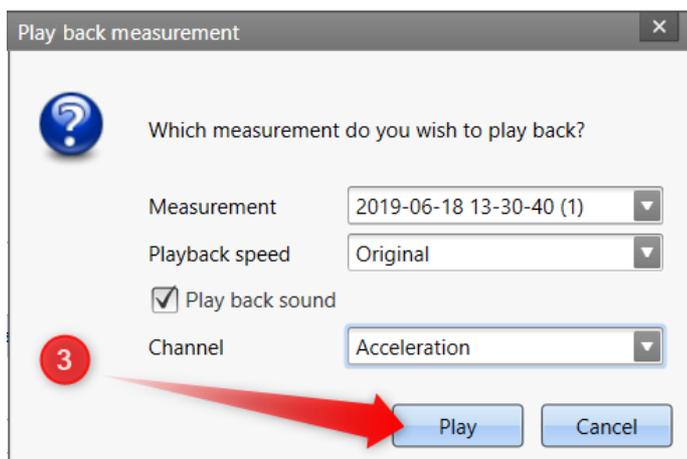


To listen to a channel's data, activate the checkbox "Play back sound"

Since it is always only possible to listen to one signal, a selection box appears, which offers the channels available ②. When you click on the button "Play" ③, the signal selected is outputted audibly.



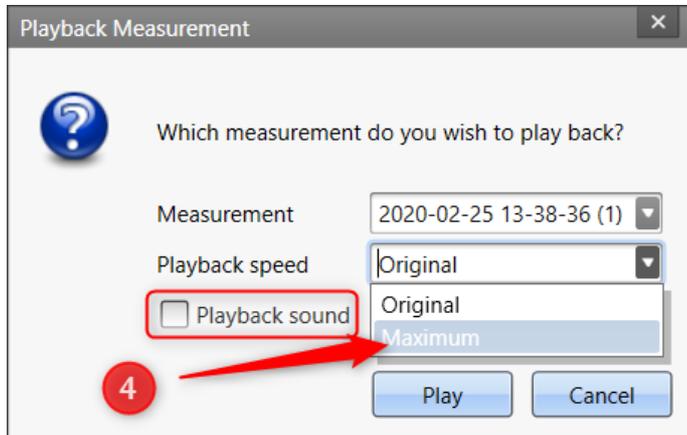
The channel selection list appears. Select the channel desired.



By means of the button "Playback", the channel is audibly outputted via the sound card.

7.4.3.2 Recalculating derived channels

Since the calculation does not need to be run at the "Original" speed, the *Playback speed* can be set to "Maximum" ④. Alternatively, a multiple of the original speed can be entered in the drop-down list. This choice is only available if the checkbox "Play back sound" is deactivated. All recorded input channels are used; no selection of channels is needed.



The data are played back at maximum speed in the Measure-dialog.

The Data Browser also allows replay and calculation of derived channels, see [following chapter](#) ⁸⁶.

Possible application

When the count of input channels is moderate and there are only some derived channels, the "maximum" playback speed is significantly faster than the original speed. When there are many input channels and very many derived variables, however, the calculation may require more time than the measurement itself. Thus, the playback function can be used to perform advanced calculations even after the measurement.

! Notes

Re-calculation of derived channels

Upon each new calculation generated by playback, a revision folder is created as a sub-folder of the measurement folder.

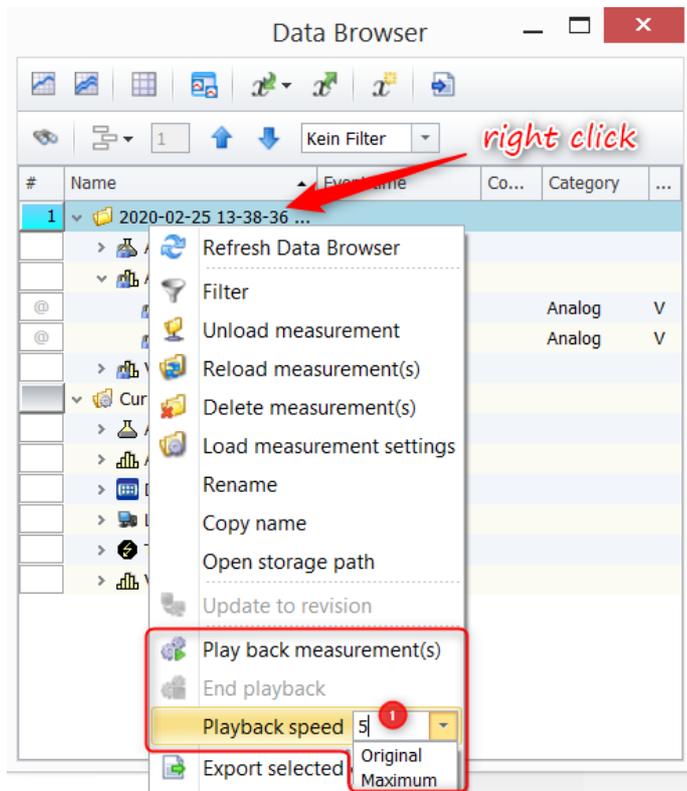
! Warning

Sampling rate for recalculation

Note that the recalculating by the playback function always uses the **sampling rate of the channel set in the setup**. If you change the sampling rate **after** a recording, the derived channels will be calculated based on a different and therefore wrong sampling rate!

7.4.3.3 Direct playback from the Data Browser

You can play back a recorded measurement directly via the context menu of the Data Browser:



Right mouse click on the completed measurement directory allows you to play back the data.

In addition to the original *playback speed*, you can play the data much faster. To do this, overwrite the input field 1 in the drop-down list. Only integer entries are possible. Select "*Maximum*" to set the highest possible playback speed. Note the warning about recalculation in the [previous chapter](#) ⁸⁵.

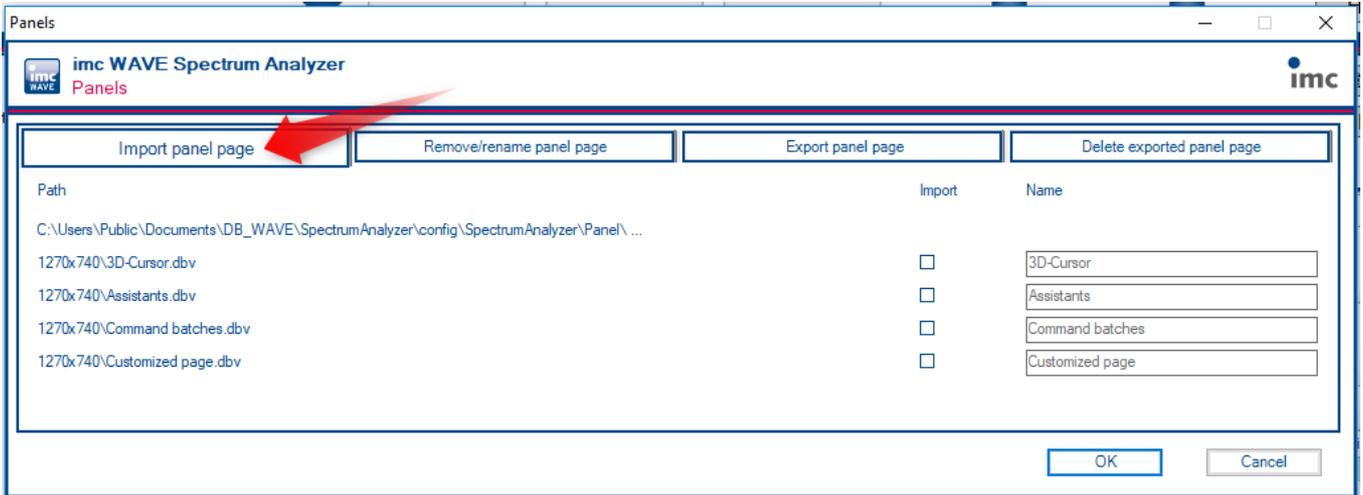
7.4.4 Managing additional pages



You can import and export additional pages, as well as rename or delete them. To do this, open the Panels dialog by clicking on the symbol button shown.

Importing a Panel page

imc WAVE provides various Panel pages which you can add by means of the following dialog.

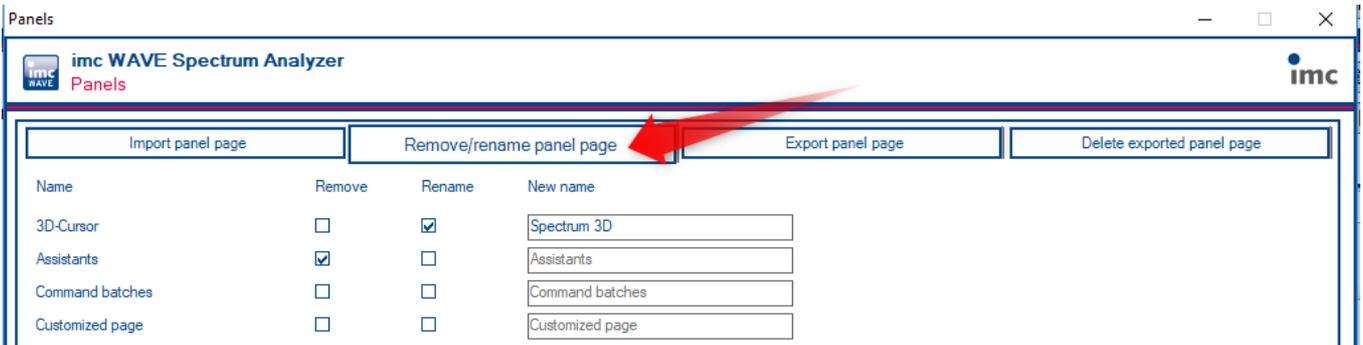


Importing Panel pages

Select the page desired in the column "Import" and confirm by clicking "OK".

Deleting/renameing a Panel seite

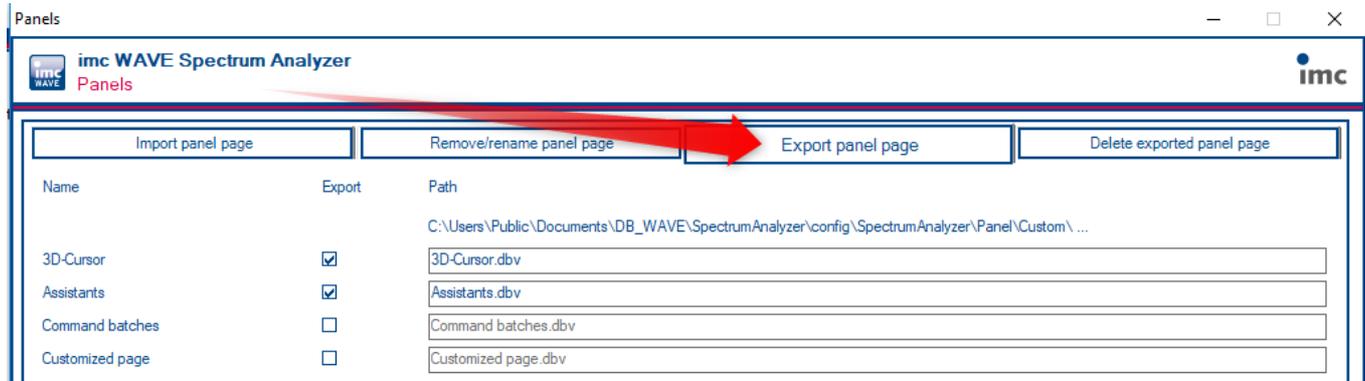
Pages which have been imported can also be removed again. In the process, the pages selected are only removed from the project, not deleted from the hard drive. As well, it is possible to rename the page.



Deleting and renaming Panel pages

Exporting a Panel page

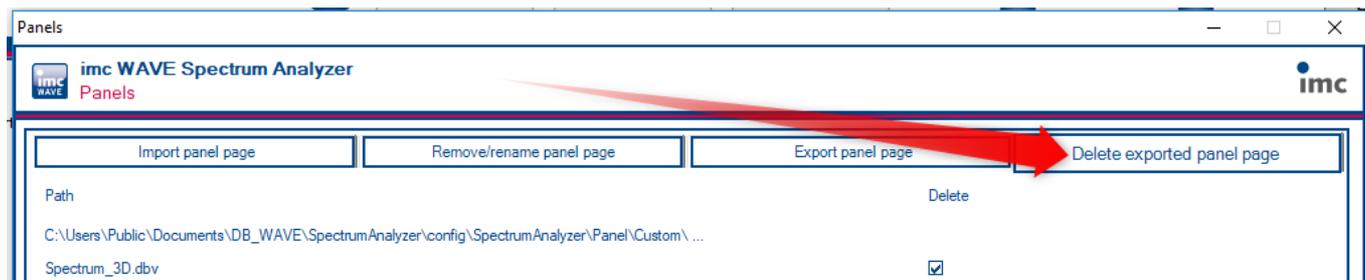
A page which has been edited can be exported under a new name.



Exporting a Panel page

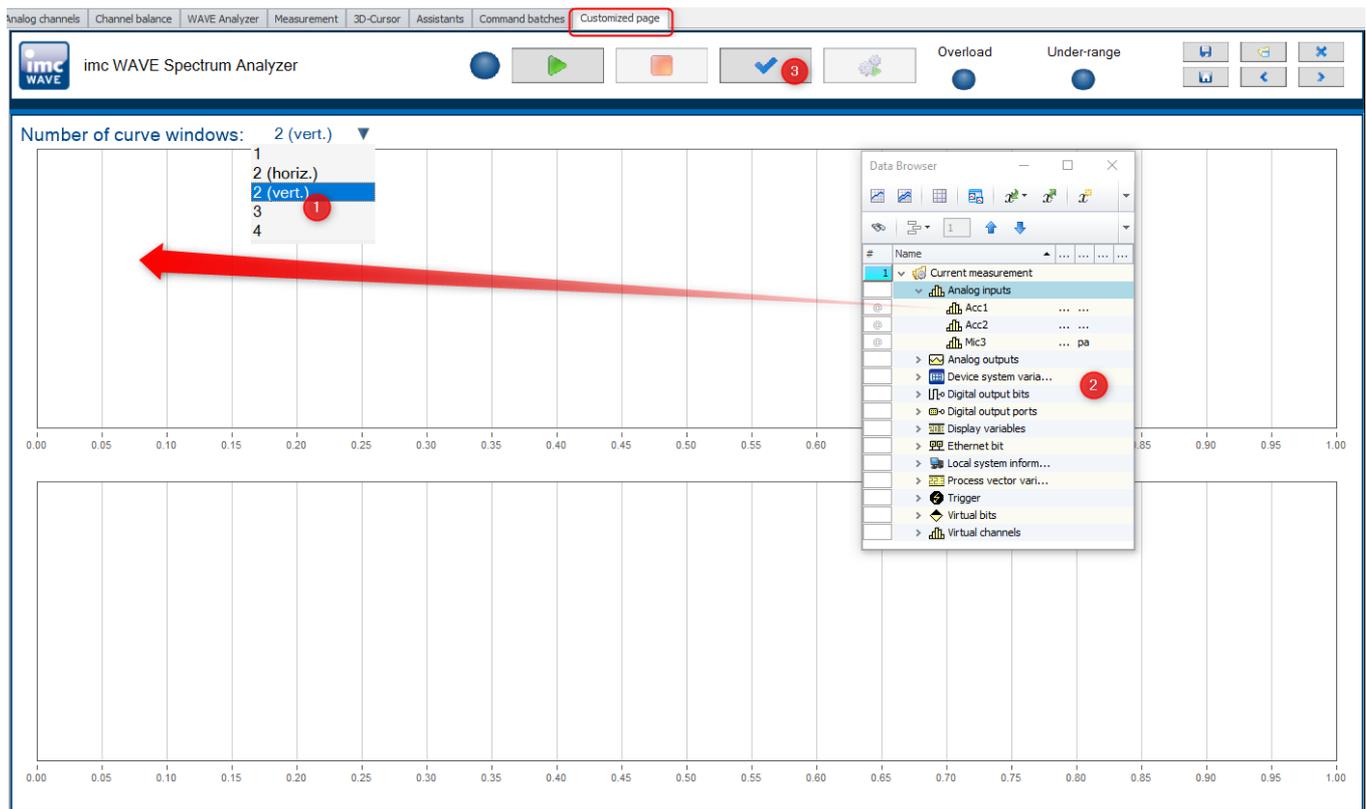
Deleting a Panel page

Previously exported Panel pages can also be deleted. To do this, select the DBV file in the column "Delete" and confirm by clicking on OK. The file will then be deleted from the hard drive.



Deleting Panel pages

7.4.4.1 Customized page

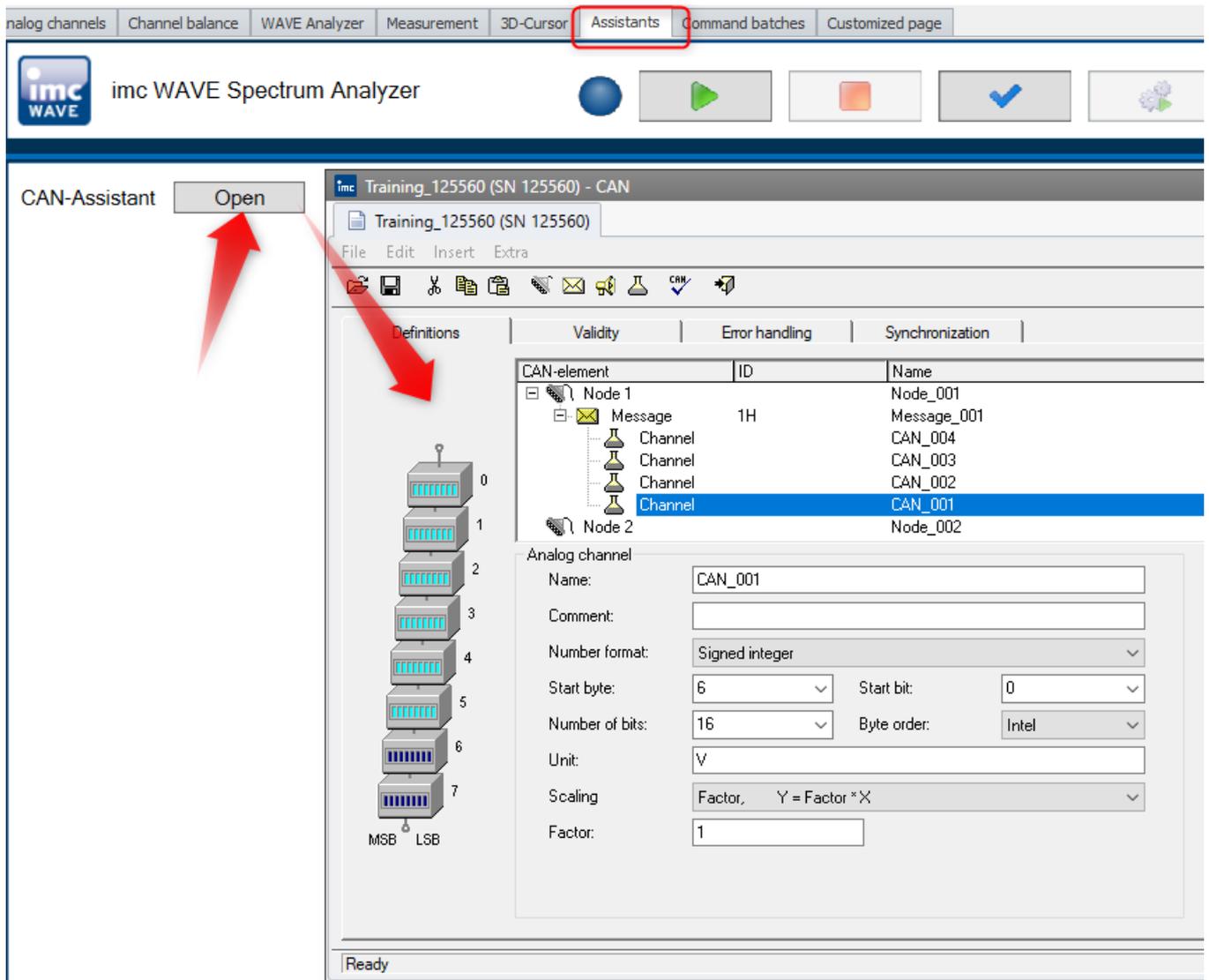


Customized page

The extra page "Customized page" can be displayed with up to four curve windows. To do this, select the desired arrangement from (1).

From the Data Browser (2), the channels are assigned to the curve windows. If the Data Browser does not yet display the edited settings, click on "Process" (3), which causes the Data Browser to refresh.

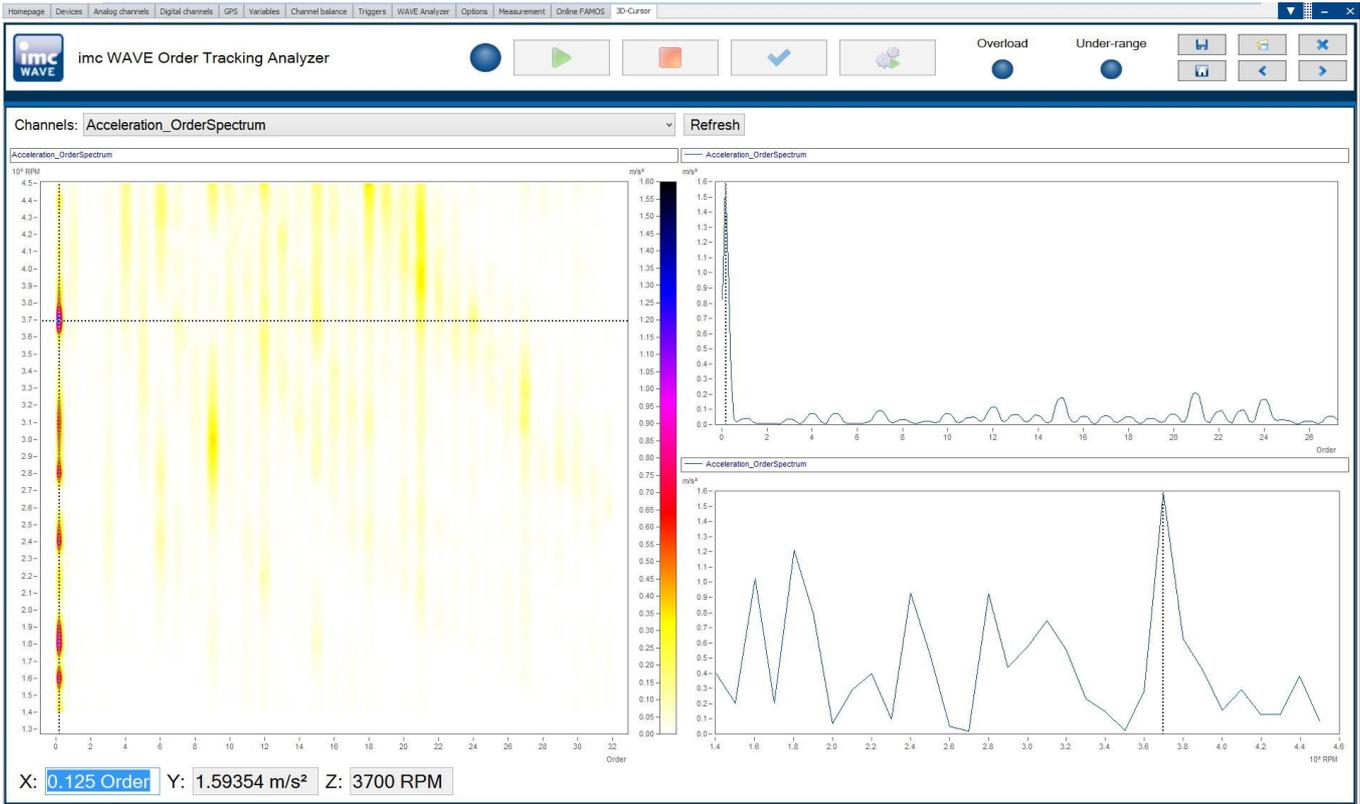
7.4.4.2 Assistants



If the device in use has fieldbus boards, these are configured by means of their respective Assistants. For this purpose, the extra page "Assistants" lists all available Assistants.

Channels defined by performing the configuration are set up in imc WAVE once the Assistant is closed, and are then available as Fieldbus inputs in tab "Analog Channels" and in the Data Browser.

7.4.4.3 3D-Cursor



7.4.4.4 (1/3) Octave labeling

In order to correctly display a spectrum calculated in the Spectrum Analyzer as a derivative channel, the option "Log.Axis" is provided.

By means of this option, a data set is calculated which displays the logarithmic 1/3-octaves. For this purpose, set up a curve window as follows:

More Channels in Curve Window

XY-plot

Axis dialog

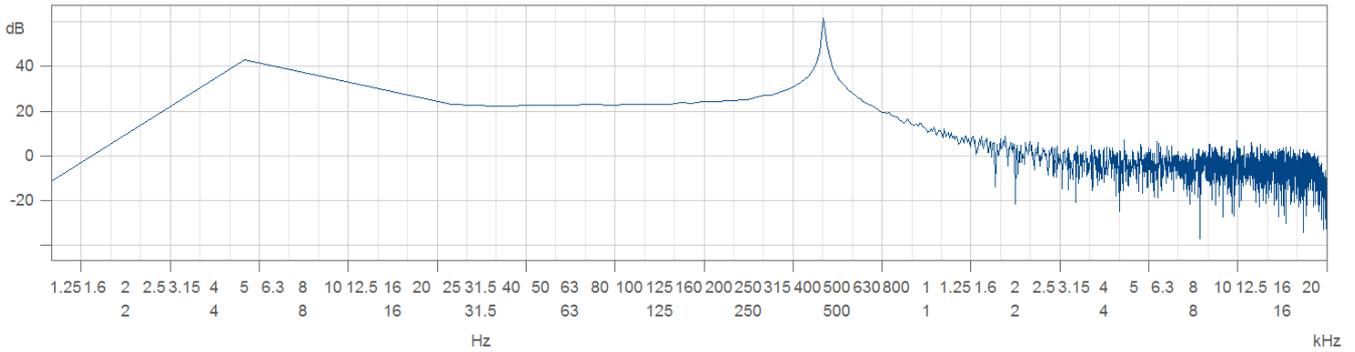
The curve window must contain both the spectrum and its data set with the suffix *_XAxis_log*.

Open the curve window dialog "More Channels in Curve Window" and drag the data sets from the list at right to the setup at left.

The list of *Available Channels* can be filtered by entering *"*XAxis*"* under *Wildcards*.

Next, combine the two data sets in an *XY-plot*

Using the button "Topic", go to the axes dialog and set the X-axis to *Third/octave labeling*



Logarithmic representation with third octave/octave labeling

7.4.4.5 Command batches

Homepage | Devices | Analog channels | Channel balance | WAVE Analyzer | Measurement | Calibration laboratory | **Command batches**

imc WAVE Spectrum Analyzer

Batch: **Command_Batch_1** Execute

Name	Hint
Command_Batch_1	
Execute commands	Occurs when...
Run imc FAMOS sequence: imc FAMOS sequence	...; SoundPre

Context menu options:

- New
- Cut
- Copy
- Paste
- Delete
- Start
- Start (from selected command to end of group)
- Start (only current command)
- Delete all
- Expand / collapse

Flow control options:

- Before
- Below
- After

Scripts/imc FAMOS options:

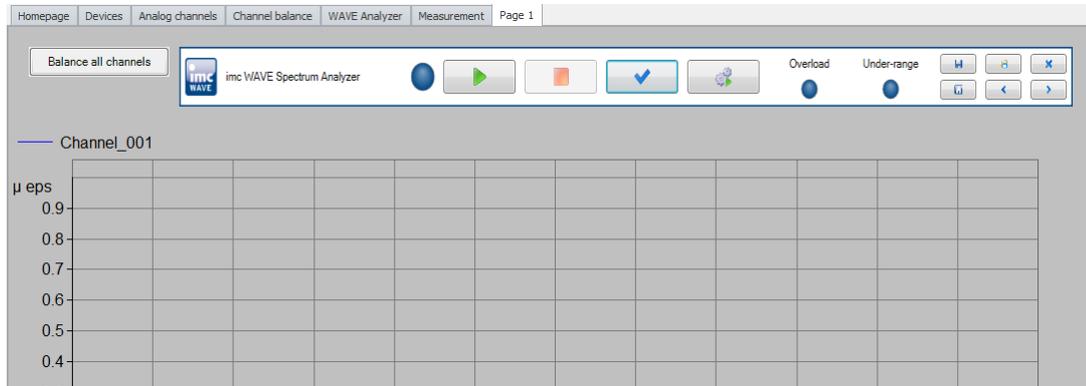
- Run imc FAMOS sequence
- Execute imc FAMOS project

Run analysis automatically upon:

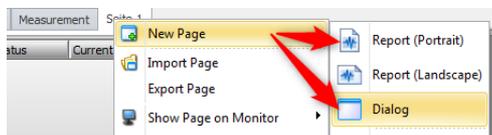
Event name	Active
Experiment_Loaded	<input type="checkbox"/>
Devices_AfterCheckConfiguration	<input type="checkbox"/>
Devices_Stopped	<input checked="" type="checkbox"/>
WAVE_Results_Saved	<input type="checkbox"/>
OnError	<input type="checkbox"/>

7.4.5 Creating additional editable Panel pages

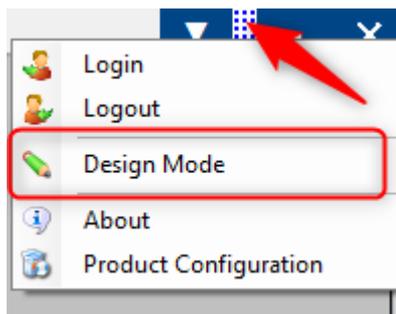
imc WAVE allows additional user-defined Panel-pages to be created for the user interface. For instance, extra report pages or measurement dialogs containing widgets and actions can be designed.



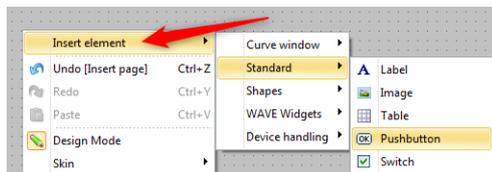
Example: user-defined page



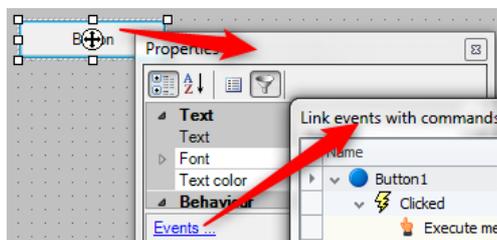
To do this, open the context menu next to one of the page tabs and select the item: "New page".



On the new page, open the context menu over the bar at the top. Here, select "Design Mode".



Next, you can modify and layout the new page. New widgets are added to the page via the context menu.



Beside the display widgets, there is also a button by means of which various commands can be run. The Commands-list has no limits. It provides access to various imc STUDIO actions. Note that not all of these are useful/applicable in imc WAVE.

7.5 Spectrum Analyzer

Spectrum analysis is a widespread pursuit which, over many years, has introduced a number of different calculated quantities of varying applicability. In the wake of the progress of PC's, certain algorithms previously impracticable due to their complexity have become feasible.

Spectrum analysis generally refers to the transformation of a time-domain signal to a complex variable domain reflecting the time-domain signal's composition. Probably the most familiar and widespread algorithm for spectrum analysis is Fourier transformation. But there are many other algorithms which enable representation of the signal in the complex variable domain.

Among other ways, these algorithms can be distinguished by whether they completely map the signal into the complex variable domain or only partially, so that unique reverse transformation back to the time domain is no longer possible.

This latter class of algorithms includes the 1/3-octave and octave filter banks, which by merging frequency ranges and by averaging RMS-values fail to provide phase data, and thus prevent reconstruction of the time-domain signal. However, these algorithms have the advantage over FFT-transformation of offering substantial data compression.

What calculated quantities are available?

The Spectrum Analyzer offers the following output quantities:

Term	Description
FFT	Fourier spectrum of input signal
Filter banks	
1/3-octaves	results returned by a filter bank of 1/3-octave filters
Octaves	results returned by a filter bank of octave filters
RMS	plot of the input signal's RMS-value
LEQ	plot of the input signal's energetic mean value

The Spectrum Analyzer performs spectral analysis in real time on time-domain signal from acoustic sensors such as microphones and accelerometers. This especially includes the calculation of octave and 1/3-octave spectra. The Spectrum Analyzer, consisting of "imc WAVE Noise", "imc WAVE Vibration" and "imc WAVE Rotation", provides a number of settings options for the purpose of parameterizing these calculations in accordance with industrial standards. These include weighting in both time- and frequency domain, with standard procedures, filters, window functions, etc.

The Inline Analysis "imc WAVE Structure" offers functions for 2 channel spectrum analysis, e.g. frequency response function or coherence.

Furthermore, the complete functionality of a sound level meter is included, that entirely covers the associated industrial standard.

Calculation functions and required licenses

With the corresponding license, you are able to activate the following analyzers:

Name and license required	Description
imc WAVE Noise	<p>Functions for noise analysis, e.g. sound pressure level, Leq, sound intensity, FFT-spectra, 1/3-octave and 1/1-octave spectrum.</p> <ul style="list-style-type: none"> • 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
imc WAVE Rotation	<p>Functions for analysis of rotating machinery</p> <ul style="list-style-type: none"> • 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
imc WAVE Structure	<p>Functions for 2-channel analysis in the frequency domain</p> <ul style="list-style-type: none"> • 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
imc WAVE Vibration	<p>Functions for vibration analysis of human vibration and machine diagnostics</p> <ul style="list-style-type: none"> • 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

imc WAVE Noise

Functions	Description
Basic analysis functions	<ul style="list-style-type: none"> • 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

imc WAVE Rotation

Functions	Description
Basic analysis functions	<ul style="list-style-type: none"> • 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

imc WAVE Structure

Functions	Description
Basic analysis functions	<ul style="list-style-type: none"> • 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

imc WAVE Vibration

Functions	Description
Basic analysis functions	<ul style="list-style-type: none"> • 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

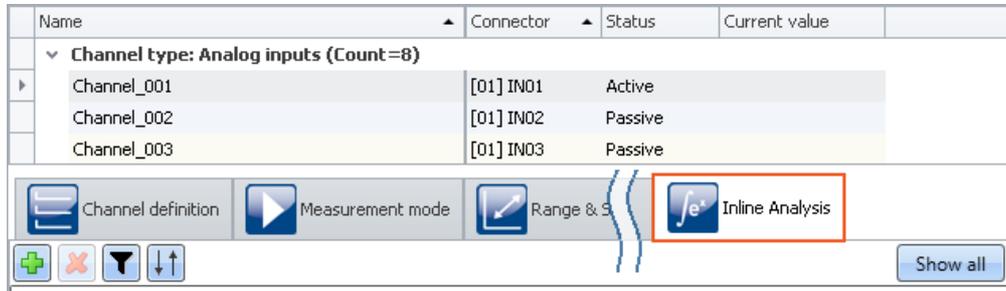
- standards-conformant measurements for human vibration and machine diagnostics
- acceptance and certification measurements in conjunction with impact of vibration on humans
- in-vehicle seating comfort: Measurements for qualification and optimization
- hand-arm vibration: handheld machinery
- acceptance 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

7.5.1 Derived channels - Inline Analysis

By using the following dialog, it is possible to compute a derivative or calculated channel from a input channel.

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)

7.5.1.1 Inline Functions - operation

Display in the dialog

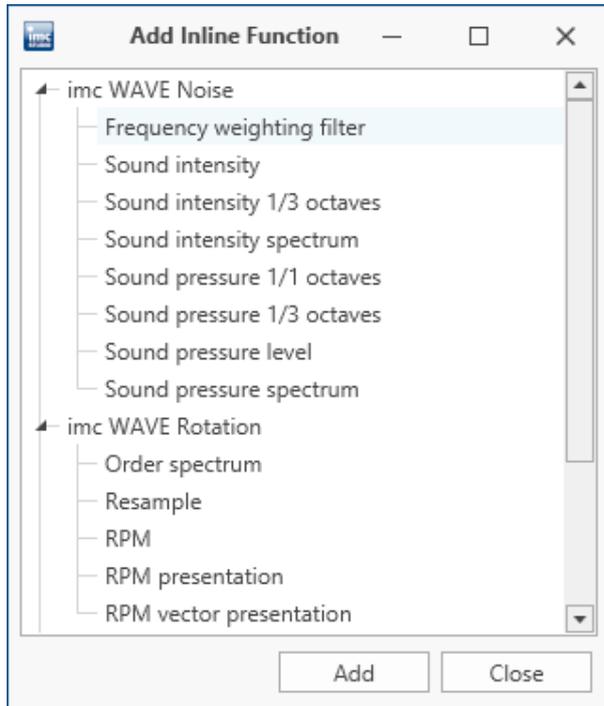
The dialog always displays all Inline Functions pertaining to the channels selected. If any selected channel has no Inline Functions, that area is empty. Clicking on the button "Show all" opens a tool window in which all Inline Functions for all channels are displayed.

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



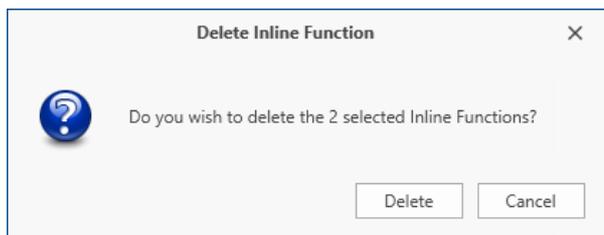
Adding new Inline Functions

- 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.

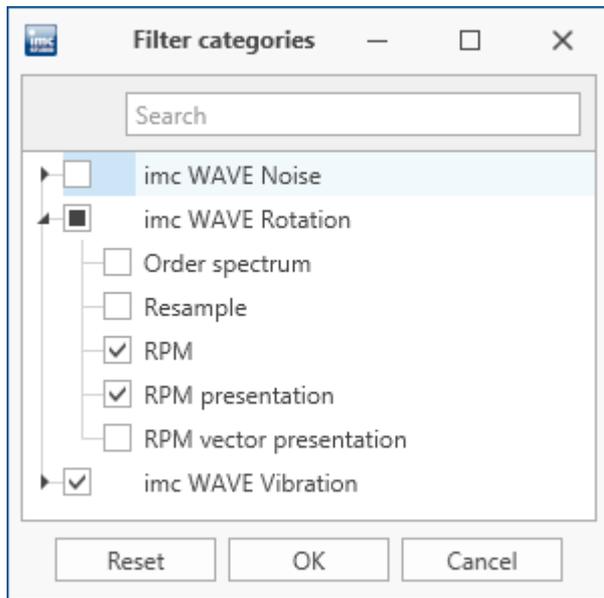
Deleting Inline Functions

- 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

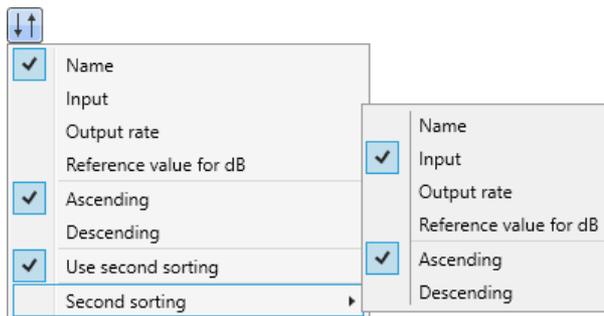
Filtering Inline Functions



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	<p>The parameter: "<i>Name</i>" defines the output-variable's name. This appears in the Data Browser and can be viewed on the "Measurement" page.</p> <p>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.</p>
Input	<p>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 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.</p> <p>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.</p>

Additional parameters may also be available depending on the Inline Function type.

7.5.1.2 imc WAVE Noise

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	<p>Weighted time signal</p> <ul style="list-style-type: none"> • Frequency-weighting: Z (without) or A, B or C. The result is also a time-domain signal.
Sound intensity level	<p>Determination of sound intensity by means of a sound intensity probe according to the two-microphone technique.</p> <ul style="list-style-type: none"> • Frequency-weighting: Z (without) or A, B or C • Distance (spacer): distance between microphones • Density: air density • Averaging: <ul style="list-style-type: none"> • <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.

Calculation	Description
Sound intensity 1/3 octaves	<p>Determination of sound intensity as 1/3-octaves by means of measurement by a sound intensity probe according to the two-microphone technique.</p> <ul style="list-style-type: none"> • Frequency-weighting: Z (without) or A, B or C • Distance (Spacer): distance between microphones, length of spacer • Density: air density • Averaging: <ul style="list-style-type: none"> • <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)	<p>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.</p> <ul style="list-style-type: none"> • Frequency-weighting: Z (without) or A, B or C • Distance (spacer): Distance between microphones, length of spacer • Density: air density • Averaging: <ul style="list-style-type: none"> • <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, 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. Observe the the notes on creating the curve window in the section "Managing additional pages⁸⁷".
Sound pressure 1/1 octaves	<p>Leq or LeqT 1/1-octaves: time-averaged sound level, equivalent continuous sound level over the measurement duration or with time interval T</p> <ul style="list-style-type: none"> • for Time-weighting: "Leq from start" or "Leq in interval" • Frequency-weighting: Z (without) or A, B or C <p>Lt 1/1-octaves: time weighted sound level</p> <ul style="list-style-type: none"> • 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	<p>Leq or LeqT 1/3-octaves: time-averaged sound level, equivalent continuous sound level over the measurement duration or with time interval T</p> <ul style="list-style-type: none"> • for Time-weighting: "Leq from start" or "Leq in interval" • Frequency-weighting: Z (without) or A, B or C <p>Lt 1/3-octaves: time weighted sound level</p> <ul style="list-style-type: none"> • for Time-weighting: F (Fast), S (Slow), I (Impulse) or Peak • Frequency-weighting: Z (without) or A, B or C

Calculation	Description
Sound pressure level	<p>Mean level, equivalent continuous sound level of the frequency-weighted sound pressure signal over the measurement duration or with time interval T</p> <ul style="list-style-type: none"> • for Time-weighting: "Leq from start" or "Leq in interval" • Frequency-weighting: Z (without) or A, B or C <p>Lt and Lt max: frequency- and time-weighted sound level</p> <ul style="list-style-type: none"> • for Time-weighting: F (Fast), S (Slow), I (Impulse) or Peak • Frequency-weighting: Z (without) or A, B or C • Maximum: Maximum of Lt
Sound pressure spectrum (FFT)	<p>Fast Fourier Transform: Amplitude spectrum</p> <ul style="list-style-type: none"> • Frequency-weighting: Z (without) or A, B or C • Result: Output of the results values as: <ul style="list-style-type: none"> • <i>RMS</i>: harmonic as RMS-value • <i>RMS²</i>: harmonic as squared RMS-value • <i>Peak</i>: harmonic as peak value/amplitude • Averaging: <ul style="list-style-type: none"> • <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, Flatop • Overlap: 0%, 10%, 25%, 33.33%, 50%, 66.67%, 75%, 90% • Log. axis: Creates an X-axis for the representation scaled with thirds. See the notes on creating the curve window in section "Managing additional pages". • Diff./Int.: <ul style="list-style-type: none"> • <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 ($2*\pi*f$). • <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 ($2*\pi*f$). • <i>integration</i>: The integral of the amplitude spectrum is found. This is performed by division of the spectrum by the angular frequency ω ($2*\pi*f$). • <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 ($2*\pi*f$). <p>Note: In all derivation and integration procedures, the result for the DC component (frequency: 0 Hz) is 0 and thus has no significance.</p>

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. $f_1 = 8$ Hz and $f_2 = 12500$ 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, delta-z, 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.

7.5.1.3 imc WAVE Rotation

Calculations

Calculation	Description
Frequency spectrum (FFT)	<p>Fast Fourier Transform: Amplitude spectrum</p> <ul style="list-style-type: none"> • Result: Output of the results values as: <ul style="list-style-type: none"> • <i>RMS</i>: harmonic as RMS-value • <i>RMS²</i>: harmonic as squared RMS-value • <i>Peak</i>: harmonic as peak value/amplitude • Averaging: <ul style="list-style-type: none"> • <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. See the notes on creating the curve window in section "Managing additional pages". • Diff./Int.: <ul style="list-style-type: none"> • <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 ($2*\pi*f$). • <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 ($2*\pi*f$). • <i>integration</i>: The integral of the amplitude spectrum is found. This is performed by division of the spectrum by the angular frequency ω ($2*\pi*f$). • <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 ($2*\pi*f$). <p>Note: In all derivation and integration procedures, the result for the DC component (frequency: 0 Hz) is 0 and thus has no significance..</p>
RPM	<p>Determining the RPMs from, for example, squarewave, sinusoidal or incremental encoder signals, with the measurement mode "Impulse Time"</p> <ul style="list-style-type: none"> • 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	<p>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.</p> <ul style="list-style-type: none"> • 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	<p>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.</p> <ul style="list-style-type: none"> • 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)	<p>Determination of the order spectrum</p> <ul style="list-style-type: none"> • 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 $2 * \text{"Maximum Order"}$ divided by "Resolution" • Resolution: distance between lines in the order spectrum

Calculation	Description
Resample	<p>Sampling of the input channel over the angle.</p> <p>The X-unit of the result is "revolutions". The values in the result are to be interpreted as RMS-values.</p> <ul style="list-style-type: none"> • 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	<p>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.</p> <p>The input can be a frequency spectrum or an order tracking spectrum, as well as 1/3-octave and octave spectra.</p> <ul style="list-style-type: none"> • 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.

7.5.1.4 imc WAVE Vibration

Calculations

"L" stands for (vibration-) level.

Calculation	Description
Filter	<ul style="list-style-type: none"> • Filter type: without, Low-pass filter, High-pass filter, Band-pass filter, Band-stop filter • Characteristic: Butterworth, Bessel, Chebychev, Critical damping • Order: integer \leq 100 • Cutoff frequency: up to one-half of the sampling rate • Diff./Int.: <ul style="list-style-type: none"> • <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.

Calculation	Description
Filter rms	<p>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.</p> <ul style="list-style-type: none"> • Frequency-weighting: without or see "Possible filters"¹¹²¹ • 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	<p>In accordance with ISO 10816/20816 - The function has two results channels.</p> <ol style="list-style-type: none"> Result (with specified name) in accordance with the settings for the filtering, time weighting, etc. "Alert"-channel, containing the zone values (A = 0, B = 1, C = 2, D = 3), according to which zone the result (1.) lies in <ul style="list-style-type: none"> • 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: <ul style="list-style-type: none"> • <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> $\text{Crest factor} = \frac{ \text{Input signal} }{\text{Input signal}_{RMS}}$ • 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.: <ul style="list-style-type: none"> • <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) <hr/> <p>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.</p> <hr/> <ul style="list-style-type: none"> • Output rate: max. like input channel, or reduced by integer factor

Calculation	Description
Vibration 1/1 octaves	<p>Leq or LeqT 1/1-octaves: time-averaged vibration level, equivalent continuous vibration level over the measurement duration or with time interval T</p> <ul style="list-style-type: none"> • 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, ...) <p>L_T 1/1-octaves: time weighted vibration level</p> <ul style="list-style-type: none"> • 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	<p>Leq or LeqT 1/3-octaves: time-averaged vibration level, equivalent continuous vibration level over the measurement duration or with time interval T</p> <ul style="list-style-type: none"> • 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, ...) <p>L_T 1/3-octaves: time weighted vibration level</p> <ul style="list-style-type: none"> • 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)	<p>Fast Fourier Transform: Amplitude spectrum</p> <ul style="list-style-type: none"> • Frequency-weighting: without or human vibration filter (e.g. Hx, DIN 45671, ...) • Result: Output of result values as: <ul style="list-style-type: none"> • <i>RMS</i>: harmonic as RMS-value • <i>RMS²</i>: harmonic as squared RMS-value • <i>Peak</i>: harmonic as peak value/amplitude • Averaging: <ul style="list-style-type: none"> • <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. See the notes on creating the curve window in section "Managing additional pages". • Diff./Int.: <ul style="list-style-type: none"> • <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 ($2*\pi*f$). • <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 ($2*\pi*f$). • <i>integration</i>: The integral of the amplitude spectrum is found. This is performed by division of the spectrum by the angular frequency ω ($2*\pi*f$). • <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 ($2*\pi*f$). <p>Note: In all derivation and integration procedures, the result for the DC component (frequency: 0 Hz) is 0 and thus has no significance.</p>

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

7.5.1.5 imc WAVE Structure

Calculations

Calculation	Description
Frequency spectrum (FFT)	<p>Fast Fourier Transform: Amplitude spectrum</p> <ul style="list-style-type: none"> • Result: output of the results values as: <ul style="list-style-type: none"> • <i>RMS</i>: harmonic as RMS-value • <i>RMS²</i>: harmonic as squared RMS-value • <i>Peak</i>: harmonic as peak/amplitude value • Averaging: <ul style="list-style-type: none"> • <i>none</i>: no averaging • <i>Number</i>: Averaging is performed according to the "<i>Avg.-type</i>" selected, over the specified count of spectra • <i>from start</i>: Averaging is performed according to the "<i>Avg.-type</i>" 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. See the notes on creating the curve window in section "Managing additional pages^[87]". • Diff./Int.: <ul style="list-style-type: none"> • <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 ($2*\pi*f$). • <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 ($2*\pi*f$). • <i>integration</i>: The integral of the amplitude spectrum is found. This is performed by division of the spectrum by the angular frequency ω ($2*\pi*f$). • <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 ($2*\pi*f$). <p>Note: In all derivation and integration procedures, the result for the DC component (frequency: 0 Hz) is 0 and thus has no significance..</p>

Calculation	Description
Coherence	<p>Determines the coherence between the input and the output time plots, over the frequency.</p> <ul style="list-style-type: none"> • 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, 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. Observe the the notes on creating the curve window in the section "Managing additional pages".
Cross power spectrum	<p>Determines the cross-power spectrum of the input and output signals over the frequency. The function has two results channels:</p> <ol style="list-style-type: none"> 1. Amplitude spectrum 2. Phase spectrum <ul style="list-style-type: none"> • Input: time plot of input (1st input signal) • Output: time plot of output (2nd input signal) • Averaging: <ul style="list-style-type: none"> • <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, 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. Observe the the notes on creating the curve window in the section "Managing additional pages".
Power spectrum density (PSD)	<p>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:</p> <ol style="list-style-type: none"> 1. Amplitude spectrum 2. Phase spectrum <ul style="list-style-type: none"> • Input: time plot of input (1st input signal) • Output: time plot of output (2nd input signal) • Averaging: <ul style="list-style-type: none"> • <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, 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. Observe the the notes on creating the curve window in the section "Managing additional pages".

Calculation	Description
Frequency response function (FRF)	<p>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:</p> <ol style="list-style-type: none"> 1. Amplitude spectrum 2. Phase spectrum <ul style="list-style-type: none"> • Input: time plot of input (1st input signal) • Output: time plot of output (2nd input signal) • Averaging: <ul style="list-style-type: none"> • <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: <ul style="list-style-type: none"> • <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: $(G_{xy} / G_{xy}) * \text{sqrt}(G_{yy} / G_{xx})$</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. Observe the the notes on creating the curve window in the section "Managing additional pages".

7.5.1.6 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: all active analog channels of the device belonging to the firmware groups A and B (imc DEVICES and imc DEVICEcore). 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.

7.5.2 Tutorial - Spectrum Analyzer

The tutorial below presents the procedure for performing a measurement with the Spectrum Analyzer.

The following devices and accessories are needed:

- An imc measurement device of either the series imc CRONOS or imc C-SERIES, with an Audio amplifier or an amplifier having a similar band width, e.g. ICPU2-8.
- 2 microphones with BNC cables. If the measurement amplifiers do not have any BNC terminals, use a BNC-ICP adapter, e.g. ACC/DSUB-ICP2 or -ICP4
- Microphone calibrator with appropriate microphone adapter

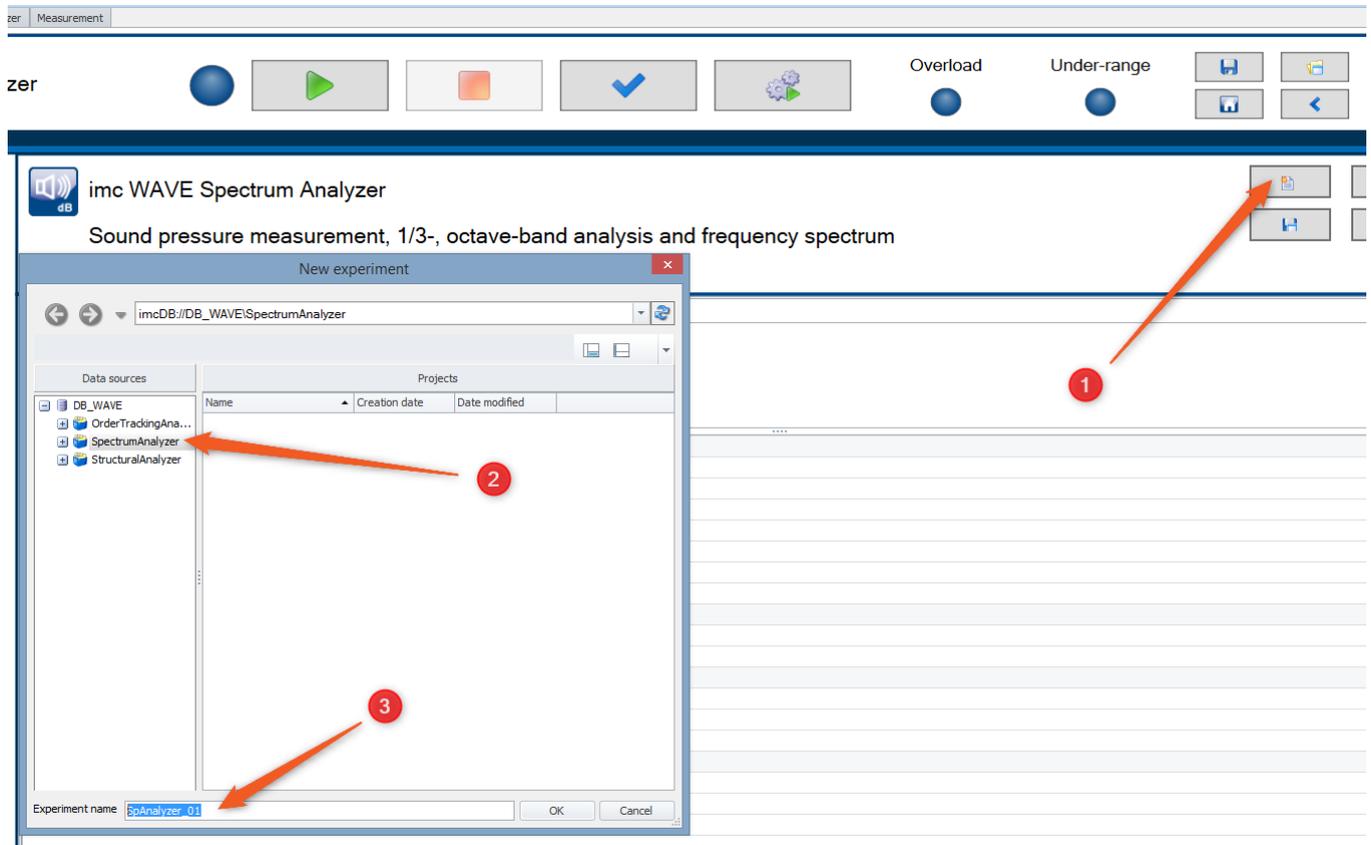


- Connect the measurement device with the PC either directly or via the local network.
- Connect both microphones at the BNC-sockets to the input channels "IN01" and "IN02".

7.5.2.1 Setting up an experiment

1. Create a new experiment named "SpAnalysator_01":

- Start by opening the dialog "New Experiment" (1)
- Next, select the "Spectrum Analyzer" (2) and enter the experiment's name (3).



- Confirm by clicking *OK*. The experiment is created for the purpose of **spectrum analysis**.

2. Selecting a device

- Go to the page "Devices" (1)
- When you click on "Search for devices" (3) under the tab "Device options" (2), the system finds and lists all available imc measurement devices in the network.
- Select the desired measurement device from the list (4).

The screenshot shows the 'imc WAVE Spectrum Analyzer' interface. The 'Devices' tab is selected, indicated by a red circle '1' and an arrow. Below the navigation bar, there are buttons for 'Storage', 'Synchronization', 'Timestart', 'Measurement options', and 'Device options'. The 'Device options' button is highlighted with a red circle '2' and an arrow. Under 'Device options', the 'Search for devices' button is highlighted with a red circle '3' and an arrow. Below this, a table lists various measurement devices. The device 'T_141399_CRFX_400' is selected, indicated by a checked checkbox and a red circle '4' with an arrow pointing to it. The table columns are: Selected, Device name, SN, Device specification, Connection status, and Measurement status.

Selected	Device name	SN	Device specification	Connection status	Measurement sta
<input type="checkbox"/>	imc_CRC_400_142782	142782	imc CRONOScompact		
<input type="checkbox"/>	T_142203_CS_7008_N	142203	imc C Series		
<input type="checkbox"/>	Gerät_M_Fuge	141989	busDAQ-2		
<input type="checkbox"/>	SICAM_Q80_14185X	141852	SICAM Q80		
<input type="checkbox"/>	H_141692_CS_7008_1	141692	imc C Serie		
<input checked="" type="checkbox"/>	T_141399_CRFX_400	141399	imc CRONOSflex	disconnected	stopped
<input type="checkbox"/>	imc_CRC_400_141212	141212	imc CRONOSflex		
<input type="checkbox"/>	S_CRFX_400_141118	141118	imc CRONOSflex		
<input type="checkbox"/>	B_140917_CS_6004_1	140917	imc C Series		
<input type="checkbox"/>	B_140916_CS_6004_1	140916	imc C Series		
<input type="checkbox"/>	B_140915_CS_6004_1	140915	imc C Series		
<input type="checkbox"/>	B_140914_CS_6004_1	140914	imc C Series		
<input type="checkbox"/>	SICAM_Q80_140135	140135	SICAM Q80		
<input type="checkbox"/>	CronosPL_Compact_2	140122	DABHS		
<input type="checkbox"/>	miniPOLARES_140101	140101	miniPOLARES		
<input type="checkbox"/>	miniPOLARES_140071	140071	miniPOLARES		
<input type="checkbox"/>	miniPOLARES_8U_140042	140042	miniPOLARES		

3. Configuring an experiment

- Go to the page "Analog channels" (1) in order to configure the measurement channels.
- Activate the first two analog channels of the audio amplifier and give them some names such as "Mic_01" and "Mic_02" (2).
- *Measurement mode: "Voltage"; Coupling: "IEPE"* (3)
- The sensitivity of the microphones is indicated in their respective data sheets. These values are to be entered under "Range & Scaling". For instance, according to the data sheet, the sensitivity of the training microphone is 50.2 mV/Pa. (4)
- The input range to select (5) is the smallest one possible, so that the signal is not yet overdriven.

The screenshot displays the 'Analog channels' configuration page in the imc WAVE Spectrum Analyzer. The interface is divided into several sections:

- Navigation Bar:** Includes 'Homepage', 'Devices', 'Analog channels' (highlighted with a red circle 1), 'Channel balance', 'WAVE Analyzer', and 'Measurement'.
- Channel List Table:**

Name	Connector	Status	Current value	Measurement mode	Range & Scaling	Sampling & Filter
Channel type: Analog inputs (Count=12)						
(Mic_01)	[01] IN01	Active		AC with current supply - Linear	±498.00796812749 Pa	50 kHz - AAF
Mic_02	[01] IN02	Active		AC with current supply - Linear	±498.00796812749 Pa	50 kHz - AAF
Channel_005	[02] IN01	Passive		DC - Linear	±60 V	50 kHz - AAF
Channel_006	[02] IN02	Passive		DC - Linear	±60 V	50 kHz - AAF
Channel_007	[02] IN03	Passive		DC - Linear	±60 V	50 kHz - AAF
Channel_008	[02] IN04	Passive		DC - Linear	±60 V	50 kHz - AAF
Channel_009	[02] IN05	Passive		DC - Linear	±60 V	50 kHz - AAF
Channel_010	[02] IN06	Passive		DC - Linear	±60 V	50 kHz - AAF
Channel_011	[02] IN07	Passive		DC - Linear	±60 V	50 kHz - AAF
Channel_012	[02] IN08	Passive		DC - Linear	±60 V	50 kHz - AAF
Channel_003	[01] IN03	Passive		DC - Linear	±100 V	50 kHz - AAF
Channel_004	[01] IN04	Passive		DC - Linear	±100 V	50 kHz - AAF
Channel type: Analog outputs (Count=4)						
- Configuration Panel:**
 - Channel name:** #
 - Input range:** ±498.00796812749 Pa (highlighted with a red circle 5)
 - Unit:** Pa
 - Factor-offset scaling:** 2-point scaling
 - Factor:** 19.9203 Pa/V
 - Sensitivity:** 0.0502 V/Pa (highlighted with a red circle 4)
 - Offset:** 0 Pa

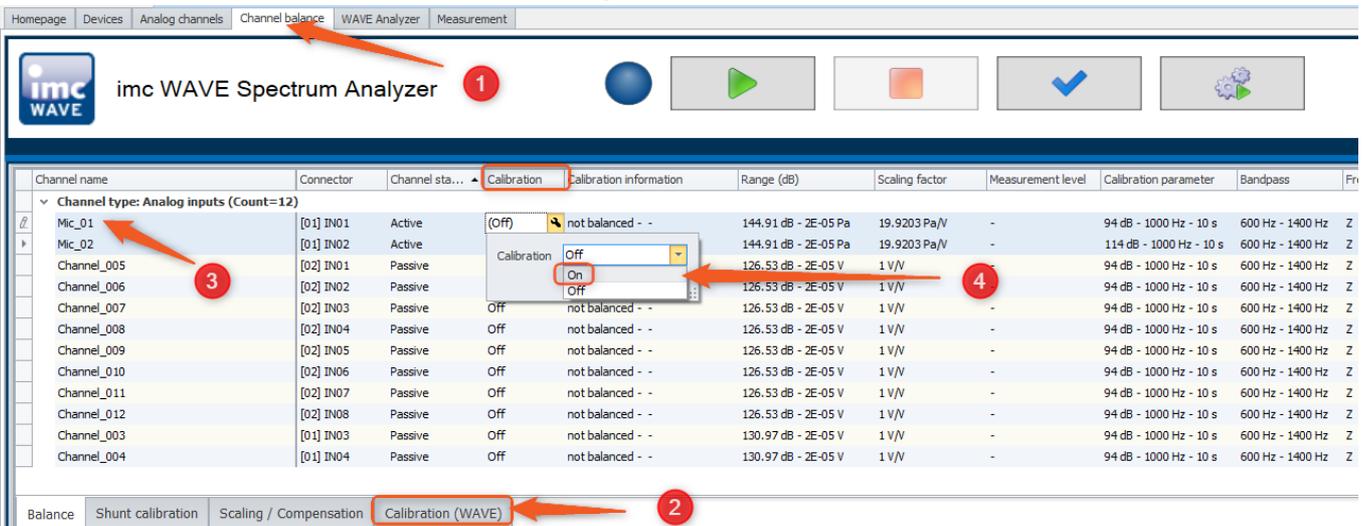
7.5.2.2 Calibrating microphones

Calibrating the microphones before the first measurement

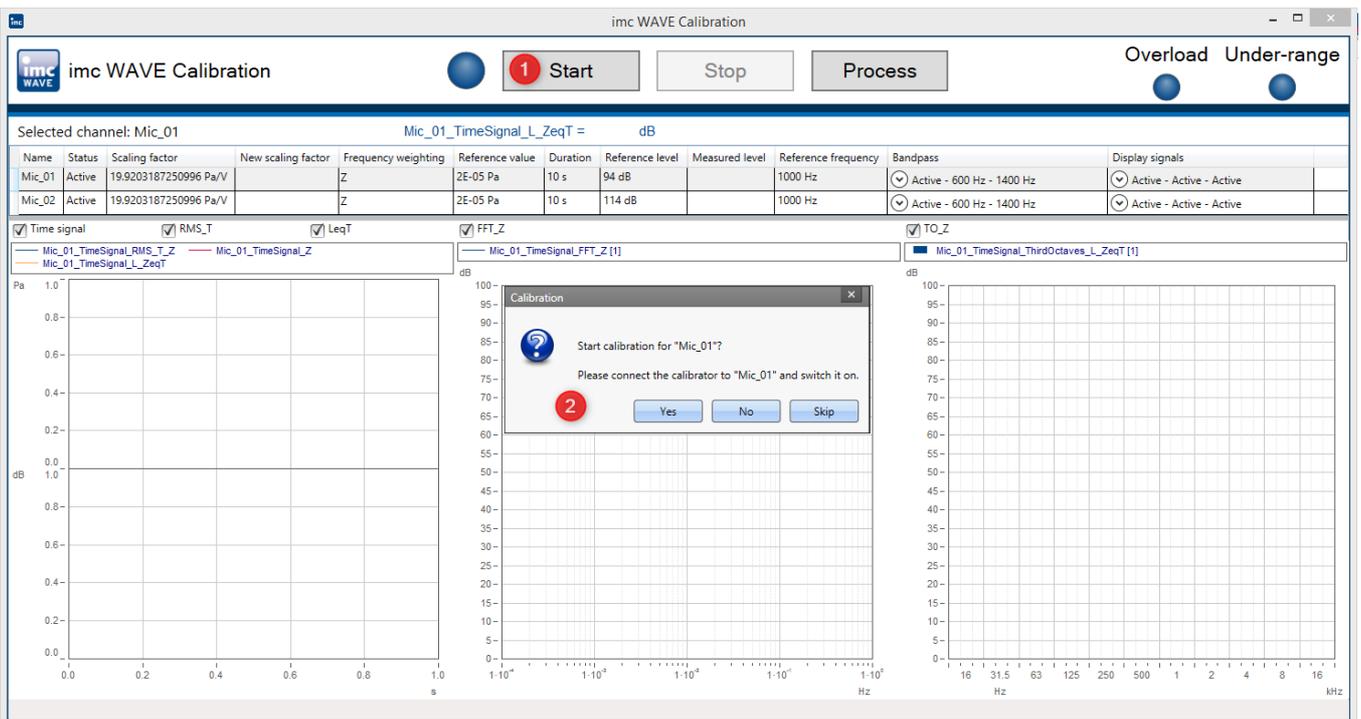
Calibration of the microphones is performed on the page "Channel balance". The signal sound level is balance to 94 dB, for example, at a frequency of 1 kHz. Independently of the selected frequency weighting, the gain for all frequency weightings A, B, C or Z is 0 dB at a frequency of 1 kHz.

The measurement channels each in succession measure a reference sound pressure level from the calibrator of either 94dB or 114 dB, for 10 seconds. The scaling factor measured is modified in the channel for the microphone.

- Go to the page "Channel balance" (1), and there, go to the tab "Calibration (WAVE)" (2)
- Select the two measurement channels "Mic_01" and "Mic_02" aus (3).
- In the column "Calibration", select the setting "On" (4).

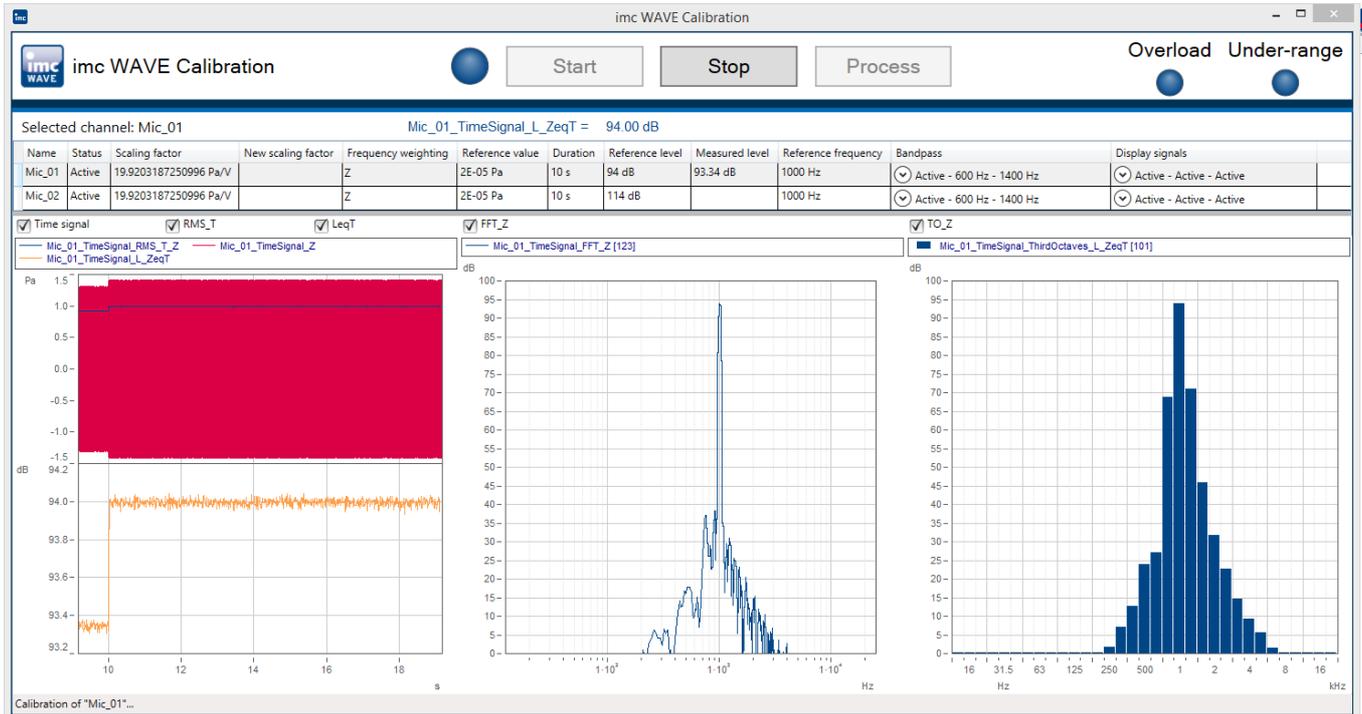


- The dialog "imc WAVE Calibration" opens:

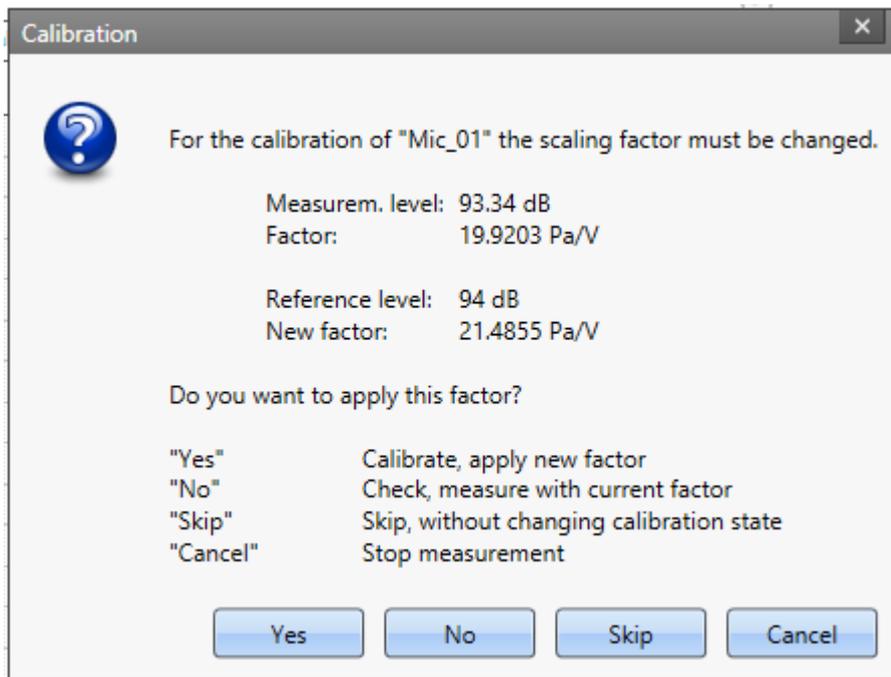


- Turn on the calibrator, set it to the sound level 94dB and conclude the dialog (2) by clicking on "Yes"

- Start the measurement (1) for the channel "Mic_01".
- After 10 seconds of measurement, the results appear:



- Apply the new factor by clicking on "Yes"



- Next, calibration of the channel "Mic_02" proceeds automatically.

The screenshot shows the 'imc WAVE Spectrum Analyzer' interface. At the top, there are navigation tabs: 'Homepage', 'Devices', 'Analog channels', 'Channel balance', 'WAVE Analyzer', and 'Measurement'. Below the tabs is a toolbar with icons for 'Start' (a red circle with '1'), 'Stop', 'Process', and 'Overload'. The main area displays a table of channel configurations.

Channel name	Connector	Channel sta...	Calibration	Calibration information	Range (dB)	Scaling factor	Measurement level	Calibration parameter	Bandpass	Frequency we
▼ Channel type: Analog inputs (Count=12)										
Mic_01	[01] IN01	Active	(Off)	balanced - 30.01.2019 - 11:3...	145.57 dB - 2E-05 Pa	21.4855 Pa/V	94 dB	94 dB - 1000 Hz - 10 s	600 Hz - 1400 Hz	Z
Mic_02	[01] IN02	Active	Off	balanced - 30.01.2019 - 11:3...	145.47 dB - 2E-05 Pa	21.2267 Pa/V	114 dB	114 dB - 1000 Hz - 10 s	600 Hz - 1400 Hz	Z
Channel_005	[02] IN01	Passive	Off	not balanced - -	126.53 dB - 2E-05 V	1 V/V	-	94 dB - 1000 Hz - 10 s	600 Hz - 1400 Hz	Z
Channel_006	[02] IN02	Passive	Off	not balanced - -	126.53 dB - 2E-05 V	1 V/V	-	94 dB - 1000 Hz - 10 s	600 Hz - 1400 Hz	Z
Channel_007	[02] IN03	Passive	Off	not balanced - -	126.53 dB - 2E-05 V	1 V/V	-	94 dB - 1000 Hz - 10 s	600 Hz - 1400 Hz	Z
Channel_008	[02] IN04	Passive	Off	not balanced - -	126.53 dB - 2E-05 V	1 V/V	-	94 dB - 1000 Hz - 10 s	600 Hz - 1400 Hz	Z
Channel_009	[02] IN05	Passive	Off	not balanced - -	126.53 dB - 2E-05 V	1 V/V	-	94 dB - 1000 Hz - 10 s	600 Hz - 1400 Hz	Z
Channel_010	[02] IN06	Passive	Off	not balanced - -	126.53 dB - 2E-05 V	1 V/V	-	94 dB - 1000 Hz - 10 s	600 Hz - 1400 Hz	Z

- Start the measurement (1) for the channel "Mic_02".
- Ensure that the microphone is connected to the channel "Mic_02" and that 114 dB are applied at it by the calibrator. Start the measurement.
- After 10 seconds of measurement, the results appear:

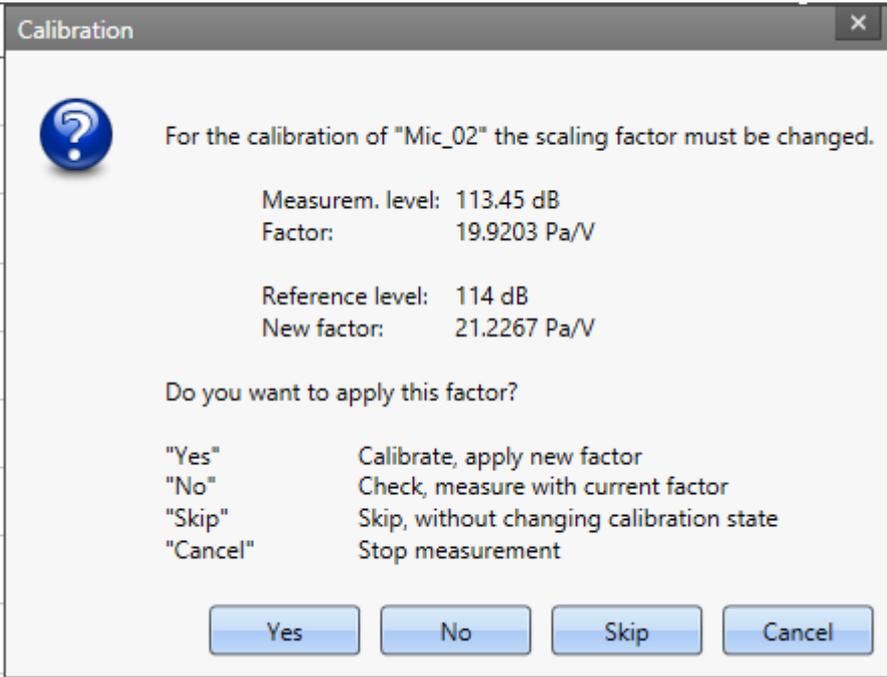
The screenshot shows the 'imc WAVE Calibration' interface. At the top, there are buttons for 'Start', 'Stop', and 'Process'. Below the buttons, it says 'Selected channel: Mic_02' and 'Mic_02_TimeSignal_L_ZeqT = 113.98 dB'. A table shows the measurement results for Mic_01 and Mic_02.

Name	Status	Scaling factor	New scaling factor	Frequency weighting	Reference value	Duration	Reference level	Measured level	Reference frequency	Bandpass	Display signals
Mic_01	Active	19.9203187250996 Pa/V	21.4855 Pa/V	Z	2E-05 Pa	10 s	94 dB	94 dB	1000 Hz	Active - 600 Hz - 1400 Hz	Active - Active - Active
Mic_02	Active	19.9203187250996 Pa/V		Z	2E-05 Pa	10 s	114 dB	113.45 dB	1000 Hz	Active - 600 Hz - 1400 Hz	Active - Active - Active

Below the table, there are three plots:

- Time signal:** Shows the time-domain signal for Mic_02. The y-axis is in Pa (15 to -15) and the x-axis is in seconds (6 to 14). The signal is a red square wave.
- FFT_Z:** Shows the frequency spectrum of the signal. The y-axis is in dB (0 to 120) and the x-axis is in Hz (10⁻¹ to 10¹). The spectrum shows a peak at 1000 Hz.
- TO_Z:** Shows the third-octaves frequency spectrum. The y-axis is in dB (0 to 120) and the x-axis is in kHz (0.16 to 16). The spectrum shows a peak at 1 kHz.

- The scaling factor for the channel "Mic_02" is provided. Apply the new factor by selecting "Yes".



- The scaling factors now appear in the Channels list:

Homepage Devices Analog channels Channel balance WAVE Analyzer Measurement

imc WAVE Spectrum Analyzer

Channel name	Connector	Channel sta...	Calibration	Calibration information	Range (dB)	Scaling factor	Measurement level	Calibration parameter	Bandpass
Channel type: Analog inputs (Count=12)									
Mic_01	[01] IN01	Active	(Off)	balanced - 30.01.2019 - 11:3...	145.57 dB - 2E-05 Pa	21.4855 Pa/V	94 dB	94 dB - 1000 Hz - 10 s	600 Hz - 1400 Hz
Mic_02	[01] IN02	Active	Off	balanced - 30.01.2019 - 11:3...	145.47 dB - 2E-05 Pa	21.2267 Pa/V	114 dB	114 dB - 1000 Hz - 10 s	600 Hz - 1400 Hz
Channel_005	[02] IN01	Passive	Off	not balanced - -	126.53 dB - 2E-05 V	1 V/V	-	94 dB - 1000 Hz - 10 s	600 Hz - 1400 Hz
Channel_006	[02] IN02	Passive	Off	not balanced - -	126.53 dB - 2E-05 V	1 V/V	-	94 dB - 1000 Hz - 10 s	600 Hz - 1400 Hz
Channel_007	[02] IN03	Passive	Off	not balanced - -	126.53 dB - 2E-05 V	1 V/V	-	94 dB - 1000 Hz - 10 s	600 Hz - 1400 Hz
Channel_008	[02] IN04	Passive	Off	not balanced - -	126.53 dB - 2E-05 V	1 V/V	-	94 dB - 1000 Hz - 10 s	600 Hz - 1400 Hz
Channel_009	[02] IN05	Passive	Off	not balanced - -	126.53 dB - 2E-05 V	1 V/V	-	94 dB - 1000 Hz - 10 s	600 Hz - 1400 Hz
Channel_010	[02] IN06	Passive	Off	not balanced - -	126.53 dB - 2E-05 V	1 V/V	-	94 dB - 1000 Hz - 10 s	600 Hz - 1400 Hz

7.5.2.3 Configuring the WAVE Analyzer

On the page "Analog channels", under the tab "Inline Analysis", the desired weightings and the channels derived from the measurement level signal are selected. The properties for the FFT-calculations are set in the same manner.

- In this example, the channel "Mic_01" is filtered with the frequency weighting "A" and the time weighting "Fast". The A-frequency weighting is usually applied.
- The channel "Mic_02", however, is filtered with the frequency weighting "B" and the time weighting "Slow".
- With a setting of at least 8192 lines and an overlap of 50%, a more exact FFT-calculation with high-resolution FFT-representation is achieved.
- For optimal representation, the FFT-window can be selected for each measurement channel, e.g. *Flat-top* for the channel "Mic_01".

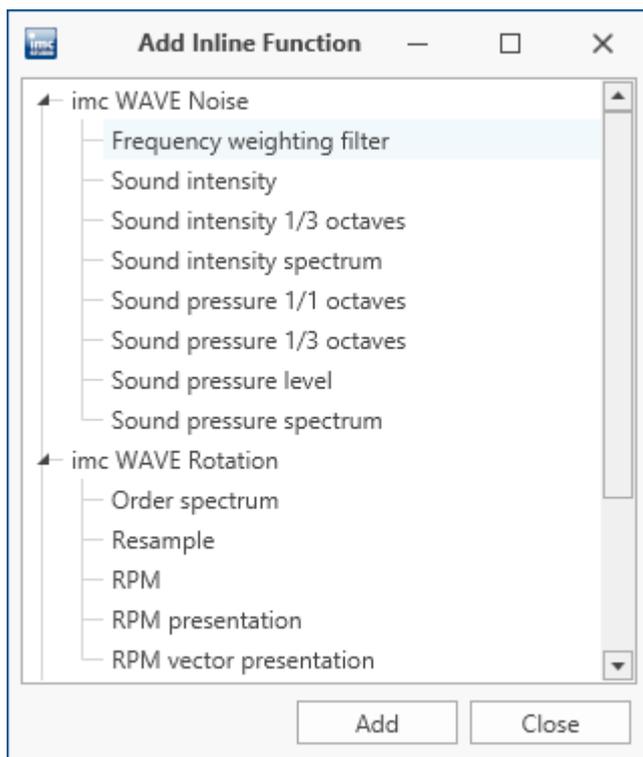
Adding calculations

By using the following dialog, it is possible to compute a derivative or calculated channel from a input channel.

Page: Analog channels

Dialog: [Inline Analysis](#)

- Select the channel from the table of channels
- Click on the plus-sign (+). A dialog appears in which all possible calculations for the channel are presented



Adding new Inline Functions

- Select a calculation and click on "Add". The calculation is added to the list.

7.5.2.4 Measuring

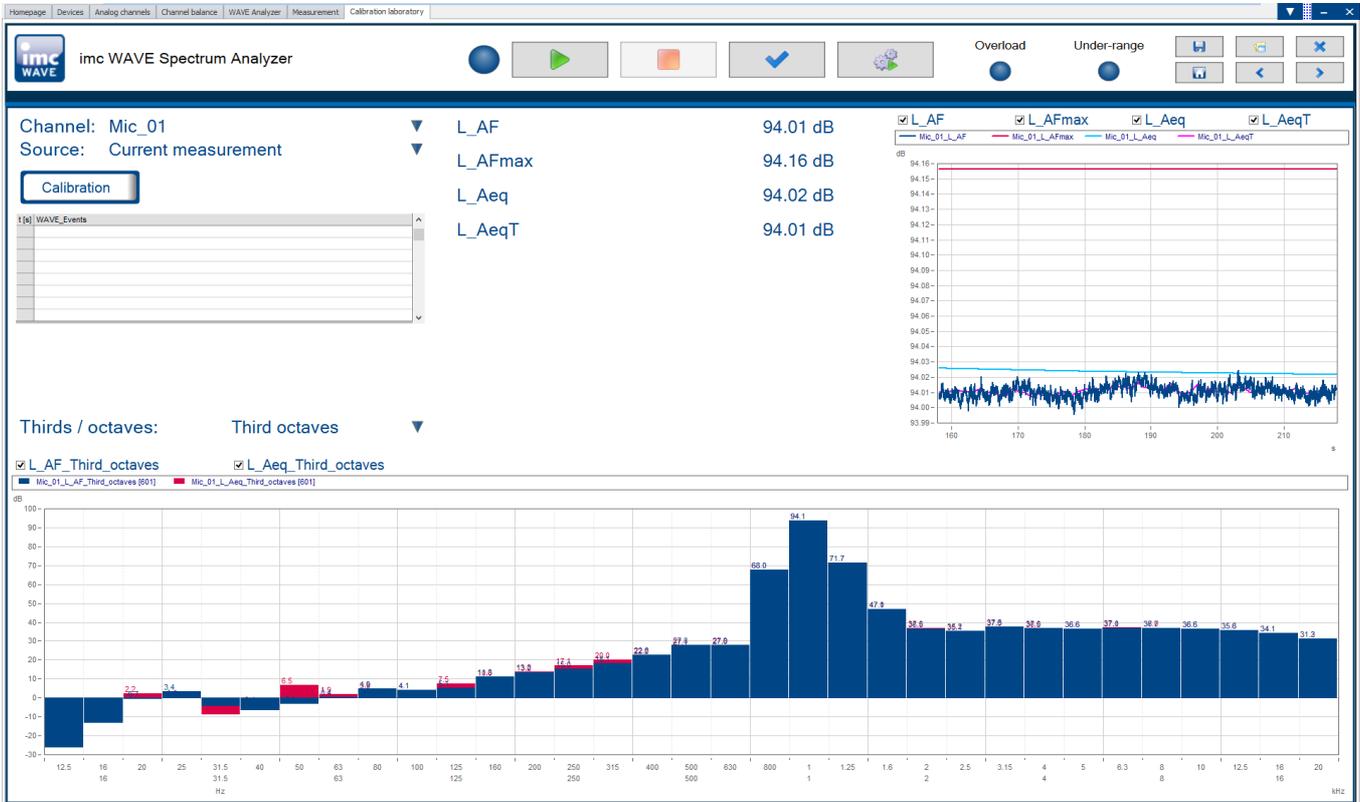
On the page "Measurement" ①, run the actual measurement by clicking on the Start-button. It is possible to switch between displays of the calculated derived channels ②. Further, it is possible to switch to display as octaves and 1/3-octaves ③.



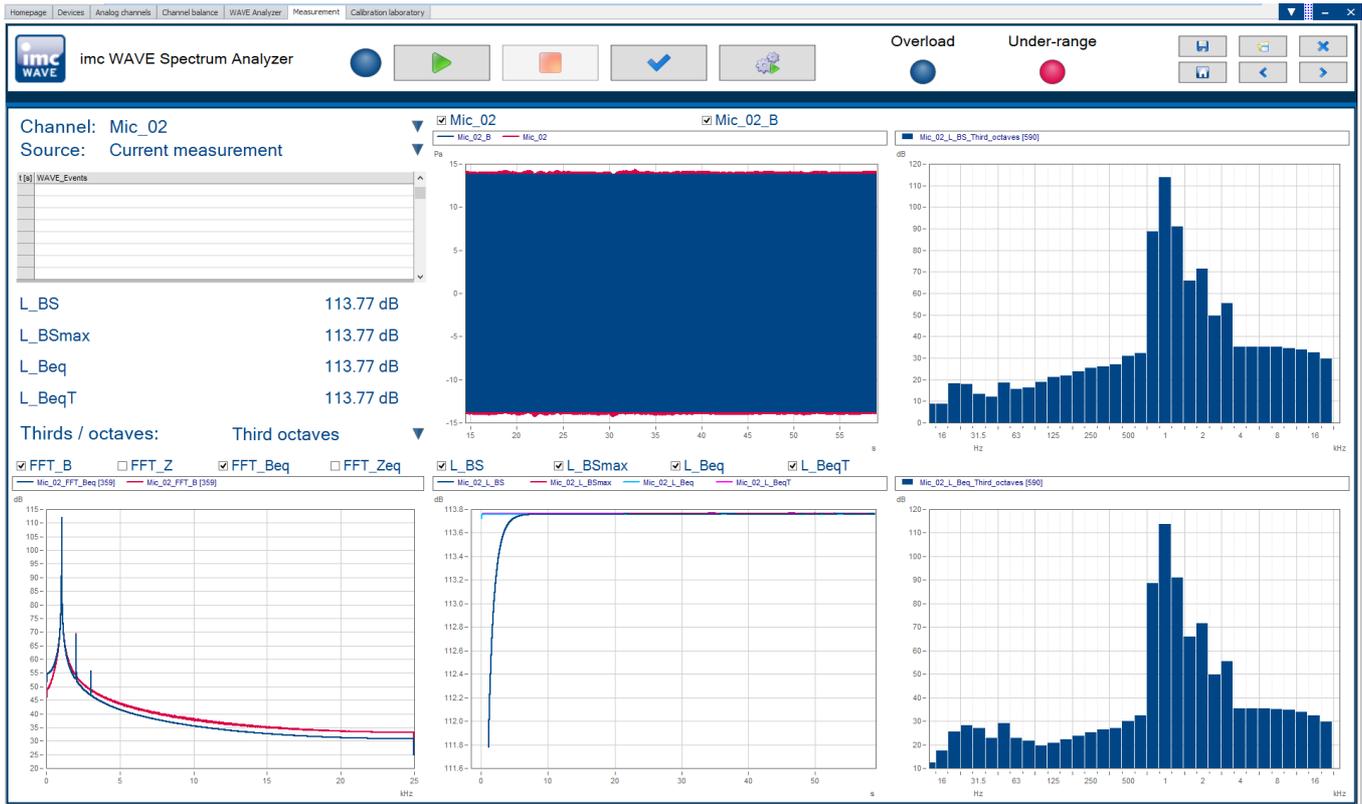
"Measurement" page for the channel "Mic_01" with 1/3-octave representation

The page "Calibration laboratory" also shows the weightings and calculated channels, e.g. the FFT_A, FFT_Aeq as well as L_AF_Third_octaves and L_Aeq_Third_octaves.

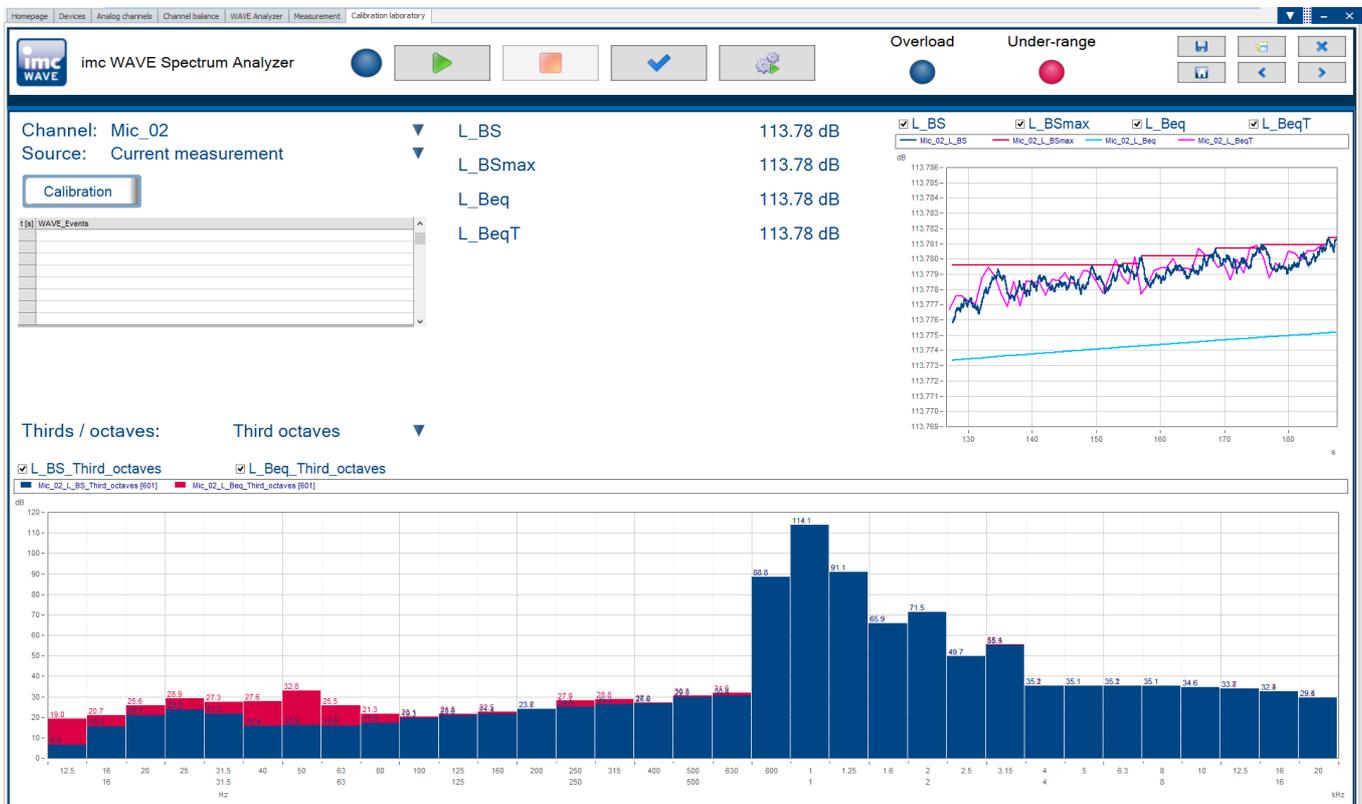
On this page, the user can even conveniently perform calibration of the channels (microphones) directly before the measurement:



"Calibration laboratory for the channel "Mic_01" with octaves-representation



"Measurement" page for the channel "Mic_02" with octaves representation



"Calibration laboratory" page for the channel "Mic_02" with 1/3-octaves representation

7.5.2.5 Command batches

The extra page "Command batches" provides additional user-defined analyses. The analyses consist of various commands and are run automatically by pre-defined events.

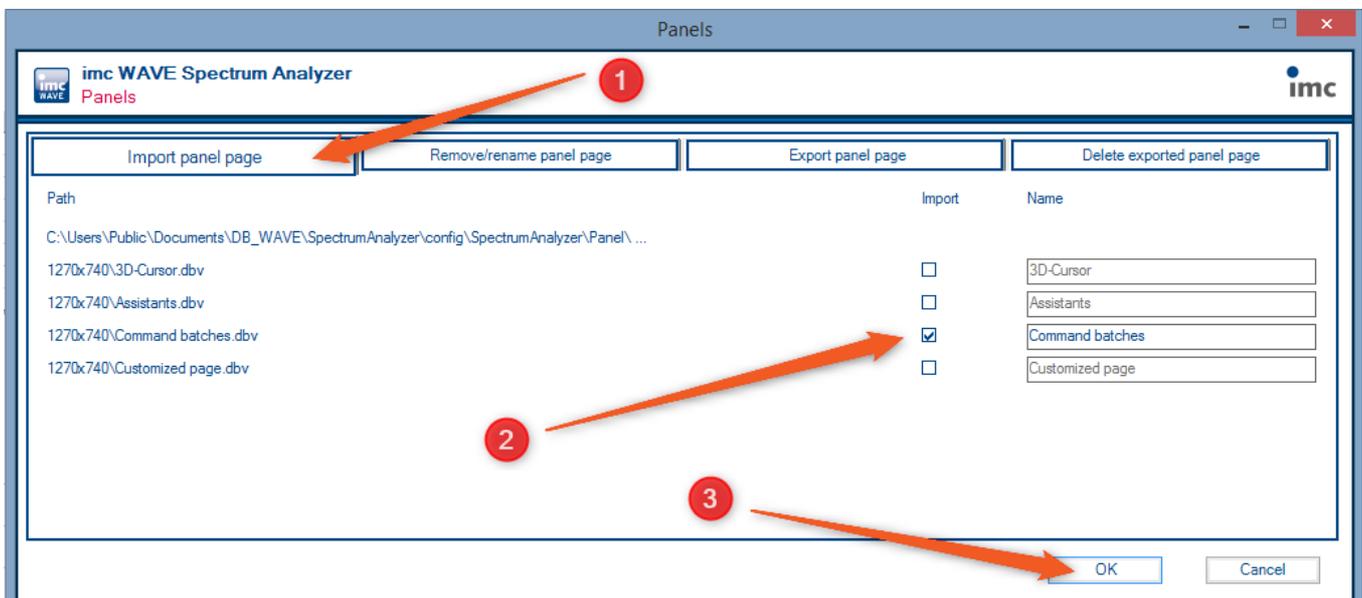
Pre-defined events include, for instance:

- After **loading** an experiment. Event: "Experiment_Load"
- At the end of the measurement. Event: "Devices_Stopped"
- After saving results. Event: "WAVE_Results_Saved"
- Upon occurrence of an error: "OnError".

In the example below, a FAMOS-sequence having the name "SoundPressureLevelCheck" is created by means of the command "Run imc FAMOS Sequence". In this FAMOS-sequence, the calculations of some channels in imc WAVE and imc FAMOS are compared. The call to this command is linked to the event "Devices_Stopped". In order to be able to run it, imc FAMOS must be installed and licensed on the local system.

1. Adding the page "Command batches"

- Import the page "Command batches" by clicking on the button "Open Panel Dialog" 



On the page "Import panel page" (1), activate the page "Command batches" (2) and close by clicking on OK (3)

2. Adding the command "Run imc FAMOS sequence"

- Select the added page "Command batches". (1)
- Using the context menu item "New" ->"Element" -> "Scripts/imc FAMOS", add the command "Run imc FAMOS sequence". (2)
- In the region "Run analysis automatically upon:", activate the event "Devices_Stopped" (3).

The screenshot shows the 'Command batches' tab in the imc WAVE Spectrum Analyzer. The interface includes a top navigation bar with tabs for 'Homepage', 'Devices', 'Analog channels', 'Channel balance', 'WAVE Analyzer', 'Measurement', 'Calibration laboratory', and 'Command batches'. Below the navigation bar is a toolbar with a play button, a stop button (marked with a red '1'), and a checkmark button. The main area displays a tree view for 'Command_Batch_1' with a sub-item 'Execute commands' containing 'Run imc FAMOS sequence: imc FAMOS sequence'. A context menu is open over this item, showing options like 'New', 'Cut', 'Copy', 'Paste', 'Delete', 'Start', and 'Expand / collapse'. A sub-menu is also open, showing 'Scripts/imc FAMOS' with options 'Run imc FAMOS sequence' and 'Execute imc FAMOS project' (marked with a red '2'). Below the tree view is a section titled 'Run analysis automatically upon:' with a table of events and checkboxes. The 'Devices_Stopped' event is checked (marked with a red '3').

Batch: Command_Batch_1 Execute

Name	Hint
Command_Batch_1	
Execute commands	Occurs when...
Run imc FAMOS sequence: imc FAMOS sequence	...

Run analysis automatically upon:

Event name	Active
Experiment_Loaded	<input type="checkbox"/>
Devices_AfterCheckConfiguration	<input type="checkbox"/>
Devices_Stopped	<input checked="" type="checkbox"/>
WAVE_Results_Saved	<input type="checkbox"/>
OnError	<input type="checkbox"/>

3. Writing the sequence

- Open the "imc FAMOS Sequence" dialog (1) and enter the analysis sequence (2). A detailed description of the use of imc FAMOS sequences in imc STUDIO/WAVE is provided [here](#) (1149):

The screenshot shows the imc WAVE Spectrum Analyzer interface. At the top, there is a navigation bar with tabs: Homepage, Devices, Analog channels, Channel balance, WAVE Analyzer, Measurement, Calibration laboratory, and Command batches. Below this is the application title bar: imc WAVE Spectrum Analyzer. A toolbar contains several icons: a blue sphere, a green play button, a red stop button, a blue checkmark, and a gear icon. Below the toolbar, there is a 'Batch:' dropdown menu set to 'Command_Batch_1' and an 'Execute' button. A list of batches is shown below, with 'Command_Batch_1' selected. A red circle '1' is placed over the 'Run imc FAMOS sequence' button in the batch list. An orange arrow points from this button to the 'imc FAMOS Sequence' dialog box. The dialog box has three tabs: 'Import/Export', 'Variable transfer', and 'Configuration'. The 'Import/Export' tab is active, showing a table for transferring variables from imc STUDIO to imc FAMOS. The table has two columns: 'imc STUDIO variable' and 'imc FAMOS variable'. The variables listed are Mic_01, Mic_01_L_AF, Mic_01_L_AeqT, Mic_01_L_Aeq, and Mic_01_L_AFmax. A red circle '2' is placed over the 'imc FAMOS sequence' text area, which contains the following script:

```

; SoundPressureLevelCheck

; WAVE
LAeq = 10*log( int( sqr( abcrating( Mic_01, 1, 0, 1, 0 ) ) / sqrt( 20e-6 ) ) / ( leng?( Mic_01 ) ) )
LAeq_Val = LAeq[ Leng?( LAeq ) ]
SetUnit( LAeq_Val, "dB", 1 )

LAFmax1 = Max( 10*log( sqr( ExpoRMS( abcrating( Mic_01, 1, -1, 1, 0 ), 0.125, 0.05 ) ) / sqrt( 2 ) ) )
SetUnit( LAFmax1, "dB", 1 )
r= Leng?( Mic_01_L_Aeq )
; WAVE last value
L_Aeq_WAVE_Val = Mic_01_L_Aeq[r]

L_AFmax_WAVE_Val = Max( Mic_01_L_AF )

; FAMOS
Signal = "Mic_01"

```

- After the measurement, the newly created channels are then available in the Data Browser in the branch "FAMOS".

7.6 Structural Analyzer

The Structural Analyzer serves the purpose of systematic analysis of the vibration behavior of mechanical structures. This is typically done by applying a strike with a special impulse hammer. This injects a well defined excitation signal into the structure, while the structure's response to this stimulus is measured using accelerometer sensors. The correlations among all synchronous signals (outputs), including that of the stimulus (input), yields transfer functions which fully characterize the structure's vibration behavior (resonances, eigenfrequencies, modes).

These calculated results in connection with the information on the associated measurement points (DOF, Degree Of Freedom), which the Structure Analyzer also administers, can be exported to subsequent analysis programs via a standardized data exchange format. Such specialized analysis programs, for instance ME'scope by Vibrant Technology, can be purchased in addition to the WAVE-Analyzer and offer advanced evaluation tools and display options for modal analysis.

Analyzer functionalities

- calculation of transfer functions and coherence
- management of DOFs
- linkage with modal analysis software

Functions	Description
Basic analysis functions	<ul style="list-style-type: none"> • auto power spectral density • transfer function • coherence function • H1, measurement with noisy output signal • time domain signal, time plot • window functions: force window or exponential window
Measurement functions	<ul style="list-style-type: none"> • acoustic announcement (speech output) of the current DOFs for assistance during measurement • automatic impulse bounce detection and evaluation of strike quality • automatic detection of overrange or strikes below a threshold level • acoustic announcement of error warnings • replay of last measurement (impulse hammer strike) for evaluation (visual plot and audio output via headphones) • display of measurement count • interactive setting of the validity flag and selection of valid strikes to be evaluated
Administration functions	<ul style="list-style-type: none"> • administration of DOF information on all measurement channels and results • roving for input and output-DOFs • data export to and ME`Scope (standard format: *.uff) • traceability of the settings for each measurement performed

Typical applications

- structural studies and modal analysis
- development and product optimization
- product qualification tests

7.6.1 Options

Homepage
Devices
Analog channels
Triggers
WAVE Analyzer
Options
DOFs
Measurement
Export



imc WAVE Structural Analyzer



Basic options

Measurement type Impact hammer ▾

Roving Input ▾

Mode Manual ▾

Count of impacts

Additional options

Automatic trigger settings

Show trigger threshold

Display impact quality

Automatic zoom None ▾

Sound

DOF announce

Sound ready

Warning invalid

Voice command

Double smash

Reject

Detect

Double smash height

Smash time

Overload

Reject

Detect

Overload input

Overload output

Storage

All

Time series

FRF average

Coherence

PSD input

Windowed t. series

FRF current

PSD output

Windows

Directory names Default ▾

Storage location

© 2024 imc Test & Measurement GmbH

imc WAVE 2023 - Manual, Doc. Rev.: 6.5 - 2024-02-29

Page 133

Parameter - Basic options	Description
Measurement type	Selection of the measurement type: <ul style="list-style-type: none"> • Impact hammer measurement
Roving	Roving input or output. For the roving component, multiple measurement points (DOFs) can be defined. Typically, a structural analysis is performed with a hammer and an accelerometer. Various measurement points (DOFs) are excited in succession. The fastest way to perform the measurements is with a fixed reference point (accelerometer). The excitations are then successively applied by the hammer to the various measurement points. The setting " <i>Input</i> " means that all inputs are roving. Alternatively, the outputs can be set as roving.
Mode	Mode of advancing to the next measurement point: <ul style="list-style-type: none"> • Manually: Advancing to the next measurement point must be confirmed manually. • Automatic: Automatic advancing to the next measurement point once the required number of valid measurements has been reached.
Count of impacts	Count of valid impacts. In the measurement mode " <i>Automatic</i> ", reaching this number causes the system to advance to the next measurement point. In the <i>manual</i> measurement mode, this parameter has no effect and is only for orientation purposes.
Parameter - Additional options	Description
Automatic trigger settings	The trigger settings (sources and events, properties, targets etc.) are set automatically. As the source, the first input is used. In that case only the threshold value and the pretrigger can be set manually. Only such channels which are highlighted as the input or output and thus are relevant to the calculation are automatically added to the WAVE-trigger. If channels which are neither selected as the input or output are to be used for triggering, the "automatic trigger setting" must be deactivated .
Show trigger threshold	The trigger threshold is represented on the Panel-page " <i>Measurement</i> " in the curve window by a line. This is to make it clearly recognizable when an impact exceeds the threshold.
Display impact quality	Indicates the validity of the individual impacts on the Panel-page " <i>Measurement</i> ". The display takes the form of either a green (valid) or red (invalid) shaded impact number under " <i>Show impact quality</i> " at the right side of the Measurement page.
Automatic zoom	Settings options for the automatic zoom of the X-axis: <ul style="list-style-type: none"> • None: no zoom. The curve window displays the entire measurement duration. • Double-impact: When a double impact is detected, the range from - Pretrigger + X-time to the next maximum double-impact + Pretrigger is displayed. • Always: The range from - Pretrigger + X-time of impact + Pretrigger is displayed. If a double-impact occurs while the system uses this setting, the display style for "<i>Double-impact</i>" is applied. <p>Note: The Y-axis is always set automatically.</p>

Parameter - Sound	Description
DOF announce	Activates the voice announcement for the next measurement point/DOF.
Sound ready	Activates an acoustic signal (1x beep) when the measurement is ready for the next impact.
Warning invalid	Activates an acoustic signal (2x beep) if the impact is invalid.
Voice command	In preparation: <i>Start, Stop</i> and <i>Next measurement point</i> by means of voice command.
Parameter - Double smash	Description
Reject	<p>If this option is activated, the measurement is ignored when a double impact is recognized. For this to happen, the subsequent option "<i>Detect</i>" must be activated, which defines a double impact as a percentual proportion of the main impact.</p> <p>When "<i>Reject</i>" is deactivated, double-impacts are treated and processed as valid impacts. However, it is recommended to activate the automatic recognition of double-impacts.</p>
Detect	Activation of the automatic recognition of double-impacts (bouncing). Rejection can only occur when recognition is defined and activated.
Double smash height	A double-impact is only detected if this percentual proportion of the stated magnitude of the initial impact is reached after the impact time.
Smash time	Time span from the triggering time to the impact time, in which no test of double-impact is performed.
Parameter - Overload	Description
Reject	<p>When this option is activated, the measurement is ignored when an overload is recognized. For this to happen, the subsequent option "<i>Detect</i>" must be activated, which defines an overload in reference to the measurement range end value of inputs/outputs.</p> <p>When "<i>Reject</i>" is deactivated, overloads are not rejected. However, it is recommended to activate the automatic recognition.</p>
Detect	Activation of the automatic recognition of an overload. Rejection can only occur if it is both defined and activated.
Overload inputs	<p>Percentual statement of the overload threshold in proportion to the measurement range end values of the input and output.</p> <p>Example:</p> <p>Measurement range: 200m/s²; Overload: 90% -> absolute overload boundary = 180m/s²</p>
Overload outputs	Procentual statement of the overload threshold in proportion to the measurement range end values of the outputs.

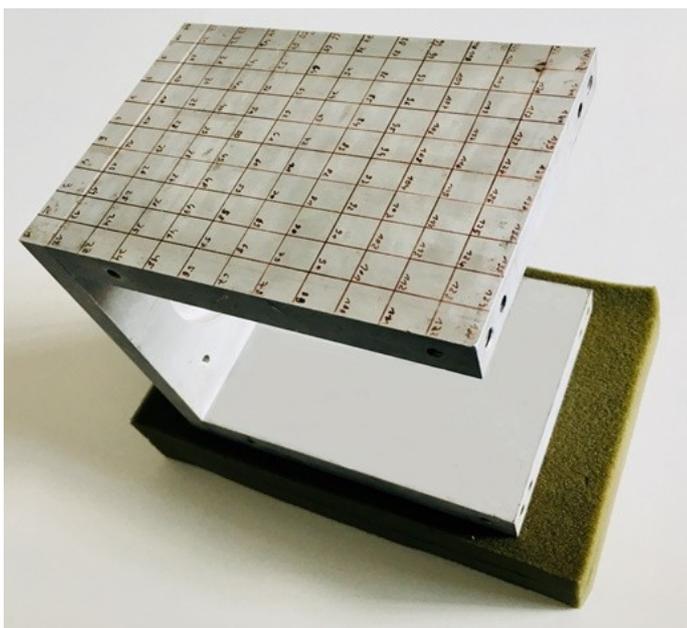
Parameter - Storage	Description
All	Switching on/off of the saving of all measured and calculated variables.
Time series	Saving of the measured time signals.
Windowed t. series	Saving of the windowed time signals if the subsequent option "Window" is used. For this purpose, the windowing ¹⁴⁰ must be activated on the WAVE Analyzer page.
Windows	Saves the window ¹⁴⁰ set for each selected channel.
FRF average	Saves the averaged frequency response functions.
FRF current	Saves the current frequency response functions.
Coherence	Saves the coherence values.
PSD in-/output	Saves power-spectral-density of input and output.
Path naming	Definition of the repository of measured data. At this time, only the default repository can be selected; other options are in preparation. <ul style="list-style-type: none"> • Default: The data storage settings appearing on the page "Devices" are used.
Storage directory	Data storage folder for measurements. User-defined data storage location is in preparation.

7.6.2 Tutorial - Structural Analyzer

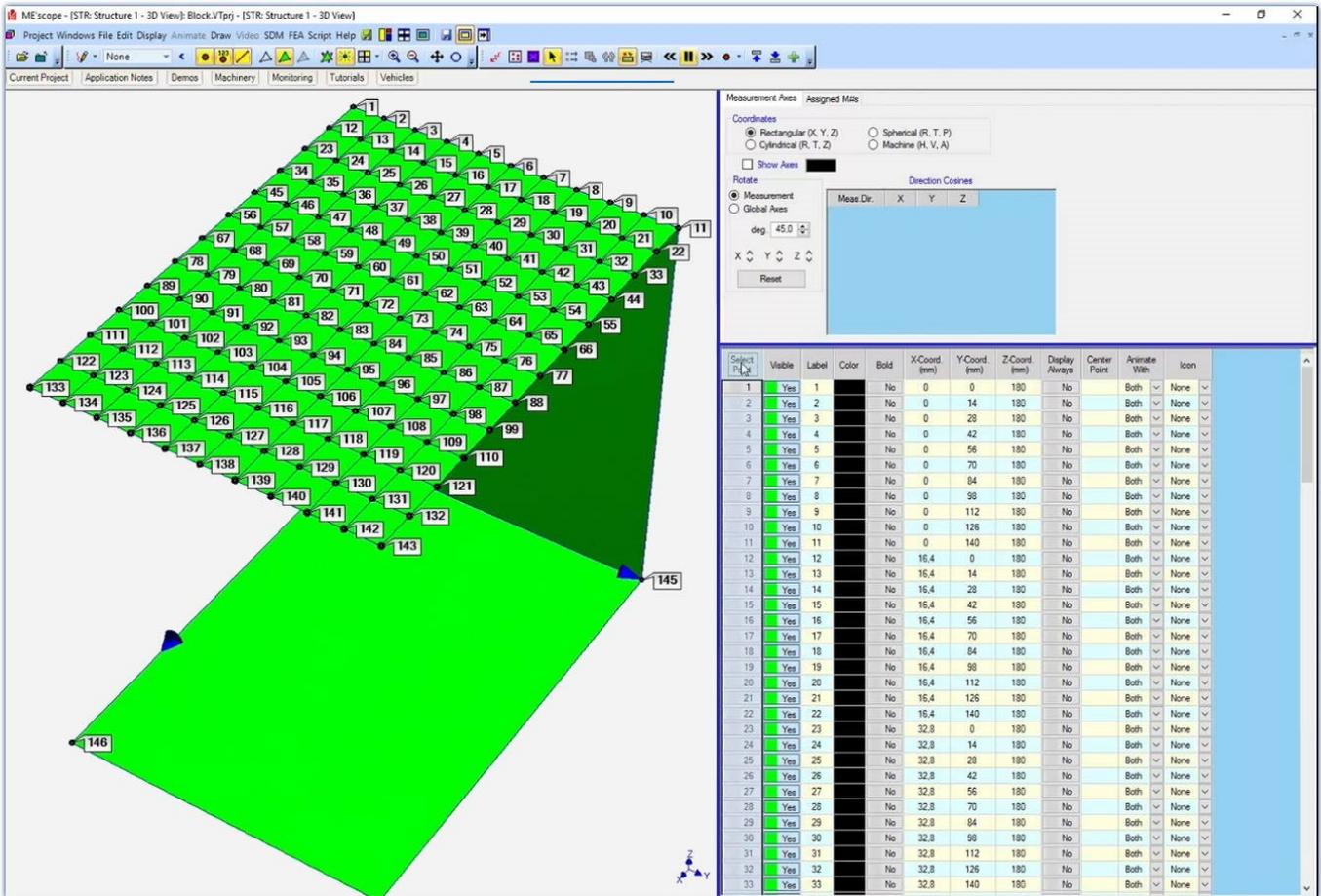
The following tutorial illustrates the procedure for performing a measurement with the Structural Analyzer.

The following devices and accessories are needed:

- An imc measurement device of either the series imc CRONOS or imc C-SERIES, with imc Online FAMOS and an
 - amplifier, e.g. UNI2-8 or ICU2-8, for measuring an impact hammer and three additional analog inputs for measurements of an imc triaxial accelerometer, e.g. SEN/ACC-AS3A002
 - Two imc charge amplifier connectors for connecting the accelerometer to the analog channels in the UNI2-8 amplifier
- a structure for which a modal analysis is to be provided



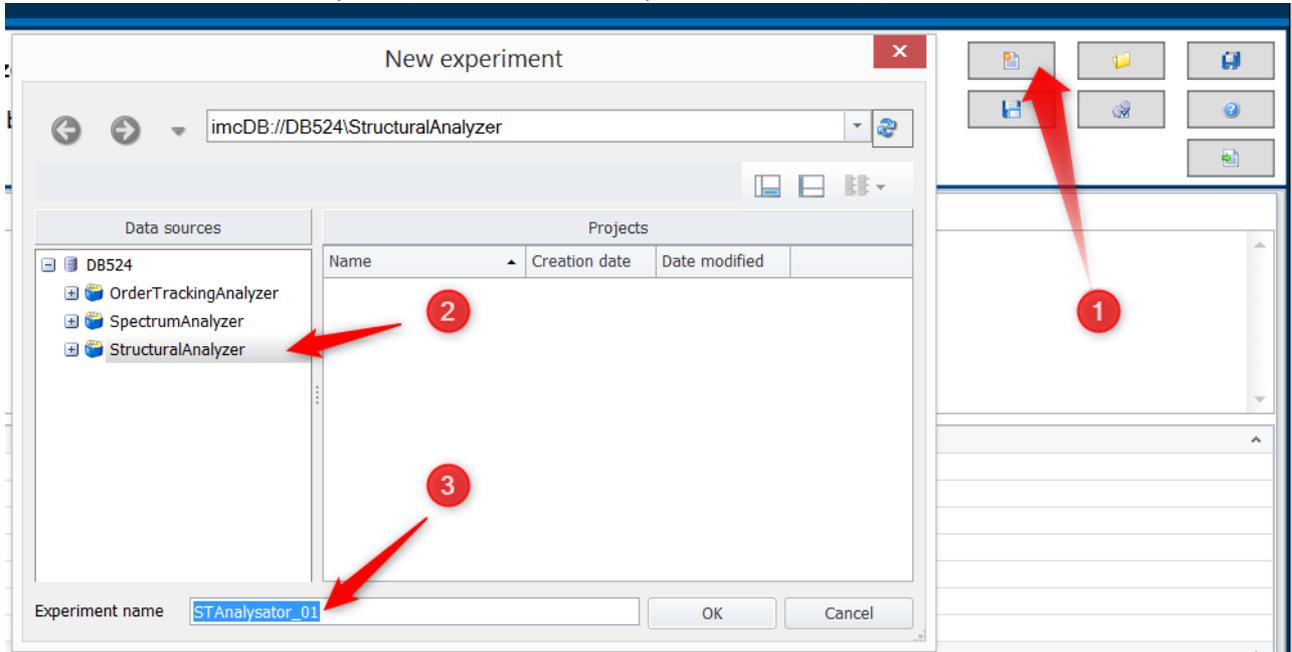
- a structural model of the measurement object, e.g. created using ME'scope and exported as a "Structure File".



7.6.2.1 Setting up an experiment

1. Create a new experiment named "STAnalysator_01":

- Start by opening the "New Experiment" dialog (1)
- Next, select "StructuralAnalyzer" (2) and enter the experiment's name (3).



- Confirm by clicking on *OK*. The experiment is set up for **structural analysis**.

2. Selecting the device

- Go to the page "Devices"
- When you click on "Search for devices" under the tab "Device options", the system finds and lists all available imc measurement devices in the network.
- Select the desired measurement device from the list.

3. Configuring the experiment

- Go to the page "Analog channels" and configure the first four channels.
The analog Channel_001 is renamed to "Force" and represents the impact hammer.
- Activate the three subsequent channels and give them names such as z.B. Acceleration_X, Acceleration_Y, Acceleration_Z. These three channels capture the motions of the triaxial sensor in the X-,Y- and Z-directions.
- Set the *Measurement mode* to "Voltage" and the *Coupling* to "DC-Linear".
- Under "Range & Scaling", the *Sensitivity* and the *Unit* for the impact hammer and the accelerometers are entered in accordance with their data sheets. E.g. the sensitivity of the imc triaxial accelerometer: **X= 10.670 mV/(s/m²); Y= 10.429 mV/(s/m²); Z= 10.510 mV/(s/m²)**, and the sensitivity of the impact hammer= **2.5mV/N**.
- Set an appropriate measurement range and modify it if necessary.

The screenshot shows the 'Analog channels' configuration page in the imc WAVE Structural Analyzer. A red arrow points to the 'Analog channels' tab. The main table lists 8 channels, with the first four highlighted in a red box:

Name	Connector	Status	C...	Measurement mode	Range & Scaling	Sampling & Filtering	Save (device)	Save (PC)
▼ Channel type: Analog inputs (Count=8)								
Force	[01] IN01	Active		DC - Linear	±2000 N	10 kHz - AAF - 819.2 ms	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Acceleration_x	[01] IN02	Active		DC - Linear	±468.603561387067 ...	10 kHz - AAF - 819.2 ms	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Acceleration_y	[01] IN03	Active		DC - Linear	±479.432352095119 ...	10 kHz - AAF - 819.2 ms	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Acceleration_z	[01] IN04	Active		DC - Linear	±475.737392959087 ...	10 kHz - AAF - 819.2 ms	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Kanal_005	[01] IN05	Passive		Full bridge - Linear	±1000 "mV/V"	10 kHz - AAF - 819.2 ms	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Kanal_006	[01] IN06	Passive		Full bridge - Linear	±1000 "mV/V"	10 kHz - AAF - 819.2 ms	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Kanal_007	[01] IN07	Passive		Full bridge - Linear	±1000 "mV/V"	10 kHz - AAF - 819.2 ms	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Below the table, the 'Range & Scaling' configuration for the selected channel 'Acceleration_x' is shown:

Channel name: Acceleration_x

Input range: ±468.603561387067 m/s²

Unit: m/s²

Factor-offset scaling: 2-point scaling

Factor: 93.7207 m/s²/V

Sensitivity: 0.01067 V/m/s²

Offset: 0 m/s²

On the page "WAVE Analyzer", additional settings for individual channels are made.

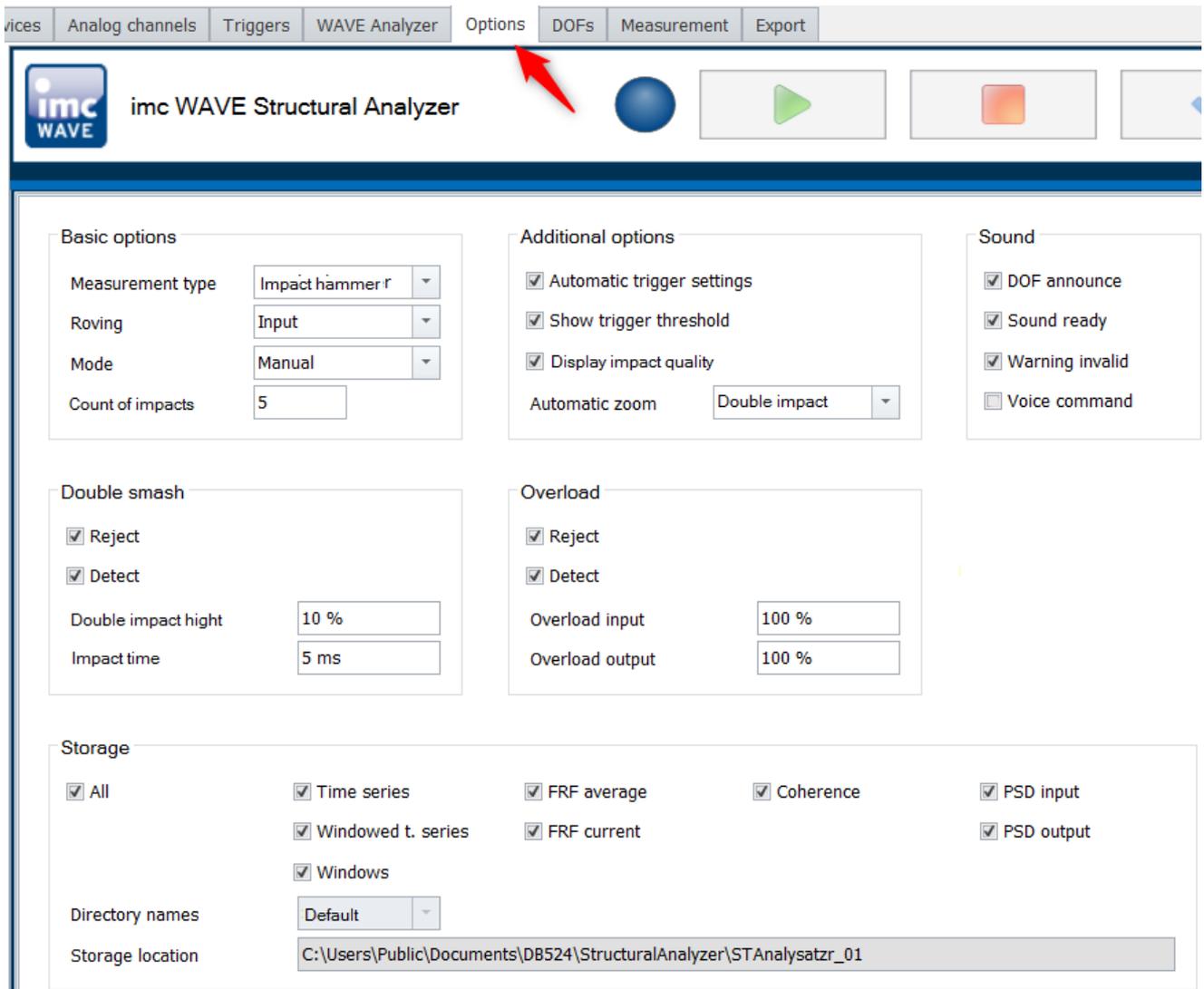
- The impact hammer channel "Force" is the only input channel in this example.
- Force is provided with the [Roving](#) ^[134]-option ^[134], since the impact hammer can strike various locations.
- The three "Acceleration" channels are output channels and, as triaxial sensors of fixed position, do not have any Roving-option.
- The windowing is set separately for each channel, in order to be able to filter the input/output channels in a sensible way.

The screenshot displays the 'WAVE Analyzer' interface. At the top, a menu bar includes 'Analog channels', 'Triggers', 'WAVE Analyzer', 'Options', 'DOFs', 'Measurement', and 'Export'. Below this is the 'imc WAVE Structural Analyzer' header with a play button and other controls. A table lists the following channels:

Name	Connector	Status	Cu...	Measurement ...	Range & Scaling	Sampling & Filter...	Input	Output	Roving	Windowing	Save (PC)
▼ Channel type: Analog inputs (Count=8)											
Force	[01] IN01	Active		DC - Linear	±2000 N	10 kHz - AAF - 81...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Exponential	<input checked="" type="checkbox"/>
Acceleration_x	[01] IN02	Active		DC - Linear	±468.60356138706...	10 kHz - AAF - 81...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Rectangle	<input checked="" type="checkbox"/>
Acceleration_y	[01] IN03	Active		DC - Linear	±479.43235209511...	10 kHz - AAF - 81...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Rectangle	<input checked="" type="checkbox"/>
Acceleration_z	[01] IN04	Active		DC - Linear	±475.73739295908...	10 kHz - AAF - 81...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Rectangle	<input checked="" type="checkbox"/>
Kanal_005	[01] IN05	Passive		Full bridge - Linear	±1000 "mV/V"	10 kHz - AAF - 81...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	None	<input checked="" type="checkbox"/>
Kanal_006	[01] IN06	Passive		Full bridge - Linear	±1000 "mV/V"	10 kHz - AAF - 81...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	None	<input checked="" type="checkbox"/>
Kanal_007	[01] IN07	Passive		Full bridge - Linear	±1000 "mV/V"	10 kHz - AAF - 81...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	None	<input checked="" type="checkbox"/>
Kanal_008	[01] IN08	Passive		Full bridge - Linear	±1000 "mV/V"	10 kHz - AAF - 81...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	None	<input checked="" type="checkbox"/>

Below the table, the 'Structural Analyzer' sub-tab is active, showing settings for the 'Force' channel. The 'I/O' section has 'Input' checked. The 'Windowing' section shows 'Exponential' windowing with T1 = -10 ms, T2 = 809.2 ms, and Texp = 1000 ms. A graph shows the exponential window curve.

On the page "Options", additional important settings for a modal analysis can be made, such as the measurement count, impact quality, voice commands for warning, readiness status, and DOF-designation, as well as the settings for the overload and the double impact. Additionally, more display and data storage options are available here:



The description of the options is provided [here](#)¹³³.

7.6.2.2 Defining DOFs

- To manage the DOF-page correctly, the experiment must be re-prepped:



- On the DOFs page, both fixed and roving DOFs are defined.
- In this example, there is only one fixed measurement point, at which we can use the accelerometer to measure and systematically analyze the structure's vibration response to the signal excitation.
- The fixed measurement point in the example was randomly selected as the point on the structure numbered 131.
- Three fixed DOFs are defined for this fixed measurement point. It comprises the vibrations of the structure in the directions X, -Y and -Z measured with the triaxial accelerometer.

- In this example, "Number of points" represents the number of points on the structure which are excited using the impact hammer.
- "Starting point" is the number of the point on the structure where the first impact is delivered.
- "Increment" is the number of points which are skipped between successive impacts/measurements.
- In this example, the Structure File is imported from the ME'scope structural model (1). Later, it will be possible to delete any points not relevant to the measurement and to define a DOF for the desired impact points.
- **Note:** The force signal's direction is the Z-direction (impact direction: from up to down)

The screenshot shows the imc WAVE Structural Analyzer interface. The 'DOFs' menu is highlighted with a red arrow. The 'Fixed DOFs' section has 'Measurement point' set to 1 and 'All' checked. The 'Roving DOFs' section has 'Starting point' set to 1, 'Increment' set to 1, and 'Number of points' set to 3. A 'Select Structure File' dialog is open, showing a file named 'Structure1.txt' selected. The 'Fixed DOFs' table is as follows:

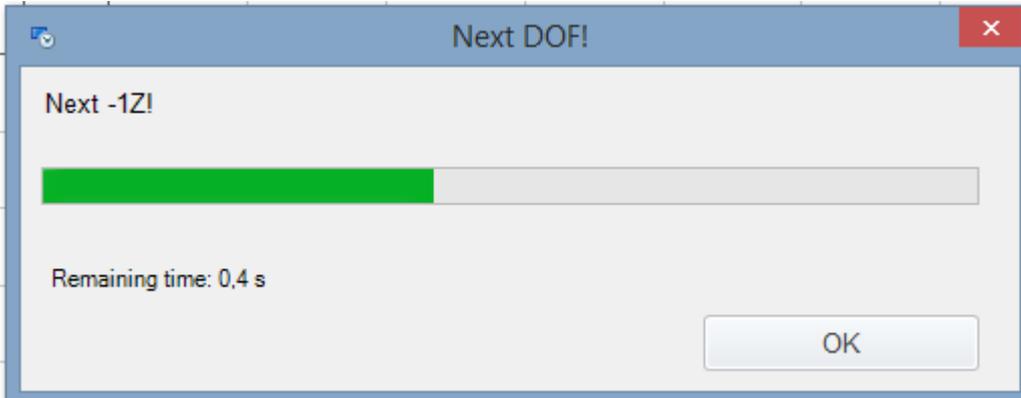
No.	Channel name	DOF
1	Acceleration_x	
2	Acceleration_y	
3	Acceleration_z	
-	Force	Input
-	Kanal_005	Passive
-	Kanal_006	Passive
-	Kanal_007	Passive
-	Kanal_008	Passive

The 'Roving DOFs' table is as follows:

Nr.	Force
1	-1Z
2	-2Z
3	-1Z
4	-2Z
5	-3Z
6	-4Z
7	-5Z
8	-6Z
9	-7Z
10	-8Z
11	-9Z
12	-10Z
13	-11Z
14	-12Z
15	-13Z

7.6.2.3 Measurement

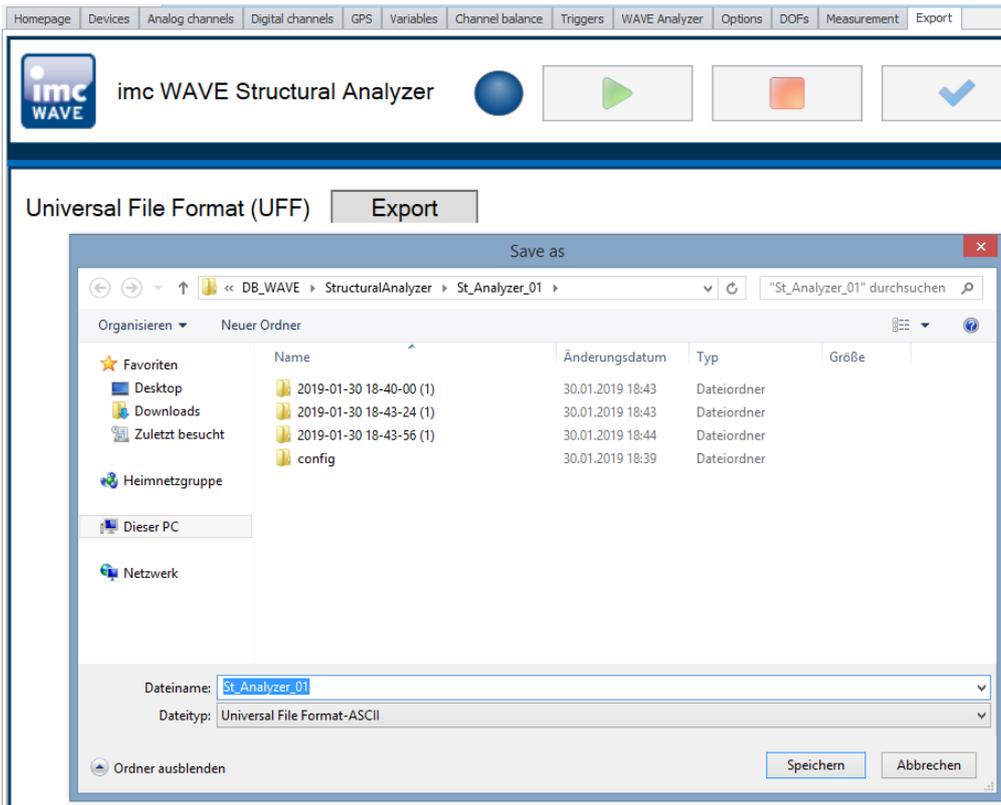
- Ensure that all channels are to be saved before starting measurement.
- Make settings on the Measurement page to produce an optimal signal display.
- Start the measurement and wait briefly until the displayed DOF-number disappears.



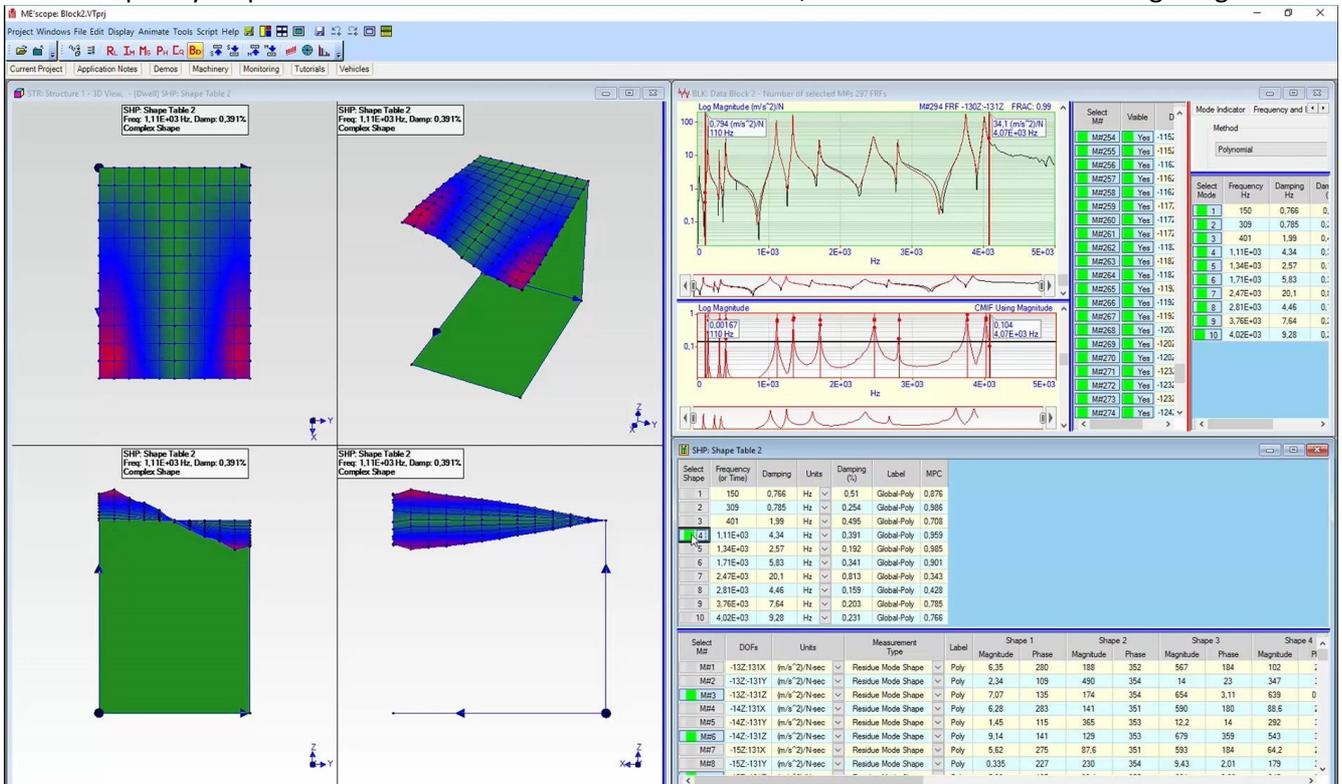
- Start the measurement by delivering the first impact at Point 1 on the structure.
- When you click on "Next", the software will guide you from one measurement point to the next until all measurement points are done.
- The correlation between all synchronous signals (outputs), including the excitation (input), enables the frequency responses to be determined.
- The frequency response (impulse response) functions comprehensively describe the structure's vibration behavior (resonances, eigenfrequencies, modes etc.).



- At the end of the measurement, you can create the (UFF) file containing the frequency response functions (FRFs) results. To do this, go to the page *Export*.



- You can then import the UFF-file to ME'scope, for example. There, a modal analysis is performed on the basis of the frequency responses and of the structure's vibration modes, as illustrated in the following image.



8 Setup - Device (general)

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

Chapter overview

Synopsis	Section
Which devices are supported?	<ul style="list-style-type: none"> • Device Overview ¹⁴⁶
Getting familiar with concepts concerning system operation	<ul style="list-style-type: none"> • Operation ¹⁶¹
What steps are necessary for performing a measurement?	<ul style="list-style-type: none"> • Performing a measurement: Procedure ¹⁷¹
Saving supplemental information on the measured data	<ul style="list-style-type: none"> • Saving metadata with the channel ¹⁷³ • Comments on the measurement - Measurement metadata ¹⁷⁶



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](#)" ¹⁸⁴



Note

If the documentation mentions a device, the comments apply similarly to other devices as well.

8.1 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 WAVE.

— 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	EOS	ARGUSfit
Driver package	imc DEVICES																		imc DEVICEScore		
Firmware group	A																		B		
Device group	A4				A5				A6	A7				B10	B11						
SN ¹	13				14				16	19				4120	416						
TCP/IP Interface [MBit/s]	100				100				100	1000				1000	1000						
Sampl.Rate ² [kHz]	400				400				2000 / 400 ³	2000 / 400 ³	2000 / 400 ³	2000	2000	2000	4000	5000					
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

8.2 Actions and Buttons

8.2.1 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.

Action	Page
Process configuration (✓)	all

8.2.2 Connect and Disconnect

imc STUDIO connects itself with/disconnects itself from all selected measurement devices (normally via LAN).

 **Note**

If the measurement device's software version is different than that of the product installation on the PC, the [Firmware Update](#)⁵⁷ follows next.

Change of the imc measurement device

When connection takes place, the system checks whether the previously registered device and the actually connected device match. There can be discrepancies when modular systems are connected. If a difference is detected, the current device setup needs to be determined. Subsequently, the changes are reflected in imc STUDIO.

 **Warning**

For devices of the [firmware group A](#)¹⁴⁶

Please note that for this reason it may not be possible to apply the existing experiment settings. For instance, if the device previously had a bridge amplifier and it was replaced with a temperature amplifier

 **Note**

For devices of the [firmware group B](#)¹⁴⁶

For transfer of the configuration from imc CANSAS*fit*- and imc ARGUS*fit*-modules which are attached to an imc ARGUS*fit* unit, the following limitation applies.

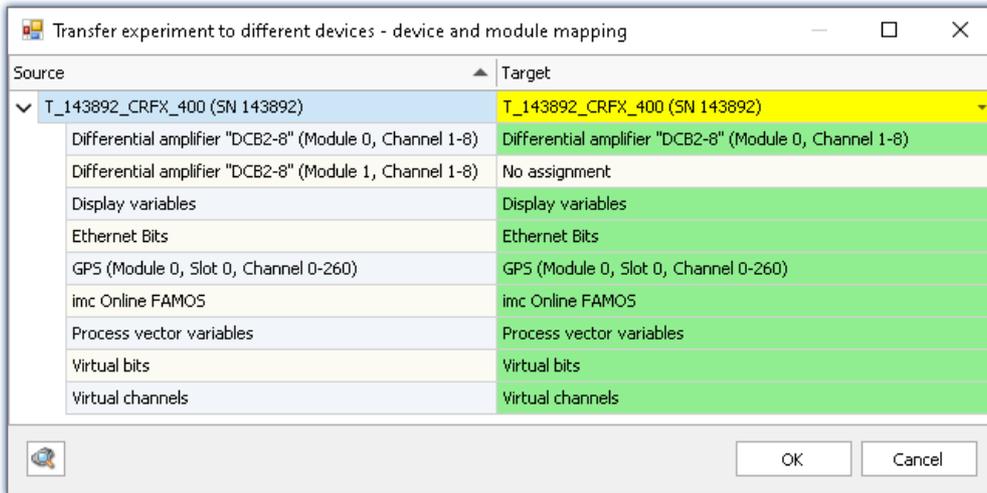
The configuration of a module can only be transferred when the following conditions are met:

- same module-type (e.g. ARGFT\UTI-6-SUP to ARGFT\UTI-6-SUP)
- same device family, meaning imc ARGUS*fit* only to imc ARGUS*fit* and imc CANSAS*fit* only to imc CANSAS*fit*
- same scope of functions (e.g. same connection terminal)

The module's hardware revision stated on the plaque does not matter.

When changes are detected, a prompt appears which asks you how to treat the current configuration. You can **discard** the configuration or **transfer** it to the "current, changed" device. In this case, transfer the configuration to the device.

A module assigning dialog appears. If just a module was added, you do not need to make any changes here. But if you have replaced a module or if internal hardware has been replaced as part of some servicing work, please verify the assignment.



Transferring in the case of a changed device configuration

At left you see the older device and at right the new one.

Below the respective device, you see a list of all of a device's "Modules" and components. What this list contains depends on the device's hardware configuration. Use the drop-down-list at right to select the appropriate modules.

The colors:

Green: All of the module's settings can be transferred.

Yellow: Not all of the module's settings can be transferred. For example, the amplifier type may be different.

Please ensure that the target module is able to use the configuration required by the source module. E.g. a universal amplifier can use some of a temperature amplifier's settings.

Red: This assignment should not be selected.

What settings are applied:

In principle, all settings are applied, except for the **calibration values** (taring, bridge balancing, two-point scaling, ...). Settings which can not be applied are ignored and listed in the **log book** following the transfer.

Connected while the device is running a measurement

If an attempt is made to establish a connection with a device which is already running a measurement, the measurement is stopped while connecting. A corresponding confirmation prompt is posted beforehand.

 **Warning**

When imc STUDIO establishes a connection with a running measurement, the current settings are discarded and the running experiment's settings are loaded.

8.2.3 Download and Reconfigure

Before first starting the measurement, the configuration must be loaded in the device. The device is initialized automatically if the configuration in the device is not up-to-date. If the device contains the current configuration, no initialization is performed.

In order to prepare the device, use one of the two possibilities below:

Action	Description
Reconfigure	All settings are analyzed and completely imported by the device.
Prepare	Changed settings are analyzed and completely imported by the device. If no changes are detected, the device's configuration is already up-to-date. The actual preparation process is not carried out in that case. If you wish to have it carried out anyway, you must run "Reconfiguration".

Action	Page
Prepare (📁) - Move the mouse over the header and press the CTRL-key	Complete
Dialog: Device options > Reconfigure	Devices

Note

- Under some circumstances the action "[Process configuration](#)¹⁴⁷" is automatic executed first, if this had not been done already.
- For preparation and reconfiguration, the PC must previously have set up a connection to the device (see "[Connect and Disconnect](#)¹⁴⁷").

FAQ

Question: The following error message is being posted: "The device has been shut down inaccurate in the past. Please check the uninterruptible power supply of your device!" What should I do?

Answer: When you shut down the device, it needs a few seconds to power down. During this time, the device is being powered by the internal UPS. If the UPS is defective, the device can not power down correctly and shuts down immediately.

Please check whether the device goes off immediately upon tripping the main switch, or only after one second or longer.

If the device goes off immediately, then presumably the UPS is not working correctly. Please get in touch with our [tech support](#)⁷.

Reference

See also:

Executed once at the start: "[OnInitAll](#)⁷⁰⁰"

8.2.4 Start and Stop measurement

The measurement is started/stopped for all devices.

 **Note**

- It may be necessary to carry out the "[preparation](#)"^[149], unless this has already taken place.
- For starting and stopping the measurement, the PC needs to have established a connection with the device (see "[Connect and Disconnect](#)"^[147]).

Action	Page
Start ()	all
Stop ()	all

Additional ways to start/stop a measurement

- via the column "[Device control](#)"^[187] in the devices-table
- using the Sequencer command "*Execute device action*". See the example: "[Starting measurement](#)"^[1178].

In contrast to the menu items, these actions do not automatically affect all devices. 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.

Also, these actions are not performed in synchronization with each other. The devices each either start or stop separately and as quickly as possible.

Measurement status (running, stopped)

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

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

 **Note**

6 flashes during measurement

During a running measurement, LED 6 flashes at a 1-second rhythm. This provides a simple visual indication of whether the measurement is running.

LED 6 does not flash,

- if it is used in the imc Online FAMOS source code,
- if flashing is deactivated in the imc Online FAMOS Options,
- if imc Online FAMOS is disabled.

8.2.5 Search for devices

The device 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.

Action	Page
Dialog: Device options > Search for devices	Devices

E.g., after the first launch of imc STUDIO, the [device table](#)¹⁸⁶ is empty. Before you can create an experiment, you must enter one or more devices into the table. To do this, perform a device search. The device table lists all devices found. The image below shows a typical situation:

Selected	Device name	SN	Device specification
<input type="checkbox"/>	T_124835_C1_1_LEMO_ET	124835	imc C1-1 LEMO
<input type="checkbox"/>	T_130039_busDAQ_X	130039	busDAQ-X
<input type="checkbox"/>	T_130311_SPARTAN_U32_CAN	130311	imc SPARTAN

Results of device search (example)

Reference

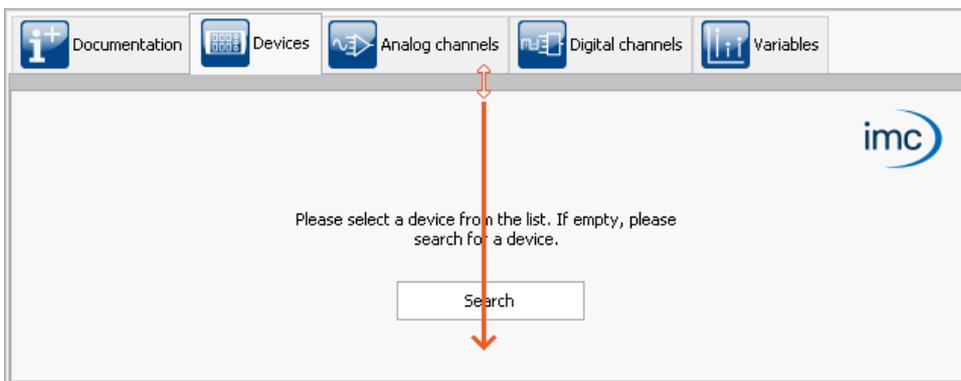
- How to correctly set **your device's network configuration** and what to make note of is presented in the chapter: "[Setting Up - Connect the device](#)"³⁹
- For more information on **selecting the device for the experiment**, see the chapter: "[Device Search - Known and Selected](#)"¹⁷¹

FAQ

I am shown the message: "Unable to find any new devices."

Answer: In some cases, the Devices/Channel list can be hidden. It depends on the view and the Windows display settings applied.

Please check whether the list may only be covered over:



The slider is all the way at the top - The lower window covers the Devices list

8.2.6 Search for devices by IP/DNS

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 "[Device Table](#)" and to establish a connection.

Action	Page
Dialog: Device options > Search for devices by IP/DNS	Devices

Reference

More information is presented in the chapter: "[Setting Up - Connect the device](#)" > "[Connection via a direct address](#)"

8.2.7 Device properties

That dialog shows the configuration of the device. Most properties depend on the hardware. For example changing the type of the display is only useful, if the device is not equipped with an internal display.

Action	Page
Dialog: Device options > Device Properties	Devices

Please contact our [tech support](#) before changing the settings (with the exception: "Access protection", "Display", "Device time, Synchronization").

Note

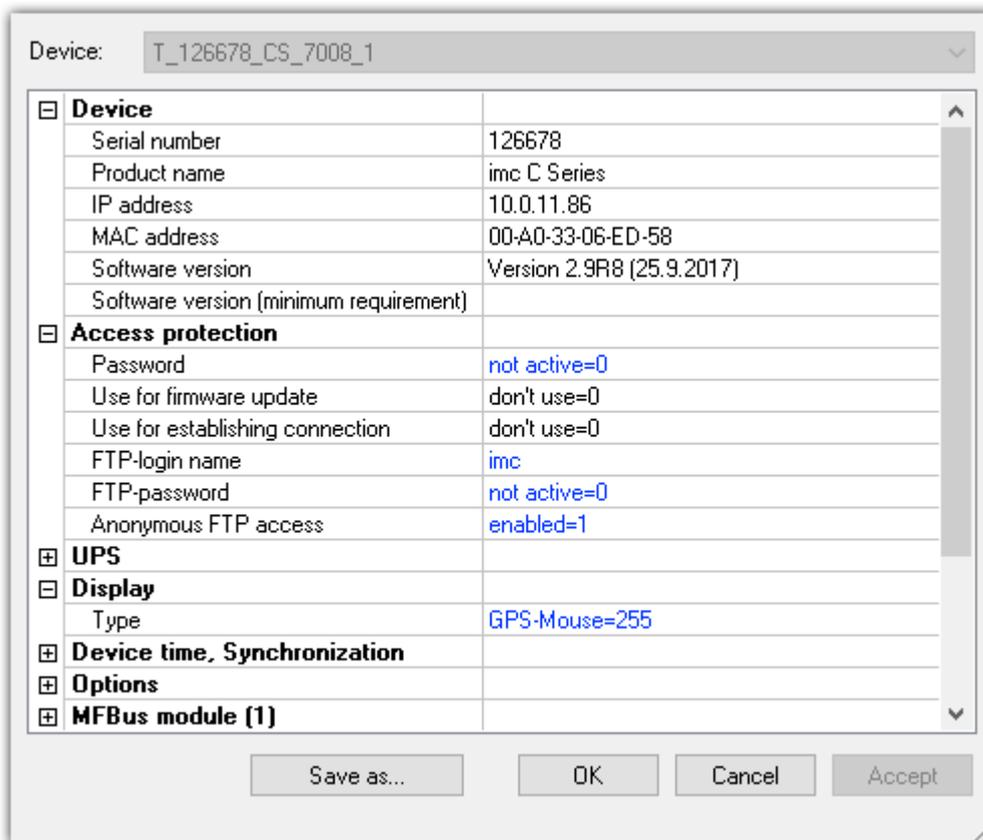
Restart

All changes of the settings need a restart of the system!

Log file

Whenever changed device properties are adopted, a text file is generated. All information on the device properties currently set is saved as soon as the button "Apply" in the Properties dialog is clicked.

- Storage location: Firmware folder of the device software, e.g. "C:\Program Files (x86)\imc\imc_DEVICES_2.15R1\Firmware".
- Name of the file: "PropertiesLog_<Devicename>_yyyy-MM-dd_HH-mm-ss.txt"



Device properties

Property: Device	Description
Serial number	Serial number of the device. Can't be changed.
Product name	Device types: imc C-SERIES, imc CRONOScompact etc.

Property: Device	Description
Software version	Version of the firmware.
Software version (minimum requirements)	The minimum required software version to run the device as configured. This prevents downgrading to a version which would not support parts of the device.
GPS receiver	Here, any available information on the GPS receiver, for example the identifier and the version, is displayed.

Property: Access protection	Description
Password	For protection against firmware update or connection setup. To delete the password, enter the valid password again.
Use for firmware update	Enable and Disable of firmware update 
Use for imc REMOTE WebServer	Enabling/disabling of operation via the device's Web-Interface.
Use for establishing connection	Enabling/disabling of the connection to the device (FTP, LAN, ...). <div style="border: 1px solid blue; padding: 5px; margin-top: 10px;">  When the device is newly integrated in consequence of a device search, it is necessary to enter the password twice. The first time is for establishing the connection in order to "know" the device, and the second time is for the purpose of entering the device in the list. </div>
Certificate	The certificate needed for setup of a connection via https (e.g. via imc REMOTE). Here, select the certificate "imcCert.cpt", which is in the installation path under "Firmware\ldif", e.g. "C:\Program Files (x86)\imc\imc_DEVICES_2.14R1\Firmware\ldif". The certificate is always valid for one year and must be replaced upon expiration. The certificate is obtained from our tech support  .
Expiration date	Expiration of the above mentioned certificate.
FTP-login name	Login name for FTP access
FTP-password	Password for FTP access
Anonymous FTP access	enabled=1: No prompt for "FTP-login name" and "FTP-password" disabled=0: The "FTP-login name" and "FTP-password" are used

Property: UPS	Description
Acquisition time after power fail [s]	Maximal measurement time after loss of supply voltage. The factory set time can be changed by the user. This entry has to be aligned with the hardware, thus may not be changed without consultation with imc. In case the power of the accumulator is not sufficient , data of devices will still be saved.

Property: Display	Description
Type	Type of display; alphanumerical, graphical and its resolution. The alphanumerical display is not supported for all devices. Devices having an internal display can only control one type. Technically an internal display is a "normal" display with a mounted cable. So even for these devices the display type can be changed. That connector could be used for a GPS-mouse , too. To change it, select the entry "GPS-Mouse=255". That option is not available for all devices. In case of doubt contact our tech support  . After switching to GPS, all GPS signals can be received by the process vector variables.

Property: Device time and synchronization	Description
Time zone	<p>Sets a time zone.</p> <p>In order to make it possible to assign a time to measured data, the measurement can only be prepared when a time zone is set for every imc device used.</p>
Automatic daylight saving time switching	<ul style="list-style-type: none"> • activated=1: the device automatically switches to/from daylight saving time. • deactivated=0: the device uses standard time all year round!
NTP	<ul style="list-style-type: none"> • Server (1), (2): IP address or name of server • max time discrepancy [ms]: States the permitted time discrepancy from the NTP-Server in ms, up to which the device is still considered synchronized. When this discrepancy is surpassed, the message – "Not synchronized" or "Synchronization in progress.." .. is displayed. For the sake of the measurement precision and comparability of measured data, we recommend increasing the value of this entry if the discrepancy is big. Otherwise it will be necessary to dispense with synchronization completely. • Interval[s] (min;max): Specifies the minimum and maximum time for the synchronization intervals, to which the following applies: <ul style="list-style-type: none"> • Short intervals provide higher accuracy, but make more demands on the network. • Long intervals may produce lower accuracy, but make less demands on the network. <hr/> <p>Accuracy:</p> <ul style="list-style-type: none"> • If the external conditions (mainly temperature, network resource requirements) only fluctuate slightly, then the accuracy may be high even with long intervals. Shorter synchronization intervals should be selected if, for instance, frequent temperature fluctuations are to be expected. • NTP-Standard: min = 16 s, max = 1024 s. This value is also used if nothing was entered. This is adequate for constant temperatures, e.g. for installation of the devices in a cabinet. <hr/> <p>Example: NTP setting: min=8 s, max=64 s</p> <p>Conditions:</p> <ul style="list-style-type: none"> • Devices were exposed to daytime-based temperature fluctuations, in other words, they were not installed in a cabinet. • Use of a company's internal NTP-server <p>Result: Among 3 devices, there are signal timing discrepancies of 0.01 .. 0.2 ms. This value is far below the uncertainty level stipulated in the standard IEC 61000-4-30 and enables very good comparison of measurement signals.</p>

Property: Device time and synchronization	Description
Max. time to wait for synchronisation [s]	<ul style="list-style-type: none"> • 0: unlimited wait (default) • -1: don't wait • xxx: wait xxx seconds <hr/> <p>Note:</p> <ul style="list-style-type: none"> • Select a waiting time which is certain to work for synchronizing with an operational synchronization input signal. Otherwise, after the waiting time the measurement will begin out of synch. However, this can only be subsequently recognized in the measured data if you record the Sync state along with the data by means of the imc Online FAMOS function IsSynchronized. • For a GPS Master with DCF-77 Slaves, a waiting time of 300s is recommended.



Synchronization signal input	<p>Specifies a default clock to be used upon activation. If the value for synchronization signal is n.s.=0, the settings remain intact after activating the device.</p> <ul style="list-style-type: none"> • 0 none: not defined. • 1 no synchronization signal: Synchronization signal input not evaluated • 2 SYNC synchronization signal: only DCF or IRIB-B is evaluated • 3 GPS synchronization signal: only GPS receiver evaluated • 4 SYNC or GPS synchronization signal: depending on which signal is applied first upon deactivation, either DCF, IRIB-B or GPS is utilized • 5 NTP synchron signal: Network Protocol used • 6 EtherCAT synchronization signal • 7 PTP: Precise <p>These setting are applied in the experiment when connected with the device and there are no other settings regarding Synchronization ^[188] had been specified for the experiment.</p>
------------------------------	--

Synchronization signal output	Specifies whether the device time is outputted as a DCF/IRIG-B signal.
-------------------------------	--

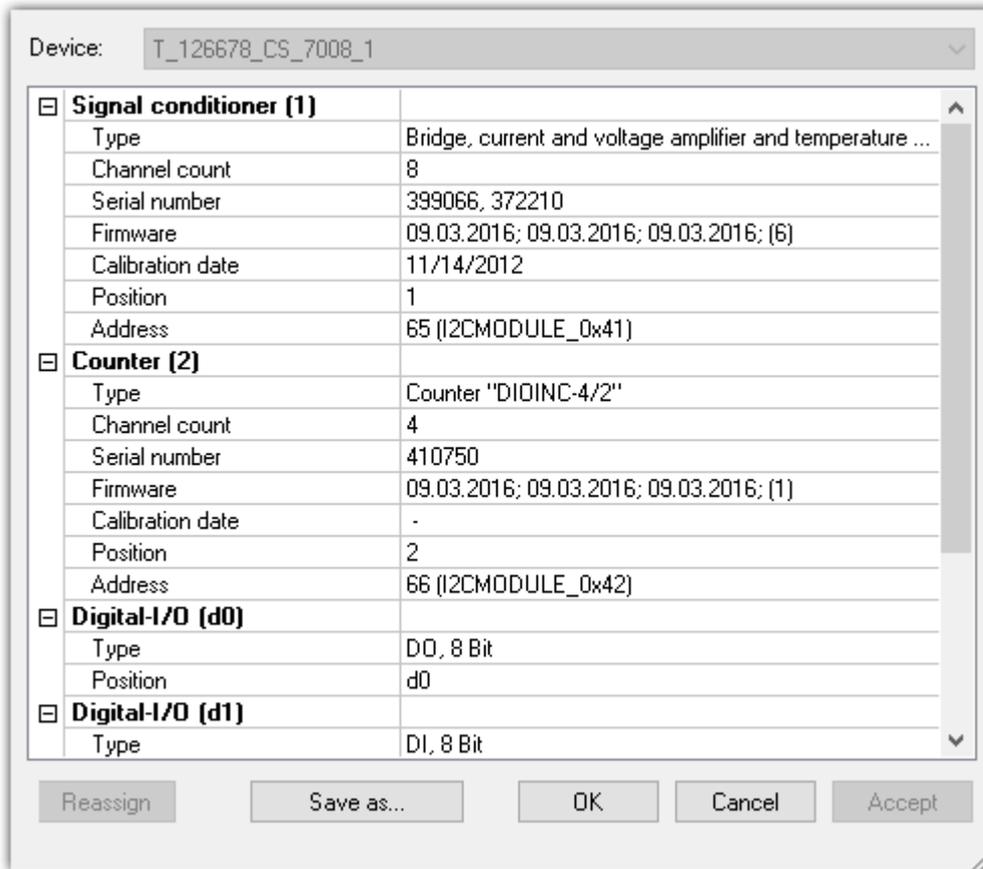
Property: Options	Description
<p>Some software options come at an extra charge. These options are enabled by means of so-called enabling codes. The enabling code is a multi-digit number and is determined for each device on the basis of its serial number.</p> <p>Options can be enabled at factory and be available when the device is delivered (see the decal on the device frame new the serial number plaque).</p> <p>The options can be enabled after delivery. Enabled options are noted in the device's configuration (Flash-EEPROM) and thus are permanent and cannot be lost.</p>	
imc Online FAMOS	Enable Online processing with imc Online FAMOS ^[498] base packet.
imc Online FAMOS Professional	This option comprises multiple subparts ("Optimization of the online programs by use of the on-chip memory", "Online Synchronous task", "Process vector"). The availability of these function option subparts, however, depends on the particular hardware configuration and is determined upon connecting with the device (see the section on " imc Online FAMOS ^[498] ").
Online classification	Enables functions of Online Class-counting Kit (see the section on " imc Online FAMOS ^[498] ").
Online-Order tracking	Enables functions of Order Tracking Kit (see the section on " imc Online FAMOS ^[498] ").
CAN-Database Import	Enables Vector-database access, DBC-format; order option */VEC-DATB (license per device)

Property: Options	Description
imc STUDIO Monitor	Shows, if the devices is prepared for imc STUDIO Monitor. That entry can not be changed.
ECU-protocols	Activates the functions for CAN-bus subscribers which support the ECU protocol (see the section on " imc Online FAMOS ^[498] " and the " imc Online FAMOS function reference ^[542] ").
Measurement time (T3) until suspend (s)	Only available to devices having a sleep-mode. Describes the minimum duration for which the suspend signal must be applied.
imc REMOTE	<p>Activation of imc REMOTE LinkSecure or imc REMOTE WebServer. This requires an activation code which must be purchased along with the device. It is also possible to perform the activation subsequently.</p> <ul style="list-style-type: none"> • imc REMOTE LinkSecure: Option for imc LINK providing secure access via an https connection • imc REMOTE WebServer: Enables imc REMOTE WebServer which provides platform-independent remote access to imc measurement systems. <hr/> <p> If imc REMOTE WebServer is enabled in the device properties, a connection is permanently occupied. The maximum number of connections to the device depends on the device - see "Device overview^[146]".</p>

8.2.8 Module properties

The dialog lists all amplifiers equipped with the device.

Action	Page
Dialog: Device options > Module Properties	Devices



Module properties

Along with the "type designation", "serial number", "firmware" and "calibration date", you will find the "address" (with bus designation) and the resulting "position" of the amplifier.

By using "Save as..." a file is created which lists the connected amplifiers

There are two groups of amplifiers:

- **Permanently installed amplifiers** use an address switch on the amplifier, which is set upon delivery of the system. With some devices, this can be changed by trained users (see "Device manual"). A permanently installed amplifier is distinguished by the address designation: **I2CMODULE**.
- **Modules** (e.g. [imc CRONOSflex or CRONOS-XT modules](#)^[241]) are integrated with a master (e.g. CRFX/CRXT Base unit or with an EtherCAT Master (ECAT-DA) for imc CRONOScompact). They obtain their module address automatically. The numbering of the module occurs upon first connection and remains intact. Along with assignment of the module number (Position), a "ScanID" (identifier for the module) is also allocated. A 7-segment display represents the module address. A CRFX/CRXT module is distinguished by the address designation: **xbus**.

Assigning the module number

- **Reassign:** The button "Reassign" overwrites the existing numbering of the modules with an uninterrupted re-numbering.
- **Assign manually:** Enter the position manually. Click the number and choose a position from the list.

Signal conditioner (3)	
Type	Bridge, current and voltage amplifier and temperature measurement unit "CRFX/UNI2-8", Linear
Channel count	8
Serial number	150101, 311424, 436294
Firmware	09.03.2016; (6)
Calibration date	07/18/2013
Position	3
Address	3
	4
	5
	6
	7
	8
	9

Assign manually

Warning

Note that changing the module configuration is equivalent to re-ordering the amplifiers. If a change in the order of module entries is made, any experiments present can no longer be used.

Removing CRFX/CRXT modules

If an module is removed, the device description retains the module numbering. A gap results in the numbering. Since the device description is saved along with the experiment, the old experiment can easily be reused in any later application of the same module.

Exchange of CRFX/CRXT modules

The device description identifies an module by its serial number. After exchanging an module with another of the same model, the new module is recognized as such by its serial number and receives a new address. In this case, correct the address (*Assign manually*), **before** resuming the use of existing experiments.

**Note****Notes on assigning the module number**

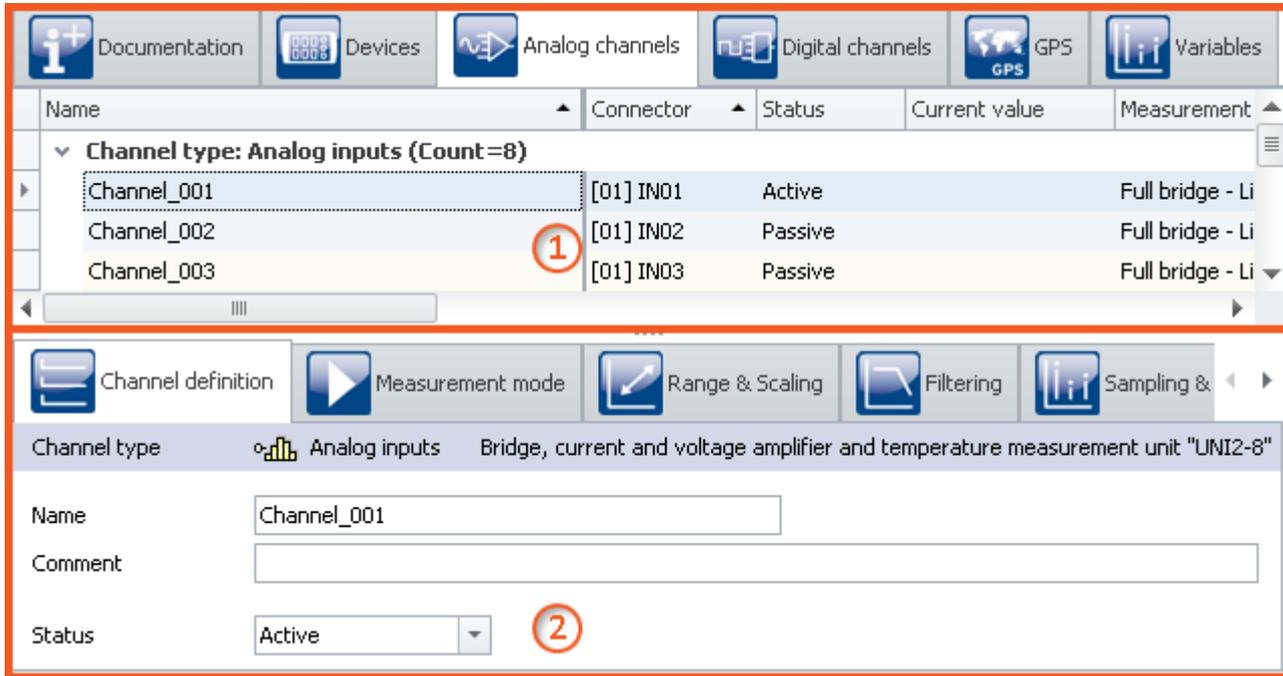
- If an CRFX/CRXT module is alternately used with different imc devices, it saves the module address for each measurement device separately. The number of module addresses is limited to 63. If the number of possible module addresses is exceeded, the oldest entry is deleted. A module can thus have different module addresses in different device configurations. The use of the same module in different devices is thus possible without needing reassignment of the module address each time.
- If the ScanID matches the module address saved on the module is used.
- If there is a conflict in the ScanID (multiple modules having the same module address), then *Reassign* of the module addresses is performed.
- For all other modules (non-matching of the ScanID), the module addresses are reassigned.
- When assigning module addresses, the last address used is applied. The newly assigned addresses follow (are appended to the end of) the last address assigned.
- When the module addresses are reassigned, a new ScanID is made and all modules are assigned a new address.

**Note****Notes on changing the device configuration**

- After the device software connects with a device, it is at this time not possible to take any changes of the device configuration (removal and/or addition of modules) into account (e.g. event notification with subsequent adaptation of the settings dialogs).
- If the device configuration is changed, changes are only taken into account after separating from and reconnecting the device with the device software
- During a running measurement, it is not possible to respond to changes of the device configuration (no "hot plug")!

8.3 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:



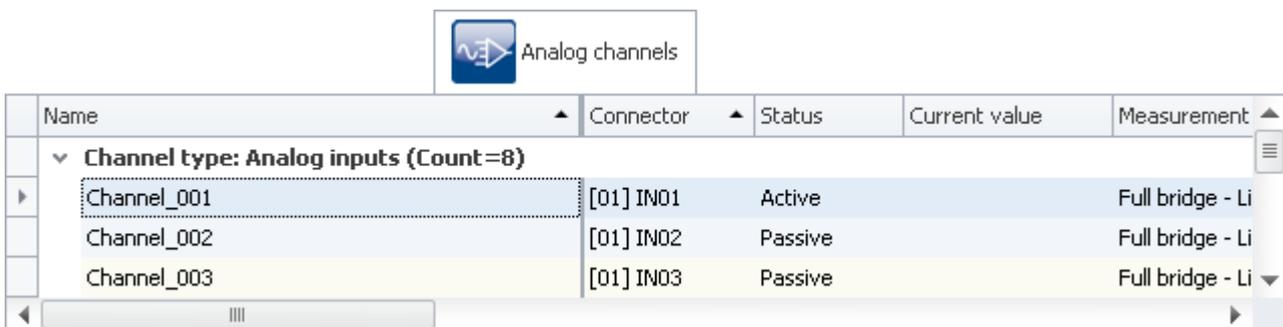
Tabular display and dialog (example)

Region	Description
1: Tabular display ¹⁶⁷	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 ¹⁶⁸	The lower window shows special dialogs ¹⁶⁸ for the most important parameters.
Assistenten ¹⁷⁰	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](#) ¹⁶⁴ in the table, as well as [in the dialog](#) ¹⁶⁹. To edit the height of the window, use the mouse to drag the dashed separator line.

8.3.1 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](#) ¹⁶⁴ both individual parameters and sets of multiple parameters (see [Selection](#) ¹⁶²) 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](#) ¹⁶⁵ to include the desired parameters in the table. Also, you can configure complex columns using the function Additional columns.

8.3.1.1 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.

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

While holding down its button, drag the mouse over the region

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

Release the mouse button at the position desired

Selection with mouse and keyboard

To do this, use the left margin column, as shown in the first image.

Selecting individual rows

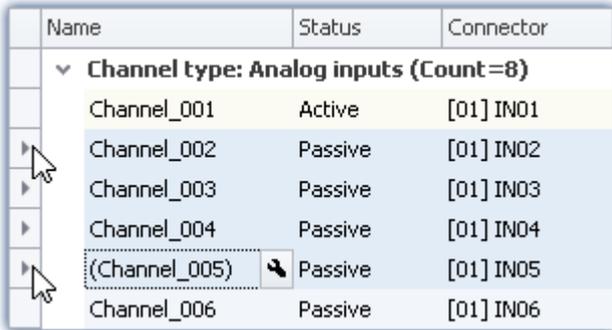
Click on the left edge of the desired entry:

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

Single line selected
(simply click on the edge of the table)

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.



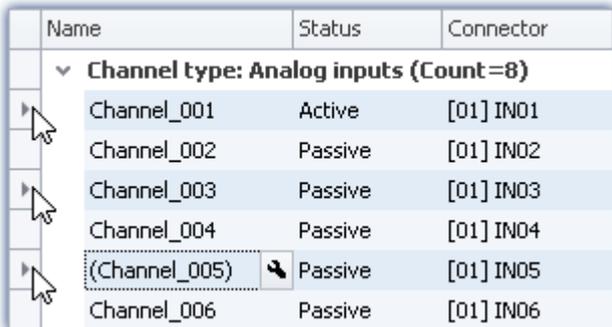
Name	Status	Connector
▼ Channel type: Analog inputs (Count=8)		
Channel_001	Active	[01] IN01
Channel_002	Passive	[01] IN02
Channel_003	Passive	[01] IN03
Channel_004	Passive	[01] IN04
(Channel_005)	Passive	[01] IN05
Channel_006	Passive	[01] IN06

*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

8.3.1.2 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](#) 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 (dropdown menu)

Control for multiple selection
(example)

Select the desired row as explained in the previous chapter ([Selection](#)). 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: ≠ (dropdown menu)

Different parameters in a multiple selection procedure
(example)

8.3.1.3 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 "<%TARGETTITLE%>"). 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 (general) > chapter "Views").

8.3.1.4 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?"
- See: "Views"

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.

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](#) ²⁸ product versions.)

8.3.1.5 Filtering columns

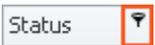
Note

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".

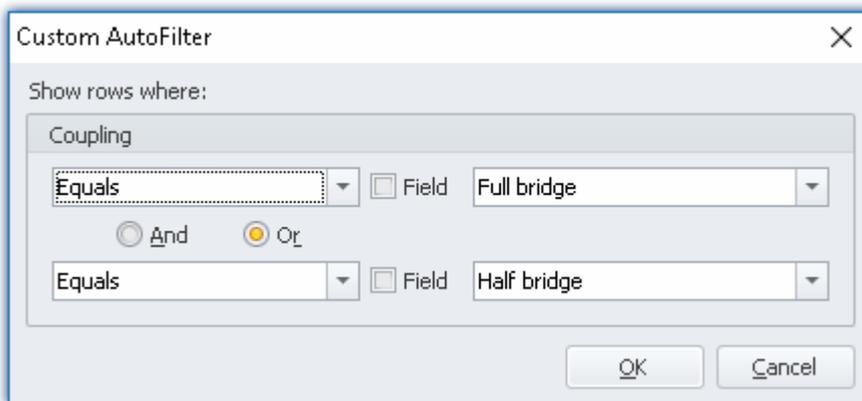
Activating/deactivating: Using the checkbox in the bar, you can activate/deactivate the filter without removing it.

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.

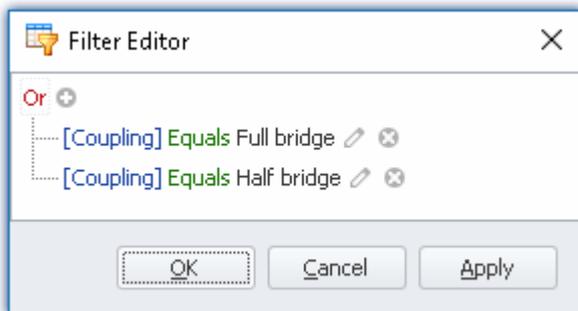
The following dialog appears. In this dialog, configure the filter as desired and then click on "OK". Note that the filter only works with exact values/texts.



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

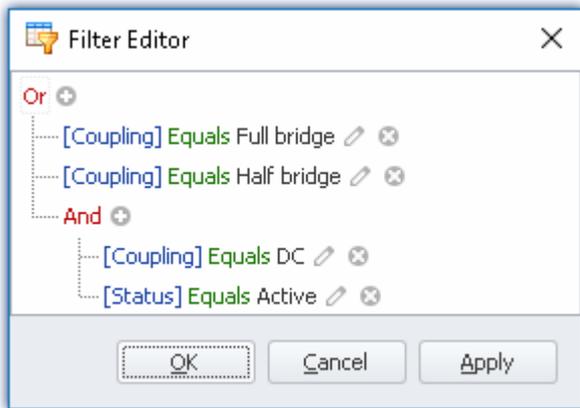
Using this dialog window, you are able to formulate complex combinations of filter conditions.

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 **X** 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.



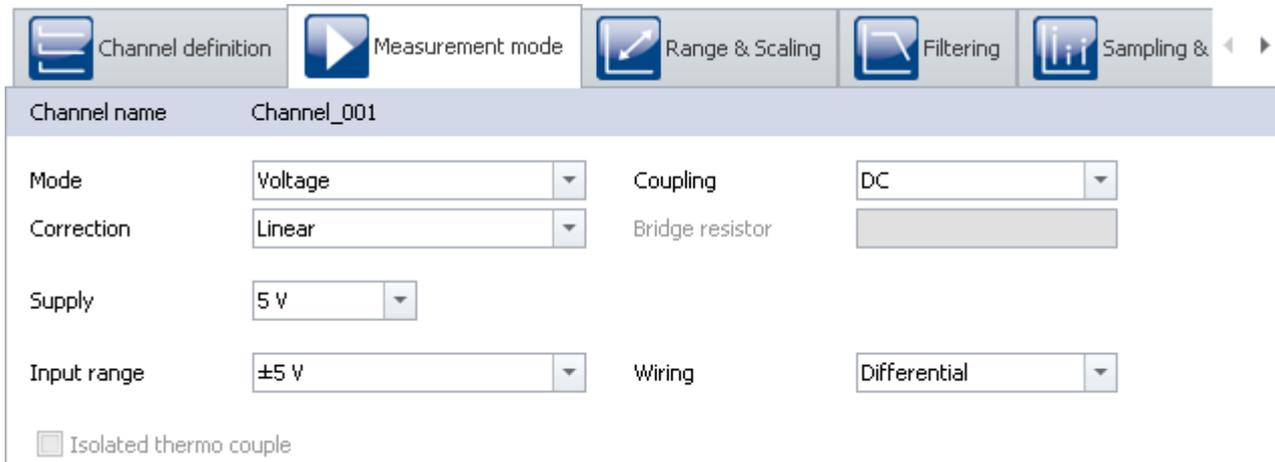
Filter setting with two groups

*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.

8.3.2 Dialogs

In this section, the most important parameters of special dialogs are offered.



Example of "Measurement mode" dialog

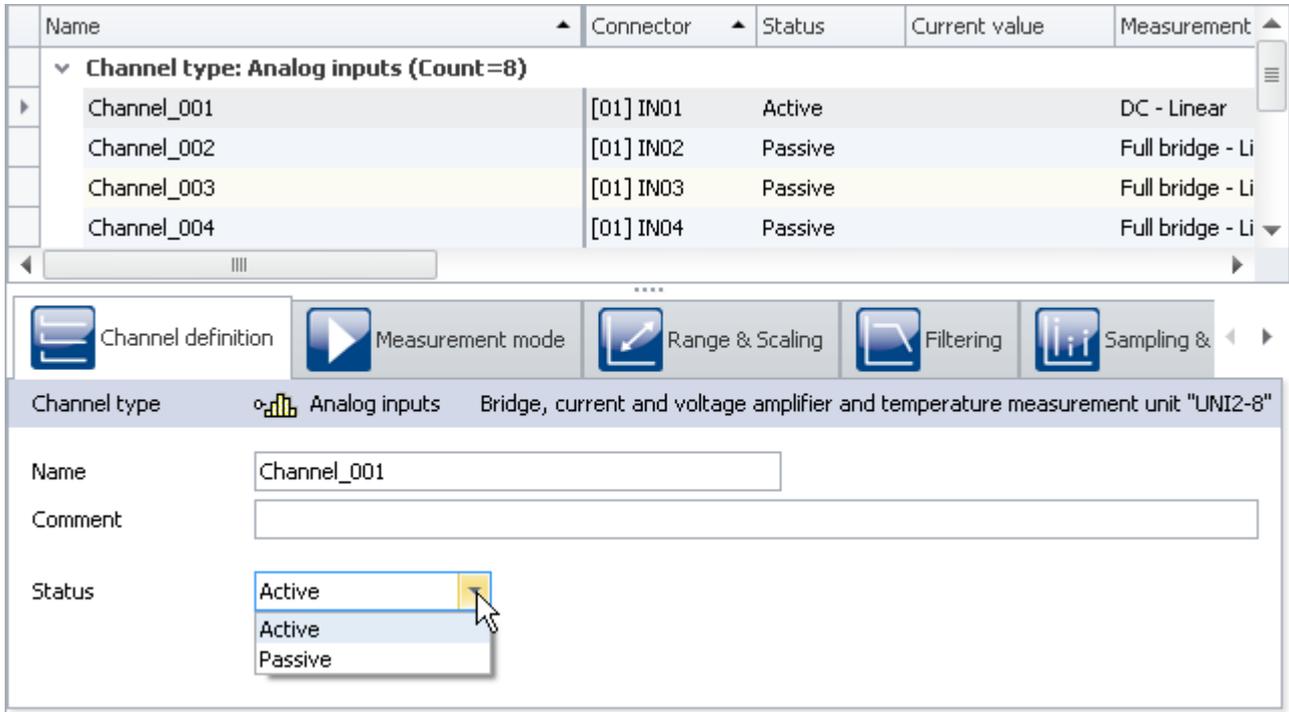
Selected parameters are presented here, distributed over multiple dialogs. The dialogs are designed in a clearly structured manner. As with the tabular display, you can also [directly edit](#) ¹⁶⁹ parameters in the dialogs.

Availability and selection of the dialogs and the settings depends on what page is open and on the selection in the table.

8.3.2.1 Direct editing

You can directly edit (change) all parameters displayed in the dialogs.

Editing individual parameters



Editing individual parameters (example)

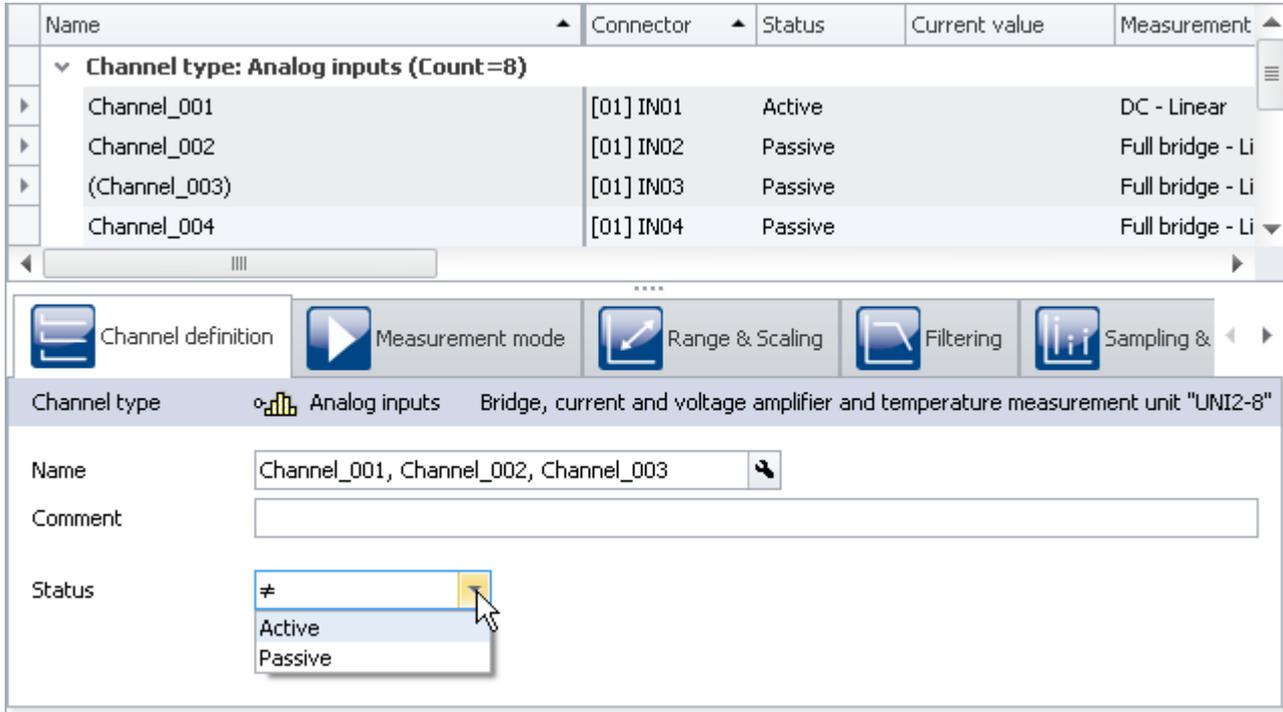
To edit a parameter, first select the corresponding table cell in the tabular display (see [Selection](#)¹⁶²). The dialogs associated with the selection are indicated. Open the dialog in which the parameter to be changed is found.

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 (≠).

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)

8.3.3 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.

8.4 Performing a measurement: Procedure

This chapter describes how the measurement procedure (the experiment) is controlled.

Action	Description
Connect signal leads	
Searching for and selecting devices ¹⁷¹	Before you can create an experiment, you must adopt one or more devices into the device table. To do this, run a device search and selection.
Firmware Update ⁵⁷	The software only works with devices having the right firmware. Upon the connection/preparation, the versions are compared. If they don't match, the firmware update is carried out.
Configure devices ¹⁸⁴	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 ¹⁴⁷	Current configuration is inspected for errors and made available to all the plug-ins. No connection with the device is required for this.
Connect ¹⁴⁷	imc WAVE connects itself with all measurement devices selected (generally via LAN).
Balance	For amplifier channels: balance the channels. When a system having amplifier channels is switched on, or the gain, gain function or filter function is changed, the offset needs to be balanced.
 Prepare ¹⁴⁹	The current configuration is exported to the device.
 Start measurement ¹⁵⁰	The measurement is started for all devices.
 Stop measurement ¹⁵⁰	The measurement is stopped for all devices.

8.4.1 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 " ¹⁸⁶ 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.

Making a device known

To make a device known, you must previously have launched a device search for it. Perform a "[device search](#)" ¹⁵¹ (). The system then searches for devices in the network.

Action	Page
Dialog: Device options > Search for devices	Devices

Finally, the devices found are listed. The following image shows a typical display:

Selected	Device name	SN	Device specification
<input type="checkbox"/>	T_124835_C1_1_LEMO_ET	124835	imc C1-1 LEMO
<input type="checkbox"/>	T_130039_busDAQ_X	130039	busDAQ-X
<input type="checkbox"/>	T_130311_SPARTAN_U32_CAN	130311	imc SPARTAN

Device search results
(example)

The devices listed now are not yet known!

To make a device which has been found "*known*", you can do one of the following:

- "*select*" the device (this means that the device is immediately selected for the experiment)

Once the device has been made "*known*", its setup is determined.

Note

Note on determining the device configuration

For the purpose of successfully determining how the device is configured, a temporary connection to the device is established.

If this is not possible, the device is not made known to the system. A corresponding entry appears in the log book. Possible reasons include:

- incorrect firmware version
- the device is in Measurement mode (which means that the device is currently performing a measurement)
- the device has established an active connection to a different computer

Now that the device is "*known*", it is available for selection, without the need for a search, the next time the program starts.

Note

Special case

Registering the device with the system by loading experiments

When an experiment is loaded in which an unknown (unregistered) device is used, the device is automatically adopted as a registered (known) device.

FAQ

Question: The device in an experiment has a different device configuration than my registered device. Will the registered device be replaced with it?

Answer: The registered device is only replaced when you establish a connection with the device. If an experiment with a registered device is loaded, but its configuration is different (e.g. CRFX with different modules), the registered device is initially not overwritten. The Device Table temporarily indicates a "new" device. When you restart imc WAVE, the original device appears once again in the table.

Only once you establish a connection with the device will the device list be updated. The previous state is discarded.

Device selection

To be selected for an experiment, a device must be "selected".

Selected	Device name	SN	Device specification
<input checked="" type="checkbox"/>	T_124835_C1_1_LEMO_ET	124835	imc C1-1 LEMO
<input type="checkbox"/>	T_130039_busDAQ_X	130039	busDAQ-X
<input type="checkbox"/>	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:

Selected	Device name	SN	Device specification	Connection status	Measurement status	Device control
<input checked="" type="checkbox"/>	T_126678_CS...	126678	imc C Series	disconnected	stopped	Connect

*Status indication in the columns
"Connection status" and "Measurement status"*

8.5 Metadata and Comments on the measurement

8.5.1 Saving metadata with the channel

Saving metadata with the channel

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](#)^[146]). 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", you have the choice of which groups of parameters to save.

Action	Page
Options (🔍)	Homepage

Option	Description
Traceability of data ^[1239]	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, 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 saved as "Link" or as "Link + in experiment", 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 **creation of the file**, 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**.

The exception is when data saving is resumed in a new file. In this case, the changed metadata are applied to the new file. This pertains to "Interval saving", for instance, and in the case of "*Suspend/resume data storage*".

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 before closing so that the settings remain in force upon next starting imc WAVE.

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".

Additionally, the column ID is included in the list appearing in the description of the respective parameters; e.g. "[Measurement mode](#)^[247]".

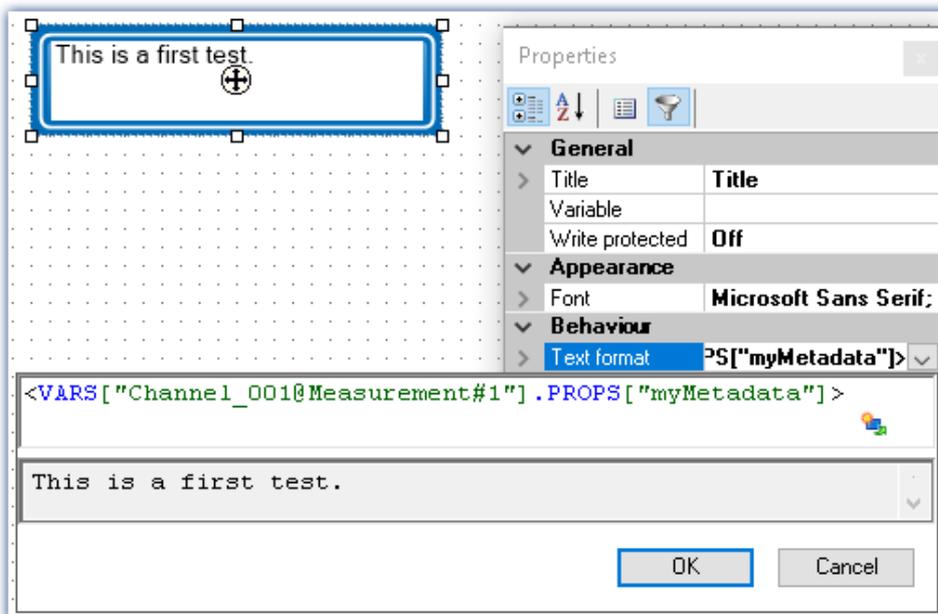
 **Example**

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](#)^[1149]". 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.

8.5.2 Comments on the measurement - Measurement metadata

By using the Comment-function, it is possible to save explanatory information along with the measurement. The metadata are saved in parallel with the saved measurement data in the same measurement folder. By default, a multi-line comment box is available for this purpose. Additional metadata (texts, numbers) can be added.

Display of the metadata in a dialog can be specified to occur either before or following the measurement. Optionally, the data can be made available in the form of variables. This makes it also possible to display the data on a Panel page, on which the values can also be edited.

The file's format is the user's choice between a text file and imc-format.

Ribbon	View
Start > Comment (📝)	all
Setup-Configuration > Comment (📝)	Complete

Measurement Meta Data

Comment

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Name	Value
SN	123456
Info1	
Info2	

Options OK Cancel

Example of entering metadata on a measurement

Options

In the Options, you specify which comments are offered as metadata and how they are edited and stored.

Option: "Metadata" > "Traceability of measurements"

Make parameters available for the Panel

Reset values before measurement

Storage file format within measurement folders: imc-Format

Delete

Name	Description	Type	Default value	Input at start	Input at stop
Kommentar		Text		<input type="checkbox"/>	<input checked="" type="checkbox"/>
SN		Number	123456	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Info1		Text		<input type="checkbox"/>	<input type="checkbox"/>
Info2		Text		<input type="checkbox"/>	<input type="checkbox"/>
*				<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Options for custom metadata on the measurement

Option	Description
Make parameters available for the Panel	A user-defined variable is created for each element of metadata (<i>Current Measurement</i>). The variable can be displayed on the Panel-pages. It is possible to change the values by means of this variable, so that the variable can be modified at any time during or before measurement. As well, the technique of importing variables can be used to define custom default values and to enter metadata presets.
Reset values before measurement	Automatically sets the value to the " <i>Default value</i> " upon starting measurement.
Storage file format within measurement folders	Specifies the target format for the metadata file. Either text-file (.txt) or in imc-format (.dat).

Parameter	Description
Name	Display name of the metadata. Used for the variable's name in the input dialog and in the results file.
Description	Optional descriptive text of the metadata.
Type	Data type of the metadata and of the target variables. Also affects the input possibilities. The available choices are Text and Number.
Default value	The default value applied to the first measurement, which can optionally be reset before every subsequent measurement.
Input at start	When the measurement is started (Start-button), a dialog appears which offers selected metadata for entry.
Input at stop	When the measurement for all devices is ended, a dialog appears which offers selected metadata for entry.

Call for entering metadata

The menu action "*Comment*" () can be run at any desired time, e.g. via the menu, or automated by means of the command: "[Execute menu action](#) ".

In this case, all existing metadata are displayed.

Conversely, if the dialog is called by means of the actions Start/Stop Measurement, only the metadata selected are offered for making entries ("*Input at start*" / "*Input at stop*").

Saving the metadata

The metadata are saved in parallel with the saved measurement data in the same measurement folder and in the selected file format. When the "*imc-Format*" is selected, the data can easily be read by imc FAMOS. Alternatively, the Text format can be selected, in which case the file can be read by common text editors.

At the end of the measurement, the file is saved. The last measurement folder contains the file. Any other measurement folders which had been generated by interval saving or by means of "*Suspend and Resume data storage*" do not receive any metadata file.

What is saved is the value which is available at that moment. Any changes to the values made after a concluded measurement apply to the next measurement, unless they are reset.

 FAQ

Question: Can metadata also be saved on the device?

Answer: Metadata saved **along with the measurement** are only saved on the PC. Metadata which are saved **in the channel file** are also saved in the files which are stored on the device.

 Note

Saving the configuration

The **configuration of the metadata** is saved with the respective **experiment**.

 Note

Customized storage location for measurement data

If you are using "*customized storage location for measurement data*" in combination with metadata, please read the notes in section "[Customized storage location for measurement data](#)" .

8.6 Information and tips

8.6.1 Enumeration classes

In an Enumeration class, the associations to "Object-groups" (e.g. analog channels, Display variables, devices, ...) are defined.

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 (MFBUDIO), 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)

8.6.2 Transferring an experiment to other devices

You can transfer a complete experiment to other devices. In the process, all possible settings are transferred along to the other devices. The transfer can be targeted for specific device modules; for this purpose, there is a software Assistant providing aid in making the desired assignment.

There are multiple ways to open the Assistant:

1. Transferring a complete experiment
2. Switching devices in an open experiment
3. Unable to set up connection to the device

Note

For devices of the [firmware group B](#) ¹⁴⁶

For transfer of the configuration from imc CANSAS*fit*- and imc ARGUS*fit*-modules which are attached to an imc ARGUS*fit* unit, the following limitation applies.

The configuration of a module can only be transferred when the following conditions are met:

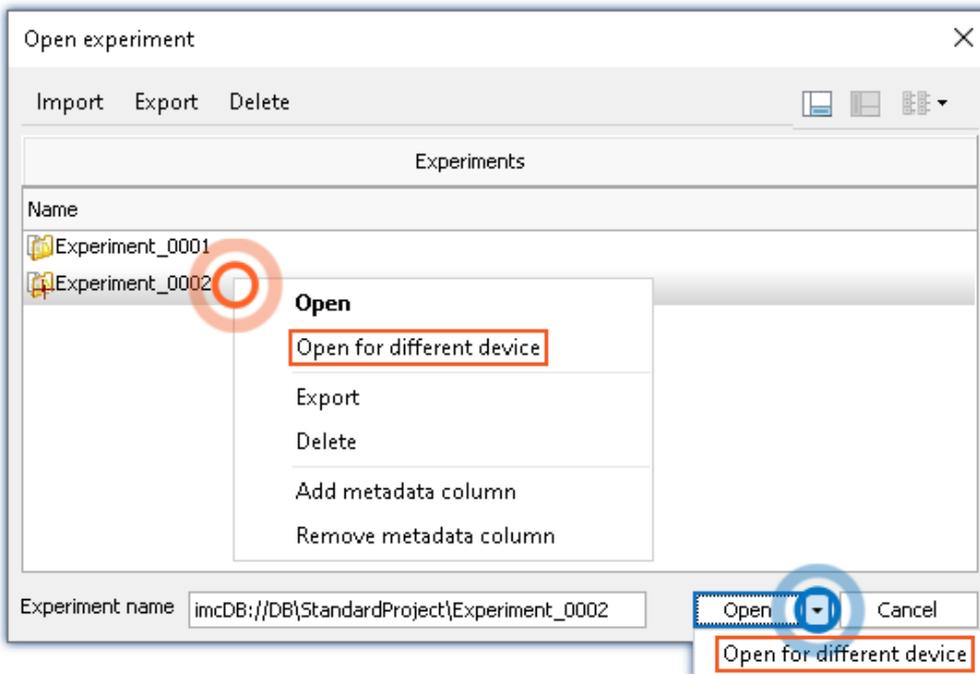
- same module-type (e.g. ARGFT\UTI-6-SUP to ARGFT\UTI-6-SUP)
- same device family, meaning imc ARGUS*fit* only to imc ARGUS*fit* and imc CANSAS*fit* only to imc CANSAS*fit*
- same scope of functions (e.g. same connection terminal)

The module's hardware revision stated on the plaque does not matter.

1. Transferring a complete experiment

Using the **open experiment dialog**:

- Open the experiment's context menu, or the drop-down list of the "Open"-button, and select "Open for different device".



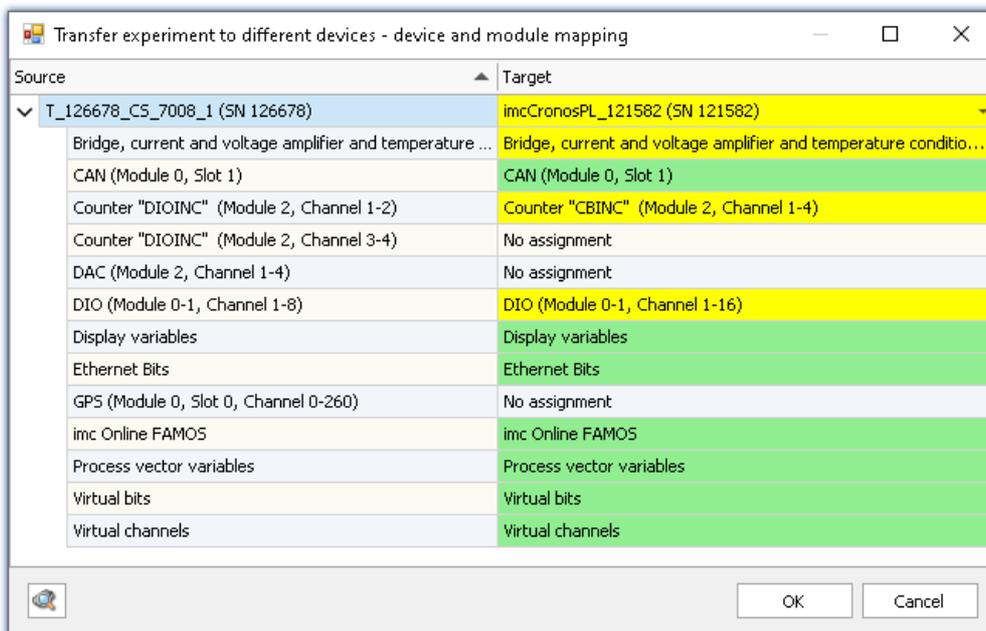
2. Switching devices in an open experiment

- Open an experiment
- Go to the Setup page: "Devices"
 - Select the device to be transferred (column: "Selected")
 - A dialog appears: Transfer the setting by means of the button: "Transfer"
 - In the next step, you can search for the device if it is not already part of the experiment. Otherwise, skip this step
- The Assistant opens

3. Unable to set up connection to the device

- If it is not possible to set up the connection to the device, it is possible to use a different device. The corresponding dialog for this purpose appears.

Configure device and module mapping



*Transferring to different device
Example: Different device types*

At left, there is a list of all devices belonging to the experiment (or only one device if it was de-selected). At right is a list of target devices; this side is initially empty and must be configured.

Below each device's entry is a list of all of its respective "modules" and components. What this list contains depends on the device's hardware configuration.

To set the target device desired, select a device from the drop-down list which is in the right-hand column on the row displaying the source device. As soon as a device is selected, the device is analyzed. The Assistant generates a suggested allocation (see example above).

When the devices are of the same model and hardware configuration, it is possible to use the default setting. If the experiment is to be transferred to a different kind of device, verify the suggested settings.

The colors:

Green: All of the module's settings can be transferred.

Yellow: Not all of the module's settings can be transferred. For example, the amplifier type may be different.

Please ensure that the target module is able to use the configuration required by the source module. E.g. a universal amplifier can use some of a temperature amplifier's settings.

Red: This assignment should not be selected.

What settings are applied:

In principle, all settings are applied, except for the **calibration values** (taring, bridge balancing, two-point scaling, ...). Settings which can not be applied are ignored and listed in the **log book** following the transfer.

As well, the **channel names** are adopted.

If multiple devices are used in an experiment, the channels are automatically assigned names with the device name as a prefix. If some of these channels were not renamed, the name remains intact after the transfer.

The device name within the channel name then no longer reflects the name of the new device.

Automated analysis

When a target device is selected, the analysis starts. The result is the suggested assignment. This can be initiated for all devices automatically, as soon as the dialog is opened. To do this, click on this button: 

When the button is clicked, all devices registered with the system are analyzed. Once this step has been concluded, it is possible to select a device.

Advantage: In the drop-down-list, it is immediately visible which devices are suitable for the transfer.

Disadvantage: **All known** devices are analyzed, requiring a corresponding amount of time, according to how many there are.



Reference

Replacing modules

A similar dialog is displayed when one of a device's modules is replaced. See "[Change of the imc measurement device](#)".

8.6.3 Report of the Setup-configuration

You can **create a report** of the **current Setup-configuration**. The report is opened in the Browser and can be **saved or printed** there.

A list of many device and channel configurations, as well as of the imc Online FAMOS-source text is shown.

Devices	
Device identifier	imcDev_12126678
Device SN	126678
Device name	T_126678_CS_7008_1
Device specification	imc C Series
Device adapter	Ethernet
Device total sample rate	454000 Hz
Synchronous start	<input type="checkbox"/>
Device features	Online DSP, imc Online FAMOS, with amplifier, PCMCIA (removeable disk)
Device storage location	Removable disk (PCMCIA)
Device storage interval	End of measurement
Device storage interval count	0
Device storage path naming	Timestamp - date time (measurement number)

Example of a report

The first columns of the device configuration. The channel configuration appears further below.

8.7 Troubleshooting

Note

Only qualified technicians¹ are allowed to make repairs on the device! Unauthorized opening or incorrect repair of the device may greatly endanger the user (electric shock, fire hazard). Devices which have been altered or tampered will no longer comply with their license and may not be used. In case of accident (e.g. damage to housing, terminals, modules or power supply, or exposure to liquids or foreign substances), turn the device off immediately, unplug the power cord and inform our [tech support](#) .

1 Authorized / qualified personnel refers to persons familiar with the setup, installation, commissioning, and operation of the product and who hold certification for their respective skills.

- [The device cannot be switched on](#) 
- [Error message when starting measurement](#) 
- [Device not found or measurement data is not displayed](#) 
- [Hardware defects](#) 
- [Error description](#) 

The Device cannot be Switched on

- Check the power supply and the fuses (below the power switch).
- After switching off the unit, wait at least 10 seconds before switching it back on.
- Switch the system off and disconnect all lines except the power supply. Then try again to switch the unit on.

Error Message when Starting Measurement

If error messages appear when starting the device, please check the following first:

- Check the connection between the unit and your PC (see "[Setting Up - Connect the device](#)"³⁹).
- Next, check that the correct connection type has been specified. (see device model plaque).
- Turn the device off and then on again. Select Ribbon "Home" > "Disconnect" followed by "Connect".

The latter command reads the hardware configuration into the software.

Device not found or Measurement Data is not displayed

Error Message 103 - Check the device <=> PC connection and the power supply.

With Ethernet-interface: If your device is not located after using *Device search*, see the notes in "[Setting Up - Connect the device](#)"³⁹.

Hardware Defects

If measurements return implausible results, follow these instructions to determine whether the hardware is defective:

1. Switch off device and disconnect all external power and signal input lines from the device.
2. Reconnect the power supply and switch device on: Does it boot correctly?
3. Connect the device to the PC (in accordance with the interface type)
4. Start the imc WAVE software
5. Select the device
6. Select Ribbon *Home* (or *Setup-Control*) > *Connect*
7. Select Ribbon *Home* (or *Setup-Control*) > *Prepare*
8. Start measurement (measurement duration 10 s)
9. Select channel and display as curve
10. Restart measurement (measurement duration 10 s)

Error Description

To help locate defects when sending a unit back for repairs, please enclose a description of the error(s) encountered and, if applicable, a graph of expected versus incorrectly returned signal courses, as well as the following information for our [tech support](#)⁷:

- Device serial number
- Software version used, export of the version information (About)
- Windows operating system version
- Experiment file from the database: <Experiment>.imcStudio

9 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.

Page	Description
 Documentation ¹⁸⁴	On this page, you can supply the experiment with descriptions and metadata .
 Devices ¹⁸⁵	Here, select the device for the measurement. Configure the data storage ¹⁸⁷ , synchronization ¹⁸⁸ and the timed start ²³³ , among other settings.
 Analog channels ²³⁷	Here, configure the channel-specific parameters. Among others, those for the analog inputs and virtual channels .
 Digital channels ²³⁷	Here, configure the channel-specific parameters. Among others, those for the digital inputs/outputs and incremental counter channels .
 GPS ²³⁷	Here, configure the channel-specific parameters relating to the GPS-channels .
 Variables ²³⁷	Here, configure the channel-specific parameters relating to the Display variables and process vector variables .
 Channel balance ²⁷⁵	On this page, you can have various adjustment and calibration types performed.
 Triggers ²⁸²	Here, configure the Trigger-Machine in order to perform targeted starting and stopping of channels.
 TEDS ⁴⁶⁵	Here, import TEDS information in order to obtain precise configuration of the channels.
 Storage options and directory structure ³⁰⁶	Saving measured data on the PC and in the device
Storage media in the device ³³⁷	This section describes how to handle the storage media of the imc measurement devices and how to use them with imc WAVE.
Fieldbusses ³⁶⁸	Operation and configuration of the various Fieldbus-systems

9.1 Documentation (experiment description)

On the page "*Homepage*", you can supply the experiment with descriptions and metadata.

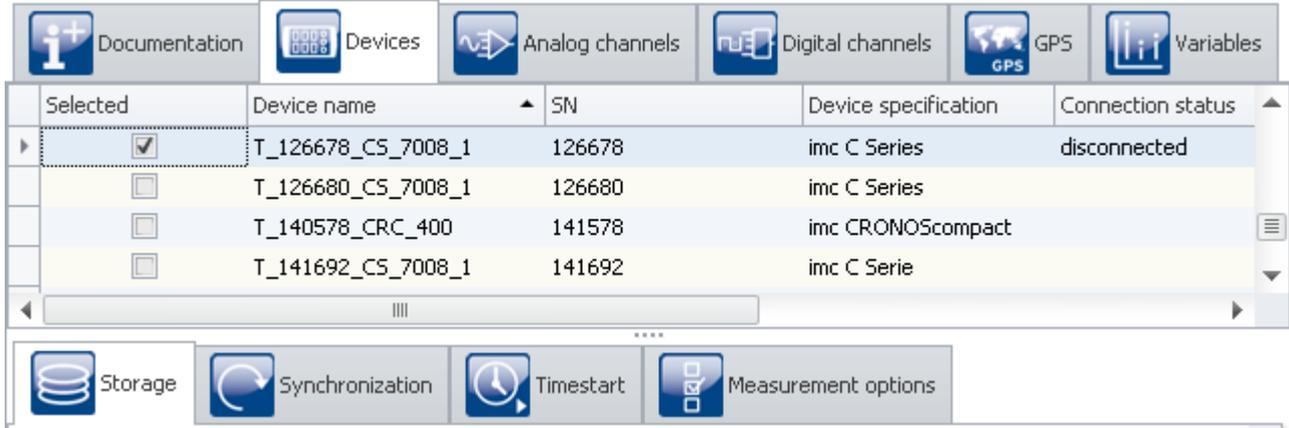
The page is subdivided into two areas: the **Table** below and a **Description**.

In the table, you can enter information pertinent to the experiment.

In the box **Description**, you can provide your experiment with up to multiple lines of explicit description.

9.2 Configuring devices

On the page **Devices**, you can set all device-specific parameters.



"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:

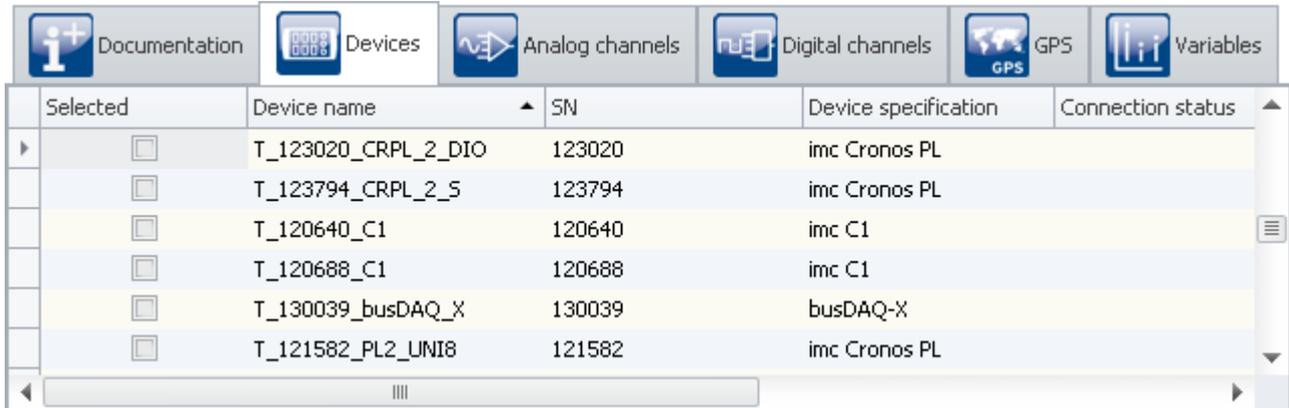
Page	Description
 Storage ¹⁸⁷	Settings for saving measured data in the device and on the PC.
 Synchronization ¹⁸⁸	Settings for synchronization to a clock or between multiple measurement devices
 Timed start ²³³	This enables you to start the measurement at a defined point in time.
 Measurement options ²³⁶	Various additional settings.

 **Reference** See also additional info on:

- Operation and configuration of the [tabular display](#) ¹⁶¹ and the [dialogs](#) ¹⁶⁸

9.2.1 Device table

In this area, the [known](#) ¹⁷¹ devices are displayed in a structured table, as seen in the example image below.



Device-Table (example)

To be able to **use a device** for the current experiment, you must first [find and select](#) ¹⁷¹ it. To **adapt the devices' settings**, select one or more devices. After that, you either [edit the table cells](#) ¹⁶⁴ directly or select the respective [dialog](#) ¹⁶⁸ for the desired parameters.

Parameter	Description		
	<i>long name</i>	<i>short name</i>	<i>column ID</i>

Selected	<i>Device is selected for measurement</i>	<i>Selected</i>	<i>Dev_SelForMeasurement</i>
----------	---	-----------------	------------------------------

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](#)" ¹⁷¹

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 "Connection" with the device. After having made the change, perform the "Prepare", 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>
-------------------	--------------------------	-------------------	--------------------------

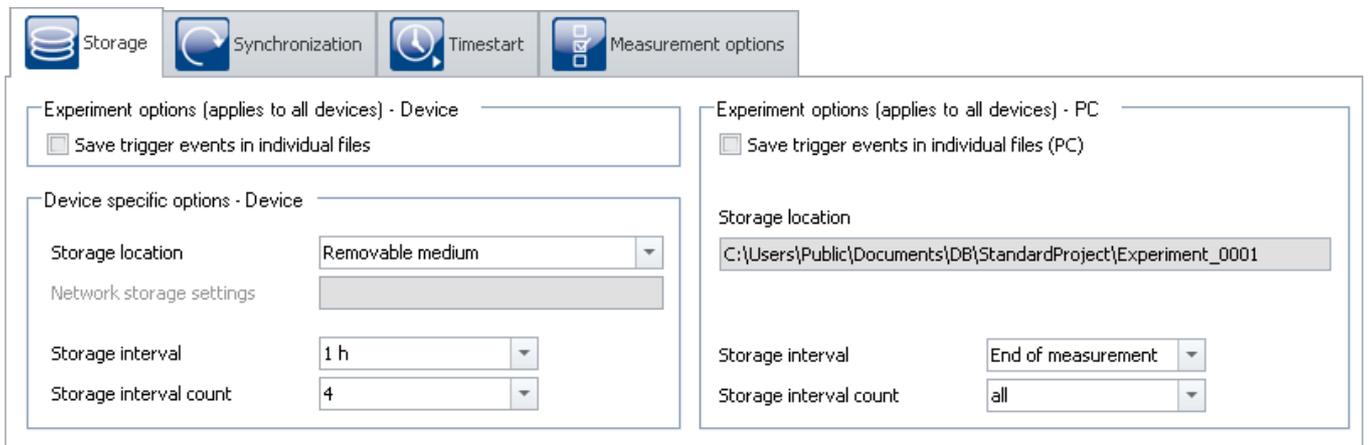
Indicates the current connection status: "connected" or "disconnected".

Parameter	Description		
	long name	short name	column ID
Measurement status	Measurement status	Measurement	eMeasurementStatus
Indicates the current status of the measurement: "stopped", "running" or "reconfigured".			
Device control	Device control	Device control	eDeviceControlAction
<p>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.</p> <p>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.</p> <p>Disconnection and re-connection with the running measurement are not possible using this button. For that purpose, use the menu item "Connect".</p>			

9.2.2 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](#)".



"Storage" dialog

This dialog sets 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 " ^[329]  In order to display the measured data associated with the events, add the column " Event time " ^[805] manually in the Data Browser.
Storage interval / Storage interval count	 See " Saving interval " ^[318]  When interval storage is active, all folders are deleted except for the specified number of intervals!

9.2.3 Synchronization

This chapter describes the different options for synchronizing imc measurement devices.

This amounts to a **unified time base for different devices** whose measured data are compared. Synchronized devices can return **measured data which are exactly matched to each other in time**.

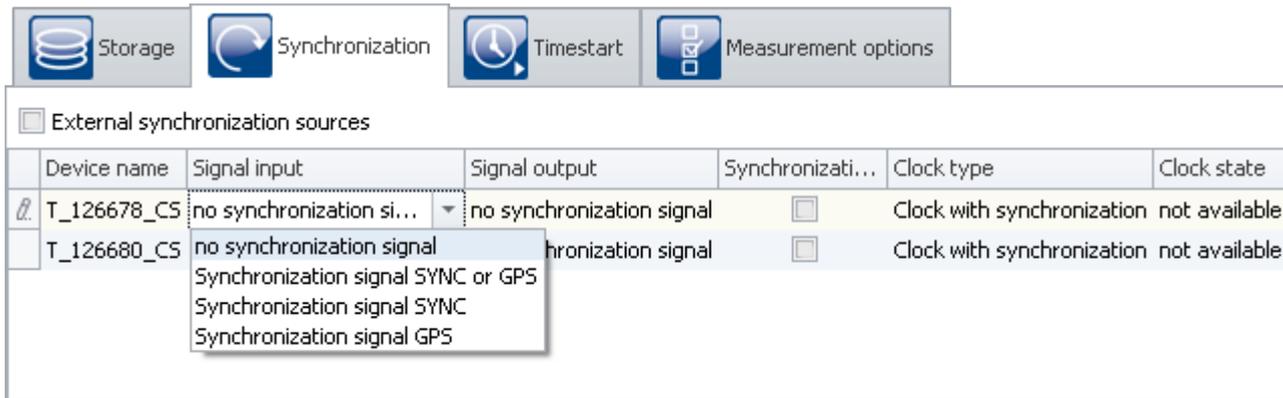
 Reference
Differences at runtime

Differences in the processing speeds between amplifiers, fieldbuses, etc., may also cause synchronization discrepancies. This topic is not covered here. Wherever it can, the system compensates for each of these devices' processing speed differentials.

Please find differences of the processing speed between measurement amplifiers in the corresponding device manual, chapter: "... > *Synchronicity* > *Delay*".

Setting synchronization

All **synchronization settings** are to be made in the dialog: "*Synchronization*" (on the Setup page "*Devices*").



In this dialog, you are able to **synchronize multiple devices with each other** or set a device according to an **external clock**. Synchronized devices can return **measured data which are exactly matched to each other in time**.

In order for the devices to be synchronized with each other, they must be **connected with the synchronization signal generator**. This can be either an external master or an imc device. Some models (e.g. DCF/IRIG-B) are connected with each other via the **SYNC terminal**. Others are synchronized via the **network** (e.g. NTP/PTP) or via **GPS**.

Parameter	Description
External synchronization sources	<p>The devices are synchronized with an external synchronization source (e.g. DCF77, IRIG-B, ...). They have no "signal output" of their own and thus can not be the "synchronization master" in consequence.</p> <p>The checkmark's only effect is that no other "Masters" can be activated.</p>
Synchronization master	<p>If no external synchronization source is used and only one device is to be used as the synchronization signal source, put a checkmark here. This setting assistance is only relevant for IRIG-B and DCF77 and must not be activated for NTP and PTP synchronisation.</p> <p>The checkmark's only effect is that no other "Masters" can be activated.</p>
Output	<p>Here, select the synchronization signal according to which the other devices are synchronized. There are different output signals depending on the particular device.</p> <ul style="list-style-type: none"> • E.g. DCF or IRIG-B. Devices which are synchronized with this device must be set for "Signal input" to either "Synchronization signal SYNC" or "Synchronization signal SYNC or GPS"^[205]. • PTP serves to synchronize clocks within distributed computer systems. It forms a self-organizing hierarchical structure. In accordance with the specified settings, an imc device only becomes the PTP-master if there is no "better" PTP-master available. See the chapter: "External timer: PTP"^[210]. • PTP-Master only^[217]: If the network includes no PTP-server true to absolute time, you can synchronize one imc device to a hardware synchronization signal (e.g. GPS). When you define that device as the PTP-master, it will no longer be synchronized by any other PTP-subscriber.
Input	<p>Here, select the incoming synchronization signal, which depends on the synchronization signal source. For DFC and IRIG-B, please select "SYNC".</p>



Note

Simultaneously starting all devices

Channels of synchronized devices have correct absolute time. **However, the devices still start serially.** To prevent this, select

- "Synchronous start" under "[Timed start](#)"^[233]
- an appropriate "[Start options](#)"^[234]. E.g. "Automatic timed start".

See: "[Simultaneous start of all devices](#)"^[235]

Guidepost

Summary	Section
What is synchronization needed for? What advantages does it have?	What's the purpose of synchronization - and how does it work? ¹⁹⁰
Brief explanation of terms. For instance: synchronization, timed start, time zone, simultaneous starting of all devices, Synchronous start (synchronized, but not simultaneous starting)	Definitions ¹⁹²
Functioning and exact description of the various clock types. Internal device clocks, external time emitter, and the PC-clock.	Clock types ¹⁹³
Various applications of synchronization. How are which clocks used? How do the devices need to be interconnected? What settings are to be made for imc WAVE? Is there anything to pay attention to?	Synchronization varieties ¹⁹⁹
What issues are involved when the synchronization signal is interrupted? What time discrepancies will occur? How to re-synchronize the signal?	Phase offset correction ²²⁰
What special issues are there in conjunction with time zones?	Working with time zones ²²⁰
If all devices are connected with an external Master: What happens if, for instance, the signal is lost? ...	What happens when? ²²⁵

9.2.3.1 What's the purpose of synchronization - and how does it work?

If you are only using one device, you may wonder what synchronization is good for. After all, your data already are associated with a correct time. For most users, it's sufficient to be able to compare measured signals with each other as long as they all were captured within the same device.

Synchronization is necessary if:

- **multiple devices** are measuring the same object simultaneously
- the **absolute time** is relevant

If **data** are captured **by different devices**, it should be possible to **track their chronological relationships** just as well as those of data originating in one device. Absolute time usually doesn't play any major role in this context.

Absolute time becomes important when **signals originating very far apart** are captured by separate devices. When such data are associated with an absolute time-stamp, they can be analyzed in conjunction with each other.

The synchronization process

After "Preparation", the devices apply whatever synchronization settings have been made.

Only **after the first preparation procedure**, the devices **synchronize** to the respective sources. The **device clock is adjusted** (phase), the **frequency is put under steady control** (frequency synchronized). When adopting the time, the **time zone** set in the device **is taken into account** (e.g. conversion of the GPS-time (UTC) to the time zone set).

Each synchronization type has special distinctions. With some, for instance, the PC clock time is adopted by the devices, with others not (for example, when an external Master is used). More information is presented in the chapter: "[VRTC - Virtual clock on the PC](#)" ¹⁹⁵.

When **starting a measurement**, its **start time** is either **determined** automatically (simple start, automatic timed start), or is specified explicitly (timed start). As soon as the start time has been determined, the device clock may not be set again (on this topic, see also the note further below, "*Time deviations*"). This also determines whether the measurement time is expressed in standard or daylight saving time.

Synchronization signal interruption (only for devices of the [groups A5-A7](#)^[146] and devices belonging to the [firmware group B](#)^[146]):

When the synchronization signal is available again after an interruption, the system determines how far the device's time deviates from the reference time. This phase error is corrected by adjusting the device clock rate until the device is once again absolutely synchronized to the reference time (on this topic, see: "[Phase error correction](#)"^[220]).



Note

Time deviations

If you click on "Start Measurement" () before the synchronization has been completed, it is no longer possible to correct any deviation by the device clock from the reference clock.

If **no time deviations** may exist, **then wait to start the measurement** until all devices are indicated as "Synchronized". Or force a [simultaneous starting of all devices](#)^[235] if you are using more than one device.

How does the deviation happen?

The starting time is set as soon as you click on "Start measurement" () . This applies to every style of start, such as "*immediately*" or starting at a "*defined time*".

If the start time has been specified, the device clock will no longer be set (by a time adjustment to adopt the absolute time). In spite of this, the device will report that it is (frequency-) synchronized! (Clock status and imc Online FAMOS "[IsSynchronized](#)")

Thus, the frequency (clock rate) is synchronized, but there is an offset from the reference time.

Upon the next preparation procedure, the system tries again to adopt the absolute time given by the external reference clock. Toward this end, the synchronization is temporarily abandoned and then re-established. Starting of the next measurement should only begin after the re-synchronization has been performed. Otherwise, this process will be repeated upon each preparation procedure!

Determining the deviation from the reference clock in conjunction with NTP- and PTP-synchronization

The device clock's remaining deviation from the reference clock is represented by the variable "*pv.State.ResidualSyncTimeDeviation*". This value is determined when "Start Measurement" is pressed. At that moment the current deviation is determined.

- "0.0" : "Initial value", "no deviation" or "Unable to determine value" (see below).
- ">0" : Deviation in seconds

No statement for any other synchronization types

With the other synchronization types (such as GPS, DCF, ...), the value cannot be determined and is always set to "0.0"!

**Note****Cases where the device clock is not adjusted**

When the PC prepares a measurement, it also sets the device's time.

The device time is **not** set by the PC if:

- a valid synchronization signal is already applied (in particular when the device is synchronized), or
- the device's synchronization output (*SYNC*) is already running (otherwise, all devices must restart the synchronization), or
- "NTP" or "PTP" is selected as the synchronization signal

9.2.3.2 Definitions**Synchronization:**

Devices are synchronized if their time bases are rigidly linked. This is achieved in either of two ways:

- *Use of an external time signal:* If the time signal of an exact time standard (absolute time encoder: e.g. DCF, GPS, NTP, ...) is available, it can be used to synchronize the devices. The time bases of the devices are then not only synchronized to each other but also to the time standard used.
- *Use of a device's own time base:* The time signal of one of the devices (master) is used to synchronize other devices (slaves). The time bases of the devices are only synchronized with each other, but not with absolute time.

Mixed-operation is possible, such that one device is synchronized with an external source and all other devices synchronize to the (Master) device.

Timed start:

The start is triggered by a particular clock at a pre-defined point in time. Each device has its own clock. If multiple devices are to start simultaneously, the precision of the start time depends on the precision of the synchronization.

Synchronous start (synchronized, but not simultaneous starting):

This operating mode only works with multiple devices, and each device's own clock triggers the start. For this purpose, all clocks must be synchronized. The master clock is one of the following:

- an externally synchronizable clock
- a device's clock module, which has been declared as the Master, or
- each device uses its own time source,
- ...

Simultaneous start of all devices:

When Synchronization is activated, simultaneous start of all devices is possible. For this purpose, certain conditions must be met. Among others, an appropriate "[Start options](#)^[234]" must be selected. E.g. "Automatic timed start".

With the option "immediately", the devices measure synchronously, and are started quickly, but serially. The measured data are then only synchronized with each other in the absolute time display.

Additionally, the option: "[Synchronous start](#)^[233]" must be activated.

See: "[Simultaneous start of all devices](#)^[235]"

Time zone and switch to/from daylight saving time:

The device software is able to take the switch to/from daylight saving time or between time zones into account. The setting is made in the [device properties](#)^[155]: "Device time, synchronization" > "Time zone" and "Automatic daylight saving time switching".

Action	Page
Dialog: Device options > Device Properties	Devices

Therms	Description
STD	STandarD time
DST	Daylight Saving Time
DST-state	DST or STD
DST-switch	switch from DST to STD or vice versa
UTC	Coordinated Universal Time (Greenwich meridian); no DST-switching!
localtime	the time prevalent in the respective time zone with regard to the current DST-state
RTC	RealTimeClock; internal device clock
SyncRTC ^[194]	Synchronizable RealTimeClock of imc devices
VRTC ^[195]	Virtual clock on the PC

9.2.3.3 Clock types

Internal clocks

Clock type	imc measurement device
SyncRTC ^[194]	For all device types: Synchronizable RealTimeClock of imc devices
VRTC ^[195]	Virtual clock on the PC

External clocks

Clock type	imc measurement device
DCF77 ^[200]	For devices belonging to the firmware group A ^[146]
IRIG-B ^[200]	For devices of the groups A5-A7 ^[146] and devices belonging to the firmware group B ^[146]
GPS ^[204]	For devices belonging to the firmware group A ^[146] and devices of the group B11 ^[146]
NTP ^[206]	For devices of the groups A5-A7 ^[146] and devices belonging to the firmware group B ^[146]
PTP ^[210]	For devices of the group A7 ^[146] with the suffix "-GP", as well as CRXT

9.2.3.3.1 Internal clocks: Device Clock - SyncRTC

The SyncRTC is the device's internal clock. SyncRTC: **S**ynchronizable **R**eal **T**ime **C**lock.

It is used to determine the start time of the first measurement after initialization, for example. Once this start time has been determined, the SyncRTC's absolute time plays no further role until the next measurement's initialization. Any time readings are derived by adding the sampling times to the start time, and the sampling times are derived from the SyncRTC's frequency.

If for instance the clock jumps (time change for daylight savings, etc.) this does not affect the measurement.

There are a variety of ways to synchronize the SyncRTC with external time sources. By doing this, it is ensured to both use the correct absolute time and the correct frequency.

Most devices* come with an external "SYNC" terminal and a GPS-input for synchronization with other clocks. The SYNC terminal can be configured as your choice of either an input or output. You can select either DCF77 or IRIG-B 002 (if supported by the device) as the signal format.

As the input signals, you can use other imc device signals or external time sources (e.g. GPS-clocks from Meinberg, Hopf, etc.).

Depending on the device type, additional synchronization mechanisms are also available. See: [Clock types: External clocks](#)  193.

*Note: Certain terminals are not included with some device and housing types!

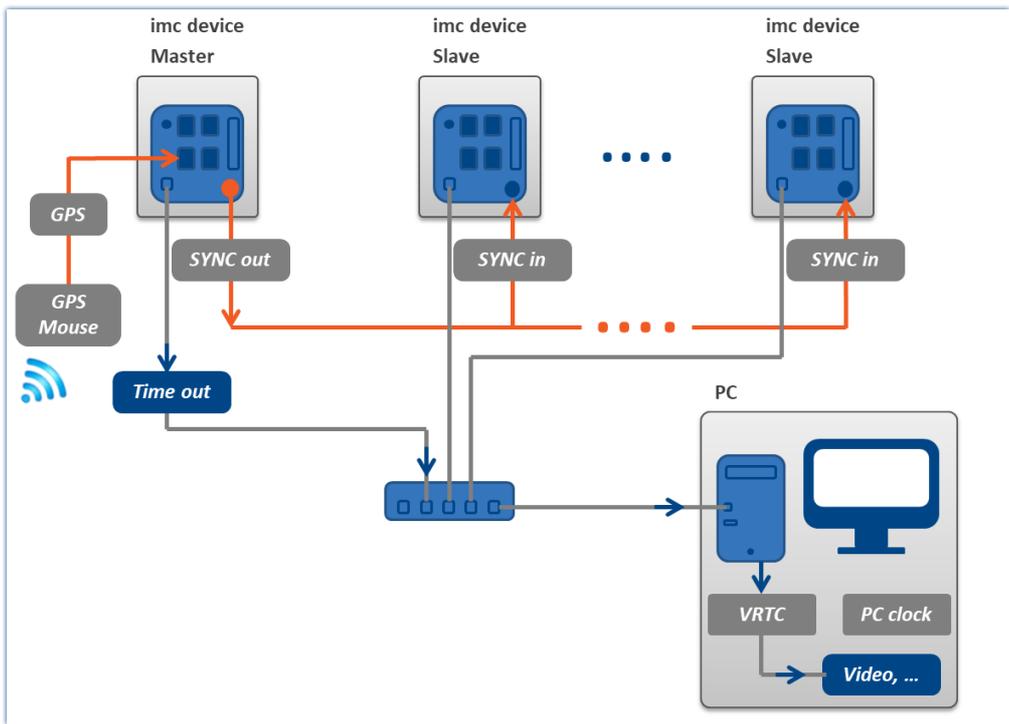
9.2.3.3.2 VRTC - Virtual clock on the PC

For the purposes of precise time output on the PC and synchronization of device results with PC results, a **virtual clock** (VRTC) is used.

The subscribers on the PC always use the virtual clock's time. Such a subscriber can be, for example, the clock on a Panel page or a Video-signal.

Outline: How does the virtual device clock work?

During the running measurement, the virtual clock synchronizes itself to the Master imc device's clock. This is needed if functions are used on the PC side which require an exact time track (e.g. Video, report channels and third-party devices such as Fos4X).



Example of synchronizing the VRTC via the Master device

On heavily burdened networks, good synchronization (of the PC's virtual device clock with the device) is not always possible. When the network load is too high, imc STUDIO posts a corresponding error message when an attempt is made to start the measurement.

It may help to increase the minimum accuracy.

Minimum accuracy:

The option defines the maximum time in milliseconds that the PC's virtual clock may deviate from the device clock. This concerns mainly non-imc devices, e.g. video. In case of video a value of 10 ms provides synchronization of measurement data to video images at 100 frames per second. Too low a value prevents measurement start.

The "Minimum accuracy" is located among the options under: "Extra" ribbon > "Options" in the region: "Setup" > "Device options" > [Virtual device clock](#)¹²⁴⁰.

**Question: Which clock time is used when preparing the devices?**

Answer: This depends on what synchronization is set for the devices:

- Without an external synchronization master, the device clocks are **set to the VRTC's time** upon preparing (all device clocks, or in a synchronization, those of the master devices).
- If you have an external synchronization master, then when performing preparation, synchronize the device clocks with the synchronization master. The virtual clock is set **according to the master device's clock**.

Subsequently, the VRTC aligns itself to the device clock.

To find more tips and notes on variations, see the section "*Details on workings of VRTC*". In particular for synchronizing NTP- and PTP, or on the topic of the behavior during measurement.

Question: Is there a distinction between the PC clock and the VRTC on the PC?

Answer: Yes. The VRTC runs on the PC and under some circumstances it can assume the PC's clock time (see previous question). However, it has its own high-resolution counter which enables it to provide significantly more precise time readings than the PC time. The largest deviation occurs in the case of external synchronization. In that case, the VRTC adopts the device clock time upon preparation.

Resolution and deviation before and after start of measurement

During the measurement, the VRTC aligns itself to the device clock. The clock rate of the devices has a resolution of 1 ms or better.

The deviations differ among the devices **up until the moment of measurement start:**

- With **devices of [groups A5-A7 and firmware group B](#)** ¹⁴⁶, at the moment the measurement starts the VRTC **typically has a 1 ms deviation**.
- With **devices of [group A4](#)** ¹⁴⁶ **and higher**, at the moment the measurement starts the VRTC **typically has a 15 ms deviation**.

Over the course of multiple seconds after starting measurement, the deviation improves. This should be noted about all VRTC subscribers, e.g. Video, report channels and third-party devices such as Fos4X.

Whenever possible, define a device of the [groups A5-A7 or firmware group B](#) ¹⁴⁶ as the synchronization master. Especially when you are using multiple different devices within a measurement setup.

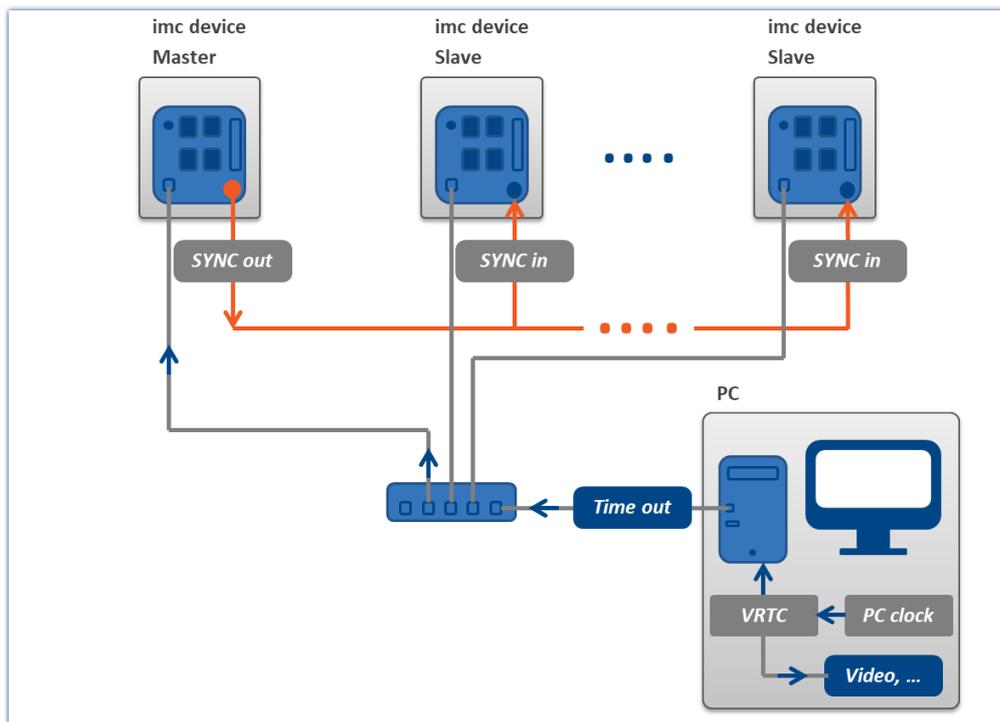
Details on workings of VRTC

When imc STUDIO starts, the VRTC adopts the PC's clock time.

If no measurement is running (or no other components require the VRTC) the system behaves as follows: The time between the PC and VRTC is calibrated every 10 s. Since there are no subscribers, the time adjustment simply jumps abruptly (in either positive or negative direction).

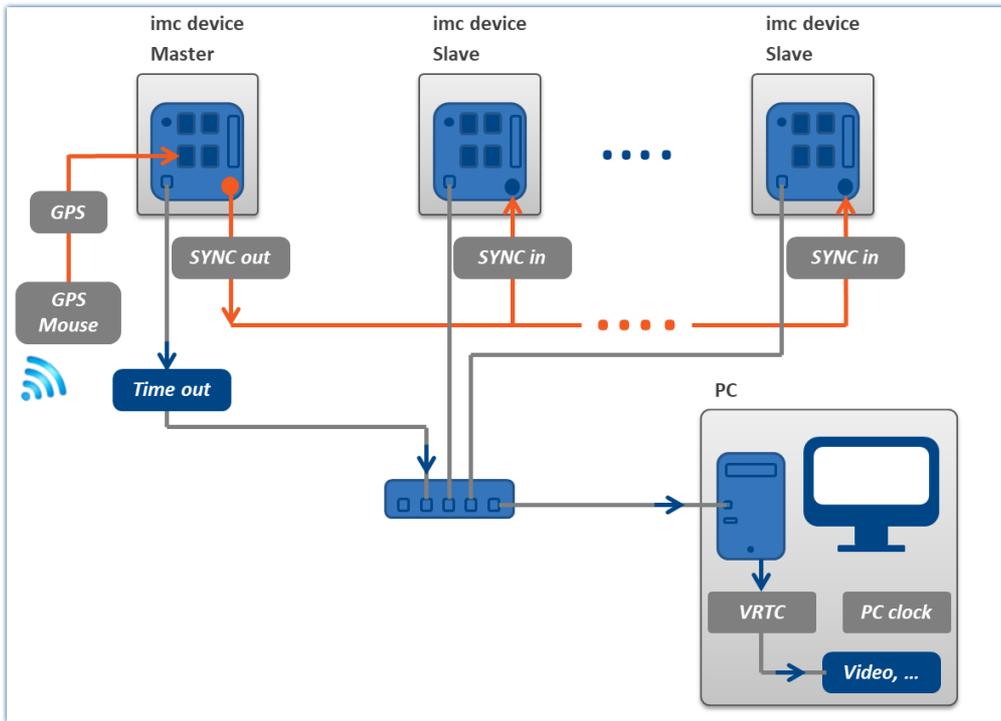
Behavior when preparing (all devices) - Setting the device clock:

- When there is no external synchronization master, then upon preparation, the device clocks are synched to the VRTC clock one single time (all devices' clocks, or in case of synchronization, those of the Master devices).



One-time-only synchronization during preparation, without any external Master

- If you have an external synchronization master, then when performing preparation, the device clocks synchronize with the synchronization master. The virtual clock is set **according to the master device's clock** a single time (see below the exception regarding NTP- and PTP-synchronization).



One-time-only synchronization during preparation, with an external Master

Behavior during measurement:

- The imc devices synchronize in accordance with their synchronization settings. These operate independently of the VRTC.
- Behavior of the VRTC
 - **No synchronization** of the devices mutually **or synchronization via SYNC** (DCF/IRIG-B):
The VRTC aligns itself to the Master device's clock (if there is no Master device, then the first one in the list).
The time alignment always increases monotonically. No abrupt skips are made.
 - **NTP, PTP:**
There is no calibration of the VRTC to the device clock! Please synchronize the PC to the same source.
The VRTC aligns itself to the PC's time.

Note The precision level depends on the operating system

The precision of the PC's internal clock rate plays a large role here. Newer operating systems have higher precision. With Win 10, one can assume a precision level of up to 100 ns.

Note Recommended methods

- Whenever possible, use defined Master devices.
- Whenever possible, define a device of the [groups A5-A7 or firmware group B₁₄₆](#) as the Master. Especially when you are using multiple different devices within a measurement setup.
- When using NTP oder PTP, always synchronize the PC also with the source.

9.2.3.4 Synchronization varieties

In the following examples, two types of synchronization are presented.

Variety	Description
Master/Slave setup	All devices are synchronized mutually via "SYNC". The clock rate is provided by a Master device (which can be synchronized to an external clock, for example). All devices are connected via the SYNC jack. No terminal resistor is needed.
Separate synchronization of all devices	Individual synchronization of each device. If connection between devices via a SYNC line is not possible, then each device can be synchronized separately to an external clock.



Note

Outage of the synchronization source

- If there is an **outage of the Master**, all devices continue to work according to their own internal clocks and drift apart in accordance with the precision stated in the technical data of the device.
- If there is an **outage of the external clock** (e.g. GPS or NTP/PTP), the Master continues according to its internal clock. However, the devices remain synchronized to each other.

9.2.3.4.1 IRIG-B or DCF77

Introduction: IRIG-B

The IRIG-B signal is applied via the SYNC socket. The measurement device processes a LOW* active 5V TTL signal and supports IRIG-B002, B000, B001, B003. As well, inverted signals are recognized and evaluated by the hardware.

For all models, the pulse rate is 100 pulses per second, DC Level Shift (DCLS), width coded, no carrier. The last digit indicates the correct information:

B000	BCD, CF, SBS	BCD - Binary Coded Decimal, Code of the time (HH,MM,SS,DDD)
B001	BCD, CF	SBS - Straight Binary Second of day, daily second (0...86400)
B002	BCD	CF - Control Functions, depending on application
B003	BCD, SBS	

The models listed here can be connected for the input signal. However, the measurement device only applies the BCD information!

Duration of synchronization: With a typical synchronization time of approx. 20 s, the IRIG-B is the fastest model available.

Note

- For the synchronization signal input, only "*Synchronization signal SYNC*" may be set.
- The time information is expected to correspond to the device's local time!
- **During signal output** (IRIG-B002) only the **BCD information** will be applied, this means no CF and no SBS information will be applied.

*: Due to the greater steepness of edges, LOW active signals achieve better precision than HIGH active signals.

Introduction: DCF77

The DCF77 signal is applied via the SYNC socket. The measurement device processes a LOW* active 5V TTL signal (1Hz pulse). As well, inverted signals are recognized and evaluated by the hardware.

Duration of synchronization: The complete time is encoded within one minute and begins at the start of the next minute. For this reason the minimum duration for the synchronization process is at least one minute!

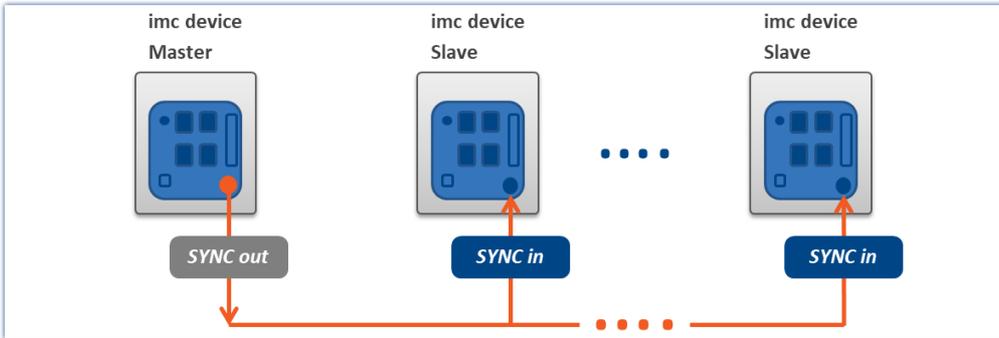
Note

For the synchronization signal input, only "*Synchronization signal SYNC*" may be set.

*: Due to the greater steepness of edges, LOW active signals achieve better precision than HIGH active signals.

Master/Slave setup

The devices are synchronized with each other, but not synchronized to absolute time as provided by an external clock.



Example for synchronization with master and slave devices

Parameter	Selection
Signal input	Master: Synchronization signal SYNC Slave: Synchronization signal SYNC
	 With DCF77 and IRIG-B, select "SYNC" as the signal input. The hardware automatically recognizes which type is involved.
Signal output	Master: Synchronization signal DCF or IRIG-B Slave: No synchronization signal
External synchronization source	No

Storage
 Synchronization
 Timestart
 Measurement options

External synchronization sources

Device name	Signal input	Signal output	Synchronization master
T_126678_CS_7008_1	Synchronization signal SYNC	Synchronization signal DCF	<input checked="" type="checkbox"/>
T_126680_CS_7008_1	Synchronization signal SYNC	no synchronization signal	<input type="checkbox"/>
T_160144_CRFX_2000	Synchronization signal SYNC	no synchronization signal	<input type="checkbox"/>

Settings in the plug-in Setup on the page "Devices" in the dialog "Synchronization"

Note

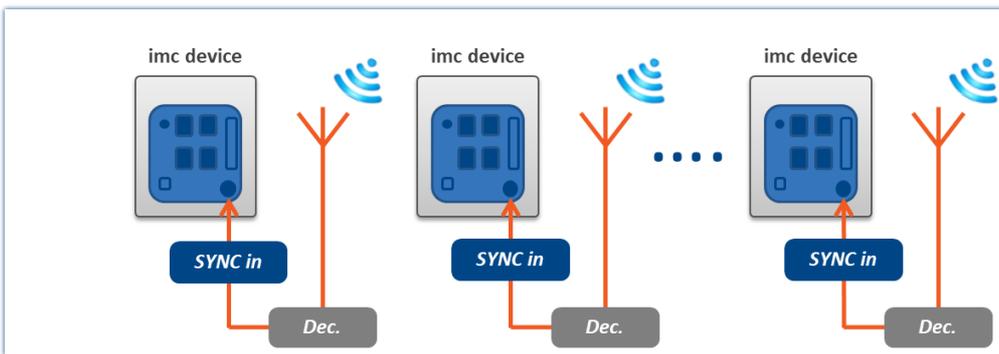
- The Master also synchronizes itself to its own signal, so that it also reports that it is synchronized. For this reason, its synchronization input is set to "Synchronization signal SYNC".
- A slave clock receiving an external clock signal is never set by PC.
- The master clock is set to PC time during preparation (see "[VRTC - Virtual clock on the PC](#)"¹⁹⁵). This also happens, for instance, after activating the devices ¹. The positive result of this is that when working with the devices, it is necessary to wait for the synchronization only once.
- In case the master device is equipped with a GPS receiver that should be ignored, the *Signal input* must be set to "Synchronization signal SYNC". Otherwise, the device might synchronize to the GPS-signal.

1: This doesn't apply if the the default synchronization setting is defined in the device properties. In that case, the time is only adjusted if the signal output is subsequently set to "No synchronization signal" on the Synchronizations-page. After the adjustment, run the initialization once and then set the desired synchronization.

Separate synchronization of all devices - external timer

Illustrated by example of: DCF77

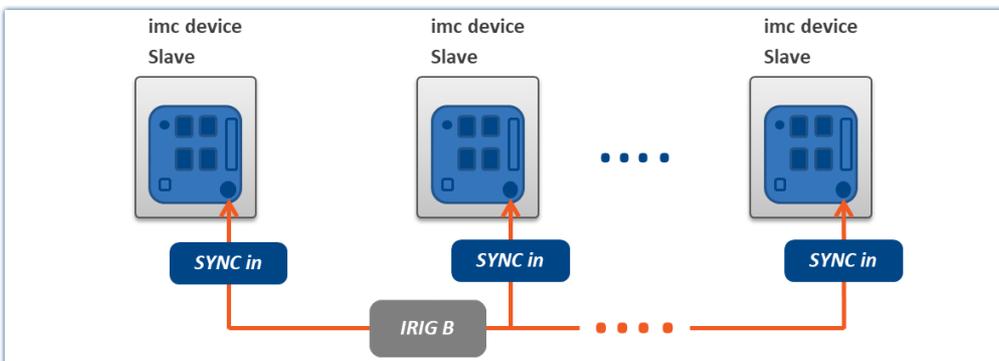
Each device is synchronized to absolute DCF77 time and thus all devices are mutually synchronized. The antenna signal must be converted to by a decoder to TTL level.



Example of individual synchronization with DCF77

Illustrated by example of: IRIG-B

Each device is synchronized to absolute IRIG-B time and thus all devices are mutually synchronized.



Example of individual synchronization with IRIG-B

Parameter	Selection
Signal input	All: Synchronization signal SYNC
	 With DCF77 and IRIG-B, select "SYNC" as the signal input. The hardware automatically recognizes which type is involved.
Signal output	All: No synchronization signal
External synchronization source	Yes

 Storage

 Synchronization

 Timestart

 Measurement options

External synchronization sources

Device name	Signal input	Signal output	Synchronization master
T_126678_CS_7008_1	Synchronization signal SYNC	no synchronization signal	<input type="checkbox"/>
T_126680_CS_7008_1	Synchronization signal SYNC	no synchronization signal	<input type="checkbox"/>
T_160144_CRFX_2000	Synchronization signal SYNC	no synchronization signal	<input type="checkbox"/>

Settings in the plug-in Setup on the page "Devices" in the dialog "Synchronization"

9.2.3.4.2 GPS receiver as timer

Introduction: GPS

The GPS (Global Positioning System) receiver is connected to the "GPS" socket. Depending on the device type, the GPS connection is either a 9-pin DSUB or a 7-pin LEMO connector. The supply is provided by the measurement device. In this way, the GPS receiver finds its positions as soon as the device is activated.

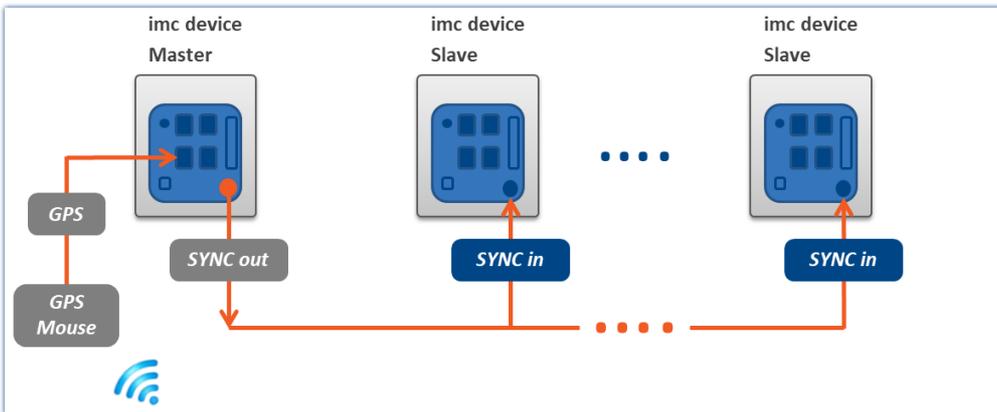
The GPS signal, received all around the world, returns, among other things, Greenwich Mean Time (GMT)

Duration of synchronization process: The minimum duration for synchronization depends on the reception and on the last location of use. Upon first use, the receiver needs a few minutes until multiple satellites and its own position have been located. Upon the next start, the last position data are used, and the satellites are found more quickly.

With a direct line of sight to the sky, the receiver is typically ready to transmit time and position data to the device after 1 minute.

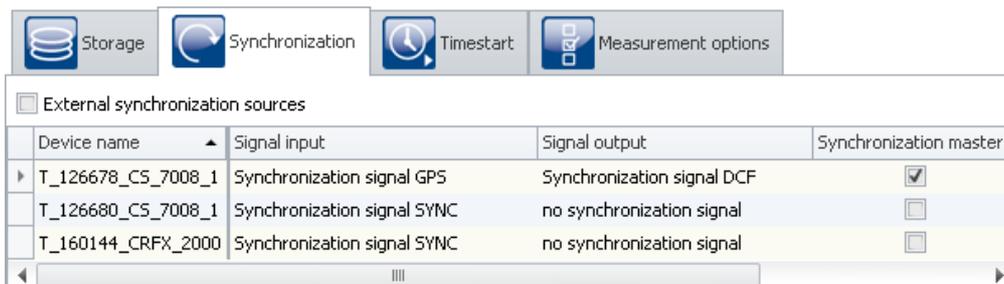
Master/Slave setup

The devices are synchronized with each other and synchronized to absolute GPS time.



Master device synchronized by GPS and slave devices by SYNC

Parameter	Selection
Signal input	Master: Synchronization signal GPS Slave: Synchronization signal SYNC
Signal output	Master: Synchronization signal DCF or IRIG-B Slave: No synchronization signal
External synchronization source	No



Settings in the plug-in Setup on the page "Devices" in the dialog "Synchronization"

Different response for GPS-synchronization + SYNC-output

- **For devices of [group A4](#)**¹⁴⁶: If DCF-output is activated for a device and simultaneously GPS is set as the synchronization input, the DCF-output only as long as the device is synchronized to the GPS-source. If the GPS-source fails, the DCF-output is temporarily deactivated!
- **For devices of [groups A5-A7](#)**¹⁴⁶ **and devices belonging to the [firmware group B](#)**¹⁴⁶: SYNC-output begins upon the first synchronization with GPS. If there is an outage of the GPS-signal, the SYNC-signal continues to be outputted anyway. All devices connected remain synchronized to the Master. Once the GPS-signal resumes, the resulting phase offset to the Master is corrected, see [Phase offset correction](#)²²⁰. In the process, all connected SYNC Slaves are automatically adjusted to have the same phase.

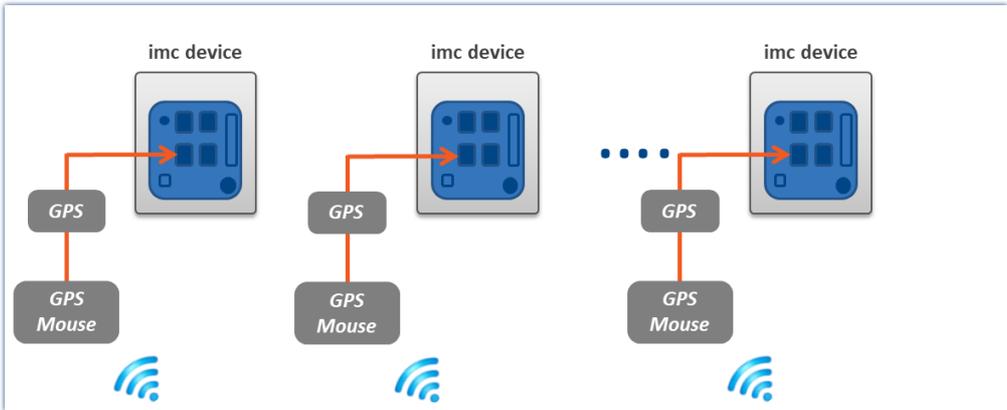
Different response for GPS- and SYNC synchronization + SYNC-output

- **For devices of [group A4](#)**¹⁴⁶: DCF-output begins immediately. If a GPS-signal is applied, the system synchronizes to it. If the GPS-signal has a temporary outage or if it is no longer available after preparation of the measurement, the device synchronizes to its own DCF-output. The GPS-signal is then no longer taken into account for the running measurement!
- **For devices of [groups A5-A7](#)**¹⁴⁶: SYNC-output begins immediately. Any GPS-signal applied always supersedes the device's own SYNC-signal as the synchronization source! If no GPS-signal is applied, the device synchronizes to its own SYNC-output. As soon as a GPS-signal is available, the device synchronizes to the GPS. The SYNC-output is briefly interrupted, so that all SYNC-Slaves adopt the GPS time. Following the first synchronization with GPS, the SYNC-output remains intact even if the GPS-signal fails!
If the GPS-signal fails again, the synchronizes to its own SYNC-signal. If the external GPS-signal returns, the device synchronizes to the device by correcting the resulting [phase offset](#)²²⁰. This automatically also adjusts all connected SYNC-Slaves to the phase.

Device name	Signal input	Signal output	Synchronization master
T_126678_CS_7008_1	Synchronization signal SYNC or GPS	Synchronization signal DCF	<input checked="" type="checkbox"/>
T_126680_CS_7008_1	Synchronization signal SYNC	no synchronization signal	<input type="checkbox"/>
T_160144_CRFX_2000	Synchronization signal SYNC	no synchronization signal	<input type="checkbox"/>

Separate synchronization of all devices

Each device is synchronized to absolute GPS time and thus all devices are mutually synchronized.



Example of individual synchronization with GPS

Parameter	Selection
Signal input	All: Synchronization signal GPS
Signal output	All: No synchronization signal
External synchronization source	Yes

Storage Synchronization Timestart Measurement options

External synchronization sources

Device name	Signal input	Signal output	Synchronization master
T_126678_CS_7008_1	Synchronization signal GPS	no synchronization signal	<input type="checkbox"/>
T_126680_CS_7008_1	Synchronization signal GPS	no synchronization signal	<input type="checkbox"/>
T_160144_CRFX_2000	Synchronization signal GPS	no synchronization signal	<input type="checkbox"/>

Settings in the plug-in Setup on the page "Devices" in the dialog "Synchronization"

9.2.3.4.3 NTP

Introduction: NTP

The Network Time Protocol (NTP) is a standard for synchronizing clocks in computer systems and uses the transport protocol UDP. It was specially developed to enable reliable distribution of time information via networks having variable package runtimes.

Precision: NTPv4 can synchronize a local system's time via the public Internet at a precision of approx. 10 ms. In local networks under ideal conditions, precision levels of 0.2 ms and even better are possible. See the technical data sheet for achievable precision.

Duration of synchronization process: Synchronization to an NTP server is a lengthy process. At intervals, the device exchanges a data package with the server. With the default settings and a large time divergence it can take up to 3 h to achieve a precision level in the range of 20 ms, or up to 12 h until the deviation is <5 ms.

In order for synchronization to begin immediately upon system activation, NTP can be set up as the measurement device's default synchronization input (see "[device properties](#)"¹⁵³).

There you can also reduce the interval time in order to achieve better precision and faster synchronization. "Min-Interval" counteracts large divergence; as the values improve, the time interval increases up to "Max-Interval".

After the first synchronization procedure, the device should remain activated for at least one hour. Only then will the frequency deviation determined be saved in the device. When the device is restarted, the frequency deviation is known and the synchronization process proceeds correspondingly faster.

The measurement device supports up to two NTP servers.

Application in imc WAVE

On the Setup page "Devices" in the dialog "Synchronization", the **signal input is configured**. For NTP-capable devices, the "Synchronization signal NTP" is available there, among others.

When the **signal input "Synchronization signal NTP"** is selected, an additional tab with the name "NTP" appears. Here, you can configure NTP for the devices. This tab only appears when the associated device is selected. Some of the parameters can also be configured as settings in the [device properties](#)¹⁵³.

You can insert the following parameters in the table or partly configure them under the "NTP" tab:

- Synchronization NTP server 1 and 2
- Synchronization NTP deviation time max.
- NTP synchronization interval
- Synchronization waiting time on self start max. (applies to all synchronization types)

For making NTP settings, the following parameter belonging to the [device properties](#)¹⁵³ is also relevant:

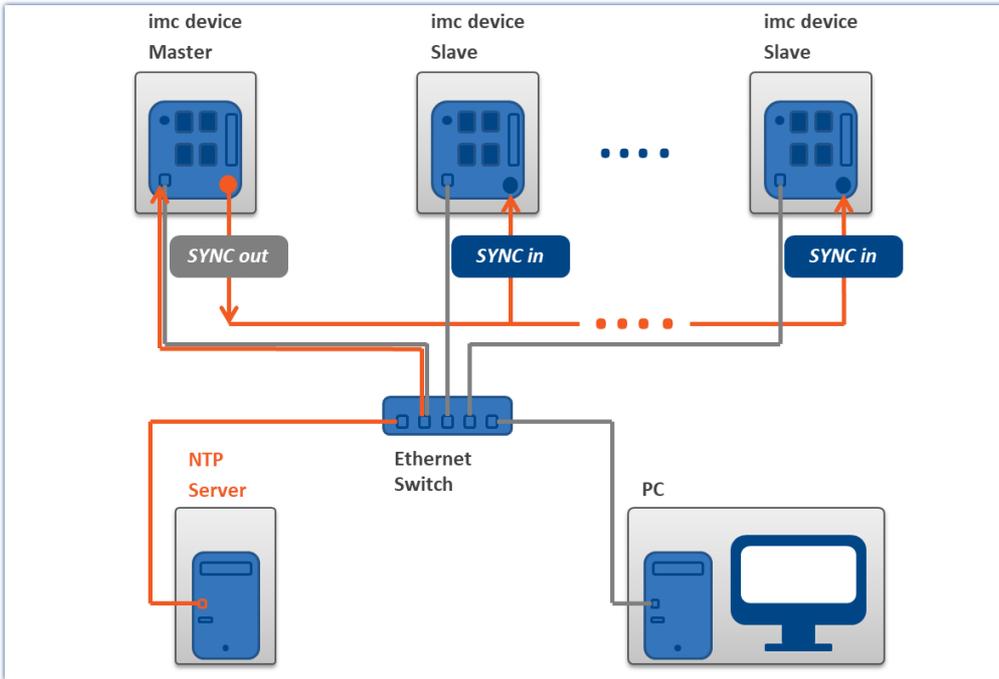
- observance of the "daylight saving's time" transitions

Note

- If the phasing between two voltages in different devices is to be compared, only the Master is to be synchronized to NTP. This synchronizes additional devices via SYNC (DCF77/IRIG-B), see "[Master/Slave setup](#)"²⁰⁸.
- The quality of the synchronization can be monitored with the help of the pv-variable "pv.SyncTimeDeviation". It returns the estimated deviation of the NTP-server' clock time in seconds. The momentary value is not relevant! only after observing its course over longer periods is it possible to get an impression of the synchronization quality!
- How constant the synchronization is depends strongly on temperature fluctuations in the device! Every °C can temporarily cause a sizable deviation. High precision is only achieved under constant temperature conditions (e.g. in switching cabinets).

Master/Slave setup

The devices are synchronized with each other and synchronized to absolute NTP time.



Master device synchronized by NTP and slave devices

Parameter	Selection
Signal input	Master: Synchronization signal NTP Slave: Synchronization signal SYNC
Signal output	Master: Synchronization signal DCF or IRIG-B Slave: No synchronization signal
External synchronization source	No

Storage Synchronization Timestart Measurement options

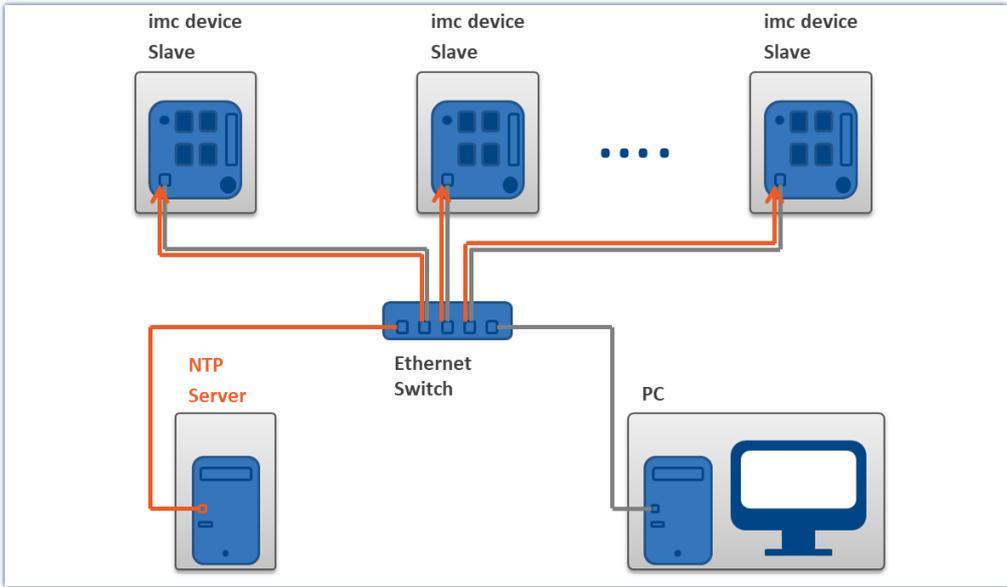
External synchronization sources

Device name	Signal input	Signal output	Synchronization master
T_126678_CS_7008_1	Synchronization signal SYNC	no synchronization signal	<input type="checkbox"/>
T_126680_CS_7008_1	Synchronization signal SYNC	no synchronization signal	<input type="checkbox"/>
T_160144_CRFX_2000	Synchronization signal NTP	Synchronization signal DCF	<input checked="" type="checkbox"/>

Settings in the plug-in Setup on the page "Devices" in the dialog "Synchronization"

Separate synchronization of all devices

Each device is synchronized to absolute NTP time. All devices are thus mutually synchronized, although to a limited extent (see: "[Precision](#)"^[206]).



Example of individual synchronization with NTP

Parameter	Selection
Signal input	All: Synchronization signal NTP
Signal output	All: No synchronization signal
External synchronization source	Yes

Storage
 Synchronization
 Timestart
 Measurement options

External synchronization sources

Device name	Signal input	Signal output	Synchronization master
T_160005_CRC	Synchronization signal NTP	no synchronization signal	<input type="checkbox"/>
T_160007_CRC	Synchronization signal NTP	no synchronization signal	<input type="checkbox"/>
T_160144_CRFx_2000	Synchronization signal NTP	no synchronization signal	<input type="checkbox"/>

Settings in the plug-in Setup on the page "Devices" in the dialog "Synchronization"

9.2.3.4.4 PTP

Introduction: PTP

The Precision Time Protocol (PTP) serves to synchronize clocks within distributed computer systems. Compared with NTP, its achievable precision is significantly higher.

Characteristics	Description
Prerequisites and limitations	<p>PTP-capable device ¹⁴⁶ are required.</p> <p>PTP-capable network equipment (switches) and optionally a PTP time server are required. The synchronization is confined to the local network. No possibility for synchronization via the Internet is provided.</p>
Structure	<p>PTP creates a self-organizing hierarchical structure, in which each device listens for information from the current time server throughout the network. If it does not identify any server, or if its own information is superior to that of the current server, it sends its own information to the network. The current server as well as all clients recognize that there is a new and better server and use the new server from then on. The previous current server stops sending its information.</p> <p>In principle, any PTP-capable imc device can also serve as a time server. By means of the configuration option "Slave only", it is possible to prevent a device from making itself available as the time server.</p>
Precision	Under good conditions, PTP achieves a precision to the extent of well less than 1 μ s. See the technical data sheet for achievable precision.
Duration of synchronization process	<p>The synchronization takes about 20-30 s after preparing the measurement. Depending on the set PTP parameter, the synchronization can last longer.</p> <p>The sync process takes longer if the measurement was started before the device could sync its time with the time server. The offset in time will then be balanced over a longer period of time by adjusting the device's internal clock (SyncRTC).</p>
Time zone	For the use of PTP it is necessary for the time zone to be set in appropriately for one's own needs, or else there may be an unintended time offset between the imc device and other devices or the PC.

Note

The parameters "*First step threshold*" and "*Step threshold*" can be used to influence the behavior before and after the start of the measurement (see the description of the parameters).

Application in imc WAVE

On the Setup page "*Devices*" in the dialog "*Synchronization*", **the signal input is configured**. For PTP-capable devices, the "*Synchronization signal PTP*" is available there, among others.

When the **signal input "Synchronization signal PTP"** is selected, an additional tab with the name "PTP" appears. Here, you can configure PTP for the devices. This tab only appears when the associated device is selected.

Presets

Using various parameters it is possible to modify the protocol to meet your requirements. imc STUDIO offers a number of presets for these parameters. Additionally, you are able to edit every parameter.

Preset	Description
IEEE 1588 2008	Standard defined by IEEE (default setting)
fos4X	Configuration for allowing synchronization with the fos4X devices, which can be used in imc STUDIO as Third Party devices.
User-defined	Enables configuration of all PTP-parameters. If "user-defined" is selected, all parameters are displayed.

Note

The values set are discarded as soon as another preset is selected.

Note

- The PTP parameters are saved along with the experiment.
- For the parameter "Signal output", there is no option like "Synchronization signal PTP".
- If the parameter "External synchronization source" is activated in the dialog "Synchronization", then for all PTP-devices "Slave only" is activated.

Comparison: NTP vs. PTP

	NTP	PTP
Precision	10 ms via internet 200 µs in the local network	below 1 µs (down to 100 ns achievable)
Settling	up to 3 h, for a 20 ms precision level <small>[206]</small>	approx. 20-30 s
Structure	internet time server supported specify time server	for local networks only self-organizing: the best server wins
Hardware	standard hardware	special devices: group A7 and higher with the suffix "-GP", as well as CRXT. PTP compatible Network switch required

PTP - Parameter

Among parameters there is a distinction between "Device-specific" and "Experiment options (for all devices)". "Device-specific" always applies to the device currently selected in the Devices table.

If you selected the preset "User-defined", you are able to modify all PTP parameters according to your requirements. The parameters are displayed as soon as "User-defined" is selected.

The following parameters can either be modified or are specified by the preset (the description of the parameters is provided further below):

Device-specific options				
Parameter	Min	Max	IEEE 1588 2008 (Default)	fos4X
Slave only	0	1	0	0
Priority 1	0	255	128	127
Priority 2	0	255	128	127
User description				
Delay asymmetry	-1000000	1000000	0	0
UDP TTL	1	255	1	1

Experiment options (for all devices)				
Parameter	Min	Max	IEEE 1588 2008 (Default)	fos4X
Domain number	0	255	0	0
Announce interval	-3	5	1	1
Sync interval	-6	5	0	-2
Delay request interval	-6	5	0	0
Announce receipt timeout	2	10	3	3
Delay mechanism	Auto, E2E, hybridE2E, P2P		E2E	E2E
Network transport	UDPv4, UDPv6, L2/IEEE 802.3		UDPv4	UDPv6
TAI-UTC offset	-32768	32767	36	36
UDPv6 scope	0x02	0x0E	0x0E	0x02
Transport specific	0x00	0x0F	0x00	0x00
Step threshold	0.0	DOUBLE_MAX	0.0	0.0
First step threshold	0.0	DOUBLE_MAX	0.00002	0.00002
PTP destination MAC address			01:1B:19:00:00:00	
P2P destination MAC address			01:80:C2:00:00:0E	

Description

Device-specific options	
Parameter	Description
Slave only	Prevents the device from providing its clock time to other subscribers. This parameter can be set for all Preset-variants.

Device-specific options	
Parameter	Description
Priority 1	Sets the priority according to which the devices choose a server. The subscriber with the lowest value becomes the server. Priority 1 is the strongest among criteria!
Priority 2	Sets the priority according to which the devices choose a server. The subscriber with the lowest value becomes the server. Priority 2 is a weak criterion! Priority 1 and the server quality characteristics specified by the device manufacturer outweigh the parameter Priority 2, which is only evaluated if all other values are identical.
User description	Identification of the device for PTP-Management programs.
Delay asymmetry	With this value, a deterministic time offset resulting from asymmetric packet run times in the network can be compensated. For this purpose, exact information or measurements are necessary. If you use appropriate PTP-switches, it should not be necessary to adjust this parameter.
UDP TTL	Lifespan of the PTP-packets in the network. If a PTP-network is to be operated beyond the confines of one router, this value must be increased accordingly. This value is only taken into account if UDPv4 was set as the network transport.

The following parameters match for all devices involved. This applies not only to the devices from imc but to all switches, PTP-servers and other devices!

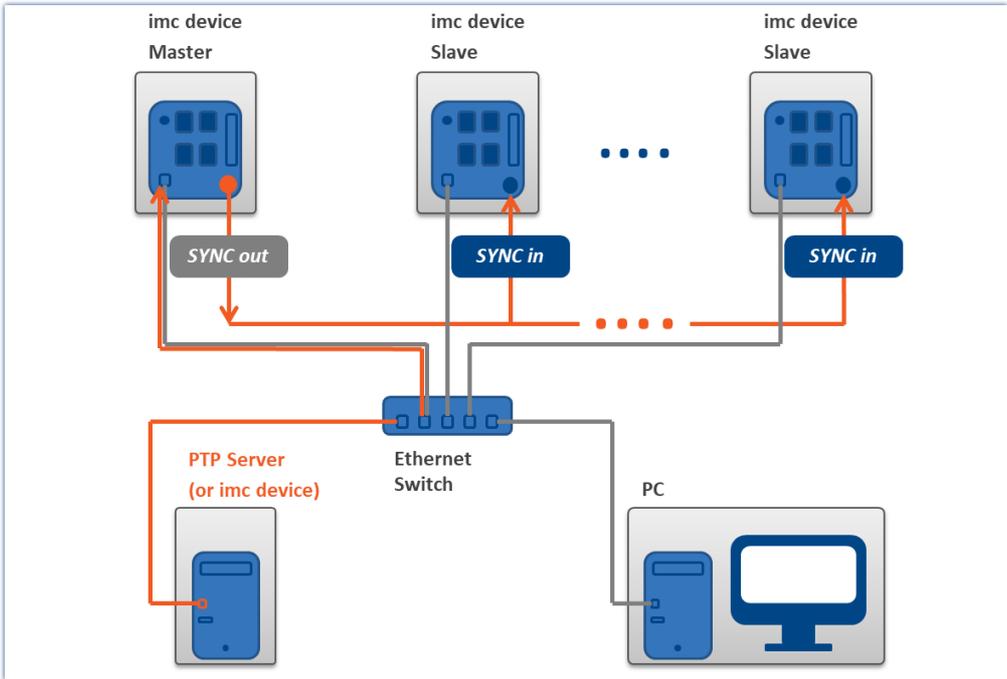
Experiment options (for all devices)	
Parameter	Description
Domain number	Using the domain number it is possible to create multiple independent PTP-domains. Devices having differing domain numbers do not influence each other.
Announce interval	This interval determines at what intervals the current time server sends its quality information to the network. The value is stated as a power of 2. The value range of -3 to 5 covers interval times of 1/8s to 32s.
Sync interval	This interval determines at what intervals the current time server sends its time information to the network. This value is stated as a power of 2. The value range of -6 to 5 covers interval times of 1/64s to 32s. When the value is less than -3, imc devices can no longer send data at the speed required. These settings are only offered for reasons of compatibility and should be avoided. Note that large values lead to long intervals and thus slow down the synchronization and decrease its precision!
Delay request interval	This interval specifies at what intervals the PTP-slaves determine the packet run times in the network. The value is stated as a power of 2. The value range of -6 to 5 covers interval times of 1/64s to 32s. When the value is less than -3, imc devices can no longer send data at the speed required. These settings are only offered for reasons of compatibility and should be avoided. Note that large values lead to long intervals and thus slow down the synchronization and decrease its precision!
Announce receipt timeout	Specifies how many "Announce Intervals" a device waits until it assumes that the current PTP-master no longer exists and takes appropriate measures. Example: Announce interval = 1, Announce receipt timeout = 3 -> The time is: $2^1s * 3 = 6s$.

Experiment options (for all devices)	
Parameter	Description
Delay mechanism	<p>For measurement of the packet run times, multiple procedures are available. Select the procedure here which you wish to use.</p> <p>E2E: With the End-To-End procedure, the packet run time along the entire route between the PTP-Slave and the PTP-Master. This procedure is the one most commonly supported and theoretically also works with switches which are not PTP-capable. For this reason it is the default. However, it is strongly to be recommended NOT using switches unless they explicitly support PTP, otherwise unpredictable behavior can result which can impair the synchronization by any unforeseeable amount. The procedure can be used without any problems in conjunction with appropriate switches.</p> <p>P2P: The Peer-To-Peer procedure can exclusively be used with P2P-capable infrastructure. Here, the packet run times are determined between two directly connected nodes (device and switch or switch and switch). On the way between the master and slave, one correction value from each switch traversed is incremented in the packet by the run-time determined and the dwell time in the switch. The receiver can calculate and incorporate the total run-time. This procedure has advantages when there are multiple switches between the master and the slaves.</p> <p>Auto: With this setting, "E2E" is used initially. As soon as a "P2P"-packet is recognized, the setting switches to "P2P".</p> <p>hybridE2E: This is an improved form of "E2E" and reduces the network load since it uses Unicast packets instead of multicast packets. However, it is not supported by all commercially available devices.</p>
Network transport	Depending on the infrastructure used, select UDPv4, UDPv6 or L2/IEEE802.3.
TAI-UTC offset	<p>By default PTP uses atomic time (Temps Atomique International, TAI) as the time base.</p> <p>In the Announce packets, along with the server's quality characteristics, the offset between atomic time and UTC is also transmitted. In order that the device can distribute the correct offset, the current value must be entered here.</p>
UDPv6 scope	Sets the lifespan of the PTP-packets if UDPv6 is set as the "Network transport".
Transport specific	Only needed if you use both imc devices and other devices in a shared PTP-network and the other devices here use a value which is not 0x00. Make the same setting here.

Experiment options (for all devices)	
Parameter	Description
Step threshold	<p>Sets the threshold value as of which a time deviation can no longer be compensated for by adjusting the device clock. This value is only taken into account once the time may no longer jump in the device clocks, which means after the start time of the first measurement after the initialization is set.</p> <p>If the value is 0, any deviation however large is compensated, which can take very long when deviations are large.</p> <p>If the value is not equal to 0 but smaller than the deviation from the PTP-Master, the frequency of the clock rate signals in the device, and thus also of all signal samples, is synchronized to the frequency of the PTP-Master. By contrast, the time deviation is NOT compensated for, but instead kept constant. In this way, it is possible to avoid the signal sampling either accelerating or slowing down over the time during which the deviation is compensated. This can be helpful for measurements in which the exactness of the sampling frequency is more important than the exactness of the absolute time.</p>
First step threshold	<p>This value works in the same way as "Step threshold", but is only used between the initialization of a measurement and when the start time of the the first measurement is determined after the initialization.</p> <p>Note:</p> <p>With the default settings, every time deviation after a measurement is initialized is eliminated within a few seconds. For very small deviations, this is accomplished by brief temporary modification of the clock frequency; for larger deviations, the device clock is directly adjusted.</p> <p>However, if a measurement is started before this procedure was completed, then regardless of how large the deviation is, it is compensated by modifying the clock frequencies in the device.</p>
PTP destination MAC address	Only change this value if you use a different MAC-address in your infrastructure.
P2P destination MAC address	Only change this value if you use a different MAC-address in your infrastructure.

Master/Slave setup

The devices are synchronized with each other and synchronized to absolute PTP time.



Master device synchronized by PTP and slave devices

Parameter	Selection
Signal input	Master: Synchronization signal PTP Slave: Synchronization signal SYNC
Signal output	Master: Synchronization signal DCF or IRIG-B Slave: No synchronization signal
External synchronization source	No

Storage Synchronization PTP Timestart Measurement options

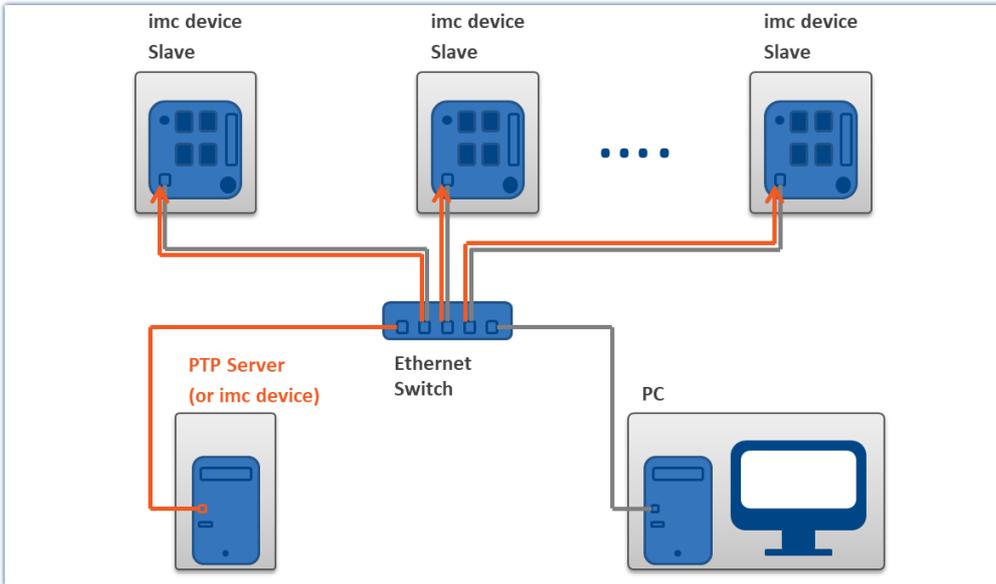
External synchronization sources

Device name	Signal input	Signal output	Synchronization master
T_126678_CS_7008_1	Synchronization signal SYNC	no synchronization signal	<input type="checkbox"/>
T_126680_CS_7008_1	Synchronization signal SYNC	no synchronization signal	<input type="checkbox"/>
imcDev__PTP_1	Synchronization signal PTP	Synchronization signal DCF	<input checked="" type="checkbox"/>

Settings in the Setup on the page "Devices" in the dialog "Synchronization"

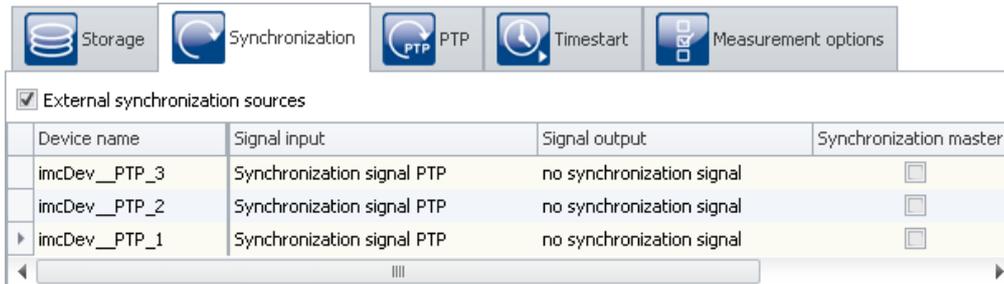
Separate synchronization of all devices

Each device is synchronized to absolute PTP time. All devices are thus mutually synchronized. See the information about precision (see: "[Precision](#)^[210]").



Example of individual synchronization with PTP

Parameter	Selection
Signal input	All: Synchronization signal PTP
Signal output	All: No synchronization signal
External synchronization source	Yes



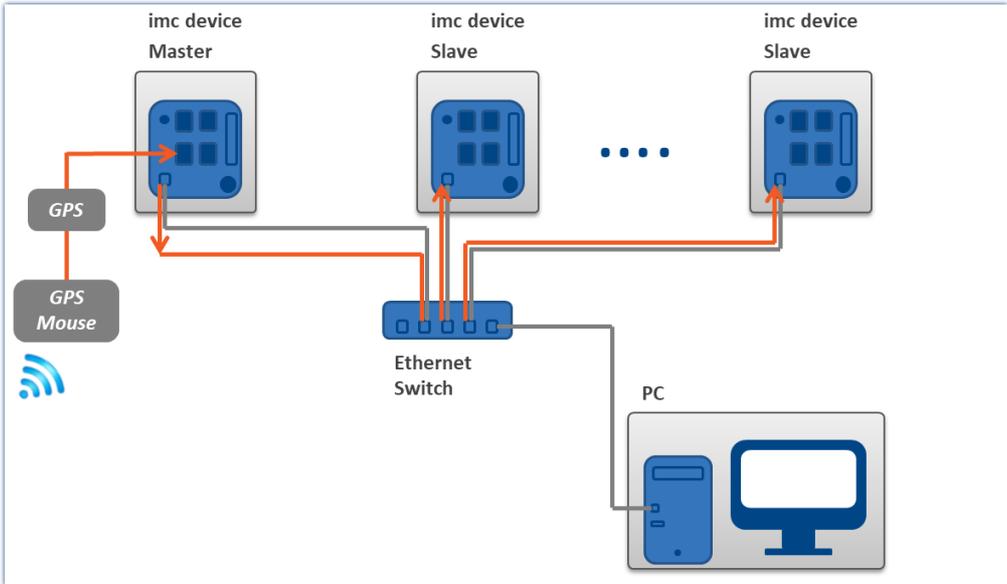
Settings in the Setup on the page "Devices" in the dialog "Synchronization"

9.2.3.4.5 PTP-Master only

Due to the presets, an imc device only becomes the PTP-Master if no "better" PTP-master is available. See section: "[Introduction: PTP](#)^[210]".

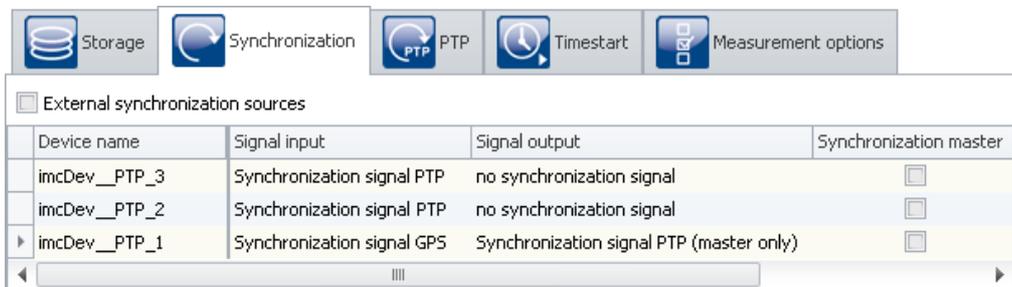
Any unsynchronized imc device does not have **precise absolute time**. These devices should not become the master in a PTP-network. The PTP-parameters are selected accordingly. Any **imc device is identifies as a "bad" clock. Other subscribers which are true to absolute time** should accordingly be identified as a "good" PTP-master.

If the network includes **no PTP-server true to absolute time**, you can **synchronize one imc device** to a hardware synchronization signal (e.g. GPS). When you define that **device as the PTP-master**, it will no longer be synchronized by any other PTP-subscriber.



A PTP-master clock synchronizes itself to the GPS-clock. All other PTP-subscribers in the network can be synchronized to this master.

Parameter	Selection
Signal input	Master: Synchronization signal GPS
	Slave: Synchronization signal PTP
Signal output	Master: Synchronization signal PTP (master only)
	Slave: No synchronization signal
External synchronization source	No



Settings in the Setup on the page "Devices" in the dialog "Synchronization"

Set the **signal input of the Master device** appropriately, e.g. GPS, SYNC, ...

Set the signal output to "*Synchronization signal PTP (Master only)*". All other imc devices in the experiment having "*Synchronization signal PTP*" as the signal input are automatically set to "*Slave-only*".

The synchronization procedure

The master **initially does not return time information** via PTP. Only after the "Master" **synchronizes to the external signal** does it return time information via the PTP-work.

- If the **only PTP-subscribers are imc devices**, they synchronize themselves with the Master specified.
- If there are **additional PTP-subscribers**, then the imc slave-devices may already be synchronized to the other subscribers. The **devices report that they are synchronized**, although they are **not synchronized with the imc-Master device**.

As soon as the **master device returns the time information**, the other **devices synchronize themselves to it**. How this is done depends on the parameters. The **devices select the subscriber of best quality** (see "[Introduction: PTP](#)"^[210]). As soon as any device is defined as the "*PTP (Master only)*", it is assigned a **slightly higher quality** than the other imc devices. Thus, other imc devices, which are not involved with the experiment but which are subscribers in the network, can also synchronize themselves to the master.

! **Note**

Network structure and further notes

- Please always be aware of your PTP-network's setup in order to avoid any unintended effects. If another subscriber has higher quality, then the slave-devices may have an "absolute" time which is different than the master-device's.
- A "*Synchronization master*" must not be selected in this variant.
- Please observe the notes in the section "[Precision](#)"^[232] and "[Constraints of the Synchronization](#)"^[233], especially regarding "*phase*" and "*RAM buffer duration*".

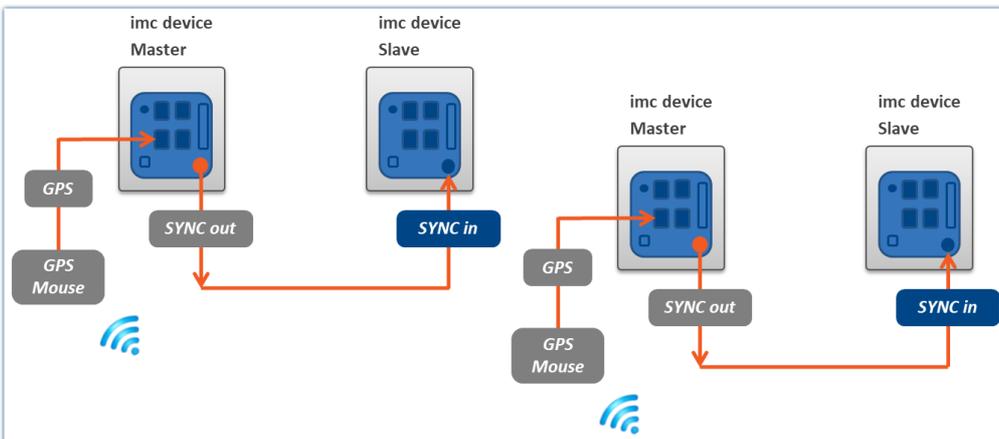
PTP-Master only, and DCF or IRIG-B output

As an additional way to output the time information as the PTP-master, the device can also be defined as the DCF- or IRIG-B-Master.

Set the signal output to "*Synchronization signal PTP (Master only) and DCF/IRIG B002*".

9.2.3.4.6 Synchronized start with multi-master-devices

If it is not possible to connect multiple devices by a SYNC line, it is possible to divide the devices into multiple Master/Slave groups:



Example for two Master/Slave combinations

Storage

Synchronization

Timestart

Measurement options

External synchronization sources

Device name	Signal input	Signal output	Synchronization master
T_126678_CS_7008_1	Synchronization signal GPS	Synchronization signal DCF	<input type="checkbox"/>
T_126680_CS_7008_1	Synchronization signal SYNC	no synchronization signal	<input type="checkbox"/>
T_141127_CRFX_400	Synchronization signal GPS	Synchronization signal IRIG B002	<input type="checkbox"/>
T_160144_CRFX_2000	Synchronization signal SYNC	no synchronization signal	<input type="checkbox"/>

Settings in the Setup on the page "Devices" in the dialog "Synchronization"

 Note

Since in multi-Master/Slave operation there are multiple masters outputting the SYNC signal, the SYNC outputs of the various masters may not be connected with each other.

9.2.3.5 Phase offset correction

For devices of the [group A4](#)

With devices of group A4, following an interruption of the synchronization signal only the frequency error is corrected (i.e. the frequency of the SyncRTC and of the signal sampling is synchronized with the time source). Any phase error is not corrected. The offset remains intact until the next initialization.

For devices of the [groups A5-A7](#) and devices belonging to the [firmware group B](#)

If the measurement device had been synchronized to a (time) source and if it receives this same source again after an interruption, the phase offset which emerged in the meantime is corrected. Signals captured by different devices are then synchronized just as well after resumption of reception as before the interruption.

If an implausible phase offset is calculated upon re-synchronization, the system assumes that the time source is different than before. In this case, no phase correction is performed during the running measurement, instead only the frequency error is corrected (i.e. the frequency of the SyncRTC and of the signal sampling is synchronized with the time source). A completely new synchronization procedure is performed when the measurement is re-initialized. The same applies in the case of a transition from DCF to GPS etc. .

9.2.3.6 Working with time zones

The abbreviations pertinent to the topic of time zones are presented in the section "[Definitions](#)" .

The time zone information includes the shift from the local standard time STD to UTC, as well as the difference between STD and DST and the respective time offsets.

Only when complete time zone information for two zones is supplied can the times be converted to local time.

One special case is a "double" hour in which the time is set back. Here, it is not enough to know in which time zone the time is stated. It is also necessary to know whether the time is time is stated in STD or DST.

If a planned DST-switch is not performed, the time may be correctly indicated due to explicit specification of the DST-state!

 Example

- Time zone for Germany, 05.07. 14:00 STD
- This moment is actually during daylight saving time and should be stated as 15:00 DST
- However, the conversion to UTC in both cases correctly returns 13:00 UTC

What are time zones good for?

In order to make it possible to assign a time to measured data, the measurement can only be prepared when a time zone is set for every imc device used.

A device would assume that UTC time applies if it is synchronized using GPS, NTP, etc. This can be resolved by specifying a time zone. Then the device applies the time zone specified, regardless of what synchronizations source is used.

The DST switch is activated in the [device properties](#)¹⁵⁵: "*Device time, synchronization*" > "*Time zone*" and "*Automatic daylight saving time switching*".

Action	Page
Dialog: Device options > Device Properties	Devices

The setting "*deactivated*" means that the device uses standard time (STD) all year, summer and winter! Even if the PC resets the device clock during daylight saving time, then, for instance 14:00 (DST in the PC) becomes 13:00 in the device (STD). The advantage of this is that there are no time skips in the data which would be reflected in the measured data!

Note

Even a DCF-or IRIG B002-source would only return STD in that case. Otherwise, the time is interpreted incorrectly - there is an error of one hour!

For the setting "*activated*", the PC's or synchronization source's time is interpreted and adopted according to the time zone expected.

In order for a time source's time to be adopted as the device's local time, the time source's time zone as well as its DST-state must be known.

The time supplied to the device via the RS232-interface in the form of NMEA-strings (GPS, IRIG B12X to GPS converter), is interpreted as UTC time.

With the NTP-protocol, the time zone of the clock source is also UTC. With PTP: Atomic time (TAI) + TAI-UTC-offset or also UTC. For this reason, the time statement is complete in such cases.

The time at the SYNC input terminal (DCF/IRIG B002) is expected to match the device's current time zone. The DST-state for these time signals can not always be determined correctly.

Example

- A device is to use DCF-synchronization and is set to the Central European time zone with DST-switching.
- The DCF-source returns the time 03.01.2010 04:58:13 hours
- The device interprets the time as STD, since according to the time zone setting this time is in STD.
- The DCF-source returns the time 14.07.2010 14:34:00 hours
- The device interprets the time as DST, since according to the time zone settings, this time is in DST.

If the source is not in the correct DST-state, there is a resulting error of one hour. For this reason, the device only adopts the source's DST-state upon the first synchronization procedure after activation or upon preparation of the measurement.

The "forbidden hour" resulting from the clock skipping forward is always converted to the corresponding next hour. A time during the "duplicate hour" prior to the clock skipping backward is always interpreted as DST!

After an interruption of the synchronization signal, upon re-synchronization the system determines whether the deviation between the time reference and the device time is approx. 1h (the difference between STD and DST). In that case, the system assumes that the DST-state of the time reference no longer matches the device's DST-state.

This can be the case when the device does not perform the DST-switch because a measurement is already in process or if the device did perform the DST-switch but not the DCF-source.

A device of [groups A5-A7 or devices belonging to the firmware group B](#) ¹⁴⁶ maintains its DST-state, but correctly adopts the source's time.

A device of [group A4](#) ¹⁴⁶ always adopts the DCF-signal's DST-state upon DCF-synchronization, but upon repeated synchronization it recognizes if the source has not performed the DST-switch. Thus, the time is interpreted correctly as long as the DST-state was correct upon the first synchronization.

Note

With the setting *Switching from/to daylight saving time* activated, the switch between standard and daylight saving time happens automatically at the appropriate time. However, the following limitations apply:

- Once a measurement's start time has been determined (Start button pressed), the device clock can no longer be adjusted. The DST-state, as well, must remain intact. For this reason, the DST-switch is only performed upon the preparation of the next measurement.
- With devices of [groups A4](#) ¹⁴⁶, the SyncRTC is adjusted directly by the DCF-synchronization. Thus it is not possible to change the DST-state as long as there is DCF-synchronization. If the DCF-signal is not valid at the time of the switch, or if it fails afterwards, the switch is performed. Upon re-synchronization to DCF, the device again adopts the DCF-signal's DST-state. If the DCF-signal fails again, there is no new switch of the DST-state!
- A measurement always uses the device's current DST-state which applied at when the measurement's start time was determined, and retains it no matter whether the SyncRTC was adjusted. Only upon the next preparation of a measurement is it possible to change the DST-state.
- Diskstart, Autostart, Suspend/Resume (imc BUSDAQ) and automatic timed start always lead to preparation of a measurement and can cause the DST-state to change.

9.2.3.6.1 Possible scenarios

Example 1

A device of [group A4](#) ¹⁴⁶ is to use DCF-synchronization and is set to a time zone with DST-switching. At the time of the switch from STD to DST, DCF-signal is interrupted.

The device switches to DST.

A measurement is started. The measurement uses DST.

The DCF-signal is applied again and the device is re-synchronized.

If the DCF-signal continues to return STD time, the SyncRTC adopts the time as STD. However, the running measurement retains DST.

If the DCF-signal returns DST because the DCF-source also has made the DST-switch, the SyncRTC adopts the time as DST.

Example 2

A device of [group A4](#) ¹⁴⁶ is to use DCF-synchronization and is set to a time zone with DST-switching. At the time of switching from STD to DST, a measurement is already running (in STD).

The connected DCF-source performs the DST-switch.

The SyncRTC adopts the DST from the DCF-signal. However, the running measurement retains STD.



Example 3

A device of [group A4](#)^[146] is to use DCF-synchronization and is set to a time zone with DST-switching. The connected DCF-source DOES NOT perform DST-switching. The device retains its DST-state.



Example 4

A device of [groups A5 to A7](#)^[146] is to use DCF-synchronization and is set to a time zone with DST-switching. The device switches to DST at the appropriate moment, no matter whether the DCF-source switches or not. The DCF-signal is interrupted. The DCF-signal is applied again and the device re-synchronizes. The device adopts the correct time in any case and the SyncRTC continues to run in DST time no matter whether the DCF-source is still running in STD or in DST.



Example 5

A device of [groups A5 to A7](#)^[146] is to use DCF-synchronization and is set to a time zone with DST-switching. At the time of the switch from STD to DST, a measurement is already running (in STD). The connected DCF-source performs the DST-switch. The SyncRTC and the running measurement retain STD.

9.2.3.6.2 Summary

If the external source returns incorrect time, for instance because

- a DST-switch was omitted, or
- if the DST-switch happens even though it is not activated for the device, or
- if the time zone does not match that of the device, or if UTC is not supplied,

then the device's time information is faulty!

Once the start time of the first measurement has been determined after the preparation, the SyncRTC may no longer be adjusted. For this reason, after setting the start time there may not be any DST-switch. For instance, a measurement whose start time is expressed in Standard Time is carried out according to Standard Time. As well, any subsequent measurements retain the same DST-state until the next preparation procedure.

For the interface's system time, the DST-switch will still be performed. The times recorded by the file system in the folders and files will have corresponding skips!

For a Diskstart, Autostart, Suspend/Resume (imc BUSDAQ) and automatic timed start, a preparation is performed! In these cases, the DST-state can change again! If the SyncRTC was unable to adopt a synchronization source's time because a measurement was running, then if needed the synchronization is interrupted during preparation and a new synchronization process is initiated! This always happens whenever necessary for exact adoption of the time.

Devices of [group A4](#) 146:

Synchronization source: External DCF-source (not the device's own DCF-output)

- *Expected time zone:* Local time according to the device's settings
- The SyncRTC is set directly by the hardware and cannot be influenced. Following synchronization, the SyncRTC always has the source's exact time.
- The system assumes that the external synchronization source returns the correct local time at all times! Upon first synchronization, the DCF-source's DST-state must be correct, otherwise there is an error of one hour!!!
- As long as the first measurement's start time has not yet been determined, the system monitors whether the next DST-boundary has been passed. As long as the DCF-signal is received there is no automatic DST-switch of the SyncRTC. If the DCF-signal fails, the DST-switch is performed. Upon re-synchronization following a DCF-signal outage, the system tries to determine whether the DCF-source also experienced the DST-switch. The DST-state of the DCF-time is adopted correctly.

Synchronization source: Device's own DCF-output (DCF-Master and DCF-Slave)

- *Expected time zone:* Local time according to the device's settings
- As long as the first measurement's start time has not yet been determined, the system monitors whether the next DST-boundary has been passed. Then the DCF-output is deactivated (so that it is possible to write to the SyncRTC), then the SyncRTC adjusted by the DST-difference and the DCF-output re-activated. If other devices synchronize to the outputted DCF-signal, they are assigned the new time.

Synchronization source: GPS / IRIG-B 12X converter

- *Expected time zone:* UTC
- The system assumes that the source returns UTC time. The time is converted to the device's time zone. Before determining the first start time, a DST-switch is performed. For this purpose, the DCF-output is temporarily deactivated, if needed. If other devices are to be synchronized to the outputted DCF-signal, they are assigned the new time.

Devices of [groups A 5 and higher](#) 146:

Synchronization source: External DCF-source (not the device's own DCF-output) / IRIG-B 002

- *Expected time zone:* Local time according to the device's settings
- The system assumes that the external synchronization source returns the correct local time at all times! Upon first synchronization, the SYNC-source's DST-state must be correct, otherwise there is an error of one hour!!!
- Prior to determining the first start time, a DST-switch is taken into account, independent of the SYNC-signal applied.
- Upon re-synchronization following an outage of the SYNC-signal, the system attempts to determine whether the source has performed the DST-transition. The time is adopted correctly. The device retains its DST-state.

Synchronization source: Device's own DCF-output (DCF-Master and DCF-Slave)

- *Expected time zone:* Local time according to the device's settings
- Prior to determining the first start time, a DST-switch is taken into account. For this purpose, the DCF-output is temporarily deactivated. If other devices synchronize to the outputted DCF-signal, they are assigned the new time.

Synchronization source: GPS / IRIG-B 12X converter / NTP / PTP (with UTC time base)

- *Expected time zone:* UTC
- The system assumes that the source returns UTC time. The time is converted to the device's time zone. Before determining the first start time, a DST-switch is performed. For this purpose, the SYNC-output is temporarily deactivated, if needed. If other devices are to be synchronized to the outputted SYNC-signal, they are assigned the new time.

Synchronization source: PTP (with atomic clock time base)

- *Expected time zone:* Atomic time (Temps Atomique International TAI)
- The source is assumed to provide Atomic Time, as well as the offset between Atomic Time and UTC of which it is aware. From this info, UTC can be calculated, making conversion to the device's time zone possible.
- If the device itself becomes the PTP-Master, it uses either the TAI-UTC-offset passed to it by the PC, or the offset which had last been communicated to it by another Master.

9.2.3.7 Application notes**9.2.3.7.1 Time-correct display of data in the curve window**

If synchronized data are to be displayed correctly in a curve window, the following setting must be made for the X-axis in the curve window: *Date/time absolute*. Otherwise the trigger times of the different signals are aligned. When different triggers are used, which is always the case when multiple devices are involved, the reference to absolute time is lost.

9.2.3.7.2 What happens when?

If possible, use the ability to run multiple devices jointly in an experiment and to connect them via the SYNC connection. Set one device as the Master which synchronized all other devices' clocks. The Master can also be synchronized externally, but this is only necessary for the absolute time. If the only issue is the mutual synchronization of the various devices' data, the Master's internal clock is perfectly adequate as the time base.

The following examples thus only pertain to external synchronization without connection of a SYNC-line between devices.

Adequate reception by the DCF77/IRIG-B or GPS receiver is not always available. How does the device behave then?

- It does not matter whether DCF77/IRIG-B or GPS is used for the time base.
- Temporary loss of synchronization can cause the device clocks to drift off by up to 1 ppm. If loss of synchronization continues, the precision may drop as low as 50 ppm in consequence of strong temperature change. Once the synchronization signal is again recognized as valid, the device clocks are synchronized again. As soon as the synchronization signal is recognized as valid, the system's behavior depends on the [device type](#)^[146]:

- With devices belonging to group A4, only the [frequency error](#)^[220] is corrected.
- Devices of groups A5 and higher attempt to compensate for the [phase error](#)^[220].

- Deviation directly after loss of synchronization (1 ppm):

$$t_{\text{delta}} = t_{\text{unsync}} \cdot 1 \cdot 10^{-6}; \quad \text{with } t_{\text{delta}} = \text{deviation in seconds}$$

$$t_{\text{unsync}} = \text{time without synchronization in seconds}$$

$$1 \cdot 10^{-6} = \text{precision of device time (1 ppm)}$$

- Deviation following long-term loss of synchronization (max. 50 ppm):

$$t_{\text{delta}} = t_{\text{unsync}} \cdot 50 \cdot 10^{-6}$$

9.2.3.7.2.1 Measurement with one device

External synchronization via DCF77, IRIG-B, GPS receiver, NTP or PTP.

Device name	Signal input	Signal output	Synchronization master
T_126678_CS_7008_1	Synchronization signal GPS	no synchronization signal	<input type="checkbox"/>

Settings in the Setup on the page "Devices" in the dialog "Synchronization"

Timed start option: "Immediately"

Synchronous start: activated

Current time 06.10.2016 11:14:09

Start option **Immediately**

Start date 06.10.2016 Start time 11:14:09

Time offset 0 s

Synchronous start

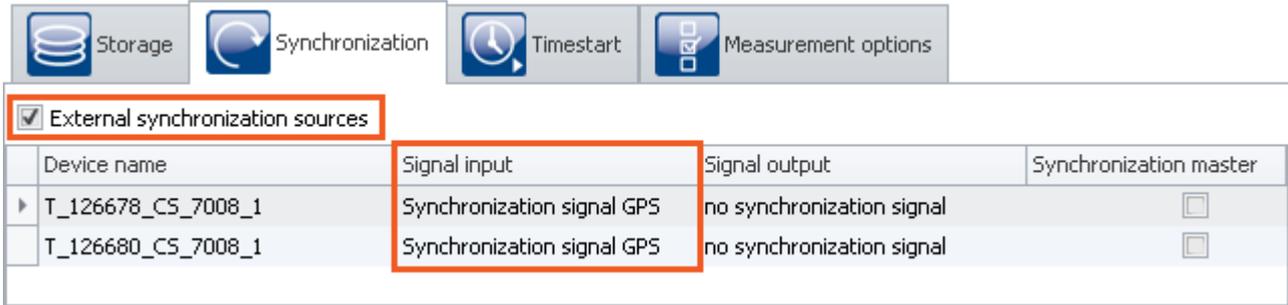
Settings in the Setup on the page "Devices" in the dialog "Timed start"

Scenario	Response	Data storage folder name
Synchronization signal always present.	Device starts and adopts time of the external clock (DCF77, IRIG-B, GPS, NTP or PTP) ^[193]	Time stamp of the devices
Synchronization is always present. The device starts with self-activation upon activation.	Device waits until synchronization and then starts with time of the external clock.	Time stamp of the devices
Synchronization initially present. After measurement start, the clock signal is lost.	Device starts and adopts the time of the external clock. After loss of synchronization, the internal device clock takes over and drifts away*.	Time stamp of the devices
Synchronization is only present after the measurement has been prepared.	Measurement can only start once the device is synchronized. Otherwise the start will not be carried out. Instead, an error message appears to notify of lacking synchronization.	Time stamp of the device or no measurement is set up
Synchronization initially present. After measurement start the clock signal is temporarily lost (e.g. 30s) and is later permanently stable.	Device starts and adopts the time of the external clock. After loss of synchronization, the internal device clock takes over and drifts away*. As soon as the synchronization signal is recognized as valid again, the subsequent system behavior depends on the device type ^[146] : With devices belonging to group A4, only the frequency error ^[220] is corrected. Devices of groups A5 and higher attempt to compensate for the phase error ^[220] .	Time stamp of the devices

* Temporary loss of synchronization can cause the device clocks to drift off by up to 1 ppm. If loss of synchronization continues, the precision may drop as low as 50 ppm in consequence of strong temperature change.

9.2.3.7.2.2 Measurement with multiple devices

Devices are **externally** synchronized **separately**. There is **no** connection via the SYNC terminals.

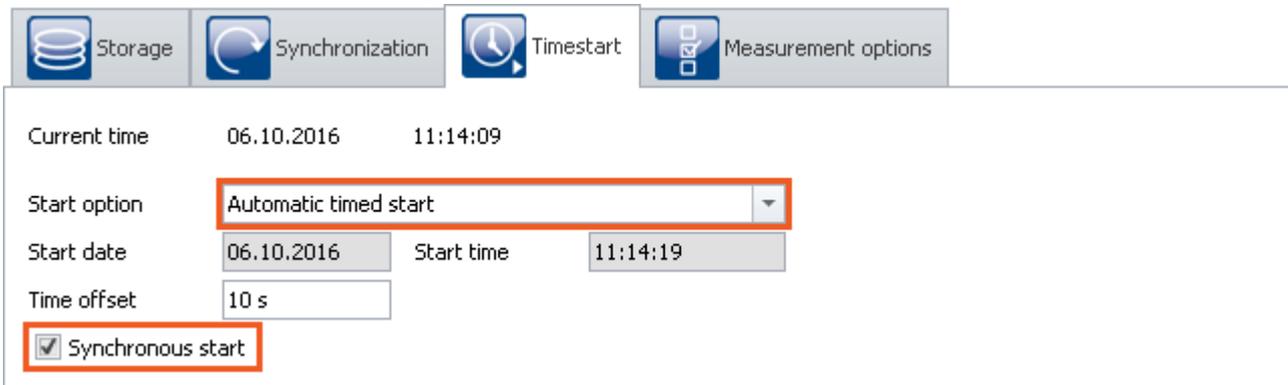


Settings in the Setup on the page "Devices" in the dialog "Timed start"

Synchronous start: activated

1) Timed start option: "Immediately": each unit behaves the same as for "[Measurement with one device](#)"²²⁸ mutually independently.

2) Timed start option: "Immediately" is not selected



Settings in the Setup on the page "Devices" in the dialog "Timed start"

Scenario	Behavior	Data storage folder names
Synchronization signal always present. Device is synchronized	Devices start and adopt the time of the external clock (DCF77, IRIG-B, GPS, NTP or PTP) ^[193]	Time stamp of the devices
Synchronization always present. Devices start with self-activation upon activation.	Devices each wait independently and then start with time of the external clock.	Time stamp of the devices
Synchronization initially present. After measurement started, the clock signal is lost.	Device starts and adopts the time of the external clock. After loss of synchronization, the internal device clock takes over and drifts away*.	Time stamp of the devices
Synchronization is only present after the measurement has been prepared.	Start does not take place. Instead, an error message appears announcing missing synchronization.	No measurement set up
Synchronization initially present. After measurement start the clock signal is temporarily lost (e.g. 30s) and is later permanently stable.	<p>Device starts and adopts the time of the external clock. After loss of synchronization, the internal device clock takes over and drifts away*. As soon as the synchronization signal is recognized as valid again, the system behavior depends on the device type ^[146]:</p> <p>With devices belonging to group A4, only the frequency error ^[220] is corrected.</p> <p>Devices of groups A5 and higher attempt to compensate for the phase error ^[220].</p>	Time stamp of the devices

* Temporary loss of synchronization can cause the device clocks to drift off by up to 1 ppm. If loss of synchronization continues, the precision may drop as low as 50 ppm in consequence of strong temperature change.

9.2.3.7.3 Synchronization control

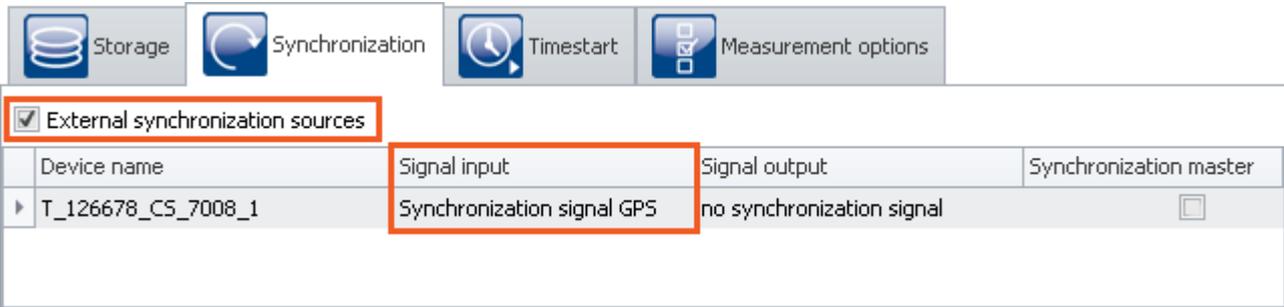
Normally, it's not necessary to stop recording of short measurements if the synchronization is lost. Use the formula presented in the section ["What happens when?"](#) ^[225] to calculate whether the resulting deviation is acceptable for your measurement.

The following functions and constraints must be taken into account for automated synchronization control:

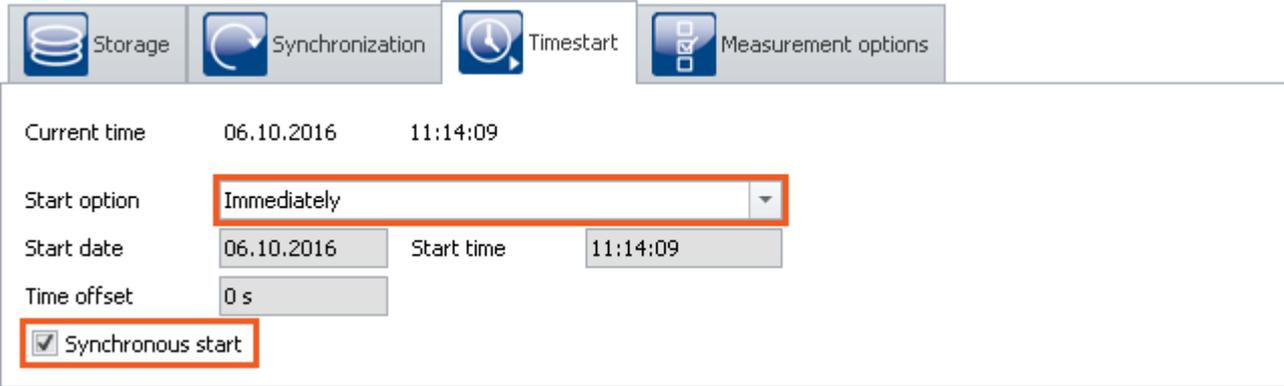
- If a device starts with a synchronized self-activated experiment, the measurement only starts once the synchronization has been established.
- **For devices of the [group A4](#)** ^[146]: After synchronization has been temporarily lost, the measurement must be re-prepared so that the external pulse clock can be received again.
- The synchronization status is requested using the imc Online FAMOS function [IsSynchronized](#).

9.2.3.7.3.1 Measurement controlled by PC

1. Synchronization settings



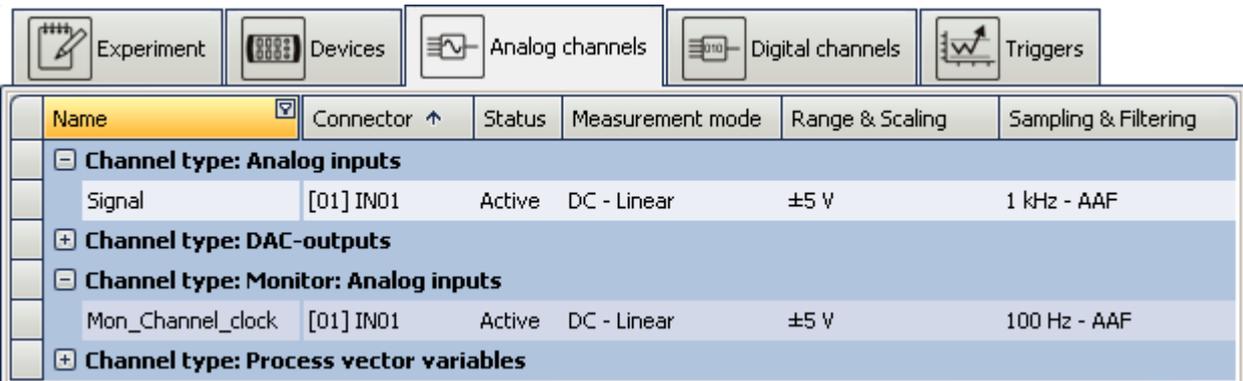
Settings in the Setup on the page "Devices" in the dialog "Synchronization"



Settings in the Setup on the page "Devices" in the dialog "Timed start"

2. Configuration

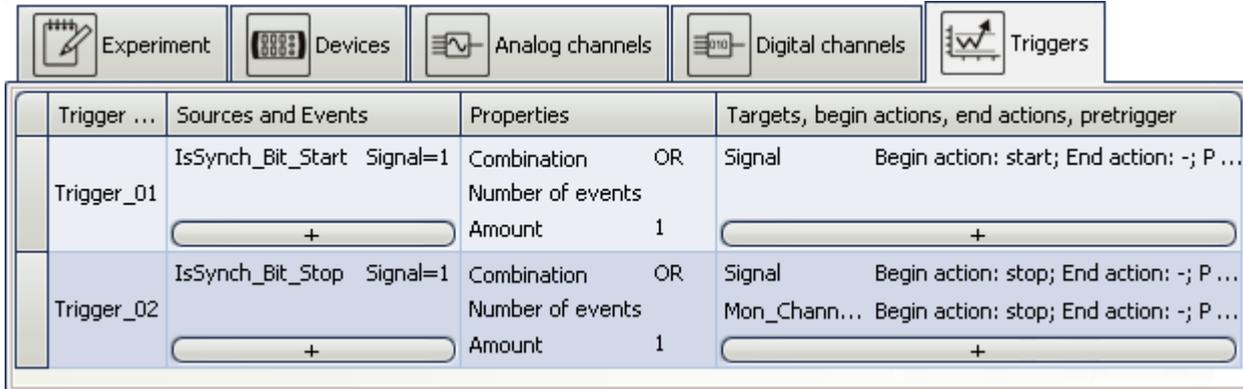
Activate along with the measurement signal to be captured a monitor channel with a sampling interval of 10ms as the clock rate. Rename two virtual bits: *IsSync_Bit_Start* and *IsSync_Bit_Stop*.



Settings in the plug-in Setup on the page "Analog channels"

3. Trigger machine

The two virtual bits display an event at *Signal=1*. The signal to be measured is started by Bit *IsSync_Bit_Start*. The clock pulse channel is started immediately when the measurement begins. Both channels are closed once the virtual bit *IsSync_Bit_Stop* is set.



Trigger machine
Settings in the plug-in Setup on the page "Triggers"

4. imc Online FAMOS

imc Online FAMOS initially evaluates the synchronization signal. The state is displayed by *LED_01* and activates a digital output channel.

A binary channel is generated which is *1* whenever the synchronization signal is not valid. When the channel is integrated into the system, the last value indicates the sum of the unsynchronized times. When a specified time is exceeded, the stop bit is set.

```

; Create control channel for synchronization with 1Hz sampling rate
; Monitor channel:10ms sampling time
IsSync_Bit_Start= IsSynchronized( mean(Mon_Channel_clock, 100, 100) )

LED_01=IsSync_Bit_Start ; Indicator with LED

; Create binary channel with 1Hz if not synchronized
_notSynchron= mean(Mon_Channel_clock, 100, 100)*0+1 - IsSync_Bit_Start

BEEP_01= _notSynchron ; beeper if not synchronized
time_notSynchron= Integral(_notSynchron) ; summarize seconds

; Stop measurement, if syncsignal is not valid for more than 50s
IsSync_Bit_Stop= greater(time_notSynchron, 50)
DOut01_Bit01 = IsSync_Bit_Stop ; Output to digital channel
    
```

9.2.3.7.3.2 Measurement controlled automatically (unmanned)

If the measurement is taking place on a system to which you have no access, a mechanism must be employed to re-start the device.

1. Preparation

- Set up the experiment as in the example [Measurement controlled by PC](#) ²³⁰.
- Save this experiment in the device as an autostart experiment.

2. Activating / deactivating the device via the REMOTE terminal or PowerFail.

Devices having a Remote terminal can be activated / deactivated by means of it. Devices without a Remote terminal are activated / deactivated simply by switching the power supply on / off. In both cases, the running measurement is closed properly before the device goes off.

Example: imc CRONOS-PL

- Activate: Connecting SWITCH and ON switches the device on
- Deactivation: Connecting SWITCH and OFF causes deactivation after 10s'

Example: imc CS-7008

- Put the main switch Position I
- Activate: Switch on the supply voltage
- Deactivate: Switch off the supply voltage; off after 10s

Temporarily switching off the supply ends the measurement if the duration exceeds [Acquisition time \(T1\) after power fail](#)¹⁵³. Upon resumption of the power supply, the device is activated, since the main switch is still in the ON-position. The autostart experiment is then automatically loaded and started once synchronization has been established.

3. Control by means of evaluating the digital output

In imc Online FAMOS, a digital output channel has already been prepared.

```
DOut01_Bit01 = IsSync_Bit_Stop ; Output to digital channel
```

This output must be used by an external control mechanism in order to connect to the Remote terminal or to temporarily switch off the supply voltage.

9.2.3.7.4 Potential difference with synchronized devices

In the present-day devices, the SYNC-terminal is isolated against voltage differentials. This applies to:

- devices belonging to firmware group B and
- devices belonging to firmware group A having a yellow ring on the SYNC-terminal

When using older devices which are synchronized via the SYNC-terminal, ensure that all devices have the same voltage level at the CHASSIS.

9.2.3.7.5 Precision

- The maximum time difference between data from different devices is 1 sampling interval, max 3ms (as long as nothing else is set, such as for instance with NTP).
- In a synchronized start, the devices are started simultaneously. However, the moment when the first sample value arrives can differ. This is due to the system architecture and depends on various factors (e.g. averaging, sampling rate etc.). However, this doesn't affect the data's time stamp. This effect is visible in the curve window when data from different devices are displayed stacked on top of each other. In absolute time display, the samples are displayed in their correct time coordinates, but the signals of the different channels don't all begin at the same time.

9.2.3.7.6 Constraints of the synchronization

1. **Number of devices:** Max. 20 devices via the SYNC-terminal
2. **Cable lengths:** The maximum length of a synch line (SYNC) is 200m.
3. **Start delay:** If channels are being used with very slow sampling rates, this means that the arming procedure will take a proportionally longer time. This applies not only to the first "start" of a measurement, but to each time the measurement is started or triggered. In particular, be careful to account for the start delay when choosing a starting time for an autostart measurement. See "[Breakdown of the Start/Stop and Triggering Behavior](#)"^[482]
4. **Phase:** Please note that the phase of a signal depends on the filter settings. To compare two channels in time, they have to be set by the same filter settings. See also filter settings and synchronization in the hardware manual.
 For devices belonging to the [firmware group A](#)^[146]: Furthermore a long RAM-buffer time and different triggers could result in a time shift. Read also the notes in section "[RAM buffer duration](#)"^[324].

9.2.4 Timed start

In normal cases, the measurement starts when the Start button is pressed. If necessary, the device is prepared beforehand, which delays the measurement's beginning.

In this dialog, you can also **set other "start options"**.

Timed start

Parameter	Description
Start option ^[234]	Select the point in time when a measurement is to be started once the Start-button has been pressed.
Start date / Start time	Calculated possible starting time. Depends on the start option selected.
Time offset ^[234]	Time by which to delay the start in order to ensure that all devices can start simultaneously. Only possible with "Automatic timed start". (Not available for the Diskstart!)
Synchronous start	This option ensures that the start only happens once the devices have been synchronized ^[188] . It does not ensure that the devices start simultaneously ^[235] .

9.2.4.1 Start option

Immediately

The measurement starts as soon as possible after the Start button is pushed. Upon the first start after the preparation process, data acquisition starts exactly at the start of the next full second.

The maximum start delay at the start is 1s. The time needed for [preparing](#)^[149] is not considered here. If it is necessary, this time must be added to the total. This can be preempted by clicking on the Prepare button.

Note

Using the Start option "*immediately*", [simultaneous starting of all devices](#)^[235] is **not possible**. With synchronization activated, simultaneous starting of the devices is **possible with "Automatic timed start"**, for example. With the option "*immediately*", the devices measure synchronously, and are started quickly, but serially.

At defined time

Not available for the Diskstart!

The measurement starts at the time set. "*At defined time*" is a setting only **appropriate to a one-time start**. For each additional start, the clock time must be re-adjusted.

Note

- Each start with the setting "*at defined time*" requires a preparation procedure for the device.
- If you start a measurement **at defined time**, for instance at 5:00 P.M., or at the next turn of the hour, there may be a delay before reading the first measurement value. This delay, caused by special exigencies of the system, depends on the sampling interval selected, and can be up to several seconds in the worst case.

Automatic timed start

Not available for the Diskstart!

Automatic timed start [enables simultaneous starting of all devices](#)^[235] in the experiment. Here, the absolute time of the measurement's start isn't important.

To ensure that all devices are ready, a **time offset** is added to the current time. Devices connected to an RS232 interface or to a modem can have offsets of more than 30s, otherwise, at least 4 s.

Note

imc Online FAMOS - OnInitAll

Every time measurement is started with "*Automatic timed start*", the control command [OnInitAll](#) is run in imc Online FAMOS. Even if the device itself did not need to be prepared.

9.2.4.2 Simultaneous start of all devices

The option "Synchronous Start" ensures that the start only happens once the devices have been synchronized. It does not ensure that the devices start simultaneously.

Simultaneous start of all devices is possible. For this purpose, certain conditions must be met.

- The option: "**Synchronous start**" must be activated.
- Appropriate "**Start options**"^[234] must be selected. E.g. "Automatic timed start".
- For all devices, a **clock signal for the signal input**^[188] must be selected.

Starting is possible even without this conditions being met, but in that case, all the devices will start serially. Channels of synchronized devices then have the correct absolute time, but the triggering time is a bit different for each device. With a curve window having the X-axis denoted in seconds, the time 0s begins with the release of the trigger. Only in the display mode with the X-axis expressed in absolute time are the channels belonging to different devices all displayed in the correct times.

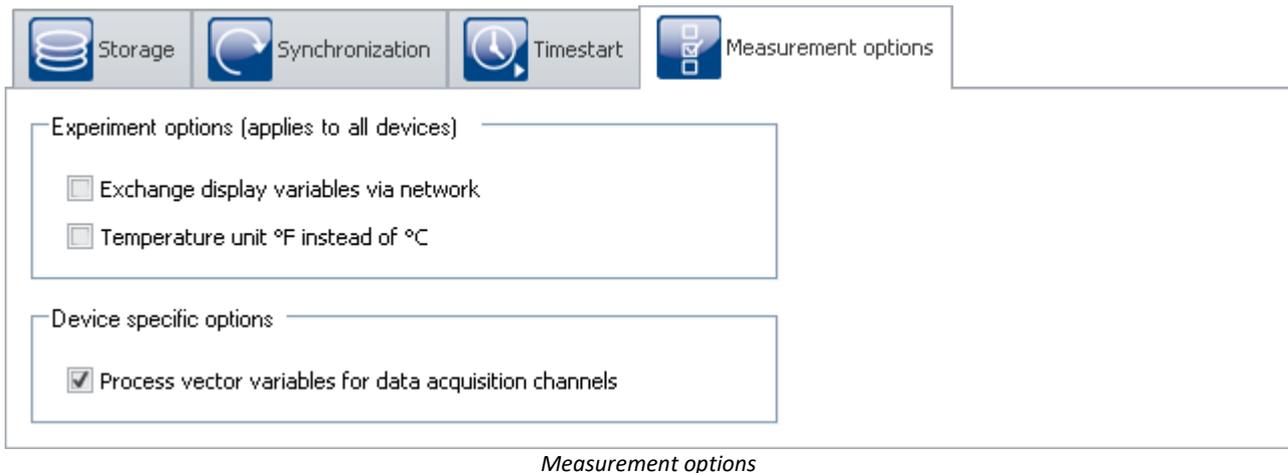


Reference

Regarding synchronization, observe the notes in the manual:

"Setup pages - Configuring device" > "Configuring Devices" > "**Synchronization**"^[188] > "**What happens when?**"^[225]

9.2.5 Measurement options



Exchange display variables via network

It is possible to exchange the display variable values via the network whenever they change. By this means, data from channels of different devices can be subjected jointly to calculation operations.

 [Reference](#)

See also:

[Exchanging values among the devices](#) ⁴⁸⁷

Temperature unit °F instead of °C

With this option, you set the default unit for all temperature channels.

 **Note**

Make sure that temperature measurements performed via Fieldbusses are set separately. All temperature channels are stated in the unit set here. If, for instance, temperatures in °C are captured via CAN-Bus, but °F is set here, the °C-temperatures from the CAN channels are stated in the unit °F without conversion.

Process vector variables for data acquisition channels

For devices belonging to the [firmware group A](#) ¹⁴⁶ (imc DEVICES)

For data acquisition channels, [process vector variables](#) ²³⁹ (pv-variables) are created. This option is activated by default.

 **Note**

[Aggregate sampling rate](#)

Even when the pv-variables are not used, they demand a portion of the device's available power. For this reason, less than the entire [aggregate sampling rate](#) ²⁶⁷ of your device is actually usable.

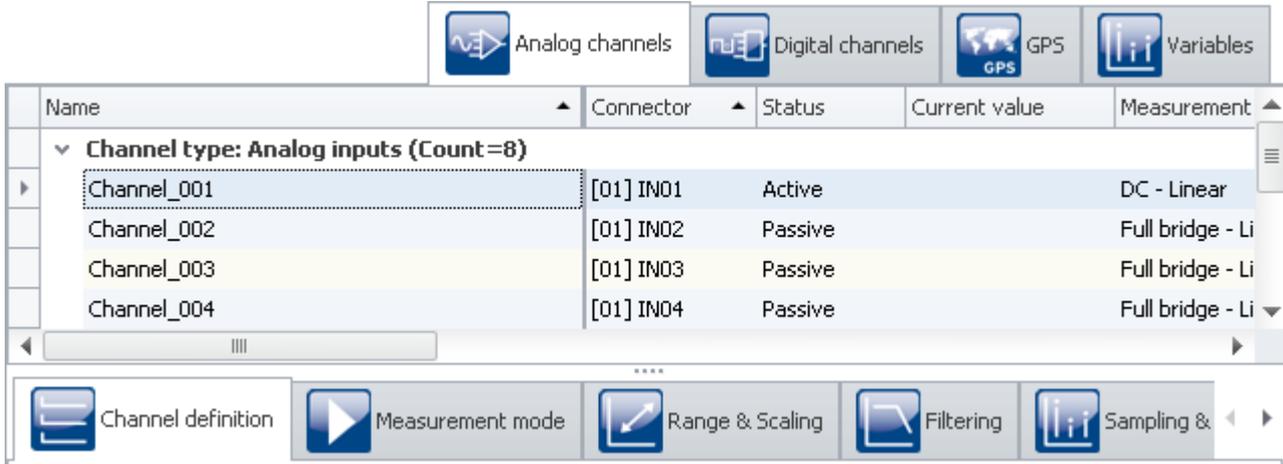
For each device, you can deactivate pv-variables for the data acquisition channels. Remove the checkmark from the [selected](#) ¹⁶² device. This makes the entire aggregate sampling rate available. Additional pv-variables (e.g. GPS data) are not affected.

For devices belonging to the [firmware group B](#) ¹⁴⁶ (imc DEVICEScore)

Deactivation is not possible.

9.3 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](#)^[171]. 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 ^[242]	Name and Status of the channel.
Measurement mode ^[247]	Fundamental configuration settings such as measurement mode , measurement range , coupling , power supply , ...
Range & Scaling ^[249]	Configuration settings such as measurement range , unit , scaling : Factor/Offset-scaling and 2-point scaling, e.g. for entries from the spec sheet.
Strain gauges ^[252]	Configuration of the strain gauge parameters . Only for devices having strain gauge channels.
Encoder ^[254]	Configuration of the incremental counter channels . Only for devices having an incremental counter modul.
Filtering ^[258]	Configuration of the input filters : AAF, low-pass, high-pass, bandpass.
Sampling & Preprocessing ^[261]	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 ^[268]	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 ^[270]	Refresh rate and data saving rate for histograms and other matrices.
Curve properties ^[270]	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](#)", you will find descriptions of the most important additional parameters.

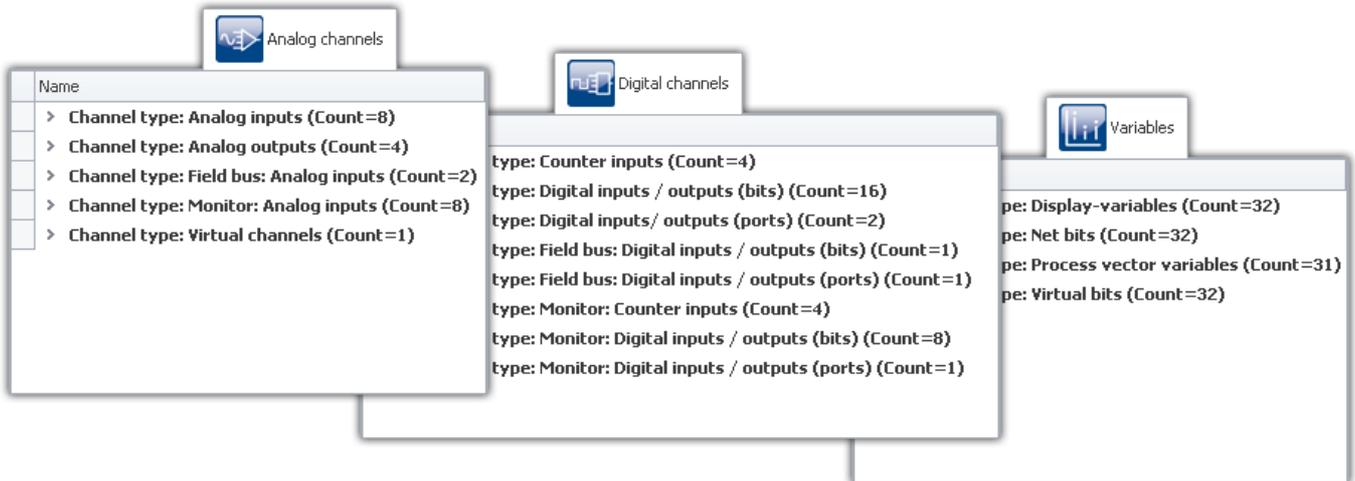
Reference

See also additional info on:

- Operation and configuration of the [tabular display](#) and the [dialogs](#)

9.3.1 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](#) directly, or open the [dialog](#) 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>	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> ⁷³⁵	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 ⁷⁴⁵ (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 ⁷³⁷	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 ⁷⁴¹	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) (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					
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	512	Channel	240	198	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		-	-	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

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")

9.3.1.1 imc CRONOSflex and CRONOS-XT modules

The device software lists the imc CRONOSflex (CRFX) and imc CRONOS-XT (CRXT) module's amplifier channels automatically in the channel table.

Information on the maximum sampling rate per channel and on the maximum aggregate sampling rate per system, as well as on other possible limitations are described in the technical specs in the manual for the respective device.

At present, there is no **Preprocessing** (e.g. Arith. Mean) possible for CRFX/CRXT amplifier channels. Channels, recorded with the modules, can be processed with imc Online FAMOS functions only (Mean(), Min(), Max(), RMS())

Connection recognition

The numbering of the module happens automatically upon first connection and remains intact. This number is displayed at the module with a 7-segment display. It appears in the channel table in the column **Connector** as slot recognition in parentheses in front of the amplifier's channel number (e.g. **[01]** IN07, **[01]** IN08, **[02]** IN01).

An overview of the modules connected is provided in the dialog "[Module properties](#)¹⁵⁸". A CRFX/CRXT module is distinguished by the address designation (*xbus...*).

Reference

Observe the notes on **Removing and exchanging modules** and on **Re-assignment of module numbers** in the description of the dialog: "[Module properties](#)¹⁵⁸".

Note

Digital CRFX modules DIO, Pulse Counter and DAC

- 32-channel DI- or DO-modules are designed as "double-modules" which behave logically as two modules and which show corresponding IDs on two 7-Segment displays.
- The maximum count of modules per device is limited. For more information on this subject, see the device's technical specs and the chapter "[Channel table](#)²⁴⁰".
- The 7-Segment display on the front of the modules shows the assigned address in HEX-format with a prefixed "d" for "digital module". Digital Flex modules which are installed within a base unit do not have a 7-Segment display and are assigned the address "d0" at factory.
- See also the description in the user's manual imc CRONOS System Family, in the chapter "*DI, DO, DAC, HRENC and SYNTH*".

9.3.2 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

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>
	<p>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.</p> <p>See "Channel Name".</p>		
Comment	<i>Channel comment</i>	<i>Comment</i>	<i>eChannelComment</i>
	<p>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.</p>		
Status	<i>Channel status</i>	<i>Status</i>	<i>eStatus</i>
	<p>The Status determines whether the particular channel is activated for the measurement/the experiment.</p> <ul style="list-style-type: none"> 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>
	<p>Pre-defined setting for the channel-color in the curve window.</p>		
Sensor	<p>Read sensor information. The sensor information is read out and used to configure the channel. see "TEDS - Sensors".</p>		

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.

9.3.2.1 Channel name

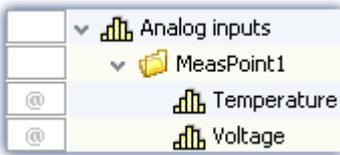
Note Using special characters in the name

If a channel name begins with a number or contains special characters such as "\?+!" or a space, in imc Online FAMOS the channel name must be set into curly braces.

Example:

Channel name = "123 My channel 100% display"
 in imc Online FAMOS = "{123 My channel 100% display}"

Joining variables to form a group



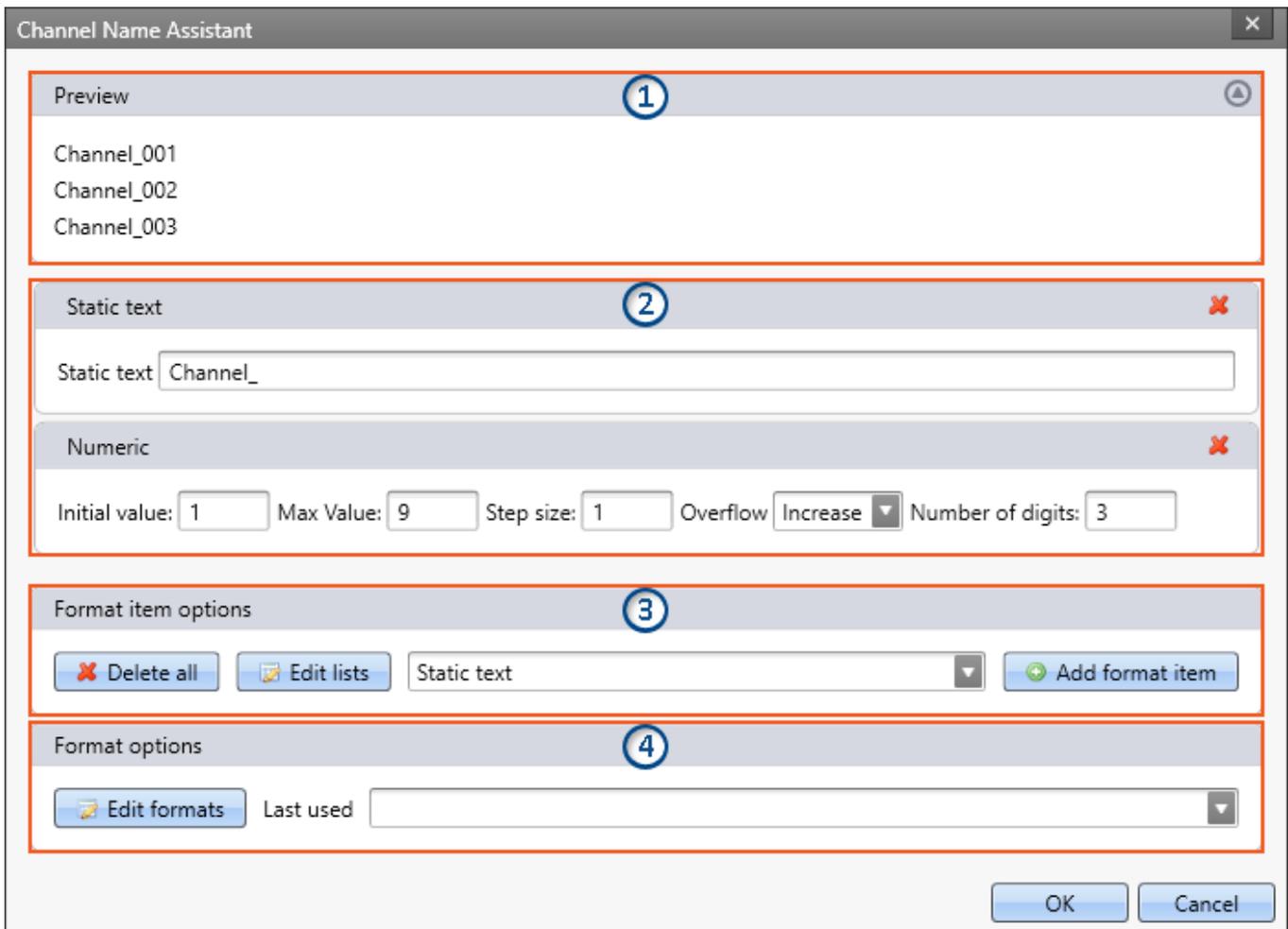
Display in the Data Browser

To join multiple variables together in a group, you can use the character '.' in the variable's name.

E.g. "MeasPoint1.Temperature", "MeasPoint1.Voltage" results in this display in the Data Browser.

Renaming multiple channels jointly (Channel Name Assistant)

By means of the "Channel Name Assistant", you can rename multiple channels simultaneously. For this purpose, there are a variety of naming formats available, such as numbering and text-lists.



Example with the Assistant's standard settings

As a result, the first channel is assigned the name "Channel_001", the second "Channel_002", ...

Area	Description
1 Preview	Here you find the first three results with the format items which are set.
2 Format items set	<p>Here, all format items are defined. Configure them here also.</p> <p>From the top downward, the items are applied to the names. You can use Drag&Drop to modify the order. In the example above, the static text: "Channel_" is applied first. Next comes the (three digit) number: "Channel_001".</p> <p>Using the red "x" icon, you can delete an item.</p>
3 New format items	<p>Create here any new format items. Select from the list an appropriate item and add it. It will appear on the bottom.</p> <p>Additionally you can edit lists and add to them. Lists are used in the item of the same name in order to use user-defined texts for the name.</p> <p>Use "Delete all" to delete the current configuration. All items are deleted.</p>
4 Saved formats	Here you can access the most recent configurations used . By means of "Edit formats", you can assign names to these configurations or delete them.

Configuring format items

Format item	Description
Static text	The text defined is adopted as it is entered. It can also be used for special characters as separators for other items, e.g. "_".
Numeric	<p>A consecutive number.</p> <ul style="list-style-type: none"> • Initial value: Starting value • Max value: Has a function only in conjunction with rotation. See "Overflow-Rotation" • Step size: By what increments the number is to be increased. E.g. "1": 1,2,3,4,... "2": 1,3,5,7,... • Overflow-Rotation: The numbers extend from the Initial value to the maximum value. After that, the number starts over at the Initial value. • Overflow-Increase: Applies the usual counting method. If one decimal place reaches the digit "9", the next higher decimal place is used: "10" • Number of digits: Minimum number of decimal places used for the number. E.g. "2": 01, 02, 03, ..., 99, 100, 101.
Alphabetical	<p>The consecutive alphabet.</p> <ul style="list-style-type: none"> • Initial value: Starting letter • Max value: Has a function only in conjunction with rotation. See "Overflow-Rotation" • Step size: By what increments the letter is to be increased. E.g. "1": A,B,C,D,... "2": A,C,E,G,... • Overflow-Rotation: The letters extend from the Initial value to the maximum value. After that, the letter starts over at the Initial value. • Overflow-Increase: If one position reaches the letter "Z", the next position is used: "AA"

Format item	Description
List	<p>Uses a defined list for the names.</p> <ul style="list-style-type: none"> • List: Here, select a list to use. If applicable, create a list first ("<i>Edit lists</i>"). • Initial value: First list entry used. • Step size: The interval at which to progress through the list. E.g. "2": first entry, third entry, ... • Overflow-Rotate: The list entries extend from the Initial value to the last entry. Subsequently, progress continues with the first entry (not the Initial value!). • Overflow-Increase: Once the last entry has been used, the next position is used: "Entry1Entry1", "Entry1Entry2", ...
Column value	<p>Uses a channel parameter as the name.</p> <p>Possible choices: Device name, Channel name, Channel comment, Connector, Module number, Device SN.</p> <p>Example: Information regarding the sensor connected is to be used in the name. When the sensor is read, the associated info is recorded in "<i>Channel comment</i>". This info can now be additionally used in the channel name.</p>



Example

Combination of items

The name is to be composed of a measurement point number. Each measurement point has three connectors: "Temperature", "Voltage" and "Current".

Previously, a list ("My List") had been defined, containing the three connector designations.

The screenshot shows a configuration interface with four stacked panels, each with a close button (red X) in the top right corner:

- Static text:** The text field contains "MeasPoint_".
- Numeric:** Initial value: 1, Max Value: 3, Step size: 1, Overflow: Rotate (dropdown), Number of digits: 1.
- Static text:** The text field contains "_".
- List:** List: My List (dropdown), Initial value: Temperature (dropdown), Step size: 1, Overflow: Rotate (dropdown).

This configuration generates the following names of variables:

- MeasPoint_1_Temperature
- MeasPoint_1_Voltage
- MeasPoint_1_Current
- MeasPoint_2_Temperature
- MeasPoint_2_Voltage
- MeasPoint_2_Current
- MeasPoint_3_Temperature
- MeasPoint_3_Voltage
- MeasPoint_3_Current

9.3.3 Measurement mode


Measurement mode

Channel name	Channel_001		
Mode	<input type="text" value="Voltage"/>	Coupling	<input type="text" value="DC"/>
Correction	<input type="text" value="Linear"/>	Bridge resistor	<input type="text" value=""/>
Supply	<input type="text" value="5 V"/>		
Input range	<input type="text" value="±5 V"/>	Wiring	<input type="text" value="Differential"/>

Isolated thermo couple

Measurement mode

Reference

The settings available depend on the amplifier type. They are described in the pertinent device manual in conjunction with the interconnections.

Parameter	Description		
	long name	short name	column ID
Mode	<i>Measurement mode</i>	<i>Mode</i>	<i>eChannelMode</i>
	Basic settings for the measurement types: "Voltage", "Current" or "Strain gauge" (Strain gauge: for further settings, a new dialog appears: Strain Gauge ^[252]).		
Coupling	<i>Coupling</i>		<i>eCoupling</i>
	Depending on the amplifier and on the mode set, DC- or bridge measurement settings are possible. If the Strain Gauge mode is selected, a variety of possible applications is listed, which lead to different input ranges ^[249] .		
Coupling is DC	<i>Coupling is DC</i>	<i>DC</i>	<i>eCouplingIsDC</i>
	For some amplifiers, it is necessary to switch the coupling for the whole amplifier board to DC-coupling. In this case, this must be set independently of the Coupling .		
Bridge resistor	<i>Bridge resistor</i>	<i>Resistor</i>	<i>eBridgeResistor</i>
	The bridge resistor must be stated for all bridge types, although it is actually only absolutely necessary for the quarter bridge.		
	For full bridges, the bridge resistor is used along with the calibration resistor to calculate the indicated strain (see: " Cable compensation without sense line " ^[282]).		
	Also, with the bridge resistor, it is possible to determine the cable resistance even if no Sense line is connected.		

Parameter	Description		
	long name	short name	column ID

Correction	Correction		eCorrection
------------	------------	--	-------------

Use of a correction curve (e.g. temperature measurement; depending on the amplifier). Requires the amplifier to operate in DC-voltage mode.

- *Linear*: No linearization of measured values
- *PT100*: Linearization according to a PT-100 thermistor characteristic curve
- *Type R, S, B, J, T, E, K, L, N, C¹*: Linearization according to a thermocouple characteristic curve
- *Sensor characteristic curve*: This is displayed when a [sensor](#)⁴⁶⁵ is used

¹ For amplifiers produced after November 2012.



The condition for **temperature measurement** is that the amplifier is working on mode: voltage (DC).

Supply	Supply		eSupply
--------	--------	--	---------

The power supply voltage can be used either for the bridge supply or the sensor supply. Each amplifier has its own power settings globally applicable to all its channels. By changing the settings for one channel, the settings of all channels belonging to the same amplifier are changed accordingly.

Some amplifiers provide the ability to switch the power supply on or off. See parameter: [Bridge supply \(On/Off\)](#)²⁷³

Input range	Input range		eRange
-------------	-------------	--	--------

Lists the input ranges available.

If you use scaling for the channel or if you have carried out an adjustment, the resulting physical input range with its unit is displayed here. For temperature measurements the input ranges depend on the thermocouple type or RTD.

Input ranges rounded



Use of taring or the entering of a value for the "Sensitivity" easily cause the generation of input ranges having very many decimal places. The display of the input ranges is rounded off to 5 significant digits. Parameter sets and channel parameters contain the original value.



See also: [Setting the expected signal value range in addition to the measurement range](#)²⁷¹

Wiring	Channel wiring	Wiring	eWiring
--------	----------------	--------	---------

The input circuit configuration

- **single ended:**
The channel signal is measured relative to ground. All channels using this setting share a common ground.
- **differential:**
The channel is measured differentially.

Parameter	Description		
	long name	short name	column ID
Isolated thermo couple	<i>Isolated thermo couple</i>		<i>eIsolatedThermoCouple</i>
	Default is isolated thermocouple. When using none isolated thermocouples, that option has to be deactivated, to avoid ground loops.		
Polarization	<i>Polarization</i>		<i>ePolarization</i>
	Activates the power supply for certain microphone types. Please observe the notes on the respective amplifiers.		

9.3.4 Range & Scaling

The screenshot shows the 'Range & Scaling' configuration window for 'Channel_001'. The main window has 'Input range' set to $\pm 5\text{ V}$ and 'Unit' set to 'V'. The 'Factor-offset scaling' is set to '2-point scaling'. A sub-dialog titled 'Factor-offset scaling' shows 'Factor' and 'Sensitivity' both set to '1 V/V' and 'Offset' set to '0 V'. A further sub-dialog shows two columns: 'Acquired' and 'Scaled'. Under 'Acquired', 'Point 1' is '0 V' and 'Point 2' is '1 V'. Under 'Scaled', 'Point 1' is '0 V' and 'Point 2' is '1 V'.

Factor-offset scaling

Parameter	Description		
	long name	short name	column ID
Input range	<i>Input range</i>		<i>eRange</i>

Lists the input ranges available.

If you use scaling for the channel or if you have carried out an adjustment, the resulting physical input range with its unit is displayed here. For temperature measurements the input ranges depend on the thermocouple type or RTD.

Input ranges rounded



Use of taring or the entering of a value for the "Sensitivity" easily cause the generation of input ranges having very many decimal places. The display of the input ranges is rounded off to 5 significant digits.

Parameter sets and channel parameters contain the original value.

Parameter	Description		
	<i>long name</i>	<i>short name</i>	<i>column ID</i>



See also: [Setting the expected signal value range in addition to the measurement range](#)

Unit	<i>Unit</i>		<i>eUserUnit</i>
------	-------------	--	------------------

Unit of measurement values.

Select SI units without milli, micro etc., which the curve window will do automatically.

Factor	<i>Scaling factor</i>	<i>Factor</i>	<i>eUserScalingFactor</i>
--------	-----------------------	---------------	---------------------------

Here, you can enter a scaling factor, in order to perform a conversion of the measured value to a physical quantity. The system will adjust the input range accordingly.

$$\text{Factor} = 1/\text{Sensitivity}$$

Sensitivity	<i>Sensitivity</i>		<i>UserScalingFactor_Reciprocal</i>
-------------	--------------------	--	-------------------------------------

Here you can enter a value for the sensitivity for the purpose of converting a measured value to physical quantities. The system will adjust the input range accordingly.

$$\text{Sensitivity} = 1/\text{Factor}$$

Offset	<i>Scaling offset</i>	<i>Offset</i>	<i>eUserOffset</i>
--------	-----------------------	---------------	--------------------

If your signal is distorted by a DC component which cannot be compensated by the amplifier, the possibility exists to mathematically cancel this offset. In that case, the measurement signal is shifted by the amount of the offset entered.

Example:

If the signal hovers around a level of +2.5 V, an offset of -2.5 V shifts the signal towards the X-axis. Since the physical input range hasn't changed, the displayed range is also shifted by the offset amount. For the example of 2.5 V, a input range of previously ± 10 V becomes a displayed input range of -12.5 V to +7.5 V.

To scale the measured values as a physical quantity, you can perform **factor-offset scaling** or **2-point scaling**. The input range is adapted accordingly.

Option	Description
Factor-offset scaling	Here you can enter a factor or the sensitivity (=1/factor) and an offset. $\text{Physical quantity} = \text{measured value} * \text{factor} + \text{offset}$
2-point scaling	Here you can assign physical pond to two measured values.



Example

Example of Factor-Offset-scaling

A bridge amplifier is operated in the 100 mV/V input range. A displacement sensor is connected which unbalances the bridge by 1.6 mV/V for one millimeter of displacement. Toward this end, enter 1.6e-3 for **Factor** and for the unit m, and the input range changes to ± 0.16 m. Also, if you enter for **Offset** a value of 5e-3, the input range changes accordingly to -155 mm ...165 mm.

**Example****Example for 2-point scaling**

A pressure sensor returns a current of 4-20 mA to correspond to inputs of 0-10 bar. The current signal is captured via a connector with a 50 Ω shunt resistor. The resulting voltage is thus:

0 V to 10 bar corresponds to 4 mA * 50 Ω to 20 mA * 50 Ω = **0.2 V to 1 V**

- For **Acquired**, enter the resulting voltage values,
- and for **Scaled**, the corresponding pressure values.

Point 1: 0.2 V corresponding to 0 bar

Point 2: 1 V corresponding to 10 bar

**Note**

Higher precision can be achieved by performing two-point scaling on the amplifier (see: [Two-point scaling](#)^[281]). By means of this procedure, all uncertainty tolerances are taken into account in the scaling. However, it is possible that the values scaled by the amplifier are additionally combined with the scaling described here.

**Note****Notes on hardware: balancing with setted offset**

Zeroing with adjusted offset leads to different results with CRFX/CRXT modules and amplifiers installed in imc CRONOScompact, imc C-SERIES or imc SPARTAN:

- CRFX/CRXT: [Zeroing](#)^[275] adjusts the channel in the physical unit, e.g. 0 bar. This applies to all modules for the acquisition of analog voltages with 16 or 24 bits, except UNI-4.
- imc CRONOScompact, imc C-SERIES, imc SPARTAN, CRFX/UNI-4: The electrical value is balanced, i.e. the entered offset is retained. After the adjustment, the physical value is displayed with a negative offset. This ensures that an offset entered subsequently is not overwritten. If you wish for the balancing to determine the offset, **only enter the factor**, and then subsequently balance the offset by means of the [Balancing function](#)^[275]. Adaptation to the behavior of CRFX/CRXT is in preparation.

**Reference****See also**

Calibration of IEPE sensors

9.3.5 Strain gauge

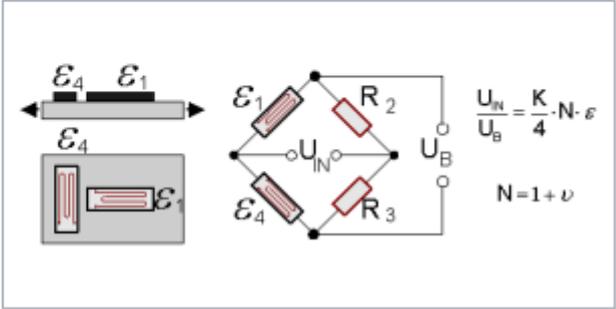
Only enabled for channels in the mode: **Strain gauge**.

To configure the strain gauge, you must set the **Mode** to *Strain gauge* in the dialog **Measurement mode**. A new dialog **Strain gauge** appears, as long as the channel is selected in the channel table.

 Strain gauge

Channel name	Channel_001		
Coupling	Poisson half bridge		
Resistor	120 Ω		
Mode	Strain		
Bridge factor N	1 + ν		
Gauge factor	2		
Unit	μ eps		
Transverse strain coeff. ν	0.3		
Modulus of elasticity E			
Input range	±-1600000 μ eps		

Half bridge with 2 active strain gages. One strain gage along main expansion, the other transverse to it. Exploits transverse contraction while providing good temperature compensation.



Parameter	Description		
	long name	short name	column ID
Coupling	<i>Coupling</i>		<i>eCoupling</i>
	<p>Lists various usage possibilities. Corresponding descriptions and sketches of the strain gauge are displayed.</p> <p>The strain gauge is pre-set to conform to the coupling-selection. It is also still possible to make appropriate settings.</p>		
Resistor (Bridge resistor)	<i>Bridge resistor</i>	<i>Resistor</i>	<i>eBridgeResistor</i>
	<p>The bridge resistor must be stated for all bridge types, although it is actually only absolutely necessary for the quarter bridge.</p> <p>For full bridges, the bridge resistor is used along with the calibration resistor to calculate the indicated strain (see: "Cable compensation without sense line"^[282]).</p> <p>Also, with the bridge resistor, it is possible to determine the cable resistance even if no Sense line is connected.</p>		
Mode	<i>Bridge mode</i>	<i>Mode</i>	<i>eBridgeMode</i>
	<i>Strain or Stress.</i>		
Bridge factor N	<i>Bridge factor N</i>		<i>eBridgeN</i>
	The available selection depends on the coupling.		
Gauge factor K	<i>Gauge factor</i>		<i>eBridgeFactor</i>
	Depends on the part's material		

Parameter	Description		
	long name	short name	column ID
Unit	<i>Bridge unit</i>	<i>Unit</i>	<i>eBridgeUnit</i>
<p>Unit of measurement values. For the mode Strain, the following units are available: $\mu\text{ eps}$ and $\mu\text{m/m}$; ($1\ \mu\text{m/m} = 1\ \mu\text{ eps}$)</p> <p>For the mode Stress, the following units are available: MPa and GPa and N/mm^2 and PSI (pound-force per square inch) ($1\ \text{MPa} = 1\ \text{N/mm}^2$; $1000\ \text{MPa} = 1\ \text{GPa}$; $1\ \text{MPa} = 145,04\ \text{PSI}$)</p>			
Transverse strain coeff. ν	<i>Transverse strain coeff. ν</i>		<i>eBridgeEps</i>
<p>Depends on the part's material</p>			
Modulus of elasticity E	<i>Modulus of elasticity E</i>		<i>eBridgeEModule</i>
<p>Depends on the part's material</p>			
Input range	<i>Input range</i>		<i>eRange</i>
<p>Lists the input ranges available.</p> <p>If you use scaling for the channel or if you have carried out an adjustment, the resulting physical input range with its unit is displayed here. For temperature measurements the input ranges depend on the thermocouple type or RTD.</p> <hr/> <p>Input ranges rounded</p> <p> Use of taring or the entering of a value for the "<i>Sensitivity</i>" easily cause the generation of input ranges having very many decimal places. The display of the input ranges is rounded off to 5 significant digits. Parameter sets and channel parameters contain the original value.</p> <hr/> <p> See also: Setting the expected signal value range in addition to the measurement range^[271]</p>			

9.3.6 Encoder (incremental counter inputs)

Only visible for devices with incremental counter inputs.

Certain properties are dependent on the hardware and are described in the respective device's manual. There, you will also find background information on the module's technical workings.

 Encoder

Channel name Counter_001

Measurement mode	<input type="text" value="Distance (diff.)"/>	Signal	<input type="text" value="One signal"/>
<input checked="" type="checkbox"/> Encoder w/o zero impulse	Scaling factor	Start edge	<input type="text"/>
	Maximum	Stop edge	<input type="text"/>
Input range	Switching level	Unit	<input type="text" value="m"/>
Signal shape	Hysteresis	Scaling offset	<input type="text" value="0 m"/>

Parameter	Description		
	long name	short name	column ID

Measurement mode	<i>Measurement mode</i>	<i>Mode</i>	<i>eChannelMode</i>
------------------	-------------------------	-------------	---------------------

Measurement types. See [Measurement mode - Overview](#) ²⁵⁶

Encoder without zero impulse	<i>Encoder w/o zero impulse</i>		<i>eNullImpuls</i>
------------------------------	---------------------------------	--	--------------------

The **zero impulse** starts the incremental counter inputs' logic unit. Measured values are only recorded once an event has occurred at the **index channel**. If measurement without a zero pulse is selected, measurement begins directly after preparation of the measurement is complete.

Signal	<i>Counter signal</i>	<i>Signal</i>	<i>eSignal</i>
--------	-----------------------	---------------	----------------

Single-track encoders or Dual-track encoders

- **One signal:** Single-track encoders capture a sequence of impulses without a specified direction. The value is always positive, as for example a frequency or (typically) an RPM-value.
- **Two signal:** Dual-track encoders return two impulse sequences, mutually offset by 90°. In this way, the counter can distinguish in which direction the sensor is moving. This type is especially suited to measurements of velocity, distance and angle.

Please observe the notes on **two-point scaling** in the section "[Single-track / Dual-track encoder](#)" ³⁶⁴.



Affects both the devices belonging to the imc C-SERIES, and also any devices belonging to the imc SPARTAN and imc CRONOS families which are equipped with the digital multiboard: DI16-DO8-ENC4 or the DI8-DO8-ENC4-DAC4.

Scaling factor	<i>Counter scaling factor</i>	<i>Scaling factor</i>	<i>eUserScalingFactorENC</i>
----------------	-------------------------------	-----------------------	------------------------------

How many pulses represent one physical unit depends on the measurement mode set.

Parameter	Description		
	<i>long name</i>	<i>short name</i>	<i>column ID</i>
Maximum	<i>Maximum</i>		<i>eMaximum</i>
<p>Maximum value or input range end value</p> <p>The measured value should lie within the upper third of the range. The resolution results from the maximum divided by 2^{16}, or 2^{15} for dual-track encoders.</p>			
Start edge	<i>Counter start edge</i>	<i>Start edge</i>	<i>eStartEdge</i>
<p>Start edge on time and pulse measurement. Depends on the measurement mode ²⁵⁶.</p>			
Stop edge	<i>Counter stop edge</i>	<i>Stop edge</i>	<i>eStopEdge</i>
<p>Stop edge on time and pulse measurement. Depends on the measurement mode ²⁵⁶.</p>			
Input range	<i>Counter input range</i>	<i>Input range</i>	<i>eInputRange</i>
<p>This setting is available only for certain incremental counters. For information on this subject, refer to the material on the respective incremental counters in the device manual.</p>			
Signal shape	<i>Counter signal shape</i>	<i>Signal shape</i>	<i>eSignalform</i>
<p>This setting is available only for certain incremental counters. For information on this subject, refer to the material on the respective incremental counters in the device manual.</p>			
Switching level	<i>Counter switching level</i>	<i>Switching level</i>	<i>eLevel</i>
<p>Here you set the lower voltage limit interpreted as the HIGH level.</p>			
Hysteresis	<i>Counter hysteresis</i>	<i>Hysteresis</i>	<i>eHysteresis</i>
<p>The hysteresis ensures that duplicate impulses are not counted in the case of noisy signals. These two settings are used to derive the signal levels as follows:</p> <p>HIGH = Signal > Switching level LOW = Signal < (Switching level - Hysteresis)</p>			
Unit	<i>Unit</i>		<i>eUserUnit</i>
<p>Measurement value unit. Depends on the measurement mode set.</p>			
Scaling offset	<i>Scaling offset</i>		<i>eUserOffset</i>
<p>Here you can enter an offset.</p>			

 Note
Low-pass filter

Set the low-pass filter accordingly (see "[Filtering](#)"^[258]). The Low-pass ensures that peaks in the signal do not lead to duplicate counts.

Make sure that the frequency of the useful signal is not below the low-pass frequency set. Otherwise, your useful signal will be filtered smooth.

Sampling times/rates

For all active incremental counter channels (and associated active Monitor channels) of one module (e.g. CRFX) or a device (e.g. C-SERIES), only one sampling interval/-rate may be set.

9.3.6.1 Measurement mode - overview
 Reference

More information about the measurement modes are described in the section: "[Measurement types](#)" > "[Incremental Counters Channels](#)"^[359] or in the respective device manual.

As a rule, what type of measurement is performed is determined by your setting for the **Measurement mode**:

Event counting

Measurement mode	Description
Events	Number of events within a sampling interval
Distance (differential)	Path traveled within one sampling interval. For this purpose, the number of pulses per meter must be entered. The absolute distance is not directly measurable but must be calculated in imc Online FAMOS.
Angle (differential)	Angle traveled within one sampling interval. For this purpose, the number of pulses per revolution must be entered. The absolute angle can be calculated in imc Online FAMOS or determined by the mode Angle (abs).
Angle (absolute)	Absolute angle. The differential angle measurement is converted to the absolute angle. By taking the zero impulse (the counter with no zero impulse should not be selected) into account, the absolute angle position is determined and indicated. Otherwise, the angle value is assumed to be 0° when the measurement begins.

Time measurement

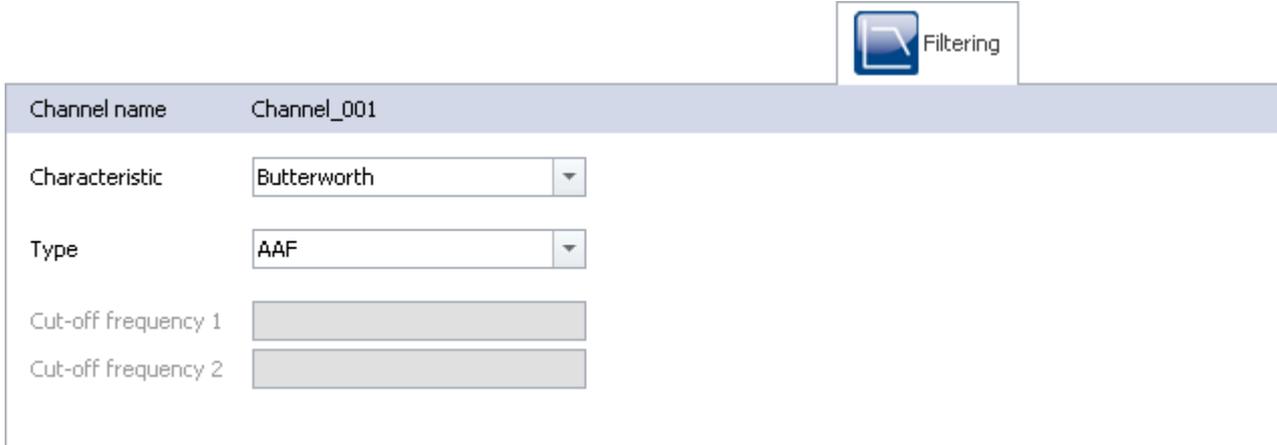
Measurement mode	Description																
Time	<p>The time between two signal edges is determined. For this purpose, the settings options for the Start and Stop of the measurement appear. There are several possibilities:</p> <table border="1"> <tr> <td>positive edge</td> <td>></td> <td>negative edge:</td> <td>↑ > ↓</td> </tr> <tr> <td>negative edge</td> <td>></td> <td>positive edge:</td> <td>↓ > ↑</td> </tr> <tr> <td>positive edge</td> <td>></td> <td>positive edge:</td> <td>↑ > ↑</td> </tr> <tr> <td>The combination negative edge</td> <td>></td> <td>negative edge:</td> <td>↓ > ↓ is not allowed</td> </tr> </table>	positive edge	>	negative edge:	↑ > ↓	negative edge	>	positive edge:	↓ > ↑	positive edge	>	positive edge:	↑ > ↑	The combination negative edge	>	negative edge:	↓ > ↓ is not allowed
positive edge	>	negative edge:	↑ > ↓														
negative edge	>	positive edge:	↓ > ↑														
positive edge	>	positive edge:	↑ > ↑														
The combination negative edge	>	negative edge:	↓ > ↓ is not allowed														
Pulse time	<p>The point in time of a signal edge within the sampling interval is returned. Certain imc Online FAMOS functions used this information, e.g. when determining a pulse signal's plot of angular speed: OtrEncoderPulsesToRpm.</p> <p> The mode Pulse time depends on the sampling interval. The entry only appears for some amplifiers if the appropriate sampling interval is set (≤ 1 ms or ≤ 100 μs; see device manual).</p>																

Combined measurement

Measurement mode	Description
Frequency	The time of two consecutive pulses is converted to frequency. If the frequency was previously multiplied or divided, this can be reflected in the scaling value. The frequency is always unsigned, for which reason there is no dual-track encoder for it.
Speed	The sequence of pulses is converted to m/s. Toward this end, the amount of pulses per meter must be entered.
RPM	The sequence of pulses is converted to revolutions per minute. Toward this end, the amount of pulses per revolution must be entered.

9.3.7 Filtering

The amplifiers are able to filter the signal before it's subject to further processing in the system. The 6th-order filter with Butterworth or Bessel characteristics performs high-, low- and band-pass filtering.



Filter settings (example: Bandpass)

Here you can select the **filter characteristics** and the **filter type** for the selected channel. The cutoff frequency 2 can only be set for bandpass filters.

Along with filter types to select by their cutoff frequencies, there is also an **anti-aliasing filter (AAF)**. This is a low-pass filter whose cutoff frequency is set to comply with the requirements of the Sampling Theorem. Thus, frequency components above ½ of the sampling frequency are suppressed. This refers exclusively to the input channel, not to its [monitor channel](#). **Aliasing effects** are possible with a monitor channel by resampling.

The default setting is AAF, where a cauer filter is calculated internally. The displayed characteristic (in the picture Butterworth) is only used at a given cut-off frequency.

9.3.7.1 Filter settings

Theoretical background

The filter setting is especially important in a signal-sampling measurement system: the theory of digital signal processing and especially the **sampling theorem** (Shannon, Nyquist) state that for such a system, the signal must be restricted to a limited frequency band to ensure that the signal has only negligible frequency components beyond one-half of the sampling frequency ("Nyquist-frequency"). Otherwise, "aliasing" can result – distortions which cannot be removed even by subsequent filtering.

The imc device is a sampling system in which the sampling time (or sampling rate) to be set is subject to this condition. The low pass filter frequency selected thus hinges on how band-limited the signal to be sampled at that rate is.

The control AAF for the filter setting stands for "Automatic Anti-aliasing Filter", and automatically selects the filter frequency in adaptation to the sampling rate selected. The rule this is based on is given by:

$$\text{AAF-Filter frequency (-80 dB)} = \text{sampling frequency} \cdot 0.6 = \text{Nyquist frequency} \cdot 1.2$$

$$\text{AAF-Filter frequency (-0.1 dB)} = \text{sampling frequency} \cdot 0.4 = \text{Nyquist frequency} \cdot 0.8$$

General filter concept

The imc system architecture is actually a two-step system in which the analog signals are sampled at a fixed "primary" sampling rate (analog-digital conversion with Sigma-Delta ADCs). Therefore a fixed-frequency analog low pass filter prevents aliasing errors to this primary rate. The value of this primary rate is not visible from the outside, depends on the channel type and is generally greater than or equal to the sampling rate which is selected in the settings interface.

The filter to be set is realized as a digital filter, which offers the advantage of precise characteristic and matching with respect to magnitude and phase. This is especially important for the sake of matching of channels which are jointly subjected to math operations.

For any data rate to be set in the system configuration (f_{sample}), then digital anti-aliasing filters (low pass filters) ensure compliance with the conditions for the Sampling Theorem. Three cases can be distinguished.

Implemented filters

Filter-setting "Filter-Type: without":

Only the (analog) anti-aliasing filter, matched to the primary data rate is in effect.

This setting can be useful if maximum bandwidth reserves are to be used and there are known limitations on the measured signal's spectral distribution, which justify not performing consistent filtering.

Filter-setting "Filter-Type: AAF":

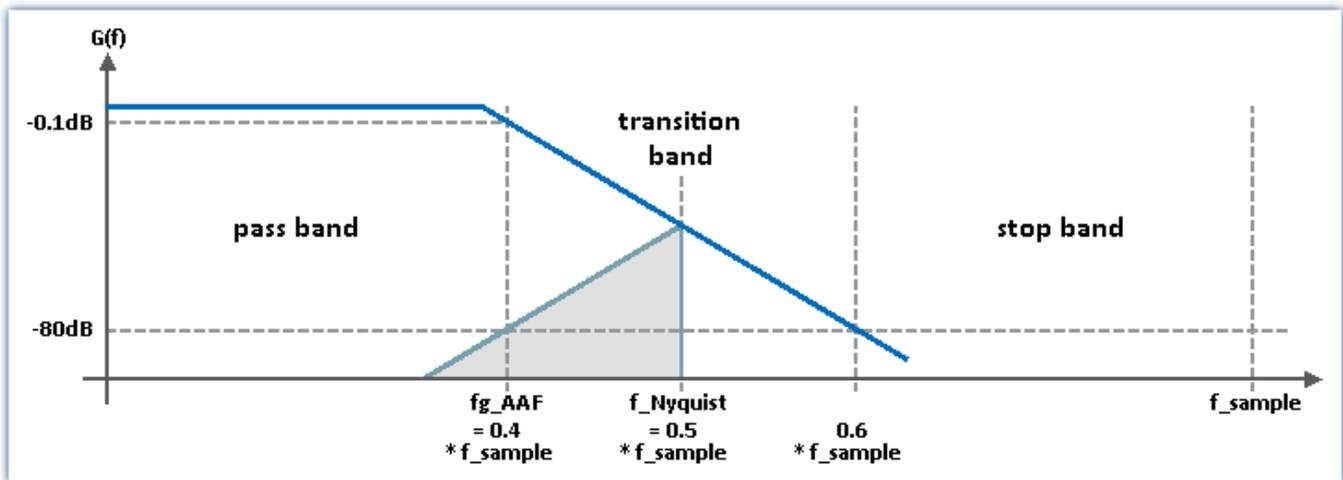
The (digital) anti-aliasing filters are elliptical Caer filters. Their "tight" characteristic curve in the frequency range makes it possible to have the cutoff frequencies approach the sampling and Nyquist frequencies much closer without having to make a compromise between the bandwidth and freedom from aliasing.

The automatic selection of the cutoff frequency in the setting "AAF" is based on the following criteria:

- In the pass band, a maximum (AC-) gain uncertainty of 0.06% = -0.005 dB is permitted. The pass band is defined by the cutoff frequency at which this value is exceeded.
- The stop band is characterized by attenuation of at least -80 dB. This damping is considered sufficient since discrete disturbance frequencies can never reach 100% amplitude: the input range is mostly filled by the useful signal.
- The transition band is typically situated symmetrically around the Nyquist-frequency. This ensures that the aliasing components reflected from the stop band back into the pass band are adequately suppressed, by at least -80 dB. Remnant components from the frequency range between Nyquist-frequency and stop band limit only reflect back into the range beyond the pass band (pass band to Nyquist), whose signal content is defined as not relevant.

The criteria stated are fulfilled with the Cauer-filters by the following configuration rule:

- $fg_AAF (-0.1 \text{ dB}) = 0.4 \cdot f_sample$
- Characteristics: Cauer; Filter-order: 8th order



Filter-setting "Filter-type: Low pass" (band pass and high pass):

A low pass frequency can be set manually, which satisfies the application's requirements. In particular, a cutoff frequency significantly below the Nyquist frequency can be set which guarantees eliminating aliasing in any case, though consequently "sacrificing" the corresponding bandwidth reserves.

- | | | |
|--|---|----------|
| with $fg_AAF (3 \text{ dB}) = f_sample / 4$ | attenuation at Nyquist-freq.: $1/64$ | = -36 dB |
| with $fg_AAF (3 \text{ dB}) = f_sample / 5$ | attenuation at Nyquist-freq.: $1/244$ | = -48 dB |
| with $fg_AAF (3 \text{ dB}) = f_sample / 10$ | attenuation at Nyquist-freq.: $1/15630$ | = -84 dB |

- Characteristics: Butterworth, 8th order (48 dB/octave)

For devices of [firmware group A](#)¹⁴⁶: Other possible filter settings are "band pass" and "high pass" - both 4th order.

9.3.8 Sampling & Preprocessing


Sampling & Preprocessing

Channel name Channel_001

Sampling

Sampling rate:

Sampling time:

Data type:

Sample count:

Duration:

X-axis:

Preprocessing

Function:

Points:

Resulting sampling rate:

Resulting sampling time:

Sampling & Preprocessing

The availability and selection of settings depends on the channel type.

Parameter	Description		
	<i>long name</i>	<i>short name</i>	<i>column ID</i>
Sampling rate	<i>Sampling rate</i>		<i>SampleRate</i>
<p>Setting for the sampling rate.</p> <p style="padding-left: 20px;">Sampling rate = 1/Sampling time</p> <p>Active channels can use a maximum of two different sampling times/rates per device.</p> <p>For all active incremental counter channels (and associated active Monitor channels) of one module (e.g. CRFX) or a device (e.g. C-SERIES), only one sampling interval/-rate may be set.</p> <p>See also: Notes on the "Sampling rate" and the "Aggregate sampling rate" ²⁶⁷</p>			
Sampling time	<i>Sampling time</i>		<i>eSampleTime</i>
<p>Setting for the sampling time.</p> <p style="padding-left: 20px;">Sampling time = 1/Sampling rate</p> <p>Active channels can use a maximum of two different sampling times/rates per device.</p> <p>For all active incremental counter channels (and associated active Monitor channels) of one module (e.g. CRFX) or a device (e.g. C-SERIES), only one sampling interval/-rate may be set.</p> <p>See also: Notes on the "Sampling rate" and the "Aggregate sampling rate" ²⁶⁷</p>			
Sample count	<i>Sample count</i>		<i>SampleCount</i>
<p>The amount of measured values per measurement. Select <i>undefined</i> (or enter "0") for unlimited measurement duration.</p> <p>When the Sample count is entered, the resulting measurement duration is corrected accordingly. The measurement duration is given by: Sample count * Sampling time.</p>			
Duration	<i>Duration</i>		<i>eDuration</i>
<p>The duration of the measurement. Select <i>undefined</i> (or enter "0") for unlimited measurement duration.</p> <p>When the measurement duration is entered, the resulting sample count is corrected accordingly. The sample count is given by Duration/Sampling time.</p>			

Parameter	Description		
	<i>long name</i>	<i>short name</i>	<i>column ID</i>
Measurement mode	<i>Measurement mode</i>	<i>Mode</i>	<i>eChannelMode</i>
<p>Basic setting for the measurement type Digital Inputs:</p> <ul style="list-style-type: none"> • Bit Input: the port indicates the current state • Sampling: the port continually captures data. These digital channels can be displayed the same way as analog channels. <p>Changes apply to the port and the corresponding bits.</p>			
Data type	<i>Channel data type</i>	<i>Data type</i>	<i>eDataType</i>
<p>Resolution of measurement results:</p> <ul style="list-style-type: none"> • 16-bit integer • Float (24-bit mode) (32-bit float (24-Bit mantissa)) <p>The availability of this option depends on the device.</p> <p>Samples with 24-bit resolution generate twice as much data volume. For 24-bit resolution, please be aware of the increased memory requirement^[755] and please observe the technical specifications stated in the user's manual for the respective device. In 16-bit mode, for devices belonging to Group A7^[146] and higher, the maximum aggregate sampling rate achievable via EtherCAT is 2000 kHz.</p>			

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>
	<ul style="list-style-type: none"> • <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).		

Preprocessing

Simple processing of measured values directly in the device. By means of preprocessing, the data rate is reduced. The setting "*Processing points*" specifies how many data points are to be processed. The returned result is a new value, and the data rate is thus reduced by a factor of this number of points. In contrast to the analogous functions in imc Online FAMOS, this technique does not allow you to set processing for a moving window of data.

Functions

The following functions are available:

Preprocessing	Group A		Group B		
	CRC, CRSL, SPAR, C-SERIES	CRFX, CRXT	EOS	ARGFT	
	Original & monitor channel	Original & monitor channel	Original & monitor channel	Original channel	Monitor channel
Arithmetic mean	●	---	---	---	●
Minimum	●	---	---	---	●
Maximum	●	---	---	---	●
RMS	●	---	---	---	●
AAF	---	---	---	---	●
Min-Max	●	---	---	---	---
Envelope	---	---	--- ²	---	●
Reduction	--- ¹	--- ¹	---	---	●

1 The reduction for the monitor channels is possible by changing the sampling rate.

2 imc EOS monitor channels are always envelope reduced channels.

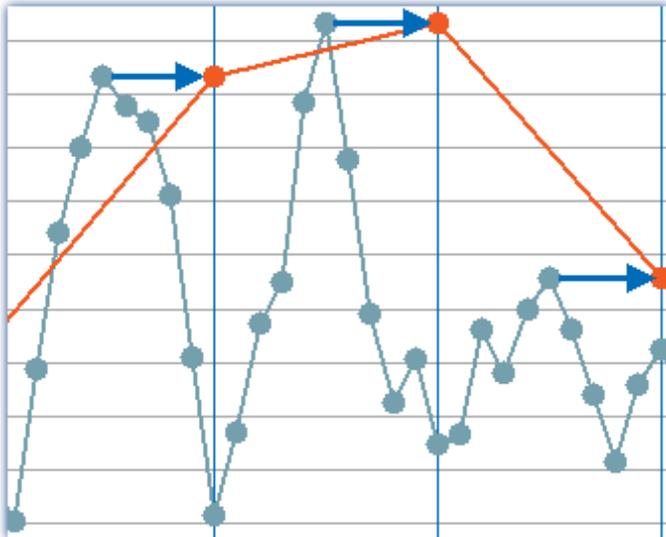
The data reduction process

Every x points, a new value appears at the **point in time of the x^{th} value**.

The following example illustrates how data reduction is proportional to the point count.

- Function: Maximum
- Points: 10

In the example, the system finds the maximum value among the last 10 values and sets it at the position of the last of the 10 values.



Example: Maximum across 10 values. The result (orange) is assigned the time stamp of the 10th value.

Due to the averaging duration, the first measured value is displayed as well as sorted with a delay. See "[Breakdown of the Start/Stop and Triggering Behavior](#)"^[482]

! Note

System resource demands

For devices belonging to the [firmware group A](#)^[146] (imc DEVICES):

The preprocessing is performed by the device's base board and **thus makes demands** on the device **processor**.

For devices belonging to the [firmware group B](#)^[146] (imc DEVICEScore):

The preprocessing is performed by the respective amplifier boards in the device and **thus does not make any demands** on the device's **processor**.

Monitor channel as envelope type

A data-reduced envelope type monitor channel (min-/max-channel) is generated.

This process is somewhat different than the other data reduction methods - See "[Envelope type monitor channel \(min-/max-channel\)](#)"^[732]

Parameters for the preprocessing

Parameter	Description		
	long name	short name	column ID
Function	<i>Processing function</i>	<i>Function</i>	<i>eProcessing</i>
	<p>Simple processing of the measured values. Different functions are available depending on the channel type/device.</p> <p>In contrast to the corresponding functions in imc Online FAMOS, it is not possible here to set processing of moving segments of data.</p> <hr/> <p> Due to the averaging duration, the first measured value is displayed as well as sorted with a delay. See "Breakdown of the Start/Stop and Triqgering Behavior"^[462]</p>		
Points	<i>Processing points</i>	<i>Points</i>	<i>eProcessingPoints</i>
	<p>The number of data points (measured values) on which the function is to be carried out. The number of points is also the reduction factor. This means that the resulting and stored sampling rate results from the set sampling rate/points. Monitor channels^[238] receive the data without preprocessing. Thus, the maximum sampling rate of a monitor channel is the set sampling rate.</p>		
Resulting sampling rate	<i>Resulting sampling rate</i>		<i>PreprocessedSampleRate</i>
	<p>Resulting count of values from the sampling rate/time set and the amount of points.</p> <p>Resulting Sampling rate = Sampling rate/Points</p>		
Resulting sampling time	<i>Resulting sampling time</i>		<i>PreprocessedSampleTime</i>
	<p>Resulting count of values from the sampling rate/time set and the amount of points.</p> <p>Resulting Sampling time = Sampling time*Points</p>		

Special function: Data reduction for digital inputs (reduction)

Only once the signal applied changes is a new measured value recorded. The respective time of data capture is recorded as a time stamp along with the measured data, for the purpose of subsequent data processing.

Therefore the port has to be set to Measurement mode: "Sampling".

Only whole ports can be captured with data reduction. Individual bits always are dependent on their port. As soon as an individual bit changes its state, the port is sampled and given a time stamp.

 **Note**

Transfer to PC is not possible: Results of the [TransRec](#) function and DI-port compressed channels are blocked from transfer to the PC.

9.3.8.1 Sampling rate and the Aggregate sampling rate

Tip: Avoid large sampling times

Say, for instance, a temperature is to be measured, and that 1 sampling per 10 s is sufficient resolution. A large sampling time has disadvantages:

- There is no averaging. If the signal is not subjected to analog smoothing, there will be no noise suppression.
- High frequency noise will cause aliasing errors.

Tip: Mean-function

- Set a higher sampling rate in the Configuration dialog. Especially for temperature measurement, sampling rates of at least 200 Hz should be set, if no analog anti-aliasing filter is used. Use averaging to reduce the data to the sampling rate desired.
- To the extent possible, use the imc Online FAMOS function [Mean](#) instead of the "Arith. Mean" on the page: "Analog/Digital channels" in the dialog: "Sampling & Preprocessing". Only the [Mean](#) function avoids the Start/Stop-delay.

For devices belonging to the [firmware group B](#)¹⁴⁶ (imc DEVICEcore):

You can set a separate sampling rate for each physical measurement channel. The aggregate sampling rate of the system is the sum of the **sampling rates of all active channels**. Active **monitor channels** are included in the calculation of the aggregate sampling rate at the reduced rate.

The aggregate sampling rate can take a maximum value of 4 MHz or 5 MHz with devices of the [firmware group B](#)¹⁴⁶.

Please see the pertinent technical data sheet to learn whether the aggregate sampling rate depends on the use of pv-variables or on the trigger count.

The sampling rates of the **virtual channels** computed by **imc FAMOS** do not contribute to the aggregate sampling rate.



Reference

Please consider the specifications in chapter "Technical Specs" of the device manual and the description of the channels configuration.

For devices belonging to the [firmware group A](#)¹⁴⁶ (imc DEVICES):

Among the measurement device's physical measurement channels, up to two different sampling times can be in use. The smallest possible sampling time is 10 μ s, corresponding to a channel sampling rate of 100 kHz (sampling frequency). The aggregate sampling rate of the system is the sum of the sampling rates of all active channels.

The aggregate sampling rate can take a maximum value of 400 kHz with devices of [group A4 to A5](#)¹⁴⁶.

The aggregate sampling rate can take a maximum value of 2 MHz via EtherCAT else 400 kHz with devices of [group A7](#)¹⁴⁶. This maximum aggregate sampling rate (2 MHz via EtherCAT) can only be achieved when not using the process vector variables, without any trigger, and only in 16-bit mode! A CRFX / DAC-8 module contributes 5 kHz and 16 bits per channel, regardless whether it is used in the experiment.

The sampling rates of the **virtual channels** computed by **imc FAMOS** do not contribute to the aggregate sampling rate. Along with the (maximum of) two "primary" sampling rates, the system can contain additional "sampling rates" resulting from the effects of certain data-reducing imc Online FAMOS functions (ReductionFactor RF).

 Note

There is one constraint when selecting **two different sampling rates**: **Two sampling rates** having the ratio **2:5** and lower than 1 ms are not permitted (e.g. 200 μ s and 500 μ s). Any attempt to set sampling rates which do not comply with this rule will cause an error message to be posted:

"The two active sampling intervals may not be in a ratio of 2:5. Error number: 365"

 Note

Note for imc CRONOSflex/CRONOS-XT

For CRFX/CRXT modules, the following limitations apply due to the CRFX systembus's 5 kHz bus rate:

- A CRFX/CRXT-channel sampling at a slower rate than 5 kHz produces the same data load on the systembus as a 5 kHz channel. Thus for instance, 20 channels sampled at 100 Hz produce a load of 100 kHz on the systembus instead of 2 kHz.
- With 2 kHz channels, the resulting data load per channel is 10 kHz due to the 2:5 ratio.
- DIO- and DAC modules support a maximum sampling rate of 5 kHz, even though 100 kHz can be set in the software.

 Note

Note for GPS channels

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 the same or faster sampling rate.

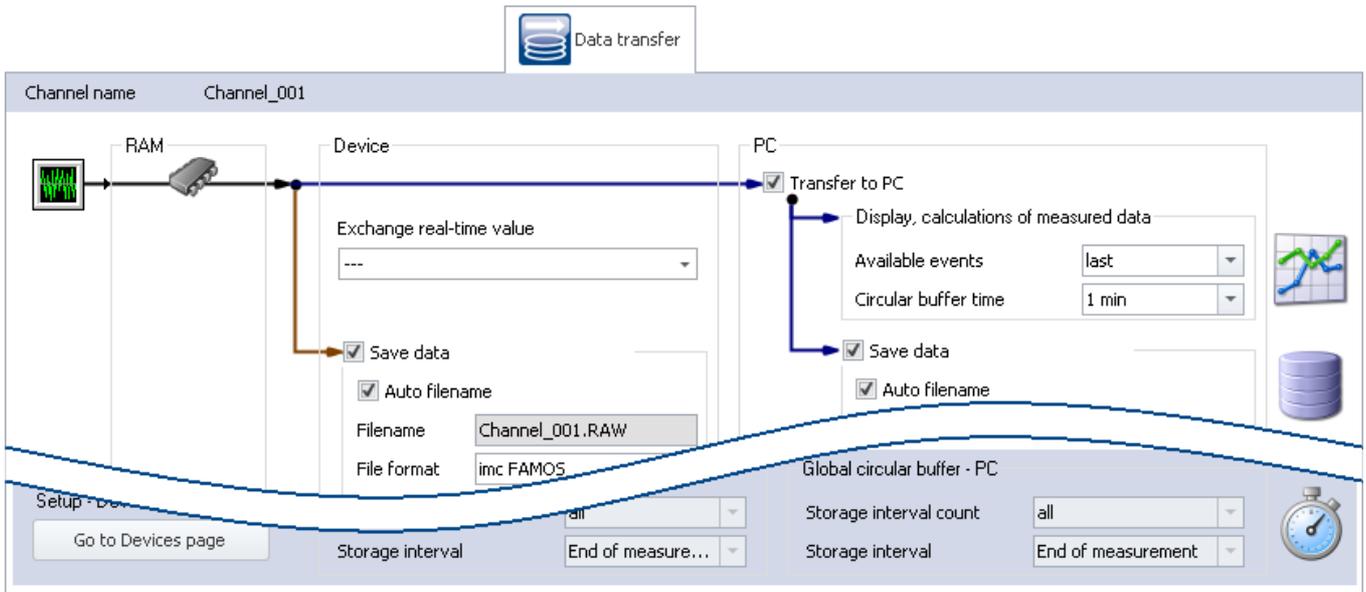
 Reference

Please consider the specifications in chapter "Technical Specs" of the device manual and the description of the channels configuration.

9.3.9 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](#)"³⁰⁶.



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 Storage ^[187] . 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](#)^[308]
- Available events and Circular buffer time: see "[Events and circular buffer](#)"^[317]
- "Exchange real-time value": See "[Exchanging values among the devices](#)"^[488]

9.3.10 Histogram / Rainflow

Only visible for certain virtual channels generated in imc Online FAMOS using appropriate functions.

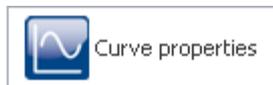
Here you can set defaults. These will affect the display and storage of the data of the respective virtual channel: e.g. Histogram or Rainflow.



Channel name	Histo1
Histogram display update interval	10 s
Histogram storage interval	5 min

Parameter	Description		
	long name	short name	column ID
Histogram display update interval	<i>Histogram display update interval</i>		<i>eHistogramUpdateInterval</i>
	Time interval for updating data which are derived from counting functions		
Histogram storage interval	<i>Histogram storage interval</i>		<i>eHistogramSaveInterval</i>
	Time interval for saving data which are derived from counting functions		

9.3.11 Curve properties



Channel name	Channel_001
Range	Auto
Min	-10
Max	10
Color	<input type="checkbox"/> 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 "Range" for the other Widgets is set as "From the variable".

The following possibilities are available:

Selection	Function	Wirkt auf Widgets
Auto	Max and Min values match the measurement range set for the channel	<ul style="list-style-type: none"> • scale-Widgets
Input range	Max- and Min-values match the measurement range set for the channel	<ul style="list-style-type: none"> • scale-Widgets • curve window (rezoom)
User-defined (min, max)	Max- and Min-values can be set in the parameters further below	<ul style="list-style-type: none"> • scale-Widgets • curve window (rezoom)

Min	<i>y-axis min</i>	<i>Min</i>	<i>eCurveYAxisMin</i>
-----	-------------------	------------	-----------------------

If "y-axis range" is set to "User-defined":

The transmitted Minimum-value for the display range.

Max	<i>y-axis max</i>	<i>Max</i>	<i>eCurveYAxisMax</i>
-----	-------------------	------------	-----------------------

If "Y-Axis option" is set to "User-defined":

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.

9.3.12 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".

Category: Channel

Parameter	Description		
	long name	short name	column ID
Balance at device startup	<i>Balance at device startup</i>	<i>Balance at startup</i>	<i>eBalanceAtDeviceStart</i>
This parameter has two purposes: <ul style="list-style-type: none"> • 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^[662]. 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 " ^[275] . Available options: Taring or bridge balancing (depending on the hardware).			
Connector	<i>Connector</i>		<i>ePlugInName</i>
The number returns a fixed identification of the variable. E.g. "[03] IN04". It is composed of: <ul style="list-style-type: none"> • [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: <ul style="list-style-type: none"> [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"		

Additional parameters are available, which are either already displayed in the channel table, or which you can add manually using the "Column chooser".

Parameter	Description		
	long name	short name	column ID
Channel number	<i>Enumerated channel number</i>	<i>Channel number</i>	<i>eEnumeratedChannelNumber</i>

Corresponds to the column "Connector", 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"

[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

Category: Module

Parameter	Description		
	long name	short name	column ID
Bridge supply	<i>Bridge supply (On/Off)</i>	<i>Bridge supply</i>	<i>eBridgeSupplyOnOffAction</i>

Some amplifiers provide the ability to switch the power supply on or off. The power supply voltage can be used either for the bridge supply or the sensor supply.

Additional parameters are available, which are either already displayed in the channel table, or which you can add manually using the "Column chooser".

Category: Other

Parameter	Description		
	<i>long name</i>	<i>short name</i>	<i>column ID</i>
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 "Prepare" 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 "Prepare" 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).

9.4 Channel balance

On the page **Channel balance**, you can have various adjustment and calibration types performed, and obtain information on the configuration.

To get to the desired page, there is a row of tabs at the lower left.

 Channel balance

Balance
Shunt calibration
Scaling / Cable compensation without Sense line

Channel name ▲	Connector ▲	Balance status	Balance action	Current value	Balance compensation
▼ Channel type: Analog inputs (Count=8)					
▶ Channel_001	[01] IN01	not balanced	Bridge ▼		0 "mV/V"; 0 "mV/V"
Channel_002	[01] IN02	not balanced	Bridge ▼		0 "mV/V"; 0 "mV/V"
Channel_003	[01] IN03	not balanced	Bridge ▼		0 "mV/V"; 0 "mV/V"
Channel_004	[01] IN04	not balanced	Bridge ▼		0 "mV/V"; 0 "mV/V"
Channel_005	[01] IN05	not balanced	Bridge ▼		0 "mV/V"; 0 "mV/V"
Channel_006	[01] IN06	not balanced	Bridge ▼		0 "mV/V"; 0 "mV/V"
Channel_007	[01] IN07	not balanced	Bridge ▼		0 "mV/V"; 0 "mV/V"
Channel_008	[01] IN08	not balanced	Bridge ▼		0 "mV/V"; 0 "mV/V"

[Balance](#) ²⁷⁵

- Tare
- Bridge
- Query wiring
- Reset charge

[Shunt calibration](#) ²⁸⁰

- Shunt calibration

[Scaling / Cable compensation without Sense line](#) ²⁸⁰

- Two-point scaling
- Cable compensation

9.4.1 Balance - tare and bridge

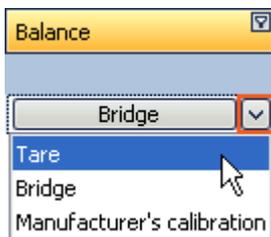
Balance

This page enables, among other things, **taring** and **bridge balancing** of amplifier channels to be performed.

 Channel balance

Balance	Shunt calibration	Scaling / Cable compensation without Sense line		Balance action	Current value	Balance compensation	Balance date	Balance time	Query	Wiring type	Reset
Channel type: Analog inputs (Count=8)											
Channel_001	[01] IN01	not balanced	Bridge	▼	0 "mV/V"; 0 "mV/V"				Wiring	▼	Execute
Channel_002	[01] IN02	not balanced	Bridge	▼	0 "mV/V"; 0 "mV/V"				Wiring	▼	Execute
Channel_003	[01] IN03	not balanced	Bridge	▼	0 "mV/V"; 0 "mV/V"				Wiring	▼	Execute
Channel_004	[01] IN04	not balanced	Bridge	▼	0 "mV/V"; 0 "mV/V"				Wiring	▼	Execute
Channel_005	[01] IN05	not balanced	Bridge	▼	0 "mV/V"; 0 "mV/V"				Wiring	▼	Execute
Channel_006	[01] IN06	not balanced	Bridge	▼	0 "mV/V"; 0 "mV/V"				Wiring	▼	Execute
Channel_007	[01] IN07	not balanced	Bridge	▼	0 "mV/V"; 0 "mV/V"				Wiring	▼	Execute
Channel_008	[01] IN08	not balanced	Bridge	▼	0 "mV/V"; 0 "mV/V"				Wiring	▼	Execute

To perform an adjustment, first select the channel and in the column *Balance* using the button a right, the desired action.



Adjustment only starts once you click on the button after making your selection.

Balance	Description
Bridge	<p>Performs physical bridge balancing for all selected bridge channels. Balancing can eliminate any offset distorting the measurement signal. The offset can be a multiple of the input range.</p> <p>With bridge balancing, the input range selected is retained. With multiple channels, the bridge balancing is performed in parallel amplifier-by-amplifier, but sequentially within an amplifier.</p>
Tare	<p>With taring, the distorting offset is canceled mathematically from the signals of the measured channels, which leads to a shift of the input ranges. The measurement chain including sensors is taken into account. Observe the notes on hardware ^[276].</p>
Manufacturer's calibration	<p>Manufacturer's calibration resets all selected channels to the original imc-factory calibration setting</p>

Note [Notes on hardware: balancing with setted offset](#)

Zeroing with adjusted offset leads to different results with CRFX/CRXT modules and amplifiers installed in imc CRONOScompact, imc C-SERIES or imc SPARTAN:

- CRFX/CRXT: [Zeroing](#) ^[275] adjusts the channel in the physical unit, e.g. 0 bar. This applies to all modules for the acquisition of analog voltages with 16 or 24 bits, except UNI-4.
- imc CRONOScompact, imc C-SERIES, imc SPARTAN, CRFX/UNI-4: The electrical value is balanced, i.e. the entered offset is retained. After the adjustment, the physical value is displayed with a negative offset. This ensures that an offset entered subsequently is not overwritten. If you wish for the balancing to determine the offset, **only enter the factor**, and then subsequently balance the offset by means of the [Balancing function](#) ^[275]. Adaptation to the behavior of CRFX/CRXT is in preparation.

Response of the balancing to changing the measurement range and measurement mode

The balancing process depends on the measurement range and measurement mode (Voltage – Current – Strain Gauge). This means that the balancing only applies for the currently valid setting. Whenever the measurement range or mode is changed, a new balancing process is necessary. The balance values are saved internally and are re-activated once the system returns to the associated setting.

For instance, if the measurement range is changed, the balance value applicable for the current range setting is saved internally, but cleared from the settings for the sake of the new measurement range. If there is already a balance value pertaining to the new measurement range, that value is loaded upon performing the preparation procedure



Example

Automatic restoration of the balance value upon changing the measurement range

Measurement range: $\pm 5V$

Value applied: 1V

Balancing performed -> shifted measurement range is -6 .. 4V

Measurement range changed to $\pm 2.5V$ -> This clears the balance value

Value applied: 2V

Balancing performed -> shifted measurement range is -4.5 .. 0.5V

Measurement range changed to $\pm 5V$ -> This clears the balance value

Prepare -> shifted measurement range is "-6 .. 4V", which matches the case of the first balance value.



Note

The results are saved with the experiment. Previous results are only deleted once a new adjustment or manufacturer's calibration has been performed.

**Question: Can I perform balancing also during a running measurement?**

Answer: Yes. It is possible to perform taring or bridge balancing while a measurement is running. Stopping the measurement is not necessary.

Upon starting the next measurement after a bridge balancing action, the "Current value" and the "Level indicator" automatically adopt the channel's value range boundaries. If the balancing is performed during the measurement, the element 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.

Note on devices belonging to firmware group B: If any changes were made to the device configuration during measurement, this is no longer possible.

Question: When is the balance OK?

Answer: Balance OK means that the balancing or taring has been performed successfully. With taring, the input range has been adapted accordingly and the procedure is complete. Successful balance is achieved once the remaining offset lies within the guaranteed tolerances.

Correspondingly, there are various error messages which can be returned.

- The connected bridge is unstable, i.e. the unbalance initially found is replaced by another, not eliminated.
- The connected bridge's initial unbalance is too high. How much of an initial unbalance can be compensated depends on the amplifier type (UNI-8, BR-4, etc.) and input range and is stated in the corresponding spec sheets.

Balancing is performed independently by the amplifier card, which receives the command to perform the balance on the signals which are connected to it. A balancing value is calculated on the basis of the measured value (DAC-settings with BR4 or DCB4; or 24Bit ADC value with Uni8 and similar systems). Subsequently, a new measurement is made using this value, which either confirms the previously calculated value or, with DAC-controlled balancing, causes an additional iteration of the process.

Question: Are the balancing values saved with an [experiment transferred when it is opened on a different device](#) 

Answer: When opening an experiment which is transferred to a different device, the balancing values saved with the experiment are also retained. Be aware that the balancing values are subject to some imprecision because they result from the balancing of the bridge and the tolerance of the bridge completion in the amplifier. This means that in the case of half or quarter bridges, an additional balancing is required to achieve high precision.

Question: Can balance values be exported as a parameter set and then imported again later?

Answer: If the exported channel's amplifier type and all relevant amplifier settings (bridge mode, measurement range, 16- or 24-bit resolution...) match those of the imported channel, then export and import are categorically possible. With **taring**, the balance value corrects the measured value and is thus not critical. With **bridge balancing**, there is an additional issue to consider since the balance values result both from the bridge balance adjustment process and from the tolerance of the bridge completion in the amplifier. For more on this topic, you can read the section: "[Import parameters](#)" .

Automatic balancing upon Diskstart/Autostart or using imc Online FAMOS

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	<i>long name</i>	<i>short name</i>	<i>column ID</i>
Balance at device startup	<i>Balance at device startup</i>	<i>Balance at startup</i>	<i>eBalanceAtDeviceStart</i>

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](#) ^[662].

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](#)" ^[275]. Available options: Taring or bridge balancing (depending on the hardware).

Query wiring

With this function, measurements of the Sense lines are taken and the circuit configuration is derived from the resulting data. In such cases, the hardware is unable to distinguish whether a circuit without Sense lines is involved or whether a bridge was accidentally not connected or connected incorrectly. Thus, this function is not intended for detecting the circuit but for verifying whether the desired circuit was correctly recognized by the hardware.

Note

With the quarter bridge, the hardware is unable to distinguish between 5- and 6-wire measurement

Resetting the charge

For charge amplifiers (coupling = "Charge/DC"), as well as for audio amplifiers with peak evaluation (filter type = "Peak"), it is periodically necessary to carry out a reset.

In the column Reset, the button *Execute* becomes active if this function is supported by the selected amplifier channel. This button manually resets the channel.

9.4.2 Shunt calibration

This page enables **shunt calibration** of amplifier channels.

 Channel balance

Balance	Shunt calibration	Scaling / Cable compensation without Sense line					
Channel name ▲	Connector ▲	Shunt calibration status	Shunt calibration	Current value	Shunt calibration duration	Shunt calibration resistor	Bridge resistor
▼ Channel type: Analog inputs (Count=8)							
Channel_001	[01] IN01	--	Execute		5 s	174.6 kΩ	120 Ω
Channel_002	[01] IN02	--	Execute		5 s	174.6 kΩ	120 Ω
Channel_003	[01] IN03	--	Execute		5 s	174.6 kΩ	120 Ω
Channel_004	[01] IN04	--	Execute		5 s	174.6 kΩ	120 Ω
Channel_005	[01] IN05	--	Execute		5 s	174.6 kΩ	120 Ω
Channel_006	[01] IN06	--	Execute		5 s	174.6 kΩ	120 Ω
Channel_007	[01] IN07	--	Execute		5 s	174.6 kΩ	120 Ω
Channel_008	[01] IN08	--	Execute		5 s	174.6 kΩ	120 Ω

When a shunt calibration is carried out, the bridge is unbalanced for a specified time by means of a shunt, a resistor connected in parallel. The shunts available depend on the amplifier type involved.

A shunt calibration can be carried out at any time during a running measurement.

The duration to be set is the minimum duration. It is much longer and depends on the system's load.


Note

The shunt calibration is to be considered as only a **function test**. Since the shunt is located within the device, the results are falsified by the resistance of the leads used. Precisely this fact is exploited, conversely, in cable compensation without a Sense line. The deviation measured is used to calculate the cable resistance.

9.4.3 Run balance or shunt calibration

Besides operation by means of the device software, it is also possible to balance or shunt calibration as follows:

- By means of the commando: "[Execute device action](#)^[1176]"

9.4.4 Scaling / Cable compensation without sense line

This page enables [Two-point scaling](#)^[281] and [Cable compensation without Sense line](#)^[282] to be performed.

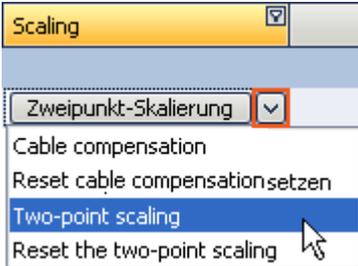
 Channel balance

Balance	Shunt calibration	Scaling / Cable compensation without Sense line				
Channel name ▲	Connector ▲	Scaling status	Scaling action	Point 1	Point 2	Cable compensation values
▼ Channel type: Analog inputs (Count=8)						
Channel_001	[01] IN01	not balanced	Two-point scaling	Acquired: -; Scaled: 0 "mV/V"	Acquired: -; Scaled: 0 "mV/V"	Cable resistor: 0 Ω; Bridge resistor: ...
Channel_002	[01] IN02	not balanced	Two-point scaling	Acquired: -; Scaled: 0 "mV/V"	Acquired: -; Scaled: 0 "mV/V"	Cable resistor: 0 Ω; Bridge resistor: ...
Channel_003	[01] IN03	not balanced	Two-point scaling	Acquired: -; Scaled: 0 "mV/V"	Acquired: -; Scaled: 0 "mV/V"	Cable resistor: 0 Ω; Bridge resistor: ...
Channel_004	[01] IN04	not balanced	Two-point scaling	Acquired: -; Scaled: 0 "mV/V"	Acquired: -; Scaled: 0 "mV/V"	Cable resistor: 0 Ω; Bridge resistor: ...
Channel_005	[01] IN05	not balanced	Two-point scaling	Acquired: -; Scaled: 0 "mV/V"	Acquired: -; Scaled: 0 "mV/V"	Cable resistor: 0 Ω; Bridge resistor: ...
Channel_006	[01] IN06	not balanced	Two-point scaling	Acquired: -; Scaled: 0 "mV/V"	Acquired: -; Scaled: 0 "mV/V"	Cable resistor: 0 Ω; Bridge resistor: ...
Channel_007	[01] IN07	not balanced	Two-point scaling	Acquired: -; Scaled: 0 "mV/V"	Acquired: -; Scaled: 0 "mV/V"	Cable resistor: 0 Ω; Bridge resistor: ...
Channel_008	[01] IN08	not balanced	Two-point scaling	Acquired: -; Scaled: 0 "mV/V"	Acquired: -; Scaled: 0 "mV/V"	Cable resistor: 0 Ω; Bridge resistor: ...

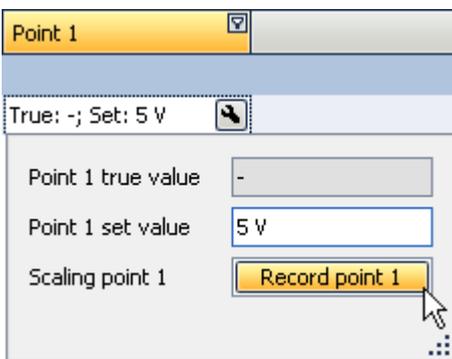
Two-point scaling

Two-point scaling allows linearization of the input range between two defined points.

To perform two-point scaling, first select the channel and in the column *scaling*, using the button at right, select *Two-point scaling*.



In the columns *Point 1/2*, set the respective target values. Record Point 1 and Point 2 in succession.



The status of the procedure is indicated in the column *Status*.

Once Points 1 and 2 have been successfully specified, the actual scaling must be performed. To do this, click on the button in the column *scaling*.

Channel name	Connector ↑	Status	Scaling
Channel type: Analog inputs			
Channel_01	[01] IN01	point 1 and 2 ok	Two-point scaling
Channel_02	[01] IN02	not balanced	Two-point scaling

Note

- If bridge balancing or bridge taring is carried out after two-point calibration, any offset present is canceled, but the slope between the two points remains intact.
- The points must be separated by at least 1% of the input range. If it is necessary for technical reasons to perform the adjustment over a small input range, there is a problem since upon return to larger input range the prevalent two-point scaling will be rejected. In this case, it will be necessary to resort to the two-point scaling on the dialog [Range & Scaling](#)'s page *Channels (Analog or Digital)*. A description is provided in "[Information and Tips](#)".
- The two-point scaling described here is performed by the amplifier module by measuring the signals applied. These signals are passed on to the base system. The base system itself is also able to perform an additional [Two-point scaling by manual input](#) by means of manual input. If both procedures are used, they are connected in series, meaning that the scaled valued from the amplifier become the input values of the base system's subsequent two-point scaling.

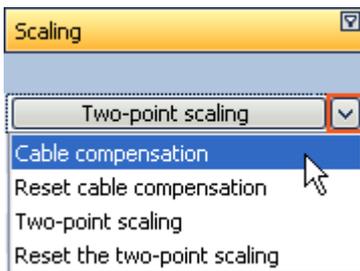
Resetting two-point scaling

To reset a two-point scale, in the column *scaling* use the button at right, *Reset the two-point scaling*. The results in the column *Point 1/2* are then no longer displayed. The two-point scaling is only deleted upon clicking on the button.

Cable compensation without sense line

If the bridge resistor is known, it's possible to determine the cable resistance, even or in fact especially if the Sense line isn't connected. Toward this end, a shunt calibration is performed. The deviation from the expected unbalance is used to determine the cable resistance.

In order to perform cable compensation, in the column *Scaling* use the button at right to select *Cable compensation*.



Set the bridge resistance if still needed.

To start the compensation, click on the button in the column *Scaling*.

After performing the cable compensation, the measured resistance values found will be displayed in the column *Cable compensation values*.

Note

- A bridge circuit with a SENSE-line is always preferable to cable compensation without a SENSE-line, in terms of the precision of the measurement results!
- For the cable compensation, the general rule of thumb in measurement engineering applies: select an input range so that the maximum expected input value reaches the top third of the input range. Otherwise, the cable compensation will not work reliably.

Resetting the cable compensation

To reset cable compensation, in the column *Scaling* select the button at right, *Reset Cable Compensation*. Only once the button has been pressed is the cable compensation deleted.

9.5 Trigger and events

A complex trigger machine is implemented in the measurement device. This chapter describes the detailed handling of the trigger options of imc measurement devices.

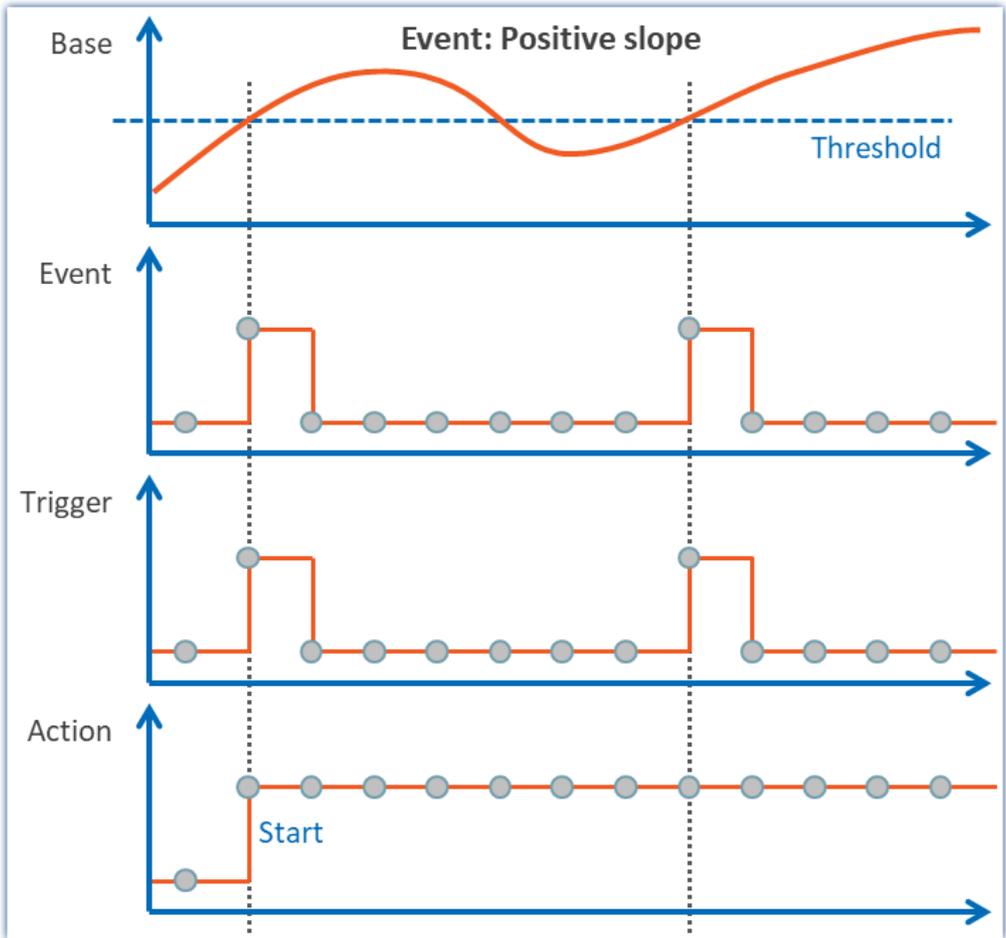
Many kinds of measurements are intended to start only once certain events occur. This applies, for instance, to measurements for which it is known in advance when measurement is to take place.

An example would be measurements of a bridge's vibration when heavy trucks drive over it. In such a case the vibration amplitude is greater than in response to a passing car, so the start of the measurement can be triggered by a specific vibration amplitude caused by a heavy truck.

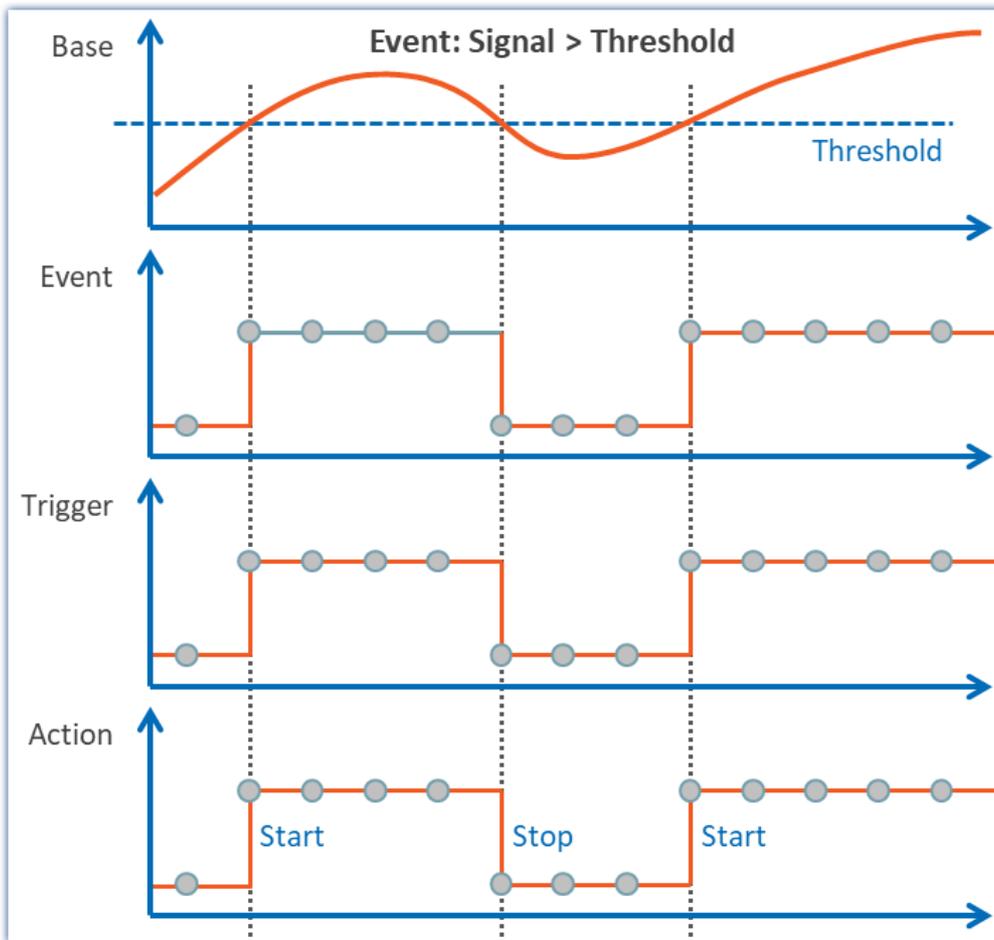
It is possible to continually monitor measurement channels and only to capture data in response to certain events. For this purpose triggers are used which start and stop channels based on the occurrence of certain events.

As soon as a certain event occurs in a signal (e.g. a particular level is exceeded > positive edge), the associated event channel's value becomes 1. This state remains in force for an increment of time. The trigger linked to the event is simultaneously triggered and the associated action is carried out.

Observe the two examples below:



An event releases a trigger and initiates an action
 Example: Positive slope



An event releases a trigger and initiates an action
 Example: Signal > Threshold

9.5.1 Overview

To begin, the "untriggered" measurement are explained. Afterwards, a description of the imc devices' [Trigger-Machine](#)^[285] is presented:

- Starting and stopping by trigger. What to make note of (see: [Sources and events](#)^[287])
- Special aspects of repeat measurements (see: [Multiple triggering \(Multi-Shot operation\)](#)^[298])
- Triggering with virtual channels (see: [Virtual Channels from imc Online FAMOS](#)^[302])
- Use of triggers in imc Online FAMOS (see: [Online-Trigger](#)^[302])
- Slow background measurement with rapid triggering in the foreground (see: [Example: Background Measurements](#)^[303])
- Event-driven setting of digital outputs (see: [Example: Setting event-driven digital outputs](#)^[304])

Untriggered measurements

Untriggered measurement is started using the Start-button (▶). No settings need to be made in order to do this. Recording of the data begins immediately. When measurement concludes is determined by the channel having the longest measurement duration. If this measurement is "undefined", close the measurement by means of the Stop-button (■).

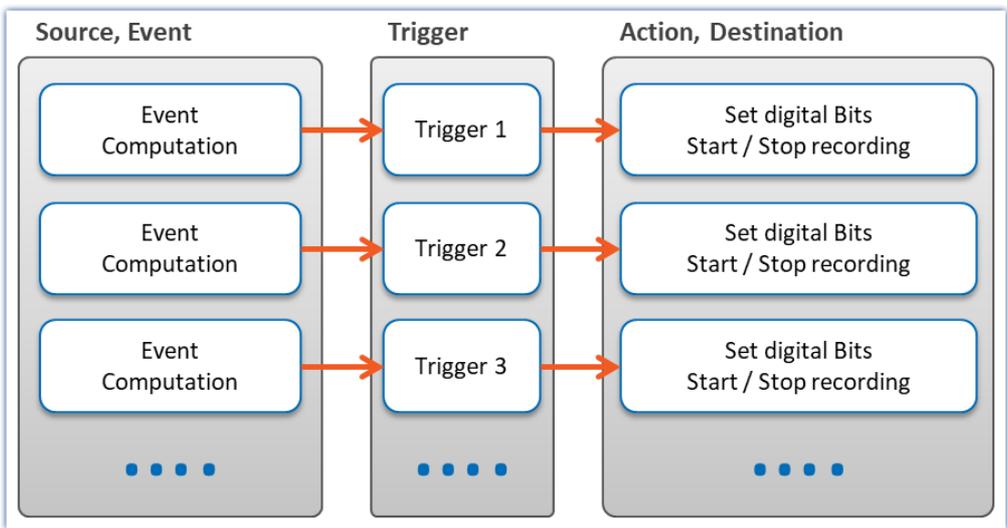
Note

Direct starting of the measurement is performed internally by an ["immediate trigger"](#)²⁹⁵. All channels which are not assigned to any other trigger start with the internal "BaseTrigger" (an "immediate trigger"). For more on this topic, see the description under ["Trigger Machine"](#)²⁸⁵.

9.5.2 Trigger machine

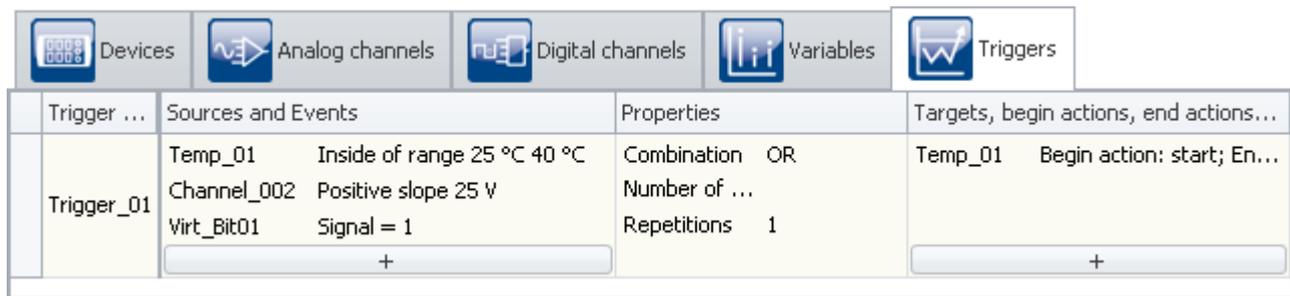
The measurement device implements a complex trigger machine which is capable of accomplishing difficult assignments. It has the following particular features:

- With the trigger machine, triggers can each be released independently (the **count** of triggers **depends on the device type**; refer to the "Technical Specs")
- Individual channels can be started and stopped in response to events
- Digital inputs can govern triggers
- Events can set digital outputs
- Virtual bits can be used for internal linkages
- Events can be combined with AND or OR operations
- The trigger machine works in real-time



Structure of the trigger machine

The relevant settings are made on the Triggers page of the dialog.



The Trigger dialog page



Note

Limitation - CRFX/CRXT

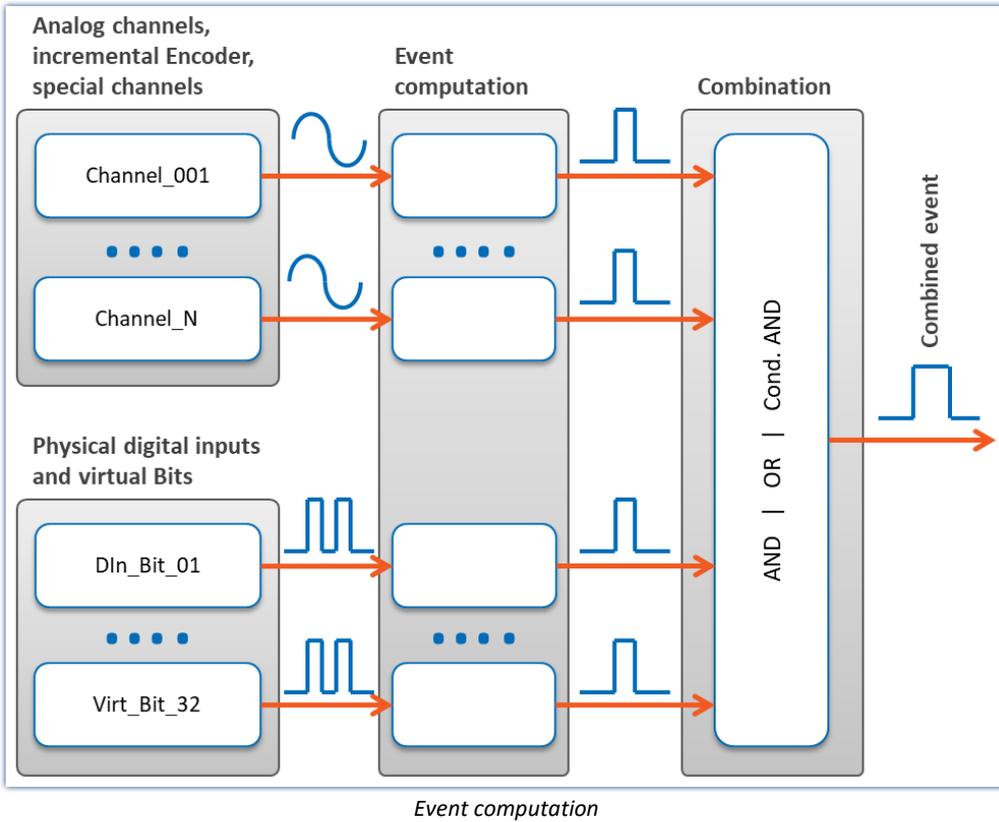
In CRFX/CRXT modules up to 8 trigger events per conditioner can be calculated.

9.5.2.1 Definitions

Term	Definitions
Event ²⁸⁷	An Event is a defined course or state of a measured signal. It is a binary unit of information, so it can only be either "true" or "false". Analog or digital variables can generate events.
Action ²⁹⁵	An Action can be the start or stop of data capture, or setting of a digital output.
Trigger	The Trigger is a defined combination of events in response to which an action is performed.
Hysteresis ²⁹³	If an channel triggers the start and end of the measurement by exceeding the threshold, the hysteresis ensures that the noise component on the signal does not end the measurement again immediately.
Dwell time ²⁹²	A Dwell time is a technique for artificially prolonging events.
Pretigger ³⁰⁰	A Pretigger serves to record the signal course leading up to an event.

9.5.2.2 Sources and events

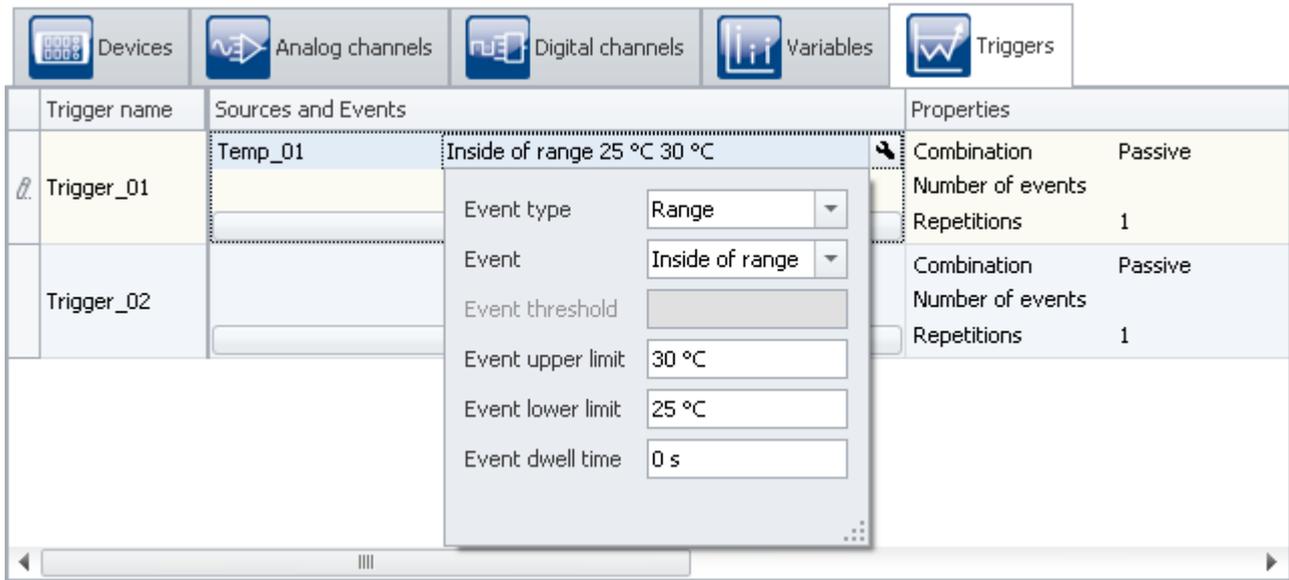
On the basis of each input channel (analog, incremental counter, Fieldbus) and each digital input (bit), it is possible to calculate an event. Multiple events can be joined to form a combined event. Calculation of the combined event is illustrated in the following outline:



Calculation of events from analog variables

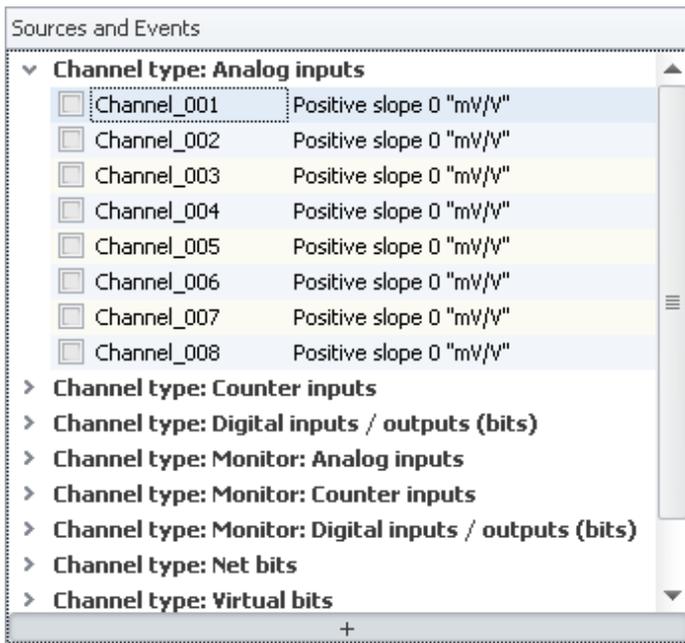
After pre-processing (correction by characteristic curve, scaling, averaging or formation of the RMS), the analog signal's physical value is entered into the event computation, which determines whether an event has occurred, according to the event type set.

The definition is made on the Setup page: "Triggers":



Example: The event occurs when the signal is within the boundaries 25 and 30 °C

Here, the channels (sources) and conditions (events) for releasing the triggers are set. Clicking the button takes you to a selection dialog for the sources.



Selecting a channel



Channels having the preprocessing "Envelope" can not serve as a trigger source.

At left are shown the available channels and at right the events set. To select a channel, click in the checkbox in front of the channel name. To configure the events, click on the event's entry, which opens a menu for setting the event.

Event type	Threshold
Event	Positive slope
Event threshold	3 "mV/V"
Event upper limit	
Event lower limit	
Event hysteresis	
Event dwell time	0 s

Setting an event



The "[Event hysteresis](#)" can only be set for a device belonging to the [firmware group B](#) (imc DEVICecore).

The available choices for "[Event type](#)" are "Threshold" and "Range". The specific event is selected next.

9.5.2.3 Event types

Event type: Threshold

The simplest and most common case is the event type "**threshold value**". In the box "[Dwell time](#)^[292]", you specify how long the event remains in effect once the condition is no longer met.

You can select the following events and their thresholds:

Event	Description
Positive slope	If the monitored channel's signal crosses the threshold from below, the event has occurred. This can theoretically happen very briefly, but in practical terms, it lasts for the space of one sampling interval. Note that in spite of the brevity of the event, a trigger may be released! This means it is even possible to set a digital output this way! In this case, the use of a " Dwell time ^[292] " is frequently helpful.
Negative slope	The event has occurred if the monitored channel's signal crosses the threshold from above. Like " <i>Positive slope</i> ", only in the reverse direction.
Signal>Level	As soon as the monitored channel's signal level is higher than the threshold, the event has occurred. This can be for a brief moment or a longer period, depending on the course of the signal.
Signal<Level	As soon as the monitored channel's signal level is higher than the threshold, the event has occurred. Like " <i>Signal > Threshold</i> ", only in the reverse direction.

Event type: Range

The user sets an amplitude range in which the event is defined as operative. The range is determined by a lower and an upper boundary. In the box "[Dwell time](#)^[292]", you specify how long the event remains in effect once the condition is no longer met.

The following events and boundaries are available for selection:

Event	Description
Entering range	If the monitored channel enters the range, whether from above or below, the event is defined to have occurred. This can theoretically happen for only a brief instant, but as a practical matter, lasts for at least one sampling interval duration. Note that in spite of the brevity of the event, a trigger may be released! This means it is even possible to set a digital output this way! In this case, the use of a " Dwell time ^[292] " is frequently helpful.
Exiting range	If the monitored channel's signal exits the range, whether upwards or downwards, the event is triggered. Like with " <i>Entering range</i> ", only in the reverse direction.
Inside of range	As long as the monitored channel's signal level is within the range, the event is in effect. This can be for a brief moment or a longer period, depending on the course of the signal.
Outside of range	As long as the monitored channel's signal level is outside of the range, the event is in effect. Like " <i>Inside of range</i> ", only in the reverse direction.

Events from digital signals and virtual bits

Digital signals come either from the device's digital inputs or the virtual bits. The 32 virtual bits (which amount to a 32-bit register), can also be read like digital inputs. For bits, there is no event type. In the box "[Dwell time](#)²⁹²", you specify how long the event remains in effect once the condition is no longer met.

The following events are available for selection:

Event	Description
Signal = 1	If the monitored bit's value is <i>1</i> , the event is defined to have occurred. This can happen for either a brief instant or for a very long period, depending on the course of the measured signal.
Signal = 0	If the monitored bit's value is <i>0</i> , the event is defined to have occurred. The opposite of " <i>Signal=1</i> ".
Signal change: 1 -> 0	If the monitored bit's value changes from <i>1</i> to <i>0</i> , the event is defined to have occurred. This can theoretically happen for only a brief instant, but as a practical matter, lasts for at least one sampling interval duration. Note that in spite of the brevity of the event, a trigger may be released! This means it is even possible to set a digital output this way! In this case, the use of a " Dwell time ²⁹² " is frequently helpful.
Signal change: 0 -> 1	If the monitored bit's value changes from <i>0</i> to <i>1</i> , the event is defined to have occurred. The opposite of " <i>Signal change 1 -> 0</i> ".

Note

For technical reasons, calculation of the trigger occurs only after 2 sampling intervals of the slowest channel. For instance, if the slowest channel is sampled at a rate of 1Hz, it means that any change of bit values is taken into account only after 2 seconds.

9.5.2.4 Dwell time: Artificially prolonging events

With events which remain in effect only for brief periods (signal edges, level-crossing), it is often necessary to prolong the event artificially by defining a "Dwell time" > 0.

Event type	Range
Event	Inside of range
Event threshold	
Event upper limit	30 °C
Event lower limit	25 °C
Event dwell time	250 ms

Einstellung der Haltezeit

maximum dwell time	
@ sampling rate	dwell time in hours
100 kHz	5
50 kHz	10
20 kHz	25
10 kHz	50
5 kHz	100
2 kHz	200
1 kHz and lower	500

Every event is prolonged by the duration of the Dwell time. By this means, even a very brief event can be retained for a defined duration.



Example

Example 1

A digital output is activated for an appreciable amount of time, even though the signal transition is very brief. A connected LED stays shining for a certain period instead of flashing for an instant.

Example 2

A measurement could be programmed to start not immediately upon detection of a signal edge, but rather following a pre-defined delay. This can be achieved, for instance, by making the signal edge first set a bit after a Dwell time and then set it back. The virtual bit's later transition from "1" to "0" is another event which is used for triggering the measurement's start.

9.5.2.5 Hysteresis

The event hysteresis can only be set for a device belonging to the [firmware group B](#) ¹⁴⁶ (imc DEVICecore).

If an channel triggers the **start and end** of the measurement by exceeding the threshold, the hysteresis ensures that the noise component on the signal does not end the measurement again immediately.

"*Signal > threshold*": The recordings starts when the threshold is exceeded, but is stopped only at `threshold - hysteresis`. With "*Signal < threshold*", the recording starts accordingly when the signal falls below the threshold and stops when the signal exceeds `threshold + hysteresis`.

The default setting of the hysteresis (auto) is 0.5% of the measuring range (maximum value-minimum value).



Example

Hysteresis

The settings shown start recording when the signal exceeds 2 V. The recording is stopped when the signal falls below 1.9 V (2 V - 0.1 V).

Event type	Threshold
Event	Signal > Threshold
Event threshold	2 V
Event upper limit	
Event lower limit	
Event hysteresis	0.1 V
Event dwell time	0 s

Setting an event

Automatic adjustment of the hysteresis to the measurement range

When the hysteresis is set to "*auto*", the following definition applies:

- "Auto" means: 0.5 % of the measurement range. Thus, changes to the measurement range affect the hysteresis.
- Exception: Temperature mode: Here, the hysteresis is 2 °C (3.6 °F) regardless of the measurement range.

Influence of the hysteresis on the measurement's start

The test of whether an event occurs is immediately resumed after a measurement for the sake of any subsequent measurement.

In conjunction with the hysteresis, the following may happen: An event occurring **before start of the measurement** can cause data recording on the channel to start, even though the reading is below the threshold at the measurement's starting time. Reason: before the start, a threshold violation was detected, and the reading is located within the hysteresis band without having passed below it.

The smaller the hysteresis, the less likely that this will happen.



Note

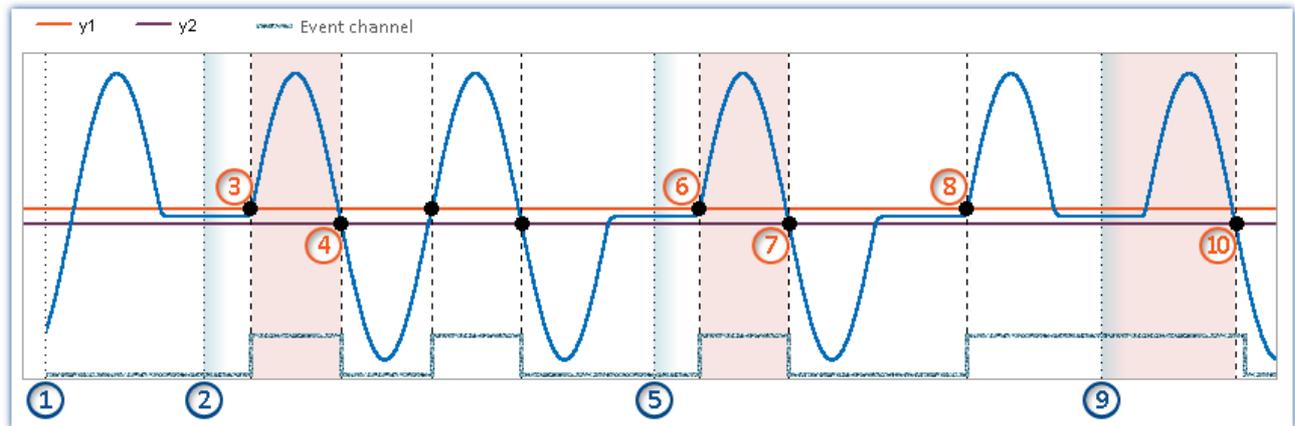
The first measurement

One **exception is the first measurement** after preparation of a device. Events occurring before the first measurement start are ignored. For this reason, it is possible to prevent this behavior by means of a targeted reconfiguration (genuine preparation) of the device.



Example

Starting within the hysteresis band



$y1$: Threshold value
 $y2$: End of the hysteresis band

Rule: Effects of events before measurement starts (if the device is already prepared):

- Events occurring before the first measurement start are ignored.
- Events occurring after the end of a measurement are not ignored for the next measurement.

Example illustrated by the event type "Threshold":

At the point of time ①, the action "Prepare" is completed

First measurement start: If measurement starts at the point in time ②, the measurement value is below the threshold value. The preceding event is ignored, since any events occurring before the first measurement start are not taken into account.

At the point in time ③, the threshold crossing is detected (internal event channel → "true"). Since the measurement is running, data recording begins.

Data recording concludes when the signal falls below the hysteresis band at the point in time ④.

Subsequent measurement starts: In subsequent measurements, events which occurred before the measurement started can have an effect. This can cause differing results. This is seen in the next two measurement starts.

- Measurement is started again at the point in time ⑤. The measured value lies below the threshold value. Before this point in time, there had been an event, but it is already over. For this reason, the event is not taken into account. At the point in time ⑥, the threshold crossing is detected (internal event channel → "true"). Since the measurement is running, data recording begins. Data recording concludes when the signal falls below the hysteresis band at the point in time ⑦.
- Measurement is started again at the point in time ⑨. The measured value lies below the threshold value. In spite of this, data recording on the channel begins immediately. Reason: After the threshold crossing at the point in time ⑧, the signal has not yet fallen below the hysteresis band (internal event channel remains "true"). Data recording is concluded when the signal falls below the hysteresis band at the point in time ⑩.

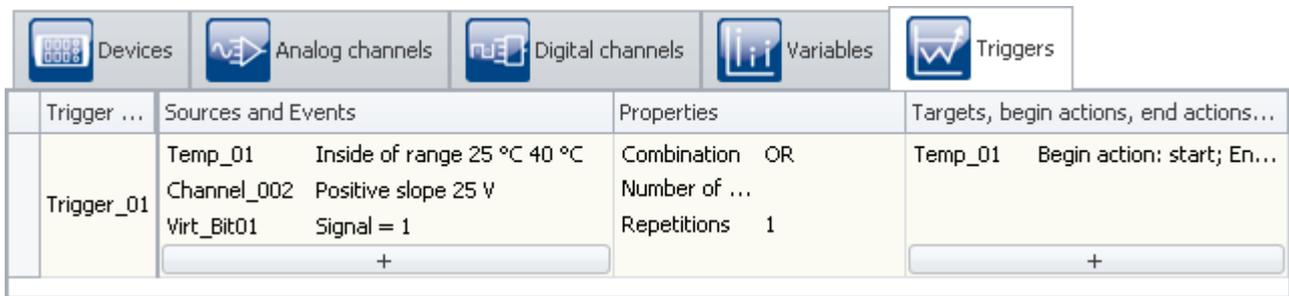
9.5.2.6 Combinations of events

It is possible to join multiple triggers in a logical expression. Five operators are available for the purpose:

Combinations	Description
AND	If all events are "true" at the same time, the combined event is "true" and the specified actions are carried out. For this reason it may not be sensible to combine two signal edge events with AND, since it is unlikely that two signals would have edges at the same time, unless the events were artificially prolonged by means of appropriate " dwell time ".
OR	As soon as one of the events is "true", the combined event is "true" and the associated actions are carried out.
Conditional AND	If a certain number of events (" Repetitions ") are "true" at the same time, the combined event is "true" and the associated actions are carried out.
Immediate	At the start of the measurement, the actions are performed. For this reason, no triggering event can be assigned. "BaseTrigger" is by default a "Immediate trigger", meaning it automatically starts all channels on which measurement is not subject to other triggers.
Passive	A passive (deactivated) trigger is no longer in use and automatically deletes all associated actions so that the channels can be entered for other purposes.

Setting the logical operation to combine events is done on the Setup page: "Triggers" in the column "Properties".

To change the combination, open the selection list in the column "Properties" and select the desired combination.



Combining events

9.5.2.7 Actions and targets

A variety of actions, triggered in a variety of ways, can be set as the target.

A trigger with the action: "Start data recording" can be triggered in a variety of ways:

- The combined event is "true"
- By means of the user interface of imc WAVE
- By means of imc Online FAMOS

There is only one way to release a trigger with the action: "Set bit" and "Stop recording of data":

- The combined event is "true"/"false"

Action: Starting data recording

Recording of the data **can only start of the trigger is "armed"** and the maximum count of trigger repetitions has not yet been reached.

A trigger is armed upon starting the measurement. This means it is ready to be released.

During data recording, the trigger is in the state "released".

If subsequently the recording of data on all of the trigger's target channels is concluded, for instance in consequence of the measurement duration set, the trigger is re-armed. If the maximum number of trigger repetitions has not yet been reached, the trigger can be released again.

Action: Setting bits

Each transition of the state of the combined event can set a bit to "1" or "0"; the combined event can include digital outputs and virtual bits. Virtual bits are in essence a 32-bit register which can be read from and written to at random. they have no physical terminal and only serve the purpose of exchanging information as well as halting states.

Note

Independence of count and measurement

- "Set bits" is **independent of the setting for "Repetition"**. It can be actuated any number of times.
- "Set bits" is **independent of whether a trigger is armed**. It is actuated independently of the measurement (after Preparation).

Note

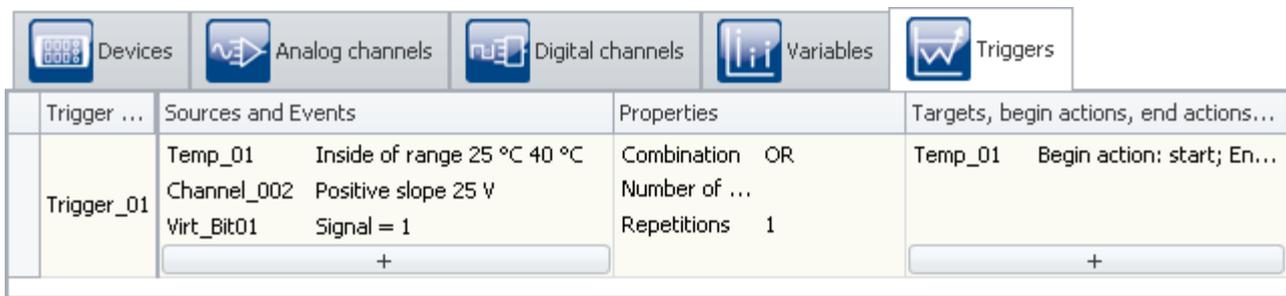
Changing the value retroactively

Setting of a bit is performed one single time until the trigger is fired again. While the trigger is active, the bit's value can be changed without the trigger always resetting it, for example, by means of the PC or imc Online FAMOS. This also applies to the virtual bits and the digital output bits.

Defining actions and targets

Making settings for the targets and actions is done on the Setup page: "Triggers" in the column "Targets, begin actions, end actions, pretrigger".

Here, the targets are specified. For instance a channel which is to measured or stopped upon release of the associated trigger.

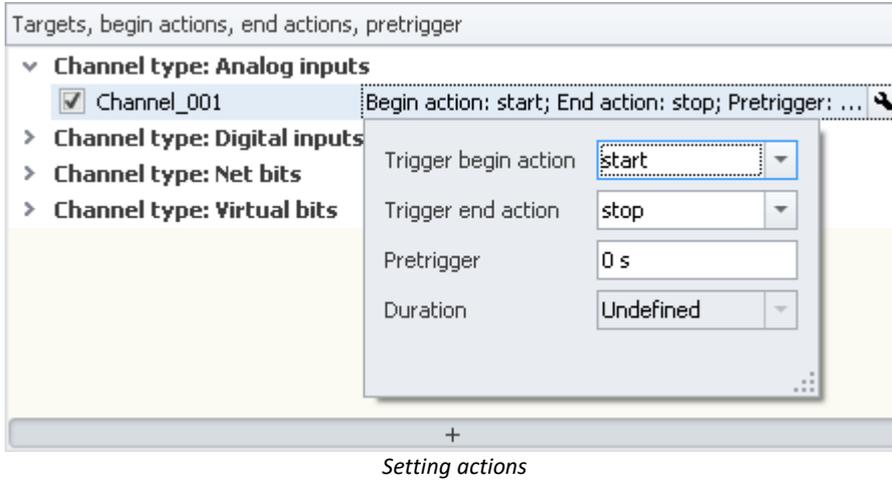


Defining targets and actions

The button  calls a dialog for selecting targets.

To select a target, click on the selection box.

To define the actions, click the left mouse button over the region at right.



Here, you can define the actions to be performed.

Parameter	Description
Trigger-start action	Action performed when the combined event becomes "true".
Trigger-end action	Action performed when the combined event becomes "false".
Pretrigger	A pretrigger determines the duration of data recording before the triggering time. See " Pretrigger ".

Possible targets for data acquisition channels

Targets	Description
Begin action: start End action: -	If the trigger is "armed" and the combined event is "true", data recording is started.
Begin action: stop End action: -	If the combined event is "true", data recording is stopped.
Begin action: start End action: stop	Like "Begin action: start". Additionally, data recording stops if the combined event is "false". Thus the channel's signal is recorded as long as the combined event is "true".

A channel can be started and/or stopped by exactly one trigger. The two actions can be either in response to two different triggers or to the same one.

Possible targets for digital bits

Targets	Description
Begin action: Begin = 1 End action: -	If the combined event is "true", the bit is set to "1".
Begin action: - End action: End = 1	If the combined event is "false", the bit is set to "0".
Begin action: Begin = 0 End action: -	If the combined event is "true", the bit is set to "0".

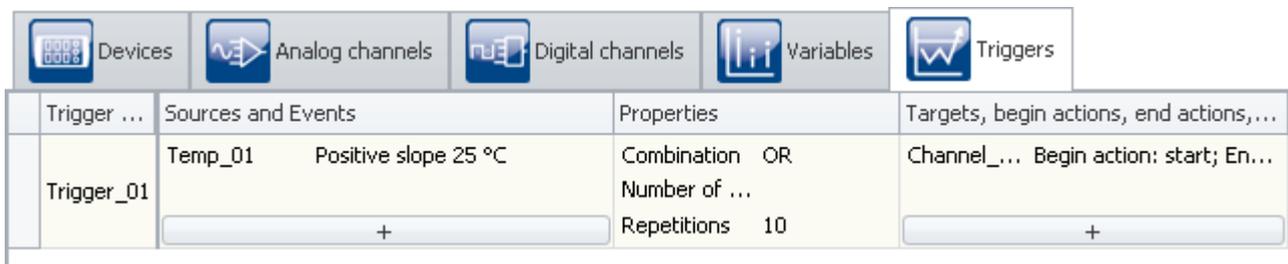
Targets	Description
Begin action: - End action: End = 0	If the combined event is "false", the bit is set to "1".
Begin action: Begin = 1 End action: End = 0	Like "Begin action: Begin = 1". Additionally, the bit is set to "0" if the combined event is "false". Thus, the bit is set to "1" as long as the combined event is "true".
Begin action: Begin = 0 End action: End = 1	Like "Begin action: Begin = 0". Additionally, the bit is set to "1" if the combined event is "false". Thus, the bit is set to 0 as long as the combined event is "true".

A bit can be set to 1/0 by exactly one trigger. The two actions can be either in response to two different triggers or to the same one.

9.5.2.8 Multiple triggering (multi-shot operation)

Within a measurement, it is possible for individual triggers to release multiple times.

Setting the "Repetitions" on the Setup page: "Trigger" in the column "Properties".



Setting the amount

Using the property: "**Repetitions**", you can specify the number of trigger releases. Normally, this repetitions is set to "unlimited". You can specify any repetitions desired.

In consequence, the trigger is rearmed after each episode of data recording until the specified number of trigger releases has been reached.

If you do not already know exactly how often you wish the trigger to release, set the number of trigger releases to "unlimited".

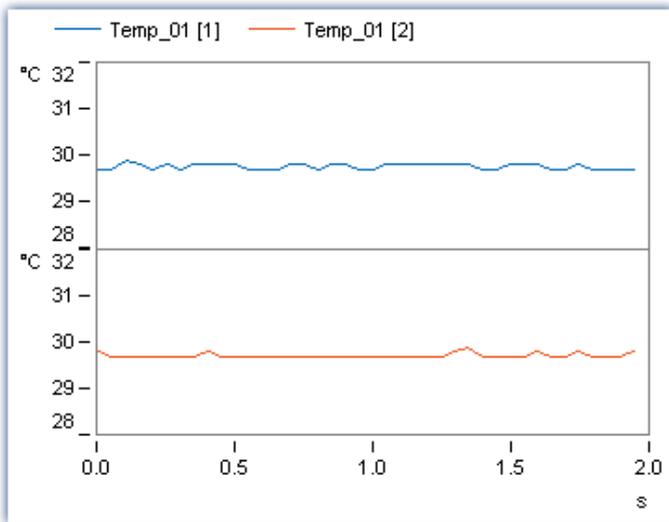
Note

- If you have set an unlimited number of trigger releases, the measurement never comes to an end by itself. Only after clicking on the Stop-button will the measurement conclude.
- The "Repetitions" can be set to **up to 32767 triggers**. Any larger number is automatically altered to "unlimited".
- "Set bits" is **independent of the setting for "Repetition"**. It can be actuated any number of times.
- Please note the remarks for [pretriggers](#)³⁰⁰.

All data belonging to a measurement are written to a common folder when the data are saved; this also applies to channels with multiple trigger releases. These channels can be saved in a file having multiple "Events", or in individual files (see the documentation for the device configuration > "[Storage](#)"¹⁸⁷).

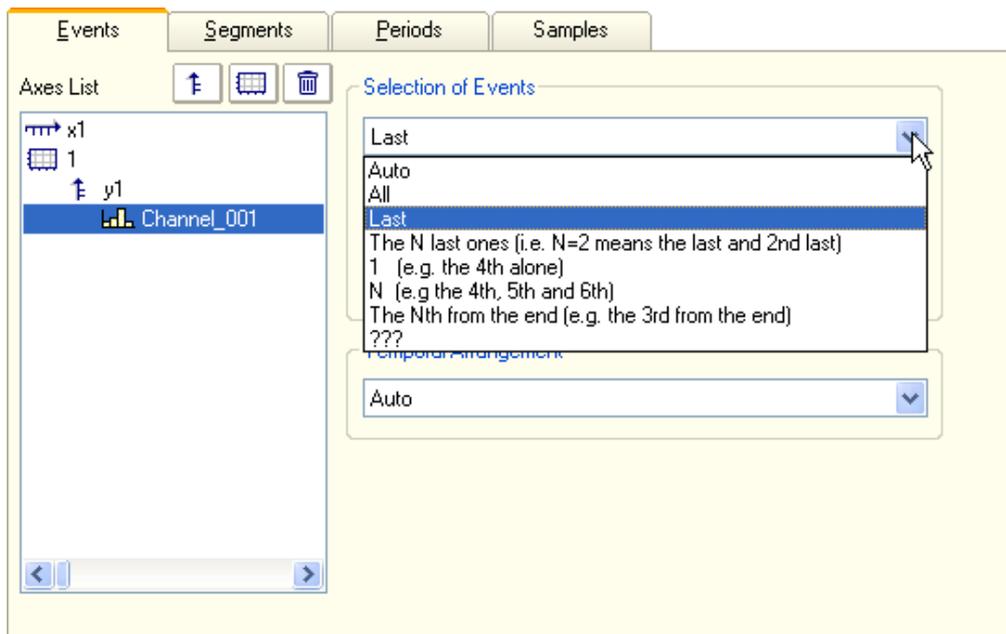
Among other things, multiple triggering enables easy, visual comparison of data from different trigger releases.

In the curve windows, you can compare the individual events:



Curve window with multiple events of the same channel

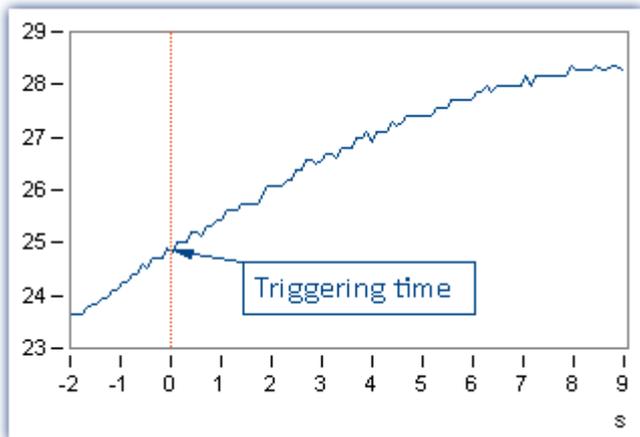
Toward this end, the curve window must be configured appropriately. Select the menu item "Configuration" > "Events, segments, periods...". On the page Events, you can select how many events you wish to have displayed.



9.5.2.9 Pretrigger

For every channel's signal recorded in response to a trigger, it is possible to also record the course of the signal leading up to the trigger event. A pretrigger determines the duration of recording before the trigger moment. The trigger moment's relative time is defined as 0. Pretrigger values have times < 0 .

Setting the pretrigger is performed on the Setup page: "Triggers" in the column "Target, begin actions, end actions, pretrigger".



Note

- The pretrigger is defined for channels which are assigned to a defined trigger. **If one deleted a trigger assignment for a channel, which is the same as assigning the channel to "BaseTrigger", the pretrigger is deleted automatically.**
- 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. For the second measurement filling the pretrigger will be started after stopping the first measurement. This is important to keep in mind, particularly with repeated trigger processes, since in such cases, the pretrigger only fills up after the preceding trigger event is complete. The pretrigger time counts toward the measurement duration set. **If the pretrigger is not completely filled, the overall measurement duration is that much shorter.**



Example

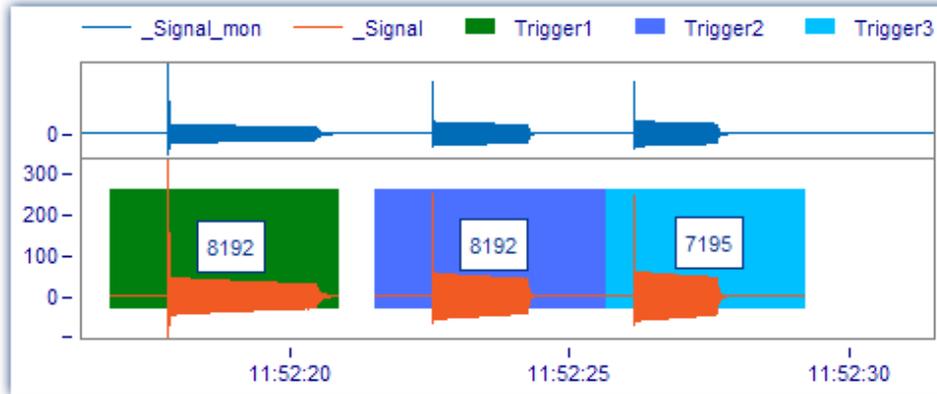
Setting for the measurement duration: 10 s, Pretrigger time: 2 s

Measurement duration after trigger = $(10-2)s = 8s$

If the trigger releases after 1s: Period of data capture = $1s + 8s = 9s$

- Example: FFT with 8192 samples of which 1024 samples are for the pretrigger:

At 500 Hz sampling rate -> Measurement duration = 4.1 s, pretrigger time: 512 ms



For the third event, the time was not sufficient for filling the pretrigger completely, and only 7195 samples were captured. Since the FFT needs 8192 for the calculation, it will not be possible to calculate the spectrum for this event.

9.5.3 The process of triggered measurement

With imc WAVE, a measurement begins when the Start-button is pressed (or when the device is activated, if it is in Autostart mode). Next, all triggers are armed. Untriggered channels which are assigned to the "*BaseTrigger*", immediately start recording data.

imc WAVE concludes measurement by itself once all component measurements specified have been performed. The measurement is over once no trigger has any outstanding repetitions to perform and all partial measurements are finished.

By clicking on the Stop-button, you can conclude a measurement before the appointed time.



Reference

See also

[Breakdown of the Start/Stop and Triggering Behavior](#) 482

9.5.4 Information and tips

9.5.4.1 Trigger machine and imc Online FAMOS

Virtual channels from imc Online FAMOS

Virtual channels generated by imc Online FAMOS can not be used directly as events for the trigger machine. Instead a virtual channel can be checked for events using imc Online FAMOS functions. The result switches one of the virtual bits which in turn can be evaluated by the trigger machine.

```
Operations
RMS_01= rms(Channel_01, 1000, 1000); RMS of voltage channel_01
Virt_Bit01= Greater( RMS_01, 230) ; Virtual Bit= 1, if RMS_01 > 230V
```

imc Online FAMOS: Exceeding the RMS value of 230V sets Virtual Bit 01

Trigger ...	Sources and Events	Properties	Targets, begin actions, end actions, pretrigger
Trigger_01	Virt_Bit01 Signal change: 0 -> 1	Combination OR Number of ... Repetitions Unlimited	Channel_02 Begin action: start; End acti...

Triggering of Channel_02 when the RMS values is exceeded, accomplished indirectly using Virt_Bit01

Note

It is possible to set a virtual bit both in the trigger machine and in imc Online FAMOS. To avoid this conflict, any bits set in imc Online FAMOS should never appear on the target side of the trigger machine.

Online-Trigger

imc Online FAMOS with [Control Commands](#) ⁵⁰¹ enables targeted actions in response to particular states of the trigger: [OnTriggerStart](#), [OnTriggerMeasure](#), [OnTriggerEnd](#).

Note

This only applies to triggers having a Start action: start.

The following applies to a trigger's control commands:

- When the trigger releases, the content of the control command [OnTriggerStart](#) is performed once.
- As long as the trigger is released, the content of the control command [OnTriggerMeasure](#) is performed continuously.
- When the triggering is finished, the content of the control command [OnTriggerEnd](#) is performed once.

If channels having different measurement durations are assigned to the same trigger, the triggering is completed at the end of the longest measurement.

- It is generally to be recommended that all of a trigger's physical channels are set with the same measurement duration.

9.5.4.2 Display before trigger events

Triggered channels are not displayed in a curve window before the trigger has been released. The data runs through the device as soon as the measurement has been started, but they can't be seen. If the sensor is disconnected or defective, it is not obvious why the trigger does not release.

For that reason we have the monitor channels. Those channels are sampled from the active channels. They can't be sampled faster than their original channels. But it is possible to define different or no trigger conditions to those monitor channels. So these channels can be seen before the original channels have been triggered.

Fieldbus channels

There are no monitor channels for most fieldbus channels. But they can be created with a trick. Use the Assistant to create within a channel's message an additional channel having a different name and the exact same settings. This channel can then be captured without triggering.

9.5.4.3 Trigger-variables in the Data Browser

The Data Browser contains not only a trigger's results channels but also **variables** for every **trigger** configured. These can represent, for example, **information on the trigger's status**. The trigger variables convey a variety of kinds of information, which can each be displayed separately and 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

Note

The Start-action must include a channel

The variables **only return values** if a channel is set for the trigger's **Start-action**.

9.5.5 Examples

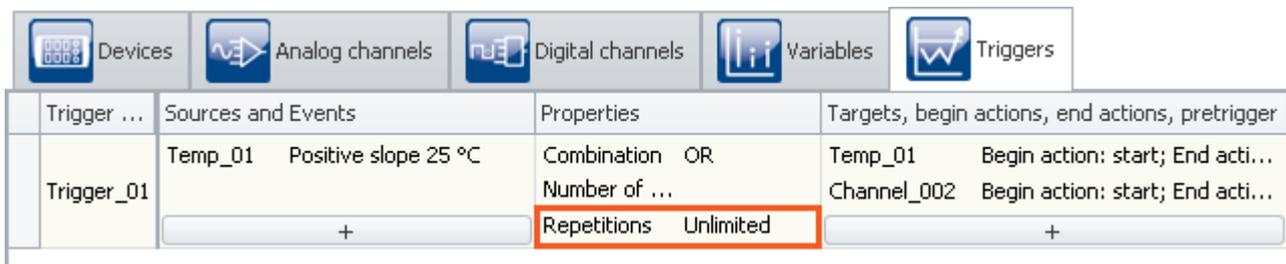
9.5.5.1 Background measurements

Two measurement tasks are to be performed in parallel by the same device. For instance, if you wished to record each activation and each braking procedure of some equipment, at high resolution. Between the recording of such measurements, there are often long pauses. Conversely, there are some signals which must continually be measured without any pause or gap in the data, e.g. temperatures and an engines RPMs.

This situation is referred to as a triggered measurement in the foreground at a high sampling rate and an untriggered background measurement at a low sampling rate.

Set the foreground measurement channel's trigger settings on the Setup page "Trigger" accordingly.

Also set "**Repetitions**" to "**Unlimited**".



Trigger settings for the measurement in the foreground

On the Setup page: "Analog/Digital channels", set the measurement duration of the background measurement channels to "Undefined". In this case, you must end measurement manually using the Stop-button.

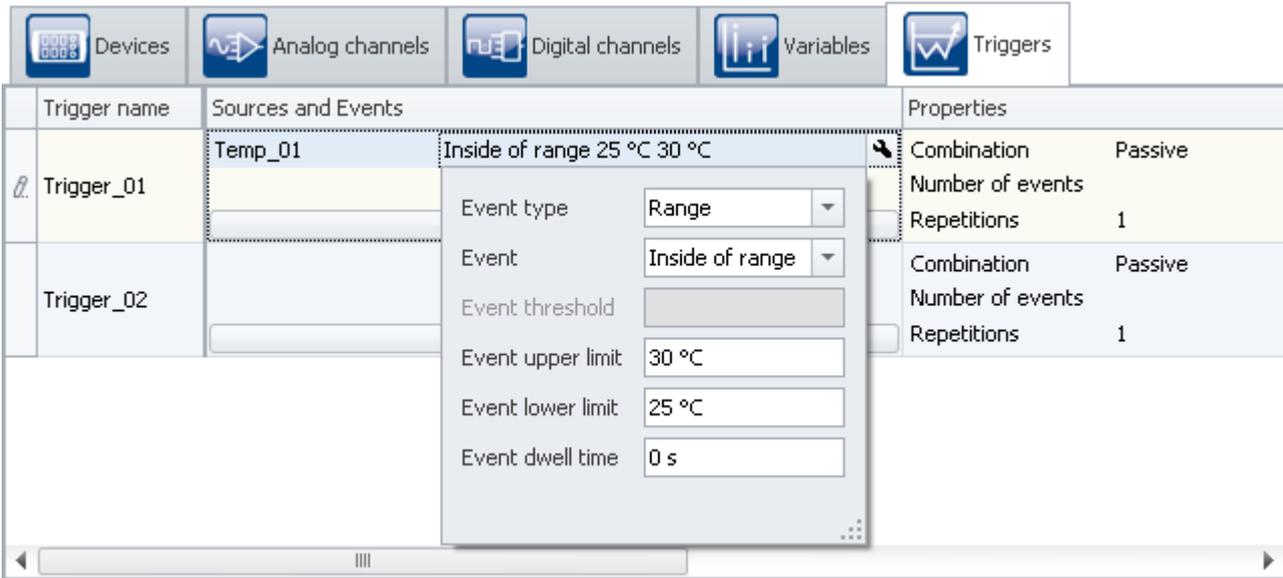
With these settings, it is possible for the trigger to release an infinite amount of times in succession during a measurement. After each concluded measurement, the trigger rearms automatically.

The measurement itself last until the background measurement is stopped by the Stop-button.

9.5.5.2 Event-controlled digital outputs

A digital output is to be set to 1 for the duration that a signal is within a specified tolerance band.

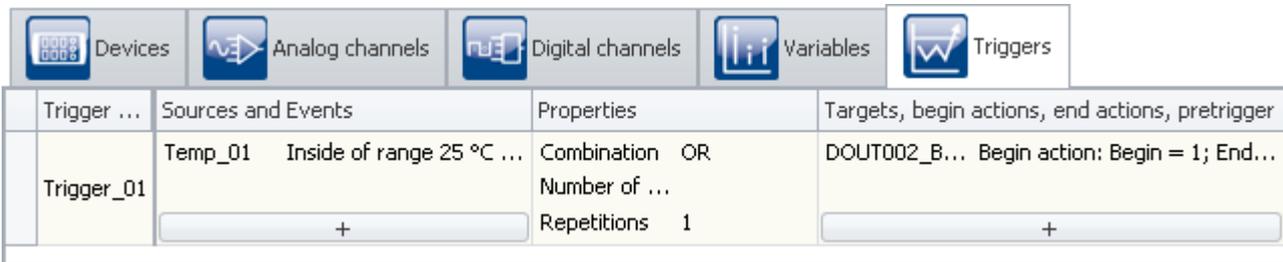
Set the channel's trigger event on the Setup page: "Triggers" accordingly:



Settings for the event

The event is defined to occur while the signal is within the specified range of 25 °C to 30 °C.

Next, the trigger is set.



Trigger settings for a bit output

Select as the "source" the channel from which the event is derived (here: "Temp_01").

Select as the "target" the desired digital output which you wish to set (here: "DOUT002_Bit01"). Set as the "Begin action": "Begin=1" and as the "End action": "End=0".

If the signal of the channel "Temp_01" is within the boundaries set while the measurement is running, the digital output bit's value is "high".

9.5.5.3 Start measurement by a virtual bit.

Problem: My measurement is started by the transition of a virtual bit from 0 to 1. This only works at the first start. If I start the measurement again without re-doing the preparation, it doesn't work.

Reason: The virtual bit is set the first time and isn't set back afterwards. At the next start, the bit is already 1, and no more transition from 0 to 1 takes place.

Solution: Once the trigger has been released, the bit must be set back again, which can also be handled at the end.

9.5.5.4 Logging triggers

- Virtual bit is the Start/Stop trigger for a channel;
- Using `RecordEvent (. .)`, the trigger's starting and stopping times are to be recorded:
`RecordEvent(vrtB1_Start_K2, "Trigger K2 ON")`
`RecordEvent(Not vrtB1_Start_K2, "Trigger K2 OFF")`
→ the first notification to arrive is "Trigger K2 OFF"
- The bit was set = 0 upon starting.

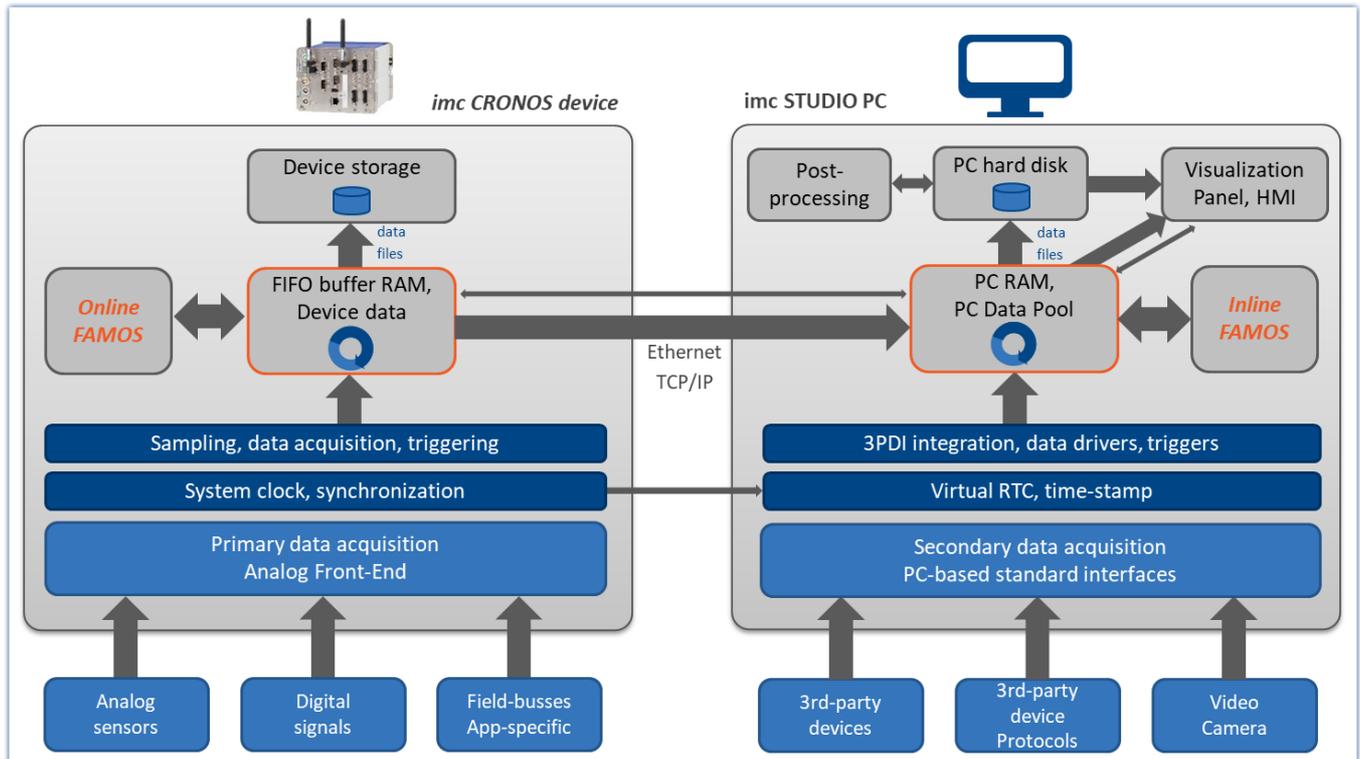
Solution: The `RecordEvent` function only indicates transitions from 0 to 1, but the `VirtBit_01` in the `OnTriggerMeasure` of `Trigger_01` is 1. If one wanted to make a record of this first event, the calls of the `RecordEvent` function must be made in "`BaseTrigger`".

9.6 Storage options and directory structure

This chapter describes the different options for saving measurement data with imc WAVE.

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](#)^[322] in the device.

This RAM buffer is read by a signal processor ("DSP").

- If you use an [imc Online FAMOS](#)^[498] 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](#)"^[308]).

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 ^[310]	By default, data saving is deactivated	The device can save the measured data to the internal storage medium ^[310] or to a network computer ^[314] (network sharing). What options are available depends on the particular device model and configuration.
Saving and displaying data on the PC ^[316]	By default, data saving is activated	<p>Here, the data stream is transferred to the controlling PC via the connection (mostly LAN) ("Transfer to the PC" ^[308]).</p> <p>The data can be displayed using the Panel pages. Additionally, the transferred measurement data can be saved on the PC.</p> <ul style="list-style-type: none"> • 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.

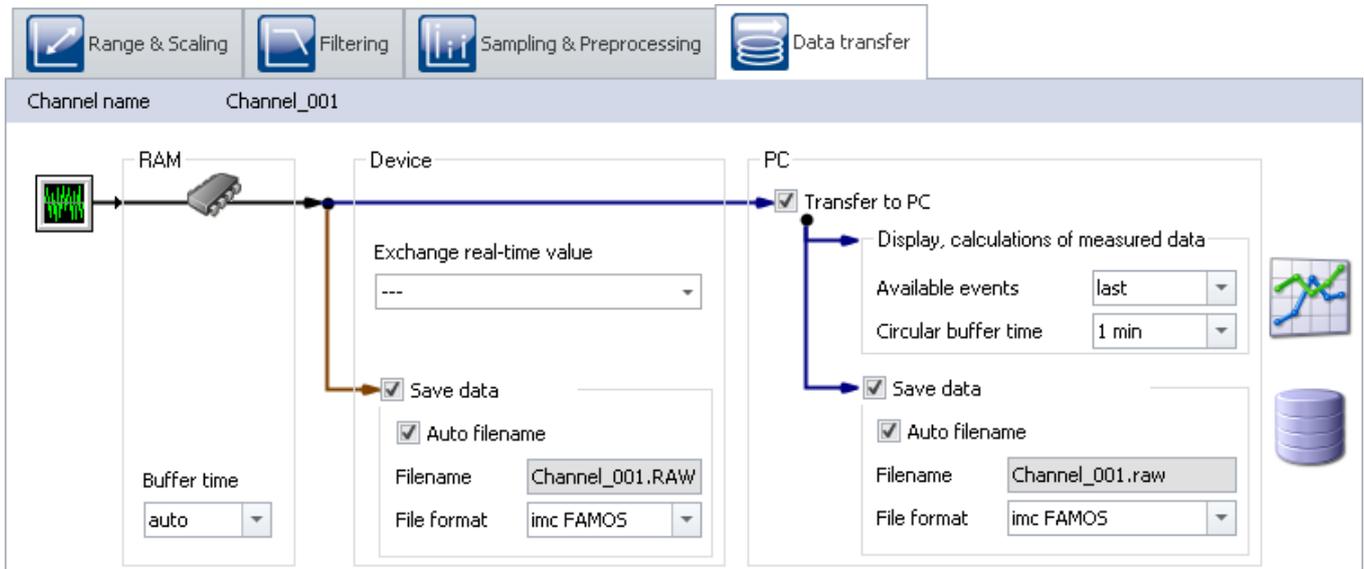
Data saving by the PC and by the device

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 ^[308] and File-Format ^[309]	Interval storage ^[318]
Circular buffer memory ^[309]	Save trigger events in individual files ^[329]
Available events ^[309]	Storage location on the device ^[310]
Transfer to PC ^[308]	

9.6.1 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 WAVE 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	<p>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.</p> <ul style="list-style-type: none"> • 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	<ul style="list-style-type: none"> • Saving on the PC³¹⁶: Saving of the measured data on the PC hard drive according to the folder structure³²⁸ set. • Saving on the device hard drive³¹⁶: Saving of the measured data device hard drive according to the folder structure³²⁸ set. Channels using this option are recorded under the current experiment name on the device's internal drive.
Auto filename	<p>If the box is checkmarked, imc WAVE will create a file name:</p> <ul style="list-style-type: none"> • PC and Device: The channel name will be used as file name

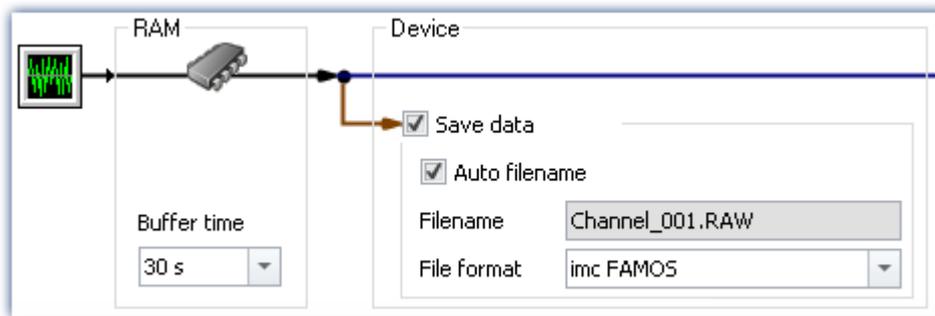
Parameter	Description								
Circular buffer time on the PC (for display/calculation)	<p>Here you set how long the maximum time segment is. When using this option, the required memory is known from the beginning of the measurement.</p> <p>Deactivating circular buffering: If you enter "0", the circular buffer is deactivated and all data are saved.</p> <hr/> <ul style="list-style-type: none"> • Recommendation: For the measured data (display and calculations), use an activated circular memory buffer. This transfers the data to the faster-working RAM memory. <p> • 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.</p> <hr/> <p> See "Events and circular buffer"^[317]"</p>								
Available events (Display)	<p>With this setting, you determine whether all events (triggered events) can be displayed, or only the last one.</p> <ul style="list-style-type: none"> • 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). <hr/> <p> • 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".</p> <p> • Note on analysis using imc FAMOS: If possible, use the measured data saved on the hard drive for analysis purposes.</p> <hr/> <p> See "Events and circular buffer"^[317]"</p>								
File format	<p>Format of the saved measured data. 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:</p> <table border="1"> <thead> <tr> <th>Format</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>imc FAMOS^[753]</td> <td>Default format for all device variables.</td> </tr> <tr> <td>imc FAMOS ZIP^[753]</td> <td>Compressed format for many fieldbus variables.</td> </tr> <tr> <td>CANalyze-MDF^[754]</td> <td>CANAlzser-MDF format for CAN-Log data.</td> </tr> </tbody> </table> <hr/> <p> "imc FAMOS ZIP" and "CANalyze-MDF": Applicable with devices of the firmware group A^[146] (imc DEVICES).</p>	Format	Description	imc FAMOS ^[753]	Default format for all device variables.	imc FAMOS ZIP ^[753]	Compressed format for many fieldbus variables.	CANalyze-MDF ^[754]	CANAlzser-MDF format for CAN-Log data.
Format	Description								
imc FAMOS ^[753]	Default format for all device variables.								
imc FAMOS ZIP ^[753]	Compressed format for many fieldbus variables.								
CANalyze-MDF ^[754]	CANAlzser-MDF format for CAN-Log data.								
Exchange real-time value	<p> See "Exchanging values among the devices"^[488]"</p>								

9.6.2 Storage in the device

Setting the storage options for the channels is performed on the Setup page: "Analog/digital channels" in the dialog: "Data transfer".

Saving to the internal storage medium

You can save the measured data directly in the device. For this purpose, an appropriate data storage medium is necessary (see "[Storage media in the device](#)"³³⁷).



By default, data saving by the device is deactivated. To activate the data saving, set a check in the box for **Save data**.

The channels are saved to the internal device drive under the current experiment name according to the [folder structure](#)³²⁸ set.

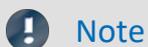
Storage location

There are several options for saving data by the device:

- Saving on a **hard drive**, which is built into the device housing.
- Saving on a **removable data carrier** which can only be exchanged while the device is deactivated (**not Hotplug-capable**)
- Saving on a **removable data carrier** which can be exchanged during operation (**Hotplug**)
- Saving on a [network computer](#)³¹⁴ (network enabled)

Which possibilities are available depends on the device hardware configuration and type.

To set the storage location, open the dialog: "Storage" on the Setup page: "Devices".



Note

The maximum file size: 2 GB

Per interval or measurement, one file is created per channel, which contains the respective measurement files.

If this file reaches **2 GB** in size, then a new measurement folder with a new file is created automatically. This folder has the same name as the first one, however it is appended with a suffix 002, e.g. "2024-02-29 14-30-00.002".

If this new file, too, reaches 2 GB in size, then a new file is created again, this time with the suffix 002.

Data transfer - Access via Windows Explorer

For devices belonging to the [firmware group B](#) (imc DEVICecore):

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

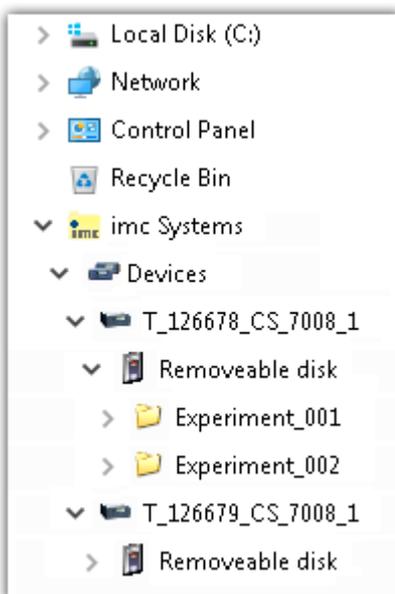
\\10.0.12.70

For devices belonging to the [firmware group A](#) (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](#)" (25) 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](#)" (343)). Subsequently, you are able to navigate to the desired data in the storage medium and thus to work with them.

Reference

Data transfer

Additional information and notes on transferring the data from the intern hard drive to the PC is presented in the section:

"Storage media in the device" > "[Data transfer - firmware group B \(imc DEVICEcore\)](#)"³³⁸

"Storage media in the device" > "[Data transfer - firmware group A \(imc DEVICES\)](#)"³⁴²

Storage Options

Information on the options for storage of the measured data is presented in the section: "[Storage Options - Settings for the Channels](#)"³⁰⁸.

9.6.2.1 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](#)¹⁴⁶ (imc DEVICES). Devices belonging to the [firmware group B](#)¹⁴⁶ (imc DEVICEcore) do not use the "*imcSyslog*" file.

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](#)⁴⁷⁵ 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!
```

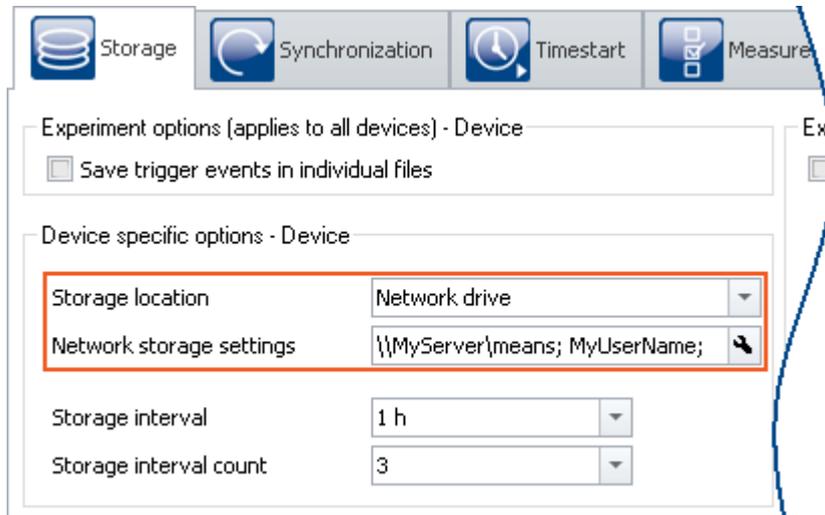
The structure is designed so that the information can be both read and also processed by a repair program.

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

9.6.3 Storage on a network drive

With the storage with the device, you can also specify a network drive as the target. I.e. the device can independently save the measured data to the network without that the program is running on the PC.

However, the device must support network data storage, otherwise the option is not available (see "[Device overview](#)"^[146]).



"Storage" dialog
Network storage

The data are not saved in the device, but on a centralized data collector.

Systems which use the Server Message Block (SMB) can be addressed as the target location (see "[Prerequisites](#)"^[315]).

In order to set data saving on a network drive, open the dialog: "Storage" on the Setup page: "Devices".

Fundamental differences from data storage on the PC:

- In contrast to storage on the PC, it is not necessary for the measured data to be saved in the imc STUDIO database with the experiment.
- The measurement results are not displayed as a saved measurement in the Data Browser.
- Technically, there is an additional difference between data storage on the PC and data storage in the device/network drive:
 - With data storage on the PC, the operating program handles the data saving.
 - Data storage on the network drive is administered by the measurement device.

This means it is no longer necessary for the operating software to be running during measurement.

Advantages:

- The operating program does not need to be running during the measurement.
- Data storage in a PC via the network thus now also works in Autostart and in Sleep/Resume modes (e.g. [imc BUSDAQ](#)^[146]).
- Centralized data storage, independent of the experiment's storage location.
- Copying of the internal drives to the central data storage area following a measurement is omitted.

Prerequisites:

- A device that supports network storage (see "[Device overview](#)"¹⁴⁶)
- Sharing of a network drive with writing permissions
- Secure network
- Target system using the Server Message Block (SMB) (formerly known as: Common Internet File System (CIFS)): LINUX, WIN2000 and Windows XP/Vista/7/8/10

Setting

The network drive is specified by means of the target PC's IP address and the name of the folder shared. The network storage path must be stated in UNC notation (\\server\share\ - DNS or only IP address possible).

Dialog "Storage"
Network storage settings

Ensure that write access to the network storage path (clearance, "share") is possible with the specified user name and password. If you did not protect the target system with a password, leave the field blank (WIN2000 and Windows XP/Vista/7/8/10 do not support this at present).

The measurement instrument creates a subfolder on the network drive for the data, which is composed of the device's name and the experiment's name.

The "*Waiting time [s]*" refers to the maximum time available for mounting the network drive when preparing the measurement.



Example

The following network path is selected (\\MyServer\means)

The device has the name: **imcCronosPL_123368**

The experiment has been saved under the name: **Test1**

When you start the measurement, the measurement results are saved to the following path:

\\MyServer\means\imcCronosPL_123368\Test1.



Note

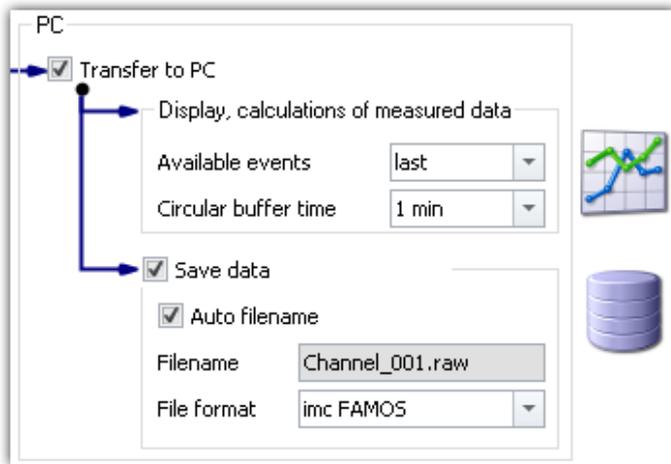
If the target PC is not available, the system reports a WINSOCK error. This disables the device for approx. 5 minutes and can only be prematurely canceled by rebooting.

9.6.4 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](#)³²⁸. Additionally, it is possible to **personally adapt how and where data are saved** using the [options](#)³³⁰.

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](#)"^[308].
- Information on display of the stored measurement data is presented in the chapter: "[Panel](#)"^[800].
- [Storage media in the device](#)^[337]

9.6.5 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](#)"^[731].

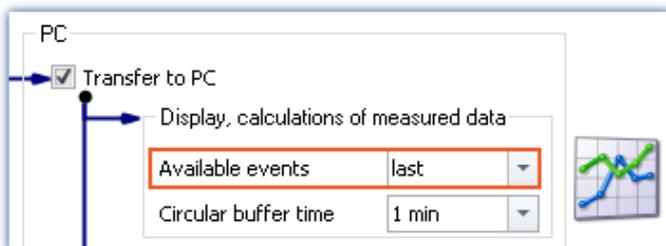
Events in data storage

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](#)"^[187] > "[Save trigger events in individual files](#)"^[329])

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)	<p>With this setting, you determine whether all events (triggered events) can be displayed, or only the last one.</p> <ul style="list-style-type: none"> • 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).

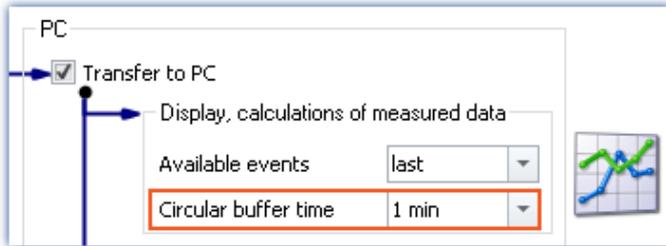
Circular buffer

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).



- 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.

9.6.6 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](#)³²¹: 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](#)³⁰⁸ on the PC or in the device for the channels desired!

Storage interval / Storage interval count

Parameter	Description
Storage interval / Storage interval count	<p>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.</p> <p>Storage interval:</p> <p>After this much time elapses, a new file is created for the duration of the next saving interval.</p> <p>Storage interval count:</p> <p>Save the last x intervals sets the desired number of intervals. Once this number is reached, the oldest interval is deleted.</p> <p>Multiplying the count and the interval determines which measurement duration is available after the measurement has been run.</p> <p>Example:</p> <p>Storage interval = 5 min, Storage interval count = 12. The measurement duration is set to 24 h.</p> <p>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.</p>

 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.



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.

9.6.6.1 Saving interval on the internal device storage medium

By means of the settings "*Storage interval*" and "*Storage interval count*", you specify a defined storage space. A **limited count** produces the same data storage effect as **circular buffer storage**. Once the count specified has been reached, the **oldest data are deleted**.

Special feature of circular buffer operation (limited count of data saving intervals)

There is a special feature with **Storage in the device** if a defined interval count is set (not applicable if "*all*" is set). If the **free memory is already exhausted** before the required folder count has been reached, the **circular buffer saving is performed with fewer intervals!** However, this only happens if there are at least four intervals.

Implementing the circular buffer operation

- A loop is used to test whether there are at least four measurement-folders in the current experiment folder.
- The oldest folder is deleted and a check is made of whether sufficient memory is now free.
- If this is not the case, the process is repeated.
- As soon as sufficient memory is available, data recording continues.
- If only three or fewer folders are left, data recording to the storage medium stops and the error message "Disk full" is posted.



FAQ

Under what circumstances will circular buffer operation not work?

Answer:

1. If the storage medium is already full and a new experiment is to be created.
2. If a measurement is of indefinite length and no saving interval was set.
3. If the storage medium memory allotted to the current experiment is so little that it is exhausted before at least four folders are saved. This is true even if the folders require different amounts of memory.

Why must there be at least four folders?

Answer:

1. For ring-operation, only the storage medium memory is used which was occupied by old folders belonging to the same experiment, or which is still free.
As a rule: no data belonging to another experiment are deleted.
2. At all times, at least two closed folder containing readable data must exist.
3. In consequence, the current folder doesn't count, because it may not contain data. Then, if at least two valid folder must exist when deletion occurs, three folders must be present before deletion. Together with the current folder, that makes four folders necessary before deletion can take place!



Note

- The number of samples per saving interval may vary, especially for heavy data throughput. However no sample will be lost.
- A concluded interval is denoted by the file "[DirClosed](#)₃₁₂".

9.6.7 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 WAVE 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 ³⁰⁹ (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](#)" ⁴⁷⁵
- "[Avoidance of Data Overflow](#)" ⁴⁷⁵

9.6.7.1 RAM time duration for imc DEVICeore devices

For devices belonging to the [firmware group B](#) ¹⁴⁶.

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 DEVICeore-devices. These settings only need to be considered when using [imc DEVICES-devices](#) ³²⁴.



Reference

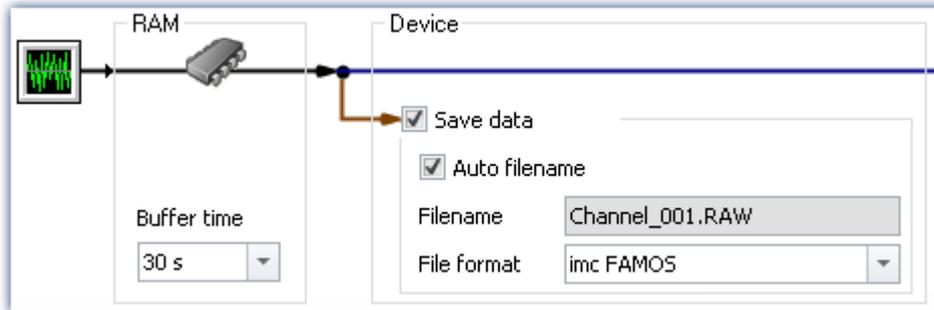
Time-stamped data

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](#)" ⁷³².

9.6.7.2 RAM time duration for imc DEVICES devices

For devices belonging to the [firmware group A](#) ¹⁴⁶.

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](#)" ⁷³².

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.

9.6.8 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.

9.6.9 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:

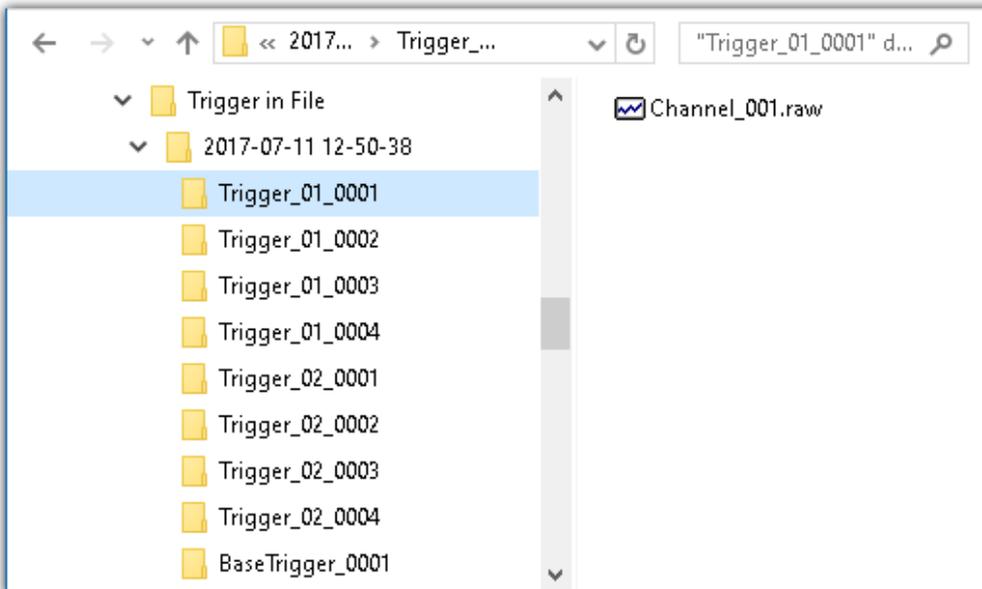
Options		Description
Save trigger events in individual files ³²⁹	Setup-page: " <i>Devices</i> ", on the dialog: " <i>Data transfer</i> "	Each trigger firing can be saved to its own separate file. For this purpose, additional subfolders are created for each trigger firing.
Customized storage location for measurement data ³³⁰	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 .

9.6.9.1 Save trigger events in individual files

The option: "[Save trigger events in individual files](#)¹⁸⁷" 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 ³¹⁷ ").	<ul style="list-style-type: none"> 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.
<pre>\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</pre>	<pre>\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</pre>

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](#)⁸⁰⁵" 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.

9.6.9.2 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](#)^[328] 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](#)"^[1238].

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 ^[328] .
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 or a "[Comment](#)" dialog 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.

9.6.10 Controlling data storage

9.6.10.1 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 in the "Execute menu action" command: "Save current data / Export current data".

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](#)²⁶⁸ 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](#)¹²⁴¹".

Options

In the [Options](#) ¹²⁴¹, 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 your own personal mapping instruction according to which to determine the channel type selection.

9.6.10.2 Close folder - imc Online FAMOS function `CloseSaveInterval`

If imc Online FAMOS is set to control commands, you'll find the function `CloseSaveInterval` in group **system**. That command is closing the current folder of measurement data and create the next folder. The function is executed with a delay, that depend on the load of the device.

9.6.11 Information and tips

9.6.11.1 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](#)^[1225]: e.g. <MEASUREMENTS["MEASUREMENT#LAST"].PATH>
- "[imc FAMOS-command](#)^[1149]" (data source)
- "[Set Measurement Number](#)^[1166]"-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".

9.6.11.2 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](#)^[312]". As soon as it is present, you have reliable access to the measured data.

9.7 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 WAVE.

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](#)"^[146]). Card-Bus interface are not recommended.

The storage media are exclusively for data acquisition with imc WAVE.

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](#)"^[146]), a distinction is made between how the storage media are to be handled.

- [For devices of the firmware group B \(imc DEVICEcore\)](#)^[337]
- [For devices of the firmware group A \(imc DEVICES\)](#)^[340]

9.7.1 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**.

9.7.1.1 Storage media

Storage media	Description
Internal hard disk	<p>Applicable to devices having a hard drive (see "Device overview"^[146]).</p> <ul style="list-style-type: none"> • 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	<p>Concerning devices with microSD slot (see "Device overview"^[146]).</p> <ul style="list-style-type: none"> • 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.

9.7.1.2 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
```

9.7.1.3 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](#)³³⁹ 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.

9.7.1.3.1 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 WAVE in the folder "*Tools\RemoveableDriveFormatter*":

- "*rufus-4.1p.exe*"

This program can be used for the purpose of formatting the data volume.

Settings required for formatting by means of Rufus 4.1

Make the **following settings** for the memory card:

- Boot selection: "*Not bootable*"
- Partition scheme: "*MBR*"
- File system: "*Large FAT32*"
- Cluster size: "*16 kilobytes*"



Note

Only **one(!)** partition may be created. Multiple partitions may cause the measuring device not to recognize the storage medium.

9.7.1.4 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	<p>Errors can have the following causes, among others:</p> <ul style="list-style-type: none"> • The data rate is too high, the storage medium can't keep up and data overflow results. • The storage medium is full. <p>If the PC is connected to the measuring device, imc WAVE documents the error with a message in the Logbook.</p>

9.7.2 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](#)"³²²). 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.

9.7.2.1 Storage media

Storage media	Description
CF Cards (Compact Flash)	<p>For devices of group A4 and A5 ^[146]:</p> <p>The device group exclusively uses CF cards for storage medium.</p>
USB Storage Medium	<p>Concerning devices with USB (see "Device overview" ^[146]). Memory sticks or external hard drives can be connected at this terminal.</p> <ul style="list-style-type: none"> • Do not use multiple storage media at the same time! Devices belonging to group A6 ^[146] 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 ^[340] 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! <div style="border: 1px solid orange; padding: 5px; margin-top: 10px;"> <p style="text-align: center;">Do not use USB hard drives with external power supply</p> <p> 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.</p> </div>
ExpressCard	<p>Applicable to devices having ExpressCard slot (see "Device overview" ^[146]).</p> <ul style="list-style-type: none"> • Remove any connected USB storage medium! Devices belonging to group A6 ^[146] 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 ^[340] functionality is available.
CFAST	<p>Concerning devices with CFAST slot (see "Device overview" ^[146]).</p> <ul style="list-style-type: none"> • 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 ^[340] functionality is available.
SSD	<p>Applicable to devices having a hard drive (see "Device overview" ^[146]).</p> <ul style="list-style-type: none"> • With SSD hard drives, Hot-Plug ^[340] 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 ^[342]. • 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 ^[345]. • In addition to the SSD, a CF/CFAST-card can be inserted in the measurement device and used alternatively.

9.7.2.2 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

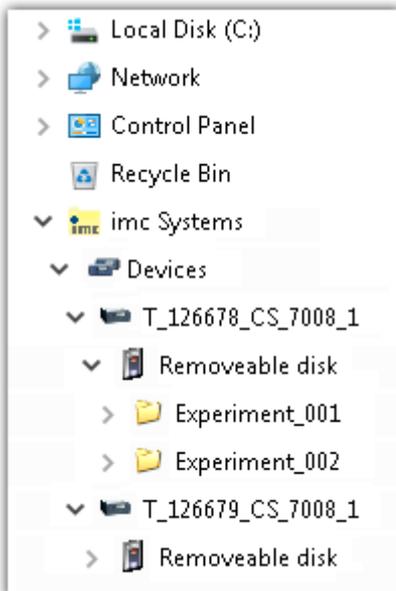
A tip on **interval saving**: Supposing the system's power supply suffers an outage during a measurement, it would not be possible to guarantee that the data file on the storage medium is terminated properly. This may lead to a failure to record the last measurement taken. Saving at intervals is a way to reduce this risk (see section "[Saving Interval](#)"^[318]).

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](#)"^[25] 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.

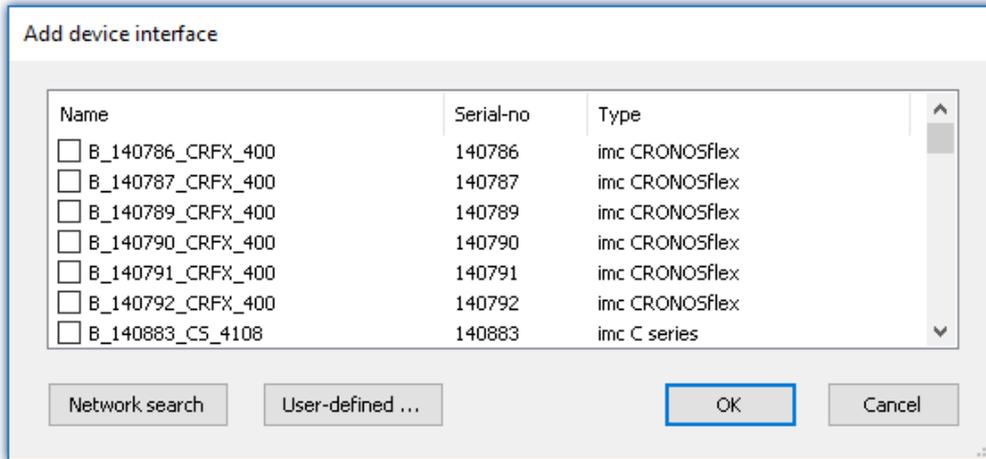
In order to get access to your device's storage medium, you must first add it to the tree diagram (see "[imc Systems - Adding a device](#)"^[343]). Subsequently, you are able to navigate to the desired data in the storage medium and thus to work with them.

9.7.2.2.1 imc Systems - Adding a device (Add)

Even if you have already been connected with the device by means of the imc WAVE 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	<p>"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.</p> <p>Select your measurement device and confirm your selection with "OK". The measurement device is then available.</p>
User-defined	<p>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.</p>

Reference

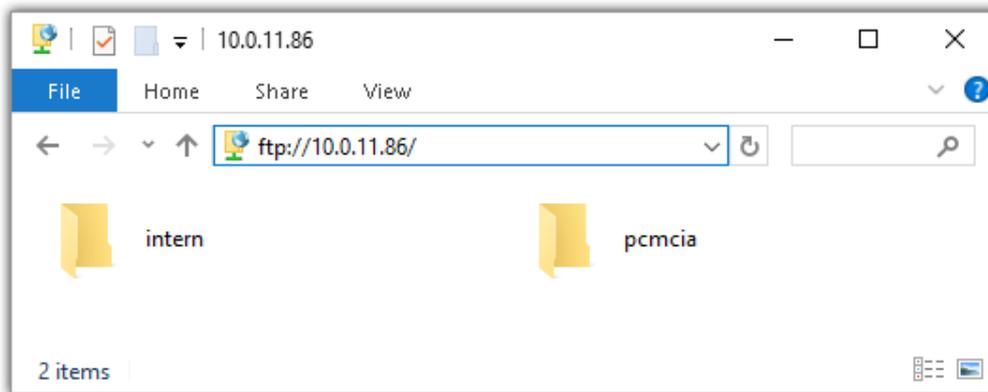
More information is presented in the section: "[Setting Up - Connect the device](#)"

- General: "[Device connection via LAN](#)"⁴⁹
- "[Connection via a direct address](#)"⁵²

9.7.2.2.2 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](#) 154.

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).

9.7.2.3 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](#)^[345] 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](#)"^[345].
- 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](#)^[318].



Notes

General restrictions applicable to file systems

Please observe the general restrictions regarding the respective file systems.

9.7.2.3.1 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](#)^[345] 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**

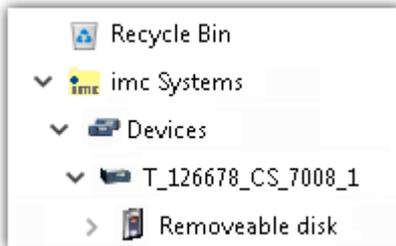
If a few channels having a high data rate are being recorded, then **large clusters** on the data carrier provide better advantage. Formatting with FAT32 on the PC and drive sizes < 8 GB creates disadvantageously small clusters, which in conjunction with the full aggregate sampling rate can lead to a 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](#)"^[342] 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!*".

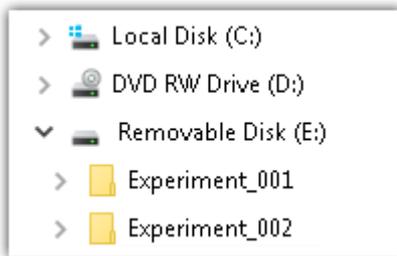
The device performs the formatting according to the following rule:

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](#)**³⁴⁴, 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](#)³⁴⁵.

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.

9.7.2.4 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 ^[7] .
The system won't recognize the data storage drive	<ul style="list-style-type: none"> • Answer 1: Check the file system: The device supports FAT32/FAT16 ^[345]. • 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	<p>Errors can have the following causes, among others:</p> <ul style="list-style-type: none"> • The data rate is too high, the storage medium can't keep up and data overflow results. • The storage medium is full. <p>The device signals any error by flashing this LED. Its further responses depend on whether or not the device is connected to the PC.</p> <ul style="list-style-type: none"> • 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 WAVE 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	<ul style="list-style-type: none"> • With a storage medium formatted by Windows to FAT32 ^[345], 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 ^[345], data overflow can occur if a high aggregate sampling rate is generated by very many low-speed channels.

9.8 Measurement types

9.8.1 Temperature measurement

Two methods are available for measuring temperature. Measurement using a **PT100** requires a constant current, e.g. of 250 μA to flow through the sensor. The temperature-dependent resistance causes a voltage drop which is correlated to a temperature according to a characteristic curve.

When measuring with **thermocouples**, the temperature is determined via the series of voltages of different alloys. The sensor generates a temperature-dependent voltage which is relative to the terminal point on the plug. To find the absolute temperature, the temperature of the terminal point must be known. This is determined with a **PT1000** directly in the terminal plug and requires a special plug type.

The measured voltage is converted into the displayed temperature value according to the characteristics of the temperature scale IPTS-68.



Note

Making settings with imc software

A temperature measurement is a voltage measurement whose measured values are converted to physical temperature values by reference to a characteristic curve. The characteristic curve is selected using the "Correction" parameter on the "Measurement mode" tab. Amplifiers which enable bridge measurement, must first be set to the "Voltage" measurement mode in order for the temperature characteristics curves to be available for selection.

9.8.1.1 Thermocouples as per DIN and IEC

The following standards apply for thermocouples, in terms of their thermoelectric voltage and tolerances:

Thermocouple	Symbol	max. temp.	defined up to	(+)	(-)
DIN IEC 584-1 (2014-07)					
Iron-constantan (Fe-CuNi)	J	750°C	1200°C	black	white
Copper-constantan (Cu-CuNi)	T	350°C	400°C	brown	white
Nickel-chromium-Nickel (NiCr-Ni)	K	1200°C	1370°C	green	white
Nickel-chromium-constantan (NiCr-CuNi)	E	900°C	1000°C	violet	white
Nicrosil-Nisil (NiCrSi-NiSi)	N	1200°C	1300°C	red	orange
Platinum-Rhodium-platinum (Pt10Rh-Pt)	S	1600°C	1760°C	orange	white
Platinum-Rhodium-platinum (Pt13Rh-Pt)	R	1600°C	1760°C	orange	white
Platinum-Rhodium-platinum (Pt30Rh-Pt6Rh)	B	1700°C	1820°C	n.a.	n.a.
DIN 43710					
Iron-constantan (Fe-CuNi)	L	600°C	900°C	red	blue
Copper-constantan (Cu-CuNi)	U	900°C	600°C	red	brown

If the thermo-wires have no identifying markings, the following **distinguishing characteristics** can help:

- Fe-CuNi: Plus-pole is magnetic
- Cu-CuNi: Plus-pole is copper-colored
- NiCr-Ni: Minus-pole is magnetic
- PtRh-Pt: Minus-pole is softer

The color-coding of compensating leads is stipulated by DIN 43713. For components conforming to IEC 60584: **The plus-pole is the same color as the shell; the minus-pole is white.**

9.8.1.2 PT100 (RTD) - measurement

RTD (PT100) sensors can be directly connected in 4-wire-configuration. An additional reference current source feeds a chain of up to 4 sensors in series.

With the imc Thermo plug, the connection terminals are already wired in such a way that this reference current loop is closed.

Note

If fewer than 4 PT100 units are connected, the current-loop must be completed by a wire jumper from the "last" RTD to -I4.

If you dispense with the "support terminals" ($\pm I1$ to $\pm I4$) provided in the imc Thermo plug for 4-wire connection, a standard terminal plug or any DSUB-15 plug can be used. The "current loop" must then be formed between +I1 (DSUB Pin 9) and -I4 (DSUB Pin 6).

9.8.1.3 imc Thermo plug (T4)

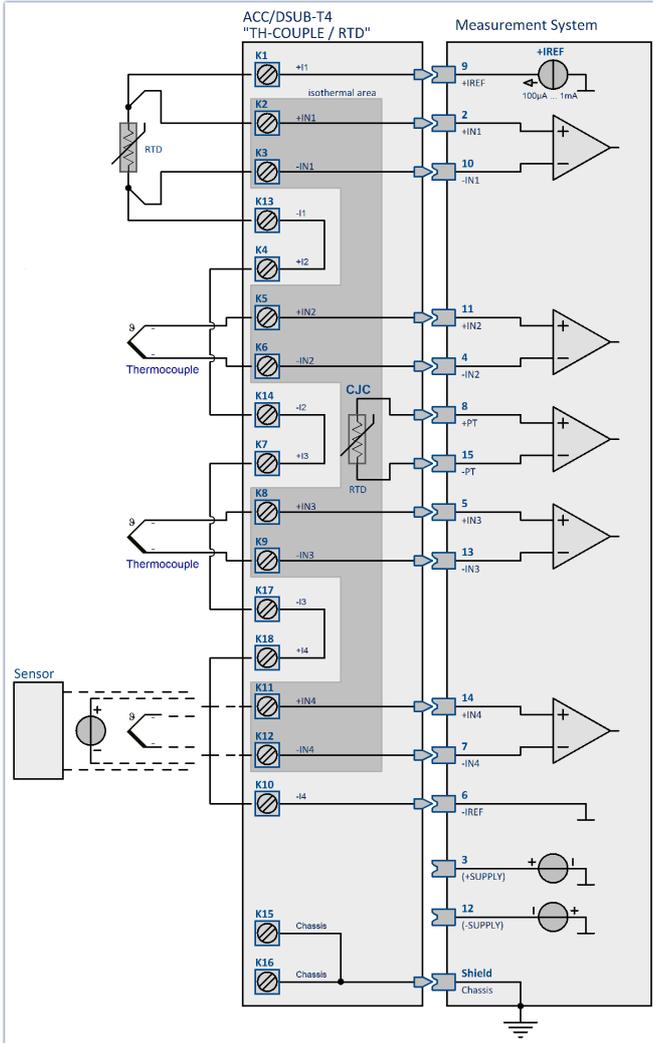
The imc Thermo plug contains a screw terminal block in a DSUB-15 plug housing with a built-in temperature sensor (PT1000) for **cold junction compensation**. This provides for direct connection of thermocouples of any type, directly to the differential inputs (+IN and -IN) without external compensation leads. That plug can also be used for **voltage** measurement.

The difficulty with thermocouple measurements are the "parasitic" thermocouples which inevitably form where parts of the contacts made of different materials meet. The temperature sensor measures the temperature at the connection terminal and compensates the corresponding "error"-voltage. Normally, the connection to this compensation point (inside the device) is made by special compensation leads or plugs made of material identical to the respective thermocouple type, in order not to create additional (uncontrolled) parasitic thermocouples.

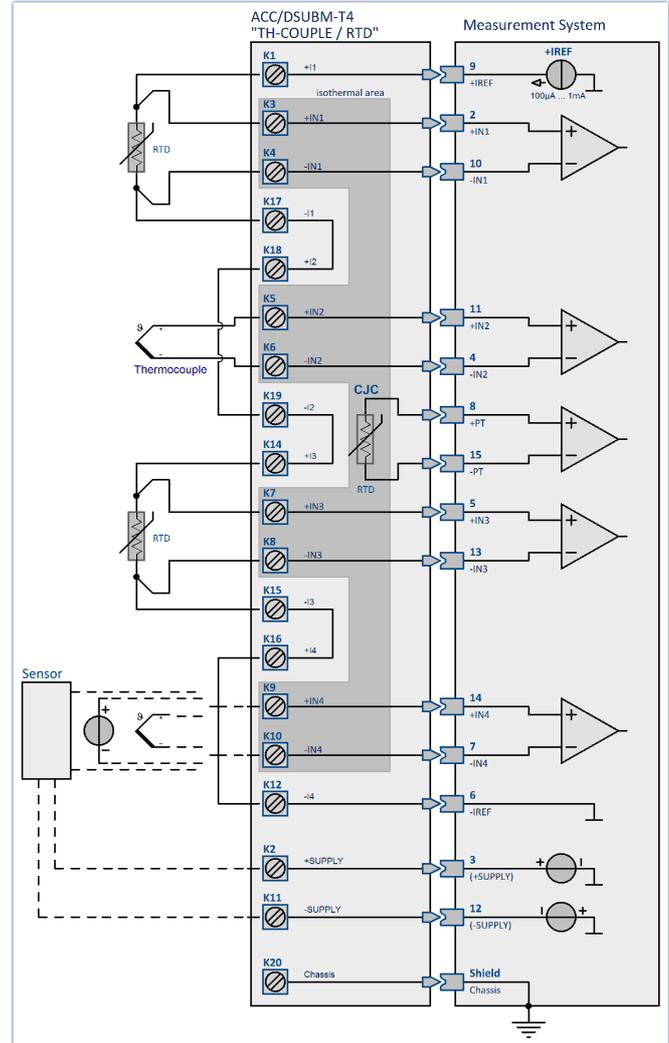
imc's system avoids the problem through the use of individual compensation sensors directly inside the plug, thus offering an especially simple, flexible and cost-effective connection solution.

9.8.1.3.1 Schematic: T4 plug

Plastic plug (ACC/DSUB-T4, discontinued)



Metal plug (ACC/DSUBM-T4)



9.8.2 Bridge measurement

Bridge channels are for taking readings from **measurement bridges** such as resistor bridges or strain gauges. The channels are equipped as **non-isolated differential** amplifiers and can alternatively be used for direct **measurement of voltages**.

There is a distinction among the following operating modes:

➤ **Sensor**

- Full bridge
- Half bridge
- Quarter bridge (120 Ω)

➤ **Strain gauge**

- Full bridge with 4 active strain gauges in uniaxial direction
- Full bridge with Poisson strain gauge in adjacent bridge arms
- Full bridge with Poisson strain gauge in opposing bridge arms
- Half bridge with one active and one passive strain gauge
- Half bridge with 2 active strain gauges in uniaxial direction
- Poisson half bridge
- Quarter bridge with 120 Ω strain gauge

9.8.3 Bridge measurements with wire strain gauges

Strain in this sense refers to the ratio of a body's original length to the change in length due to a force exerted upon it.

$$\varepsilon = \frac{dL}{L}$$

By selecting "Strain gauge" as the measurement mode, common bridge circuits and configurations for wire strain gauges are offered for selection. The scaling can be adjusted in terms of typical parameters for strain measurements such as the gauge factor or Poisson's ratio, the transversal expansion coefficient.

If a strain gauge adheres to a test object, the strain on the object is transmitted to the bridge circuit. The changes in the lengths of the bridge arms cause their impedances to change. There is a correlation between the changes in length and the changes in resistance:

$$\varepsilon = \frac{dL}{L} = \frac{dR/R}{k}$$

Legend:	
ε	strain
dL	change in length
L	original length
dR	change in resistance
R	resistance of strain gauge
k	Gauge factor, describing the ratio of relative length change to change in resistance

The changes in resistance caused by the strain are very small. For this reason, a bridge circuit is used to translate these changes into voltage changes. Depending on the circuit, from one to four strain gauges can be employed as bridge resistors.

Assuming that all bridge resistors have the same value, we have:

$$V_a = V_e \cdot \frac{dR}{4 \cdot R} = \frac{V_e}{4 \cdot R} \cdot k \cdot \varepsilon$$

Legend:	
V_a	measurement voltage
V_e	excitation voltage

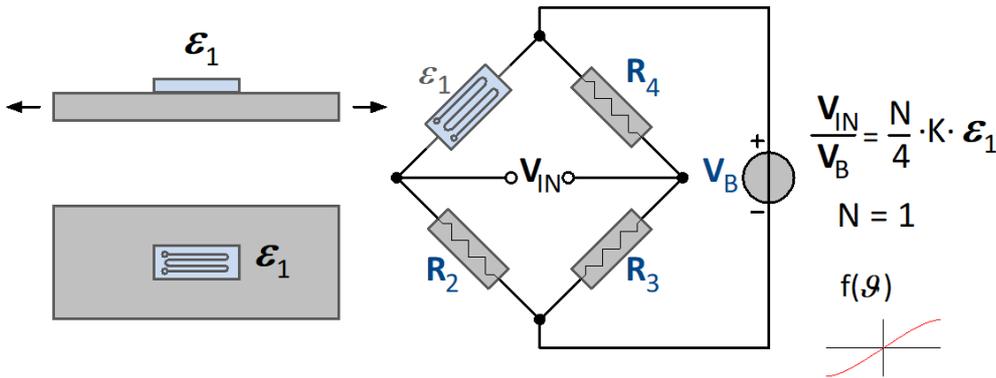
$$\varepsilon = \frac{V_a \cdot 4}{V_e \cdot k}$$

For concrete measurement tasks, the arrangement of the strain gauges on the test object is important, as well as the circuitry of the bridge. On the "Bridge circuit", you can select from among typical arrangements. A graphic below shows the position on the test object and the bridge circuitry. Notes on the selected arrangement are displayed in the text box.

Note

For an easier operation, ranges that are unsuitable are hidden in the operating software.

9.8.3.1 Quarter bridge for 120 Ohm strain gauge



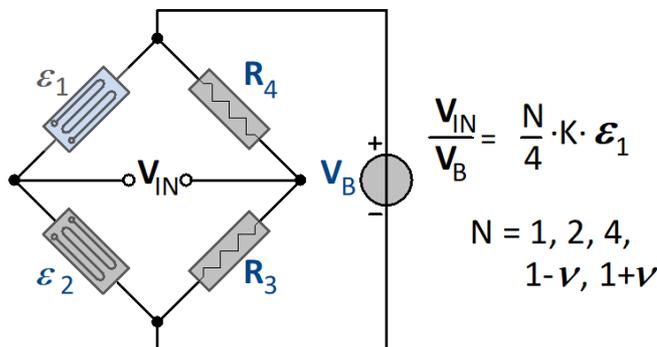
This strain gauge arrangement uses an **active** strain gauge which is positioned on the test object in a uniaxial stress field. This strain gauge is joined by three passive resistors within the module to form a full bridge. The strain gauge can have a resistance value of 120 Ω.

This arrangement does not come with temperature compensation. The strain is computed as:

$$\varepsilon \left[\frac{\mu\text{m}}{\text{m}} \right] = \frac{4 \cdot 1000}{k} \cdot \frac{V_a}{V_e} \left[\frac{\text{mV}}{\text{V}} \right]$$

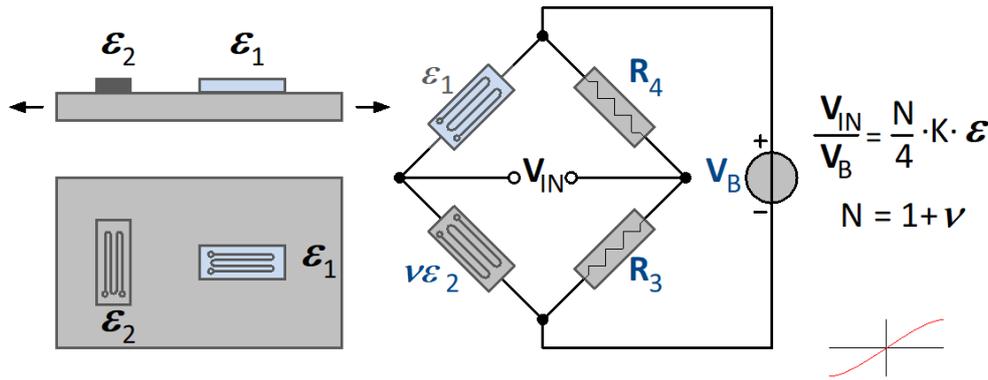
Legend:	
k	gauge factor

9.8.3.2 General half bridge



General half bridge with bridge completion in measurement device. N has to be set from a list.

9.8.3.3 Poisson half bridge

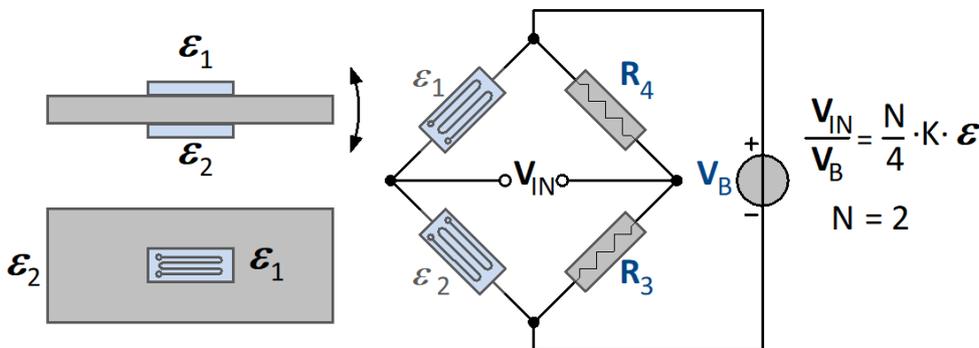


In this circuit, two active strain gauges are used. The strain gauge is positioned transverse to the main direction of strain. The transversal contraction is exploited. For this reason, the Poisson's ratio for the material, which is its transversal expansion coefficient, must be supplied along with the gauge factor. This circuit offers good temperature compensation. The strain is computed as:

$$\varepsilon \left[\frac{\mu\text{m}}{\text{m}} \right] = \frac{4 \cdot 1000}{k \cdot (1 + \nu)} \cdot \frac{V_a \text{ [mV]}}{V_e \text{ [V]}}$$

Legend:	
k	gauge factor
ν	Poisson's ratio of test object material

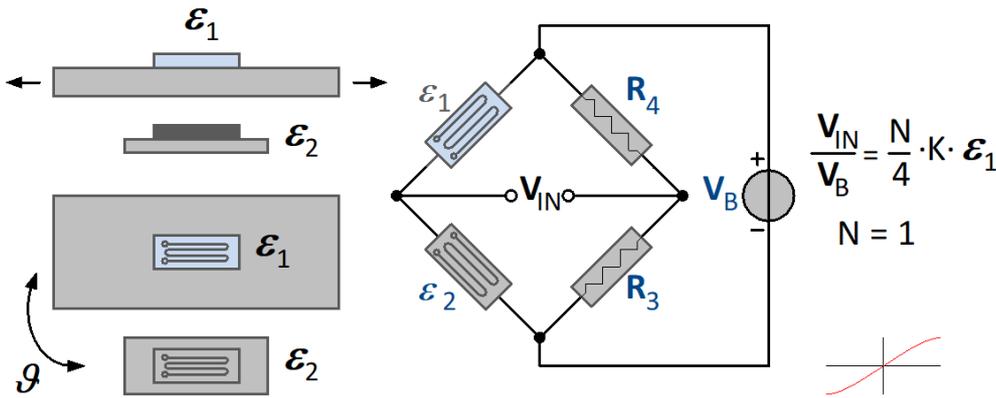
9.8.3.4 Half bridge with two active strain gauges in uniaxial direction



Two active strain gauges are placed under stress in opposite directions but equal magnitude, i.e. one strain gauge is under compression and another under equal tension. (bending beam circuit). This arrangement doubles the measurement's sensitivity to a bending moment. On the other hand, longitudinal force, torque and temperature are all compensated for. The strain is computed as:

$$\varepsilon \left[\frac{\mu\text{m}}{\text{m}} \right] = \frac{4 \cdot 1000}{2 \cdot k} \cdot \frac{V_a \text{ [mV]}}{V_e \text{ [V]}}$$

9.8.3.5 Half bridges with one active and one passive strain gauge

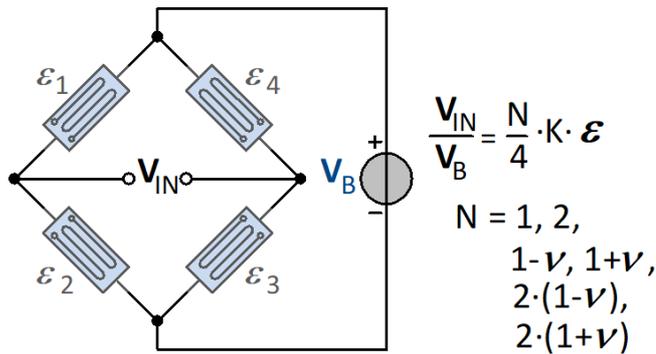


This circuit involves strain gauges. The first one is positioned on the test object, the second on a sample of the same material under the same ambient temperature and serves the purpose of temperature compensation. The strain is computed as:

$$\epsilon \left[\frac{\mu\text{m}}{\text{m}} \right] = \frac{4 \cdot 1000}{k} \cdot \frac{V_a \left[\text{mV} \right]}{V_e \left[\text{V} \right]}$$

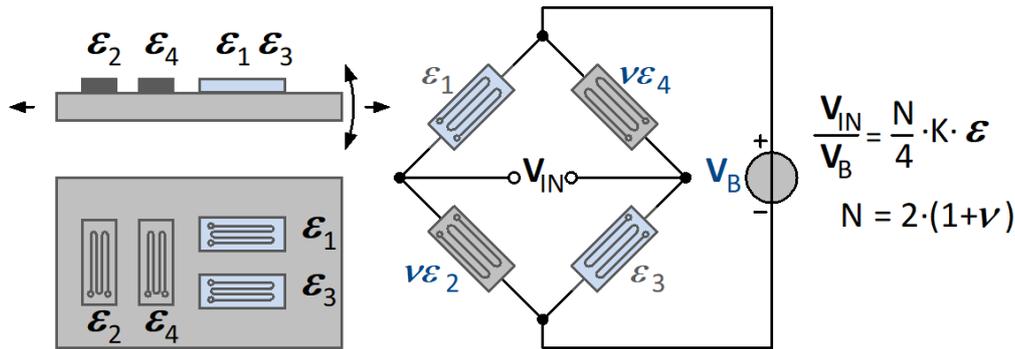
Legend:	
k	gauge factor

9.8.3.6 General Full bridge



General full bridge. N has to be set from a list.

9.8.3.7 Full bridge with Poisson strain gauges in opposed branches

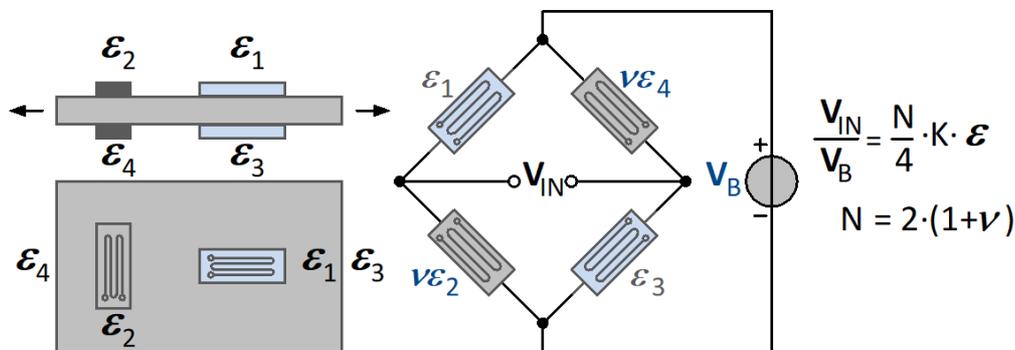


Two active strain gauges are positioned along the longitudinal strain and are joined by two transversally positioned strain gauges to complete the bridge (torsion bar arrangement). In the bridge, the longitudinal strain gauges are located in opposite branches. This circuit provides better exploitation of transversal contraction and longitudinal force as well as good temperature compensation. In this arrangement, the transversal expansion coefficient must be specified. The strain is computed as:

$$\epsilon \left[\frac{\mu\text{m}}{\text{m}} \right] = \frac{4 \cdot 1000}{2 \cdot k \cdot (1 + \nu)} \cdot \frac{V_a \text{ [mV]}}{V_e \text{ [V]}}$$

Legend:	
k	gauge factor
ν	Poisson's ratio of test object material

9.8.3.8 Full bridge with Poisson strain gauges in adjacent branches

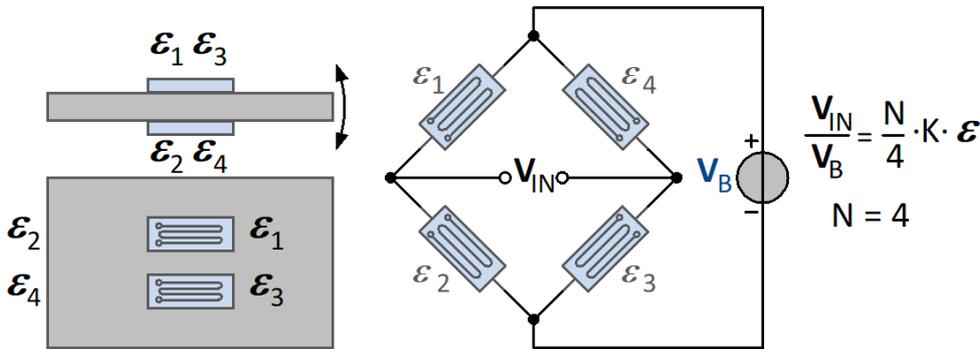


Full bridge with 4 active strain gauges. 2 active strain gauges complemented by 2 transverse Poisson strain gauges. They are located in opposed bridge arms. Higher exploitation of transverse contraction longitudinal expansion while providing good temperature compensation.

$$\epsilon \left[\frac{\mu\text{m}}{\text{m}} \right] = \frac{4 \cdot 1000}{2 \cdot k \cdot (1 + \nu)} \cdot \frac{V_a \text{ [mV]}}{V_e \text{ [V]}}$$

Legend:	
k	gauge factor
ν	Poisson's ratio of test object material

9.8.3.9 Full bridge with 4 active strain gauges in uniaxial direction



The circuit consists of 4 active strain gauges. Two are under compression and the others under equal tension. The strain gauges under tension are positioned in opposite bridge arms. The sensitivity to the moment of bending is increased. At the same time, longitudinal force, torque and temperature are compensated. The strain is computed as:

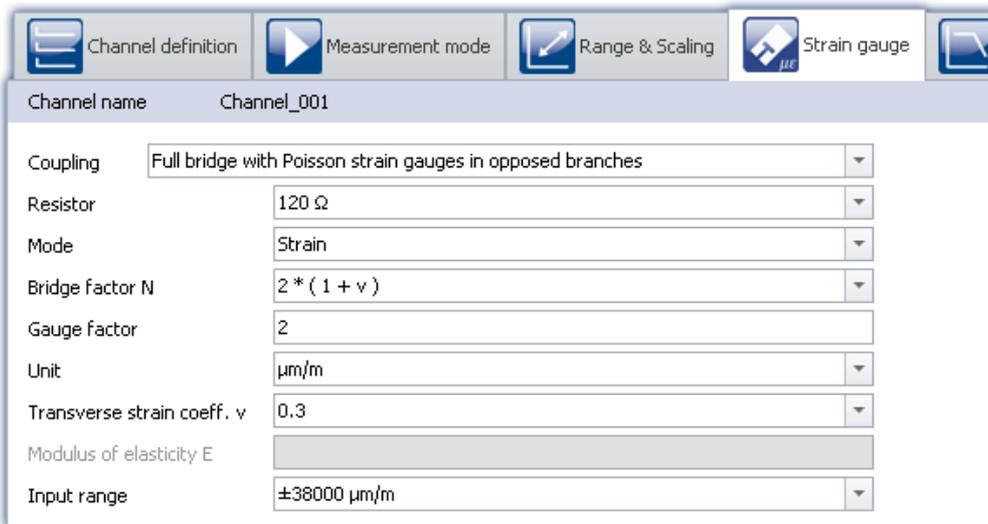
$$\epsilon \left[\frac{\mu\text{m}}{\text{m}} \right] = \frac{4 \cdot 1000}{4 \cdot k} \cdot \frac{V_a}{V_e} \left[\frac{\text{mV}}{\text{V}} \right]$$

Legend:	
k	gauge factor

9.8.3.10 Scaling for the strain analysis

It is possible to choose whether to determine the strain or the mechanical stress suffered by the part. In the range of elastic deformation, the axial stress (force / cross section) is proportional to the strain. The proportionality factor is the modulus of elasticity.

Mechanical stress = modulus of elasticity * strain (Hooke's law)



Gauge factor (K-factor)

The K-factor is the ratio by which the mechanical quantity (elongation) is transformed to the electrical quantity (change in resistance). The typical range is between 1.9 and 4.7. The exact value can be found in the spec sheet for the strain gauge used. If the value entered for this parameter is outside of this range, a warning message will appear but the module can still be configured.

Transverse strain coeff.

(poisson's ratio): If a body suffers compression or tension and is able to be freely deformed, then not only its length but also its thickness changes. This phenomenon is known as transversal contraction. It can be shown that for each kind of material, the relative change in length is proportional to the relative change in thickness D . The transversal elongation coefficient (Poisson's ratio) is the material-dependent proportionality factor. The material constant is in the range 0.2 to 0.5.

In bridge circuits where the strain gauges are positioned transversally to the main direction of strain, this constant must be supplied by the user. The ratios for various materials are available in the list box. These values are only for orientation and may need to be adjusted.

Elastic modulus

The elastic modulus E , is a material parameter characterizing how a body is deformed under the action of pressure or tension in the direction of the force. The unit for E is N/mm^2 . This value must be entered for the mechanical stress to be determined. The e -moduli for various materials are available in the list box. These values are only for orientation and may need to be adjusted.

Unit

When the strain is determined, the readings appear with the unit $\mu\text{m/m}$.

For the mechanical stress one can toggle between GPa and N/mm^2 .

$$1 \text{ GPa} = 10^3 \text{ N/mm}^2$$

Note that the elastic modulus is always in GPa .

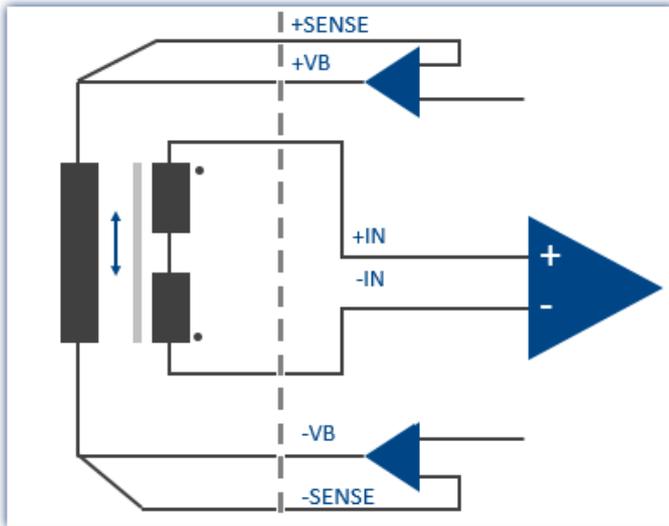
9.8.4 LVDT

LVDT and inductive displacement transducers are operated with carrier frequency mode bridge amplifiers such as C-60xx or C-60xx-1 or module BR(2)-4.

LDVT (Linear Variable Differential Transformer) and RVDT (Rotary Variable Differential Transformer) are based on the transformer principle with an actively driven primary coil. They are configured as full bridge circuit where the two inversely phased secondary windings are compensating each other in the neutral position. Depending on type and winding ratios, the typical output signals are in the range of 200 mV/V but can also exceed 1000 mV/V.

In contrast, inductive displacement sensors do not possess any galvanically isolated primary but consist of two differential coils, wound in phase, that form an inductive half bridge circuit. Their signal range is typical 80 mV/V.

Full-Bridge

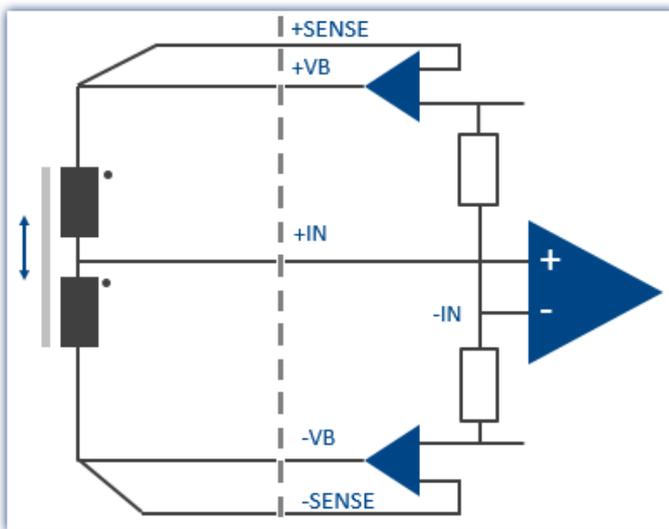


Typically 200 mV/V signal or even < 1000 mV/V.

Opposite winding directions of the two secondaries

Galvanically isolated primary / secondary

Half-Bridge



Typically 80 mV/V signal.

Uniform winding directions of the two coils.

9.8.5 Incremental Counters Channels

The incremental counter channels are for measuring **time** or **frequency**-based signals. In contrast to the analog channels as well as to the digital inputs, the channels are not sampled at a selected, fixed rate, but instead time intervals between slopes (transitions) or number of pulses of the digital signal are measured.

The **counters** used (set individually for each of the 4 channels) achieve time resolutions of up to 31 ns (32 MHz); which is far beyond the abilities of **sampling procedures** (under comparable conditions). The *sampling rate* which the user must set is actually the rate at which the system evaluates the results of the digital counter or the values of the quantities derived from the counters.

Note

Sampling rate for incremental counter channels

Only one sampling rate can be set per module.

9.8.5.1 Signals and conditioning

9.8.5.1.1 Mode

The various modes comprise the following measurement types:

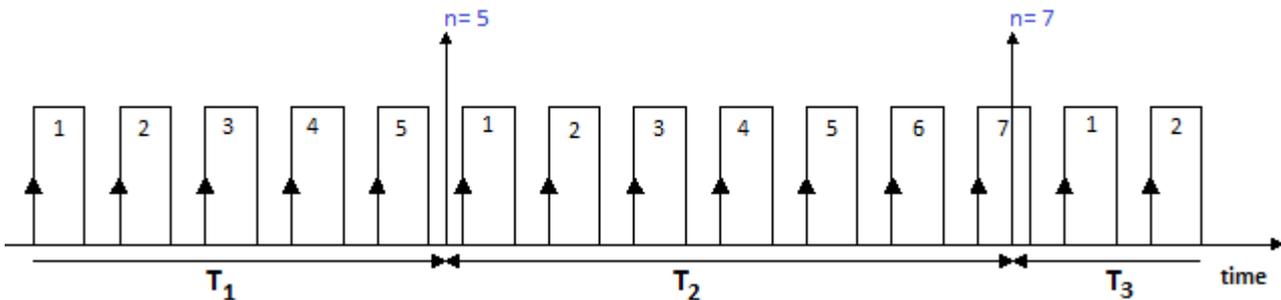
Event-counting	Time	Combined measurements
<ul style="list-style-type: none"> • events • distance(differential) • angle (differential) • angle (sum) • angle (abs 0-360°) • distance (sum) 	<ul style="list-style-type: none"> • time • pulse time 	<ul style="list-style-type: none"> • frequency • speed • RPM

Event-Counting

The following variables are derived from **Event counting**:

- [events](#)³⁶⁵
- [distance\(differential\)](#)³⁶⁵
- [angle \(differential\)](#)³⁶⁵
- [distance \(abs.\)](#)³⁶⁵
- [angle \(abs.\)](#)³⁶⁵

The amount of events occurring within one sampling interval is counted. The event counter counts the sensor pulses within the sampling interval. **An event is a positive edge in the measurement signal which exceeds a user-determined threshold value.**

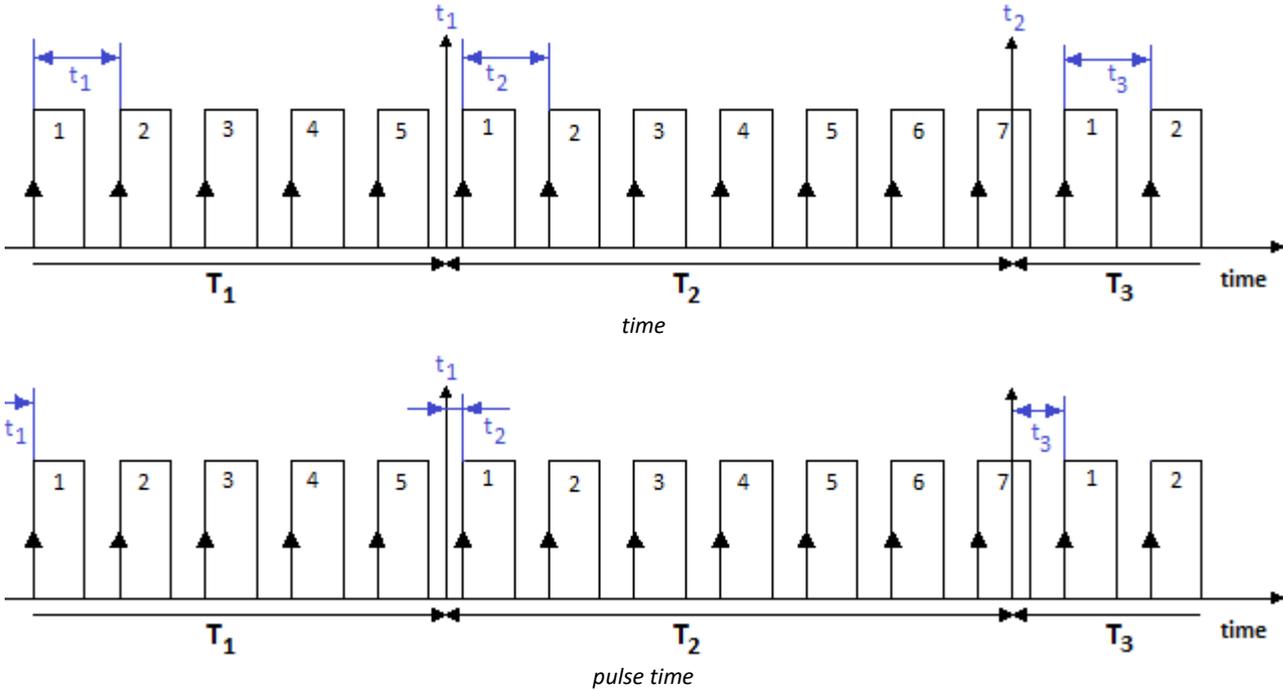


Time Measurements

Exclusive measurement of **time** is performed as:

- [time](#)^[366] (of two successive signal edges)
- [pulse time](#)^[367] (time from the beginning of one sampling interval until the next signal edge)

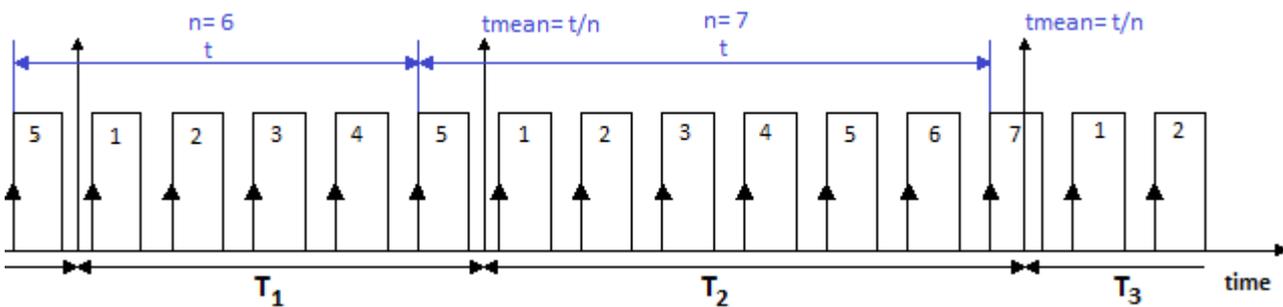
Any other pulses occurring within the sampling interval are not evaluated for these measurement types.



Combination Mode

Determining a frequency and the derivative quantities RPM and velocity is based on the **combination of event counting and time measurement**. In other words, during a sampling interval, the number of events occurring as well as the time interval between the first and last event are measured:

- [frequency](#)^[368]
- [speed](#)^[368]
- [RPM](#)^[368]



The frequency is determined as the number of events counted divided by the time between the first and the last "complete" event in the interval. An event is complete when a positive edge is succeeded by a subsequent positive edge.

The frequencies must lie within the bandwidth of the module used. If the maximum frequency is exceeded during a measurement, the system returns the input range end value instead of the true measured values.

The derivative quantities displacement and angle measurement have the following settings:

- Choice of [single-track and dual-track encoder](#)³⁶⁴
- Start of measurement with or without ["Zero impulse"](#)³⁶⁴
- Number of pulses (per unit)

The input ranges and resolutions for the RPM or velocity also depend on the number of encoder pulses set. If the number of pulses is known, the RPM and velocity values can easily be computed:

Parameter	Description
RPM	Input range = ([Frequency input range in Hz] * 60 / [Encoder pulses per revolution]) in RPM Resolution = ([Frequency resolution in Hz] * 60 / [Encoder pulses per revolution]) in RPM

Behavior in response to missing signal pulses

If a sequence of signal pulses is slowing down and then one sampling interval elapses without any pulse, no calculation can be performed for that sampling interval. In that case, the system assumes that the rotation speed is simply decreasing and an attenuating signal course is extrapolated. This "estimated" measurement value is then closer to the true value than the value determined from the preceding sampling interval. This technique has demonstrated its validity in practice.

Note

In extreme cases, the sensor does not return any more pulses, e.g. in case of a sudden outage. Then the algorithm generates an attenuation curve, meaning values > 0, even if the measurement object is actually no longer moving.

9.8.5.1.2 Measurement procedures

Measurement procedures	Description
Differential measurement procedures	The quantities derived from <i>event-counting</i> , Events , Distance and Angle denoted by the annotation (diff.) are "differential" measurements. The quantity measured is the respective change of displacement or angle within the last sampling interval. (positive or, for dual-track encoders, negative also) or the newly occurred events (always positive). If, for instance, the total displacement is desired, it must be calculated by integration of the differential measurements using imc Online FAMOS functions.
Cumulative measurements	The quantities derived from <i>event-counting</i> , Distance and Angle appearing with the annotation (abs.) are " cumulative " measurements. In cumulative measurements, the return value is the sum of all displacement or angle changes, or of all event which occurred.

9.8.5.1.3 Scaling

A **maximum** value must be entered under **Input range** (max. frequency etc, depend on mode). This **Maximum** determines the scaling factor of the computational processing and amounts to the range which is represented by the available numerical format of 16bits. Depending on the measurement mode (quantity to be measured), it is to be declared as an input range's unit or in terms of a corresponding max. pulse rate.

In the interest of maximizing the **measurement resolution** it is recommended to set this value accordingly.

The **Scaling** is a sensor specification which states the relation between the pulse rate of the sensor and it's corresponding physical units (sensitivity). This is also the place to enter a conversion factor for the sensor along with any physical quantity desired, for instance, to translate the revolutions of a flow gauge to a corresponding volume.

The table below summarizes the various **measurement types' units**; the **bold/cursive** letters denote the (fixed) primary quantity, followed by its (editable) default physical unit:

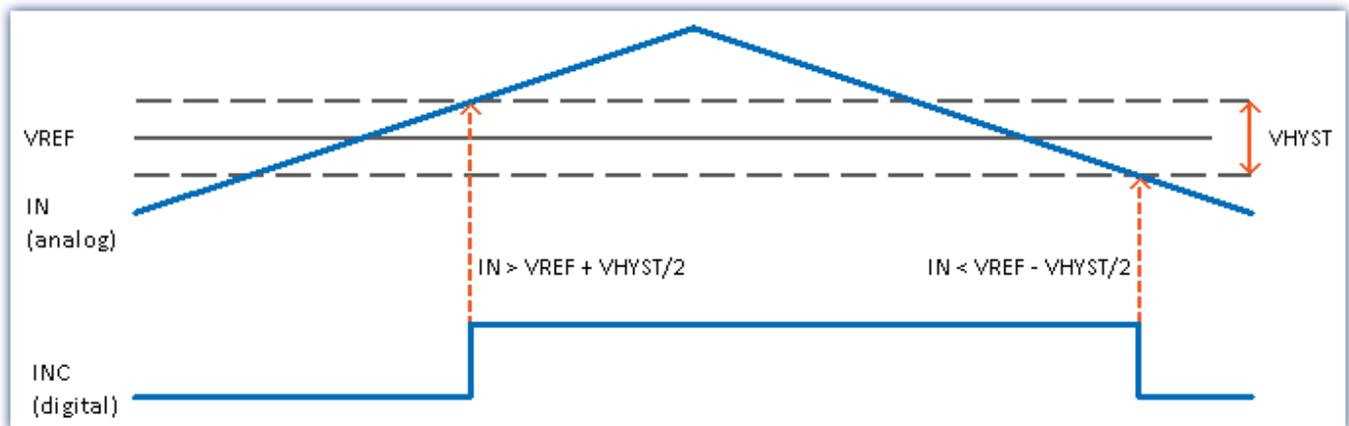
Measurement quantity	(Sensor-) scaling	Range	Maximum
Linear motion	pulse/m	m	m/s
Angle	pulse/U	U	U/min
Velocity	pulse/m	m/s	m/s
RPM	pulse/U	U/min	U/min
Event	pulse/pulse	1 pulse	Hz
Frequency	Hz/Hz	Hz	Hz
Time	s/s	s	s
Pulse time	Hz/code	Hz	Hz

9.8.5.1.4 Comparator conditioning

The incremental counters' special properties make **special demands for signal quality**: the very high resolution offered by the detector or counter means that even very short impulses can be captured and evaluated, which sampling-based measurement methods (such as for the digital inputs of the DI16 module) would not (or almost never) be able to detect. Therefore, the digital signals must have clear edges in order not to produce disturbed readings. Spurious impulses or contact bouncing can lead to artifacts such as enormous peaks in RPM-signals etc..

Simple sensors working on the principles of induction or photoelectric relays often emit unconditioned analog signals which must be evaluated according to a threshold condition. Aside from that, problems can occur even with conditioned encoder signals (e.g. TTL-levels) due to long cables, bad reference voltages, ground loops or interference. imc incremental counter channels are able to counteract these problems thanks to a special 3-stage conditioning unit.

First comes a high-impedance **differential amplifier** (± 10 V range, 100 k Ω) which enables reliable acquisition from a sensor even over a long cable as well as effective suppression of common mode interference and ground loops. Next, a (configurable) **smoothing filter** offers additional interference suppression adapted to the measurement situation. Lastly, a **comparator** with adjustable threshold and hysteresis serves as a digital detector. The (adjustable) **hysteresis** also serves to suppress interference.



The *digital signal* changes from **0 to 1** when the *analog signal* exceeds the **$VREF + VHYST/2$** threshold.

The *digital signal* changes from **1 to 0** when the *analog signal* falls below the **$VREF - VHYST/2$** threshold.

The size of the hysteresis represents the width of a range-band inside of which the signal can fluctuate (due to signal noise and interference) without an impulse being recorded.

Ranges:

- VREF (Threshold) = -10 V .. +10V
- VHYS (Hysteresis) = +100 mV...+4V
- Low pass filter: None, 20 kHz, 2 kHz, 200 Hz

9.8.5.1.5 Single-track / Dual-track encoder

The **single-track encoder** returns a simple pulse sequence. This means that the pulse count and the time between pulses can be determined, but not the rotation direction of the incremental counter.

A **dual-track encoder** returns two pulse sequences with a 90° offset. Along with the pulse frequency, the rotation direction can also be indicated as positive or negative. To configure a measurement with a dual-track encoder, set the parameter "**Counter signal**"²⁵⁴ which is on the Setup page "Digital channels" under the tab "Encoder", along with the desired "Mode".

Note**Problems with two-point scaling of analog inputs**

Affects both the devices belonging to the imc C-SERIES, and also any devices belonging to the imc SPARTAN and imc CRONOS families which are equipped with the digital multiboard: DI16-DO8-ENC4 or the DI8-DO8-ENC4-DAC4.

When an input is set to **dual-track encoder**, it is not possible to **calibrate** the scale with **two-point scaling** for any **analog inputs**. When you click "Record" to take a measurement, the following message appears: "*The device is not prepared to allow necessary initialization! Please execute menu action "Prepare" (device control)! imcDevices V2.x Adapter*"

However, the "Prepare" procedure does not solve the problem. Instead, temporarily set the incremental counter inputs of the modules affected to "Single-track encoder" in order to be able to measure the two data points used for two-point scaling.

9.8.5.1.6 Zero pulse (index)

The **zero pulse** starts the incremental counter channels' counter mechanism. This means the measured values are only recorded, if an event occurs at the **index-channel**. If measurement without a zero pulse is selected, the measurement starts directly upon starting the measurement.

The **index signal** is differential and set by the comparator settings of the **first** incremental counter channel of the module, even for modules that have several index tracks. The bandwidth is limited to 20kHz.

Note

- By default, the option "Encoder w/o zero impulse" is activated in imc WAVE. If this option is deactivated and the zero pulse fails to appear, the encoder module does not start the measurement at all! In that case, the channels only return zero.

9.8.5.2 Mode (events-counting)

Mode - Events	Description
Events	<p>The event counter counts the sensor pulses which occur during a single time interval (differential event counting). The interval corresponds to the sampling time set by the user. The maximum event frequency is about 500 kHz.</p> <p>An event is a positive edge in the measurement signal which exceeds the user-set threshold value.</p> <p>The derivative quantities displacement and angle measurement have the following settings:</p> <ul style="list-style-type: none"> • Choice of single-track and dual-track encoder ³⁶⁴ • Start of measurement with or without "Zero impulse" ³⁶⁴ • Number of pulses (per unit)
Mode - Distance	Description
Distance (differential)	Path traveled within one sampling interval. For this purpose, the number of pulses per meter must be entered.
Distance (absolute)	Absolute distance. The differential distance measurement is converted to the absolute distance. By taking the zero impulse (the counter with no zero impulse should not be selected) into account, the absolute distance position is determined and indicated. Otherwise, the distance value is assumed to be 0° when the measurement begins.
Mode - Angle	Description
Angle (differential)	Angle traveled within one sampling interval. For this purpose, the number of pulses per revolution must be entered. The absolute angle can be calculated in imc Online FAMOS or determined by the mode Angle(abs).
Angle (absolute)	The differential angle measurement is converted to the absolute angle. By taking the zero impulse (the counter with no zero impulse should not be selected) into account, the absolute angle position is determined and indicated. Otherwise, the angle value is assumed to be 0° when the measurement begins.
Angle (sum)	The differential angle measurement is converted to the cumulative angle. In the process, any zero pulse is evaluated only one time. For this reason, angles which are > 360° are possible.

Note

When using incremental counter modules that work internally with a 16-bit counter, encoders with high pulse rates can lead to overflows. The count is always carried out with sign: $2^{16} = 65536$, i. e. ± 32767 . With dual-track encoders the pulse number is quadrupled internally and leads to a maximum number of pulses per revolution of 8192. For encoders with more pulses per revolution, the hardware must have a 32 bit counter, e. g. imc CANSASfit-ENC6, otherwise an event count must be carried out instead and converted with imc Online FAMOS.

9.8.5.3 Mode (time measurement)

Time measurement

The time measurement mode allows the definition of **edge conditions** between which the time interval is to be measured.

The following combinations are possible:

positive edge	>	negative edge:	(↑ > ↓)
negative edge	>	positive edge:	(↓ > ↑)
positive edge	>	positive edge:	(↑ > ↑)
The combination negative edge	>	negative edge:	(↓ > ↓) is not allowed

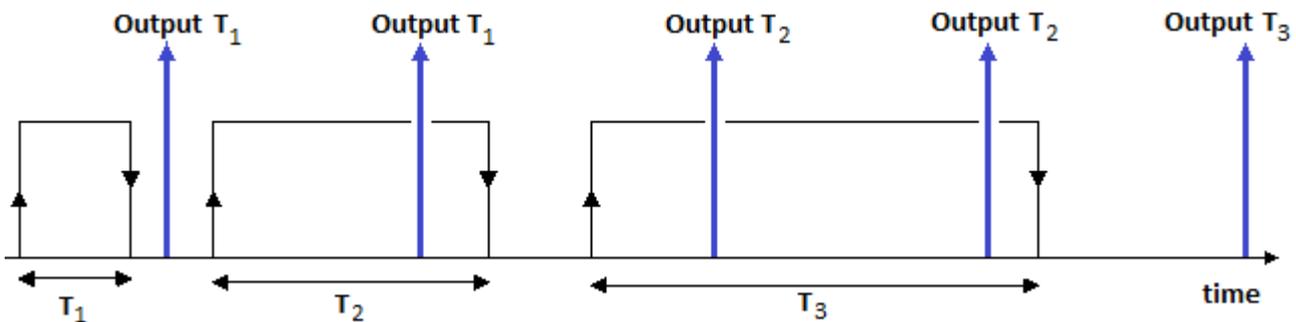
To ensure a high time resolution for the measurement results, suitable scaling must be set for the measurement. An **input range (INC4)** or **Max. time (s) (ENC-6)** specifies the maximum time interval which can be measured between the selected starting and stopping edge. **The time between the signal edges may not be greater than the selected input range.** If the maximum time interval is exceeded during measurement, the system returns the input value range end instead of the true measured value.

Input range	Time resolution	Input range	Time resolution
1 ms	31,25 ns	250 ms	8 μs
2 ms	62,50 ns	500 ms	16 μs
4 ms	125 ns	1 s	32 μs
8 ms	250 ns	2 s	64 μs
16 ms	500 ns	4 s	128 μs
30 ms	1 μs	8 s	256 μs
60 ms	2 μs	16 s	512 μs
120 ms	4 μs	30 s	1024 μs

Time resolution of INC4

The time resolution corresponds to the value of an LSB (Least Significant Bit).

During sampling intervals when no time measurement was possible (because either a starting or stopping edge was missing), the last valid return value continues to be returned until a time measurement is completed. If there is no valid return value, zero is returned. If more than one time measurement is completed during a single sampling interval (due to multiple starting and stopping edges), the last time measured is returned.



Above is illustrated a measured signal from which time readings are taken. Each reading starts at a positive edge in the signal and is stopped at a negative edge. The "up" arrows indicate the times at which the system returns a result. The returned values in this case are T1 –twice; T2 –twice; and T3.

Pulse Time

The point in time at which the edge is located within the sampling interval is determined. This information is needed by some functions in imc Online FAMOS, e.g. for determining the course of the RPMs from a pulse signal: OtrEncoderPulsesToRpm.

The measurement variable **Pulse Time** refers to phase-based data which is only relevant to special applications (particularly order-tracking analysis). It is required for subsequent online calculations. It represents the time between the last detected (asynchronous) pulse and the (synchronous) sampling time at which the counter readings were sampled and evaluated. The unit associated with this variable is called *Code*.



Note

The mode *Pulse Time* depends on the sampling rate. For all ENC-4 types, the entry is visible only if the sampling rate is equal or smaller 1ms. For HRENC-4 the sampling rate must be equal or less 100µs.

PWM

Pulse width modulation (PWM) is a type of modulation in which a technical variable (e.g. electrical current) switches between two values. In the process, the **duty cycle ratio is modulated at constant frequency**. PWM is also known as pulse duration modulation (PDM).

A good illustration of this modulation type would be a switch used to continually switch a heater on and off. The higher the ratio of the on-time to the off-time, the higher the average heating power is.

Measurement of PWM can not be performed directly with the device software. However, if the frequency is known, it is possible to perform it indirectly by time measurement with the following settings:

The **ratio** is the *Duration of HIGH (signal) level* over the *Period duration*.

The *Duration of HIGH (signal) level* is obtained by means of a **time measurement** from *positive to negative (signal) edge*.

The *Period duration* is the **inverse of the frequency**, which must be known.

$$\text{PWM} = t_{\text{pulse}} / t_{\text{Period duration}} * 100\% \quad \text{or} \quad t_{\text{pulse}} * f * 100\%$$

Example:

f= 50Hz, Pulse duration = 10ms

Scaling: $t_{\text{pulse}} * f * 100\% / s = 5000\%/s$

at 10ms: $0.01s * 5000\%/s = 50\%$

This can be entered directly via the scaling:

Channel definition		Encoder		Filtering		Sampling & Preprocessing		Data transfer	
Channel name	PWM								
Measurement mode	Time		Signal	One signal					
<input checked="" type="checkbox"/> Encoder w/o zero impulse	Scaling factor	5000 %/s		Start edge	Positive slope				
	Maximum	0.02 s		Stop edge	Negative slope				
Input range	±10 V		Switching level	1.5 V		Unit	%		
Signal shape			Hysteresis	0.5 V		Scaling offset	0 %		

Settings for PWM measurement in time mode

9.8.5.4 Mode (combined measurement)

Mode	Beschreibung
Frequency	Frequency is determined by means of a combination measurement ³⁶¹ . If the frequency was previously multiplied or divided, this can be reflected in the scaling value. The frequency is always unsigned, for which reason there is no dual-track encoder for it.
Speed	The sequence of pulses is converted to m/s by means of a combination measurement ³⁶¹ . Toward this end, the number of pulses per meter must be entered.
RPM	The sequence of pulses is converted to revolutions per minute by means of a combination measurement ³⁶¹ . Toward this end, the number of pulses per revolution must be entered.

9.8.5.5 Information and tips

Problem: Measured value does not settle at zero

Answer: This problem is seen where RPM, velocity or frequency signals trail off to zero.

Whenever individual pulses are missing, the system measures the elapsed time and interpolates an RPM- (or velocity, etc.)-value on this basis. Physically speaking, this is the most realistic assumption to make about a rotating machine, but using this technique means that some results are only estimates.

If no pulses arrive for a long time, there is no definitive information on which to base calculations. In this case, the last value calculated may remain >0.

9.9 Fieldbusses

This chapter describes the different fieldbusses and their operation with imc WAVE.

9.9.1 General notes on fieldbusses

Fieldbus channels: Operating methodology

Analog channels are configured using the Setup page.

There is a dialog corresponding to each fieldbus, the fieldbus assistant. In a sense, these dialogs allows you to configure the "digital signal conditioning" for the fieldbus channels. Just like with the amplifier channels, the channels defined in the fieldbus assistant are also available on the Setup page for the digital channels. For the user, these appear as additional analog inputs.

Therefore, you first need to define how to extract the channels from the messages using the fieldbus assistant. Once the fieldbus channels have been created, the storage and recording settings can be specified on the Setup page for the digital channels. Fieldbus channels can also be used in imc Online FAMOS.

Fieldbus channels: Synchronization with analog channels

Parallel recording of analog data and fieldbus channels is possible.

With fieldbusses, it is generally not possible to determine exactly when the AD-conversion at the sending subscriber has taken place. The resulting time offset is not taken into account.

The time of a message's reception is registered in the device with a minimum uncertainty of 100 μ s. This is the time at which the message reception is complete. As well, due to the various pipelines (AD-converter, time-offset correction, trigger-calculation), there is generally a time uncertainty of one sampling period in determining times. In most fieldbusses, it is even two sampling periods.

If the clocks of an analog data recording channel and a fieldbus channel are to be compared, then the possible maximum time offset Δt_{\max} is

$$\Delta t_{\max} = \max(100 \mu\text{s}, 2 \cdot \max(t_{\text{samp-analog}}, t_{\text{samp-FB}})) + t_{\text{FB-delay}}$$

$\max(a, b)$	function that returns the larger of two values
$t_{\text{samp-analog}}$	the sampling period used on the analog channels
$t_{\text{samp-FB}}$	the sampling period used on the fieldbus channels
$t_{\text{FB-delay}}$	the maximum delay between A/D conversion in fieldbus sensor and actual transmission on the fieldbus



Example

A fieldbus channel is sampled every 2 ms and an analog channel every 1 ms. The maximum delay on the fieldbus is 3 ms.

$$\Delta t_{\max} = 2 \cdot 2 \text{ ms} + 3 \text{ ms} = 7 \text{ ms}$$

Treatment of channel names

When importing channel names, it is possible that the naming rules valid for imc devices are not observed. In general, the names are adopted and when used in imc Online FAMOS or imc FAMOS etc. are enclosed with {...}. Only exception: In order to avoid mistaking channels' names with those of process vector variables, any channel names beginning with "pv." are renamed to "pv_".

General notes on sampling and time stamping

Each channel which is recorded with the fieldbus, can be stored in two different ways.

- Sampled with a fixed rate
- Time stamp for each sample



Reference

See also

- [Data recording with sampling or with time stamp](#) ^[472]
- [Capturing fieldbus input data](#) ^[472]
- [Time-stamped channels - Setting the RAM-buffer duration, circular buffer duration and pretrigger duration](#) ^[732]

9.9.2 CAN-bus interface

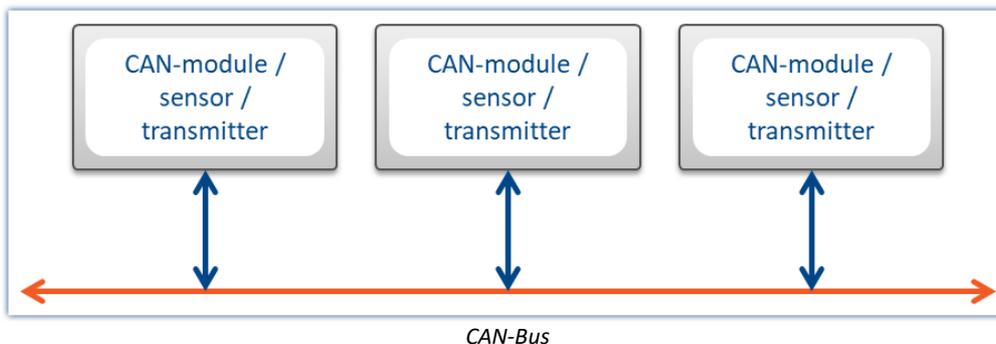
What is the CAN-Bus?

A CAN-Bus interface is optionally available for imc measurement devices, offering full support of CAN-Bus sensors using basic data formats. It is thus possible to take measurements via the CAN-Bus and via the analog and digital inputs practically synchronously.

The CAN-Bus (Controller Area Network) is a serial transmission line connecting all modules in a two-wire circuit. To prevent signal reflection, terminators are connected at the cable ends. The module holder is designed to conform to CAN-Bus operation per CiA standards (CiA Draft Standard 102 Version 2.0, CAN Physical Layer for Industrial Applications).

Multiple sensors and devices can be connected to the CAN-Bus, and can output their **measured** values at regular intervals onto the bus, for instance. Every device and every sensor sends its data as a **message** on the bus. Up to 8 bytes of data can be sent with one message. Every message possesses a unique **identifier**. A signal's identifying information such as the channel name and units is not sent with the message. Instead it is usually recorded in a database which provides a key to how the message's data is encoded, based on its identifier.

Every CAN-module is connected to the CAN-Bus at a node.



9.9.2.1 Informations

Further references

- CiA® Draft Standard 102 Version 2.0: CAN Physical Layer
- CAN Controller Area Network, Wolfhard Lawrenz, Hüthig Verlage, 1994 Heidelberg
- ISO / DIS 11898 (ISO 11519-2) for bus driver
- ISO / OSI reference model
- CAN FD (ISO 11898-1:2015)

Exception: no support of Remote Frames (RTR)

Bus features

The CAN-Bus interface includes the following features:

- CAN Transceiver in accordance with ISO / DIS 11898, isolated
- Baud rate can be adjusted per software
- Standard identifier on CAN-Bus: 11-bit identifier (0..2047)
- Extended identifier on CAN-Bus: 29-bit identifier (0 .. 536870911)
- CAN FD (ISO 11898-1:2015)

Note

Also observe the general notes on file buses in regard to [control philosophy](#)^[368] and the sampling rate either as [fixed sampling rate](#)^[472] or [time stamp](#)^[472].

9.9.2.2 Connection to the device

9.9.2.2.1 CAN port pin configuration

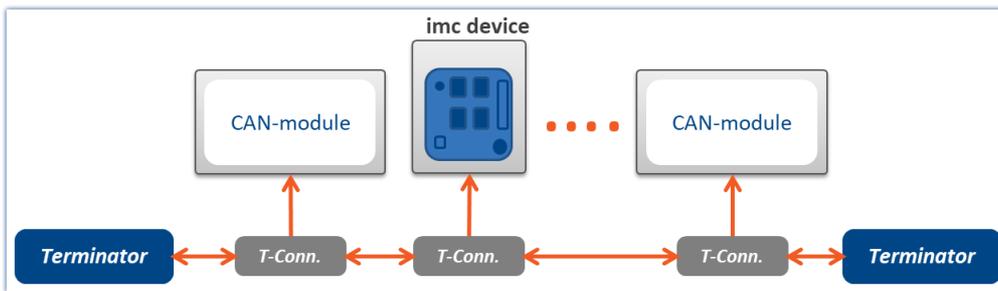
The device comes with a pair of terminals for each CAN-node, configured as follows:

DSUB-PIN	Signal	Description	Usage
1	+CAN_SUPPLY	depending on hardware	supply I < 1 A
2	CAN_L	Dominant low bus line	Connected
3	CAN_GND	CAN Ground	Connected
4	nc	Reserved	Not used
5	-CAN_SUPPLY	depending on hardware	
6	CAN_GND	Optional CAN Ground	Connected
7	CAN_H	Dominant high bus line	Connected
8	nc	Reserved (error line)	Not used
9	nc		Not used

9.9.2.2.2 CAN cabling (Y-adaptor)

Device connection with a Y-adaptor

The CAN interfaces from imc provide exactly one DSUB9 connection for each node. Connection to a CAN bus therefore requires a Y-adaptor.



Device connected with Y-adaptor

To guarantee a CAN data transfer rate of 1 Mbit/s, the **maximum length** for the pathway through a Y-adaptor is **30 cm**. Usually the wiring within the device is already 30 cm. If using an external Y-adaptor it has to be attached directly at the connector.

In this case it doesn't matter whether the other CAN participants are connected with or without a Y-adaptor. The diagram shows just one of several connection possibilities.

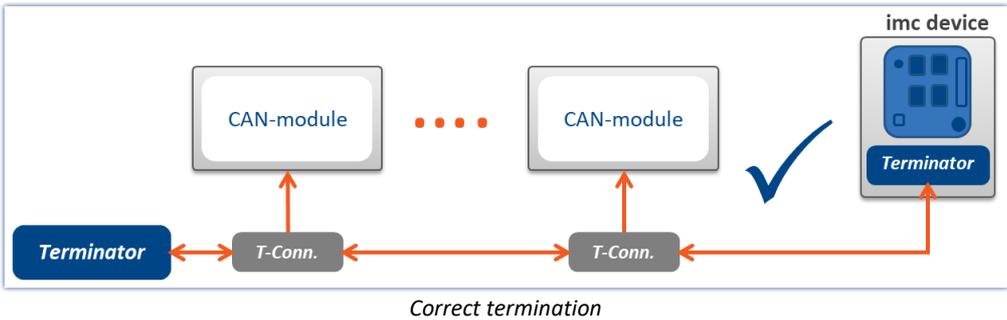
9.9.2.2.3 Connecting terminators

- Terminators with 124 Ω in accordance with CiA.
- Terminators are connected between pin 2 and 7.
- Terminators must be used at both ends of the bus. No further terminators are allowed.

Terminator in device

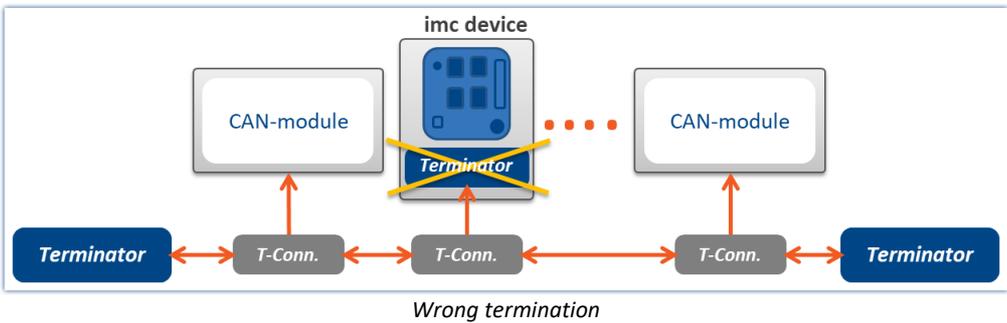
Devices equipped with CAN-bus interface, can switch a terminator resistor internal via software. If the device is connected at one end of the CAN line, an external Y-cable with terminator is not needed.

- See: [Setting for a device of firmware group A](#) ³⁸⁴
- See: [Setting for a device of firmware group B](#) ⁴²⁰



Warning

If the CAN-bus is already terminated correctly, that option cannot be used!

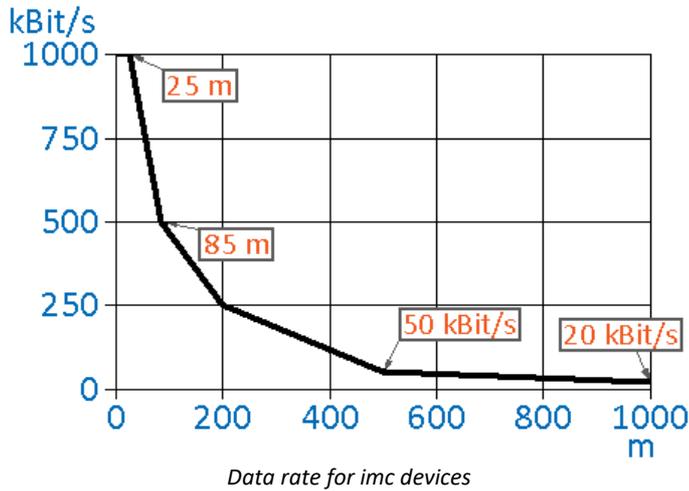


9.9.2.3 Technical specification

9.9.2.3.1 CAN transfer rate

With the transfer rate, it is necessary to distinguish between standard CAN and CAN FD (CAN with flexible data rate). The following description pertains primarily to the classic, standard CAN-Bus.

As signal lead lines become longer, the maximum reliable data transfer rate decreases.



Line length [m]	Data rate [kbit/s]	
3	8000	CAN FD
5	5000	CAN FD
12,5	2000	CAN FD
25	1000	
85	500	
200	250	
500	50	
1000	20	

Net data transmission rate

Conversely, the net data rate depends on the package size and on whether the Standard Frame or the Extended Frame is used.

A high net data rate is achieved by using the maximum number of data bytes for each message. If each message in the **CAN standard** is optimally filled with **8 bytes** and the standard frame is used, the result is **576.6 kbit/s**.

The table shows the net data rates for **CAN standard** 1 Mbit/s.

Data length	Net data rate with CAN-standard 1Mbit/s	
	Standard Frame	Extended Frame
0	-	-
1	72,1 kbit/s	61,1 kbit/s
2	144,1 kbit/s	122,1 kbit/s
3	216,2 kbit/s	183,2 kbit/s
4	288,3 kbit/s	244,3 kbit/s
5	360,4 kbit/s	305,3 kbit/s
6	432,4 kbit/s	366,4 kbit/s
7	504,5 kbit/s	427,5 kbit/s
8	576,6 kbit/s	488,5 kbit/s

With **CAN FD**, even **without** the faster data transmission in the data phase, the larger data volume alone provides considerable advantages. At a nominal bit rate of **1 Mbit/s**, the net data transmission rate for a CAN FD frame with a data length of **64 bytes** is **903 kbit/s**.

9.9.2.3.2 CAN FD

CAN FD (CAN with flexible data rate) offers a **8 higher data rate of up to 8 Mbit/s** and up to **64 Bytes of payload data per message**.

Fundamentally, the same network topology can be used as with the classic CAN-protocol. However, the previous CAN-controllers are not able to send or receive CAN FD messages > 8 Bytes. For this reason, CAN and CAN FD can work together in one bus-system if:

- no Remote-Frame is sent (something which imc software does not generate),
- no CAN-messages > 8 Bytes are used
- no [Switch Bit Rate](#)³⁹⁸ is used.

Remarks on CAN FD

- It is possible to send and receive larger messages with [imc Online FAMOS](#)³⁹⁹.
- Fundamentally, all CAN-experiments can be transferred to devices with CAN FD.
- Messages can be sent at a higher baud rate using the option "[Switch Bit Rate](#)³⁹⁸".
- CAN FD configurations can be imported to CAN-devices. The configuration is then converted to CAN and resulting errors must subsequently be eliminated by the user.

9.9.2.3.3 Number of CAN nodes

Correlation between line length, number of nodes and cable cross section. The cross section of the line should increase in size as more CAN-nodes are served by the system.

Line length	Number of nodes/ Cable cross section		
	32	64	100
100 m	0.25 mm ²	0.25 mm ²	0.25 mm ²
250 m	0.34 mm ²	0.5 mm ²	0.5 mm ²
500 m	0.75 mm ²	0.75 mm ²	1 mm ²

9.9.2.3.4 Specifications from CAN bus in imc WAVE

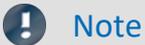
- Baud rate up to 1 Mbit/s; 8 Mbit/s with CAN FD
- Multiple independent CAN nodes (ports)
- Aggregate sampling rate on CAN channels: typically up to 15000 samples/s, depends heavily on data types, no. of channels and processing. Analog aggregate sampling rate is not affected.

Devices of the [firmware group A](#) ¹⁴⁶ (e.g. CRONOS family, C-SERIES, SPARTAN, BUSDAQ)

- The devices can manage up to 512 channels (analog, virtual and CAN channels).
- CAN channel triggers are computed together with analog channel triggers on the unit's motherboard. This can affect the max. aggregate sampling rate on CAN channels!

Devices of the [firmware group B](#) ¹⁴⁶ (e.g. ARGUSfit)

- The devices can manage up to 1000 channels (analog, virtual and CAN channels).
- The CAN bus channels are triggered on the CAN module. The maximum aggregate sampling rate is not influenced by this.



Note

Bus Decoder

More channels can be acquired using the Bus Decoder.

9.9.2.3.5 What one needs to know to connect a CAN-sensor

All CAN-Bus sensors transmit messages on the CAN-Bus using a specific data structure. In order for the messages to be understood, a consistent structure must be used. The following parameters (values serve as an example) must be contained in a parameter:

What is supported?

Baud rate:	500 kbit/s
ID:	455
Length:	5 bytes
Repeat rate:	every 10 ms

The contents of the various messages are usually listed in a table. Consider the following typical CAN application:

Signal Name	Byte	Bit	Value Range	Conversion
Speed	1	0..7	0...255	50*value
Status	2	4	0-1	---
Compressor ON	2	5	0-1	---
Clutch moment specification	3	0..7	-128..127	*20, in N×m
Braking moment	4	0..7	0..255	*0.39
ABS Indicator	5	0	0..1	---

 **Note**

- Bytes are numbered from 0 to 7 in the imc WAVE software. Byte 0 is the first byte and Byte 7 is the last byte in an 8-byte message. Bit 0 of each byte is the LSB and Bit 7 is the MSB. Some tables (such as the one above) may differ slightly, in that the bits and bytes are indexed from 1 to 8. In these cases, you need to subtract 1 from these values.
- Also observe the general notes on file buses in regard to [control philosophy](#)^[368] and the sampling rate either as [fixed sampling rate](#)^[472] or [time stamp](#)^[472].

9.9.2.3.6 Supported CAN protocols

What is supported?

- Only layers 1 and 2 of the layer model
- channels which can be identified uniquely from a single identifier and contain exactly one sample per identifier
- sensors which must be commanded via the CAN-Bus to transmit or measure
- device receives data
- device sends data
- data structure as follows:
 Identifier indicates which byte contains channel data and the format of the data. Analog sensor data and binary signals (only 0 or 1) can be read.

 **Example**

The following table illustrates a type of application which is supported:

Signal Name	Byte	Bit	Value Range	Conversion
Speed	1	0..7 unsigned	0...255	50*value
Status	2	4	0-1	---
Compressor ON	2	5	0-1	--

What is not supported?

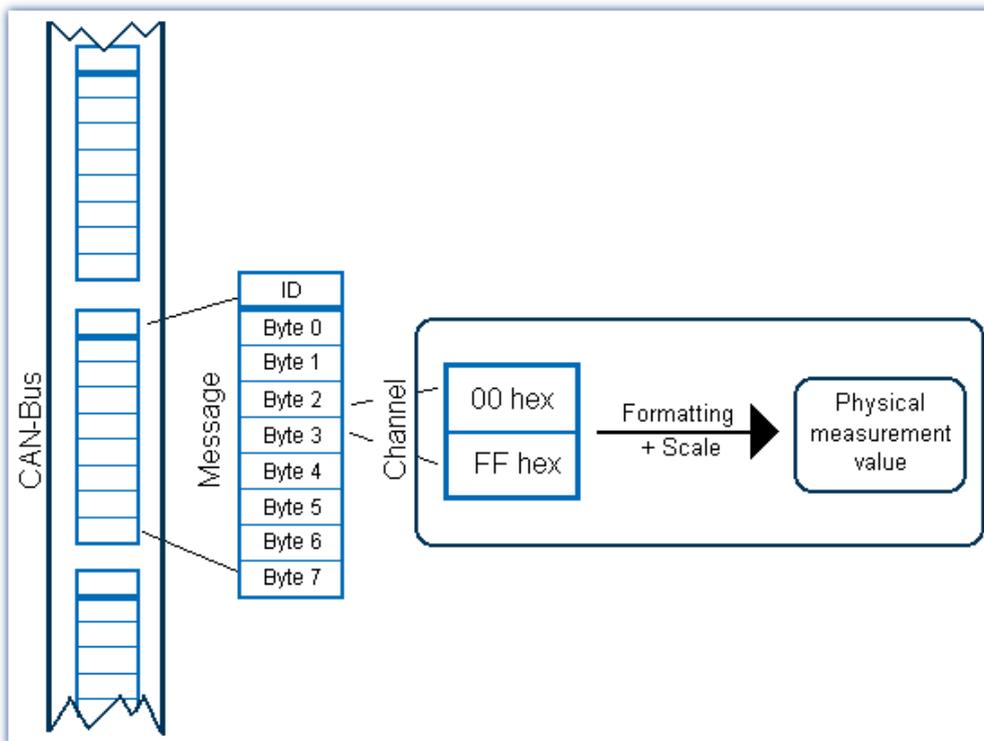
- terminal channels
- recording and analysis of 100-byte error protocols
- control of other devices on the CAN-Bus by the measurement device
- consideration of complex conditions
- compensation for CAN-Bus delay
- multiple protocols
- higher protocols

All major imc WAVE features available for CAN channels

- Online processing using imc Online FAMOS
- Analog data can be mixed with CAN data for processing
- Data storage on the device internal hard drive and/ or PC hard drive.
- Complete access to the units trigger machine
- The device can be operated as a stand-alone unit, i.e. without a connected PC.
- Binary signals from CAN-Bus possible

How are CAN-Bus channels created in devices?

Several sensors and devices can be connected to a CAN-Bus. Each of these transmits data with an identifier at a constant clock rate to the CAN-Bus. The identifier indicates the origin and meaning of the data. An 8-byte data packet is attached to each of these (64 Byte with CAN FD).



Format

Individual bits or bytes can be read from a message

- These are usually specific measurement values or conditions
- Supported number formats include: 1 bit, 1-8 Byte, signed/ unsigned integers, real numbers, Intel or Motorola byte orders
- Integer values can be converted to physical quantities

Time stamp alternative

Time stamping enables you to protocol the exact transfer time of a CAN-Bus message

- This data can be saved and visualized
- Data can also be equidistantly resampled for digital filtering, computation of FFT's etc.

CAN error handling

Bus clients, individual client sensors or the bus line itself may be defective

- Complete configuration of error handling

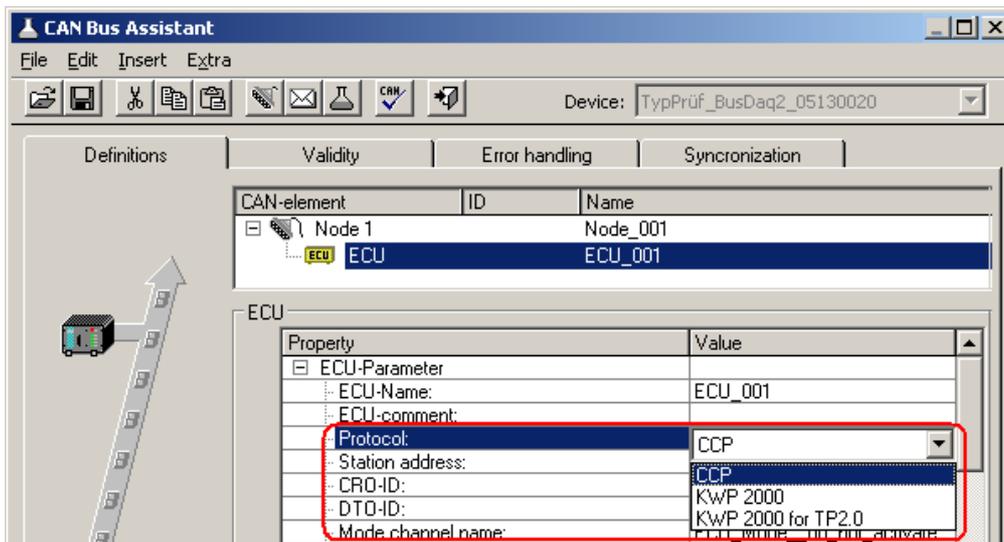
9.9.2.3.7 ECU control unit protocols

The device software comes with functions for (motor-) control units (ECUs) and special control unit protocols. Using these, it is possible to query and record internal control unit values, and to start special control unit values.

Motor control units are connected via the CAN-Bus. The ability to query internal control unit values requires compliance with special protocols.

System requirements	
Hardware	<ul style="list-style-type: none"> • A device with CAN-module is necessary with the common device control functions and all desired protocols being enabled or • Devices of firmware group A ¹⁴⁶
Software	If the special imc Online FAMOS functions are used, then the device must be enabled for imc Online FAMOS.

The support ECU protocols can be found [here](#) ⁴⁴⁰.



Properties of ECU messages

To capture internal control unit values, it is necessary that the control units and the values (channels) to acquire be configured with the help of the CAN-Bus Assistant.

If the return values of the control unit commands are to be analyzed, or if special control unit functions are to be run, then it is necessary to use special functions in an imc Online FAMOS program with control commands.

For the description of the configuration see [Control units in the CAN-Bus Assistant](#)⁴⁴⁰.

Basic functions

Keyword Protocol 2000 with ISO Transport Protocol (ISO 14230-3)		Keyword Protocol 2000 with VW TP 2.0 (ISO 14230-3)	
A subset of keyword protocol 2000 is used with addressing modes Normal and NormalFixed.			
10h	StartDiagnosticSession	10h	StartDiagnosticSession
13h	ReadDiagnosticTroubleCodes	13h	ReadDiagnosticTroubleCodes
14h	ClearDiagnosticInformation	14h	ClearDiagnosticInformation
17h	ReadStatusOfDiagnosticTroubleCodes	17h	ReadStatusOfDiagnosticTroubleCodes
18h	ReadDTCsByStatus	18h	ReadDTCsByStatus
20h	StopDiagnosticSession	20h	StopDiagnosticSession
21h	ReadDataByLocalIdentifier	21h	ReadDataByLocalIdentifier
22h	ReadDataByCommonIdentifier	22h	ReadDataByCommonIdentifier
23h	ReadDataByAddress	23h	ReadDataByAddress
27h	SecurityAccess	27h	SecurityAccess
2Eh	WriteDataByCommonIdentifier	2Eh	WriteDataByCommonIdentifier
31h	StartRoutineByLocalIdentifier	31h	StartRoutineByLocalIdentifier
33h	RequestRoutineResultsByLocalIdentifier	33h	RequestRoutineResultsByLocalIdentifier
38h	StartRoutineByAddress	38h	StartRoutineByAddress
3Ah	RequestRoutineResultsByAddress	3Ah	RequestRoutineResultsByAddress
3Bh	WriteDataByLocalIdentifier	3Bh	WriteDataByLocalIdentifier
3Dh	WriteDataByAddress (Maximal zwei Byte Daten)	3Dh	WriteDataByAddress (Maximal zwei Byte Daten)
3Eh	TesterPresent	3Eh	TesterPresent

CCP		XCP		Diagnostics on CAN (ISO 15765)	
01h	CONNECT	D3h	ALLOC_ODT_ENTRY	10h	StartDiagnosticSession
03h	DNLOAD	D4h	ALLOC_ODT	12h	ReadFreezeFrameData
02h	SET_MTA	D5h	ALLOC_DAQ	14h	ClearDiagnosticInformation
04h	UPLOAD	D6h	FREE_DAQ	17h	ReadStatusOfDiagnosticTroubleCodes
07h	DISCONNECT	E0h	SET_DAQ_LIST_MODE	21h	ReadDataByLocalIdentifier
0Fh	SHORT_UP	E1h	WRITE_DAQ	22h	ReadDataByIdentifier
12h	GET_SEED	E2h	SET_DAQ_PTR	23h	ReadMemoryByAddress
13h	UNLOCK	E3h	CLEAR_DAQ_LIST	27h	SecurityAccess
14h	GET_DAQ_SIZE	F0h	DNLOAD	2Ch	DynamicallyDefineLocalIdentifier
15h	SET_DAQ_PTR	F4h	SHORT_UPLOAD	31h	StartRoutineByLocalIdentifier
16h	WRITE_DAQ	F5h	UPLOAD	3Bh	WriteDataByLocalIdentifier
		F6h	SET_MTA	3Eh	TesterPresent
		F7h	UNLOCK		
		F8h	GET_SEED		
		FEh	DISCONNECT		
		FFh	CONNECT		

Standards

Keyword Protocol 2000 with ISO transport protocol

- Keyword protocol 2000: ISO 14230-3
- ISO transport protocol: ISO 15762-2

Options and add-on's

The different protocols are optional.

9.9.2.4 CAN channel data formats

The following data formats are available for recording CAN channels:

CAN-Bus Format	Recording with sampling	Recording with time stamping
Unsigned/ signed integer: 1 to 16 bit	2 byte integer	2 byte integer + 6 byte time stamp
Unsigned/ signed integer: 17 to 32 bit	4 byte real	4 byte real + 6 byte time stamp
Real 4, 8 byte	4 byte real	4 byte real + 6 byte time stamp
Digital port with 1 to 16 bit tracks	2 byte word for compact 16 bits	2 byte word for compact 16 bit + 6 byte time stamp

As you can see, each format has different memory requirements. This is important to know when you are estimating how much memory you will need and how precise the measurement values will be.

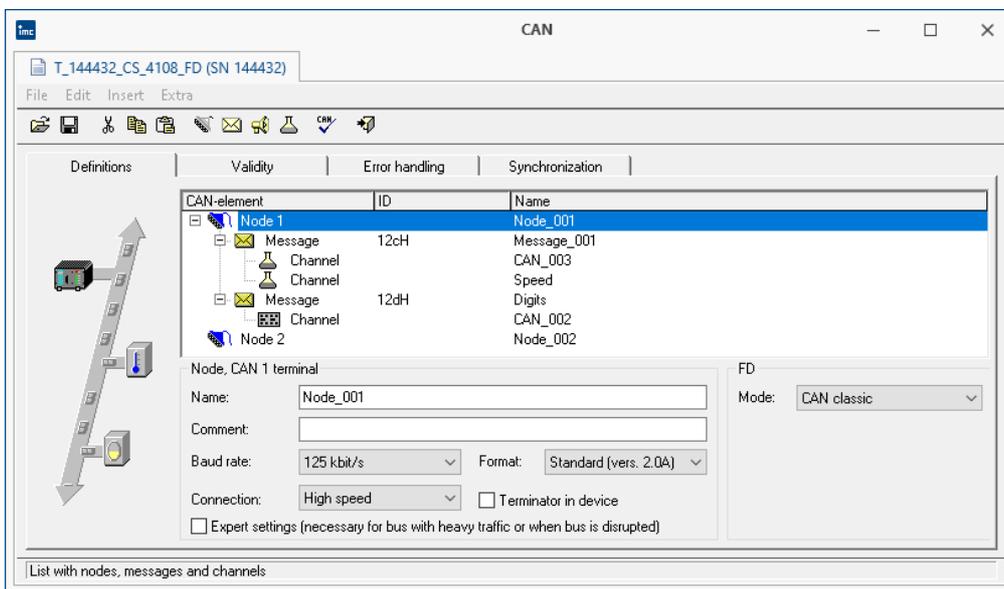
 **Example**

A CAN sensor is sending messages consisting of 8-bit numbers. These are to be directly interpreted as the measurement values and equidistantly sampled (i.e. without time stamping). Therefore, a 2 byte integer is stored for each value.

9.9.2.5 CAN-Bus Assistant (firmware group A)

For the purpose of configuring nodes, messages and channels belonging to the device of the [firmware group A](#) (146) (CRONOS, BUSDAQ, SPARTAN, etc.), the *CAN-Bus-Assistant* is used. The assistant for [firmware group B](#) (146) (ARGUS) is located [here](#) (418).

Action	Page
CAN (CAN)	Assistants



CAN-Bus Assistant

Together, the CAN nodes, messages and channels form a CAN configuration. These can be saved for re-use as ASCII configuration files (*.CBA). We recommend creating a separate folder for these files. Each line in a CBA file consists of a code word and a corresponding value. If necessary, some the values can be edited directly using a text editor like the Windows Notepad. In general, we do not recommend changing information concerning the bitmaps - these properties should only be edited using the CAN-Bus Assistant.

The CAN-Bus Assistant does not contain an *OK* button for confirming changes. Instead, all changes are applied directly. Invalid values are shown in red in the status bar. You can also perform a [Configuration Check](#) (412).

 **Note**

It is possible, however, to exit the Assistant without correcting the changes - however, no CAN channels will be created! Therefore, if you wish to create CAN channels, the configuration must be correct before you exit the Assistant.

9.9.2.5.1 Quick tutorial

The Quick Tutorial shows you how to create a simple CAN-Bus configuration for a single CAN channel. After completing the tutorial, you should be able to create more complex configurations as well.

Step 1: Start the CAN Assistant

Start the CAN-dialog on the "Assistants" page. to start the Assistant. A single node is visible the first time you start.

Step 2: Define a node

Change the node name to a more suitable name for your project. Specify the baud rate from the drop-down list. Select a format for the node (Standard or Extended format).

Step 3: Create a message

Next, you need to define a message. Select the node and click on the message icon. A message object is added below the node.

Step 4: Configure the message

Change the message name to suit your project. Assign an ID to the message (between 0 and 2031 for Standard format, between 0 and 536870911 for Extended format).

Step 5: Create a channel

Next, you need to define a channel. Select the message and click on the channel icon. A channel object is added below the node.

Step 6: Configure the channel

Change the channel name to suit your project and specify the numerical format of the channel from the drop-down list.

Step 7: Check configuration

Perform a configuration check by clicking on the checkmark button. The results of the check are displayed in the status bar.

Step 8: Quit the Assistant

Once you have exited the CAN-Bus Assistant, you can now use the CAN configuration when measuring via the CAN interface. The CAN-channel you defined now appears in the software interface.

9.9.2.5.2 Editing a CAN configuration

The properties of various CAN objects are arranged in drop-down lists and text boxes. By selecting several objects in the list, you can change their properties all at once. This is only possible, of course, when objects of the same type have been selected. Right-clicking in the list opens a context menu where you can select all objects of a certain type (e.g. nodes, messages or channels).

Note

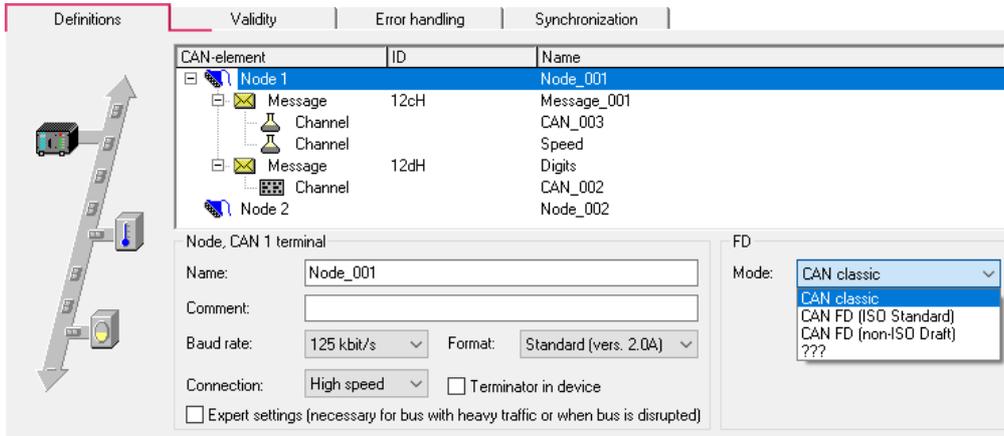
Any changes you make to the properties are applied immediately. Even incorrect settings (e.g. channel names) are applied. Any errors are shown in the status bar at the bottom of the list.

It is not necessary to click on an <OK> button to apply any changes or new settings - however, there is no <Cancel> button to undo any unwanted changes!

In general, we recommend defining the properties for new objects one-by-one. It is also possible to change settings for several like objects simultaneously, but properties which must be unique (e.g. channel name) will be grayed out.

9.9.2.5.3 Nodes

9.9.2.5.3.1 Definition



Definition nodes

FD-Mode:

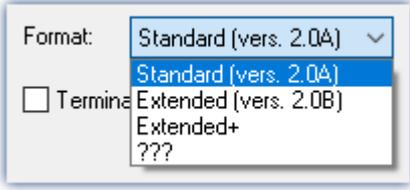
If the device is equipped with a CAN FD interface, the entry for FD appears on the *Definitions* card. Here is where the transfer protocol is defined:

Mode	Description
CAN classic	CAN High Speed in accordance with ISO 11898, CAN Low Speed in accordance with ISO 11519. The symbol for the node is shaded grey-blue.
CAN FD (ISO Standard)	in accordance with ISO 11898-1:2015 The symbol for the node is shaded red-blue.
CAN FD (non-ISO Draft)	according to an earlier draft from BOSCH. The symbol for the node is shaded red-blue.

Note

Mixed operation with both classical CAN-modules and CAN FD are covered in the ISO standard 11898-1:2015, if the CAN-controllers involved support it. Mixed operation with modules which work with "old" controllers, however, is not possible. When receiving CAN FD messages, these modules would generate error frames.

Parameter	Description
Name	A unique name for each node with up to 65 characters. The default name for a new node is "Node_" + index (e.g. 001). If several devices are being used, a _j is appended to the name, where j is the device index number. All names must be comply with imc FAMOS variable syntax, just as all other device channel names.
Comment	This text accompanies the node name and is usually intended to add a more detailed description of the node. The maximum length of this field is 255 characters.
Baud rate	The baud rate specifies how fast the bits are transferred over the serial bus. All modules in a given CAN-Bus must use the same baud rate! Select the baud rate from the drop-down list: 5kbit/s to 10Mbit/s. The default value is 125kbit/s.
Data rate (with CAN FD)	If CAN FD is set for the FD-mode, the entry for the data rate is shown. If the bus exclusively has subscribers whose controllers support CAN FD, it is possible to transfer different settings for the Baud rate and data rate. While transferring at the higher data rate, the subscribers using the slower Baud rate rate are then switched to passive.

Parameter	Description
Format	<div style="display: flex; align-items: center;"> <div style="margin-right: 20px;">  <p>Format: <input type="checkbox"/> Terminator</p> <p style="text-align: center;"><i>Format: Standard or Extended</i></p> </div> <div> <p>"Standard", "Extended" or "Extended+".</p> <p>The difference between Standard and Extended is that the Standard-format ID length is 11 bits (IDs from 0 .. 2031 are permitted) and in Extended-format 29 bits (IDs from 0 .. 536870911 are permitted). Extended-format allows for greater distances between consecutive IDs. This makes it possible to extend the system without major changes to the configuration.</p> <p>The "Extended+" format is able to receive "Standard" and "Extended" identifier. For sending the message then select in the Assistant [396] either "Device sending message" for a 29-bit identifier or "Device sending message in Standard-format" for 11 bits.</p> </div> </div>
Connection	This parameter determines which node is assigned to which port. Node 1 is generally assigned to CAN 1 and Node 2 to CAN 2, no other nodes are assigned to any port and must be deleted from the CAN-configuration in order for it to be consistent.
Terminator	A terminal resistor can be connected to a node via software. If the measurement device is located at the end of the CAN line, this eliminates the need for a Y-cable with an external terminator. See also " Connecting terminators " [372].

Expert settings

CAN-protocol (classic)

This option is only needed for very rare cases. If there are disturbances on the bus, you can make adjustments for each node here.

Parameter	Description
Slope detection (soft synchronization), width of the time window	<p>All subscribers to the bus have the same basic clock rate. However each module comes with its own quartz. Thus, phase differences can arise. The first edge of a message is used to get all subscribers in phase (hardware synchronization). Edges within a message can be used to compensate subsequently arising phase differences within a message. The width of the time window determines by what value a bit is shortened or lengthened in order to re-align the phase.</p> <p>Using this option it's possible to alter the time window in which the edges are expected.</p>
Number of samples per bit	If the bus has achieved a steady state, the level is read by the receiver by means of sampling, and the bit is determined to be either 1 or 0. If there is disturbance on the bus, it's possible that the disturbance will be picked up in a sample. This option helps by increasing the number of samples to three. If the samples show different levels, the majority is taken as the bus level.
Sample location in bit time	Here, the position within the bit time can be altered. By default, it is set to 50%. With long leads, an impulse reaches its level only after a certain delay. With this option, the sampling time can be shifted backward in order to compensate for the delay. The setting is expressed in percent of the total time.

Note

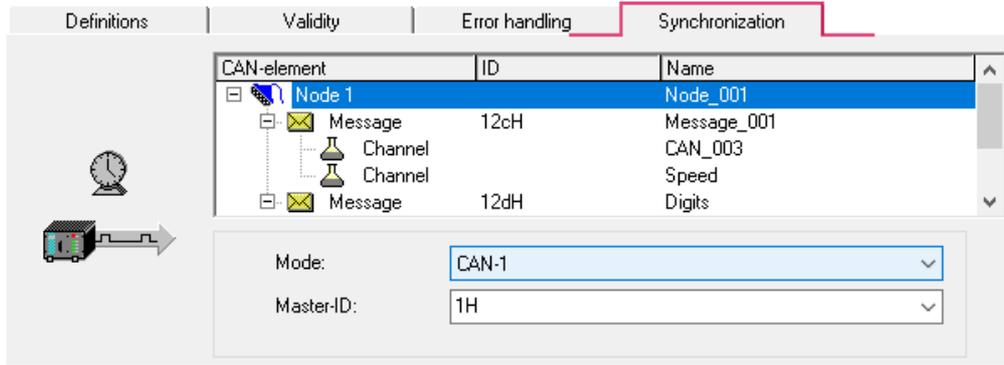
- In most cases, disturbance on the bus can be compensated by shifting the **position of the sampling time** backwards.
- Disturbance is usually the result of long cables. The low-pass effect of such long lines prevents voltage levels from changing rapidly. In extreme cases, this can mean that the correct level at the center of the bit time (50%) has not been reliably reached.

CAN FD

For transfer at the CAN FD data rate, deviating settings can be made. Only the number of samples per bit is omitted here.

9.9.2.5.3.2 Synchronization

For the purpose of synchronizing any CAN-modules connected, such as imc CANSAS, the measurement system is capable of sending a message in CAN-1 protocol. In such a case, your measurement system functions as the Master. If the imc CANSAS module is configured accordingly, it synchronizes itself to this message.



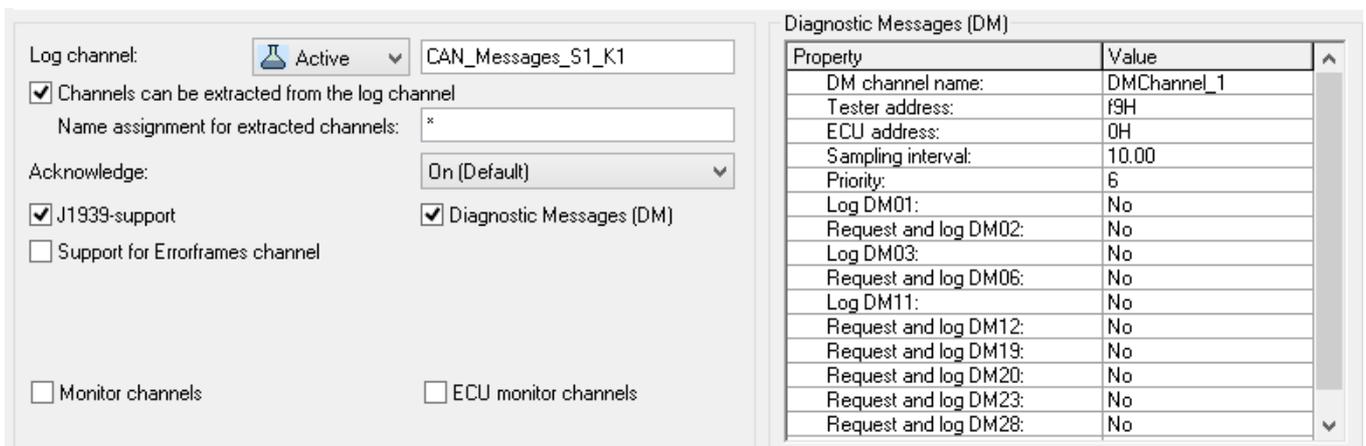
Synchronization of a node

The **CAN-1 Protocol** message is sent at a one-second clock rate by the device working as the Master. imc CANSAS modules working as Slaves receive this CAN message and synchronize themselves to it. The message used takes a format corresponding to the CAN-1 protocol. Other imc CANSAS modules, as well, can work as the CAN-1 Master. It is recommended that the device capturing the measured data works as the CAN-1 Master.

Conditions

- All devices (in other words the Master and all Slaves) are connected to the same CAN-Bus line.
- No gateway or router may be connected between devices. In the CAN-1 protocol, special consideration is given to achieve very high precision.
- For imc BUSDAQ ([group A4](#)¹⁴⁶): The imc measurement device may not be operated in Sleep/Resume mode.

9.9.2.5.3.3 Validity



Validity of a node

Log channel

Parameter	Description
Log channel	<p>With Log channel "Active", all messages belonging to the node are logged in imc WAVE. In the input field you can enter the name of the log channel. By default, a name with slot and node number is generated. There is exactly one log channel per node. Message IDs and data bytes are logged.</p> <p>You can filter a selection of messages by creating receiving messages³⁹⁷ with specific IDs whose "Message log" is set to "Log CAN bus message". In this case, only these messages are logged.</p>
Extract all channels from log channel	<p>The option allows the data stream to be unpacked as individual channels via imc FAMOS. For reasons of data security, this option is deactivated by default. In consequence, sensitive information will not be inadvertently incorporated into the log channel's file of measured data.</p> <p>Beware of the following:</p> <p>Extraction of individual channel signals from the data stream by means of the Bus Decoder or via imc FAMOS is only possible if the box is check marked. In this case, the option must be explicitly activated.</p>

A global channel for logging CAN-messages is also available in ASCII-time stamp data format. By default, imc WAVE creates this global channel with the name CAN_Messages_S(i)_N(j), i: slot index (1..8), j: node index (1,2). If you have set messages of different slots and nodes to be logged, a corresponding number of global channels will be created.

Under "Table" menu item of the curve window context menu (accessed by right mouse-click), it is possible to set the text format to "imc DEVICES CAN-message". At the left of the row in the table, the message ID is stated, to the right of it the message data bytes. The values are displayed in hexadecimal format.

Message log channel can be saved as MDF files to the device disk. See also section "[Storage options](#)"⁷⁵⁴.

Acknowledge

Disables the transmission of Acknowledge messages. The CAN controller usually sends this packages to confirm successful received messages.

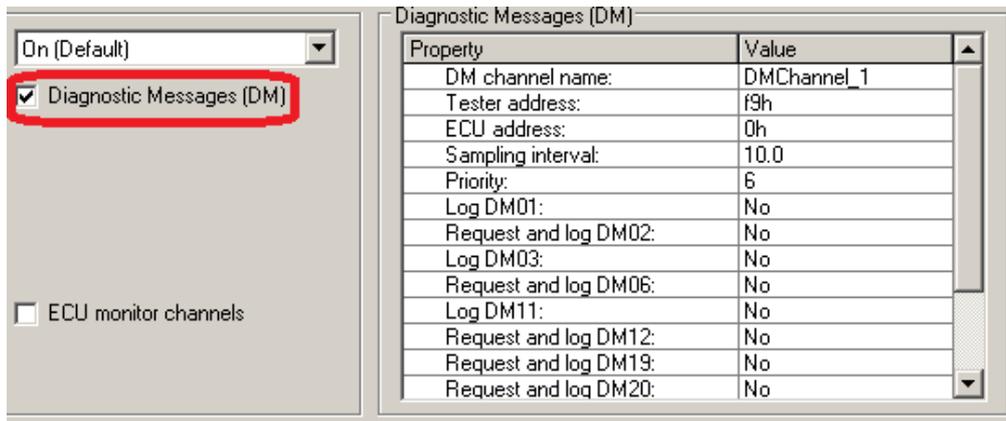
J1939-support

Activated the J1939-support, see also [Messages: Validity](#) ⁴⁰¹

Note

This option is only available, if the identifier is set to *Extended Format 2.0B* or *Extended+*.

Also, DM-services for J1939 can be set:



Diagnostic Messages (DM)

The error codes are captured with imc WAVE as Time Stamp ASCII channels. To display the channel in the curve window, select the item "Table" in the menu *Configuration\Display*. There, set the *Text Format: Hex*.

The error code format is structured as follows:

Byte	Description
Byte: 1: Format specifier	<p>A log entry's first byte is specifies the format</p> <ul style="list-style-type: none"> • ach = ASCII (VIN) • d1h = DM1 messages and others with the same structure (DM1, DM2, DM6, DM12, DM23, DM28) • d0h = record of a "Clear" command (DM3, DM11) • d2h = binary (DM19, DM20) • dfh = for errors (currently not used)
Byte: 2: for format specifiers d1h , d2h and d0h	<p>DM-number, pattern: DM# with # equaling 1, 2, 6, ...</p> <ul style="list-style-type: none"> • 1 = DM1 • 2 = DM2 • 6 = DM6 • ...
Byte: 3: for format specifiers d1h , d2h and d0h	Source
Byte: 4: for format specifier d0h	Target
Byte: 4 and subsequent: for format specifiers d1h and d2h	according to J1939

Example of DM1 and similar:

Byte	Bit	Description
Byte: 1	bits 8-7	Malfunction Indicator Lamp Status
	bits 6-5	Red Stop Lamp Status
	bits 4-3	Amber Warning Lamp Status
	bits 2-1	Protect Lamp Status
Byte: 2	bits 8-7	Flash Malfunction Indicator Lamp
	bits 6-5	Flash Red Stop Lamp
	bits 4-3	Flash Amber Warning Lamp
	bits 2-1	Flash Protect Lamp
Byte: 3	bits 8-1	SPN, 8 least significant bits of SPN (most significant at bit 8)
Byte: 4	bits 8-1	SPN, second byte of SPN (most significant at bit 8)
Byte: 5	bits 8-6	SPN, 3 most significant bits (most significant at bit 8)
	bits 5-1	FMI (most significant at bit 5)
Byte: 6	bit 8	SPN Conversion Method
	bits 7-1	Occurrence Count

 **Note**

When the occurrence count is not available it should be set to all ones which is a value of 127.

 **Example**

The following illustrates the message format for when there is more than one diagnostic trouble code.

Given:

- a=lamp status
- b=SPN
- c=FMI
- d=CM and OC

Message form will be as follows: a,b,c,d,b,c,d,b,c,d,b,c,d....etc.

With format specifier ach = ASCII

Byte	Description
Byte: 2	Higher-value byte of PGN
Byte: 3	Lower-value byte of PGN
Byte: 4	Source
Folgende Bytes	contents according to J1939 standard (ASCII string)

Abbreviations

CM	SPN Conversion Method	
DM1	Diagnostic Message 1	Active Diagnostic Trouble Codes
DM2	Diagnostic Message 2	Previously Active Diagnostic Trouble Codes
DM3	Diagnostic Message 3	Diagnostic Data Clear/Reset for Previously Active DTCs
DM6	Diagnostic Message 6	Emission Related Pending DTCs
DM11	Diagnostic Message 11	Diagnostic Data Clear/Reset for Active DTCs
DM12	Diagnostic Message 12	Emissions Related Active DTCs
DM19	Diagnostic Message 19	Calibration Information
DM20	Diagnostic Message 20	Monitor Performance Ratio
DM23	Diagnostic Message 23	Previously Active Emission Related Faults
DM28	Diagnostic Message 28	Permanent DTCs
FMI	Failure Mode Indicator	
MI	Malfunction Indicator	
MIL	Malfunction Indicator Lamp	
OC	Occurrence Count	
PG	Parameter Group	
PGN	Parameter Group Number	
PID	Parameter Identifier (SAE J1587 or SAE J1979)	
SPN	Suspect Parameter Number	

Errorframes channel

With this option the CAN controller creates a new channel to record the number of error frames within a sample interval. A name can be defined by the edit field *Channel name*.

Wake On CAN

Prerequisites

This function is only available for the new **imc BUSDAQ**, [group A4](#)¹⁴⁶. The devices must be configured via the Remote connection in accordance with the description in the device manual.

Description

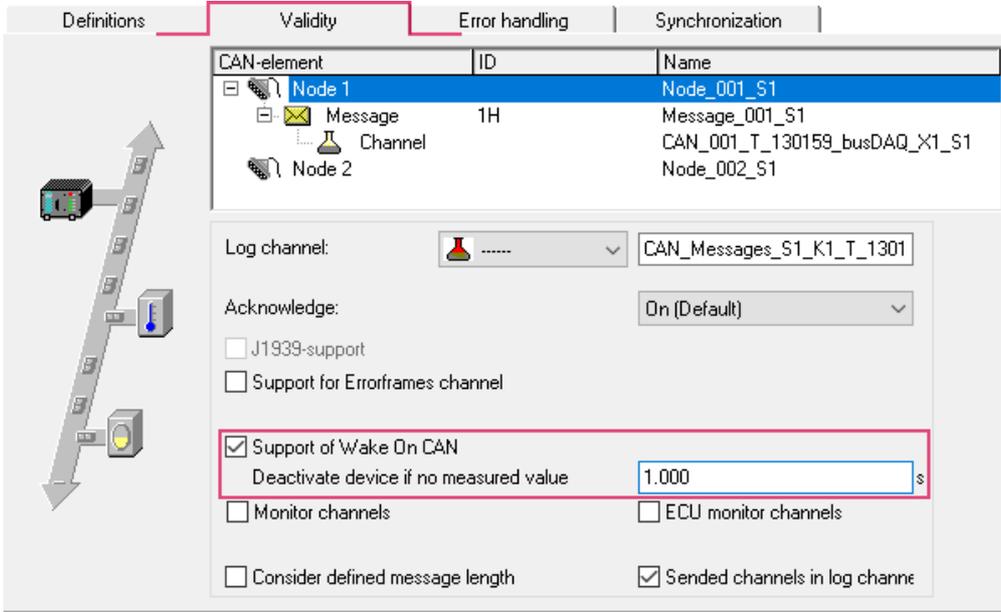
Wake On CAN enables the functionality of Sleep mode in dependence on the CAN-Bus activity. As soon as data arrive at the CAN-node, the device starts measurement. After a specifiable time without bus activity, the device returns to the Sleep mode.

Systems supply or activate the measurement device separately from the connected CAN-bus thus avoid premature startup of data capture and the resulting unnecessary power consumption.

Each node can be adjusted individually.

The prerequisite is the appropriate configuration of the Remote connection, described previously, which enables the hardware-driven Sleep mode.

- Without *Wake On CAN*, the Sleep mode is available without any additional preparation in the operating program.
- The *Wake On CAN* function is a supplemental AND condition and must be activated in CAN-Bus-Assistant.



Activation of Wake on CAN

Parameter	Description
Support of Wake On CAN:	The option <i>Support of Wake On CAN</i> appears on the page Validity if a node is selected in the CAN-element tree diagram. If it isn't shown, this indicates that your device's hardware isn't set up for this function.
Deactivate device if no measured value x s:	Enter the time interval whose elapse without the arrival of data activate the sleep mode. Intervals between 1ms and 14h (50400s) are possible.

Monitorchannels

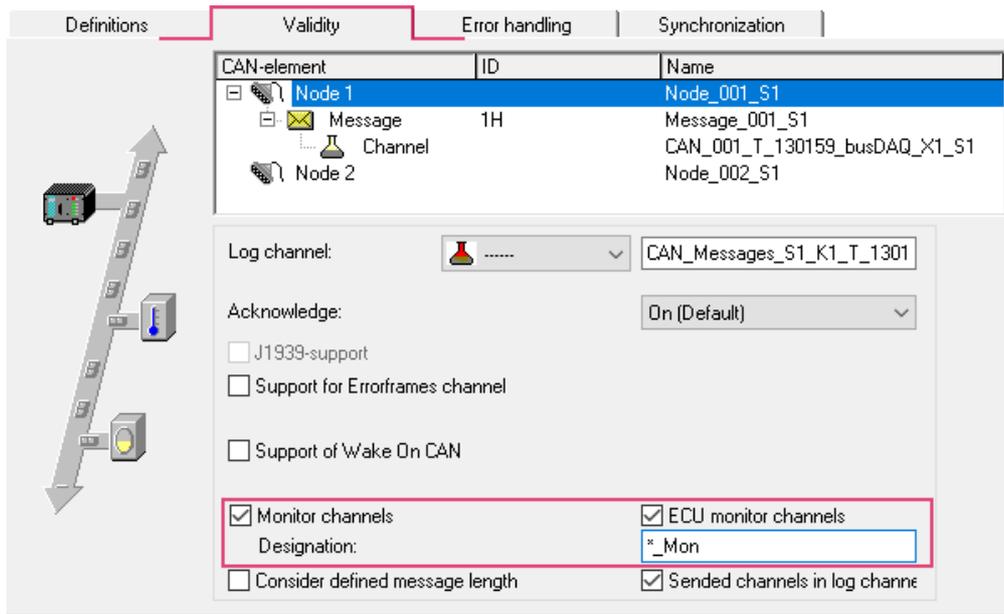
Triggered Fieldbus channels are not visible prior to the trigger event. Therefore, for analog channels there are monitor channels which can be started independently of the original channel's trigger.

Create monitor channels for all channels of a node:

It is possible to create a monitor channel for each CAN channel or ECU channel. To do this, activate the option *Monitor channels* or respectively *ECU monitor channels* on the node's *Validity* page.

Designation:

Monitor channels receive a name extension which can be freely defined in the box *Designation*.



Activate monitor channels for CAN- or ECU-channels

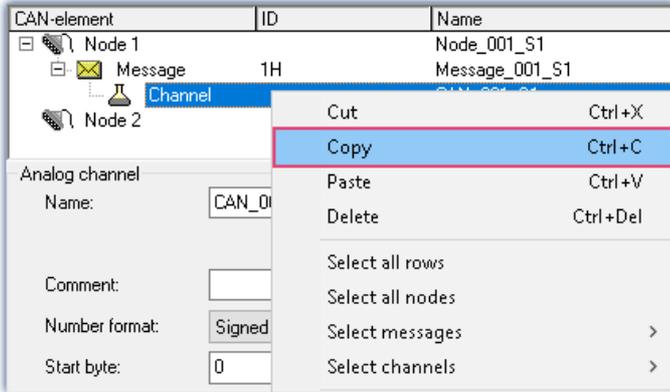
Note

Monitor channels are created for all of a node's CAN- or ECU-channels.

Create monitor channels for single channels:

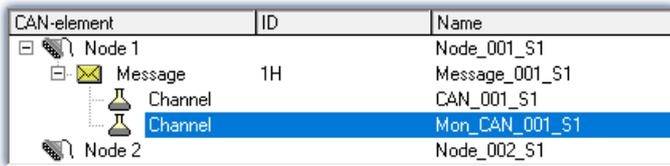
Use the Assistant to create within a channel's message an additional channel having a different name and the exact same settings. This channel can then be captured without triggering.

Copy:



Create individual fieldbus monitor channels: copy

and Paste:



Create individual fieldbus monitor channels: paste

Note

Monitor channels, created by modules that are synchronized by CAN-1, may not be sampled with a lower sampling rate.

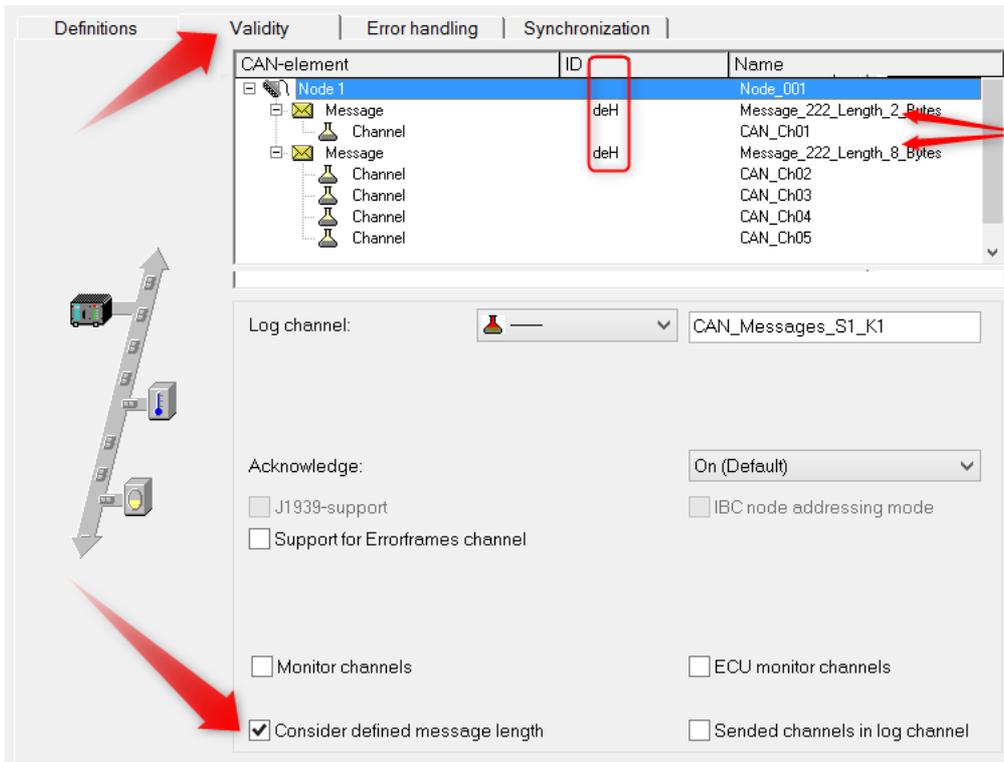
Consider defined message length

Per default, the length of a message defined in the CAN Assistant is not strictly observed and is only checked in some situations. Messages of an ID can only be created once per node, otherwise the syntax check of the CAN Assistant reports an error.

If option "*Consider defined message length*" is activated for a node, the message length transmitted on the bus is compared against the length defined in the CAN Assistant. If different, no channels are extracted from this message. However, the message protocol (log channel, CAN bus dump, log file) also records messages with differing lengths.

This option now allows a **message** with a **certain ID** to be assigned to a node **several times with different lengths**. The same ID with the same length is not allowed more than once.

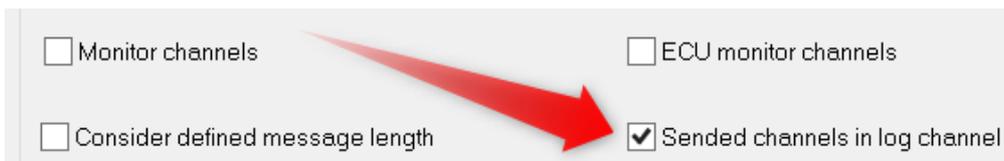
Example:



Consider defined message length

Sended channels in log channel

Transmitted CAN channel data do appear in the log channel and therefore they can be extracted from the log channel and by using `OnCanMessageReceive` in imc Online FAMOS. With this option transmit channels can be extracted from the data stream.



Sended channels in log channel

9.9.2.5.3.4 Sensor initialization

Many CAN sensors need to be initialized after being turned on. The initialization usually consists of one or more messages.

The device can send a defined set of messages to the CAN-Bus. The initialization sequence is performed only once at the very beginning of a measurement once the configuration is complete. Restarting a measurement (i.e. stopping it and starting it again) does not usually re-initialize the sensors. An exception to this is when the configuration is changed and the measurement is reinitialized.

The same applies to Autostart measurements. The initialization consists of a sequence of messages. If hand-shaking (i.e. confirmation from the sensor) is required, time-outs can be used. Many sensors perform hand-shaking by sending a return message. The unit waits for the sensor's response before sending the next message.

Initialization dialog

The initialization sequence can be defined using the *Edit > Sensor-initialization*:

No.	Node	The device is sending message							The device is waiting for confirmation			Comment	
		Send-ID	B0	B1	B2	B3	B4	B5	B6	B7	Confirm-ID		TimeOut
1	CAN 1	32H	1H	2H	4H	8H	10H	20H	40H	80H	--	--	Rotation
2	CAN 1	8H	12H	12H	12H	12H	--	--	--	--	--	--	Speed

Nodes: CAN 1, CAN classic: Node_001

Comment: Speed

Send ID (Standard): 8H No. of bytes: 4

Byte: 0: 12H 1: 12H 2: 12H 3: 12H

Confirmation

Buttons: Insert row, Append row, Delete row, Close

Sensor initialization

For initialization, messages are sent to the sensors. It is possible to specify confirmation of receipt of the messages. The list in the dialog shows the entire initialization sequence.

Parameter	Description
Nodes	Specify which of the nodes (ports) to use.
Comment	Additional description of the initialization sequence or other information - maximum 200 characters.
ID (Send)	An identifier for the message with the initialization sequence. Allowed values are from 0 to 2031 or 0H .. 7eFH for Standard-format, for Extended-format 0 .. 536870911 or 0H .. 1ffffffH. The ID is always displayed in hex format but you can enter it as a hexadecimal or decimal number. Numbers entered in Hex-format must be designated as such by using H or h (e.g. 2ah).
No. of bytes (Send)	The number of bytes to use for the message - allowed are 0 to 8 bytes.
Byte 0 to Byte 7 (Send)	The contents of the individual bytes in the initialization sequence. The allowed value range is 0 - 255 (decimal) or 0H - ffH (hexadecimal). The values are always displayed in hex format but you can enter them as a hexadecimal or decimal numbers. Numbers entered in Hex-format must be designated as such by using H or h (e.g. 2ah).
ID (Confirmation)	The identifier for the confirmation (hand-shaking) message which the sensor returns to the device to indicate that the initialization sequence has been carried out. The unit waits for this message before continuing with the next initialization sequence in the list.
Time-out (Confirmation)	The device waits for the confirmation message until the time-out limit has been reached. If the confirmation message is not received, the next initialization sequence in the list is performed. The sensor is then not initialized. The allowable values range is 1ms to 1000ms.

9.9.2.5.4 Message

9.9.2.5.4.1 Definition

Receiving a message

Message

Name:

Comment:

Device receiving / sending messages:

Identifier (ID):

Data bytes (0..8):

Clock: ms

Transmitter name:

Message log:

Receiving a message

Parameter	Description
Name	A unique name for each message with up to 65 characters. The default name for a new message is Message_" + index (e.g. 001). If several devices are being used, a _j is appended to the name, where j is the device index number. All names must be comply with imc FAMOS variable syntax, just as all other device channel names.
Comment	This text accompanies the message name and is usually intended for adding a more detailed description of the message. The maximum length of this field is 255 characters.
Device receiving / sending messages	<p>A message can only be reset from sending to receiving or vice versa if there is no channel contained in the message. This can be converted via the menu item "Edit" > "Transform message"⁴¹².</p> <p>Depending on whether the node is set with the format Extended+³⁸⁴, there can be an entry "Device sending message in Standard-format", which makes it possible to send messages in the Extended+ format with an 11-bit identifier. In this case, if the setting "Device sending message" is selected, the extended identifier having 29 bits is used.</p>
ID	<p>Each message on the CAN-Bus contains an identifier (ID) which indicates the origin and meaning of the data. No two messages in a node can have the same ID. The identifier also indicates the priority of the message and, once again, no two messages can have the same priority. When two messages are to be sent at the same time, the message with the lower ID is sent first. Allowed values are: 0 to 2031 (Standard ID format) and 0 to 536870911 (Extended ID format).</p> <p>Identifiers can also be specified as a hexadecimals. Append an "h" or "H" to the hexadecimal, e.g. 2ACH or e4h.</p> <p>If set to <i>All messages</i>, information on all messages of the current node are logged. For this purpose, only channels having the format <i>Log channel</i> may be assigned to the message (see below for a description of how to make log channel settings). Alternatively, the "Message log" control can be set to "Log CAN-Bus message". Then all node messages will be logged along with their data types.</p>
Data bytes	This is the number of bytes in the message, 0 to 8 bytes or 64 with CAN FD are allowed. It is not possible to send partial bytes in a message but it is also not necessary that every bit in the byte be occupied.

Parameter	Description
Clock	Messages are sent via the CAN-Bus at a regular rate. The value for the clock specified here is a set point, not necessarily the actual value. The clock is a property of the sensor. The allowed range is 0.001ms to 100,000ms. The sampling rate of the CAN channel to be created is determined by the clock value specified here.
Transmitter name	The transmitter name describes the transmitting sensor. This name is not used for any other purposes - therefore it need not be unique.
Message log	<p>If you wish to log all of a node's messages, activate the log channel on the page with the tab "Validity". You don't have to do anything else.</p> <p>If you wish to log individual messages, again, activate the log channel in the "Validity" tab first. Then create an additional receiving message and select "<i>Log CAN bus message</i>" under <i>Message log</i>. Select the desired message using the identifier.</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p>Note "Identifier (ID)" as "All Messages"</p> <p>The setting "<i>Identifier (ID)</i>" on "<i>All Messages</i>" originates from the previous setting method and leads to an error message with the current version. Instead, activate the log channel in the Validity tab of the node and remove all protocol messages with specific IDs if necessary.</p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p>Note Manual evaluation TSA</p> <p>If the log channel is evaluated by the TSA functions in FAMOS manually, the CAN-ID is handled as user date: 4 bytes without sign in INTEL order. If the Most Significant Bit is set, it is an extended-ID, otherwise standard ID. To get the ID itself, the value must be masked with 1fffffffH.</p> </div>

Sending messages

Message

Name:

Comment:

▼

ID: ▼

Data bytes (0..8): ▼

Clock: ms ▼

Receiver:

Replacement values: B0 B1 B2 B3 B4 B5 B6 B7

Sending a message

The CAN-Bus-Assistant is used to create a formal framework of messages and channels. The message can be set by imc Online FAMOS or with process vector variables, see below.

Parameter	Description
Name, Comment, ID, Bytes	<p>As for messages to be received.</p> <p>Data bytes for CAN FD: Messages can be sent at the data rate set for FD³⁸³. For this purpose, the option Bit Rate Switch must be activated, which appears at the CAN FD node.</p> <p>Data bytes (0..64): <input type="text" value="20"/> <input checked="" type="checkbox"/> Bit Rate Switch</p>
Receiving / sending messages	<p>A message can only be reset from sending to receiving or vice versa if there is no channel contained in the message. This can be converted via the menu item Edit\Transform message⁴¹².</p>
Clock	<p>Messages are sent via the CAN-Bus at a regular rate. The <i>Clock</i>-value is the interval between transmission of messages. The interval must be at least 1ms. The value set here serves in imc Online FAMOS as the default setting for the clock rate for cyclic transmission in the Formula-Assistant.</p>
Receiver	<p>The receiver name describes the receiving sensor. This name is not used for any other purposes - therefore it need not be unique.</p>
Replacement values	<p>When messages are transmitted with the help of imc Online FAMOS, the replacement values serve as data bytes if there are no current data or if certain message bytes are not filled.</p> <p>With CAN FD, from the 9th Byte onward, additional boxes for entering replacement values appear.</p>

Note

If a [node](#)³⁸⁴ is set to **Extended** or **Extended+ format**, a message can be send as **extended** or as **standard-format**! By default, the message is then sent in **extended** format. To send a message on an **Extend(+)** node in **standard format**, select "*Device sending message in Standard format*".



Device sending messag in Standard format.

Send channels with imc Online FAMOS

The CAN-Bus-Assistant is used to create a formal framework of messages and channels. Using imc Online FAMOS, this framework is filled with data and transmitted. The channels or single values are transmitted in the format set in the CAN-Bus-Assistant. As for the reception of CAN-messages, channels are defined for each message with their data types, Start-bits, Start-Bytes, number of bits etc. imc Online FAMOS uses these data to generate a transmission function for each message. With the help of this function, for instance, a virtual channel in imc Online FAMOS is assigned to the channel defined here in the CAN-Bus-Assistant and is sent in the specified data format.

With imc Online FAMOS, it is also possible to send larger messages in conjunction with **CAN FD**.

CAN messages can be sent **asynchronously** or **synchronously** in imc Online FAMOS.

Asynchronous transmission is performed **without control commands** by the *SendMessage_Message_name* after a transmission message has been created in the CAN assistant. If **control commands** are **activated**, transmission takes place in the *OnTriggerMeasure* of the channel to be transmitted.

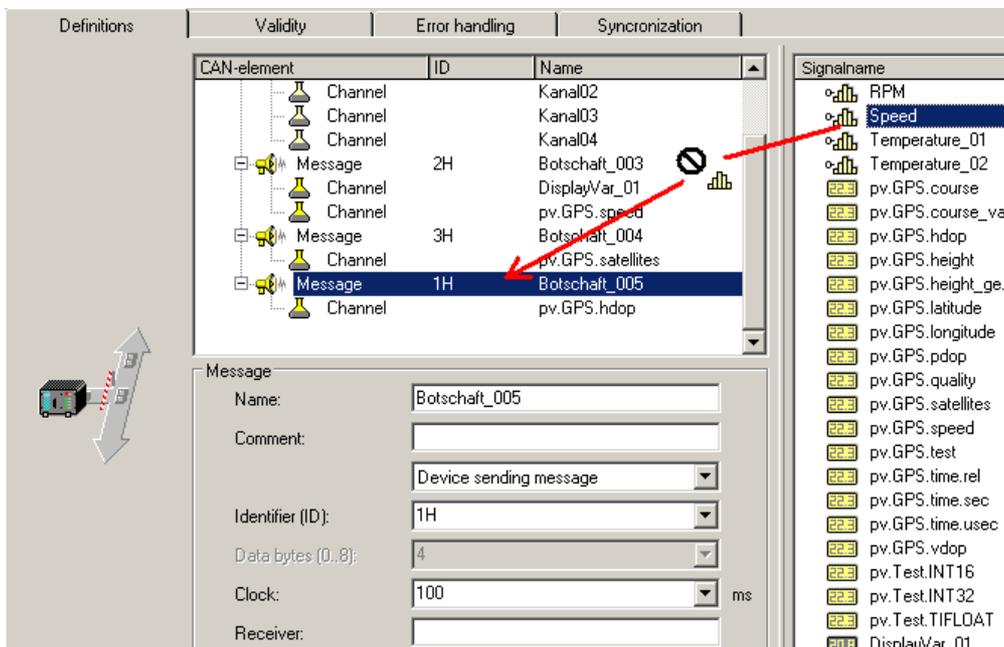
Sending is **synchronous** in a *SyncTask*. Due to the system, only **one SyncTask** can send CAN messages at a time. Sending in different synchronous tasks is not allowed.

For examples to send messages, see [Sending CAN messages](#) ⁴³³.

Send channels with process vector variables

Channels, process vector variables and display variables can be assigned out of the CAN-Bus-Assistant directly. If a message is set to send mode, a list with *Sending channel name comes up*. Via double click or drag&drop the channel is assigned to the send message. No further settings in imc Online FAMOS are necessary.

Prerequisite is the activation of imc Online FAMOS, even if the editor is not needed.



Sending process vector variables via CAN

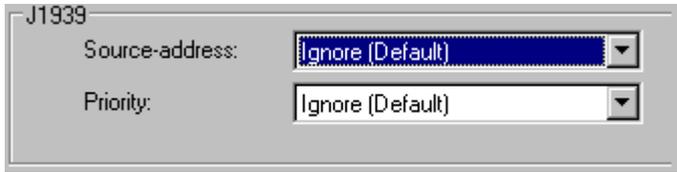
 Note

- Only messages can be copied. Single send channels can not be copied.
- When sending messages, if there is no recipient (yet), an error message sometimes appears, indicating that CAN-messages have been lost.

9.9.2.5.4.2 Validity

J1939

To work with J1939, the setting **J1939-support** is activated on the control *CAN-node* on the *Validity* page. After that the setting options on the *Validity for messages* card become visible:



J1939 settings at Message-Validity

Both the **Source-address** and the **Priority** of messages can either be accounted for or ignored. With J1939, the source address and priority are ignored by default. The source address is in the bottom 8 bits of the extended identifier and the priority in the top 3 bits.

With *Ignore (Default)*, the corresponding bits are masked out.

In J1939, the extended message identifier is comprised as follows:

$$ID_Extended = Source-Address + ProgramGroupNumber * 28 + Priority * 2^{26}$$

For standard IDs, the following applies:

$$ID_Standard = PGN-number + Priority * 2^8$$

When importing from a DBC-file, J1939 messages could be greater than 8 Byte. Those messages can be received from several messages.

Note

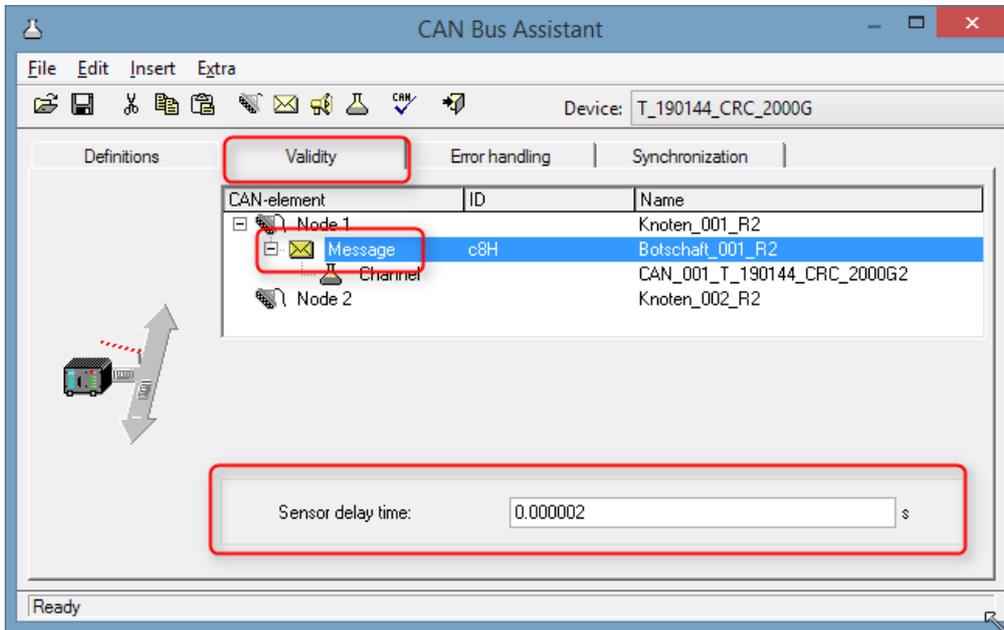
With the J1939 protocol, by default only bits 8 ... 25 (Parameter Group Number, PGN) of CAN messages are taken into account by default, bits 0 ... 7 (Source Address, SA) and bits 26 ... 28 (Priority) of CAN messages are ignored.

If the bits 8 ... 25 (Parameter Group Number, PGN) do not lead to uniqueness of the message IDs, the bits 0 ... 7 (source address, SA) or / and the bits 26 ... 28 (priority) of the CAN messages must also be taken into account.

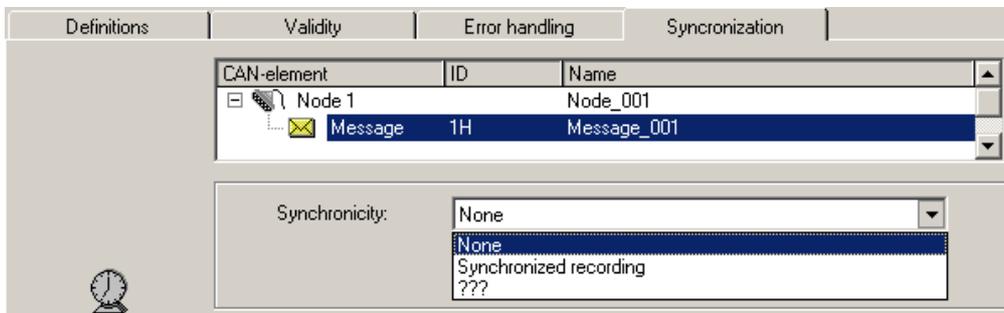
These settings must be changed in the J1939 node in the CAN messages that prevent uniqueness of the message ID (in case of doubt for all CAN messages).

Sensor delay time

Here it is possible to take into account the sensor's delayed response upon reception of CAN-messages. Differences in transit times can be compensated.



9.9.2.5.4.3 Synchronization



Synchronization a message

Synchronization is activated at the node. At this node, other devices which do not understand the CAN-1 protocol can be used alongside imc CANSAS. It should be possible to record these devices' messages, even without monitoring of the synchronization. If synchronized recording is selected for such messages, error messages will be posted whenever the messages fail to arrive at the clock rate expected.

Since the integrated imc CANSAS software knows the scope of the imc CANSAS modules' functions, their messages are automatically pre-set appropriately. For example, if a imc CANSAS module is set as the CAN-1 Slave in the imc CANSAS software, its message is automatically set in the CAN-Bus-Assistant to "Synchronized recording".

9.9.2.5.5 Channel

9.9.2.5.5.1 Definition

Analog CAN channels

Analog signals are digitized by the CAN-sensors with a specific number of bits. The numerical format can be set to *signed integer, unsigned integer or real number, or digital (1 bit)*.

Analog channel

Name:

Extracted name:

Comment:

Number format:

Start byte: Start bit:

Number of bits: Byte order:

Unit:

Scaling:

Factor: Offset:

Definition of a analog channel

Parameter	Description
Name	See Message ^[396]
Extracted name	Name for channel ^[407] for transmission channels extracted from the log channel ^[407] . To avoid duplicate names in the send dialog ^[399] .
Comment	See Message ^[396]
Numerical format	Data can be sent and received as signed or unsigned integers, real numbers and in digital (1Bit) format. The option 16bits (see below) is only available for receiving mode.



Note

Information about the numerical format

Some manufacturers may not indicate clearly in their CAN tables which number format they use.

For example, the table may indicate that Bytes 1 and 2 represent the guidance angle and that this value is to be multiplied with 0.0639 to arrive at the actual value in degrees. If no other information has been given, there is no definite way of determining the number format other than trial and error. Our tech support cannot help you in such cases! Sometimes however, there are implicit clues as to what the format is.

Examples:

- If the range minimum is 0 and the maximum is ffff hex, then unsigned integers are being used because they always start at zero.
- If the range is given as 7fff = -700°C and 8000 hex = +700°C, then signed integers in two's complement form are being used. The zero (0 hex) lies directly in the middle between positive and negative integers.

The most certain method of determining the number format is to run the sensor through its entire range.

If the correct number format has been selected, the curve that you measure will be continuous. Otherwise, the upper and lower halves are exchanged, which appears as a discontinuity. A discontinuous curve is therefore an indication of the wrong integral format being used.

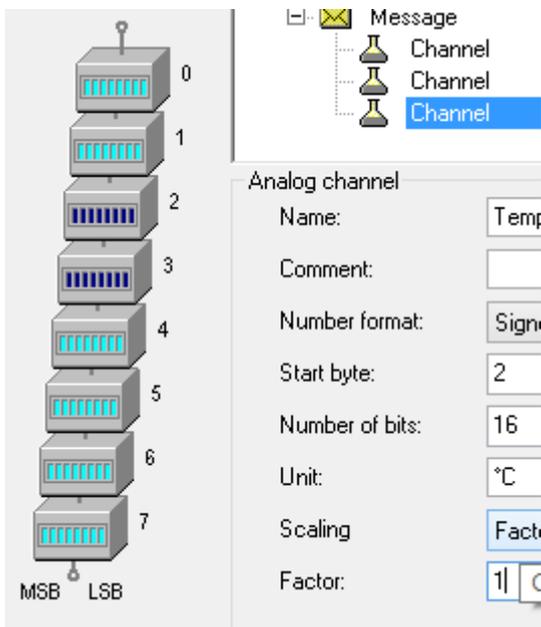
Parameter	Description
Start Byte	The start byte is the location of the message byte where the channel bits actually begin. It is allowed to have overlapping channels within a message. In other words, two channels can share the same start byte in a message.
Start bit	The start bit specifies where the number starts within the start byte.
Bits	This is the number of bits used in the channel. The remaining bits continue in the direction of the increasing byte indices. For signed and unsigned integers, 1 to 32 bits are allowed. Real numbers can range from 32 (floating point) to 64 (double).

Note

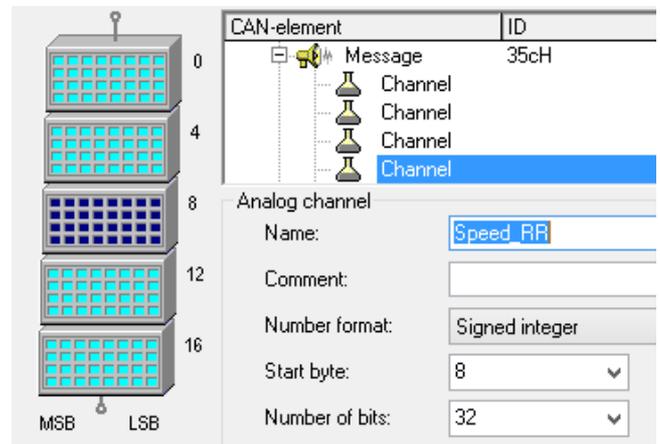
Information about the graphical display and the format

Graphical display of the position within the message

The Assistant displays the specified position within the message graphically. In this way, any overlapping position allocations can easily be detected:



Bit allocation in classic CAN-Bus



Bit allocation in CAN-FD

Note

Byte Order | Format

This is the bit order used by the CAN module, either **Intel format** or **Motorola format**.

Intel format: The start bit is the LSB. The **LSB** has a **lower byte index** than the MSB.

Example: start byte = 5, start bit = 3, bits = 18

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 5	X	X	X	X	LSB			
Byte 6	X	X	X	X	X	X	X	X
Byte 7				MSB	X	X	X	X

Motorola format: The start bit is the MSB. The **LSB** has a **higher byte index** than the MSB.

Example: start byte = 5, start bit = 3, bits = 18

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 5					MSB	X	X	X
Byte 6	X	X	X	X	X	X	X	X
Byte 7	X	X	X	X	X	LSB		

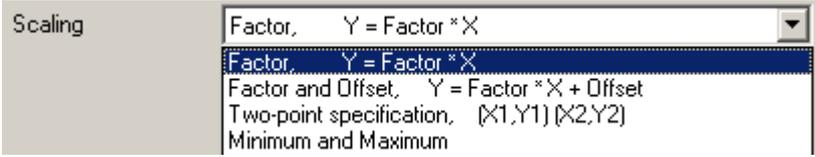
Note on recognizing Intel or Motorola byte order

- **The CAN tables from manufacturers do not always indicate whether Intel or Motorola byte order is used.** It is sometimes possible to determine the byte order indirectly. Within one message typically only one of the variations will be used.
- *Example:* Two byte data packet, message Byte 3 is LOW and message Byte 4 is HIGH Intel byte order because HIGH byte contains MSB.
- *Example:* Bits 0 to 14 contain data and Bit 0 is LSB
Data is contained in Bytes 0 and 1 but only the lowest seven bits in Byte 1 are occupied. Therefore, LSB is in the byte with the smaller index which means that Intel format is used.

Parameter	Description
Unit	The physical units of the channel, maximum 24 characters. The units are used when displaying the channel, e.g. on the Y-axis in a curve window. Note that units should be specified without prefixes. Examples of units that you can use are: m, s, kg, N, m/s. If you need to use prefixes (e.g. n-, μ-, M-) or non-SI units (e.g. inches, pounds), always place these between quotation marks (e.g. "inches").
Scaling	Here is determined how to convert the physical value to a CAN bus integer. There are different choices available, to determine the conversion.

Note

Notes for scaling



The Scaling options

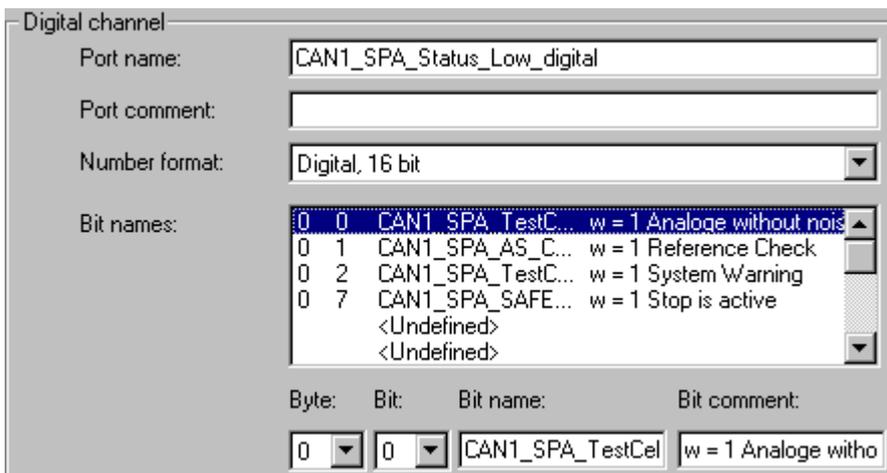
Y = physical measurement value

X = CAN bus integer value

Scaling	Description
Factor, $Y=Factor \cdot X$	<p>Receiving: for received signals (X) the following applies: $Y = Factor \cdot X$</p> <p>Sending: for physical values (Y) to be sent, the integer value is calculated as follows: $X = Y / Factor$</p>
Factor and Offset, $Y = Factor \cdot X + Offset$	<p>Receiving: for received signals (X) the following applies: $Y = Factor \cdot X + Offset$</p> <p>Sending: for physical values (Y) to be sent, the integer value is calculated as follows: $X = (Y - Offset) / Factor$</p>
Two-point specification, (X1, Y1), (X2, Y2)	<p>X1: x-coordinate of 1st point Y1: y-coordinate of 1st point</p> <p>X2: x-coordinate of 2nd point Y2: y-coordinate of 2nd point</p>
Minimum and Maximum	The lower limit of the specified number format with the specified number of bits corresponds to this minimum (e.g. signed integers, 8 bit, range: 0 to 255, minimum corresponds to 0, maximum corresponds to 255)

Digital CAN channels

The number format is set to Digital, 16 bit (only to be set for reception of CAN-Bus messages).



Definition of a digital channel

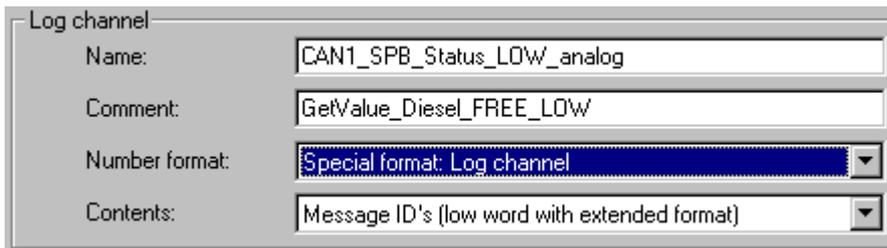
A digital CAN channel is analogous to a digital 16-bit port in imc WAVE - it consists of 16 separate bits or one **word**. Each bit is of course a binary value, i.e. either 0 or 1. Each bit in the word does not actually have to be used. The entire 16 bit port receives a name as do each of the bits therein. These names are employed when using them in other components of the software, e.g. in triggers and imc Online FAMOS.

Parameter	Description
Name, Comment	see Message ³⁹⁶
Numerical format	must be set to 16 bit
Byte	determines in which of the message's bytes the digital bit is located.
Bit	position of the bit within the selected byte
Bit name, Bit comment	see Message ³⁹⁶

Log-channel

The numerical format is set to *Special format: Log-channel*. This special log channel allows filtering by low or high word of the message IDs or the number of data bytes. A log channel that contains the complete data transfer is set in [Node/Validity](#) ³⁸⁵.

This log channel is only available for messages in **receive mode**. At the same time, the message affected must be set to *All messages*.



Definition of a log-channel

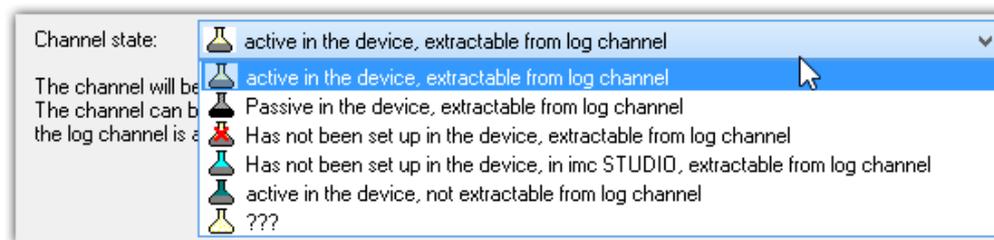
Information about all of the node's messages is logged. The message-IDs or the number of data bytes can be logged. In default format (11-bit message IDs), the message ID's are logged directly. In Extended-format (29-bit message-IDs), the High and Low word of the message ID's can only be logged separately. Each ID must then be constructed by combining the Low word and High word.

Contents: Message-IDs (low word for Extended-format); High word of message-IDs (only for Extended-format) or Number of data Bytes.

9.9.2.5.5.2 Validity

The dialog illustrated below appears if the system is set to receive CAN-Bus messages.

Channel-State

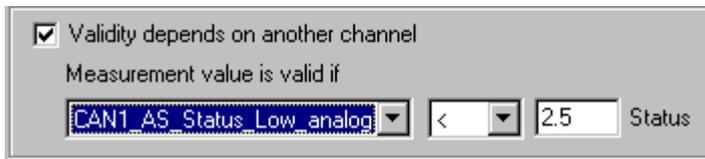


		log channel ³⁹⁷ extractable with		
device setup		e.g. imc FAMOS	imc STUDIO	Comment
	active	yes	no	Normal data acquisition on a CAN-channel. Default: active
	passive	yes	no	Normal data acquisition on a CAN-channel. Default: passive
	no	yes	no	Channel extracted offline, with imc FAMOS for instance.
	no	yes	yes	Channel extracted offline, with imc FAMOS for instance.
	active	no	no	Normal data acquisition on a CAN-channel. No data transfer to the log channel.

If a channel is present in the *Device channel list*, it can be either activated or deactivated for recording. I.e., the channel appears as a Fieldbus channel in the Setup.

Extensive CAN-configurations (e.g. *.DBC) may exceed the maximum channel count which the device can administer. In such a case, it is possible to exclude certain channels from the device using the channel status " Has not been set up in the device". Then these ones don't appear in the Setup. The data belonging to "not activated" channels can instead be extracted subsequent to the measurement. For this purpose, a **log channel** must be activated for the associated node; see [here](#)³⁹⁷.

Validity depends on another channel



Depending on another channel

If the current channel is to depend on another channel in the same message, check the option *Validity depends on another channel*.

The channel on which the current one depends is referred to as a *mode channel*. A mode channel must be contained in the same CAN message and use the unsigned integer number format with less than 16 bits. The drop-down list to the left contains all of the valid mode channels.

The following two conditions must also be fulfilled by a mode channel:

- A channel cannot depend on itself.
- a mode channel cannot depend on another mode channel.

The value of the current channel is only valid when the mode channel condition (e.g. $CAN_001 \leq 5$) has been satisfied. Otherwise, your measurement device behaves as if the message has not been received. Enter the comparison value in real number format using the "analog" range of the mode channel - do not enter an unsigned integer value!

9.9.2.5.5.3 Error handling dialog

The dialog illustrated below appears if the system is set to *receive* CAN-Bus messages.

Error handling

This dialog allows you to define error handling for invalid measurement values and time-out errors. If an invalid measurement value appears, or if after a certain amount of time has elapsed no further message has arrived in the device, you can perform error handling. To enable it, check the Error handling box and the desired options Invalid measurement value and/or Time out.

Invalid measurement value:

Enter the comparison value for analog channels in real number format using the display range. Select a comparison operator from the drop-down list.

Enter the comparison value for digital CAN channels (ports) using the decimal equivalent of the binary value, i.e. no bits = 0, add 2^{bitindex} for each bit which is HIGH. For example if the first and third bits (Bits 0 and 2) are HIGH and all others are LOW, the decimal equivalent is $0 + 2^0 + 2^2 = 5$. If the comparison condition is not met, error handling is performed (see below).

Time out:

Define a time out limit here. This is the maximum time which the sensor has to send the message. Many sensors send messages at regular intervals - this option is often useful for monitoring them.

Error handling:

The error handling box is displayed whenever any of the two above options is checked. Select one the following error handling options from the drop-down list:

- Ignore channel with this message
- Last channel value
- Replacement value

The error handling functions slightly differently, depending whether the channel is continuously sampled or whether time-stamping.

Error correction	Effect with continuous sampling	Effect with time stamping
Ignore channel with this message	Same as <i>Last channel value</i>	System behaves as if the message was never received - no time-stamped sample is created.
Last channel value	The last valid channel value is generated each sampling interval until a valid sample arrives.	<i>Time out error:</i> a new sample is generated using the last valid channel value and is stamped with the time-out limit. <i>Invalid value:</i> a sample is generated using the last valid channel value and is stamped with the message time stamp.
Replacement value	The replacement value is generated each sampling interval until a valid sample arrives.	<i>Time out error:</i> a new sample is generated using the replacement value and is stamped with the time-out limit. <i>Invalid value:</i> a sample is generated using the replacement value and is stamped with the message time stamp.
No error correction used	Same as <i>Last channel value</i>	Same as <i>Ignore channel with this message</i>

 **Note**

- **Input range of substitute values for error handling**

Certain substitute values may lie outside of the input range for the CAN channel.

- Channels with 16 bits or less can be given substitute values having a 16-bit input range. Example: For a CAN channel with 8 bits, a substitute value of up to 16 bits is permitted.
- Channels with more than 16 bits and less than or equal 32 bits can be given substitute values having a 32-bit input range. Example: For a CAN channel with 24 bits, a substitute value of up to 32 bits is permitted.
- Consequently, the following exceptions exist: Channels with exactly 16 bits or 32 bit-channels only permit substitute values having the same input range.

9.9.2.5.6 Menu

File

Menu item	Description
Restart (Ctrl + N)	Clears the node-, message- and channel lists. A single node is created with default settings.
 Load (Ctrl + O)	Loads a complete CAN-configuration. A dialog is called which enables you to select a CBA-file from a folder of CBA-files. For the selected device (or the selected slot if there are several in the device), the current CAN-configuration is completely replaced by the one from the file. The current configuration is first deleted and then the new one is loaded from the CBA-file.
 Save (Ctrl + S)	Saves the current CAN-Bus configuration as a *.CBA file. If the configuration has not yet been saved, the Save As... dialog is started.
Save as	Saves a CAN-configuration under a new filename.
Import	Loads can configuration a selected node. Beside the CAN assistant format CBA, the DBC is supported, if the PC has the DBC option installed or the device is prepared for ECU (A2L).
Export	With Export, the sendable messages of one node can be saved as DBC file.
 Quit	This quits the CAN-Bus Assistant. The CAN-Bus Assistant automatically checks all of the settings first before closing and informs of any errors. You are then given the option of quitting or not. If you decide not to, The CAN-Bus Assistant shows you the error or errors so that you can fix it/ them easily. Note: If you quit the CAN-Bus Assistant without fixing the errors, no CAN channels are created.

Edit

Menu item	Description
 Cut (Ctrl + X)	Cuts the selected lines to the Windows Clipboard. Cutting a message also copies all of the channels contained in the message; cutting a node copies all of the messages and all of the channels therein. Once copied to the Clipboard, the lines are deleted from the list.
 Copy (Ctrl + C)	Copies the selected lines to the Windows Clipboard. In contrast to the Cut function, only those lines are copied to the Clipboard which have been selected.
 Paste (Ctrl + V)	Pastes lines from the Clipboard into the current configuration.
Delete (Ctrl + Del)	Deletes the selected lines from the configuration. Note: Deleting a message also deletes all of the channels contained in the message; deleting a node copies all of the messages and all of the channels therein.
Find	Using the <i>Find</i> dialog, you can search by names of channels, messages, nodes, etc.
Select All Lines	This selects all nodes, messages and channels in the current configuration. This makes it easy to cut or copy the entire list.
Select All Nodes	This selects all nodes in the current configuration. This makes it easy to adjust settings in all of nodes simultaneously or to see which settings are identical in all nodes and which differ.
Select All Messages	This selects all messages in the current configuration.
Select All Channels	This selects all channels in the current configuration.
Show Messages	In this display mode, only the nodes and messages are visible.
Show Channels	In this display mode, all of the nodes, messages and channels are visible, i.e. the entire configuration.

Menu item	Description
Sort	<p>Messages can be ordered by the following criteria:</p> <ul style="list-style-type: none"> • messages by name • messages by IDs • channels by name • channels by start bits
Transform messages	<p>A receive message is converted into a send message. This means that a configuration with reception messages can be used to simply create a sending station.</p> <p>"Empty" transmission messages can also be transformed into receiving messages. This conversion may generate (automatically) unique message names. Therefore, the same CAN Assistant content does not necessarily result from the conversion of receiving to sending messages and subsequent conversion of sending messages into receiving messages.</p>
Sensor Initialization	See detailed description ³⁹⁴ .
Apply	This applies the current settings immediately and checks that they're correct. If a setting is incorrect, a message in the status bar indicates this. Most settings in the CAN-Bus Assistant are applied automatically or when you exit the dialog, so this menu option should be rarely needed.
 Complete Check	We recommend that you check the configuration often. If any errors are found, they are indicated in the status bar and the incorrect field or fields are highlighted. This makes it easy to fix any incorrect or implausible settings.

Insert

Menu item	Description
 Nodes	<p>Inserts a new node into the configuration list. A node is only inserted when exactly one line is selected. If a node is selected, the new node is inserted above it. If a message or channel is selected, the new node is inserted above the next node which follows, or to the bottom of the list if no other nodes exist.</p> <p>The first node is always assigned to CAN Port 1 and the second node to CAN Port 2. None of the other nodes can be assigned to any of the ports. If you create any more than two nodes, they can only be used temporarily and should be deleted before quitting the CAN-Bus Assistant. Otherwise the configuration will be "incorrect" and no CAN channels will be created.</p>
 Message	<p>Inserts a new message into the configuration list. A message is only inserted when exactly one line is selected. If a node is selected, the new message is created below any existing ones. If a message is selected, the new message is inserted above it. If a channel is selected, the new message is inserted below the message to which the channel belongs.</p> <p>By default, the message created is always a receiving message. See also Receiving a message ³⁹⁶.</p>
 Sendable message	Inserts a new message to send channels. See also Sending messages ³⁹⁷ .
 Channel	<p>Inserts a new channel into the configuration list. A channel is only inserted when exactly one line is selected. If a node is selected, a message is created with the new channel. The new message is created below any existing ones. If a message is selected, the new channel is inserted below any other existing channels in the message. If a channel is selected, the new channel is inserted below it.</p> <p>By default, the channel created is always an analog channel.</p>

Menu item	Description
ECU	Shows the control unit properties ^[440] and the values to acquire. That entry is only enabled for devices prepared for the support of ECU.

Extras

Menu item	Description
Options	See " Options " ^[416]

9.9.2.5.6.1 Import

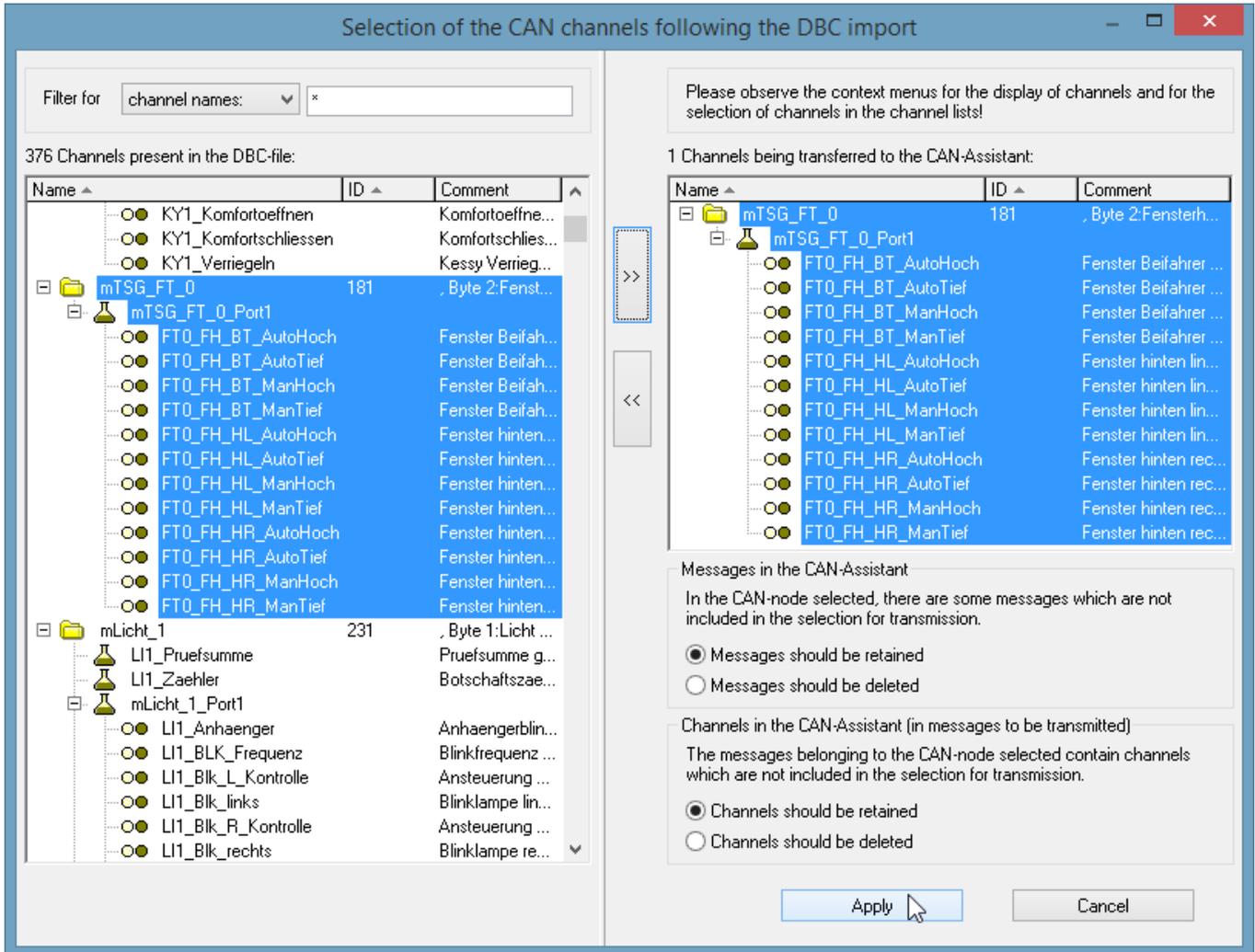
Note

At this time, direct import of more than 1000 channels is not possible. In order that channels from such an import file can still be used, any channels not needed must be blocked for the system. Load the import file, go to the page [Channels-Validity](#)^[407] and assign to the unneeded channels the status "Do not adopt in system"  -.

DBC

The import of CAN-Database (*.DBC) requires the purchased and installed expansion module "Import CAN-Database (*.DBC)". DBC-files can be created using the program CANdb (© Vector Informatik GmbH) from Vector Informatik GmbH. CANdb is a program for administrating the CANalyzer's (© Vector Informatik GmbH) CAN system data base. DBC-files which were created using Versions 3.03 to 3.20 of the CANdb program are supported. These versions of CANdb belong to Versions 2.0a to 3.0 of the CANalyzer.

DBC-files mostly contain all channels sent via a single CAN-node. Upon importing such channels, you have the option to filter for targeted channels relevant to your experiment.



DBC-import filter

The [J1939⁴⁰¹](#) standard will be adhered to with DBC files.

CBA

As with loading a **CBA**-file, the current CAN-configuration for the selected device (or the selected slot if there are several in the device) is completely replaced by the data from the CAN-Database, i.e. the current CAN-configuration is deleted and then the CAN-Database (DBC-file) is imported. In the CAN-Bus-Assistant, a node is added as a default because the CAN-Database does not contain imc WAVE node data such as the Baud rate. If the current configuration already includes a node, the node remains and retains its properties even after the DBC-file is loaded. But the node settings should be re-checked.

ECU (A2L)

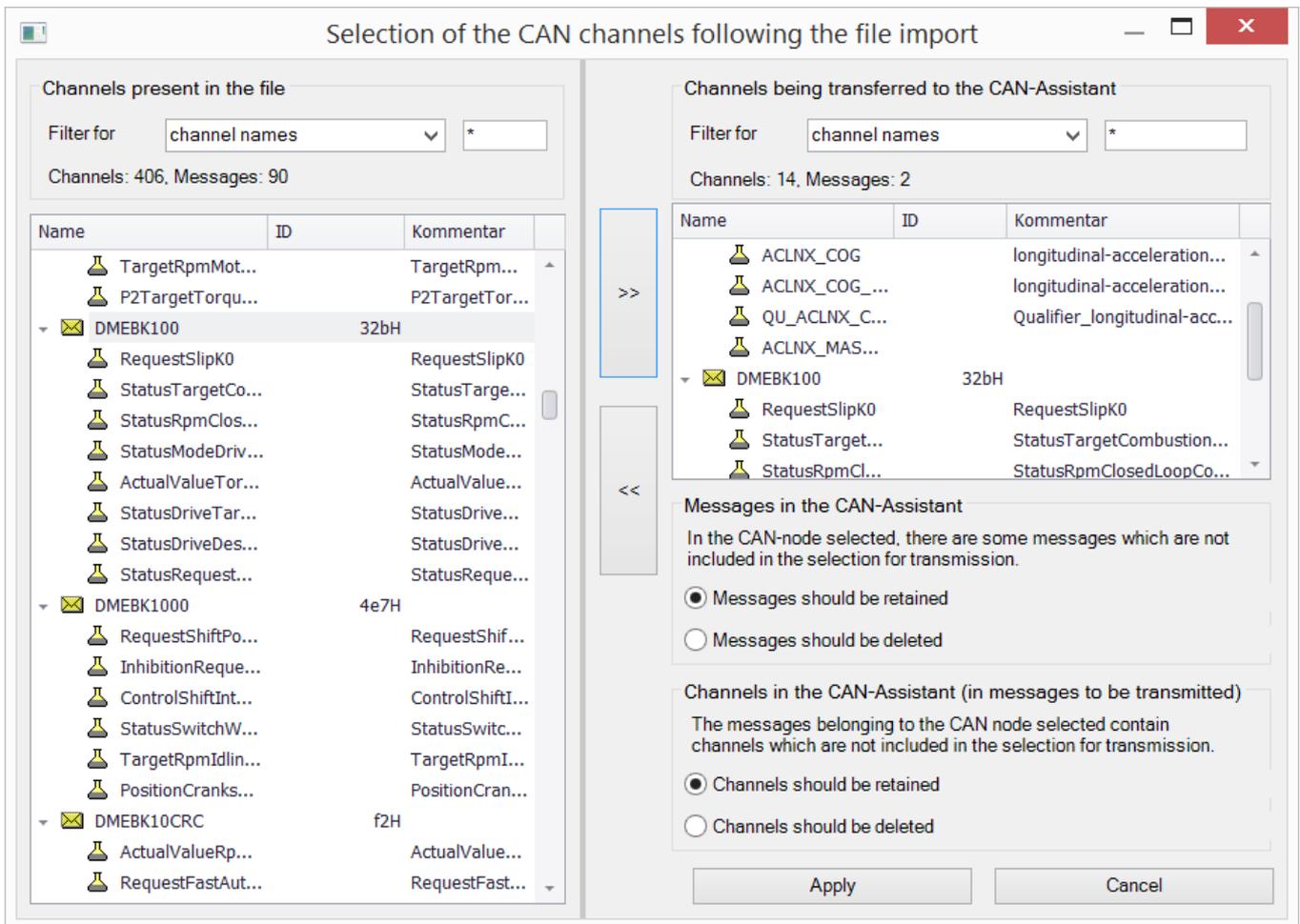
For users of the *Import CAN-Database (*.DBC)* and **ECU (A2L)**, the same applies as for loading a complete configuration. Additionally when importing ECUs a single ECU can be selected (not only a node). Then that ECU will be replaced. In case the name of the ECU and the one in the file is different, a message box appears to confirm the replacement. If the A2L file contains short and long event names, the long ones are used.

The extended format XCPplus is supported.

Further settings for too long channel names and characteristic curves of type CURVE and VAL_BLK are available in menu [Extra\Options](#)⁴¹⁷.

ARXML

The ARXML format was defined by the development partnership **AUTOSAR (AUTomotive Open System ARchitecture)**. Once the file has been opened, it is possible to import individual messages/channels in the selection assistant.



CAN FD

CAN FD configurations can be exported to CAN-devices. The configuration is then converted to CAN and any resulting errors must be subsequently eliminated by the user.

9.9.2.5.6.2 Options

For changed message clock rate:

If the option "Apply as sampling interval for respective channels?" is active, the sampling rate of the Field bus channels in the imc WAVE software is automatically set.

Example: A CAN database is imported. Further settings for the Fieldbus channels are made in the CAN-Bus-Assistant. The sampling rates on the Base-page are at first set to match the associated channel's message rate. However, the user can make settings on the Base page which deviate from that clock rate. If you then return to the CAN-Bus-Assistant and re-load the database, or if you change the message clock rate directly, the sampling rate in the Base page is set equal to the message rate, if the option "Apply as sampling interval for respective channels?" is active. If it isn't active, the sampling rates previously set in the Base page are retained.

CAN-database (*.DBC)import:

Treatment of duplicate channel names

Treatment of duplicate channel names are not allowed in imc WAVE. In order to load a CAN-configuration with a duplicate name anyway, there are these possibilities:

Configuration	Description
Retain	The names are adopted. But the configuration cannot be applied. The Assistant highlights the duplicate names, which must then be edited manually by the user.
Don't accept	Duplicate channels are marked as defective and ignored by the administration.
Append message ID / Append message name	The message ID or message name is appended to the duplicate names, which are thus rendered unique.
Don't accept in CAN-Assistant	The duplicate channels are not imported to the CAN-Bus-Assistant and are thus no longer displayed. The channel with the lowest ID, in other words, with the highest priority, is adopted.

Channels with invalid properties

Configuration	Description
Don't accept	With "Don't accept", channels with incorrect properties (e.g. invalid bit count, invalid start bit, invalid characters in the channel name,...) are not adopted in the administration. The check then doesn't acknowledge any error. But the channels are correspondingly marked in the CAN-Bus-Assistant display. The channels can be re-activated if required.
Don't accept in CAN-Assistant	The invalid channels are not imported to the CAN-Bus-Assistant and are thus no longer displayed.

Calls dialog for the name extension

If this is selected, when importing a DBC-file, the dialog for the name extension is displayed after the file is selected. This dialog is for the purpose of being able to import the same DBC-files on different nodes (slots) of a device and still have unique names.

Set J1939-Protocol, if byte number of a message > 8 and extended-format

With CAN FD, the protocol can be set to J1939 in Extended Format automatically.

Import the specified devices as CAN-Senders, filter for device name

Messages of marked device names for send devices will be read as "Send messages".

When importing A2L files:

- Replace
 - characteristic curves of the type CURVE and VAL_BLK with type VALUE. Further information see [here](#)⁴⁵⁸.
 - Measurement vectors with ARRAY_SIZE > 0 and matrices as single vector elements (measurement channels).
- Option for saving original names of the imported channels in the channel comments.

9.9.2.5.7 imc WAVE - Configuration / processing of fieldbus channels

Analog inputs

The channels created using the CAN-Bus-Assistant appear in the channel list under **Analog channels > Fieldbus: Analog inputs** for further processing.

Digital I/O's

The digital channels created using the CAN-Bus-Assistant appear in the channel list under **Digital channels > Fieldbus: Digital inputs / outputs**.

9.9.2.5.7.1 Settings

If fieldbus channels are selected in the channel list (assuming the device currently allows this), then additional setting options are available.

Parameter	Description		
	<i>long name</i>	<i>short name</i>	<i>column ID</i>
X-axis	X-axis		eXFormatVariable
	<ul style="list-style-type: none"> • <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	Duration		eDuration
	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	Assignment		eAllocation
	The data can additionally be assigned to a Display variable (only for analog Fieldbus inputs).		

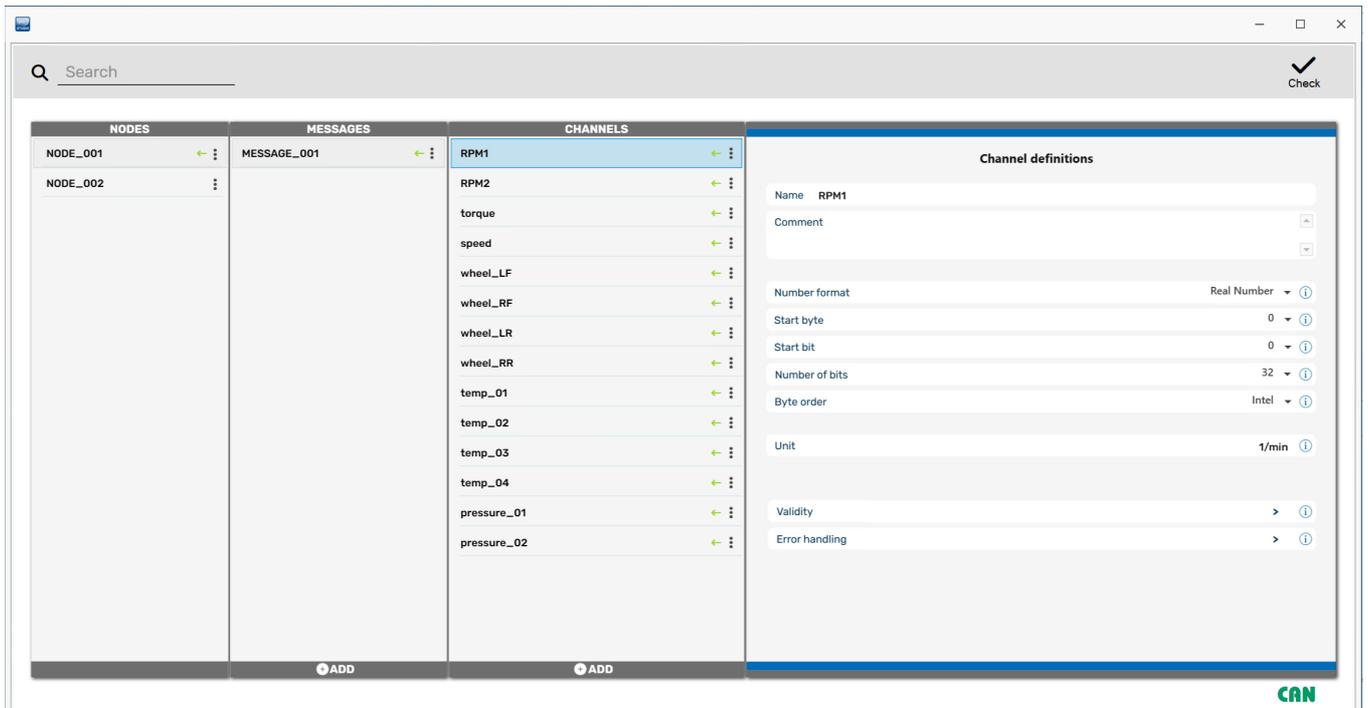
9.9.2.5.7.2 Global channels for logging CAN-Bus messages

These channels are for collecting the data from corresponding sensors which are connected to a CAN-Bus. The data are captured via a special CAN-Bus interface. The necessary definitions can be made using the CAN-Bus Assistant. The IDs and data bytes of CAN-messages are logged in these channels. The channels are expressed in ASCII time stamp format. The measurement duration can be adjusted.

9.9.2.6 CAN-Bus-Assistant (firmware group B)

For the purpose of configuring nodes, messages and channels belonging to the device of the [firmware group B](#) (ARGUS), the "CAN-Bus-Assistant" is used. The assistant for [firmware group A](#) (CRONOS, BUSDAQ, SPARTAN, etc.) is located [here](#).

Action	Page
CAN (CAN)	Assistants



CAN-Editor

Together, the CAN nodes, messages and channels form a CAN configuration. This configuration can be exported as a JSON-file. JSON-files are saved in the ASCII-format. Thus the content of these files can be viewed using a text editor.

In the CAN-Bus-Assistant, there is no OK-button. Any entries are adopted immediately. If you make any invalid entries, a notification will appear in the status bar at the bottom in red script upon making the change. In order to detect any errors, it is recommended that you perform a configuration check from time to time. 

 **Note**

It is possible, however, to exit the Assistant without correcting the changes - however, no CAN channels will be created!

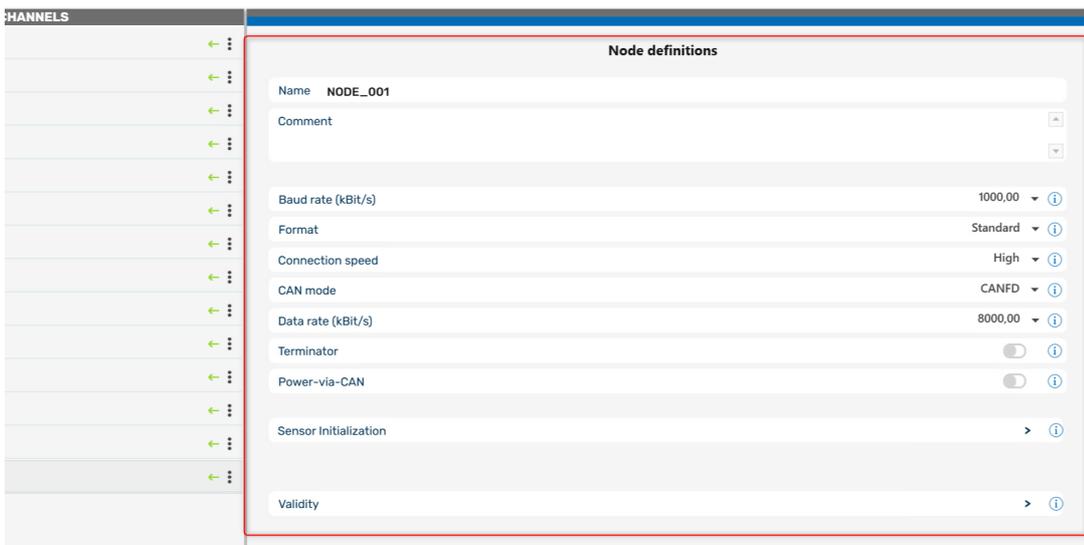
9.9.2.6.1 Editing a CAN configuration

The assistant can be operated both by mouse and by touchscreen. On the right side, the properties are shown of whichever element is selected under Node, Messages or Channels.

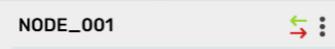
! Notes

Any changes made are applied immediately without the need to click on the *Enter-* or *OK-*button, even if they are incorrect, such as for the channel name. Any error is indicated in the status bar.
There is also no *Cancel-*button to correct unintentional changes!

9.9.2.6.2 Nodes



Definition nodes

Parameter	Description
Name	<p>Unique name by which to distinguish the node, up to 65 characters long. The default is "NODE_" + Index (e.g.: 001) of the node is suggested. If multiple devices are present, the device index is additionally appended. The name must conform to the syntax of an imc FAMOS-variable.</p> <p>Whether a node sends and/or receives messages is indicated by a red/green arrow:</p> 
Comment	Freely defined text describing the node, up to 255 characters.
Baud rate (kbit/s)	The baud rate specifies the interval between individual bits. All modules belonging to the same node must have the same baud rate. The drop-down list "Baud rate" offers a choice of rates from 5kbit/s to 1Mbit/s. The default value is 125kbit/s.
Format	<p>"Standard", "Extended" or "Extended+".</p> <p>The difference between Standard and Extended is that the Standard-format ID length is 11 bits (IDs from 0 .. 2031 are permitted) and in Extended-format 29 bits (IDs from 0 .. 536870911 are permitted). Extended-format allows for greater distances between consecutive IDs. This makes it possible to extend the system without major changes to the configuration.</p>
Connection speed	High Speed is the default and complies with ISO 11898. Low Speed corresponds to ISO 11519 and is applied in the automotive industry among other areas.

Parameter	Description
CAN mode	CAN classic: CAN High Speed as per ISO 11898, CAN Low Speed as per ISO 11519. CAN FD (ISO Standard): in accordance with ISO 11898-1:2015. CAN FD (non-ISO Draft): based on an earlier design from BOSCH.
Data rate (kbit/s) (only with CAN FD):	In <i>CAN FD</i> mode, the input box for the data rate appears. If the controllers of all subscribers in the Bus support CAN FD, then different settings for the Baud rate and data rate can be transmitted. When transmitting at the high data rate, the subscribers having the slower Baud rate are switched to passive status. Caution: In order for the data of a send message to be sent at the data rate set here, the message's option Bitrate switch ⁴²⁵ must be activated.
Terminator	A terminal resistor can be connected to a node via software. If the measurement device is located at the end of the CAN line, this eliminates the need for a Y-cable with an external terminator. See also " Connecting terminators " ³⁷² .
Power-via-CAN	Connectable power supply for imc CANSAS modules. By means of this switch, the power supply for the CAN FD module is activated via the CAN-Bus cable (Pin 1(+) and Pin 5(-)). This voltage is isolated from the CAN-Bus. Caution: The module power supply and its pinout are imc proprietary accessories only appropriate to imc CANSAS modules.

9.9.2.6.2.1 Sensor initialization

Many sensors require a one-time initialization in the form of one or more messages. This initialization is usually necessary only once, upon activation of the sensors.

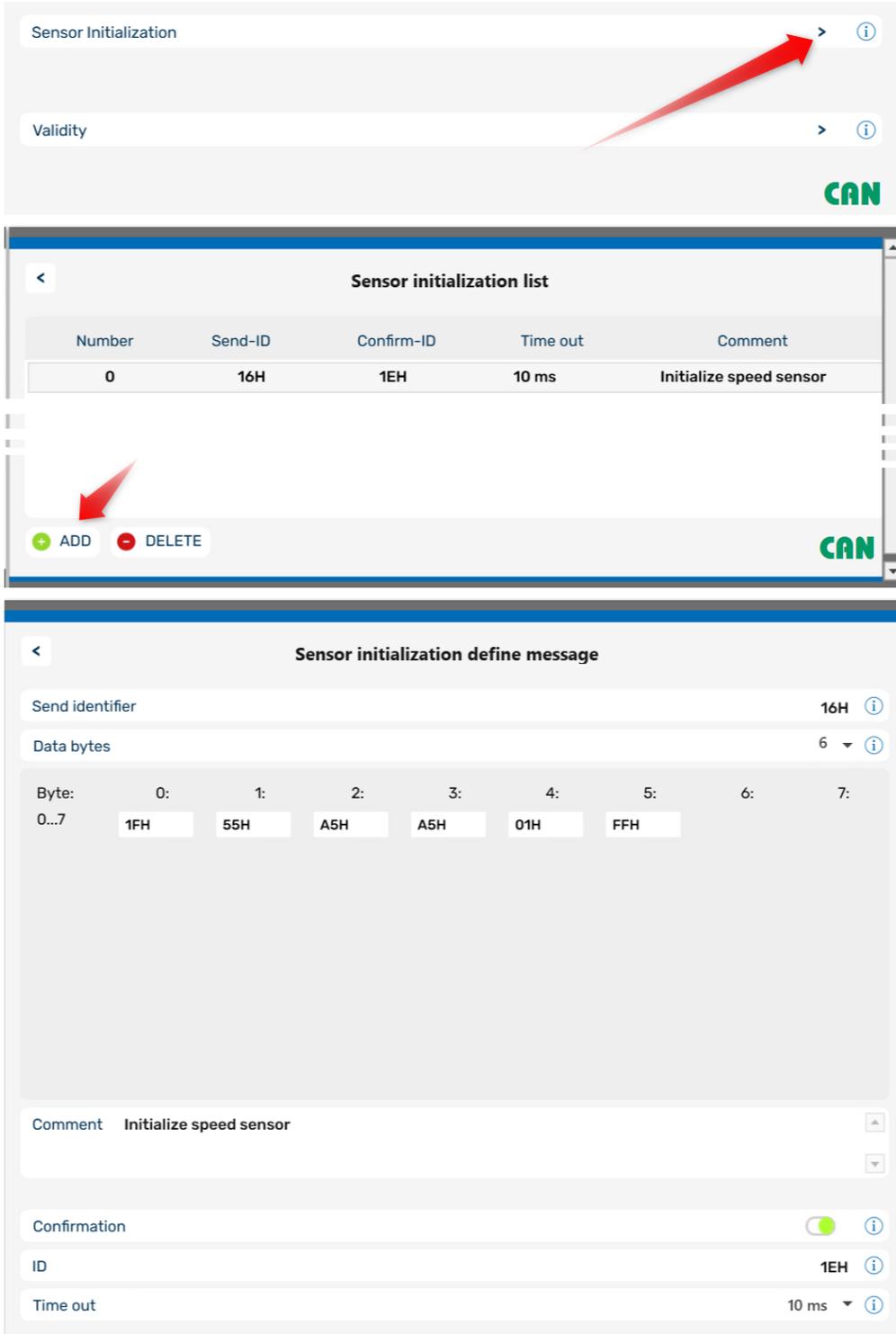
For one single time, the device can output a specified sequence of messages on the CAN-Bus. This initialization sequence is performed once only upon preparing for measurement after configuring the device. If the configuration isn't change, no initialization is performed any more upon next starting.

For devices that start automatically, the initialization is performed in the same way upon at the first start.

The initialization consists of a sequence of messages. After each message, it is optionally possible to wait for a response from the sensor by means of *Time out*. Many sensors acknowledge a parameterization message with an additional message. The device then waits for the sensor's response before sending the next message.

Defining the initialization message

The Sensor Initialization list is accessed from the node's Properties page.



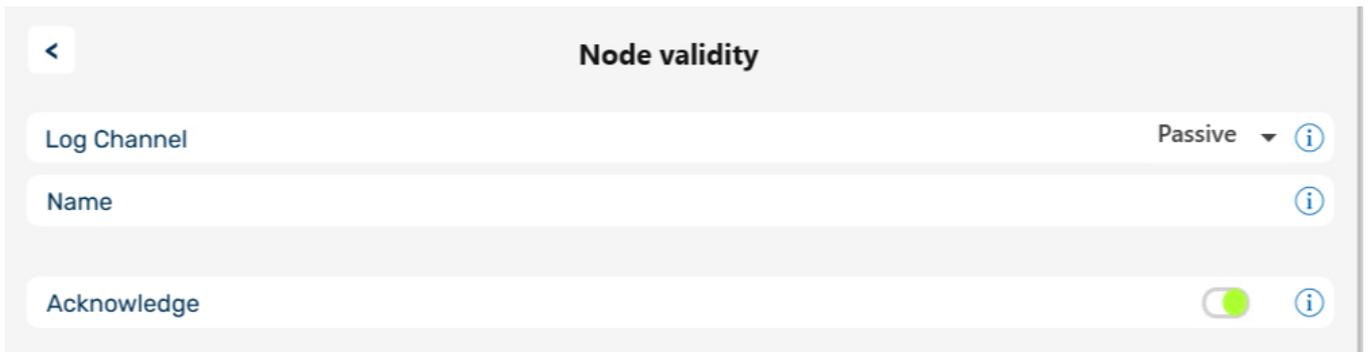
Multiple messages can be created. New messages can be created by clicking on ADD.

For the purpose of initialization, messages are sent to the sensors. These messages can be acknowledged. The list in the dialog shows the entire initialization sequence.

Sensor initialization

Parameter	Description
Send identifier	An identifier for the message with the initialization sequence. Allowed values are from 0 to 2031 or 0H .. 7efH for Standard-format, for Extended-format 0 .. 536870911 or 0H .. 1ffffffH. The ID is always displayed in hex format but you can enter it as a hexadecimal or decimal number. Numbers entered in Hex-format must be designated as such by using H or h (e.g. 2ah)
Data bytes	The number of bytes to be used for the message. Up to 8 bytes on the classic CAN-Bus; up to 64 bytes on CAN FD.
Byte 0 .. 7 (63)	Content of the individual byte which is sent to the sensor. 0 .. 255 or 0H .. ffH are allowed.
Comment	Additional description of the initialization sequence or other information - maximum 200 characters.
Confirmation	Activation of an acknowledgement message.
ID	Identifier of the acknowledgement message. Both Standard- and the Extended-format are possible. This message must be sent by the sensor. The device waits for arrival of such a message before resuming at the next line of the list.
Time out	If after this time (1 ms .. 1000 ms) no acknowledgement is received, the initialization sequence is continued. In this case, the sensor may not be initialized.

9.9.2.6.2.2 Validity



Validity of a node

Log Channel

Parameter	Description
Log Channel	When Log channel "Active", all messages belonging to the node are logged in imc WAVE. In the input field <i>Name</i> you can enter a unique name for the log channel. By default, a name with slot and node number is generated. There is exactly one log channel per node. Message IDs and data bytes are logged.

Acknowledge

Disables the transmission of Acknowledge messages. The CAN controller usually sends this packages to confirm successful received messages.

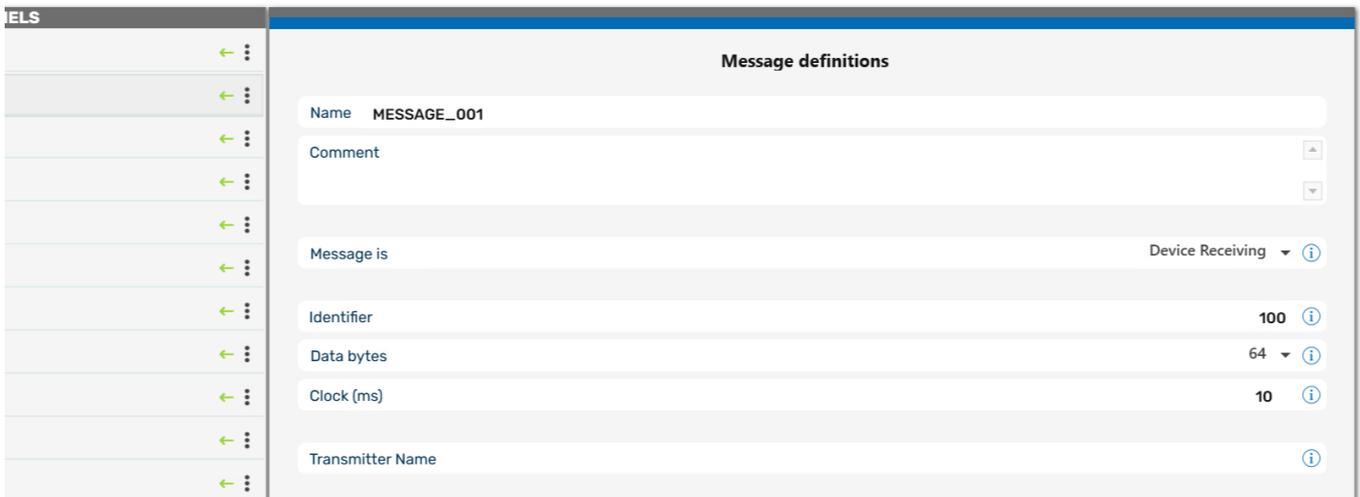
9.9.2.6.2.3 Context menu

Right-click the node name or the three dots to display a shortcut menu with the following options:

Parameter	Description
Import	Restores settings from a JSON file. The available formats are "JSON" and DBC".
Export	Exports the settings of a node, including all messages and channels, as a "JSON" or "DBC" file.
Revert	All changes to the node are discarded and the settings are restored back to their state at the time the assistant was opened.
Clear	Discard settings. All messages with channels are deleted.

9.9.2.6.3 Messages

Receiving a message



Received message

Parameter	Description
Name	A unique name for each message with up to 65 characters. The default name for a new message is Message_" + index (e.g. 001). If several devices are being used, a _j is appended to the name, where j is the device index number. All names must be comply with imc FAMOS variable syntax, just as all other device channel names..
Comment	Freely definable text describing the message and having a maximum of 255 characters.
Message is	Specifies whether the message is to be sent or received. Received messages are indicated by a green arrow. <div style="border: 1px solid gray; padding: 2px; display: inline-block; margin-top: 5px;">MESSAGE_001 ← ⋮</div>
Identifier	Each message on the CAN-Bus contains an identifier (ID) which indicates the origin and meaning of the data. No two messages in a node can have the same ID. The identifier also indicates the priority of the message and, once again, no two messages can have the same priority. When two messages are to be sent at the same time, the message with the lower ID is sent first. Allowed values are: 0 to 2031 (Standard ID format) and 0 to 536870911 (Extended ID format). Identifiers can also be specified as a hexadecimals. Append an "h" or "H" to the hexadecimal, e.g. 2ACH or e4h. Otherwise the number is interpreted as decimal.
Data bytes	Number of bytes in the message, 0 to 8, or 64 if the node has been created in CAN FD mode ^[420] . For instance, if 4 bytes is entered, 4 bytes will be sent per message. It is not possible to send partial bytes in a message but it is also not necessary that every bit in the byte be occupied.
Clock (ms)	Messages are sent via the CAN-Bus at a regular rate. The value for the clock specified here is a set point, not necessarily the actual value. The clock is a property of the sensor. The allowed range is 0.001ms to 100,000ms. The sampling rate of the CAN channel to be created is determined by the clock value specified
Transmitter name	Name of the sender of the message. Only for description.

Sending messages

Message definitions

Name

Comment

Message is Device Sending

Identifier 100

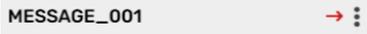
Data bytes 64

Bitrate switch

Clock (ms) 10

Receiver Name

Sendende Botschaft

Parameter	Description
Name, Comment, Identifier, Data bytes, Clock	As for messages to be received. ⁴²⁴
Bitrate switch	Data bytes for CAN FD: Messages can be sent at the data rate set for FD ³⁸³ . For this purpose, the option Bit Rate Switch must be activated, which appears at the CAN FD node.
Message is	Messages to be sent are indicated by a red arrow. 
Clock	Messages are sent via the CAN-Bus at a regular rate. The <i>Clock</i> -value is the interval between transmission of messages. The interval must be at least 1ms. The value set here serves in imc Online FAMOS as the default setting for the clock rate for cyclic transmission in the Formula-Assistant..
Receiver Name	The receiver name describes the receiving sensor. This name is not used for any other purposes - therefore it need not be unique.

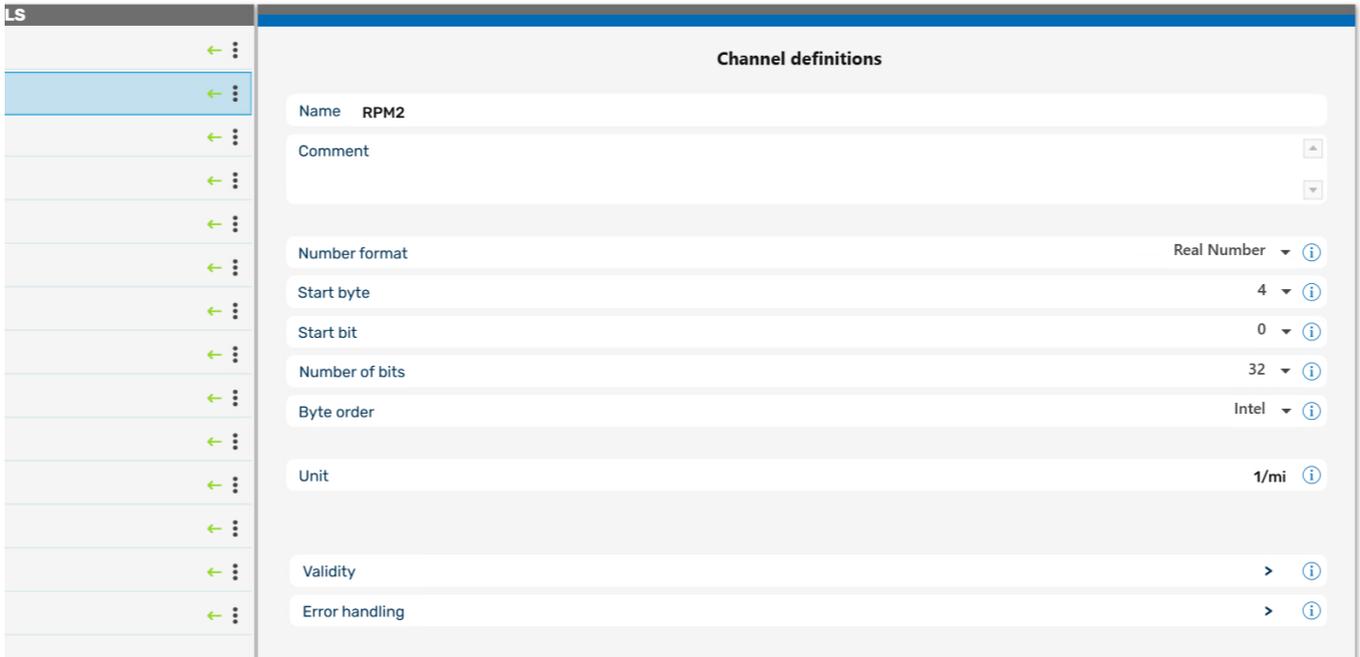
Context menu of Messages

Right-clicking on the name of the message or on the three dots accesses a way to delete the message.

9.9.2.6.4 Channels

Analog CAN-channels

Analog signals are digitized by the CAN-sensors with a specific number of bits. The numerical format can be set to *signed integer*, *unsigned integer* or *real number*.



Definition of a analog channel

Parameter	Description
Name	Unique name with up to 65 characters. Default is "CHANNEL_" + channel's index. The name must conform to the syntax of a imc FAMOS-variable.
Comment	Freely defined text describing the message; up to 255 characters
Numerical format	Data can be sent and received as signed or unsigned integers, real numbers or in digital (1Bit) format.

Note

Information about the numerical format

Some manufacturers may not indicate clearly in their CAN tables which number format they use.

For example, the table may indicate that Bytes 1 and 2 represent the guidance angle and that this value is to be multiplied with 0.0639 to arrive at the actual value in degrees. If no other information has been given, there is no definite way of determining the number format other than trial and error. Our tech support cannot help you in such cases! Sometimes however, there are implicit clues as to what the format is.

Examples:

- If the range minimum is 0 and the maximum is ffff hex, then unsigned integers are being used because they always start at zero.
- If the range is given as 7fff = -700°C and 8000 hex = +700°C, then signed integers in two's complement form are being used. The zero (0 hex) lies directly in the middle between positive and negative integers.

The most certain method of determining the number format is to run the sensor through its entire range.

If the correct number format has been selected, the curve that you measure will be continuous. Otherwise, the upper and lower halves are exchanged, which appears as a discontinuity. A discontinuous curve is therefore an indication of the wrong integral format being used.

Parameter	Description
Start byte	Specifies in which byte of the message the number begins. Byte 0 is the first transmitted byte of the CAN-message.
Start bit	The start bit specifies where the number starts within the start byte.
Number of bits	This is the number of bits used in the channel. The remaining bits continue in the direction of the increasing byte indices. For signed and unsigned integers, 1 to 32 bits are allowed. Real numbers can range from 32 (floating point) to 64 (double).
Byte order	Bit order used by the CAN-module: either Intel-format or Motorola-format

Note

Byte Order | Format

Intel format: The start bit is the LSB. The **LSB** has a **lower byte index** than the MSB.

Example: start byte = 5, start bit = 3, bits = 18

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 5	X	X	X	X	LSB			
Byte 6	X	X	X	X	X	X	X	X
Byte 7				MSB	X	X	X	X

Motorola format: The start bit is the MSB. The **LSB** has a **higher byte index** than the MSB.

Example: start byte = 5, start bit = 3, bits = 18

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 5					MSB	X	X	X
Byte 6	X	X	X	X	X	X	X	X
Byte 7	X	X	X	X	X	LSB		

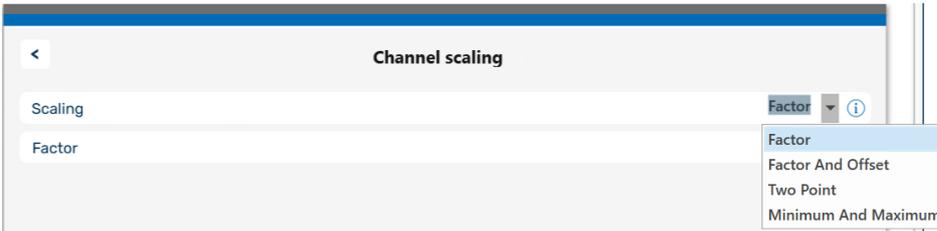
Note on recognizing Intel or Motorola byte order

- **The CAN tables from manufacturers do not always indicate whether Intel or Motorola byte order is used.** It is sometimes possible to determine the byte order indirectly. Within one message typically only one of the variations will be used.
- **Example:** Two byte data packet, message Byte 3 is LOW and message Byte 4 is HIGH Intel byte order because HIGH byte contains MSB.
- **Example:** Bits 0 to 14 contain data and Bit 0 is LSB
Data is contained in Bytes 0 and 1 but only the lowest seven bits in Byte 1 are occupied. Therefore, LSB is in the byte with the smaller index which means that Intel format is used.

Parameter	Description
Unit	The physical units of the channel, maximum 24 characters. The units are used when displaying the channel, e.g. on the Y-axis in a curve window. Note that units should be specified without prefixes. Examples of units that you can use are: m, s, kg, N, m/s. If you need to use prefixes (e.g. n-, μ-, M-) or non-SI units (e.g. inches, pounds), always place these between quotation marks (e.g. "inches").
Scaling	Here is determined how to convert the physical value to a CAN bus integer. There are different choices available, to determine the conversion.

Note

Notes for scaling



The Scaling options

Y = physical measurement value

X = CAN bus integer value

Scaling	Description
Factor, $Y=Factor*X$	<p>Receiving: for received signals (X) the following applies: $Y = Factor * X$</p> <p>Sending: for physical values (Y) to be sent, the integer value is calculated as follows: $X = Y / Factor$</p>
Factor and Offset, $Y = Factor * X + Offset$	<p>Receiving: for received signals (X) the following applies: $Y = Factor * X + Offset$</p> <p>Sending: for physical values (Y) to be sent, the integer value is calculated as follows: $X = (Y - Offset)/Factor$</p>
Two-point specification, (X1, Y1), (X2, Y2)	<p>X1: x-coordinate of 1st point Y1: y-coordinate of 1st point</p> <p>X2: x-coordinate of 2nd point Y2: y-coordinate of 2nd point</p>
Minimum and Maximum	The lower limit of the specified number format with the specified number of bits corresponds to this minimum (e.g. signed integers, 8 bit, range: 0 to 255, minimum corresponds to 0, maximum corresponds to 255)

Context menu of Channels

By right clicking on the channel name or the three dots, you can access to the command for deleting the channel.

9.9.2.6.4.1 Validity

Parameter	Description
Channel state	Active by default. The channel can be set to passive. The settings remain intact, but the channel is not set up in the data logger and does not demand any resources.

9.9.2.6.4.2 Error handling

The dialog illustrated below appears if the system is set to *receive* CAN-Bus messages.

If an invalid measurement value occurs or if no more messages are received by the device after a certain amount of time, *Error handling* can be used to notify the user. When *Error handling* is activated, the error handling option *Invalid measurement value* and/or *Time out* are displayed.

Invalid measurement value

The comparison value (Channel measurement value) for an analog channel is expected as a real number with physical units.

Time out

Define a time out limit here. This is the maximum time which the sensor has to send the message. Many sensors send messages at regular intervals - this option is often useful for monitoring them.

Error Handling

The available options are

- *Ignore Channel with this message*
- *Last channel value*
- *Replacement Value* - Value indicating an error, e.g. -999

The error handling functions slightly differently, depending whether the channel is continuously sampled or whether time-stamping.

Error correction	Effect with continuous sampling	Effect with time stamping
Ignore channel with this message	Same as <i>Last channel value</i>	Es wird kein Sample erzeugt.
Last channel value	The last valid channel value is generated each sampling interval until a valid sample arrives.	<i>Time out error:</i> A new sample is generated using the last valid channel value and is stamped with the time-out limit. <i>Invalid measurement value:</i> a sample is generated using the last valid channel value and is stamped with the message time stamp.
Replacement value	The replacement value is generated each sampling interval until a valid sample arrives.	Like <i>Last value</i> , but with <i>replacement value</i> .
No error correction used	Same as <i>Last channel value</i>	Same as <i>Ignore channel with this message</i>

 **Note**

- **Input range of replacement values for error handling**
Certain replacement values may be outside of the input range for the CAN channel.
- Channels with 16 bits or less can receive replacement values having a 16-bit input range. Example: For a CAN channel with 8 bits, a replacement value of up to 16 bits is permitted.
- Channels with more than 16 bits and less than or equal to 32 bits can be given replacement values having a 32-bit input range. Example: For a CAN channel with 24 bits, a replacement value of up to 32 bits is permitted.
- Consequently, the following exceptions exist: Channels with exactly 16 bits or 32 bit-channels only permit replacement values having the same input range.

9.9.2.6.5 imc WAVE - Configuration / Processing of fieldbus channels

Analog inputs

The channels created using the CAN-Bus-Assistant appear in the channel list under **Analog channels > Fieldbus: Analog inputs** for further processing.

Digital I/O's

The digital channels created using the CAN-Bus-Assistant appear in the channel list under **Digital channels > Fieldbus: Digital inputs / outputs**.

9.9.2.6.5.1 Settings

If fieldbus channels are selected in the channel list (assuming the device currently allows this), then additional setting options are available.

Parameter	Description		
	<i>long name</i>	<i>short name</i>	<i>column ID</i>
X-axis	<i>X-axis</i>		<i>eXFormatVariable</i>
	<ul style="list-style-type: none"> • <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).		

9.9.2.7 Application examples

9.9.2.7.1 Sending CAN messages

For this example, an imc measurement device with a CAN-interface is necessary.

9.9.2.7.1.1 Balancing a bridge module via CAN

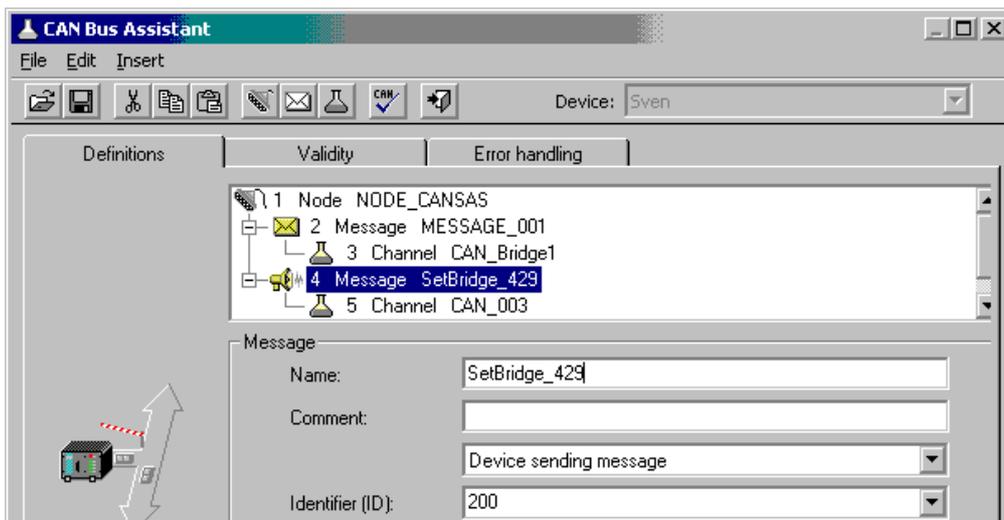
A CANSAS bridge module is used to take a strain gauge measurement. The bridge is to be remotely balanced in its rest state.

Some imc CANSAS-modules are able to receive CAN-messages for balancing and shunt calibration. The devices must previously be prepared for this by means of the CANSAS configuration software, refer to the CANSAS manual for details. The example assumes that balancing takes place whenever the number 10Hex is sent in a message with the identifier 200.

CAN-Bus Assistant settings:

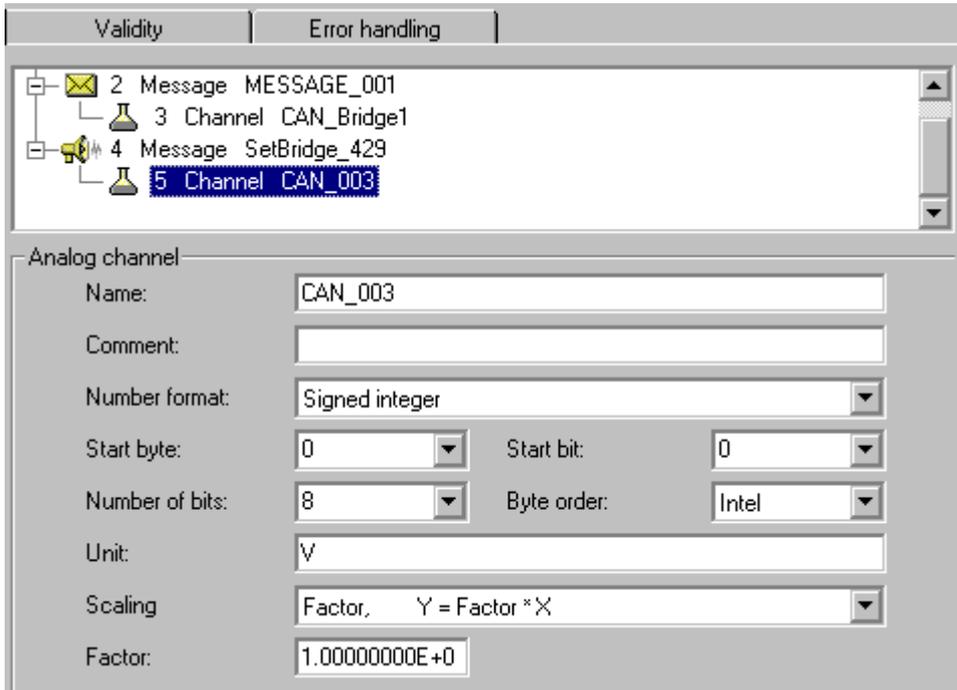
As message "SetBridge_429" is created. The name is arbitrary, the suffix _429 is an aid in locating the module by the serial number. The message must be given the ID which was previously set in imc CANSAS-module for reception purposes, the default is 200.

The channel CAN_Bridge1 measures the strain gauge signal, and thus we will be able to observe how the offset disappears if we balance the bridge while the measurement is running.



Create a message for sending the balance command

The channel "CAN_003" is also created. We intend to send the number 10 Hex via this channel and reserve 8 bits for it.



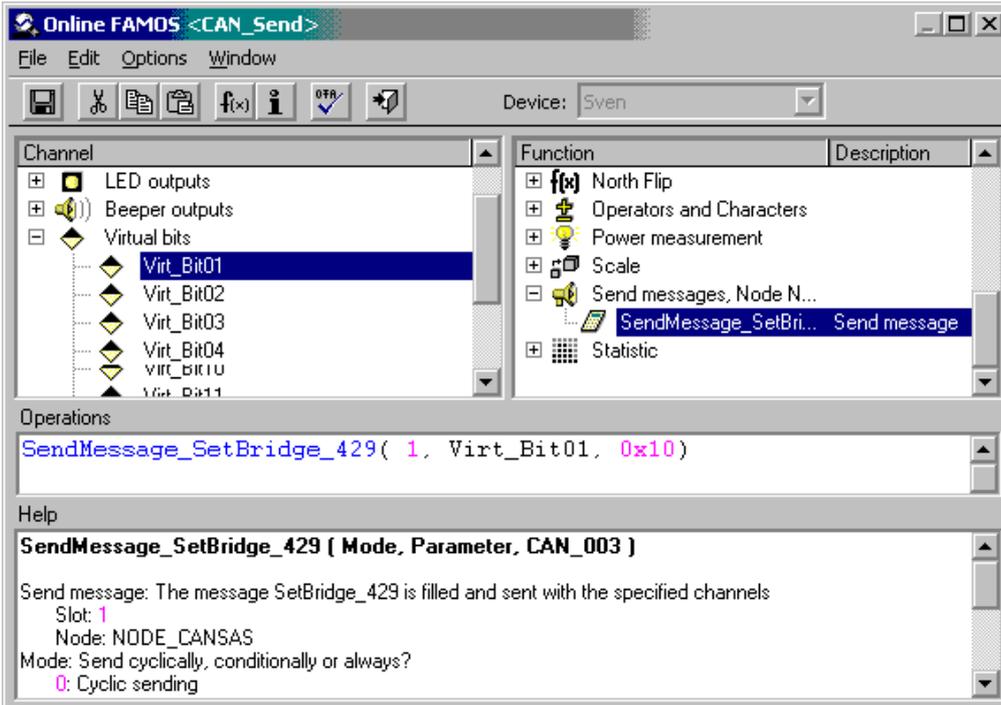
Channel for the balance command

The balancing process is to be carried out on command. For this purpose we use a virtual bit which we rename in the configuration. Later, the DIODAC dialog will provide a convenient way to set this bit. Of course, it would also be possible to use a digital input bit which is set by means of a button.

imc Online FAMOS:

Once a message has been created in the CAN-Bus-Assistant (and only then!), a new entry appears in imc Online FAMOS, *Send messages, Nodes N...*. There you will find a function which starts with *SendMessage_* and bears the name of your message, in this case *SendMessage_SetBridge_429*. This function is able to send a message in three different ways: cyclically, condition-dependently, and, if you are working with control commands, permanently.

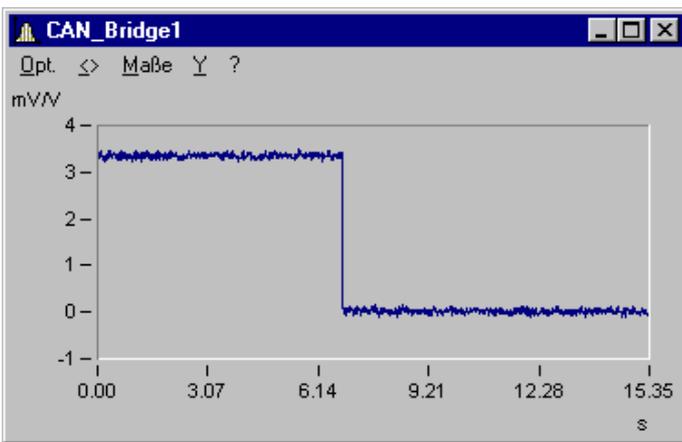
We select the operating type *With condition* and select our virtual bit *Virt_Bit01* as the condition. The number we send is 10 Hex, which causes imc CANSAS-module to be balanced.



SendMessage-command in imc Online FAMOS

We create a Panel page showing the strain gauge and a push button linked with the virtual bit for the balancing. Start the measurement. At first the bridge is unbalanced. Set the virtual balance bit with the push button.

The bridge is balanced:



The bridge is balanced and shows 0 mV/V

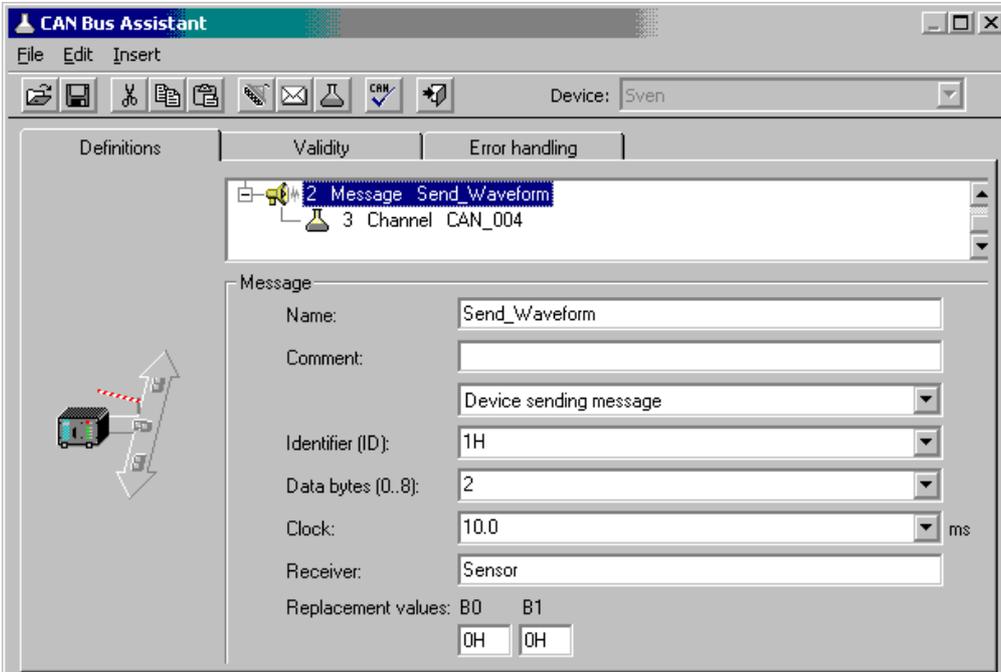
9.9.2.7.1.2 Initiating a shunt calibration for a bridge module via CAN

Proceed as for [Balancing a bridge module via CAN](#)⁴³³. Simply substitute the HEX-code to send with 20Hex.

9.9.2.7.1.3 Outputting a waveform to a DAC module

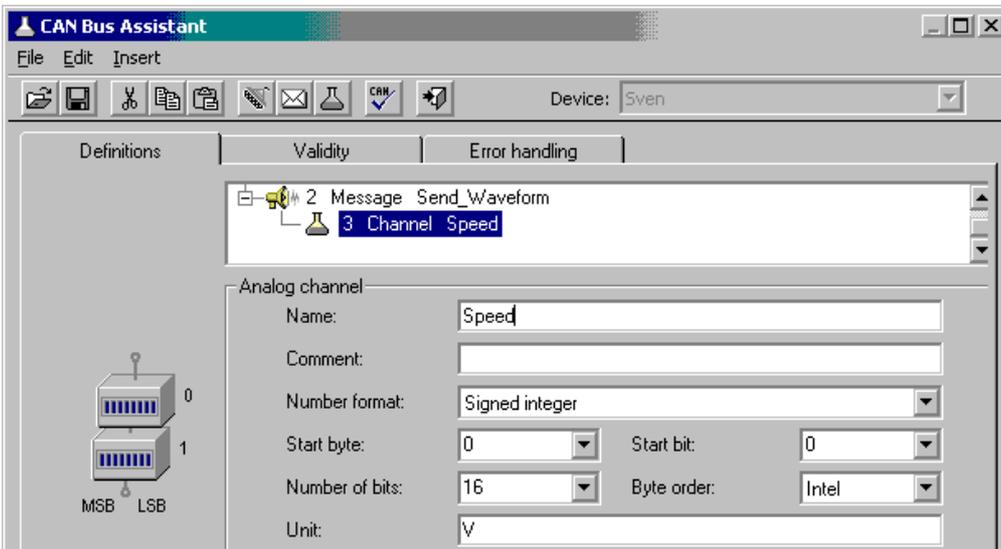
Cyclical output of values is another example. In this example, a ramp is generated. The data set to be outputted is loaded in the form of a characteristic curve. The `Charact` function is used to convert the ramp to the output values. This is sent as a message to a DAC module.

CAN-Bus-Assistant settings: Message



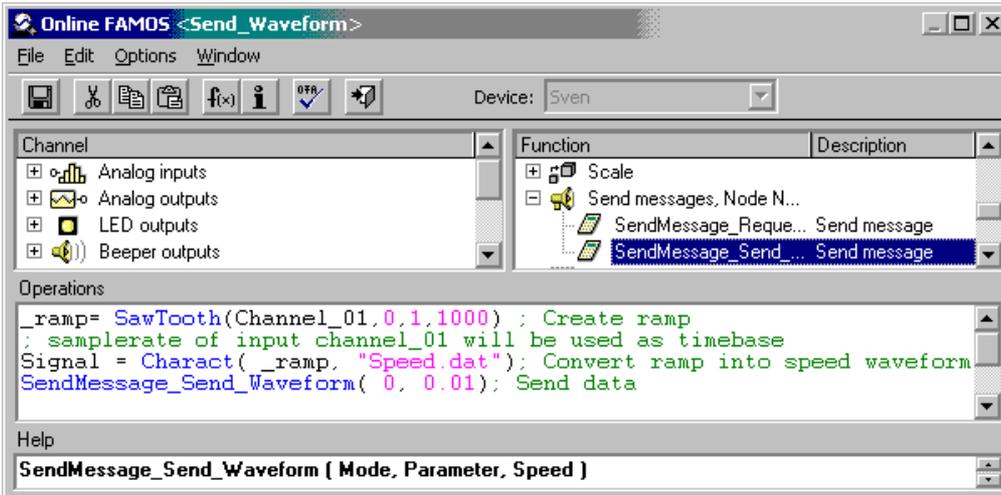
Create a send message with a channel

Channel:



The send channel is configured to the format of the data, that will be transferred by imc Online FAMOS afterwards

The text in imc Online FAMOS then looks like this:



Transfer the data in imc Online FAMOS

Note

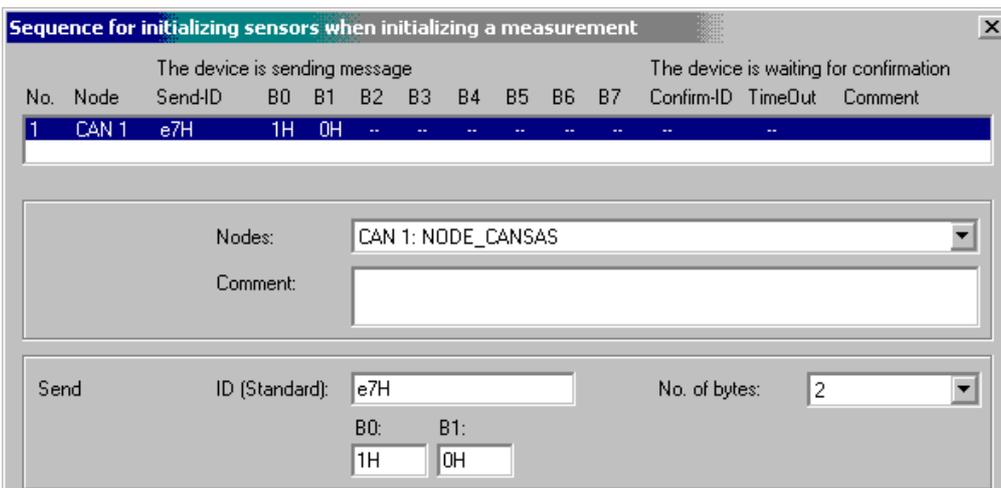
- The ramp must have the same length and same data rate as the data set to be outputted.
- In this example, an underline "_" was prefixed to the ramp. This causes the ramp to be retained in the device as an local auxiliary variable and cannot be found by the curve window.
- The data set to output must be in the folder which is specified under Options-Folders.
- The data set to be outputted must be expressed in imc FAMOS format. You can create the data set using imc FAMOS or simply copy data from an already recorded measurement.

9.9.2.7.2 Initializing a sensor and requesting data

Many sensors don't operate cyclically and must be "started". For this purpose, they need information about the initialization ID as well as on the numbers which are sent for the initialization. Both are noted in the description of your sensor.

In order to create an initialization sequence, open the CAN-Bus-Assistant. Once you are there, select the menu item Edit > [Sensor initialization](#) ³⁹⁴.

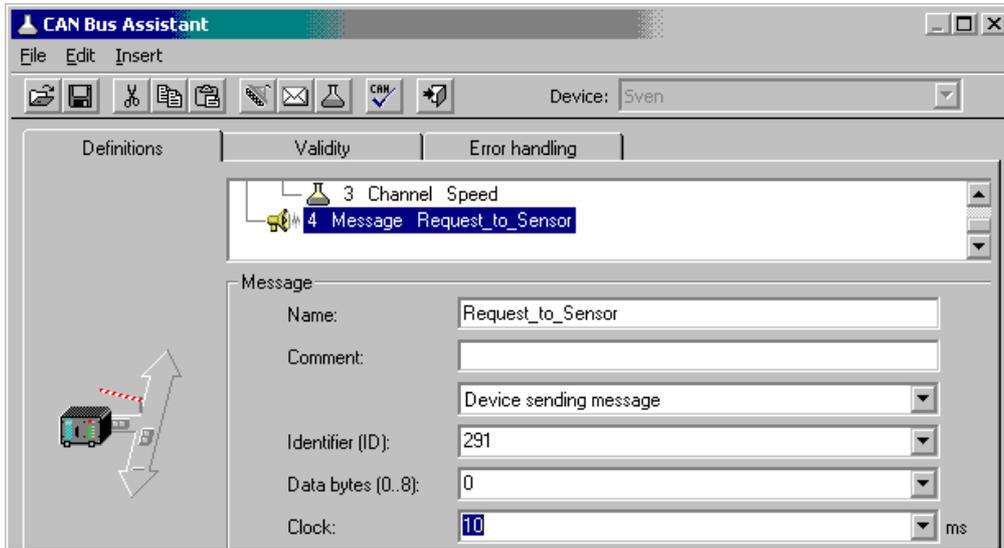
A possible sensor initialization code to enter could be as follows:



Example for initializing a sensor

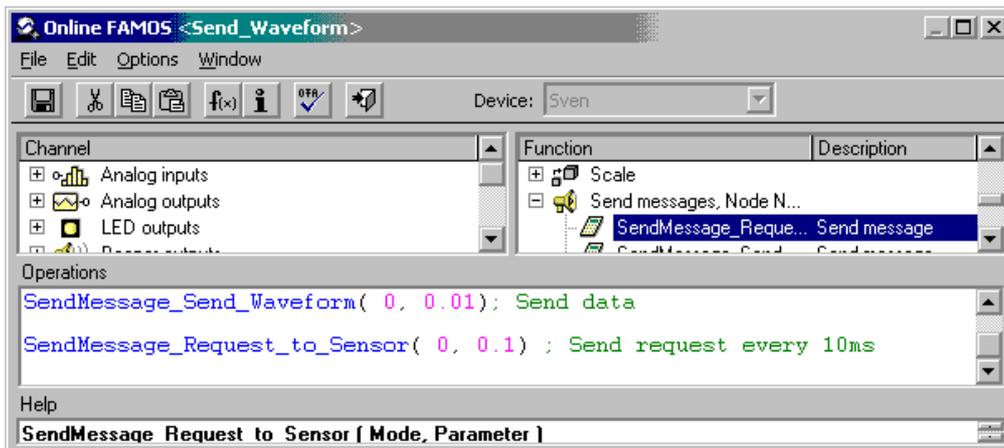
If your sensor requires a request for data, you will need to send a message to it. The ID and contents of the message are stated in the sensor's documentation.

In the CAN-Bus-Assistant, you must create a message and then put it in sending mode:

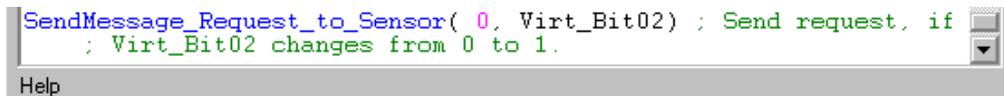


Create a send message

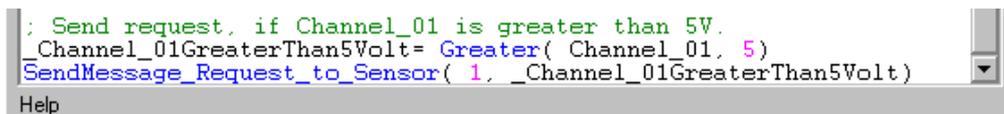
In imc Online FAMOS, select the SendMessage function and specify whether to make the data request cyclical or to make it depend on a state transition:



Example of cyclical sending at a 10ms rate



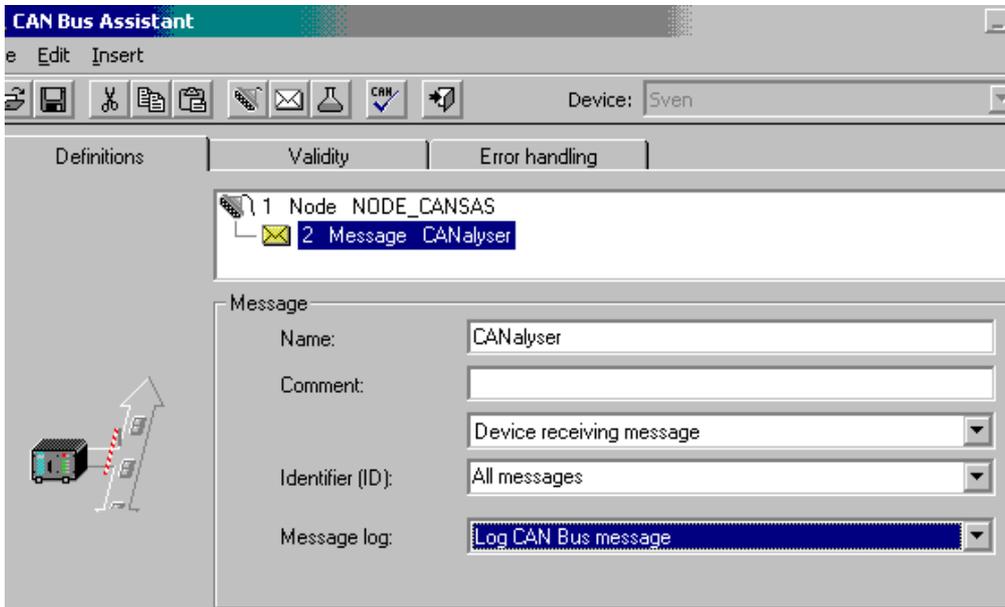
Example of an event-dependent request



Example of a request in response to a virtual bit, for instance by means of the DIODAC dialog.

9.9.2.7.3 Setting imc WAVE as a CAN analyzer

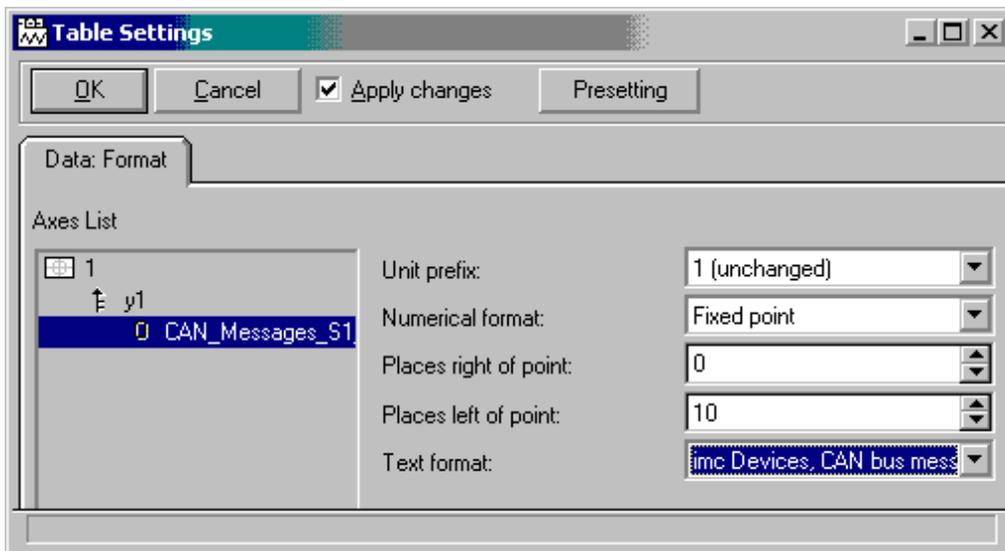
It is possible to make a comprehensive log of all messages. For this purpose a message must be created in the CAN-Bus-Assistant which receives all the IDs. Furthermore you must set *Log CAN-Bus message* under Message log:



Create a log message

At this point, a channel is created whose tree-diagram entry appears together with the Fieldbus inputs. Display this channel in a curve window in tabular form. Select the menu item *Options Display Absolute Date/Time*.

Right-click the mouse in the table and select *Table...* Select as the text format *imc DEVICES, CAN-messages*.



Date, Time	CAN_Messages_S1_K1
19.06.2003 13:14:54.3063	0064: 73 4d 73 4d 73 4d
19.06.2003 13:14:54.5063	0064: 73 4d 73 4d 73 4d
19.06.2003 13:14:54.7063	0064: 73 4d 73 4d 73 4d
19.06.2003 13:14:54.9063	0064: 73 4d 73 4d 73 4d
19.06.2003 13:14:55.1063	0064: 73 4d 73 4d 73 4d
19.06.2003 13:14:55.3063	0064: 73 4d 73 4d 73 4d
19.06.2003 13:14:55.5063	0064: 73 4d 73 4d 73 4d
19.06.2003 13:14:55.7063	0064: 73 4d 73 4d 73 4d
19.06.2003 13:14:55.9063	0064: 73 4d 73 4d 73 4d

Display of CAN messages as a CAN analyzer

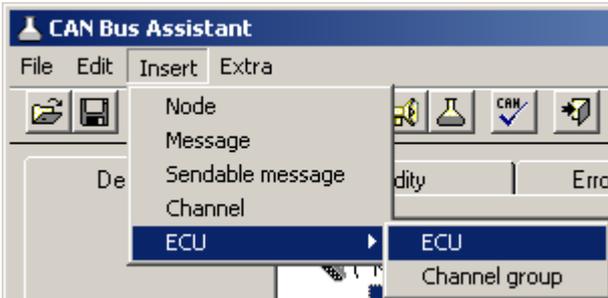
9.9.2.8 Control units in the CAN-bus assistant (ECU)

Using the CAN-Bus Assistant, the control unit properties and the values to acquire are selected. This can also be accomplished by importing a file describing the control unit or by configuring step by step.

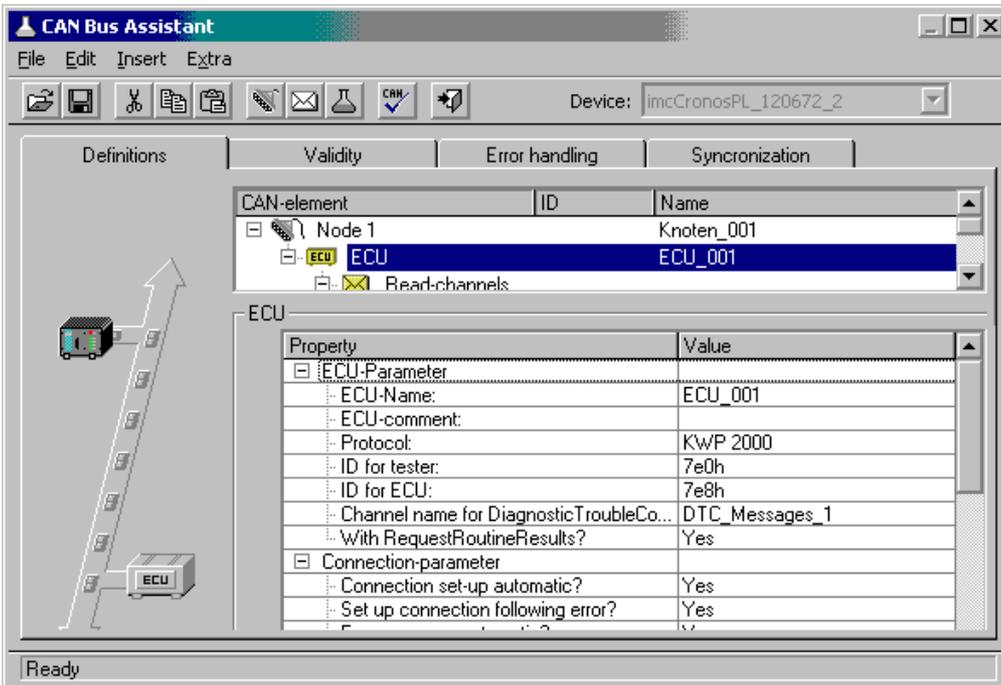
Requirements see [here](#) .

9.9.2.8.1 Configuring a new control unit

Start by assigning a new ECU to a CAN-node. Next, the ECU's properties must be configured in accordance with the control unit and protocol used. To open the control unit properties select menu [Insert - ECU](#)^[412].

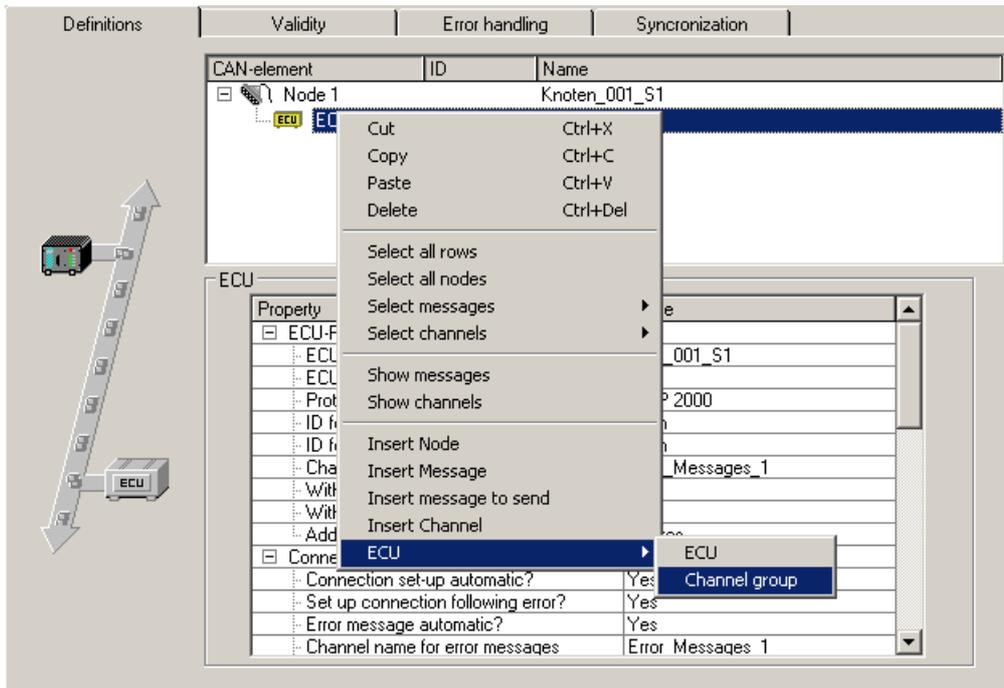


Open control unit properties



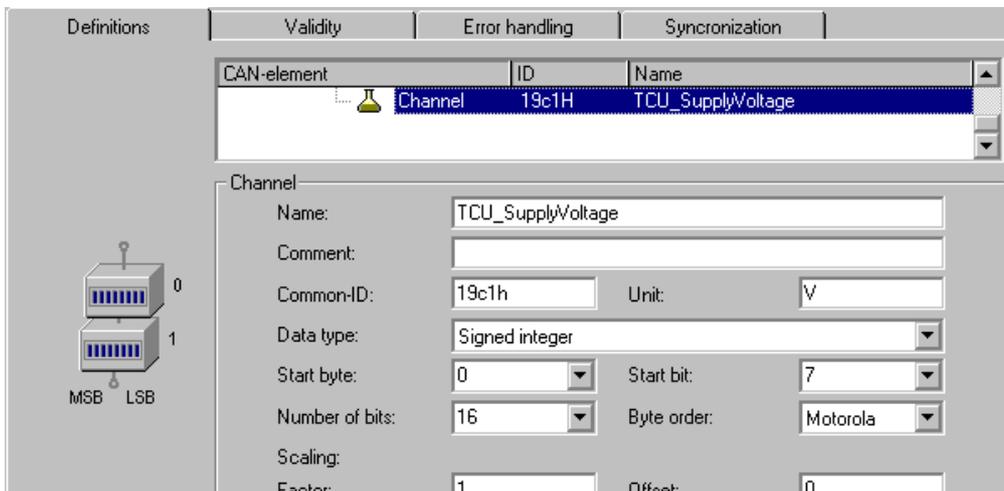
ECU properties

Channels representing the values to be captured can be assigned to the new ECU. For this purpose, a new channel group is created.



Create a channel group

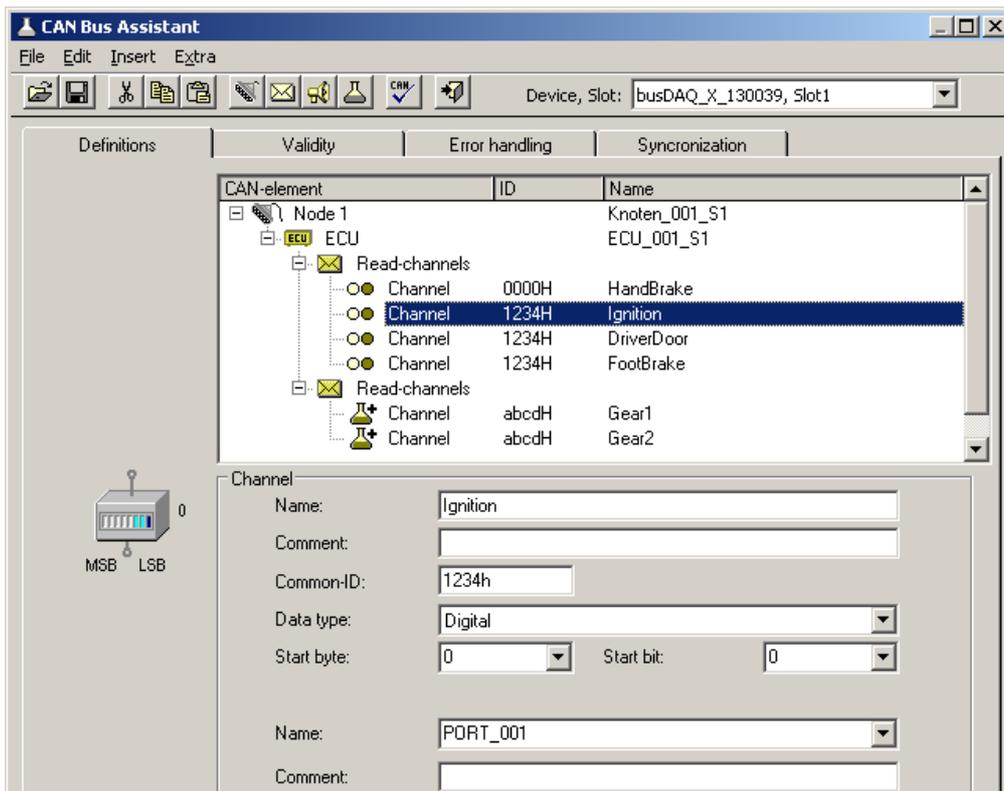
A new channel is automatically created for a new channel group.



ECU channel

An ECU's channels are arranged in channel groups. A channel group determines how the ECU's values are addressed. If the ECU's values are to be addressed using varying addressing modes, then use multiple channel groups.

It is also possible to use channel groups for one's own organization purposes. If one of the ECU's addresses returns multiple, combined values, for example multiple individual bits each with a different interpretation, then these can be gathered together into one channel group.



Combining several bits

The object (channel) assigned to an ECU-address normally has a fixed size, in bytes. In order for such an object's values to be captured, the right quantity must be used when querying it. To do this, define all of an object's values. Configure any extra channels as passive. Alternatively, define at least one channel in the object's highest-value byte and configure it as passive if no capture is desired.

If the protocol supported by the ECU used allows it, it is also possible to create channel groups for the purposes of importing and/or exporting channels and for starting routines.

Create a separate channel group for each routine, in which the routine's parameters are to be defined.

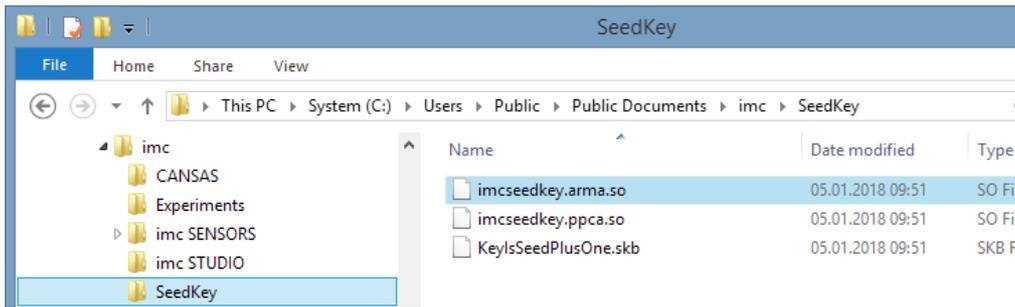
The channels then appear in the Channel-Table.

9.9.2.8.1.1 Seed & key

Access to certain diagnostic services can be protected using the **See-and-Key method**. This is done either via a **fixed key** or a **random seed value**. The algorithm required to generate the random seed value is stored as a file in the device. For each algorithm, a file must be created according to the following scheme:

- "name.architecture.so" (all lower case)
- **name**: freely defined by user
- **architecture**: designates the platform on which the file can be executed
- "arma" for all platforms to date (devices as of [group A4](#)¹⁴⁶ and higher)
- **so**: under Linux, typical file extension for dynamically loadable libraries.
- **.skb**: Alternative format of Vector-CANape.

This file must be located in the directory "C:\Users\Public\Documents\imc\SeedKey".

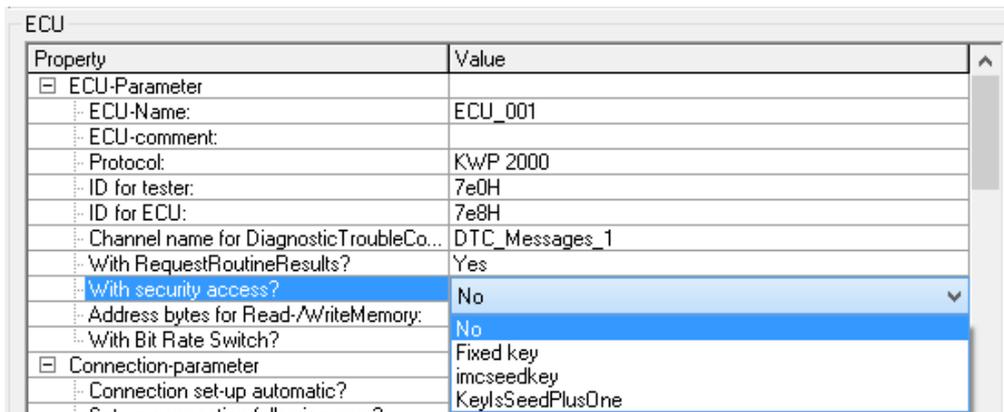


Directory "SeedKey" for ".so" and ".skb" files

Settings

The following parameters must be set in the settings of the ECU message, whose protocol is supported by the seed-and-key procedure:

- The "With security access?" property offers the options "No", "Fixed key" and all file names which are available in the SeedKey directory. If a file is selected, it is transferred within the configuration during preparation.



Settings for "With security access?"

- Selection "Fixed key": If "With security access?" is set to Fixed Key, LogInKey must be set to the key for the ECU. Only one fixed seed/key pair is supported.
- Selection of an encryption file: If there are .so or .skb (KeylsSeedPlusOne) files in the [SeedKey directory](#) ⁴⁴⁴, they are also available at "With security access?"

9.9.2.8.2 Control units in imc Online FAMOS

The use of special functions for controlling control units is only possible with **imc Online FAMOS with Control Commands**.

In order to be able to evaluate return values from commands for the control unit, the special control command `OnECUCmdReturn_*` must be used (* stands for the control unit's designation (e.g. ECU_01)).

OnECUCmdReturn_* (Return, ECUCmd, CmdID)

Result: Return value

The return value is composed of two parts.

$$kt * 256 + k$$

	Description
kt = 0	Communication error k = 0: no error, command successfully issued and acknowledged. k = 1: timeout, the control device did not respond in time. k = 2: sequence error, the ECU responses were not received in the correct sequence. k = 3: command buffer overflow. Only twenty commands (function calls) per ECU can be buffered.
kt = 1	protocol error The ECU has responded to the command with the error code k.
kt = 2	Return status The control unit has announced the end of a function which could not be executed successfully with the return status k.

ECUCmd: ID of the command

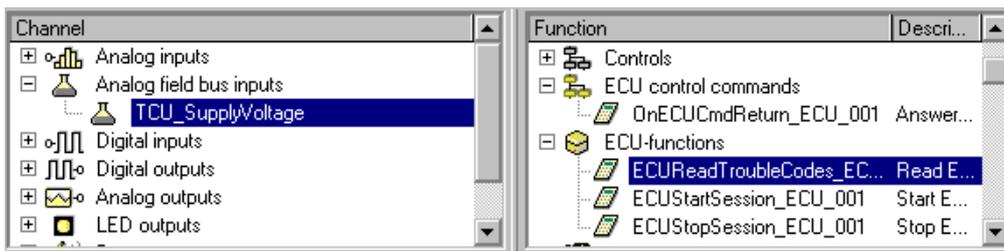
The command's ID consists of three parts.

$$b * 64 * 1024 + i * 256 + c$$

	Description
b	the bus node to which the control device is assigned b = 0: the first node in the CAN-Bus Assistant b = 1: the second node
i	index of the ECU i = 0: the first ECU assigned to the bus node
c	the protocol-specific command code

CmdID: Identifier or address of the command

In addition, there are special functions for the control units:



Control units in imc Online FAMOS

```

; Executed during measurement
; By setting the virtual bit Virt_Bit01,
; a function with the ID= 0x68 in the control unit is started.
OnTriggerMeasure(BaseTrigger)
  if Virt_Bit01 = 1
    Virt_Bit01 = 0
    ECURoutine_ECU_001( 0x68, 9, 1)
  End
End
; Execute when receiving an answer
OnECUCmdReturn_ECU_001( OECR_Return, OECR_ECUCmd, OECR_ProcCmd )
; Checking the answer of the function with ID = 0x68
  If OECR_Return = 0 AND OECR_ProcCmd = 0x68
    ECUReadTroubleCodes_ECU_001( )
  End
End

```

9.9.2.8.3 KWP2000 (on CAN)

For this protocol, it is generally necessary to configure:

- *ID for tester*: Messages are sent to the ECU with this Identifier.
- *ID for ECU*: Response messages from the ECU having this Identifier are expected.
- Either values, or starting of the desired functions.
- Seed-and-Key settings, see [here](#)⁴⁴³.
- *With RequestRoutineResults?*: If the system is to wait for the results of a routing of long duration, having a particular protocol, enter "Yes".
- *With Bit Rate Switch?*: With [CAN FD](#)³⁸³ all send messages can be send with higher baud rate.

Additional parameters

- *Channel name for DiagnosticTroubleCodes*: Unique name for the channel, to which the requested DTCs are written.
- *StartDiagnosticSession*: Here the user enters the byte sequence for a diagnostics session, which can be understood by the ECU. A diagnostics session must be selected which meets the desired requirements. This means exporting of values and starting the desired functions.
- *Addressbytes for Read-/WriteMemory*: The default value for KWP2000 is the address made of three bytes.
- *Tester present*: How to respond to the message notifying the ECU that test personnel is present.
- *don't send*: the message is not sent.
- *send if needed*: only send if no other message has been sent to the ECU for the period of "Tester present cycle".
- *always send*: The message is always set at the rate of the "Tester present cycle", without regard to other messages.
- *send at the start and always*: At the beginning there is no delay by the period "Tester present cycle"; instead the message is sent immediately.
- *Standard-timeout*: Time span in seconds within which the ECU must respond.

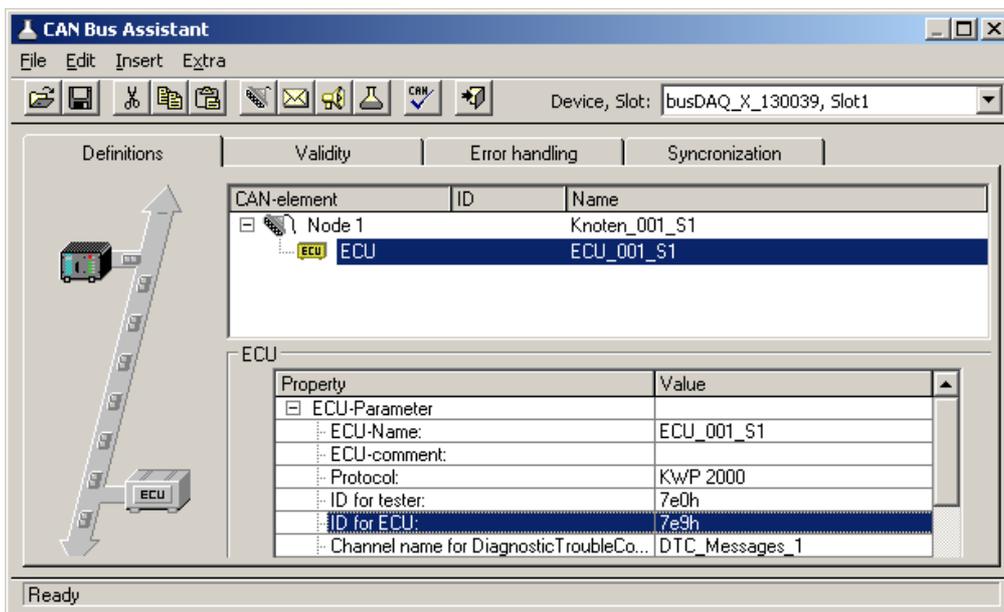
- *Expanded timeout*: Time span in seconds within which the ECU must conclude any routines started, which could not be concluded within the Standard-Timeout period.
- *Tester present cycle*: Cycle at which the measurement device sends a life-sign message to the ECU.
- *StartDiagnosticSession*: The second byte determines the "Diagnostic Session". This property is specific to the ECU.
- *LogInKey*: For cases where "With security access?" was set to Yes, the key for the Seed/Key procedure is entered here.
- *ReadDiagnosticTroubleCodesByStatus*: The second byte is the status queried. The third and fourth bytes are the High Byte and Low Byte of the queried DTCs group.
- *ClearDiagnosticInformation, ReadDiagnosticTroubleCodes, ReadStatusOfDiagnosticTroubleCodes*: The second and third bytes are the High Byte and Low Byte of the DTCs group.
- *P3 time out*

Special imc Online FAMOS functions for ECU's:

Function	Description
ECUStartSession_*	Performs the service StartDiagnosticSession.
ECUStopSession_*	Performs the service StopDiagnosticSession.
ECUReadTroubleCodes_*	The list of DTCs is imported and entered in the channel for DTCs (e.g. DTC_Messages_1)
ECUSend_*	Exports to ECU objects or starts routines

Sample TCU with KWP2000 protocol (on CAN)

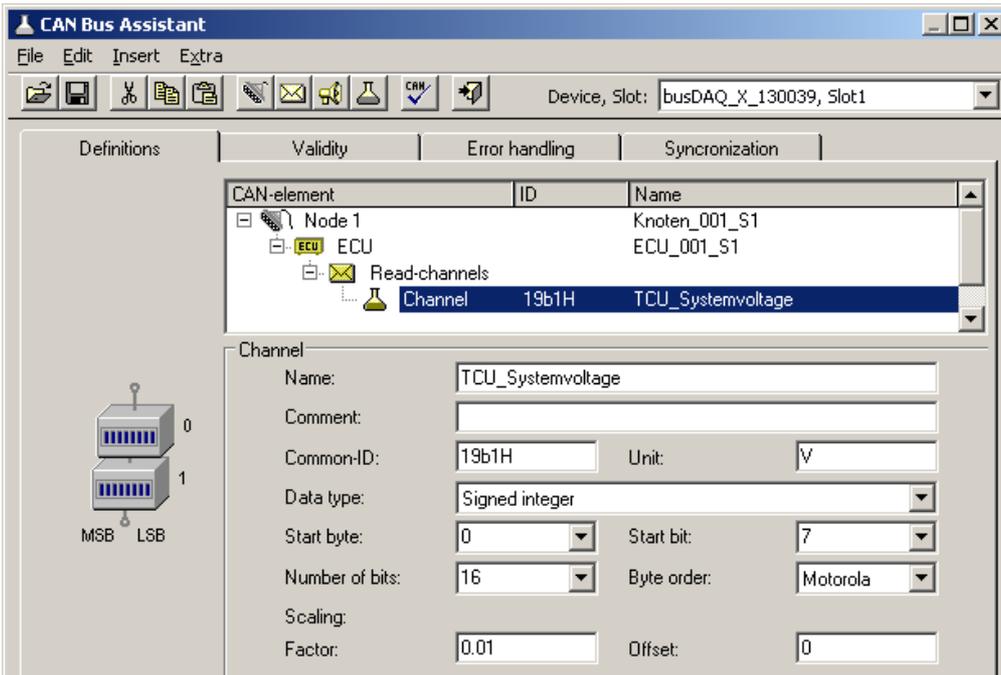
Once a new ECU has been added, the protocol is selected and the ID configured.



Set the identifier of the control unit

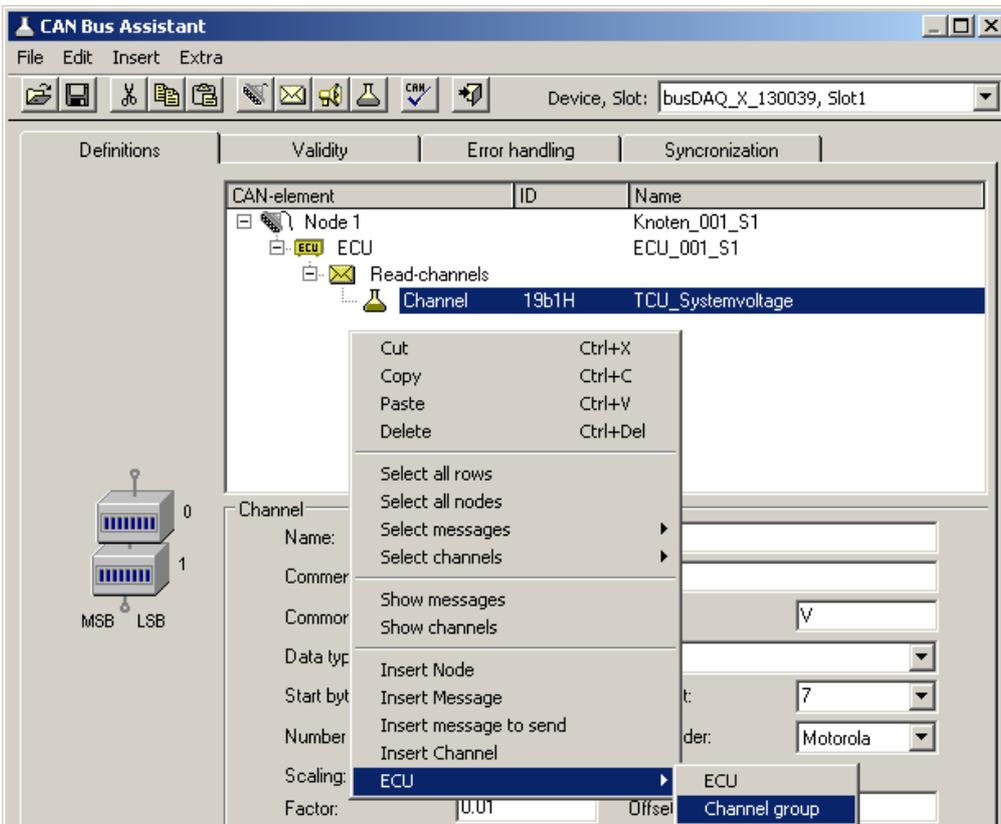
The value for StartDiagnosticSession is left at the default setting.

Now the first channel is configured with the name "TCU_Systemvoltage" and with the Common-ID 0x19C1. This is an unsigned integer with 16 bits and a factor of 0.01.

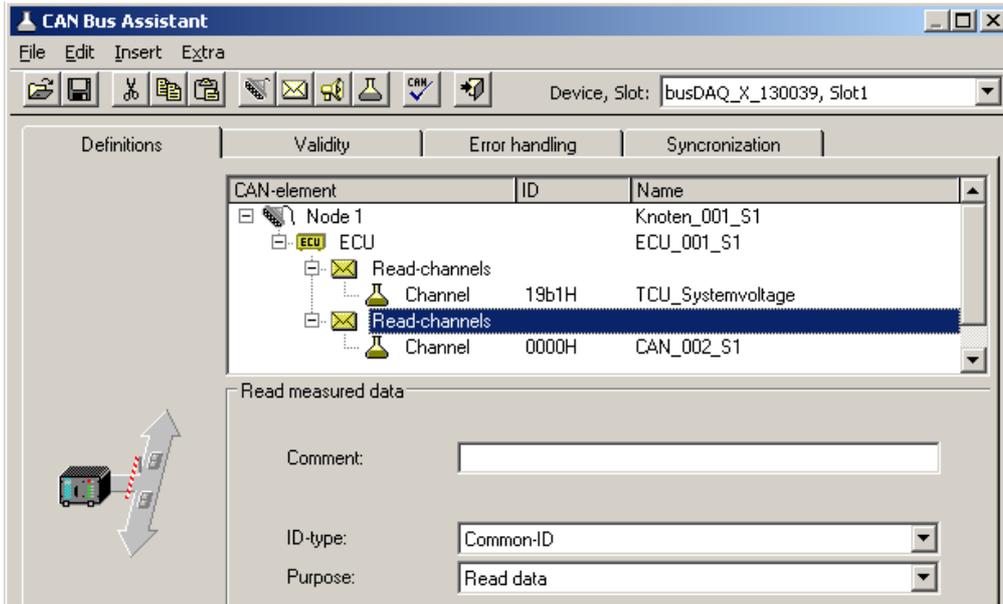


Configuring the channel

In order to be able to start an ECU routine from imc Online FAMOS, the routine must be configured in the CAN-Bus Assistant. For this purpose, a new channel group is first created.

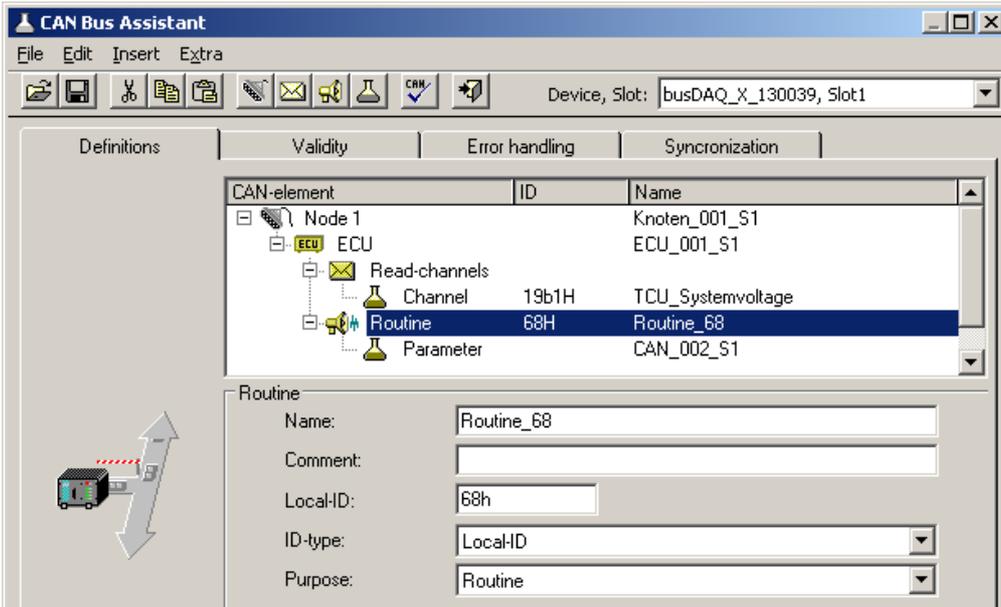


Create a channel group



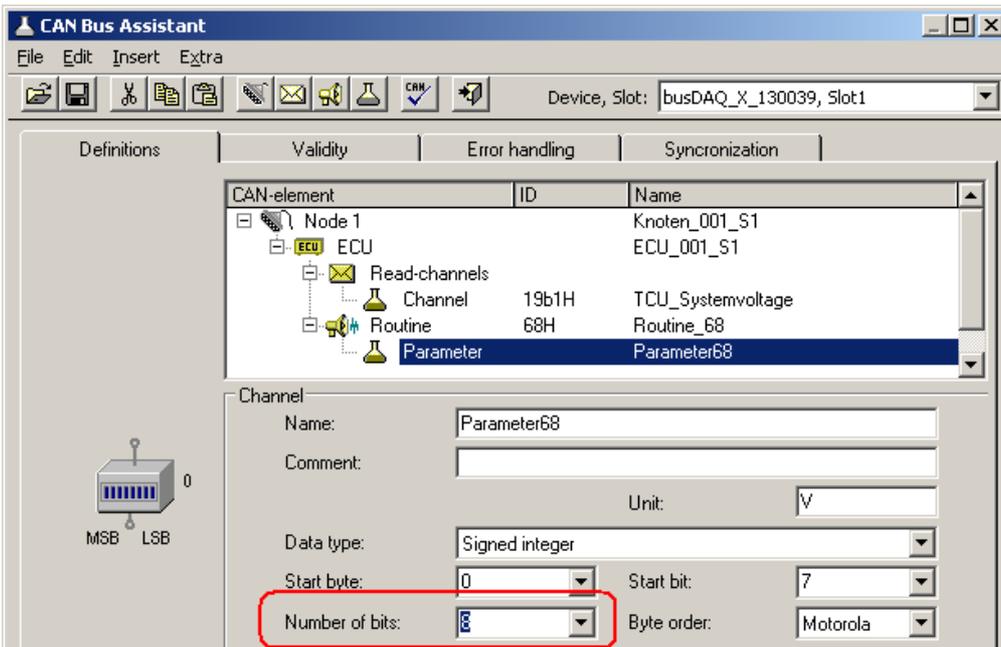
Channels of the channel group

The new channel group s configured as the routine's *Purpose* with a "Local Identifier" of 68h.



Setting the local identifier

Next, the parameter is given a name and configured for 8 bits.



Set paramter to 8 bit

This concludes configuration of the CAN-Bus Assistant.

imc Online FAMOS is now configured so that when the virtual bit 1 is set, the ECU's Routine_68 starts with Parameter 9. If the routine concludes successfully, the list of DTCs is imported.

```

; Initialization before the first measurement
OnInitAll
End

; continuous running
OnAlways
End

; run at start of measurement
OnTriggerStart(BaseTrigger)
End
    
```

```

; run during measurement
OnTriggerMeasure(BaseTrigger)
  if Virt_Bit01 = 1
    Virt_Bit01 = 0
    ECUSend_Routine68( 9 )
  End
End

; run at end of measurement
OnTriggerEnd(BaseTrigger)
End

; run in response to command
OnECUCmdReturn_ECU_001( OECR_Return, OECR_ECUCmd, OECR_CmdID )
  If OECR_Return = 0 AND OECR_CmdID = 0x68
    ECURedTroubleCodes_ECU_001( )
  End
End

```

Note

- If the values of the channels to be captured are queried one-by-one from the ECU, then the sampling interval configured for the channel is used. As for other CAN channels as well, the sampling interval can be set to values from 100µs on. ECUs normally require longer to respond (typ. 30 ms). For this reason, configuring a shorter sampling interval than the ECU's capabilities is not sensible.
- An ECU normally only executes one command at a time. If ECU values are captured, it is not possible to query them if any ECU function of long duration has been started. Therefore, the channels displaying the ECU's values can not receive any new values for the duration of the function. Depending on the configuration, they either display the last known value or a dummy value.
- In KWP2000 (on CAN), if a LogInKey is specified, the Key is also sent in response to Seed 00h 00h if the ECU is already enabled.

9.9.2.8.3.1 KWP2000 for TP2.0

Unless otherwise specified, the same descriptions of the properties apply as appearing under KWP2000.

source-address: The source address for the dynamic channel structure. Default: zero; for the tester.

target-address: The target address for the dynamic channel structure. Default: one; for the motor ECU.

Dynamic setup?: States whether to have dynamic channel structure, or whether to use permanent Identifiers instead.

9.9.2.8.3.2 Diagnostic On CAN

For *Diagnostic On CAN*, the same notes apply as for KWP2000, with the following differences:

- The service *ReadFreezeFrameData* is supported.
- The service *DynamicallyDefineLocalID* is supported.
- The service *RequestRoutineResultsByLocalID* is not supported.

DynamicallyDefineLocalID

- For a standards-conformant ECU, the properties for *Dynamically defined lists* can be set as follows:
- *Length of definition message*: -1 for automatic uses the maximum possible length. If the length is limited by the ECU's requirements, make this setting.
- *Defined data length*: -1, for automatic, uses the maximum possible block length. If the block length is limited by the ECU's requirements, make this setting.
- *Nested lists?*: If the lists cannot be used independently of each other, then set *Yes here*. Then, only one list will be queried during import. However, it is possible to use the previous list for definition purposes.

- *With Local- and Common-ID?:* If all Identifier types can be used for definition purposes, set "Yes" here. "No" is to be set if only addresses can be used.
- *Delete dynamic lists?:* If the dynamic lists need to be deleted before a new definition, then set "Yes" here.

Special imc Online FAMOS functions:

Function	Description
ECUStartSession_*	Runs the service StartDiagnosticSession.
ECUReadStatusOfDTC_*	The list of DTCs is imported by Service 17h and adopted in the channel for DTCs (e.g. DTC_Messages_1)
ECUClearDiagInformation_*	Deletes the list of DTCs.
ECUReadFreezeFrameData_*	The data of a particular FreezeFrame-number associated with a 'Diagnostic Trouble Code' are imported.
ECUSend_*	Writes to ECU objects or starts routines

 **Note**

For the DiagOnCAN-protocol 3-byte- and 4-byte-addresses with one or two length bytes are supported.

9.9.2.8.4 OBD-2

OBD-2 enables reading of standardized ECU channels. The ECU protocol OBD-2 is implemented in accordance with the SAE J1979DA:2021-04-21.

In imc Online FAMOS, there is for OBD-2 the function [ECUClearDiagInformation_*](#), which deletes diagnostics information saved in the control unit.

 **Note**

Operation requires skills in the OBD-2 protocol.

General notes

OnBoardDiagnostik, abbreviated OBD-2, is a standardized control mechanism enabling quick access to a vehicle's control instruments and sensor data. This interface is legally required in new vehicles.

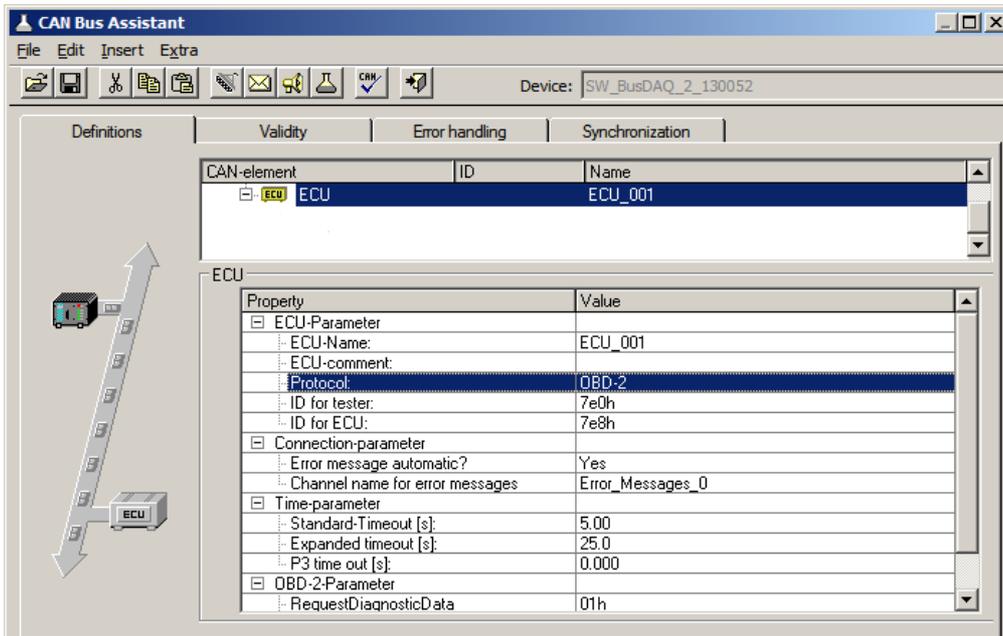
OnBoardDiagnostik monitors and saves even errors occurring during vehicle operation. Certain sensor data are recorded at fault condition.

These so-called freeze frames make quite effective troubleshooting and diagnostics possible. Necessary information must be supplied by the vehicle manufacturer. By this means, the vehicle data can be read out and used for diagnostics purposes.

The signal connection terminal is located in the passenger compartment, never in the hood or trunk.

9.9.2.8.4.1 Settings in the assistant

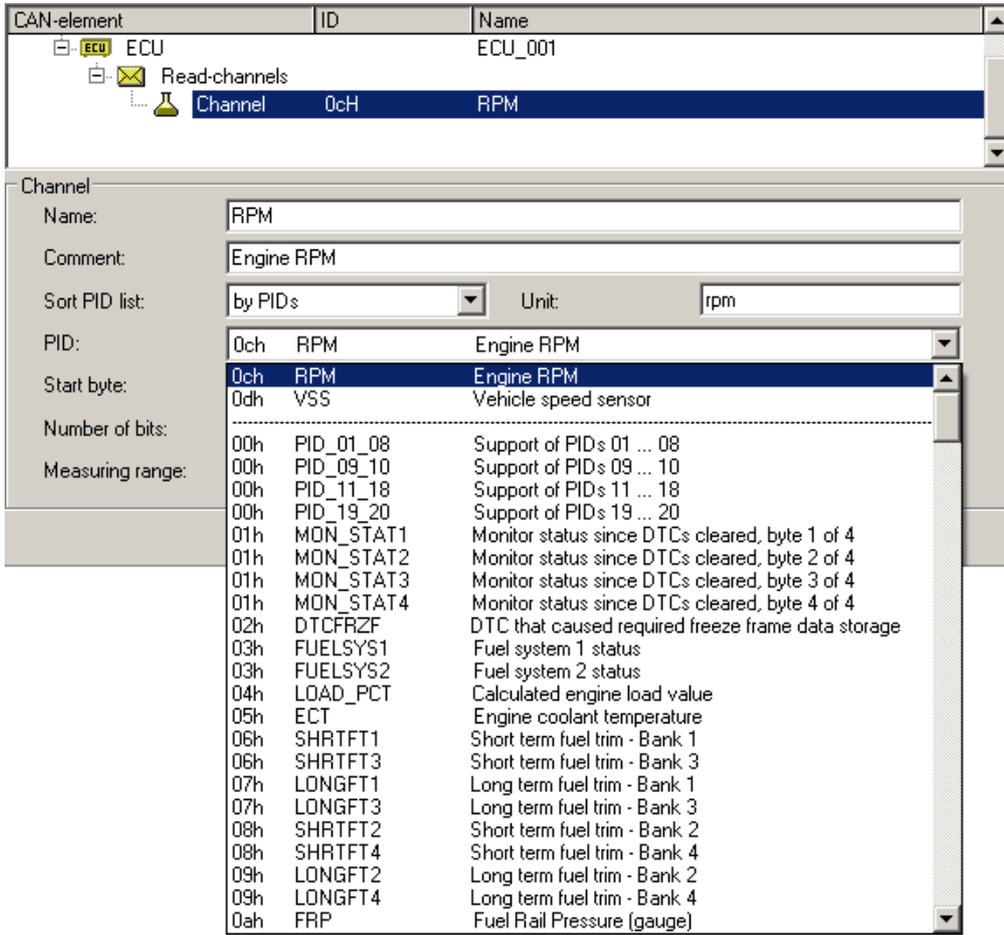
To select OBD-2 channels, add a [ECU](#)⁴⁴¹ to the configuration. Under ECU-Parameters, set the protocol type to OBD-2, before adding any channel:



ECU-protocol: OBD-2

With this type, there are only a few ECU parameters to set, e.g. there is no log-in command, no Security and no dynamic lists.

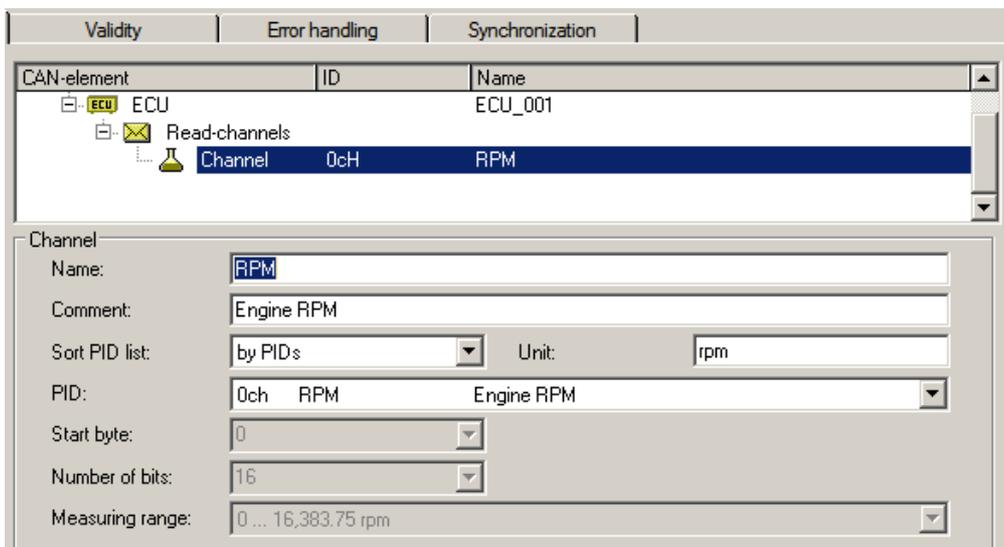
For selection of the ECU channels, permanent PIDs are set. Create a message and a channel and select the channel via the PID list:



Select a OBD-2 channel

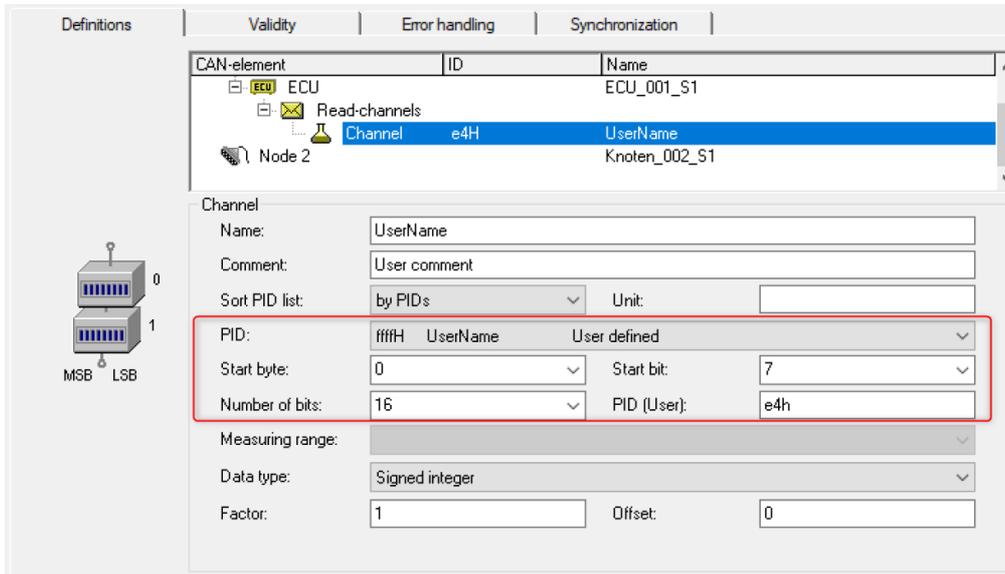
The PID can also be sorted by channel names, channel description or PID.

After selecting a PID, the *Name*, *Scaling*, *Unit*, *Comment*, *Start byte*, *Number of bits* and *Measuring range* for this PID are set automatically.



Properties of a OBD-2 channel

The last selection in the PID list is the entry "fffH UserName User defined". This selection can be used to add further PIDs that are not listed as standard. Another input field "PID (User)" appears here, in which the PID can be freely selected.



user defined PIDs

9.9.2.8.5 CCP

For this protocol, you must usually configure the following:

- *CRO-ID (ID for Tester)*: With this Identifier, messages are sent to the ECU.
- *DTO-ID (ID for ECU)*: Response messages from the ECU are expected to have this Identifier.
- *Station address*: The station address serves to distinguish the ECUs in cases where multiple ECUs are using the same Identifier.
- Seed-and-Key settings, see [here](#)⁴⁴³.
- *DAQ-lists*: If the ECU supports DAQ-lists, it is possible to use DAQ-lists to relieve the burdens on the ECU and the CAN-Bus.

Additional parameters

- *DAQ-lists only with bytes*: It is necessary to enter "Yes", if only single bytes can be defined in the DAQ-lists. In that case, two entries must be used for a 16-bit wide value.
- *Byte order*: Which sequence of bytes is to be used by default.
- *Mode channel name*: Unique name for the additional channel required for the use of DAQ-lists.
- *With SHORT_UP?*: If the ECU understands the command SHORT_UP, it is an easy way to query data.
- *With START_STOP_ALL?*: If the ECU understands the command START_STOP_ALL, then all DAQ-lists can be started together.
- *ECUs use same IDs?*: If multiple ECUs use the same Identifier, then each time the ECU is accessed, a connection having a specified station address, and subsequent disconnection, are necessary.
- *Standard-Timeout*: Time duration in seconds within which the ECU must respond.
- *LogInKey for send channels*: For the case that "With security access?" is set to Yes, the key for enabling write access is entered here.
- *LogInKey for DAQ-lists*: For the case that "With security access?" is set to Yes, the key for enabling the DAQ-lists is entered here.

- **Events:** In order to be able to use DAQ-lists, it is first necessary to define the associated events. An ECU event is a cyclical prompt to send specific DAQ-lists. An event can be triggered by timer or by an RPM-value, for instance, of the crankshaft. imc WAVE can only process time-domain events.
- **Properties of event:** Name: for designating an event
- **ID:** The identification used by the ECU. Usually begins with zero and ends with the number of events minus 1. For an entry which isn't used, enter -1.
- **base clock:** The minimum time period between two consecutive events.
- **Connection set-up automatic?:** Yes or No

DAQ-lists

In order to be able to use DAQ-lists, it is necessary to define the ECU's 'Events'. These are the sampling intervals available for the ECU. An 'Event' is defined with an arbitrary name, the Event-ID and the event's base clock rate.

To define a DAQ-list, the following must be specified:

- **Number of list:** The identification used by the ECU. Normally begins with zero and ends with the number of DAQ-lists minus 1. For an entry which isn't used, enter -1.
- **Event:** Specifies which event can trigger the DAQ-list. *Variable* is the setting in which the DAQ-list can be triggered by any event. The name of a particular event must be specified if the DAQ-list is to be triggered by only that event.
- **CAN-ID:** Identifier with which messages can be sent; -1 for the DTO-ID. If the DAQ-list can use other Identifiers instead of the DTO-ID, it is possible to set an unused Identifier. For many DAQ-lists, a different, permanent Identifier must be set.
- **MAX_ODT:** The maximum number of tables (Object Descriptor Table) per DAQ-list.
- **MAX_ODT_ENTRY:** The number of possible entries in a table (ODT).
- **FirstPID:** The PID of this DAQ-list's first ODT. The PID is used to identify the associated ODT. For each of the subsequent ODTs, a 1 is added. -1, for automatic, uses 0 for the first DAQ-list, MAX_ODT of the first DAQ-list for the second DAQ-list, etc.
- **With reduction:** Is Yes, if the basic clock rate needs to be subdivided. Then, messages are only sent for every n events. An ECU's channels are arranged in the DAQ-lists according to their sampling intervals. The channels with the shortest sampling interval are used first. If the DAQ-lists aren't sufficient for all channels, then the remaining channels are polled (read directly at every sampling instant).

Special imc Online FAMOS functions:

Function	Description
ECUSend_*	writes to ECU objects



Example

1. There are no DAQ lists ore they are not to be used.

The parameter number is set to -1 for all DAQ-lists.

2. A DAQ-list

```
Number: 0
Event: Variable
CAN-ID: -1
MAX_ODT: 10
MAX_ODT_ENTRY: 7
First PID: -1
With reduction? No
```

3. Two DAQ-lists each having a fixed event, special "First PID" and different size

```
Number: 1
Event: 0
CAN-ID: -1
MAX_ODT: 12
MAX_ODT_ENTRY: 7
First PID: 0
With reduction? No
```

```
Number: 0
Event: 1
CAN-ID: -1
MAX_ODT: 8
MAX_ODT_ENTRY: 7
First PID: 16
With reduction? No
```

9.9.2.8.6 XCP

For XCP, the same notes apply as for CCP, with the following differences:

- XCP does not have station addresses
- Seed-and-Key settings, see [here](#)⁴⁴³.
- *Dynamic DAQ-lists*: If DAQ-lists can be created as needed by the ECU, enter "Yes" here.

Additional parameters

- *Alignment in bytes*: Which alignment to adhere to in defining the ODTs.
- Messages with XCP standard are supported (short format: only the necessary bytes will be sent). Also messages are supported that are extended to 8 bytes.

9.9.2.8.7 GMLAN

The ECU protocol **GMLAN** resembles KWP2000. Import of A2L-files is possible.

The GMLAN-protocol has been implemented according to GMW 3110 Version 1.5 specifications (04.02.2004).

For dynamic lists, the following commands are additionally supported:

DynamicallyDefineMessage: assignment of a PID-list to a DPID

DefinePIDByAddress: assignment of unique PIDs to the (channel) addresses

ReadDataByPacketIdentifier: definition of DPID-list

With dynamic lists, the parameters in the group '*Dynamically defined lists*' must be set appropriately.

ModeChannelName: For dynamic lists, an additional channel is required (for internal purposes, such as with CCP).

Addressbytes: 2-, 3- and 4-Byte addresses are supported.

Supported imc Online FAMOS functions:

- ECUStartSession_**
- ECUStopSession_**

9.9.2.8.8 UDS

The UDS protocol is similar to the KWP2000 protocol and is implemented according to the ISO 14229-1 standard (04.2007).

- Seed-and-Key settings, see [here](#)⁴⁴³.

Additionally supported commands for dynamic lists:

- DynamicallyDefineDataID*: Assignment of addresses (channels) to unique PIDs
- ReadDataByPeriodicIdentifier*: Assignment of a PID-list to one of the specified sampling rates

With dynamic lists, it is necessary to set the parameters in the group '*Dynamically defined lists*' appropriately. The dynamic lists serve the purpose of transmitting multiple channels on the bus as quickly as possible while loading the bus as little as possible.

- ModeChannelName*: For dynamic lists, an additional channel is needed for internal purposes.
- AddressBytes*: 2-, 3- and 4-Byte addresses are supported

Functions supported in imc Online FAMOS:

- ECUStartSession_** : Executes the service StartDiagnosticSession.
- ECUStopSession_** : Executes the service StopDiagnosticSession.
- ECUClearDiagInformation_** : The list of diagnostics information is deleted
- ECUSend_** : Writes to ECU objects or starts routines

9.9.2.8.9 A2L: CURVE and VAL_BLK

Characteristic curves of the types *CURVE* and *VAL_BLK* of an A2L-file can be imported for the ECU-protocols CCP and XCP.

The option *Replace characteristic curves of the type CURVE and VAL_BLK with type VALUE* (see the menu item [Extra > Options](#)⁴¹³), all characteristic curves of both these types are replaced with characteristic curves of the type *VALUE*. This generates as many characteristic curves of this type as there were values in the original characteristic curve. For each selected characteristic curve of the type *VALUE*, one write-function is generated in imc Online FAMOS.

Without this option, for each characteristic curve of the type *CURVE* or *VAL_BLK*, a channel with the additional dialog element *Vector length*: is created, which can only be exported. For each characteristic curve, one write-function is automatically generated in imc Online FAMOS, in which the vector index must be supplied when it is called. Thus, there is one write-function imc Online FAMOS per characteristic curve.

Example: imc Online FAMOS:

```

; run during the measurement
OnTriggerMeasure(BaseTrigger)
  if IsV02 > 0
    ECUSend_CCP_struct_gen_vector_02( 54.5, 1) ; Index: 1,2
    ECUSend_CCP_struct_gen_vector_02( 58.5, 2)
    IsV02 = 0
  end
end
    
```

9.9.2.8.10 Properties in common

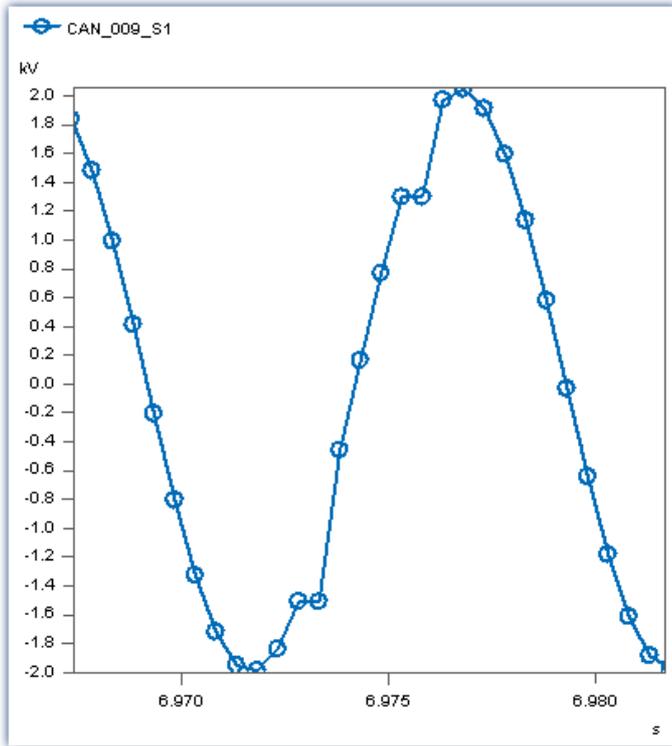
Property	Description
ECU-name, ECU-comment	An arbitrarily specifiable designation for the ECU, and a comment on the same. If the ECU's settings are imported from an A2L-file, then the ECU-name should not be changed. Upon re-importing a changed A2L-file, it is possible to adopt either only the changes, or also additional channels.
Protocol	This property should be set first, since the protocol used determines the particular properties. The protocol to be used in communicating with the ECU is to be chosen. As soon as a channel group has been created for the ECU, it is no longer possible to change the protocol.
Error messages automatically?	If the setting is "Yes", any error messages are automatically outputted in the error message channel. With "No", it is possible to evaluate errors occurring in the section "OnECUCmdReturn" in imc Online FAMOS.
Connection after error?	For the setting "Yes", the system attempts to set up a new connection whenever protocol timeouts occurred or messages containing measured data were delayed too long.
Channel name for error messages	Unique name for the error message channel.
Protocol-parameter	In the protocol-parameters, the command codes for the selected protocol are defined. These settings can mostly be left intact. Settings which need to be edited or checked are explained in conjunction with the protocol-specific properties.

9.9.2.9 Information and tips

Question: Why is the CAN-TimeStampASCII history not displayed completely?

Answer: The pretrigger for CAN-TimeStampASCII is never greater than 512Bytes. Even if a longer pretrigger time is set, a maximum of 512Bytes history is shown.

Question: At high sampling rates with CAN-Bus modules, some indicated values are duplicated or, in rarer cases, incorrect.



Answer: The CAN-Bus modules feature a considerable amount of jitter in outputting the measured data due to factors affecting the overall system. If the data are recorded at a high sampling rate, some values may be duplicated or, in rarer cases, erroneous data may appear.

Solution:

- a) **Universal:**
Sample the CAN-Bus channel with a time stamp instead of equidistantly.
- b) **When using imc WAVE:**
Use the CAN-1 synchronization. This is absolutely necessary if subsequent data processing is performed with in imc Online FAMOS.

9.10 Special modules and Third party devices

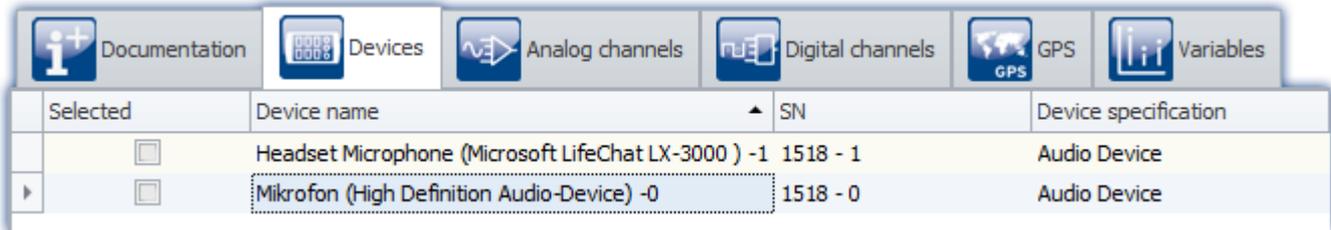
9.10.1 Third party devices

The following complete third-party devices are available:

- [AudioDevice](#)  461
- [ChannelLoader](#)  462
- [FunctionSimulator](#)  463

9.10.1.1 AudioDevice

It is possible to add the audio devices of your computer to imc WAVE. Therefore open the 3rd Party Device Management and add the AudioDevice. Afterward all audio devices are listed in the device list.

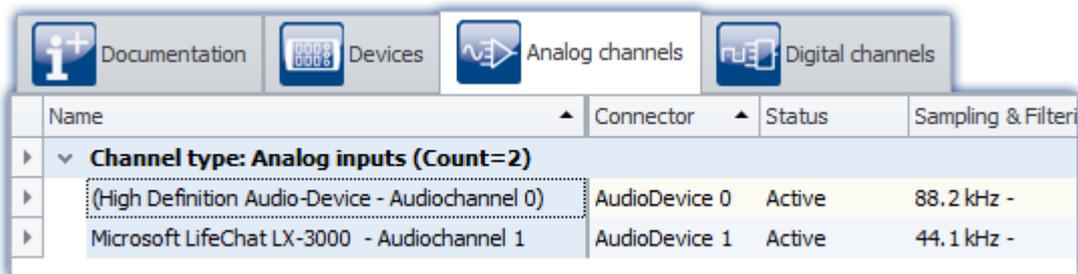


Selected	Device name	SN	Device specification
<input type="checkbox"/>	Headset Microphone (Microsoft LifeChat LX-3000) -1	1518 - 1	Audio Device
<input type="checkbox"/>	Mikrofon (High Definition Audio-Device) -0	1518 - 0	Audio Device

Device list: AudioDevice (Example)

On the **analog channel** page the audio channels are listed and also can be saved to the PC.

If multiple channels are available for a device, it is only possible to set one sample frequency for each channel of the device.



Name	Connector	Status	Sampling & Filter
Channel type: Analog inputs (Count=2)			
(High Definition Audio-Device - Audiochannel 0)	AudioDevice 0	Active	88.2 kHz -
Microsoft LifeChat LX-3000 - Audiochannel 1	AudioDevice 1	Active	44.1 kHz -

Channel list: AudioDevice (Example)

Note

- If the audio device is disconnected during the measurement, a error for each audio channel will appear in the logbook, showing that the data transfer is interrupted for this channel. To continue the measurement of this channel, start the audio device by the device control action in the Device table on the Setup page.
- If the experiment is opened on another computer, the corresponding hardware (USB device, sound card) must be available.
- If necessary, check your firewall for settings or rules: the process "imc.Studio.exe" (Program folder) needs the rights to access to the audio device.

Note

Remarks to certain devices

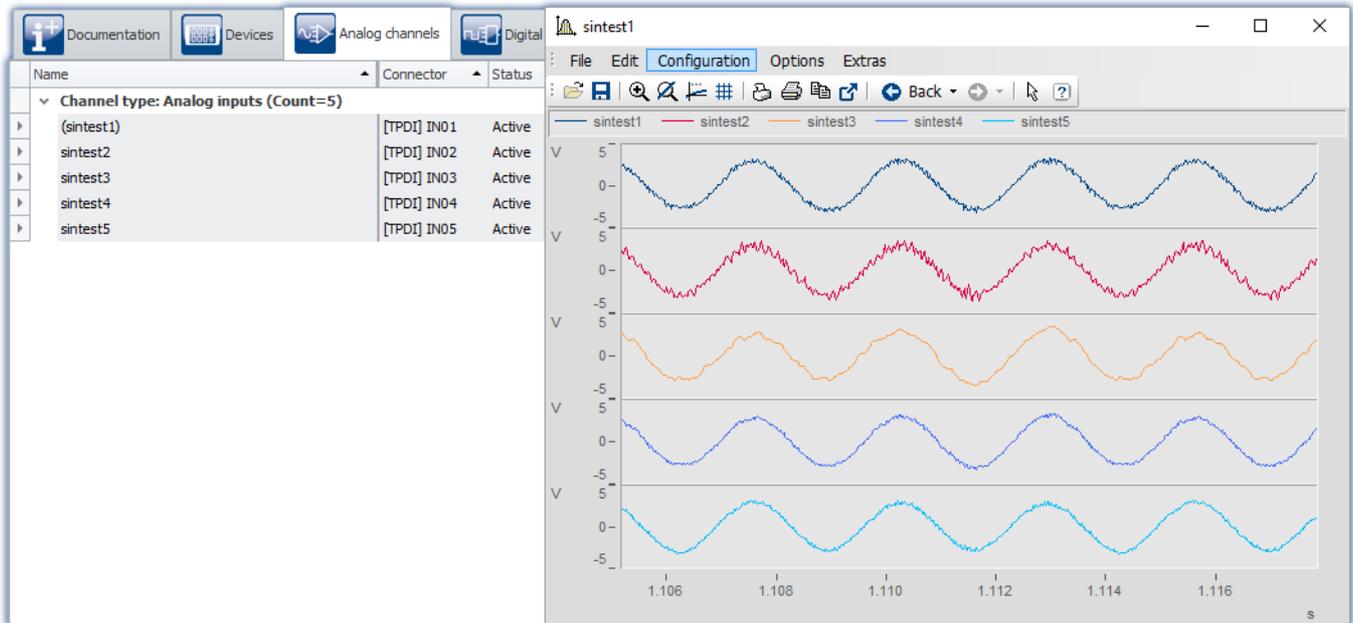
- With the webcam **Logitech HD Pro C920**, it was noticed that it was only delivering one audio channel, although according to the Windows settings there should be two channels. This may also apply to other audio devices.
- With the Headset **Microsoft LifeChat LX-3000**, it was noticed that after disconnecting and reconnecting the USB cable, a new measurement is only possible after the *second* start.

9.10.1.2 ChannelLoader

The ChannelLoader is for simulating or replaying data (e.g. recorded measurement data or imc FAMOS files). It can be added using the 3rd Party Device Management.

After selecting the ChannelLoader a file-open-dialog appears where one or more files can be selected. After the preparation the files are listed as channels on the analog channels pages.

After starting the device the signals are played periodically as input signal.



ChannelLoader: sintest1 - sintest5

Missing data

If the file to be played back is missing, an error message appears which provides the expected path and filename of the missing file.

The 3rd-party device uses experiment-specific files

When a file is selected which is located below the Experiment-folder, the location is saved in "relative" terms with respect to the experiment. If the location of the experiment changes, the file will still be found as long as it is also in the new experiment. The file from the old experiment-folder is not us.

Example: You create a dat-file in the "Meta"-folder. If you now export the experiment and import it to a different device or under a new name, the "Meta"-folder is also present there (together with the dat-file). By means of the relative path, the copied dat-file is found and used.

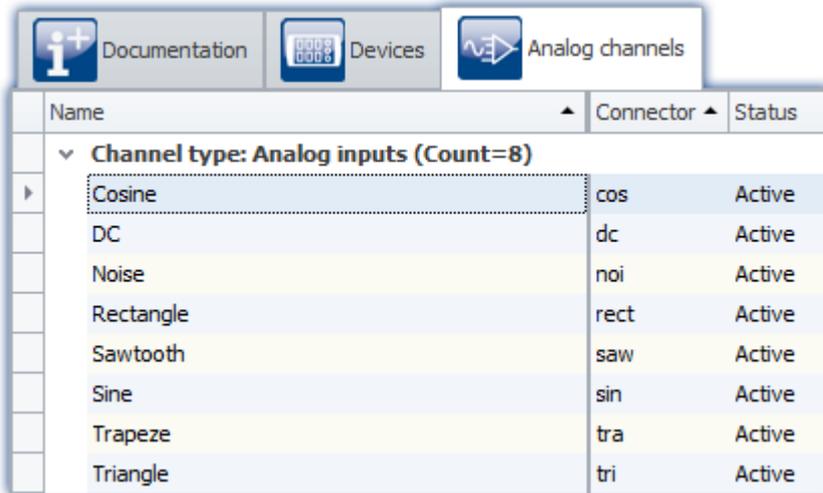
Note: Other folder names can also be used. See the advantage of the "[Meta](#)" folder.

Whether the path is saved in relative or absolute terms depends on the relationship to the Experiment-folder. If the file is below the Experiment-folder, the path is "relative". If the file is outside of the Experiment-folder, the path is "absolute".

9.10.1.3 FunctionSimulator

The FunctionSimulator is device which simulates various kinds of mathematics functions, including sine, cosine, triangle, sawtooth and squarewave.

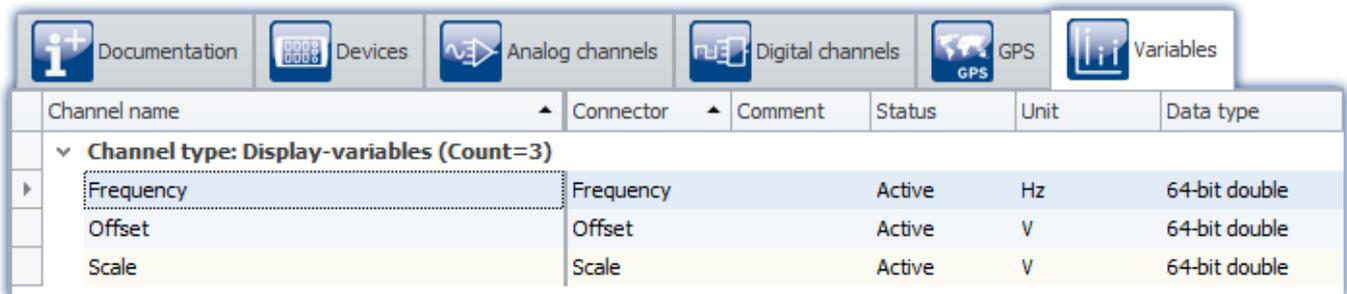
The analog channels then appear on the Setup-page under "Analog channels".



Name	Connector	Status
Channel type: Analog inputs (Count=8)		
Cosine	cos	Active
DC	dc	Active
Noise	noi	Active
Rectangle	rect	Active
Sawtooth	saw	Active
Sine	sin	Active
Trapeze	tra	Active
Triangle	tri	Active

Setup: FunctionSimulator (Analog Channels)

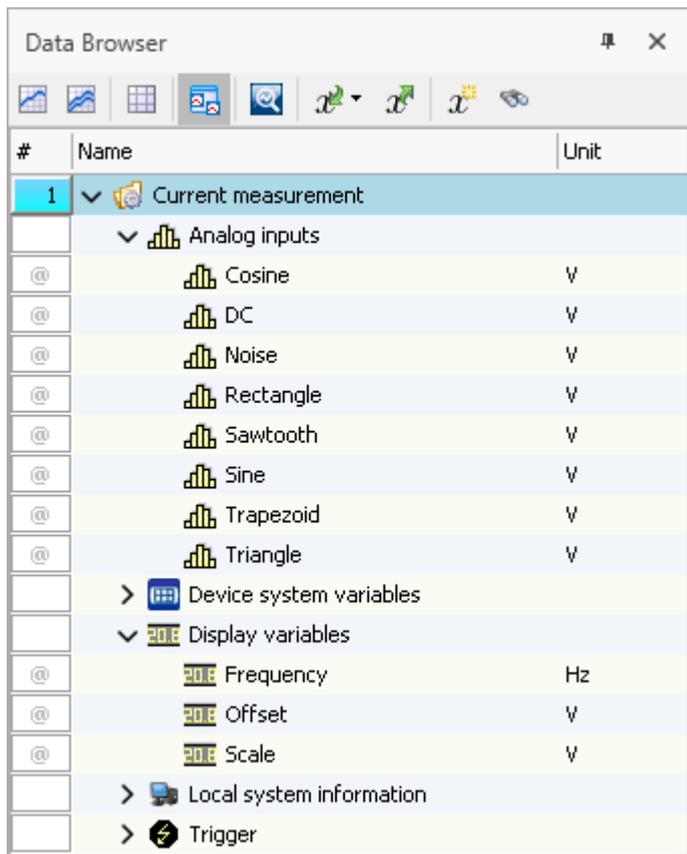
It is also possible to adjust the frequency, the scaling and the offset. The Display variables appear on the Setup page under "Variables".



Channel name	Connector	Comment	Status	Unit	Data type
Channel type: Display-variables (Count=3)					
Frequency	Frequency		Active	Hz	64-bit double
Offset	Offset		Active	V	64-bit double
Scale	Scale		Active	V	64-bit double

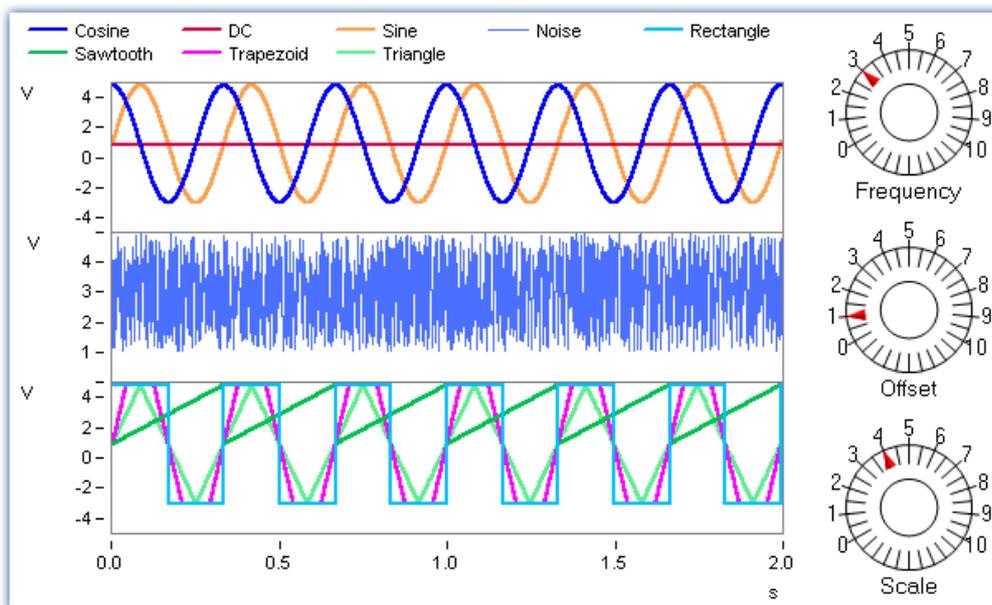
Setup: FunctionSimulator (Variables)

After selection and configuration of the device, the channels and Display variable are listed in the Data Browser.



Data Browser: FunctionSimulator

The signals can be displayed in the curve window.



Curve window: FunctionSimulator

By changing the Display variables, it is possible to alter the frequency, amplitude and offset during runtime.

 Note

- The FunctionSimulator can be used as trigger target action of an imc device.
- Important: The time zone in the imc device must be corresponding to the PC's time zone.

9.11 Sensors, characteristic curves and TEDS - for imc DEVICES devices

Supported imc measurement device groups

A device belonging to the [firmware group A](#) ¹⁴⁶ (imc DEVICES) - e.g. CRONOS Familie, C-SERIE, SPARTAN, BUSDAQ.

Definition of terms

Term	Description
Sensor	There are a variety of sensor types. They differ not only in shape, size, in the physical principles on which they are based, as well as in terms of their conversion between variables, but also in terms of properties, spec sheets and in how they are connected with measurement devices.
imc SENSORS	<p>imc WAVE supports export of sensor spec-sheet data from a sensor database and the application of the data in configuring channels.</p> <p>For this purpose, the program imc SENSORS must be installed. imc SENSORS is a tool for managing and editing information on sensors. This applies in particular to the specs in the spec sheets and the calibration values.</p> <p> For a detailed description of the database itself, see the user's manual for imc SENSORS.</p>
TEDS	imc WAVE supports readout of sensor spec-sheet data from a sensor TEDS and the application of this information in configuring channels.

imc WAVE Analyzer

On the pages "*imc WAVE Analyzer*", you can use sensor information from TEDS to configure channels.

9.11.1 What are TEDS

The sensor TEDS are serial ROMs. They are connected to an amplifier channel via a digital signal line (*1-wire-PROM*). Each sensor TEDS has a unique ROM-ID (*Sensor-ID, silicon serial number*). A sensor TEDS can also contain additional information on a sensor. Electrically erasable and re-writable sensor TEDS are referred to as *Sensor-EEPROMs*.

 Note

Some amplifiers support *TC-TEDS* (see the respective device manual for more information).

- Therefore the amplifier has to be equipped with a TC-TEDS-adapter. Instead of the standard DSUB-plugs, SMMI-thermocouples for type-K can be connected.
- By using suitable SMMI-thermocouple plugs (TC-TEDS) sensor information for thermocouples can be imported.

9.11.2 Writing sensor information to the channel

Prerequisites:

- The amplifier and imc WAVE must both support the sensor. The sensor information is only applied if the channel also supports the properties entered.
- A list of modules supported is presented in the chapter: "[List of supported sensors \(TEDS/imc SENSORS\)](#)".
- A list of amplifiers/devices which support sensors having characteristic curves is presented in the chapter "[List of amplifiers which support sensors having characteristic curves](#)".

 Note

Saving imported sensor information

The sensor information once imported and linked to a channel are **saved with the experiment**. If the experiment is later transferred (e.g. copied to another PC), this sensor information isn't lost!

TEDS - Read sensor information

imc STUDIO must be connected with the device.

Open the page: "*imc WAVE Analyzer*" > "*Channel definition*".

Read sensor information

- Select the channel desired, to which the TEDS is connected (multi-selection is possible if a TEDS is connected to each channel).
- From the list, select: "*Read sensor information*"
- Click on the button

The sensor information is read out and used to configure the channel.

 Note

Sampling rate

If a **sampling rate** is defined in the TEDS, **it is ignored**. Since more than two different sampling rates per device are not possible, this would in most cases lead to an interruption of the reading process.

9.11.3 Reset channel's sensor information

- Open the page: "imc WAVE Analyzer" > "Channel definition".
- Select the channel desired (multi-selection is possible)
- From the list, select: "Reset channel's sensor information"
- Click on the button

The sensor information is deleted.



Note

The configuration remains intact

The channel configuration is not reset to its condition before importing. The current configuration remains intact as long as the sensor did not set any parameters which were not possible without sensor information. In that case, the default value is restored.

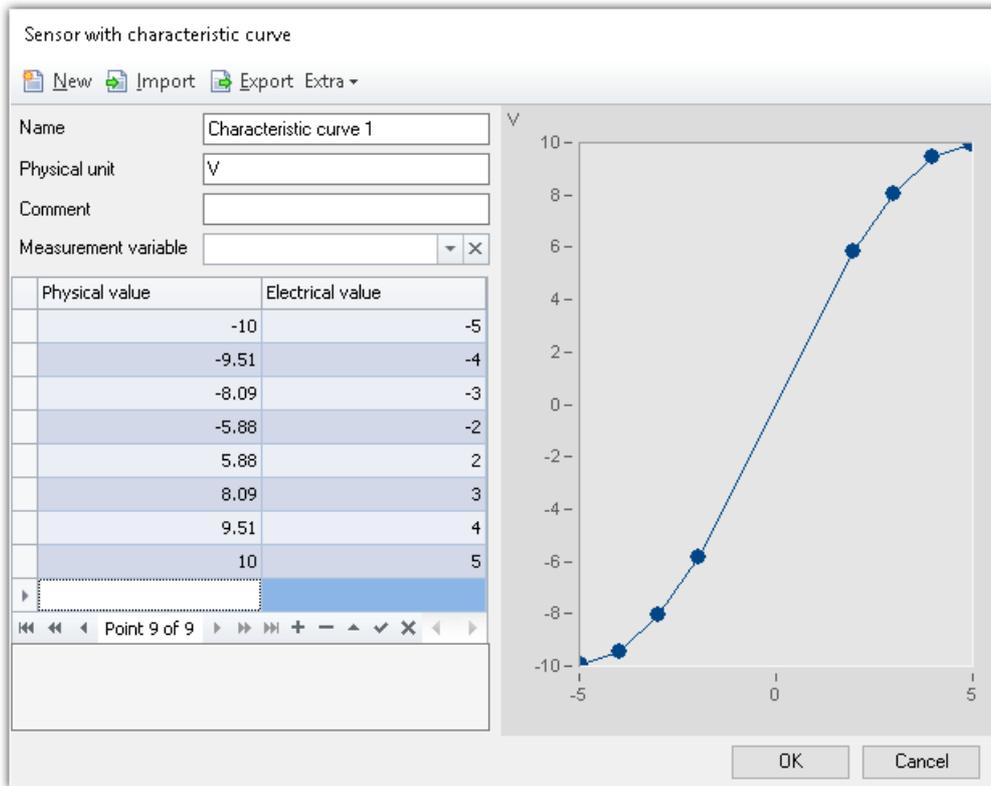
9.11.4 Creating a new sensor characteristic curve

Sensor characteristic curve provide the ability to convert measured electrical readings to physical values by reference to a plot of the relationship between those two variables.

Purpose	Description
Subsequent conversion, e.g. via imc Online FAMOS	Using the command "Measure a user-defined characteristic curve" , a characteristic curve is saved in a dat-file. This file can subsequently be imported as a supplemental file for imc Online FAMOS, for example.

The Dialog for entering the characteristic curve's points:

Entry technique	Description
Point measurement	<p>The electrical values are imported directly from the sensor.</p> <p> Please note that the signal's stability is neither monitored or controlled, so for this reason it only makes sense to apply this functionality to very stable constant signals.</p>



Example of a user-defined characteristic curve

Parameters of a characteristic curve

Parameter	Description
Name	Name of the characteristic curve to be created
Physical unit	Unit of the corrected value
Comment	
Measurement variable	Optional: Entry of values by means of the measured variable The electrical values can be imported from the sensor directly. Select the sensor's relevant pv-variable. The electrical values can now no longer be entered manually, but are instead measured directly. The measured values are entered.
Adjusting variable	- no function
Delay [s]	Optional: The current value is imported once the time entered has elapsed since the button is pressed.

Configuration by means of the command "*Measure a user-defined characteristic curve*"

In this command, the settings for the characteristic curve dialog are pre-defined. Parameters can be pre-defined and hidden, or editable. Additionally, a target file needs to be defined:

Parameter	Description
Filename	Storage location of the characteristic curve file in dat-format

Running the command

When the command is run, the dialog for capturing the characteristic curve's individual points appears. This is preconfigured according to the settings saved for the command. Once the data points have been captured and the dialog closed, the file is saved on the hard drive. Then it can be imported with a subsequent command for imc Online FAMOS, for example.

Entering/measuring points for the characteristic curve

A characteristic curve always consists of at least three points. In the table, the values for the characteristic curve are entered/measured. The characteristic curve is displayed accordingly in the curve window.

Values for the characteristic curve:

- Electrical value: the reading measured by the sensor
- Physical value: the value to be indicated in response to the corresponding electrical reading.

Measuring the values by means of the measured variable

The electrical values can be read from the sensor directly. Please note that the signal's stability is neither monitored or controlled, so for this reason it only makes sense to apply this functionality to very stable constant signals.

Select the sensor's associated pv-variable in the "*Measurement variable*" field. Now the electrical values can no longer be entered manually, but are measured directly. The measured value is automatically entered.

To read out a value, select the associated cell and click on the button at the cell's right edge:

	Physical value	Electrical value
	3	-494.11
	2	-353.11
▶	1	-295.123 ▶
*		

Acquiring a new electrical value via the measurement variable

When a "*Delay*" is defined, the value is only imported once the specified time has elapsed since the button was pushed.

9.11.5 List of amplifiers which support sensors having characteristic curves

Support of sensor characteristic curves is enabled for following devices:

Amplifier	Device				
	CRPL/SL	CRC	CRFX	CRXT	CANSAS
ICPU-8	●	---	---	---	---
DCB-8	●	---	---	---	---
LV2-8	●	---	---	---	---
UNI-8	●	---	---	---	●
ISO2-8	●	●	●	● as of 2.13 R1	---
ISOF-8	---	---	●	● as of 2.13 R1	---
UNI-4	●	●	●	● as of 2.13 R1	---
SC2-32	●	●	---	---	---
ICPU2-8	∅	●	●	● as of 2.13 R1	---
UNI2-8	∅	●	●	● as of 2.13 R1	---
DCB2-8	∅	●	●	● as of 2.13 R1	---
B-8	∅	●	●	● as of 2.13 R1	---
LV3-8	∅	●	●	● as of 2.13 R1	---

Device	Firmware
Cx-41xx-N	●
Cx-41xx-FD	●
SPAR-N	●

- : Feature supported imc STUDIO 5.2 R10 included firmware 2.13 R1
- ∅: Feature currently not supported
- : Amplifier not available for this device series

9.11.6 List of sensor supported (TEDS/imc SENSORS)

Implemented sensor information	
Accelerometer	Based on the ICP principle: current-fed; voltage measured with AC coupling
Bridge	Common bridge, bridge sensor, e.g. for force, no strain gauge
LVDT	LVDT sensors, differential coils. Supplied with AC voltage.
Microphone	Microphone with built-in amplifier
Pt100	Pt100 and related, e.g. Pt1000
StrainGauge	Strain-gauge bridge
Thermocouple	Thermocouple
Voltage	General voltage sensor
Current	Sensors with amperage output, e.g. 0..20mA or 4..20mA
Amplifier	Conditioner, amplifier with voltage output
Potentiometric	Potentiometric sensor, operated in half-bridge configuration, for example

Channel types that do not support TEDS or measurement modes that are not transferable from imc SENSORS to imc STUDIO

Encoder	Incremental encoder sensors
DigitalIn	Digital input of the measurement device, e.g. for querying a switch
ActorDigital	Digital output of the measuring device, e.g. relay for switching / control
Resistance	Resistor, usually current-fed and whose voltage drop is measured to derive the resistance, which is often proportional to another physical quantity.
Thermistor	Thermistor, resistance thermometer with non-linear characteristic curve
PiezoElectric	Piezo-electric (acceleration/force/sound), requiring a charge amplifier
ActorVoltage	Analog voltage output
ActorCurrent	Analog current output
ActorPulse	Pulse output, e.g. PWM (pulse width modulator)

9.12 Information and tips

9.12.1 Initial value for variables - Beginning the measurement - Jumps at the output



FAQ

Question: What happens when multiple "writers" each try to set their own value during the preparation? For instance, if a value is set for the DAC via a Widget, and also in imc Online FAMOS in the control command: "OnInitAll".

Answer: To prevent signal jumps on a DAC-output channel, then during initialization, the system checks whether a DAC output channel is initialized in the "OnInitAll" in the imc Online FAMOS-code. If so, this value is used and any previously set value from the data pool (e.g. via a Widget) is ignored.

When "*preparing*" (reconfiguring) imc Online FAMOS wins and the value in the imc WAVE variable is overwritten.

9.12.2 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?

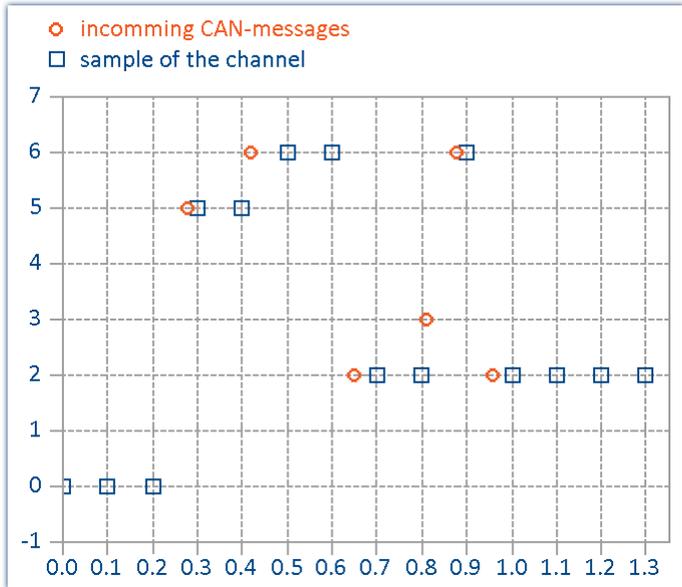
- When you intend to record the exact moment in time when a reading is captured.
- When the sensor is transmitting values at extremely irregular intervals, so that equidistant sampling would unnecessarily increase the data volume.

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.

9.12.2.1 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 Fieldbus is always recorded

This method has important consequences:

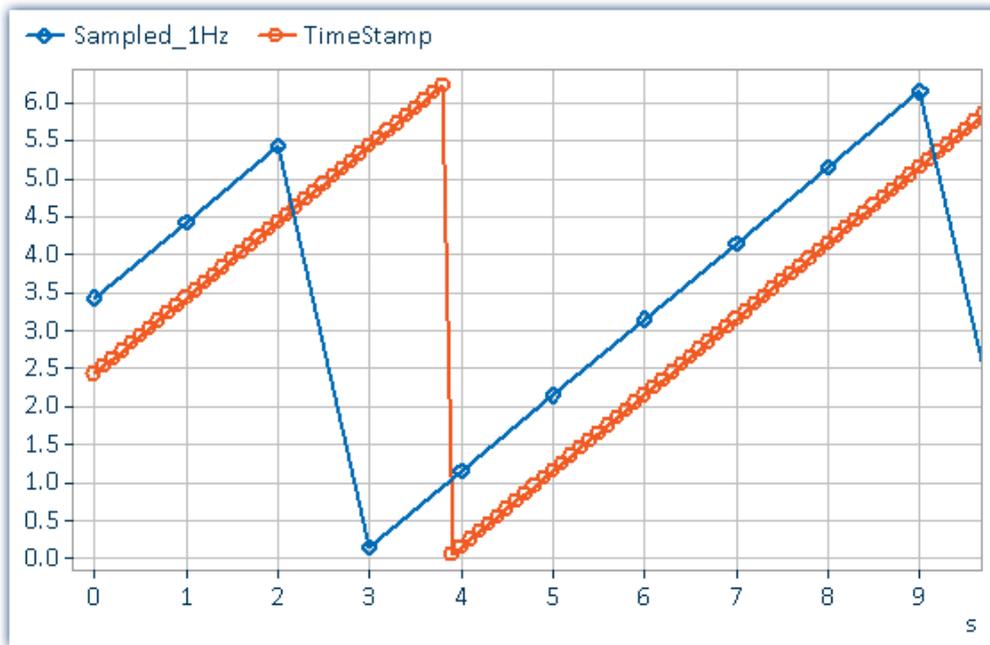
- As long as no messages have been received, the fieldbus channel is filled with zeroes. This can only occur at the beginning of an untriggered measurement before the first fieldbus messages have been received. When a measurement is stopped and restarted without being re-initialized, there are usually 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.

9.12.3 Pulse spikes in the signal

If the data capture is performed without using an analog low-pass filter, then disturbed signals may show outliers, peaks or spikes.

Tip: Nonlinear filters

Use the imc Online FAMOS functions [Median3](#), [Median5](#) and [SlopeClip](#) to cut out such disturbances. If low-pass filters are used (whether analog or digital), the peaks are only "mashed flat" and cause a local offset error.

9.12.4 Why are channels of one device not synchronized?

Answer 1: You are using different filter settings. Also the setting "AAF" for different sampling rates result in different low-pass filter settings. To sample two channels absolutely synchronous, the filter and sampling rate settings must be the same.

Answer 2: The RAM-buffer time is too big. In certain cases, the [RAM-buffer](#)³²⁴ size could result in sample shift. (Only applies to devices belonging to the [firmware group A](#)¹⁴⁶ (imc DEVICES))

Answer 3: Channels are assigned to different triggers. Only with curve window setting "date/time absolute" the channels are displayed correctly in respect of time. If the curve window is set to "linear" (s) the time of the triggers will be displayed together.

Reference

Other scenarios are described in the section "[Synchronization](#)"²²⁵".

9.12.5 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	<p>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).</p> <p>As soon as a new file is started, the time stamp is again accurate (e.g. interval boundary).</p> <p> For devices belonging to the firmware group A ^[146] (imc DEVICES): In the "SysLog"-file ^[312], it is exactly specified when a data overflow occurred and how many data points are missing. Thus, the file can be corrected.</p>
Data storage on the PC	<p>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.</p>



Reference

See also

- ["Behavior upon outage of the network connection"](#) ^[323] (RAM-buffer)
- ["Avoidance of Data Overflow"](#) ^[475]

9.12.6 Avoidance of data overflow

Very many factors can contribute to a data overflow. There is a concrete correlation to the following factors:

1. [Data Rate, Channel Count and Formatting](#) ^[476]
2. [Occupied Memory, Measurement Duration and Folder Count](#) ^[477]
3. [Manufacturer and Age of the Storage Medium](#) ^[477]
4. [RAM buffer duration of the channels](#) ^[322]

See also:

- [Set long-term measurement correctly](#) ^[477]
- [More tips](#) ^[478]

The accumulating data volume can **not** be exactly calculated for the following settings:

- Triggers, especially ones with long pre-trigger times
- imc Online FAMOS functions
- CAN log channels

 Note

- In order to avoid data loss, it is urgently recommended to **test** whether the anticipated accumulating data volume within a specified time frame can reliably be saved on the desired 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.

 Reference

See also

["Behavior upon outage of the network connection"](#) 

Data Rate, Channel Count and Formatting

Every storage medium has a maximum data rate at which data can be continuously written to the medium. The manufacturers commonly state maximum values which cannot be achieved in practise.

Every storage medium has a maximum data rate at which data are continuously written to the medium. The maximum data rate is normally stated in kByte/s. Note that the size of a measured value (sample) can vary from 2 Bytes (some signal conditioner channels) to 4 Bytes (e.g. virtual channels) all the way to 10 Bytes (e.g. Float values with time stamp). With an aggregate sampling rate of 400 kSample/s (only signal conditioner channels), this results in a data rate of 800kByte/s for the storage medium! Older flashcard memory models are not able to go that high; flashcards "age" over the duration of their use.

For devices of the firmware group A (imc DEVICES):

The size and number of assignment units (clusters) and thus also the [file system](#)  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.

Occupied Memory, Measurement Duration and Folder Count

The larger the data volume and the more folders on the storage medium, the longer it takes to find free clusters. In order to ensure high data rates, it is recommended that the storage medium be deleted from the measurement or be re-formatted. This also applies to large card volumes.

Make an estimate of the prospective number of folders. Particularly in the modes "[Interval data saving](#)"³¹⁸ and "[Trigger events in separate folders](#)"³²⁹, a great many folders can be created in a short time.

Note

Creating more than 1000 folders is to be avoided, since this significantly slows down the file system.

Manufacturer and Age of the Storage Medium

In recent years, manufacturers have succeeded in raising the maximum data rate per second. Be certain to check your card before using it. The cards offered by imc are tested for their data rate and functionality.

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.

Set long-term measurement correctly

Say, for instance, you wish to make a really long measurement. You set the channel sampling rate to 100 kHz, the measurement duration to "*undefined*" and the circular buffer time for the display to "*unlimited*". Then you set the channel data to be displayed in the curve window and start the measurement. Here's what happens:

1. The measurement runs smoothly, at first.
2. After a while, there seem to be pauses for several seconds. The software seems to run more slowly.
3. Later, a "*Data overflow*"-message appears.

Diagnosis: As measurement data are captured by the measurement device, the operating system requests memory. Also, the relocation of memory space can take considerable time. If this time is too long, the device is no longer able to stash the data, the PC doesn't read the data out in time, data are lost, a data overflow occurs.

→ Tip: Circular buffer memory

- For the purpose of displaying the channel data from indefinitely long measurement runs, use the circular buffer memory. This circular buffer displays the data for a specified duration, e.g. 30 minutes. This memory is demanded only once at the start of the measurement and is cyclically overwritten. It is then possible to show up to the last 30 minutes in the curve window. The system continues to work at constant speed, for as long as needed.
- Data can also be saved without the circular buffer memory. This will not require any of the operating system's RAM.

→ Tip: Scroll-mode

When the curve window is used to display data, it should be used in "[Scroll-mode](#)⁹⁹⁹". In Scroll-mode, the curve window always displays the same amount of data, so it always keeps the same speed. If you don't set the X-axis to Automatic (or Unzoom), the amount of data displayed constantly increases and plotting takes longer and longer. Scroll-mode actually cannot prevent data overflow in the PC-RAM! But it is a helpful technique for keeping the displayed time window constant.

More tips

If the error message "*Data overflow*" appears during a measurement, it means that some measured data have been lost. The device buffers captured data for a specified time interval. If the PC is too busy to read in data for longer than this time, the device memory will run out of capacity and any newly captured data will overwrite the previous data.

→ Tip: Increase buffer time

For devices belonging to the [firmware group A](#)¹⁴⁶ (imc DEVICES): Use the controls on the page: "*Analog/Digital channels*" in the dialog: "*Data transfer*" to set a [greater buffer time](#)³²⁴ for each channel. "Auto" is min. 10 s; it is often possible to set a much longer time. Set the same buffer time for all channels transferring data to the PC, and make it as long as possible. Channels which are only processed by the device don't need extensive buffering; 2 s are usually enough.

For devices belonging to the [firmware group B](#)¹⁴⁶ (imc DEVCORE): Here, no modifications are necessary. The [buffer duration](#)³²³ is optimally allocated automatically.

→ Tip: Reduce data rate

If possible, reduce the data rate by means of the data reduction functions in imc Online FAMOS.

→ Tip: PC processing time, Scroll-mode

Avoid performing actions with the PC which demand much processing time. E.g., working with other applications, using Unzoom or automatic X-axis in the curve window when there are huge amounts of data, using complicated curve window display options like Waterfall, Color map, many events, plotting with big dots, ...

→ Tip: Monitor the system variable "[Critical channel fill level](#)⁷³⁹"

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.

9.12.7 Parameterizing bridges

Question: How can I parameterize a bridge in imc WAVE, if my scaling factor is unknown?

Answer: Condition: The bridge can be unbalanced with a defined load.

We demonstrate the issue using the example of torque measurement, in which a weight is used to apply a defined torque of 4.17 Nm to the shaft.

1. Calculating the actual scaling factor: Activate the channel on the page: **Analog/Digital channels** in the dialog: **Channel definition**. Set the desired sampling rate (Dialog: **Sampling & Preprocessing**). Leave the scaling factor (Factor) (Dialog: **Range & Scaling**) at the default value 1 "mV/V" / "mV/V".
2. Select the desired bridge type (half-, full-, quarter bridge in the Voltage mode) in the dialog: **Measurement mode** and set the necessary power supply. Initially, the maximum possible input range is selected.
3. Save the experiment and then on the page: **Channel balance** in the dialog: **Balance** select the desired channel from the channel list. Activate bridge balancing in the column *Balance*.
4. Once the bridge has been balanced, the shaft is stressed with the defined torque and a measurement is performed. The measured value is, for instance, 0.5mV/V (bridge diagonal voltage). Naturally, this value is relatively imprecise, since the input range is 500mV/V.
5. Stop the measurement. On the page: **Analog/Digital channels** open the dialog: **Measurement mode** and then select the next bigger input range from 0.5mV/V, in other words 1mV/V and then repeat the measurement. Now you obtain a more exact measured value of, for example, 0.541mV/V.
6. Using this value, we can calculate the scaling factor: $4.17 \text{ Nm} / 0.541 \text{ mV/V} = 7,708 \text{ "Nm" /mV/V}$. This value can now be entered in the dialog: **Range & Scaling** for the scaling factor Factor.
7. You now have a choice of input ranges in physical units in the dialog: **Measurement mode**. Here, select the maximum expected number of measured values as the desired input range.
8. If the input range was changed, it is necessary to repeat the bridge balancing!

Note

- **Caution: Every time the input is changed, it is necessary to perform bridge balancing again before starting the measurement!**
- The balancing values are stored with the experiment and imported by the device upon preparation of the measurement. This means that after activating or deactivating the measurement device, the balancing values are not available in the device! Exception: Autostart / Diskstart.

9.12.8 Scaling and input range

imc WAVE returns measured data correctly scaled in the appropriate physical unit. However, there are a number of different settings possibilities which may cause confusion.

Basically the input range is derived from the selected electrical input range on the page: **Analog/Digital channels** in the dialog: **Measurement mode** and the scaling factor (Factor) on the page: **Analog/Digital channels** in the dialog: **Range & Scaling**.

The following possibilities are available:

- [Scaling factor without offset](#) 
- [Scaling factor with offset](#) 
- [Calculated two-point scaling](#) 
- [Two-point scaling by means of the amplifier adjustment dialog](#) 

Scaling Factor without Offset

If the expected electrical input range is not known, ignore step 1 below.

In the example, a full bridge is described which returns a maximum electrical value of 35 mV/V at a scaling factor of 356.65 N per mV/V.

Step	Description
1. Input range	On the page: Analog/Digital channels in the dialog: Range & Scaling , select the appropriate input range so as to make the maximum expected value fall in the upper third of the input range. For an expected maximum value of 35 mV/V, the appropriate input range is 50 mV/V.
2. Scaling factor	Enter the scaling factor as the Factor; in this example 356.65 N/mV/V.
3. Input range	Note that afterwards the input range is also indicated together with the physical unit.

The full bridge used in the example normally causes an offset in its rest state. For measurement bridges, this offset should always be compensated using on the page: **Channel balance** the dialog: **Balance**.

Scaling Factor with Offset

The example presents a displacement sensor which delivers 0 V at -11.55 mm. The scaling is 0.2597 V/mm. The maximum value is 25 mm.

Step	Description
1. Input range	On the page: Analog/Digital channels in the dialog: Measurement mode , select the appropriate voltage range. The total displacement is $25 - (-11.55) \text{ mm} = 36.55 \text{ mm}$. This means that the sensor delivers a max. value of $36.55 \text{ mm} \cdot 0.2597 \text{ V/mm} = 9.492 \text{ V}$. If necessary, set the coupling to DC and then set an input range of 10 V.
2. Scaling factor	Go to the dialog: Range & Scaling . The scaling factor must be converted to physical units per volts: $Y\text{-factor} = 1 / 0.2597 \text{ V/mm} = 3.8506 \text{ mm/V}$ Note that the value must be expressed in the SI-unit "m" and not "mm".
3. Offset	Enter the offset if -11.55 mm.
4. Input range	The offset entered causes the input range to shift.

Calculated Two-point Scaling

The example describes a pressure sensor delivering a current of 4 to 20 mA. The spec sheet indicates that 4 mA represents applied pressure of 0 bar and 17.5 mA represents 10 bar.

Step	Description
1. Input range and Coupling	On the page: Analog/Digital channels in the dialog: Measurement mode , select the appropriate current range. The UNI-8 used in our example must first be set for DC-coupling. Next select current measurement and then an input range of 20 mA.
2. Range & Scaling	Go to the dialog: Range & Scaling .
3. enter the values	Next enter the two value pairs from the spec sheet.

Two-point Scaling by means of the Amplifier Balance Dialog

Along with calculated two-point scaling which is performed by the measurement system's motherboard, it is possible to have the scaling performed already by the amplifier module. In this case, it is no longer necessary to deal with the offset and factor on the page: **Analog/Digital channels**. However, this method requires measurements of both reference points which means a connection is needed to the device.

With this technique, you need to know the expected electrical measurement value in order to be able to select the appropriate input range. This may no longer be changed later, since that would make the adjustment values obsolete.

In the example below, a power sensor is to measure the load on a crane. A standard weight of 100 kg is available. The sensor delivers a maximum of 8 V.

Step	Description
1. Input range and Coupling	On the page: Analog/Digital channels in the dialog: Measurement mode , select the appropriate voltage input range. If necessary, set the coupling to DC and then select an input range of 10 V.
2. Scaling factor	Go in the dialog: Range & Scaling . Enter 1 N/V as the Factor.
3. Channel balance: Point 1	<p>Activate the device and connect it with the PC.</p> <p>Go on the page: Channel balance in the dialog: Scaling / Cable compensation without Sense line.</p> <p>Connect the sensor and enter 0 N under "<i>Point 1 set value</i>". Measure the unstressed crane by clicking on the button "<i>Scaling point 1</i>".</p>
4. Channel balance: Point 2	Apply a load to the crane with the standard weight. The force in Newtons is $100 \text{ kg} \cdot 9.861 \text{ m/s}^2 = 986.1 \text{ N}$. Click on " <i>Scaling point 2</i> ".
5. Input range	<p>In the <i>Scaling</i> column, click on the button: <i>Two-point scaling</i></p> <p>Go back to the page: Analog/Digital channels in the dialog: Range & Scaling. The resulting input range is displayed.</p>

The method is recommended because it takes the entire measurement chain into account.

It is theoretically possible to enter an additional Y-factor, and an offset.

9.12.9 Breakdown of the start/stop and triggering behavior

How does the system behave exactly upon starting? How many clock cycles are required until the trigger is released? What are the effects of pre-processing?

Here you will find various applications examples which clarify these questions.

There is a distinction among three device types.

- fixt configuration devices belonging to the [firmware group A](#)^[146] (imc DEVICES), such as C-SERIES, CRONOS-SL, CRONOScompact, ...
- dynamically configurable devices belonging to the [firmware group A](#)^[146] (imc DEVICES), such as CRONOSflex, CRONOS-XT
- devices belonging to the [firmware group B](#)^[146] (imc DEVICEcore), such as ARGUSfit, EOS

The following examples each present one device type as an illustration for their respective group. They thus represent the other devices within their group:

1: CS | 2: CRFX | 3: ARGUSfit

Starting with "BaseTrigger" (without extra trigger)

CS	CRFX	ARGUSfit
The first measured value appears directly. The subsequent measured values then each appear after elapse of the sampling interval specified.	The first measured value appears after a delay. Only once the second measured value is available, both values are displayed. The subsequent measured values then each appear after elapse of the sampling interval specified.	The first measured value appears at the full second. The subsequent measured values then each appear after elapse of the sampling interval specified.
Example: sampling interval: 5 s The first measured value appears immediately. Its relative position is 0 s. The second after 5 s, ...	Example: sampling interval: 5 s The first two measured values appear after 5 s. Their relative positions are 0 s and 5 s. The third appears after 5 s more, ...	Example: sampling interval: 0.01 s The first measured value appears at the next full second. Its relative position is 0 s. The second after 0.01 s, ...

Re-start

CS and CRFX	ARGUSfit
The first measured value appears directly at its associated clock pulse in reference to the previous measurement. This means that if two sampling rates are used, the slower one is not synchronized to the relative position 0 s, as faster channels are.	Like the first start. The first measured value appears at the full second.

Stopping the measurement:

CS	CRFX	ARGUSfit
The measurement stops immediately; no subsequent measured value appears.	Once the measurement is stopped, one more measured value for each channel appears after the sampling interval specified. Only after that, the measurement is really concluded. Stopping is thus delayed by a maximum of one sampling interval of the slowest channel.	The measurement stops immediately; no subsequent measured value appears.

Pre-processing: e.g. arith. mean

CS	CRFX
Display and sorting of the first measured value is delayed due to the duration of the averaging. The subsequent measured values appear after elapse of the specified sampling interval, multiplied by the number of reduction points.	Pre-processing not possible
<p>Example: Sampling interval: 1 s Function reduction points: 8</p> <p>The first measured value appears after 8 s. Its relative position is at 8 s and not synchronized to the others at 0 s. The second appears after another 8 s, ...</p>	

Re-start, with pre-processing

CS	CRFX
The first measured value appears directly at its associated reduced clock pulse in reference to the previous measurement. This means that if two sampling rates are used, the slower, reduced channel is not synchronized to the relative position 0 s, as faster channels are.	Pre-processing not possible

Starting with a fast channel governed by "BaseTrigger" and a slow channel governed by "Trigger_01" | Trigger-source: Virt-Bit

CS and CRFX	ARGUSfit
The first measured value appears directly at its clock pulse once the trigger has released. The second value appears after elapse of the sampling interval specified.	---
Example: Sampling interval: 5 s; the trigger releases after 21 s The first measured value appears (in absolute terms) 25 s after the start.	---

Stopping "Trigger_01" (measurement continues)

CS and CRFX	ARGUSfit
Once the trigger is stopped, two more measured values appear	---
Example: sampling interval: 5 s; triggered measurement is concluded after 41 s At each of the following times, another measured value appears: 45 s and 50 s	---

Starting with fast channel governed by "BaseTrigger" and slow channel governed by "Trigger_01" | Trigger-source: same channel

CS	CRFX	ARGUSfit
For: Signal > Level or Positive slope The first measured value appears two clock pulses after the event. The second value appears after elapse of the sampling interval specified.	For: Signal > Level or Positive slope The first measured value appears directly at its clock pulse once the trigger has released. The second value appears after elapse of the sampling interval specified.	For: Signal > Level or Positive slope The first measured value appears directly at its clock pulse once the trigger has released. The second value appears after elapse of the sampling interval specified.
Example: Sampling interval: 5 s; the trigger releases after 21 s. The first measured value appears (in absolute terms) 30 s after the start.	Example: Sampling interval: 5 s; the trigger releases after 21 s The first measured value appears (in absolute terms) 25 s after the start.	Example: Sampling interval: 0.01 s; the trigger releases after 5.001 s; The first measured value appears (in absolute terms) 5.02 s after the start.

Stopping "Trigger_01" (measurement continues)

CS	CRFX
<p>For: Signal > Level</p> <p>After stopping the triggered measurement, three more measured values appear.</p>	<p>For: Signal > Level or Positive slope</p> <p>After stopping the triggered measurement, one more measured value appears.</p>
<p>Example: Sampling interval: 5 s; the triggered measurement concludes after 41 s.</p> <p>At the following times, more measured values appear: 45 s, 50 s and 55 s</p>	<p>Example: Sampling interval: 5 s; the triggered measurement concludes after 41 s.</p> <p>At the following time, one more measured value appears: 45 s</p>
<p>For: Positive slope</p> <p>After stopping the triggered measurement, two more measured values appear.</p>	
<p>Example: Sampling interval: 5 s; the triggered measurement concludes after 41 s.</p> <p>At the following times, more measured values appear: 45 s and 50 s</p>	

9.12.10 Exchanging values among the devices

For exchanging or processing values across devices, there are a variety of approaches and methods. Some are specific to a device type, others are specific to the imc WAVE components (e.g. imc Inline FAMOS).

Exchanging via digital outputs/inputs

Communication via individual bits/port

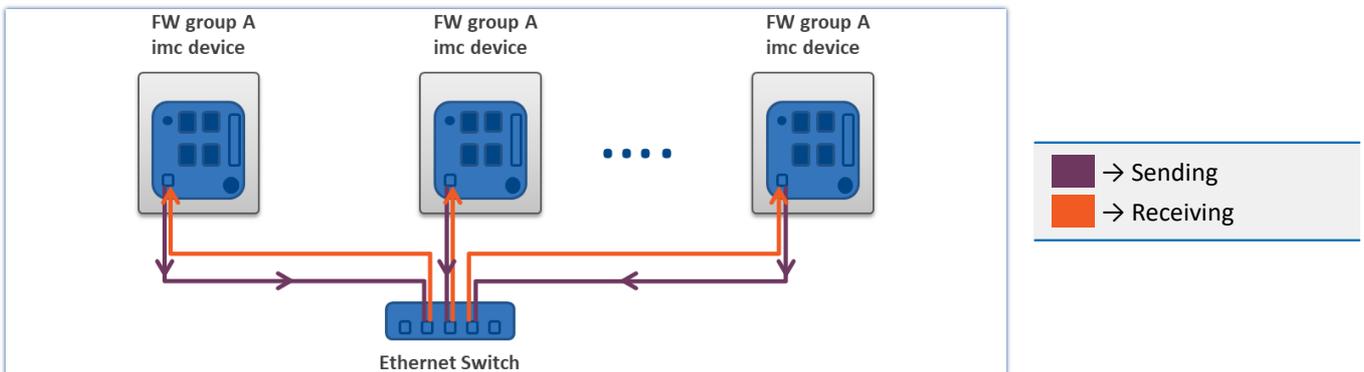
- Applies to device which are equipped with the appropriate module.

Using the digital outputs, control functions can be implemented. In this way, communication between the devices is also possible via an appropriate connection line. Refer to the respective device manuals for more information on connection.



Exchanging variables across devices via Ethernet

- Applicable to devices belonging to [firmware group A](#) ¹⁴⁶ (e.g. imc CRONOS, imc C-SERIES, ...).



Communication via individual bits - use of Ethernet bit variables

If the value of an Ethernet bit changes, this change is recognized and applied by all imc devices located in the same network. This means not only by the devices belonging to the experiment.

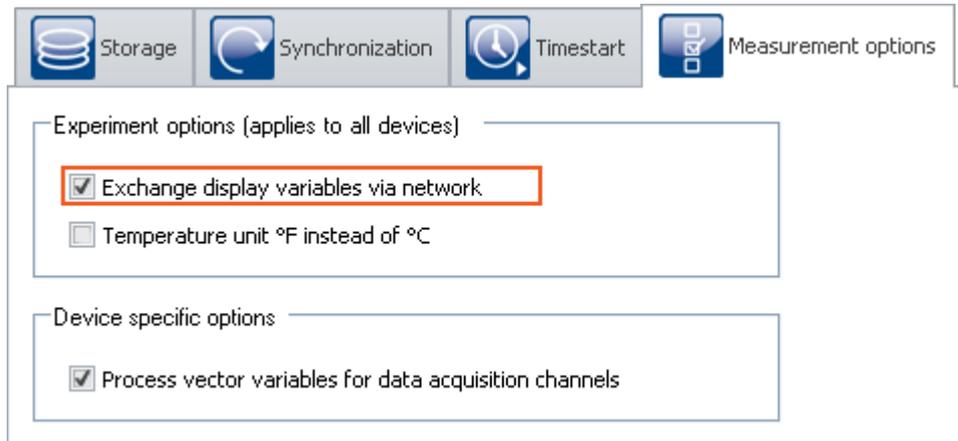
No additional settings are necessary.

Communication by means of variable - use of Display variables

If the value of Display variable changes, this change is recognized and applied by all imc devices located in the same network. This means not only by the devices belonging to the experiment.

Prerequisite:

- The measurement option "Devices" > "[Exchange display variables via network](#)²³⁶" is activated for the devices.



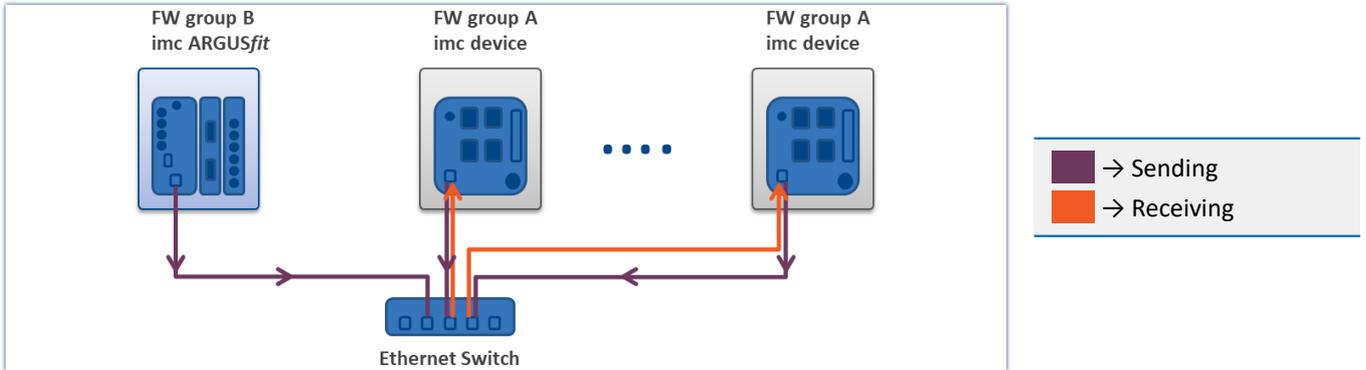
Activation for devices of firmware group A (z.B. imc CRONOS...)

! Notes

- The maximum sending-/reading rate is 10 ms. For this reason, any changes occurring at a higher speed (such as on channels sampled at high-speed) can not be transmitted.
- Checking and sending of the display variables burdens the measurement device's and the Ethernet's resources. For this reason, the function should only be activated if data exchanging between two devices is intended.

Exchanging variables via the Ethernet - from imc ARGUSfit to devices belonging to firmware group A

imc ARGUSfit devices **can send values** to the Display variables of imc devices belonging to firmware group A (e.g. imc CRONOS, imc C-SERIES, ...). However, **reception of values** from other devices is **not possible**.

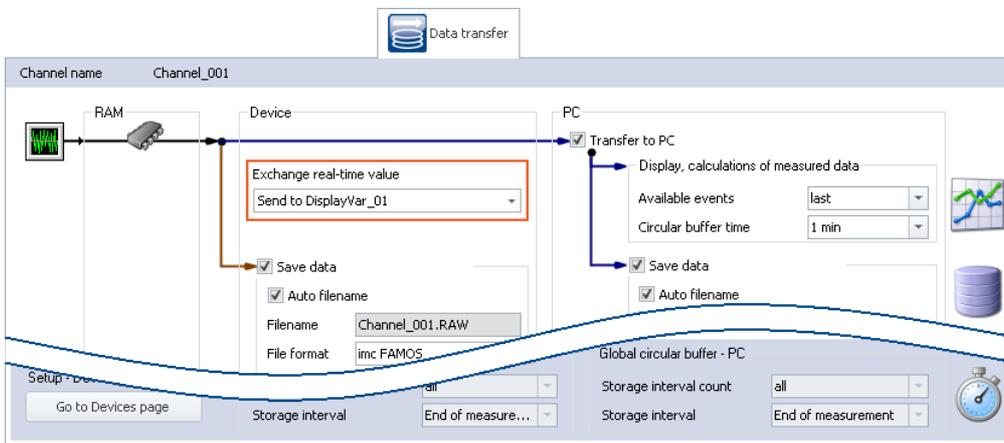


For each individual channel it is possible to select whether to send it to a Display variable.

If the value at the imc ARGUSfit channel changes, this change is recognized and applied by all devices of firmware group A which are located within the network and have the appropriate settings. This means not only by the devices belonging to the experiment.

Prerequisites:

- Firmware group A device: The measurement option "Devices" > "[Exchange display variables via network](#)"²³⁶ is activated for the devices - see also "[Use of the Display variables](#)"⁴⁸⁷"
- For the parameter "[Data transfer](#)"²⁶⁸ > "Exchange real-time value", a Display variable is to be selected for the channels to send.



Assigning an imc ARGUSfit channel to a Display variable

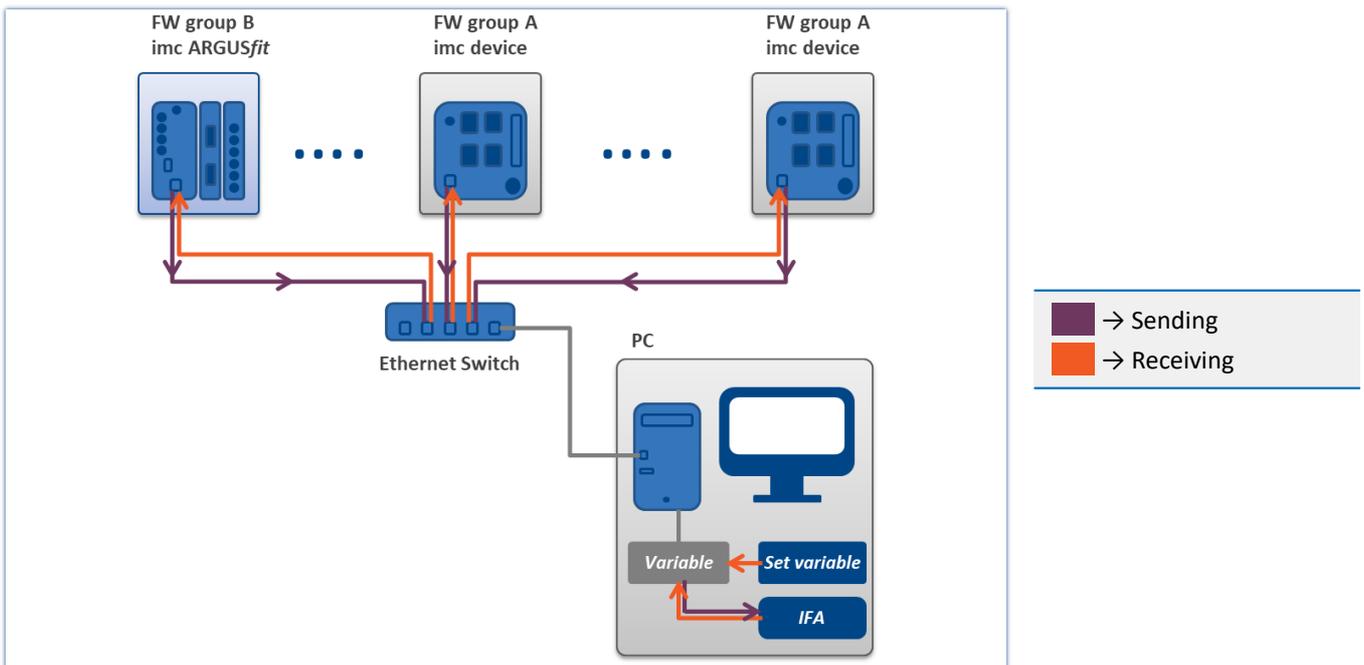
! Notes

- For each channel (maximum of 32), one Display variable can be selected as the target.
- The maximum sending-/reading rate is 10 ms. For this reason, any changes occurring at a higher speed (such as on channels sampled at high-speed) can not be transmitted.
- Always the last value detected by the device is sent. If any additional values have been accumulated in the device, they are ignored.
- Immediately upon sending the values, the action "Prepare" begins. Even if the actual measurement procedure has not yet begun.
- Sending the values burdens the Ethernet, therefore this function should only be activated if data exchange between to devices is intended.
- For imc EOS devices, this function is not available.

Exchanging values among devices by means of processing on a PC

By means of imc WAVE-functions, the values can be provided to and exchanged among the devices in various ways.

For example, this can be done either at a specific point in time by setting variables using commands such as "[Set variables](#)"^[1198], or continually by means of calculation operations in imc Inline FAMOS.



10 Setup - Advanced device functions

Here, certain special topics regarding "Setup" are presented.

The following chapters apply to devices of the [firmware group A](#) ¹⁴⁶ (imc DEVICES) - e.g. CRONOS family, C-SERIES, SPARTAN, BUSDAQ.

Chapter overview

Synopsis	Section
----------	---------

11 Data analysis and signal processing

This chapter describes the various data analysis and signal processing possibilities available.

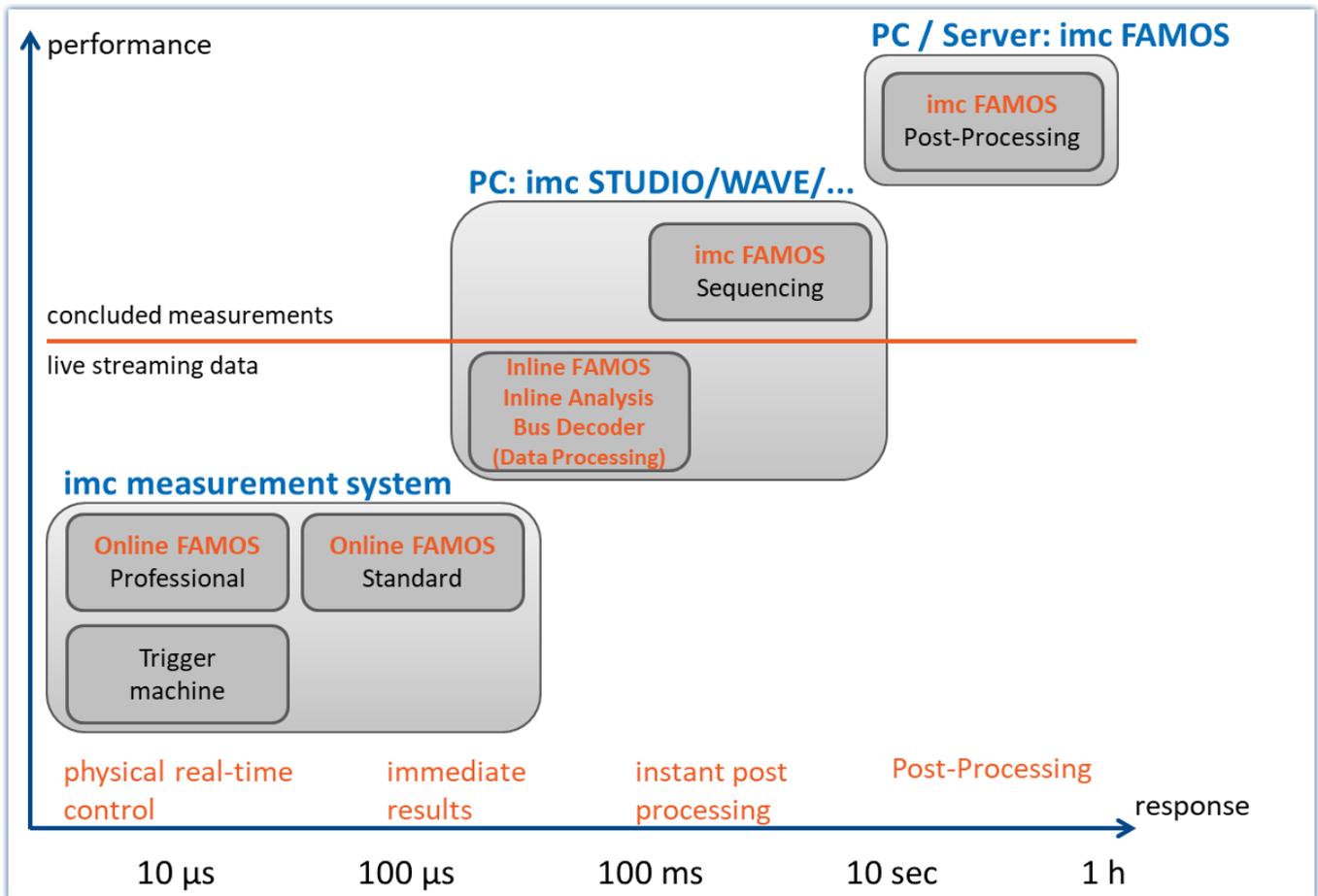
11.1 Overview of the tools

Tool	Description
imc Online FAMOS ⁴⁹⁸	imc Online FAMOS is a powerful extension for imc DAQ systems. It offers a variety of real-time functions for pre-processing and signal analysis. The mathematical analysis functions are executed on the signal analysis platform integrated in the measurement device. This means that analysis results are available immediately and also independently of the PC.
imc Inline FAMOS	Not available
Inline Analysis	Equals to the derived channels
imc FAMOS ⁷²¹	<p>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.</p> <p>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.</p> <p>In imc Online 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.</p> <p>For all applications having online functionality requirements, it is not adequate to wait until the end of the measurement.</p>
Bus Decoder	Not available
Powertrain Monitoring	Not available

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



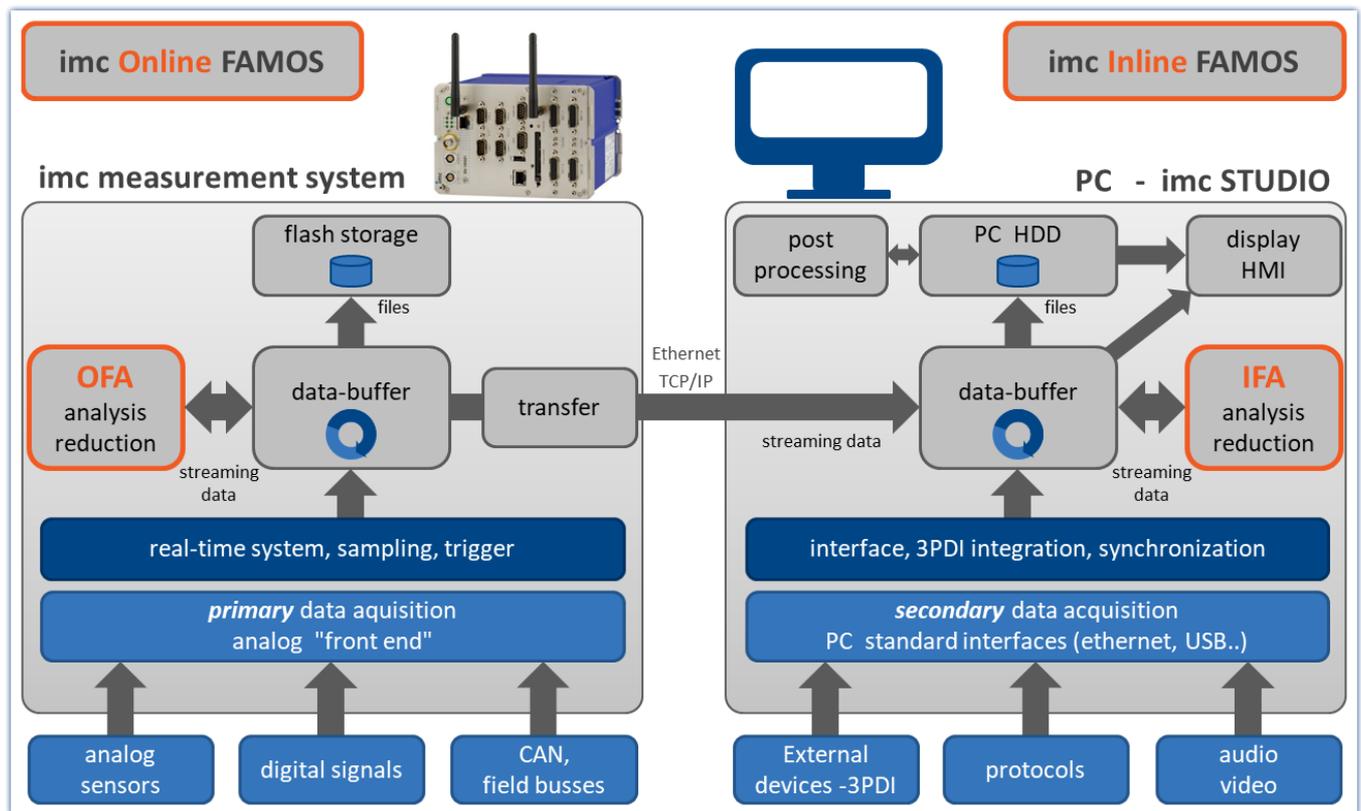
11.1.1 Comparing imc Online FAMOS with imc Inline FAMOS

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 <ul style="list-style-type: none"> • no calculations across multiple devices possible • stand-alone capability 	Processing occurs on the PC, and not where the data are captured. <ul style="list-style-type: none"> • 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: <ul style="list-style-type: none"> • 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](#)"⁴⁹⁶.

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 Online 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 Online FAMOS functions, then this is comparable to an analog circuit. Thus, imc Online FAMOS can be viewed as a substitute for a freely configurable analog pre-processing circuit.

What can imc Online FAMOS not do?

- It is impossible to formulate any processes which do **not** utilize **continuously streaming data**.
- imc Online FAMOS does not replace a synthesizer! For output rates of up to 10 kHz, the [Synchronous Task](#)⁵⁰³ can be used. For this purpose, imc Online FAMOS Professional is necessary.
- imc Online 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. If your application requires definite response times, then these can be guaranteed with an [Synchronous Task](#)⁵⁰³ of imc Online FAMOS Professional.
- 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 Online FAMOS, the bit count must be set to 31 bits following import of the DBC file.

11.1.1.1 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.

11.1.2 imc Online FAMOS - System requirements and limitations

imc Online FAMOS and imc Online FAMOS Professional

With the exception of imc BUSLOG, all devices which support imc WAVE are also able to run imc Online FAMOS (or imc Online FAMOS Professional).

Maximum amount of variables in imc Online FAMOS:

Internally, imc Online FAMOS administers up to 999 variables (single values + virtual channels). However, be aware that only a maximum of 512 channels can be administered per device.

**Note**

If more than 500 variables are renamed when performing editing in the imc Online FAMOS Assistant before the Assistant is closed, an administration error occurs. This effect does not occur if the Assistant is closed and then later reopened before more than 500 variables were altered.

When imc Online FAMOS is Available?

imc Online FAMOS and imc Online FAMOS Professional can be accessed as an option. It must be enabled for each device individually. If the device is ordered with imc Online FAMOS or imc Online FAMOS Professional, it will be delivered already enabled.

If you wish to upgrade a device with imc Online FAMOS or imc Online FAMOS Professional subsequent to delivery, contact our [tech support](#)  for more details.

11.2 imc Online FAMOS

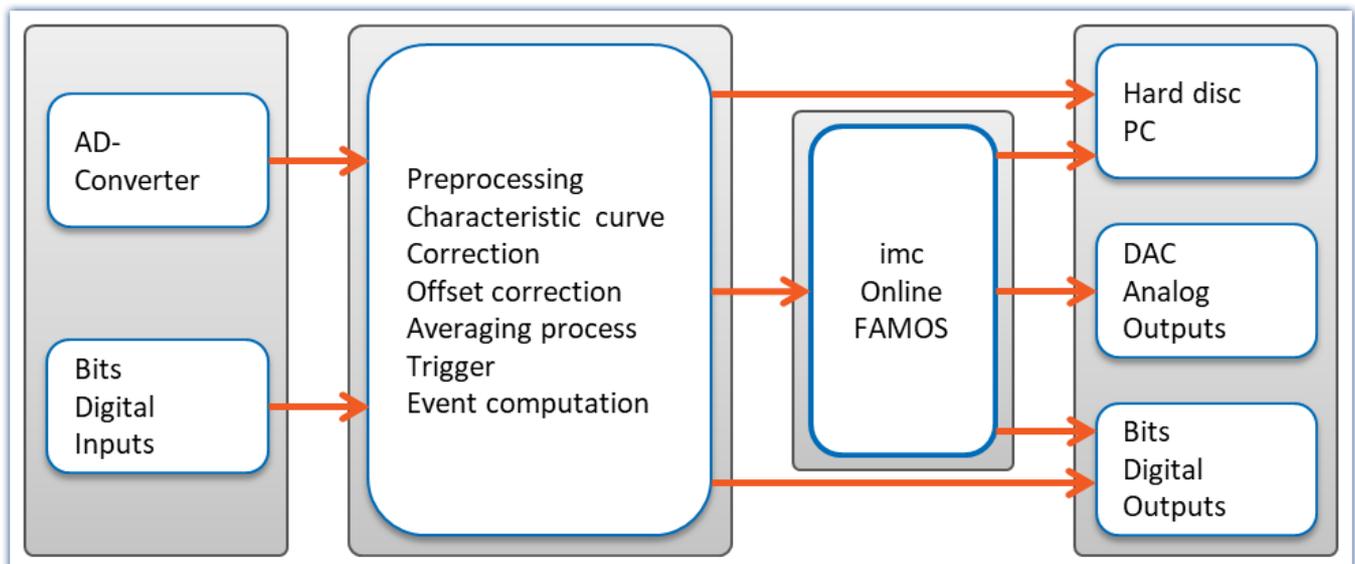
imc Online FAMOS

imc Online FAMOS is a powerful extension for imc DAQ systems. It offers a variety of real-time functions for pre-processing and signal analysis. The mathematical analysis functions are executed on the signal analysis platform integrated in the measurement device. This means that analysis results are available immediately and also independently of the PC. Such pre-processing can also yield significant data reduction and thus reduce the amount of data to be exchanged between the DAQ system and the PC.

The results are available in imc WAVE as virtual channels. Virtual channels contain data calculated on the basis of existing input channels. This may be, for example, an averaged input channel, or the difference between the signal values from two input channels.

The virtual channels can be redirected to DAC (digital-analog-converter) or DIO (digital input/output, bits) outputs on the device. For example, you compute the difference between two analog input signals (digitally), and convert the result back to an analog signal at the DAC output. The DIO outputs can be used, for example, to set a digital bit whenever a certain signal level has been reached.

The following diagram shows how data stream through the device and which of them can be accessed by imc Online FAMOS:



Analog/ digital conversion of analog input data and/ or acquisition of digital input data is carried out on the unit's main board. All of these input data are usually subjected to a certain amount of preprocessing, for example, characteristic and offset correction, event and trigger computations etc. The trigger system can also be used to directly set output bits.

Instead of direct transfer to the PC, the data streams with measured channels signals are rerouted through imc Online FAMOS. These are imc Online FAMOS's inputs. Results calculated by imc Online FAMOS are mostly virtual channels which are passed on to the PC like other input channels. Additionally, note that even digital output bits and the DACs can be outputted as virtual channels.

11.2.1 Overview

imc Online FAMOS works with streams of binary data.

Take a simple example where the difference between two channels is computed:

$$\mathbf{Difference = Input1 - Input2}$$

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

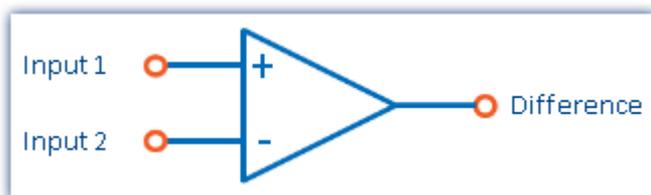
Compute difference

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, $\mathbf{Difference = Input1 - Input2}$, 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):

$$\mathbf{DAC1 = Input1 - Input2}$$

The following steps are executed internally:

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 `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.

11.2.1.1 What happens when imc Online FAMOS overflows?

General notes on data overflow in software

imc Online FAMOS operates without synchronization to the data capture. This means that imc Online FAMOS works through its program in an uninterrupted loop.

imc Online 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 Online FAMOS to do.

If the online calculation is so complex that new samples accumulate by the time the calculation is done, then imc Online FAMOS enters the new data into the next calculation iteration. If the average number of new samples remains equal, there is no overload.

imc Online 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 Online 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.

The following applies to imc Online FAMOS: For brief measurements, of course, this is tolerable if the [RAM-buffer duration](#)^[322] is adequate. But if the accumulated data exceeds the RAM buffer depth, the software will announce a data overflow!

Special case applicable to imc Online FAMOS: Overflow in the synchronous task

When using the [Synchronous Task](#)^[503], it is necessary to additionally ensure that the commands are performed within the allocated time interval. If this does not succeed, the device announces the overflow with `LED_06` and activation of the beeper. When the function `SyncOverload` is called in the synchronous task, neither `LED_06` nor the beeper are activated. Instead, the overflow can be indicated using a variable.



Example

```
OnSyncTask( 0.01 )
  if Greater( pv.Channel_002, 5)
    Virt_Bit01 = 1
    ; More functions...
  else
    Virt_Bit02 = 1
    ; More functions...
  end
  ; Upon overflow, the virtual channel changes from 0 to 10
  pv.SyncOverloaded = SyncOverload( 10 )
End
```

11.2.1.2 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 Online 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 Online 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](#)"⁵¹⁸.

11.2.1.2.1 Transforming to imc Online FAMOS with/without Control commands

By default, imc Online FAMOS starts without control commands.

Activate the Control Commands via the [context menu](#)⁵¹¹ in the Editor or the menu item "Extra" > "With control commands" (via the menu applies only to imc Online FAMOS):

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
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*".

11.2.1.2.2 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 Online FAMOS function reference](#)^[542]".

11.2.1.3 imc Online FAMOS Professional

imc Online FAMOS Professional is the package for effective use of the process vector and for operation of the measurement device on the test station. By this means, it is possible to perform monitoring, and open- and closed loop control functions. The package also provides a significant performance enhancement for all purely measurement engineering applications (data-logger operation) in calculating virtual channels.

With imc Online FAMOS Professional you have

- [Performance enhancement](#)^[503]
- [Synchronous Tasks](#)^[503] including accessories
- Direct access to the [process vector](#)^[741]
- [PID-Controller](#)^[532]

11.2.1.3.1 Performance enhancement

imc Online FAMOS Professional serves to accelerate online computations. Thus it increases the system's maximum possible throughput without causing data overflow in long-term measurements.

! Note

Internal information

The accuracy is not reduced by that feature. The increased speed is achieved by as many functions as possible using the signal analysis platform's internal memory. Accessing commands in such memory space can happen much more quickly.

However, this memory is much smaller, so that it's usually not possible to keep all program components within it. By means of the imc Online FAMOS Professional, the compiler shifts the functions into the internal memory until it's full. Therefore those functions whose gain on speed is the highest will be preferred.

In general all functions are considered. But memory intensive functions will be preferred, because they speed up most.

Examples

- FFT: 1,5 –2,5x
- Basic calculations: 2x
- Digital filters: 3-4x

11.2.1.3.2 Synchronous tasks

Along with the control command `OnTimer`, imc Online FAMOS Professional **with control commands** includes a control command for deterministic reactions. At a specified clock rate, imc Online FAMOS' otherwise asynchronous evaluation process is interrupted by a higher priority task. During this interrupt, the command lines entered for the synchronous task are executed.

```

1  [+ OnInitAll...
3
4  [+ OnAlways...
6
7  [- OnSyncTask(0.1)
8     Virt_Bit01 = 1 - Virt_Bit01
9     DOUT001_Bit01 = Virt_Bit01
10  [- End
11
12 [+ OnTriggerStart(Trigger_48)...
14

```

- Enhancement of OnTimer
- True interrupt handler
- Precise timing
- Access to input channels via the process vector
- Loop control

The user can configure up to **5** synchronous tasks. The cycle times extend from **100µs up to 1s** in steps of 1-2-5. With Synchronous Tasks, there is an effective way to access the current measurement values with the help of the process vector.

By this means, monitoring, and open- and closed-loop control functions can be realized. In particular, the switch/case control commands can be used to achieve state control.

In Synchronous Tasks, not all of imc Online FAMOS' functions are allowed. This pertains to functions, which use a collection of samples like [FFT](#), [Mean](#), [RMS](#) etc. But comparison functions, basic math functions and control functions are possible.

In imc Online FAMOS, *closed-loop control* is only possible in a Synchronous Task: two-point controls and PID-controls are included with the Synchronous Task functions. The PID-controls are provided with a variety of additional functions, e.g. controller block, output limiting, and change of input parameters during operation.

It is possible to use the vector functions [VMax](#), [VMean](#), [VMin](#), [VSum](#), [VRMS](#), [VIsAnyGreater](#), [VValueAtXValue](#), [VXValueOfMax](#), [VXvalueOfMin](#), [VXValueWithYValue](#) in synchronous tasks. In the synchronous task, the functions named can be used on vectors generated using the function [VectorFromFile](#). Also supported are functions [Monoflop](#), [MonoflopRT](#), [JKFlipFlop](#) and [RSFlipFlop](#).

Restrictions

LEDs can't be controlled inside a synchronous task. Instead a virtual bit has to be controlled inside the [SyncTask](#). Than, that virtual bit set the LED in [OnAlways](#).

11.2.2 Operation

11.2.2.1 Opening the editor

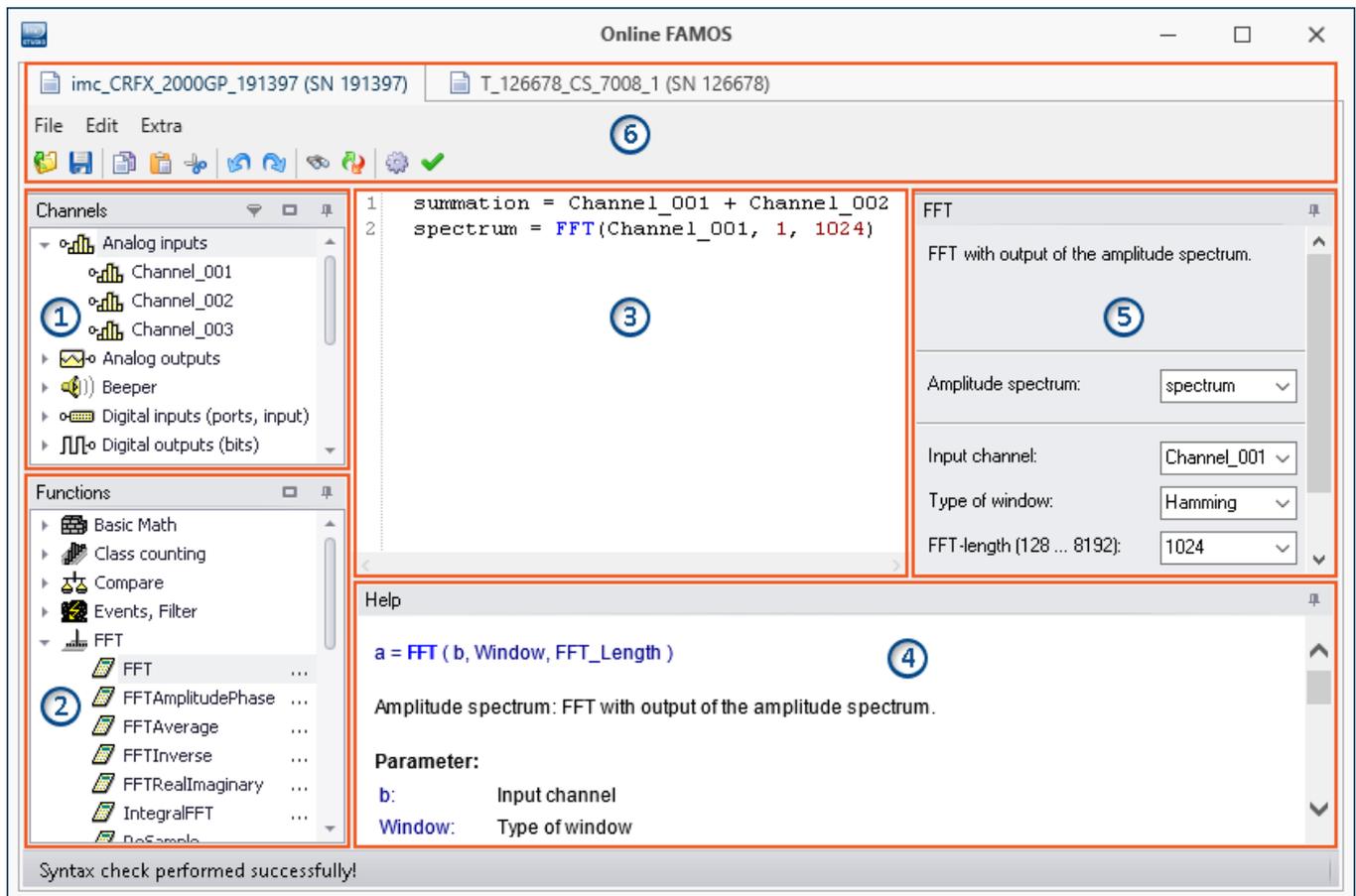
imc Online FAMOS

Select "imc Online FAMOS" from the Ribbon "*Home*" (or "*Setup-Configuration*"). The imc Online FAMOS Editor is displayed in a dialog.

Note

If the menu item imc Online FAMOS is not accessible, check in your calibration certificate whether your device comes with this optional feature. For the software to detect whether imc Online FAMOS is available, the device must have been connected at least once.

11.2.2.2 User interface



The window is subdivided into six regions:

1. [Variables list](#) ⁵⁰⁶
2. [Function list](#) ⁵⁰⁶
3. [Editor](#) ⁵⁰⁶ (text box) for the calculation operations
4. [Help](#) ⁵⁰⁷
5. [Formula assistant](#) ⁵⁰⁷
6. [Menu and actions](#) ⁵⁰⁹ (only in imc Online FAMOS)

Region 1: Variables List

Here is the list of all available variables supported (imc Online FAMOS: device 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		Digital outputs (bits) (of all DIO-ports)
	Digital inputs (ports)		Analog outputs (DAC)
	Digital outputs (ports)		housing LED's
	calculated virtual channels		Virtual bits
	calculated local channels		Ethernet bits
	calculated local single value variables		Display variables
	Process vector variables ⁷⁴¹		Beeper (miniature speaker)
	Digital inputs (bits) (of all DIO-ports set for "Bit-input")		Trigger which starts an analog channel. Only those triggers are listed to which at least one channel is assigned.

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.

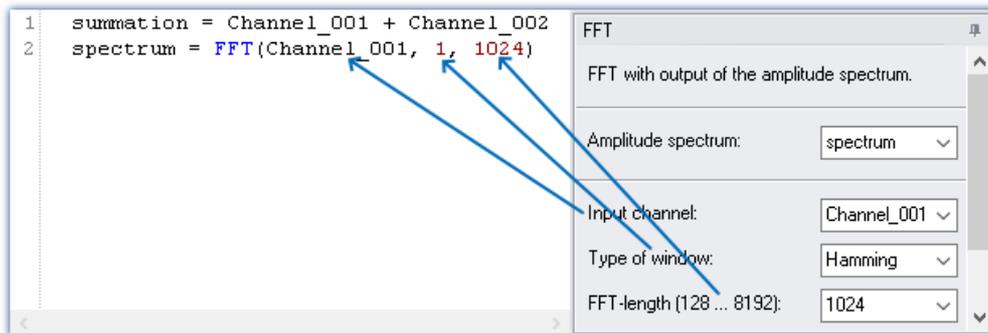
Region 4: Help

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.

11.2.2.3 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.

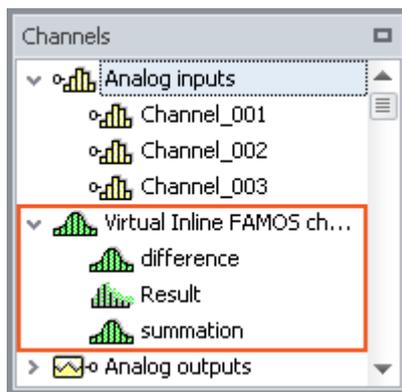
Adding a variable or function

- 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*" (📄).

imc Online FAMOS - Close

Close imc Online FAMOS now. In the table of channels in the Setup, the newly defined channels appear as additional **virtual channels**.

11.2.2.4 Menu

All functions can be accessed via the "[context menu](#)⁵¹¹".

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!
 New	-	This clears the contents of the editor and deletes the virtual channels from the Variables List.

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 ⁵²⁷	(F5)	Opens the Properties window. There, the properties of virtual channels can be edited subsequently.
 Syntax check	(F6)	<p>The source text is checked for errors.</p> <p>If an error occurs, a notification appears in the status bar! The faulty spot is highlighted in the Editor.</p> <p>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.</p>

Extras - Menu

Menu item	Description
 Options	It is possible to deactivate automatic LED-flashing. (See LED-flashing during measurement ⁵³⁹)
Supplemental files	See " Supplemental files " ⁵³⁷ - Menu item is only available for devices of firmware group B ¹⁴⁶ .
 Directories	In imc WAVE, the default folder can not be changed. See " Supplemental files " ⁵³⁷ .
 imc Online FAMOS with control commands	Activate control commands ⁵⁰¹ .
 imc Online FAMOS without control commands	Deactivate control commands ⁵⁰¹ .

11.2.2.5 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)	<p>The source text is checked for errors.</p> <p>If an error occurs, a notification appears in the status bar! The faulty spot is highlighted in the Editor.</p> <p>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.</p>
 With control commands	-	Deactivates/activates the Control Commands  .
 Properties	(F5)	Opens the Properties window. There, the properties of virtual channels can be edited subsequently (see Virtual channels  .
 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.
 Collapse foldings	(Ctrl + J)	All folds are folded.
 Unfold foldings	(Ctrl+Shift + J)	All folds are folded out.
 Autocomplete	(F7)	<p>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.</p> <ul style="list-style-type: none"> • 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

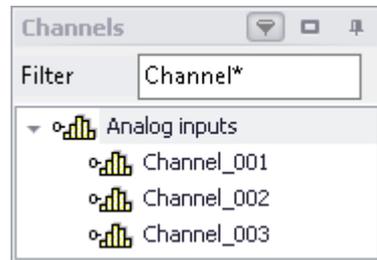
Right-clicking in the Variables List calls the following context menu:

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 Wildcards

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.

11.2.3 Variables and syntax

11.2.3.1 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
```

11.2.3.2 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".

11.2.3.3 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 DAC2 = Channel_001 / 2	You can assign constants or data streams to a DAC. With data streams, the most recent valid value appears at the DAC.
------------------------------------	---

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 BEEP1 = 0 BEEP1 = STRI(Channel_001, -5, 5)	Assigning 1 (TRUE) to the beeper turns it on and assigning 0 (FALSE) to it turns it off.
--	--

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 "[LED-flashing during measurement](#)"⁵³⁹)

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 Online 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 Online 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)	

11.2.3.4 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)
```

11.2.3.5 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 *)
```

11.2.3.6 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](#).

11.2.4 Variables and syntax with Control commands

The following descriptions apply to **imc Online 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](#)"⁵⁰¹.

11.2.4.1 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](#)⁵²⁵).

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 Online FAMOS: with 24-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

```

11.2.4.2 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 VirtBit_01 > 0 AND VirtBit_02 = 0
or
Virt_Bit01 = DisplayVar_01 < -10 OR DisplayVar_01 > 10

```

11.2.4.3 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

```

11.2.4.4 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 Online 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 Online FAMOS	imc Online 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 Online FAMOS and placed before the query, and the single value generated is then queried:

Coded in imc Online FAMOS	imc Online 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 Online FAMOS	imc Online 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 Online 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 Online 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 Online FAMOS accepted this, but this could lead to incorrect queries. The following construct leads to an error message in current imc Online FAMOS versions:

```
Bools= VectorFromFile("Bools.dat")
Switch_B= Bools[2]
...
If Switch_B ; -> Error message
...

```

Therefore `If` queries in current imc Online 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 Online FAMOS code must be changed according to the last example.

11.2.4.5 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 )
    switch Status
        ; OnTimer-Block for Timer 1
        ; 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
        end
        ; finish condition
        ; 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

```

11.2.5 Types of variables

The following is a list of the usual variable types which you can define in imc Online 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>float Var2= 0.0 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 int pv.Var4= 0xFFFF ; hex</code>	<code>int pv.Variable = 0</code>	OnInitAll
Single value as Float	<code>float pv.Var2= 0.0 pv.Var3= 0 pv.Var4= 0xFFFF ; hex</code>	<code>pv.VariableA = 0 float pv.VariableB = 0</code>	OnInitAll

Local texts ()

Description	Example	Syntax	Definition
Text	<code>Text = "Hello"</code>	various functions	

11.2.5.1 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
```

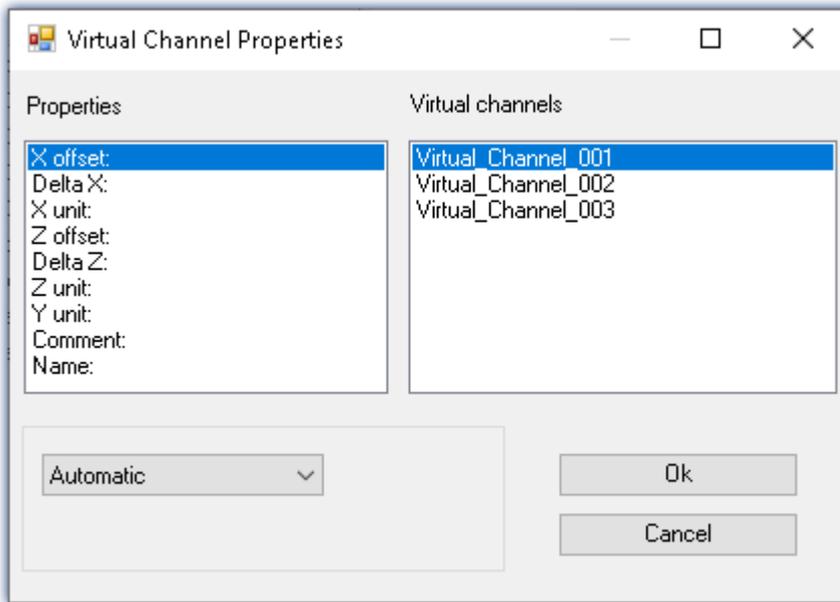
- The vector elements are initialized at 0.
- The vector length of `sv` must be selected appropriately. Make sure to use only possible vector indexes, especially for variable vector indexes.
`sv[1]...sv[vector length]`
- Local vectors can be used in synchronous tasks, too.
- Elements of local vectors 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.

11.2.6 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](#)^[511] in the Editor, or via the menu item "Edit" > "Properties" (via the menu item applies only to imc Online FAMOS):

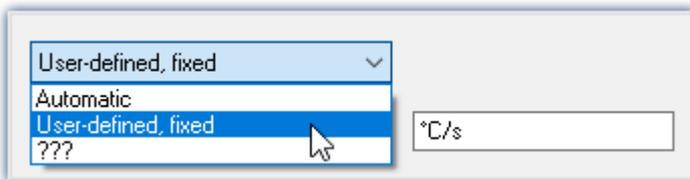
Menu item	Description
 Properties	Opens the Properties window. There, the properties of virtual channels can be edited subsequently (see " Virtual channels " ^[513]).



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.

11.2.7 Calculation examples

11.2.7.1 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)
```

11.2.7.2 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.

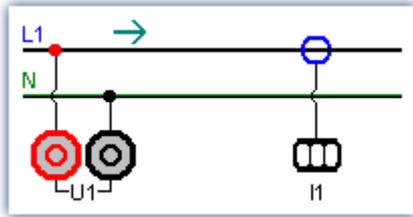
The assigned name, `_I_Channel`, begins with an underline, identifying it as an internal variable. This isn't strictly necessary, but it helps to conserve memory.

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.

11.2.7.3 Power measurements

Here follows the description for the Single-phase, Two-phase- and Three-phase-power measurement.

11.2.7.3.1 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.

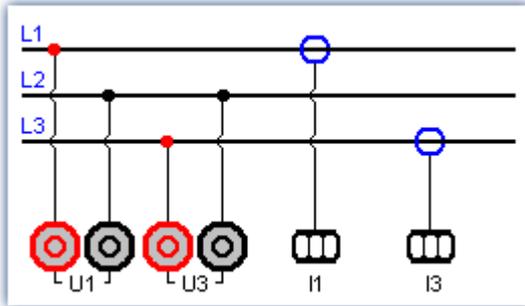
Power Factor

$$\cos(\varphi) = \frac{P}{S}$$

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.

11.2.7.3.2 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.

$$S = \frac{\sqrt{3}}{2} \int (U_{rms 1} \cdot I_{rms 1} + U_{rms 3} \cdot I_{rms 3}) dt$$

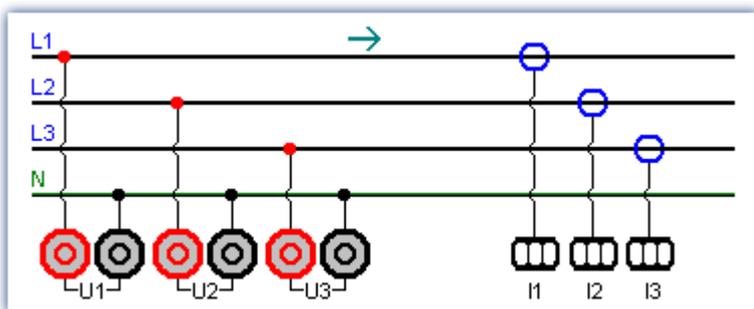
It is required that the symmetry of the supply circuit (three equal voltages with phase shifts of 120°) be stable. The root-mean-squares are calculated according to the algorithm under the operation effective value (root-mean-square).

Reactive power and Power Factor

See [Single-phase power measurement](#) ⁵³⁰

11.2.7.3.3 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](#) ⁵³⁰.



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

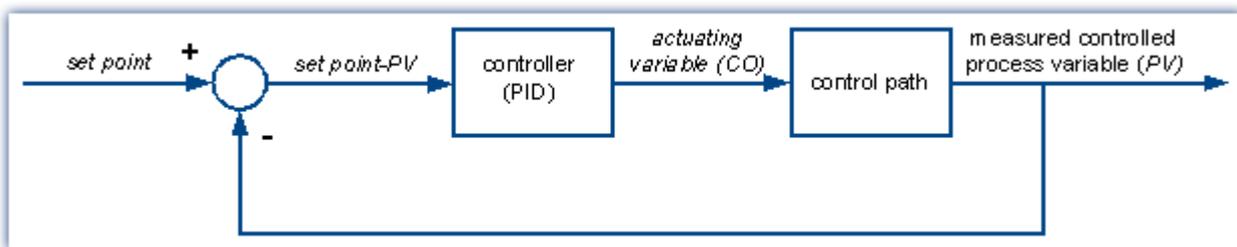
See [Single-phase power measurement](#) 

11.2.7.4 PID-controller

The controller functions are only available if imc Online FAMOS Professional is enabled for the device.

In a controller, a process value (PV) measured by sensors is continually compared with a specified setpoint value. If a system deviation (setpoint – PV) occurs, the controller generates a controller output (CO) which affects the system's energy flow in such a way to cause the process value to approach the setpoint. Then the system remains unaffected by the controller until the setpoint is changed or a disturbance occurs. Whether and how the controlled variable/ process value reaches the setpoint depends on the controller. Depending on the quantity to be controlled and the desired control response, a variety of different types of controller can be employed.

The following is a simplified sketch of a control loop:



Simplified sketch of a control loop

A pure *P-controller* is not able to completely correct disturbances, and can cause oscillation if the P-component (KP) is too large. It may react immediately to disturbances or changed setpoints, but still operates in exactly due to the remaining system deviation.

An *I-controller* (integrator), on the other hand, can completely correct a system deviation, but it does so much more slowly and also has a tendency to lead to oscillations.

A *D-controller* working alone is useless due to its differentiating response, since it only puts out a controller output when the system deviation changes, and thus doesn't correct system deviation as such. It is used in conjunction with other controllers in order to accelerate the correction of system deviations.

A *PI-controller* unites the advantages of a P- with an I-controller, since the P-component quickly reacts to system deviations and the I-components ensures that no system deviation remains. A *PD-controller* is very quick in providing adjustment thanks to its P- and D-components, but it leaves a persistent system deviation.

A *PID-controller* is constructed by combining the three basic elements P-controller, I-controller and D-controller. It possesses the positive properties of the individual controllers, i.e. it adjusts quickly, leaves no deviation and reaches the setpoint without overshooting it. However, it has the drawback that the D-component amplifies high-frequency disturbance signals. To prevent this, there is the possibility of specifying an upper cutoff frequency for the D-component. This type of controller is known as continuously acting, since it reacts to any change. Because each component or proportional coefficient like P-component (KP), I-component (KI) and D-component (KD) can be defined separately, then all types of controllers from the simple P-controller to the PID-controller can be achieved. The controller output CO then results from the addition of the three outputs of the P-, I- and D-controllers:

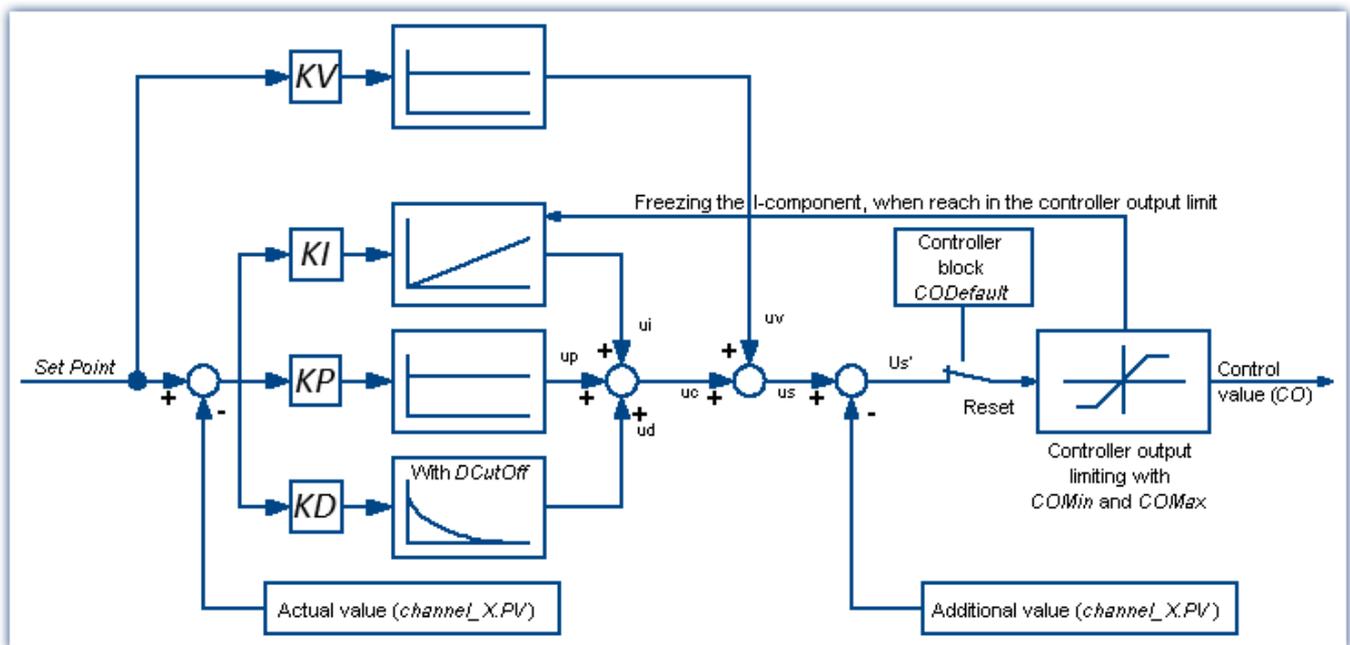
$$CO = KP \cdot (SetPoint - PV) + KI \cdot \int (SetPoint - PV) + KD \cdot (SetPoint - PV)'$$

11.2.7.4.1 The PID-controller in imc Online FAMOS Professional

The constants KP, KI and KD are specified for a continuous (non-discrete) controller and can be changed online at any time during the control process. The PID-controller is computed in the *synchronous task* with the selected cycle time, and the process value PV (any input channel) is provided to it in a process vector.

The entire controller structure in imc Online FAMOS consists, besides the PID-controller, of:

- an input control element (P-element with proportional coefficient KV),
- an additional summation point, via which a further measurement channel can be inputted in the form of an extra value,
- a controller block with adjustable default value and controller output limiting. The controller output limiting also comes with tracking of the I-component when the control variable limit has been reached. Toward this end, the I-component is frozen until the control variable limit is canceled.
- Additionally, it's possible to set an upper cutoff frequency (DCutOff) for the D-component, in order to avoid amplifying noise signals.



This as well as all of the PID-controller's other configuration possibilities are described below and can also be found in the imc Online FAMOS help dialog.

Parameter	Description
KP	<p>P-component</p> <p>Use in the controller equation: $CO = KP * (SetPoint - PV) + \dots$ with PV = process value, CO = control output, SetPoint = setpoint KP's unit is: [unit for CO] / [unit for PV]</p>
KI	<p>I-component</p> <p>Use in the controller equation: $CO = \dots + KI * \text{Integral} (SetPoint - PV) + \dots$ with PV= process value, CO = control output, SetPoint = setpoint KI's unit is: [unit for CO] / (s * [unit for PV])</p> <p>When the function .Calc is called, one integration step is carried out. The synchronous task's cycle time is used as the basis.</p>
KD	<p>D-component</p> <p>Use in the controller equation: $CO = \dots + KD * \text{differentiator} (SetPoint - PV) + \dots$ with PV= process value, CO = control output, SetPoint = setpoint KD's unit is: s * [unit for CO] / [unit for PV]</p> <p>When DCutOff is not equal to 0, be sure that the differentiator is band-limited.</p>
KV	<p>Factor for the input control</p> <p>Use in the controller equation: $CO = \dots + KV * SetPoint + \dots$ with CO = control output, SetPoint = setpoint KV's unit is: [unit for CO] / [unit for PV]</p>
SetPoint	<p>setpoint</p> <p>The setpoint is used when the .Calc function is called.</p>
Reset	<p>Controller block</p> <p>if = 0, the controller is working normally. if = 1, the controller is blocked.</p> <p>When the controller is blocked, its output is set to the default output value .CODEfault (generally = 0). Also, when the controller is blocked, the integrator is reset. The integral is thus kept at zero.</p> <p>If .Reset is changed, it only takes effect the next time the function .Calc is called.</p>
CODEfault	<p>Default for the controller output</p> <p>When a controller block is in effect, the controller's output is set to this default value. This value is generally = 0. This value should lie within the range [COMin...COMax].</p>
COMin	<p>Lower limit for the setpoint</p> <p>The controller output is limited to the range [COMin...COMax].</p> <p>When controller output limiting is in effect, the PID-controller's integrator is frozen at its current value.</p>
COMax	<p>Upper limit for the setpoint</p> <p>The controller output is limited to the range [COMin...COMax].</p> <p>When controller output limiting is in effect, the PID-controller's integrator is frozen at its current value.</p>

Parameter	Description
Xinput	<p>Extra input, fed into the controller after the P-, I-, D-components.</p> <p>Use in the controller equation: $CO = \dots - Xinput + \dots$</p> <p>With CO = controller output.</p> <p>Xinput's unit is: [unit for CO]</p>
DcutOff	<p>Upper cutoff frequency (in Hz) for the D-component</p> <p>To prevent amplifying the noise, the D-component is frequency-limited. Thus it is then only effective up to a certain frequency .DCutOff. While at lower frequency components the gain is proportional to the frequency, at higher frequency components the gain is constant. This constitutes a 1st order high-pass filter whose cutoff frequency is located just at .DCutOff.</p> <p>If this value is set to 0, the frequency limiting is deactivated. In that case, a digital differentiator is computed.</p> <p>Use in the controller equation:</p> <p>for $DCutOff \neq 0$: $CO = \dots + KD * \text{high-pass}(\text{SetPoint} - PV) + \dots$</p> <p>for $DCutOff = 0$: $CO = \dots + KD * \text{differentiator}(\text{SetPoint} - PV) + \dots$</p>
CO	<p>controller output</p> <p>The controller output can be queried as .CO. It is also already returned by the function .Calc.</p>
PV	<p>process value</p> <p>The process value is the instantaneous value measured for the controlled variable and is queried as .PV. It is already set when the function .Calc is called and therefore generally needs not be set explicitly.</p>

The controller is computed in imc Online FAMOS *with Control Commands* in the command OnSyncTask with the function .Calc. It must be initialized previously, since it initially contains no values, but already needs values upon the first cycle pass in order to compute the PID-controller.

Initialization is performed in the *OnInitAll*-block with the values specified for KP, KI and KD. All other elements are initialized as 0.0, except DCutOff, which is set to approx. $0.1 / [\text{controller cycle interval}]$.

```
Controller = CtPID( P_component, I_component, D_component )
Controller output = CtPID.Calc( Process Value)
```



Example

```

OnInitAll
    EngineController = CtPID( 20.0, 0.5, 0 ) ; P, I, D
    EngineController.SetPoint = 6000          ; set rpm to 6000
    EngineController.COMin = 0.0             ; min and min range must be defined
    EngineController.COMax = 10000.0
    ; DplayVar_11 - 13 pass KP, KI and KD values if Virt_Bit03 is set by the user.
End
OnTriggerMeasure( BaseTrigger ) ; when the measurement is running
    If Virt_Bit02 = 1 ; take new set point from display variable
        EngineController.SetPoint = DisplayVar_01
        Virt_Bit02 = 0
    End
    If Virt_Bit03 = 1 ; Read K-components from display variable
        EngineController.KP = DisplayVar_11
        EngineController.KI = DisplayVar_12
        EngineController.KD = DisplayVar_13
        Virt_Bit03 = 0
    End
End
OnTriggerStop()
    EngineController.SetPoint = 0
End
OnSyncTask( 0.1 )
    ; controller output scaled for DAC
    DAC_VoltageEngine = EngineController.Calc( pv.Speed )

    If Virt_Bit01 = 1 ; reset controller
        EngineController.Reset = 1
        Virt_Bit01 = 0
    End
End

```

The `.Calc` function calculates exactly one controller step. For the currently transferred process value, the new controller output is determined as the return value. The return value is then typically outputted at a DAC. The process value is subsequently available `.PV`, the controller output in `.CO`.

If controller parameters are changed, then these changes only take effect upon the next call of the function `.Calc`, because then and only then is a new controller output computed.

The function `.Calc()` is only allowed in synchronous tasks. The synchronous task's cycle interval is used to discretize the controller.

11.2.7.4.2 Two-position controller

The two-position controller (discrete controller) is, like the PID-controller, computed in the synchronous task with the function `.Calc`. However, it must previously be initialized in the `OnInitAll`-block by creating the necessary controller structure along with it. In the process, the controller's setpoint `.SetPoint` is set = 0, and the controller output `.CO` is set = 0.

The controller's method of functioning: It only outputs specific controller outputs when pre-defined conditions are met. 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.

```
Controller = CtTwoPos(Hysteresis, Output_Inversion )
```



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.Temperatur_01 ) ; set controller output
  ...
  If VirtBit_01 <> 0
    Thermostat.SetPoint = 22.0 ; change setpoint
    VirtBit_01 = 0
  End
End

```

The hysteresis is the distance between the upper and lower reversal points. It should be > 0. The two reversal points lie symmetrically around the setpoint. If, for instance, the setpoint is = 20 and the hysteresis = 2, then the resulting reversal points are 21 and 19.

11.2.8 Information and tips

11.2.8.1 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. 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.

11.2.8.2 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 Online 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)
```

11.2.8.3 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*".

11.2.8.4 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 Online 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
RowCl	Axis arrangement: X-axis is target class
RowAmplitude	Axis arrangement: X-axis is amplitude
RowClStart	Axis arrangement: X-axis is start class
RowMeanValue	Axis arrangement: X-axis is mean value
UnitCl	Result unit: classes
UnitInput	Result unit: as input channel
EndClOpen	Open-ended outer bins
EndClClosed	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>.

11.2.8.5 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!

11.2.8.6 LED-flashing during measurement

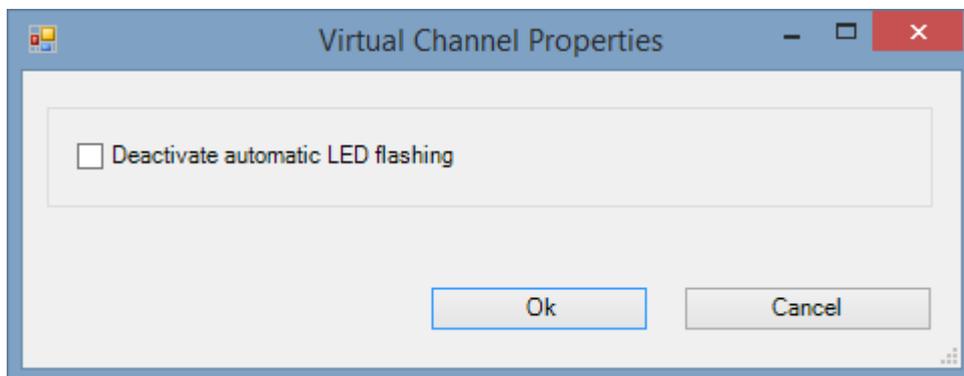
During a running measurement, LED 6 flashes at a 1-second rhythm. This provides a simple visual indication of whether the measurement is running.

LED 6 does not flash,

- if it is used in the imc Online FAMOS source code,
- if flashing is deactivated in the Options,
- if imc Online FAMOS is disabled.

Deactivating flashing in the Options

- Open the imc Online FAMOS options: Menu "Extra" > "Options"
- Activate the option: "Deactivate automatic LED flashing"



11.2.8.7 Using an external program editor

It's possible to enter the imc Online FAMOS source text using an external program editor (e.g. Notepad++, ...).

- Open the imc Online FAMOS options: Menu "Extra" > "Options"
- Activate the option: "Edit imc Online FAMOS source text in the external program editor?"
- Select the editor

Open the external editor by menu "Edit" > "Start External Editor" or push function key "F4".

Exchange of the imc Online FAMOS source text is performed by means of a temp file. This file is deleted as soon as the external editor is closed.

Note

In order that the content in the imc Online FAMOS editor and in the external editor match, the external editor may not keep the exchange file open after import, but should be able to automatically recognized any changes in the file.

- Example: WINWORD doesn't keep files open upon loading them. Automatic update of the temporary file in WINWORD is not possible, because imc Online FAMOS cannot access the open file.
- Example: NOTEPAD opens file and displays their contents. The file can later be altered by imc Online FAMOS, but the editor doesn't recognize the file alteration.

11.2.8.8 Tips and tricks

imc Online FAMOS - Computation uncertainty in summations

Goal: An incremental encoder is to be used to find a cumulative displacement. An incremental encoder delivers 400 pulses per meter = 2.5 mm/pulse.

If imc Online FAMOS computes the total displacement as:

```
Total_Path = sum (incremental_Channel, 1)
```

then the computed path will deviate from the actual displacement after a while.

Why does this happen?

imc Online FAMOS processes channel data in so-called Float format:

Float = 4 byte real number = 7 significant digits

4-Byte Float

-3.4028235E+38 to -1.1754944E-38, 0.0E+0, +1.1754944E-38 to +3.4028235E+38

In this system, rational numbers such as 2.5 can only be represented with limited precision. The number 10/4 is not expressed as 2.5 but rather as 2.4999999... This amounts to a discrepancy of 0.000223516% and is actually negligible. However, if this error is compounded throughout a calculation, larger errors can result. In this case, a new value is added to the sum every 2ms (assuming every 2ms 1 pulse = 2.5mm), in other words 500 times per second = 30.000 times per minute = 1.800.000 times per hour etc.

Thus a small cause can produce a large effect.

→ Approaches to a solution: imc Online FAMOS

1. Rounding

```
_inc=Round(Inc_1*400) ;400 pulses/m
Path_online2=sum(_inc,1)/400
```

2. Fewer samples = set a longer sampling interval.

Reduce the error in consequence of aggregate summation by reducing the terms summated:

- On the page: **Analog/Digital channels** in the dialog: **Sampling & Preprocessing**, set sampling for every 1 sec, for instance, instead of 2ms.

- Or in **imc Online FAMOS**

```
;500 = 1/ sampling interval = 1 / 2 ms    400 = pulse count / m
;take path increments every 1sec with resolution 1.
Inc_1s=mean(Inc_1, 500, 500) * (400*500)
Inc_1s_round=Round(Inc_1s) ;round to whole number
;summation and division by pulse count.
Path_1s=sum(Inc_1s_round,1)/400
```

imc Online FAMOS - Insufficient Global RAM

When imc Online FAMOS is exited, a check is made of whether there is enough space in the device's global RAM for the virtual channels which were set up. If the input channels already take up too much of the device's buffer memory, the virtual channels cannot be created.

Tip: Temporarily slow down input channels

- Exit imc Online FAMOS, in spite of syntax-errors. Set a large sampling time for the input channels.
- Call imc Online FAMOS. Change anything, which will cause imc Online FAMOS to be re-compiled (e.g. enter and delete a space). Then exit imc Online FAMOS again.
- Now set on the page: **Analog/Digital channels** in the dialog: **Data transfer** the desired RAM buffer duration for each channel.
- Then go to the page: **Analog/Digital channels** in the dialog: **Sampling & Preprocessing** and set the desired sampling time for the input channels.

imc Online FAMOS - Digital Filters

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:

$$f_{\text{sample}}/f_{\text{g}} < (\text{Order})^{\text{th}} \text{Root}(1000000); \text{ where } f_{\text{sample}} = \text{ sampling frequency and } f_{\text{g}} = \text{ cutoff frequency}$$

Examples:

Order	Ratio	$f_{\text{sample}}/f_{\text{g}}$
1st	1st Root(1000 000)	< 1000000
2nd	2nd Root(1000 000)	< 1000
3rd	3rd Root(1000 000)	< 100
4th	4th Root(1000 000)	< 32
5th	5th Root(1000 000)	< 16
6th	6th Root(1000 000)	< 10
7th	7th Root(1000 000)	< 7
8th	8th Root(1000 000)	< 6

The occurrence of numerical problems also depends on the filter characteristics set and on the particular implementation of the filter.

FAQ

Question: A data stream in imc Online FAMOS is processed in conjunction with a Display variable. This isn't performed accurately down to a single sample.

Answer: When trying to multiply a status cell with a waveform, the processing depends on how the internal FIFOs are filled. Thus, slight offsets can occur.

Question: There are problems in imc Online FAMOS when using experiments at long path names.

Answer: When trying to multiply a status cell with a waveform, the processing depends on how the internal FIFOs are filled. Thus, slight offsets can occur.

Question: The fieldbus channels are not available in imc Online FAMOS, but they are listed in the main dialog.

Answer: Fieldbus channels recorded in time stamping mode, can't be proceeded in imc Online FAMOS.

Question: When outputting to a DAC, not every value is outputted?

Answer: Two possible causes:

1. imc Online FAMOS needs longer than the sampling interval for the throughput of the channels to be processed. This means that the data come in chunks, of which only the respective last value can be transferred on to the DAC.
2. The sampling rate multiplied by the input channel's RAM buffer duration is $> 2^{16}$. However, the addressing is accomplished using a 16-bit pointer. This means that the samples are not addressed individually but rather in blocks. In imc Online FAMOS, the data are then also transferred in blocks. However, only the respective last value in a block is transferred to the DAC.

Question: How many channels @100kHz sampling rate can be proceeded online by the function SoundPressureLevel simultaneously?

Answer: imc Online FAMOS can proceed up to 3 channels (independently from the main board).

11.2.9 imc Online 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.

11.2.9.1 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](#)⁵⁹⁹.

(^) 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](#)⁵⁹⁹.

(=) 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
```

11.2.9.2 Functions (alphabetic)

11.2.9.2.1 A

ABCRating

ABC-Rating: 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.

a = ABCRating(b, FrequencyRating, TimeConstant, ReductionFactor)

a: Results

b: Input channel (Signal to be evaluated)

FrequencyRating: The signal's frequency 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

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

```
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 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

11.2.9.2.2 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)

11.2.9.2.3 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.

Reference

See the example in the chapter: "[CAN-Bus Interface module](#)³⁷⁰" > ... > "[Outputting a waveform to a DAC](#)⁴³⁶".

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 Online FAMOS: The characteristic curve is limited to 50000 values.

ClHistogram

Classification histogram: amplitude time-of-stay procedure

a = ClHistogram(b, MinValue, MaxValue, Total, ResUnit, EndClass)

a: Results	ResUnit: Result units
b: Input channel	0: classes 1: like input channel
MinValue: Lower limit of range	EndClass: End classes are
MaxValue: Upper limit of range	0: closed 1: open
Total: Number of classes; ≥ 1	

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 1: like input channel
MinValue: Lower limit of range	EndClass: End classes are
MaxValue: Upper limit of range	0: closed 1: open
Total: Number of classes; ≥ 1	
Reference: Reference level	
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

CIMarkov

Data classification using Markov procedure

a = CIMarkov(b, MinValue, MaxValue, Total, Hysteresis, Axes, ResType, ResUnits, EndClass, Option)

a: Results	ResUnits: Result units
b: Input channel	0: classes 1: like input channel
MinValue: Lower limit of range	EndClass: End classes are
MaxValue: Upper limit of range	0: closed 1: open
Total: Number of classes; ≥ 1	Option: Algorithm versions
Hysteresis: Hysteresis for peak identification; ≥ 0.0	0: main algorithm
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	

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 = CIMarkov(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

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
Specify as a percentage of the 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

Option: Algorithm versions

0: main algorithm

1: 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 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	ResUnits: Result units 0: classes 1: like input channel
b: Input channel	EndClass: End classes are 0: closed 1: open
MinValue: Lower limit of range	Algorithm: use 0: main algorithm 1: main algorithm with Clormann-Seeger correction
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	

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.

CIRangePairCount

Classification histogram: Using the range pair counting procedure.

a = **CIRangePairCount**(**b**, **MinValue**, **MaxValue**, **Total**, **Hyst**, **ResUnit**)

a : Results	Hyst : Hysteresis; in physical units, ≥ 0
b : Input channel	ResUnit : Unit of the Result
MinValue : Lower limit of range	0: classes
MaxValue : Upper limit of range	1: like input channel
Total : Number of classes; ≥ 2	

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 = CIRangePairCount( 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.
-

CIRevolutionsHistogram

Revolutions histogram: Counts the revolutions into the classes of the input channel.

a = **CIRevolutionsHistogram**(**b**, **c**, **MinValue**, **MaxValue**, **Total**, **ResUnit**, **EndClass**)

a : Result	ResUnit : Result units
b : Input channel	0: classes
c : speed in rpm	1: like input channel
MinValue : Lower limit of range	EndClass : End classes are
MaxValue : Upper limit of range	0: closed
Total : 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.

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
```

11.2.9.2.4 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
  Dbl1 = DelayBuffer( 100 )
  Value = 0
  DelayValue = 0
End

OnSyncTask( 0.01 )
  If Virt_Bit01 > 0
    Dbl1.Fill( -1000.0 )
    Virt_Bit01 = 0
  End
  ...
  ; delay if the output value by 1s (100*0.01s):
  DelayValue = Dbl1.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 = Dbl1.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:

$$0 \ a_2 \ a_1 \ b_2 \ b_1 \ b_0 \ a_2 \ a_1 \ b_2 \ b_1 \ b_0 \ \dots \ a_2 \ a_1 \ b_2 \ b_1 \ b_0$$

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
```

$$y[t] = b0 * x[t] + b1 * x[t-1] + b2 * x[t-2] + a1 * y[t-1] + a2 * y[t-2]$$

$$y[t] = 0.000317864 * x[t] + 0.000635728 * x[t-1] + 0.000317864 * x[t-2] + 1.94894 * y[t-1] + -0.950121 * y[t-2]$$

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 WAVE 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

The enumeration of the page number starts at 1. This corresponds to the enumeration in the function [DisplaySetPage](#).

Button: Button index.

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

11.2.9.2.5 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, 0x0000FFFF )
  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

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

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 [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.

11.2.9.2.6 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 )
```

Calculation 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: 128 ... 8192
b: Input channel	1: Hamming	In imc Inline FAMOS: 128 ... 131072
	2: Hanning	
	3: Blackman	
	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 $1+(\text{FFT_Length}/2)$.



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`; 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 complex 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:

128 ... 8192

1: Hamming

In imc Inline FAMOS:

128 ... 131072

2: Hanning

3: Blackman

b: Input channel

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 $1 + (\text{FFT_Length}/2)$.



Example

```
FFTRealImag( Reall, Imagl, Signal_01, 5, 8192 )
```

Calculation of a complex 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:

$fsample/fg < (Order)th \text{ Root}(1000000)$; where $fsample$ = sampling frequency and fg = cutoff frequency

See also "Tips and Tricks" > "[imc Online FAMOS - Digital Filters](#)".

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.



Example

The function returns

Input channel value	Result value
1,2	1
1,9	1
-1,2	-2
-1,9	-2

11.2.9.2.7 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, supplm. 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.

Samplenummer = **GetSampleCount**(Channel)

Samplenummer: 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.

11.2.9.2.8 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

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 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](#)₆₇₁)); 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.

11.2.9.2.9 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:

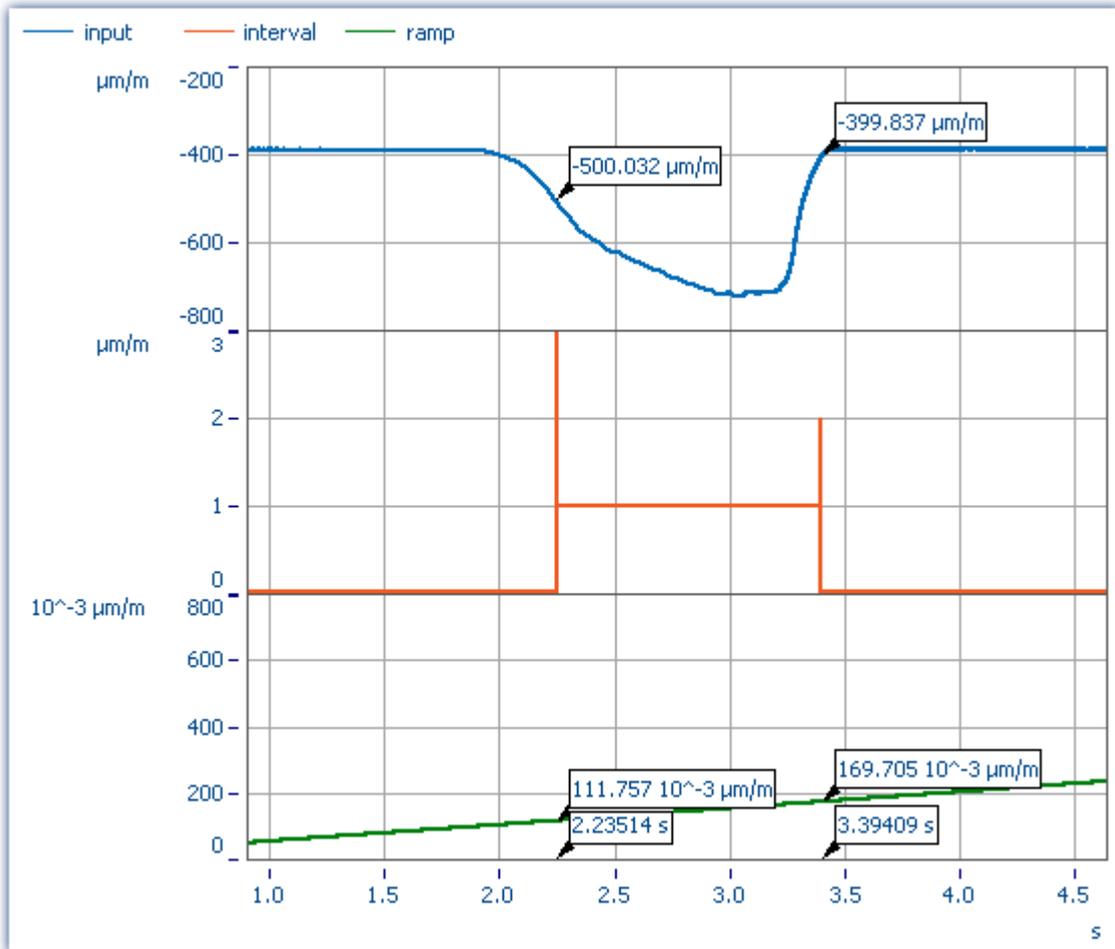
 Example

An input channel is analyzed in the interval from one descending edge at $-500 \mu\text{m}/\text{m}$ until an ascending edge at $-400 \mu\text{m}/\text{m}$. 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 = $0.1697 \mu\text{m}/\text{m}$
Minimum = $0.1118 \mu\text{m}/\text{m}$

 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

Intervall: 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](#) ⁶⁰⁰.



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](#)^[600].



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](#)^[600].



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 pulse 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	AmtPoints: Number of encoder pulses per total angle or unit length
Signal: Measurement signal	Unit: Value of the result
Interval: Interval signal	IsSegmented: Is the result segmented?
MaxAngle: Value of total angle or unit length	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 $2 \cdot \pi$ 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 `IntervalFromLevel`.

`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 `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](#)^[600].



Example

```
; Determine the root mean square of a sine signal
; Theoretical value = 0.707107, independent of the interval limit specified
; by IntervalFromLevel
saw1 = SawTooth( Channel_001, 0, 0.01, 628 )
signal1 = Sin( saw1 )
interval = IntervalFromLevel( 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	b: clock, only for establishing the data rate.
1: synchronized	
0: not synchronized	
-1: unable to determine/no clock present	

JKFlipFlop

JK flip-flop function

a = JKFlipFlop(J, K)

a: Result	J: J input
	K: K input

Returns a 1 for HIGH (H) state or 0 for LOW (L) state.

A change to HIGH state occurs if J is not equal to zero and K is equal to zero.

Likewise, a change to LOW state occurs if J is equal to zero and K is not equal to zero.

If both J and K are both equal to zero, the state does not change.

If neither J nor K are equal to zero, the state changes.

The function may also be called with single values (e.g. virtual bit or Display variable) as its parameter. In order to obtain useful results in such a case, calls of the function should occur at constant time intervals, e.g. in a Timer or a synchronous task.



Example

```
; The LED flashes while the signal is larger than 9.
res = Greater( Channel_001, 9 )
LED_01 = JKFlipFlop( res, res )
```

11.2.9.2.10 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, FrequencyRating, ReductionFactor)

LEQSignal: Evaluated signal	FrequencyRating: Frequency rating of the signal	ReductionFactor: Factor for resampling, ≥ 1 .
Signal: Signal to be evaluated	0: no rating 1: A-rating 2: B-rating 3: C-rating	For a reduction factor = 1, no resampling

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 $20 \cdot 10^{-6}$ Pa and expressed in dB. The result can always be interpreted as the mean sonic pressure level of all data already captured.

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 in such a way that multiple values per second are generated.



Example

```
LEQ_Chan1 = LEQ( Chan1, 1, 1000 )
```

The signal is subject to an A-rating and resampled by the factor 1000. The signal originally had a sampling frequency of 20kHz, the result only has 20 Hz. The result is the mean sound pressure level LEQ, expressed 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)

11.2.9.2.11 M

Max

Maximum of a window, with reduction

$a = \text{Max}(b, \text{Windowwidth}, \text{Reduction}[, \text{Mode}])$

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

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 `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

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 [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 $1 \cdot \text{RF}$, 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

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 **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

If Monoflop detects a positive edge in the input signal (a transition of zero to non-zero) it produces a pulse of predetermined duration with amplitude 1.

Not retriggerable, i.e. signal edges are only evaluated after the end of the pulse.

The function may also be called with a single value (e.g. virtual bit or Display variable) as its first parameter. In order to obtain useful results in such a case, calls of the function should occur at constant time intervals, e.g. in a Timer or a synchronous task.

MonoflopRT

Retriggerable monoflop

a = MonoflopRT(b, Duration)

a: result

Duration: Pulse duration in samples; ≥ 1

b: input channel

If MonoflopRT detects a positive edge in the input signal (a transition of zero to non-zero) it produces a pulse of predetermined duration with amplitude 1.

Additional positive edges within this duration prolong the output pulse.

The function may also be called with a single value (e.g. virtual bit or Display variable) as its first parameter. In order to obtain useful results in such a case, calls of the function should occur at constant time intervals, e.g. in a Timer or a synchronous task.

11.2.9.2.12 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**^[698] 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**^[698] is in a combination as illustrated in the example below. Instead of averaging, the standard deviation, i.e. the function **StDev**^[670], 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

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

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 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](#)⁶⁷¹)); 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.

11.2.9.2.13 O

OnECUCmdReturn_ECU_001

See [ECU-functions](#) ⁵⁷⁹.

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

EncoderType: Type of encoder, how many missing teeth?

b: Input channel - Pulse_Signal: Time-history of the pulse signal

0: Default

Signal_Type: Type of pulse signal

1: 1 tooth missing

0: Number of events

2: 2 teeth missing

1: Time of pulse

EncoderPulse: Pulses per revolution, ≥ 1

2: Sampled rectangle

RPM_Min: Minimum occurring RPM-value

3: Sine signal

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.

$$\text{Maximum possible RPM [revs/min]} = 60 / (\text{encoder pulses} * \text{sampling interval [s]})$$

- 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 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 0-1sequence.

**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 an integer number of samples. But the Product $\text{PeriodLength} * \text{PeriodAmount}$ must be of an 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.

Sinus: Results of the calculation (sinusoidal oscillation)

SampleCount: Length of the sinusoidal oscillation found as the results (in Samples)

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.

```
OtrFrequLine3( 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	Resolution: Resolution of the order spectrum [1, 1/2, 1/3, 1/4, ...]
Vibration: Time-history of vibration signal	OrderMax: The highest order (line) in the order spectrum
RPM: Time-history of the RPMs. In scaled to U/min	AveragingType: The processing of all spectra within one RPM-class
RPM_Min: Lower limit of desired RPM-range	0: Arithmetic mean
RPM_Max: Upper limit of desired RPM-range	1: Maximum
RPM_Bin_Width: Width (interval) of an RPM-class	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	RPM_Min: Lower limit of desired RPM-range
Vibration: Time-history of vibration signal	RPM_Max: Upper limit of desired RPM-range
PulseSignal: Time-history of pulse signal	RPM_Bin_Width: Width (interval) of an RPM-class
SignalType: Type of pulse signal	Resolution: Resolution of the order spectrum [1, 1/2, 1/3, 1/4, ...]
0: Number of events	OrderMax: The highest order (line) in the order spectrum
1: Time of pulse	AveragingType: The processing of all spectra within one RPM-class
2: Sampled rectangle	0: Arithmet. mean
3: Sine signal	1: Maximum
EncoderType: Type of encoder, how many missing teeth?	2: Minimum
0: Default	
1: 1 tooth missing	
2: 2 teeth missing	
EncoderPulse: Pulses per revolution, ≥ 1	

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 $\text{OrderRef} * \text{Oversampling}$

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

RPM_Max: Maximum desired RPM-value in revs/min

Vibration: Time history of the vibration signal

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:

$$\text{OrderMax} \ll 10 / (\text{SampleTime_Vibration} * \text{RPM_Max})$$

where SampleTime_Vibration is the sampling time of the signal 'Vibration'.

$$\text{OrderMax} = \text{OrderMed} * \text{sqrt} (1 + \text{WidthPercent} / 100)$$

\ll 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: $\text{OrderMax} = 1.7 \ll 10 / (0.0005 * 6000) = 3.33$

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	WidthPercent: Bandpass filter bandwidth in percent, 10% ... 100%
Vibration: Time history of the vibration signal	FilterOrder: The filter order of the bandpass filter
RPM: Time history of the RPMs. Scaled in rotations/minute	Interpolation: Specifies whether the results are interpolated?
RPM_Min: Lower end of desired RPM-range	0: No interpolation
RPM_Max: Upper end of desired RPM-range	1: Constant interpolation (centered around auxiliary values)
RPM_ClassWidth: Bin spacing (class width) of RPM classes	2: Linear interpolation
OrderMiddle: The order (line) to be isolated	

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	Calculation: How are the values of an RPM-class represented mathematically?
Vibration: Time history of the vibration signal	0: Root-mean square (default)
RPM: Time history of the RPMs. Scaled in rotations/minute.	1: Arithmetic mean
RPM_Min: Lower end of desired RPM-range	2: Minimum
RPM_Max: Upper end of desired RPM-range	3: Maximum
RPM_ClassWidth: Bin spacing (class width) of RPM classes	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	RPM_Bin_Width: Width (interval) of an RPM-class
Spectrum_Sequence: Time-history of a spectrum	Processing: How are the values from a single RPM-class processed?
RPM: Time-history of RPM signal	0: Arithmetic mean
RPM_Min: Lower limit of desired RPM-range	1: Maximum
RPM_Max: Upper limit of desired RPM-range	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	WindowType: Window function for the FFT used
Vibration: Time history of the vibration signal	0: Rectangular
RPM: Time history of the RPMs. Scaled in rotations/minute	1: Hamming
RPM_Min: Lower end of desired RPM-range	2: Hanning
RPM_Max: Upper end of desired RPM-range	3: Blackman
RPM_ClassWidth: Bin spacing (class width) of RPM classes	4: Blackman / Harris
WindowWidth: Width of the time-window in data points, 100 ... 8192	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

Order3dB: Order at which the low-pass filter dampens the signal by 3dB

Vibration: Time history of vibration signal

FilterOrder: The low-pass filter order

RPM: Time history of RPMs. Scaled in revs/min

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%.

11.2.9.2.14 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](#)⁵³⁰".

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

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)

uRMS3: Channel with 3rd phase RMS voltage data (u3)

iRMS3: Channel with 3rd phase RMS current data (i3)

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

u3: Channel with 3rd phase voltage data

i3: Channel with 3rd 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](#)".

PulseDuration

Pulse duration: A signal's average pulse duration in the window, with resampling.

a = **PulseDuration**(**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

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 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`)); 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

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

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 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](#)₆₇₁)); 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

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 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](#)^[671]); 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.

11.2.9.2.15 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 Online 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 Online 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

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 `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.

Restore

Restore pv.x = 0

With this attribute, the value of a pv-variable can be restored in the device automatically if the device is prepared for a measurement.

For instance, a pv-variable's last value can be restored in the device, if the measurement will be restarted, also if the device has to be prepared again.



Reference

See the data type documentation for more information: "*Channel types, variable types, data types and data storage formats*" > "*Process Vector Variables*" > "[Saving/Restoring pv-variables](#)".



Example

```
OnInitAll
  Int Restore pv.x = 0 ; for Integer pv-variables
  Restore pv.y = 0.0 ; for Float pv-variables
End
```

With the syntax `Restore pv.x = 0`, the pv-variable is loaded with the last value upon each start, even if "= 0" seems to imply that the variable is set to 0.

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

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 **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 $1 \cdot \text{RF}$, 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, $\text{Eps1} \geq \text{Eps2}$.

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 ($\varepsilon_a, \varepsilon_b, \varepsilon_c$) with measured strains in $\mu\text{m}/\text{m}$

Type: Rosette type

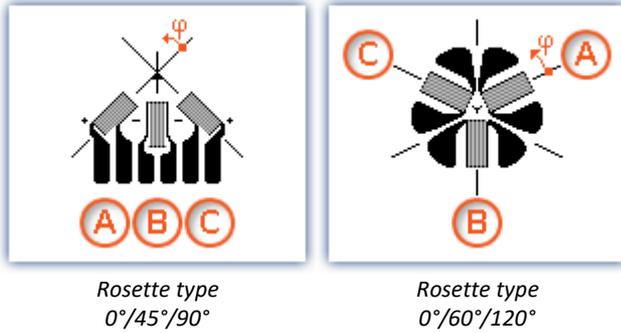
1: Rectangular rosette ($0^\circ/45^\circ/90^\circ$)

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 $\mu\text{m}/\text{m}$



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 ° (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°

$$\phi_{p,q} = \frac{1}{2} \arctan\left(\frac{2\varepsilon_b - \varepsilon_a - \varepsilon_c}{\varepsilon_a - \varepsilon_c}\right)$$

Formulas used	
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](#)⁶⁵⁸

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 HIGH (H) state or 0 for LOW (L) state.

A state change from L to H occurs if S is not equal to zero and R is equal to zero.

Likewise, a state change from H to L occurs if S is equal to zero and R is not equal to zero.

Otherwise, the state does not change.

The function may also be called with a single value (e.g. virtual bit or Display variable) as its first parameter. In order to obtain useful results in such a case, calls of the function should occur at constant time intervals, e.g. in a Timer or a synchronous task.



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	<i>long name</i>	<i>short name</i>	<i>column ID</i>
Balance at device startup	<i>Balance at device startup</i>	<i>Balance at startup</i>	<i>eBalanceAtDeviceStart</i>

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](#)⁶⁶².

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()

11.2.9.2.16 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?

Parameter: Depending on Mode: Pulse rate, condition or reserved

CAN_001: 1st channel to send

0: Cyclic sending

If mode = 0: Pulse rate in s

[**CAN_002**], [...]: If 2 channels, for instance, are defined in the CAN-Assistant

1: Conditional sending

If mode = 1: Condition

2: Send always (only for Control commands)

If mode = 2: Reserved, 0

Operation Mode 2 can only be used with control commands.

Configuring a message to transmit is the condition for applying this function (see chapter "*Fieldbusses*" > "*CAN-Bus Interface*" > "*CAN-Bus Assistant*" > ... > "*Sending messages*"³⁹⁷). 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

```

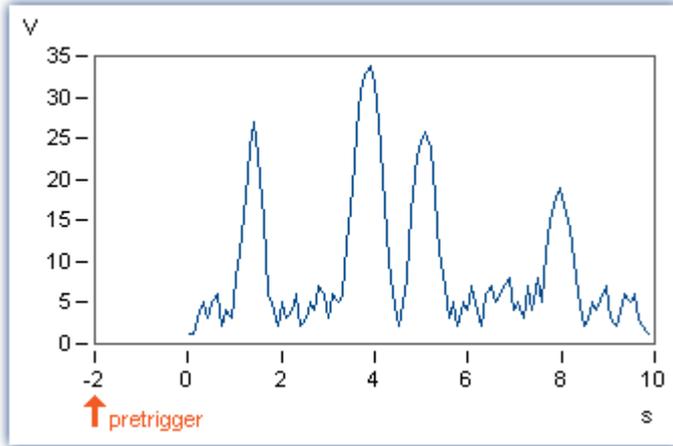
The virtual channel created is filled according to assignments of single values. Whenever an assignment is carried out, this single value is added to the virtual channel as a sample value. Thus it can happen that the virtual channel contains no samples. On the other hand, a large number of samples can 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.

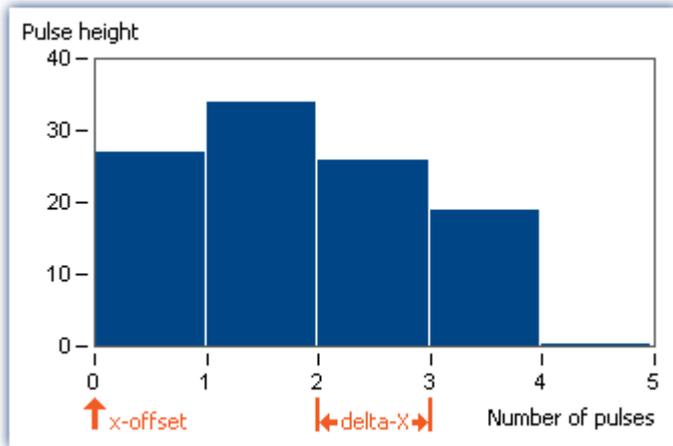
By default, the clock rate is set to 1s and no units are displayed in the curve window.

 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	Time rating: Averaging time, time rating of filtered data
Vibration: Time-history of vibration signal, the time scaled in seconds	-1: Fast (0.125s)
f1: Center frequency of lowest 1/3 octave in Hz 10, 12.5, 16, 20, 25, 31.5, 40, 50, 63, 80, 100, ...	-2: Slow (1s)
f2: Center frequency of highest 1/3 octave in Hz 1000, 1250, 1600, 2000, 2500, 3150, 4000, 5000, 6300, 8000, 10000, 12500, 16000, ...	-3: Pulse
Frequency weighting: Frequency weighting for the result	-4: Peak
0: linear	-5: RMS in interval
1: A-weighting	-6: RMS from start
2: B-weighting	> 0.0: Time constant is defined by the user, given in seconds
3: C-weighting	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.
4: D-weighting	

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-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

b: Input channel

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

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 `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 $1 \cdot \text{RF}$, 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

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 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

The optional "Mode" parameter applies only to imc Online FAMOS. It does not exist for imc Inline FAMOS and is interpreted as 1.

In contrast to the funktion [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
```

11.2.9.2.17 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.

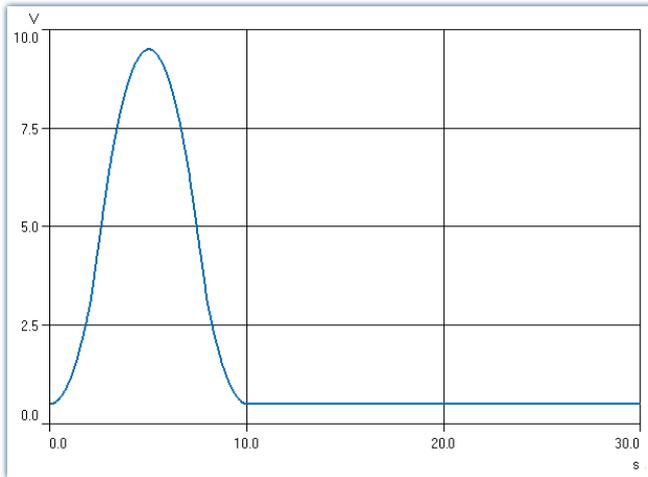
**Note****Transfer to PC is not possible**

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.

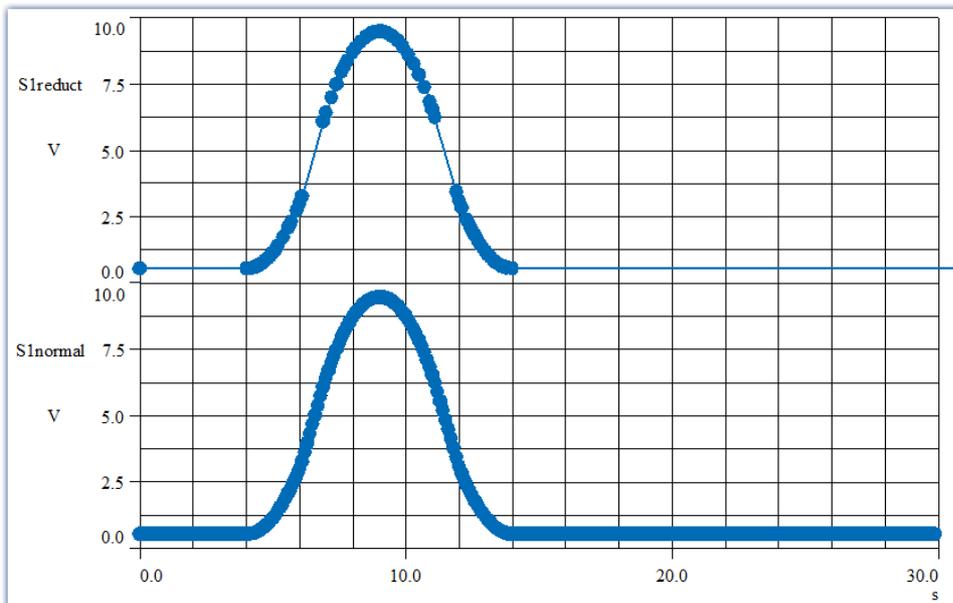


In this example, the signal is recorded once with and once without **TransRec** and saved.

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

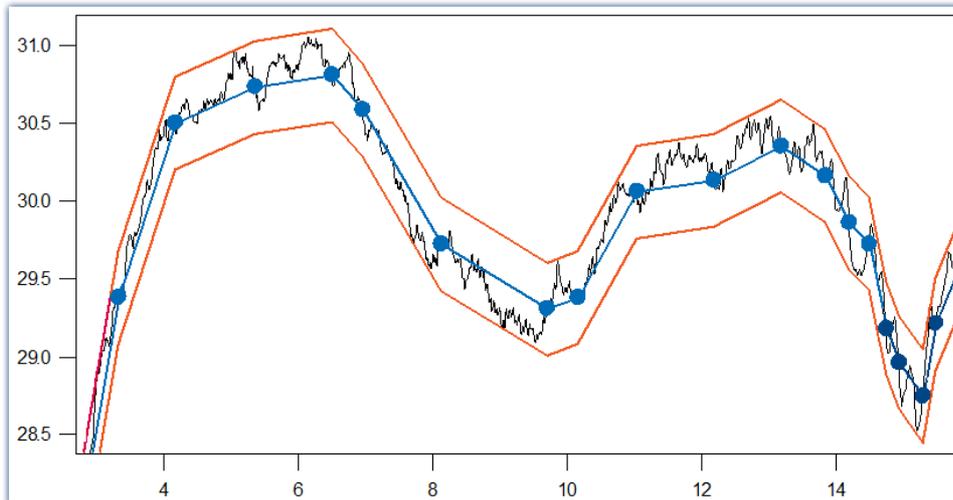
The data reduction only adds measured points wherever the curve features significant curvature. Straight-line signal plots of any slope are recognized and summarized as a straight line between two measured points.



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.

11.2.9.2.18 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.

11.2.9.2.19 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

$$\mathbf{a} = \mathbf{VMin}(\mathbf{b})$$

a: Output value

b: Input vector series

VMinV

Minimum from vector series: minimum values from a reduced vector series

$$\mathbf{a} = \mathbf{VMinV}(\mathbf{b}, \text{WindowOption}, \text{RF})$$

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

$$\mathbf{a} = \mathbf{VRedV}(\mathbf{b}, \text{RF})$$

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

$$\mathbf{a} = \mathbf{VRMS}(\mathbf{b})$$

a: Output value

b: Input vector series

VSum

Sum from vector: returns the sum of the elements in each input vector

$$\mathbf{a} = \mathbf{VSum}(\mathbf{b})$$

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

11.2.9.2.20 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`^[616] 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`^[616] and `WindRoseCorr` is in a combination as illustrated in the example below. Instead of averaging, the standard deviation, i.e. the function `StDev`^[670], can be used.



Example

```
NC = NorthCorrection( Channel, 10 )
NC_Mean = Mean( NC, 10, 10 )
Result = WindRoseCorr( NC_Mean )
```

11.2.9.3 Functions with Control commands

Prerequisite:

Only usable with control commands!

Function Groups	Operators and Symbols	Additional functions
Measurement Process States ^[699]	(=) Equal Operator (Comparison) ^[544]	AND ^[552]
Loops, Conditions ^[706]	(<>) Inequality operator ^[545]	CloseSaveInterval ^[559]
Timer Functions ^[710]	(<) Less than operator ^[545]	CurrentValue ^[568]
CanMsg ^[711]	(<=) Less than or equal operator ^[545]	DelayBuffer* ^[569]
Controller Functions ^[716]	(>) Greater than operator ^[546]	ECU* ^[578]
	(>=) Greater than or equal operator ^[546]	GetDateTime ^[589]
		NOT ^[617]
		OnECUCmdReturn ECU * ^[579]
		OR ^[619]
		ReadyForPowerOff ^[651]
		SyncOverload ^[675]
		VectorChannel ^[688]
		VectorChannelSet ^[689]
		VectorStatic ^[690]

11.2.9.3.1 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
```



Note

Prepare and Reconfigure

The "*OnInitAll*" block runs when the device needs to be prepared again. But this is not applicable to repeated measurements, unless the device configuration has been changed between measurements. When changes are made to the Panel pages or to PC-side analyses, such as imc Inline FAMOS, Bus Decoder, the device configuration remains unaffected. For this reason, no preparation is performed. Even **manually clicking the button "Prepare"** does **not cause preparation** to be performed. In this case, the system checks whether any changes had been made to the device configuration. Then preparation is performed if this applies.

Preparing the device **can be forced** by selecting the menu item "**Reconfiguration**".

See also: "[Prepare and Reconfigure](#)"¹⁴⁹

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 a 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](#)^[675].



Reference

For more information on **controller** see in the documentation on imc Online FAMOS: "*imc Online FAMOS Professional*" > "[PID-Controller](#)^[532]".

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](#)⁷¹⁰).

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
```

11.2.9.3.2 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](#)⁵²³.

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
```

11.2.9.3.3 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

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

End
```

11.2.9.3.4 CanMsg

Prerequisite:

Only available in: imc Online FAMOS!

CanMsg

CanMsg-initialization: Initialization of the CanMsg-structure with default values.

CanMsg1 = CanMsg()

CanMsg1: CanMsg-structure

To send CAN- messages with CanMsg-structures, the function `CanMsg` must be called in the `OnInitAll` section. In the process, a `CanMsg`-structure is created and initialized. MessageID, message-length and message-content must be set before the CAN-message is sent. The message-ID and message-length are set with the variables `.ID` and `.Len`. The individual message bytes are set with the variables `.Byte0Byte7` or with the function `.SetData`. The CAN-message thus created is sent by means of the `.Transmit1_S1`.



Example

sending a CAN-message

```

OnInitAll
  CanMsg1 = CanMsg ()
End

OnAlways
  If VirtBit_01 > 0
    CanMsg1.ID = 100
    CanMsg1.Len = 2
    CanMsg1.Byte0 = 10
    CanMsg1.Byte1 = 20
    CanMsg1.Transmit1_S1 ()
    VirtBit_01 = 0
  End
End

```



Example

receiving a CAN-message

```

OnInitAll
  CanID = 0
  CanLen = 0
  CanByte0 = 0
  CanByte1 = 0
End

OnCanMessageReceive1_S1 ( CanMsg1 )
  CanID = CanMsg1.ID
  CanLen = CanMsg1.Len
  CanByte0 = CanMsg1.Byte0
  CanByte1 = CanMsg1.Byte1
End

OnTriggerMeasure ( BaseTrigger )
  VID = Channel_001*0 + CanID
  VLen = Channel_001*0 + CanLen
  VByte0 = Channel_001*0 + CanByte0
  VByte1 = Channel_001*0 + CanByte1
End

```



Note

The `CanMsg` functions only appear if the CAN-Assistant had previously been opened. No changes to operations in the CAN-Assistant need to have happened, but only once it has been opened does imc Online FAMOS learn that there is a CAN Bus interface.

CanMsg.GetData

Value from CAN-message: A value is extracted from a CAN-message.

Result = CanMsg.GetData(StartByte, StartBit, BitCount, BitSequence, Format)

Result: Value extracted from the CAN-message

StartByte: Initial byte in the message, 0..7

StartBit: Initial bite in the message, 0..7

BitCount: Number of bits in the value to be sent, 1..32

ByteSequence: Intel- or Motorola-format

1: Intel

2: Motorola

Format: Numerical format of the value to be sent,

1: signed integer

2: unsigned integer

3: real number (32 bit)

4: digital bit

To use this function, it's necessary to generate a `OnCanMessageReceive1_S1`-section, in which the function `.GetData` is called. The content of the received CAN-messages is extracted from the message with the variables `.Byte0Byte7` or with the function `.GetData`.



Example

```

OnInitAll
  CanID    = 0
  CanLen   = 0
  CanByte0 = 0
  CanByte1 = 0
End

OnCanMessageReceive1_S1( CanMsg1 )
  CanID    = CanMsg1.ID
  CanLen   = CanMsg1.Len
  CanByte0 = CanMsg1.GetData( 0, 4, 16, 1, 1 )
  CanByte1 = CanMsg1.GetData( 2, 4, 16, 1, 1 )
End

OnTriggerMeasure( BaseTrigger )
  VID     = Channel_001*0 + CanID
  VLen    = Channel_001*0 + CanLen
  VByte0  = Channel_001*0 + CanByte0
  VByte1  = Channel_001*0 + CanByte1
End

```

CanMsg.SetData

Value in message: A value is inserted into a CAN-message.

CanMsg.SetData(Value, StartByte, StartBit, BitCount, ByteSequence, Format, Factor, Offset)

Value: Value to be sent	Format: Numerical format of the value to send,
StartByte: Initial byte in the message, 0..7	1: signed integer
StartBit: Initial bit in the message, 0..7	2: unsigned integer
BitCount: Number of bits in the value to be sent, 1..32	3: real number (32 bit)
ByteSequence: Intel- or Motorola-Format,	4: digital bit
1: Intel	Factor: Weighting of the value to be sent with a factor
2: Motorola	Offset: Weighting of the value to be sent with an offset

To use the function, a `CanMsg`-structure must be defined in the `OnInitAll` section.

Message-ID, message-length and message-content must be set before the CAN-message is sent. The message-ID and message-length are set with the variables `.ID` and `.Len`. The individual message bytes are set with the variables `.Byte0` ... `.Byte7` or with the function `.SetData`. The CAN-message thus created is sent by means of the function `.Transmit1_S1`.



Example

```

OnInitAll
  CanMsg1 = CanMsg()
End

OnAlways
  If VirtBit_01 > 0
    CanMsg1.ID = 100
    CanMsg1.Len = 4
    CanMsg1.SetData( 1000, 0, 4, 16, 1, 1, 1.0, 0.0 )
    CanMsg1.SetData( 2000, 2, 4, 16, 1, 1, 1.0, 0.0 )
    CanMsg1.Transmit1_S1()
    VirtBit_01 = 0
  End
End

```

CanMsg.Transmit1_S1

Send a CAN-message: A CAN-message is sent to Node 1.

CanMsg.Transmit1_S1()

To use the function, a `CanMsg`-structure must be defined in the `OnInitAll` section.

Message-ID, message-length and message-content must be set before the CAN-message is sent. The message-ID and message-length are set with the variables `.ID` and `.Len`. The individual message bytes are set with the variables `.Byte0` ... `.Byte7` or with the function `.SetData`. The CAN-message thus created is sent by means of the function `.Transmit1_S1`.

**Example 1**

```
OnInitAll
    CanMsg1 = CanMsg()
End

OnAlways
    If VirtBit_01 > 0
        CanMsg1.ID = 100
        CanMsg1.Len = 2
        CanMsg1.Byte0 = 10
        CanMsg1.Byte1 = 20
        CanMsg1.Transmit1_S1()
        VirtBit_01 = 0
    End
End
```

**Example 2**

```
OnInitAll
    CanMsg1 = CanMsg()
End

OnAlways
    If VirtBit_01 > 0
        CanMsg1.ID = 100
        CanMsg1.Len = 4
        CanMsg1.SetData( 1000, 0, 4, 16, 1, 1, 1.0, 0.0 )
        CanMsg1.SetData( 2000, 2, 4, 16, 1, 1, 1.0, 0.0 )
        CanMsg1.Transmit1_S1()
        VirtBit_01 = 0
    End
End
```

OnCanMessageReceive1_S1

This section deals with CAN-messages which are received from the device and can't be permanently set in the CAN-Assistant.

OnCanMessageReceive1_S1(CanMsg1)

CanMsg1: CanMsg-structure to be newly created

The `CanMsg`-structure passed in is created and initialized for this section. The elements of the `CanMsg`-structure are treated like local single value variables in the section `OnCANMessageReceive_S1` and may also be used only within this section. These variables are filled when the command for reading CAN-messages is run, and can be used and queried in the section `OnCANMessageReceive_S1` on the right side of formulas. They are single values.

The content of the CAN-message can be extracted from the message either with the variables `.Byte0` ... `.Byte7` or with the function `.GetData`.



Example

```
OnInitAll
  CanID      = 0
  CanLen     = 0
  CanByte0   = 0
  CanByte1   = 0
End

OnCanMessageReceive1_S1( CanMsg1 )
  CanID      = CanMsg1.ID
  CanLen     = CanMsg1.Len
  CanByte0   = CanMsg1.Byte0
  CanByte1   = CanMsg1.Byte1
End

OnTriggerMeasure( BaseTrigger )
  VID       = Channel_001*0 + CanID
  VLen      = Channel_001*0 + CanLen
  VByte0    = Channel_001*0 + CanByte0
  VByte1    = Channel_001*0 + CanByte1
End
```

11.2.9.3.5 Controller functions

Prerequisite:

Only available in: imc Online FAMOS Professional!



Reference

For more information on **controller** see in the documentation on "*imc Online FAMOS*" > "*Calculation examples*" > "[PID-Controller](#)⁵³²".

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. $0.1 / [\text{controller's cycle time}]$.

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.



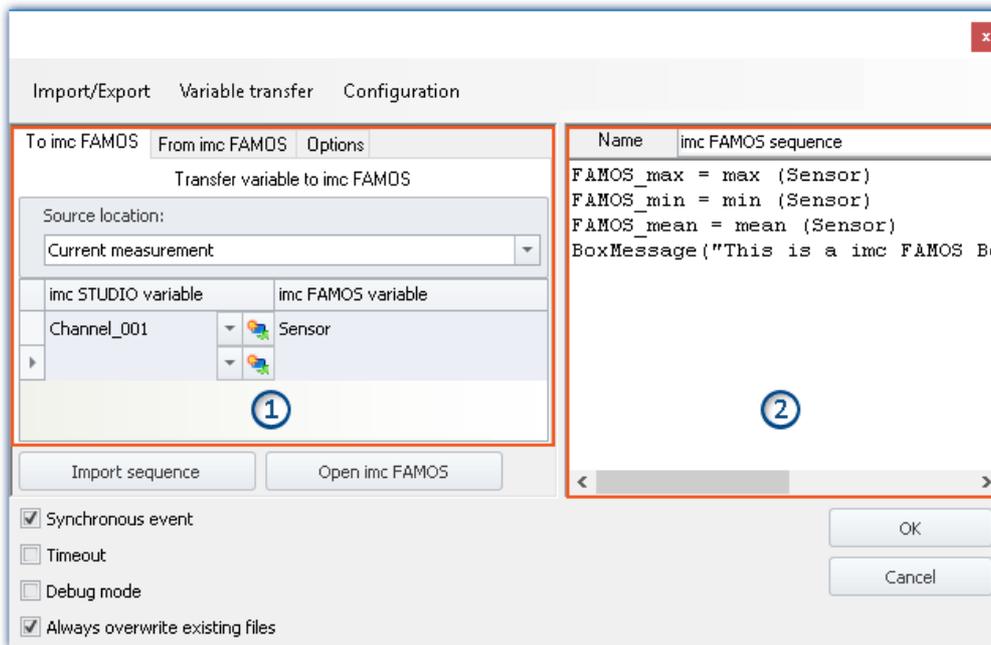
Example

```
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
```

11.3 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](#) ⁷²⁴ (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](#) ⁷²⁴ 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](#) ⁷²⁴ 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

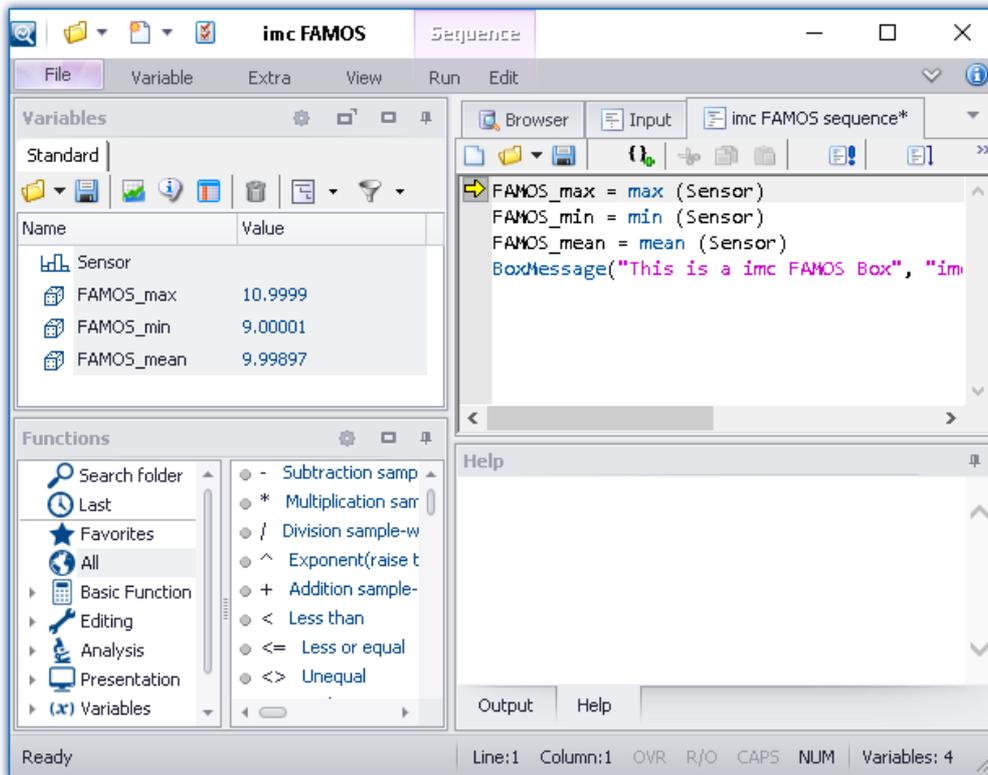
Checkbox	Description
Synchronous event	<p>"<i>Synchronous event</i>" means that the source waits until imc FAMOS has concluded the sequence. This means:</p> <ul style="list-style-type: none"> • Sequencer: The next command line is only started when imc FAMOS has concluded the sequence. • Automation: The next step only starts when imc FAMOS has concluded the sequence. <p>If "<i>Synchronous event</i>" is deactivated, the new command line or step runs immediately. Meanwhile, the sequence continues to run in parallel.</p> <p>See also the notes on "Parallel processing of sequences"⁷²⁹.</p>
Timeout	<p>The option "<i>Timeout</i>" (only available if the dialog is called from a command) only appears if "<i>Synchronous Event</i>" is activated.</p> <ul style="list-style-type: none"> • At the latest, after a specified timeout time, the Sequencer runs the next line.
Debug Mode	<p>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.</p> <p>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.</p> <hr/> <p>Please note that in Debug Mode, the command needs longer to execute:</p> <ul style="list-style-type: none"> • 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. <p>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.</p>
Always overwrite existing files	<p>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.</p>

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](#)⁷²³ 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

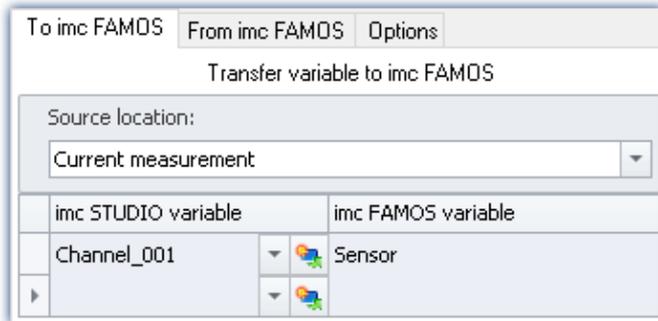
When exporting an experiment, the external files called in the sequence are not automatically included! Copy these files separately.

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](#)"⁷²).

11.3.1 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")



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](#)⁷²³).

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. <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  More detailed information on the "Last measurement" is presented in the chapter: "<i>Storage options and directory structure</i>" > "The last measurement"³³⁵ </div>
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.

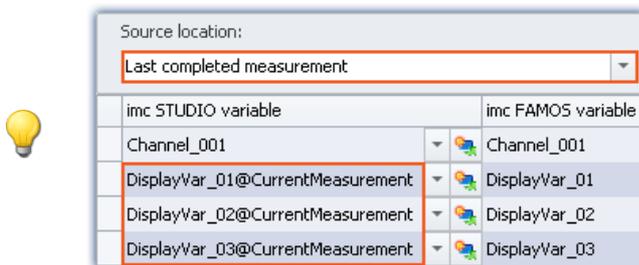
In this case, use the following syntax for the imc STUDIO variable:

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*".

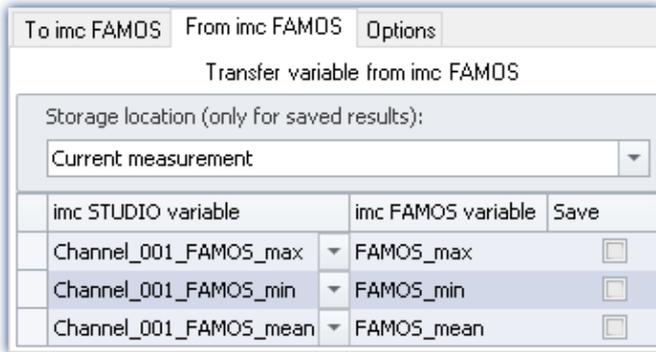


In this example, a particular measurement is selected. Additionally, variables from "*Current Measurement*" are also transferred. In this case, use the following syntax for the imc STUDIO variable:

<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>	

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

11.3.2 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 WAVE

Upon closing imc WAVE, 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](#) (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.

12 Channel types, variable types, data types and data storage formats

In imc WAVE, 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?	<ul style="list-style-type: none"> • Data types ^[730] • Structured data sets - Event-based or sequenced channel ^[731] • Time-stamped channels - Setting the RAM-buffer duration, circular buffer duration and pretrigger duration ^[732]
What variable types and channel types are there? Where do the variables come from?	<ul style="list-style-type: none"> • Monitor channels ^[735] and GPS variables ^[745] • Trigger, devices and system variables ^[738] • Process vector variables ^[741] and Ethernet bits ^[737] • User-defined variables ^[749]
In what format are the variables saved? How to export/import configurations?	<ul style="list-style-type: none"> • Data file format - Saving measured data during measurement ^[753] • Memory requirements for measurement data ^[755] • Parameter set ^[759]

12.1 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	<p>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.</p> <p>These include for example the analog channels, equidistantly captured fieldbus channels, ...</p>
Single values	<p>A single value is an individual real number having a name and in some cases a unit as well.</p> <p>These include, for example: Display-, Process vector variables, user-defined numeric variables, ...</p>
Bits	<p>A variable having the values 0 or 1.</p> <p>These include, for example: Ethernet-bits, virtual bits</p>
Text-variables	<p>A user-defined text.</p> <p>These include, for example: user-defined text-variables</p>

Data type	Description
XY-data sets	<p>Time-stamped channel with numerical values.</p> <p>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.</p> <p>Example: CAN-channel with the setting "x-axis": "<i>Time-stamp</i>"</p> <hr/> <p> Observe the notes on making the settings for RAM-buffer duration, circular buffer duration and pretrigger duration ⁷³².</p>
TimeStamp-ASCII (TSA)	<p>Time-stamped channel with ASCII-values, also known as TSA.</p> <p>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 fieldbuses.</p> <p>Example: RecordEvent/RecordText, various fieldbus protocols, ...</p> <hr/> <p> Observe the notes on making the settings for RAM-buffer duration, circular buffer duration and pretrigger duration ⁷³².</p>
Vectors	<p>A defined set of single values.</p> <p>These include, for example: various vectors from imc Online FAMOS</p>
Text array	<p>Text arrays from imc FAMOS can't be read and evaluated in imc WAVE by ordinary means. Widgets are not able to handle text arrays.</p> <p>It is possible to access variables via a script.</p> <p>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 WAVE and then passed back to imc FAMOS at a later time, in order to be filled by imc FAMOS with additional entries.</p>

12.1.1 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](#)" ³¹⁷

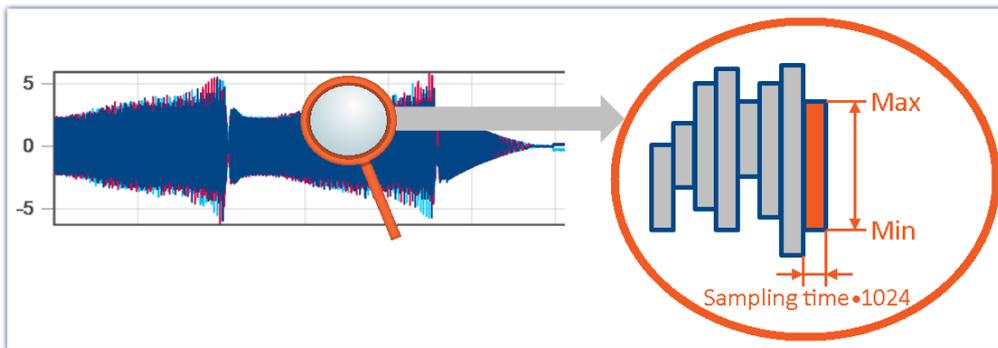
12.1.2 Envelope type monitor channel (min-/max-channel)

Some analog channels can generate a **reduced** envelope type monitor channel (min-/max-channel). The envelope ensures a complete overview of the measurement and smooth changing of the display even at a high data rate, without burdening the network with an excessive data rate.

The system cyclically condenses **x data points** down to **two new values**: the respective **minimum** and **maximum values** of the x points. The channel's data rate is thus reduced by a factor of x/2.

This reduced count of values is displayed as an envelope curve:

- With imc EOS monitor channels, the factor x is permanently set as 1024.
- With imc ARGUSfit monitor channels, the factor x can be freely specified in the [preprocessing](#) ²⁶⁴.



Envelope type monitor channels
Example of reduction by 1024 points
Resulting reduction of the data rate: 512

Envelope type monitor channels are available on the Setup page: "Analog Inputs" under the channel type: "Monitor Channels".

- The imc EOS monitor channels are by default envelope type monitor channels.
- For imc ARGUSfit monitor channels, the preprocessing must be set to "Envelope".

! Note

Signal processing

The envelope type monitor channels are not suitable for signal processing computations in, for example, imc Online FAMOS/imc Inline FAMOS!

12.1.3 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 WAVE 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:

$$t_{toset} = t_{real} \cdot \frac{S_{real}}{S_{assumed}}$$

t_{toset}	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.

$$t_{toset} = t_{real} \cdot \frac{S_{real}}{S_{assumed}}$$

$$t_{toset} = 60 \text{ s} \cdot \frac{1200 \text{ Samples/s}}{100 \text{ Samples/s}}$$

$$t_{toset} = 720 \text{ s}$$

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_{toset} 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:

$$t_{\text{toreset}} = t_{\text{real}} \cdot D_{\text{real}} / D_{\text{assumed}}$$

t_{toreset}	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] $D_{\text{assumed}} = S_{\text{assumed}} \cdot L_{\text{assumed}}$ $D_{\text{assumed}} = 100 \text{ Sample/s} \cdot 20 \text{ Bytes/Sample}$ $D_{\text{assumed}} = 2000 \text{ Bytes/s}$
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: $L_{\text{real}} = 48 \text{ Bytes/Sample}$

$$D_{\text{real}} = S_{\text{real}} \cdot (L_{\text{real}} + 12 \text{ Bytes/Sample})$$

$$D_{\text{real}} = 150 \text{ Samples/s} \cdot (48 \text{ Bytes/Sample} + 12 \text{ Bytes/Sample})$$

$$D_{\text{real}} = 9000 \text{ Bytes/s}$$

The desired buffer duration t_{real} is 1 min.

$$t_{\text{toreset}} = t_{\text{real}} \cdot \frac{D_{\text{real}}}{D_{\text{assumed}}}$$

$$t_{\text{toreset}} = 60 \text{ s} \cdot \frac{9000 \text{ Bytes/Sample}}{2000 \text{ Bytes/s}}$$

$$t_{\text{toreset}} = 279 \text{ s}$$

12.2 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](#)"^[238].

There are also local variables on the PC. These are generated using the dialog "[User-defined variables](#)"^[749].

In the next few sections 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](#)^[735]
- [Ethernet bits](#)^[737]
- [Trigger, devices and system variables](#)^[738]
- [Process vector variables \(pv-variables\)](#)^[741]
- [GPS](#)^[745]
- [User-defined variables](#)^[749]

12.2.1 Monitor channels

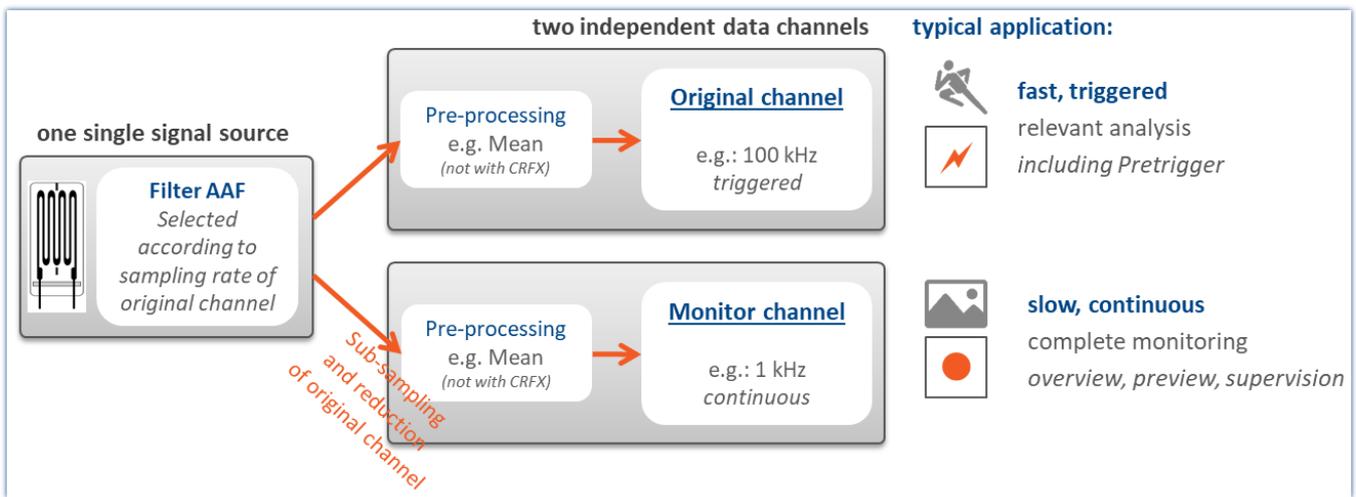
Monitor channels - For devices belonging to the [firmware group A](#)¹⁴⁶ (imc DEVICES)

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](#)"²⁶⁴", 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

Monitor channels - For devices belonging to the [firmware group B](#)¹⁴⁶ (imc DEVICEcore) - (from [B11](#)¹⁴⁶ - e.g. imc ARGUSfit)

Digital copies of the input signals of the same name for slow, continuous monitoring. Their settings for triggering, data storage and the **preprocessing** 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.

The **data reduction** is configured by the "[preprocessing](#)²⁶⁴". The sampling rate itself is **not changeable** and matches that of the original channel.

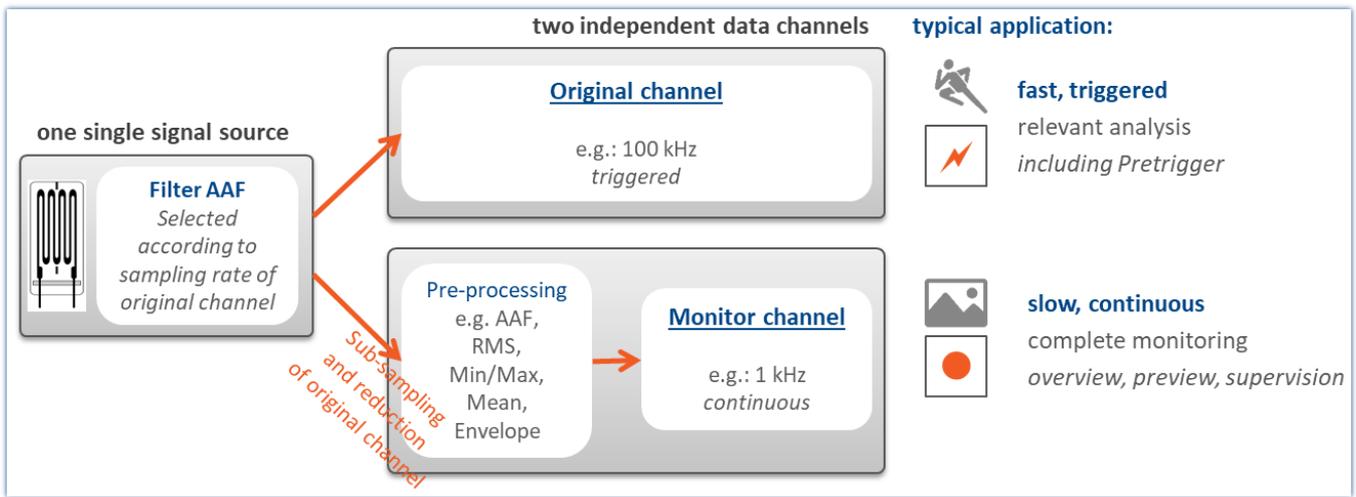
In order to **avoid aliasing effects**, the Monitor channel can be provided with its own AAF in the "[preprocessing](#)²⁶⁶". Additionally, **further data reduction** such as RMS and envelopes are possible by means of the preprocessing.

Note **Other advantages of preprocessing**

- Reduction of the data rate by means of **filtering, RMS-value, or min/max-statistics**.
- **Trigger conditions** can be defined based on the RMS-values or on the filtered values. In this way, the sound intensity, for example, can serve as the trigger source.
- **Envelope representation** allows reduction of the data rate without loss of visual information.

The following channel types can have Monitor channels: Analog inputs.

Each Monitor channel comes with its own process vector variable (pv-variable), which reflects the instantaneous value. This is performed in an analogous way to the original channel. The value is read out after preprocessing of the Monitor channel.



Concept for monitor channels

Envelope type monitor channels - applicable to imc EOS devices - [device group B10](#)¹⁴⁶

Every analog channel can **additionally return a data-reduced** envelope type monitor channel (min-/max-channel) - See "[Envelope type monitor channel \(min-/max-channel\)](#)⁷³²"

12.2.2 Ethernet bits

imc WAVE 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.

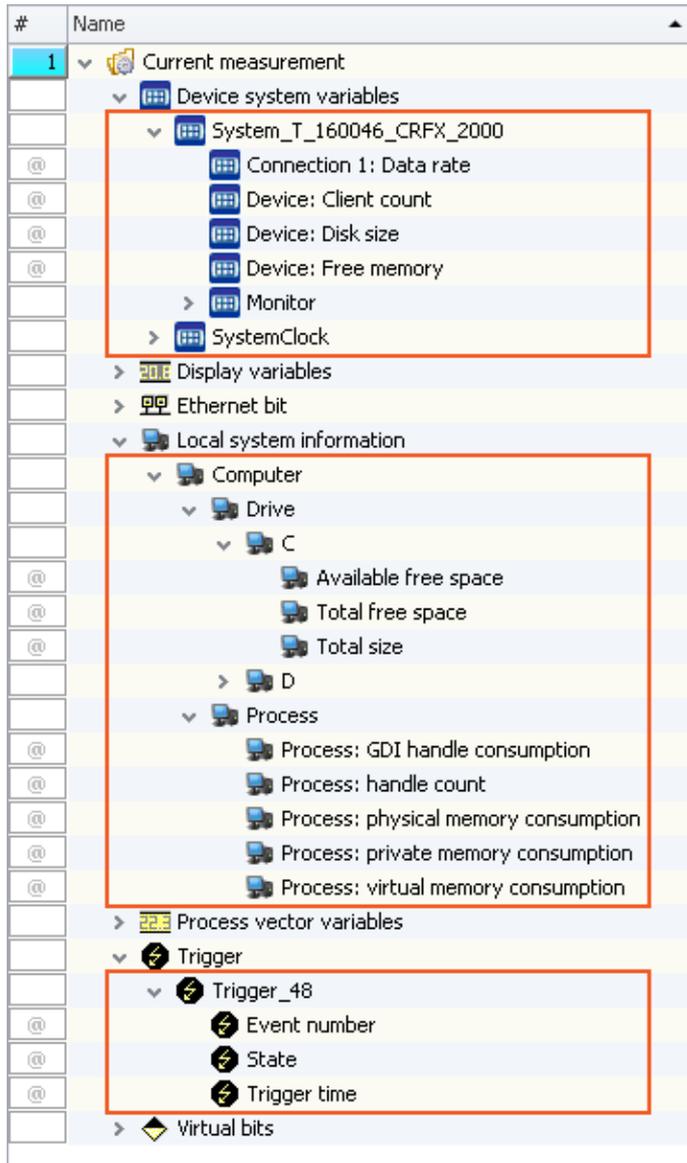
In this context, note the following:

- 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

12.2.3 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 WAVE, 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. To prevent data overflows, see the notes in these chapters/sections: " Setup pages - Configuring device " > " Information and Tips " > " Avoidance of Data Overflow ".
Critical channel name	Name of "critical channel", see " Critical channel fill level ".
Connection 2 Monitor 1	First connection with imc STUDIO Monitor (corresponds to "Connection 1").
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) <ul style="list-style-type: none"> • 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) <ul style="list-style-type: none"> • 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 WAVE: connected (1), not connected (0) <ul style="list-style-type: none"> • 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 WAVE (" VRTC ".). This time is used for all components of imc WAVE.

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.

12.2.4 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 ¹⁴⁶¹ device.
Display GPS Information ⁷⁴⁵¹	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](#) ¹⁴⁶¹ 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

You can create user-defined pv-variables from a variety of locations (prerequisite: "[imc Online FAMOS Professional](#)"⁵⁰²):

- [imc Online FAMOS](#)⁵²⁶ with Control Commands

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.

Status	Description
Active	<p>The variable exists in the device and is transferred to the PC.</p> <p>The following rules apply to user-defined pv-variables:</p> <p>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.</p> <p>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).</p> <p>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></p>
Passive	<p>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.</p> <p>For user-defined pv-variables, the following applies:</p> <p>If the initialization in imc Online FAMOS is removed, the variable is deleted from the list.</p>
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.

Deactivation of pv-variables for data acquisition channels

For devices belonging to the [firmware group A](#) ¹⁴⁶ (imc DEVICES)

By default, pv-variables are activated for the data acquisition channels (e.g. the analog channels).

Even if they are not used, they make great demands on the device system resources. Therefore, your device's entire [aggregate sampling rate](#) ²⁶⁷ is only available if the pv-variables are deactivated. Otherwise you receive messages indicating that the aggregate sampling rate has been exceeded, although this is not true mathematically.

How to **deactivate the pv-variables** for data acquisition channels is described in the section: Setup > "[Measurement options](#)" ²³⁶.

For devices belonging to the [firmware group B](#) ¹⁴⁶ (imc DEVICEScore)

Deactivation is not possible.

Names for device-pv-variables

If multiple devices are used in an experiment, a pv-variable's name is automatically extended with the device name (complete name). The name supplement is applied to the pre-defined pv-variables.

E.g. "pv.GPS.longitude_imcDev__05123456": pre-defined pv-variable for the GPS position of device "imcDev__05123456".

Data type of the pv-variables

A pv-variable has a defined data type. You can create the following data types by means of, for instance, imc Online FAMOS / Setup or Automation:

For devices belonging to the [firmware group A](#) ¹⁴⁶ (imc DEVICES)

Data type	Description	imc Online FAMOS	Other
32 bit integer	Signed integer	Initialization: <code>int pv.myPV_Int = 123</code>	Setup: 32 bit integer Automation: Integer
TI-float	Real number (floating point) Optimized for imc Online FAMOS-processing	Initialization: <code>pv.myPV_float = 123.45</code> <code>float</code> <code>pv.myPV_float = 123.45</code>	Setup: TI-float Automation: Floating point
Float	Real number (floating point); IEEE-Float e.g. from CRFX/CRXT- or CAN channels	Initialization: --- Only for application in calculations	Setup: Float Automation: ---

12.2.4.1 Update rate of pv-variables

The current measurement values for the Process Vector are updated at the respective **channel's clock rate**, but not higher than 10kHz. The filter that has been set for the channel is also used, exactly as for the channel's triggered acquisition.

Data from **imc CRONOSflex modules** are transmitted at a **maximum transfer rate** of 5 kHz (200 μ s). Data recorded at higher sampling rates are transferred in blocks. This means that pv-variables of imc CRONOSflex modules are **updated at a maximum rate of 5 kHz**. The pv-variables are thus subject to the following constraints: pv-variables from channels with sampling rates < 5 kHz can only be transferred at multiples of 200 μ s:

Channel sampling rate	Update rate of the PVV
5 kHz	5 kHz
2 kHz	1 kHz
1 kHz	1 kHz

12.2.4.2 Saving/Restoring pv-variables

Prerequisite: A [firmware group A](#) ¹⁴⁶ device.

When a measurement is concluded or the device is deactivated, the current values of pv-variables can be saved in a way so that they are restored as the initial values upon restarting the measurement.

The current values are saved to a file in the device. This storage file is in non-volatile memory and remains intact even when the device is off.

Note

- There is only one storage file in the device; separate files are not created for the various experiments!
- The decision whether a pv-variable is restored with the same name depends on the restoration strategy used.
- A fully functional UPS is required.

Declaration

To save and restore pv-variables in the device, the desired pv-variables are provided with an attribute.

In an imc Online FAMOS program, this attribute is set using the keyword `restore` with a pv-variable is declared:

```
restore pv.Counter = 1
```

Save/Restore times

The current pv-variable values are saved at the following times:

- before (re-)configuration (configuration by user PC, Diskstart/Autostart); more precisely: after data capture is stopped
- when the device is deactivated (event POWER SHUTDOWN)

The pv-variable values are restored at the following times:

- after (re-)configuration (configuration by user PC, Diskstart/Autostart); more precisely: after opening and starting the online task and before starting the measurement

Restoration strategies

The attribute has a parameter for specifying the restoration strategy.

A variety of restoration strategies can be specified:

- Matching name, data type, scaling and comment
By specifying a string (as the comment), it is possible to distinguish the same pv-variable name in different tasks (e.g. different experiments).
Note: This restoration strategy is used by the operator software "imc DEVICES" (imc Online FAMOS).
- Matching name, data type, ldx, system address, and scaling (exact matching)
Note: If the original experiment is changed, this usually results in a change of the ldx or system address, so that the values of the pv-variables are not restored!
- Matching name and comment
By specifying a string (as the comment), it is possible to distinguish the same pv-variable name in different tasks (e.g. different experiments).
Note: This restoration strategy is used by the imc STUDIO plug-in Automation.

Deleting the process vector backup file

To delete the saving file, proceed as follows:

- Open the plug-in **Setup**
- Open the page **Devices**

By means of the parameter **Process vector backup file**, you can delete the file. Add the parameter as a column in the table or use the **Details** tool window (hidden parameters must be made visible).

- Select the devices desired in the Device Table
- Click on the button *Delete* for the parameter **Process vector backup file**

What happens when you change experiments?

- The pv-variable remains intact in the device if an imc Online FAMOS variable of the same name and of the same type (Integer or Float) exists.
- If an experiment is run which does not include the variable, the variable is deleted from the process vector saving file and is thus no longer present.
- If the variable is used in the context of the imc STUDIO Automation, it remains intact as long as the Automation Task of the same name is present.

12.2.4.3 pv-variables as monitor channels

In the plug-in Panel, it is possible to link Widgets with the pv-variables. Since these are already updated following the preparation, in other words before release of a trigger, this technique is a simple alternative to monitor channels.

12.2.5 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 Online 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)

12.2.6 User-defined variables

imc WAVE provides the user with the ability to **create** and use **user-defined variables**.



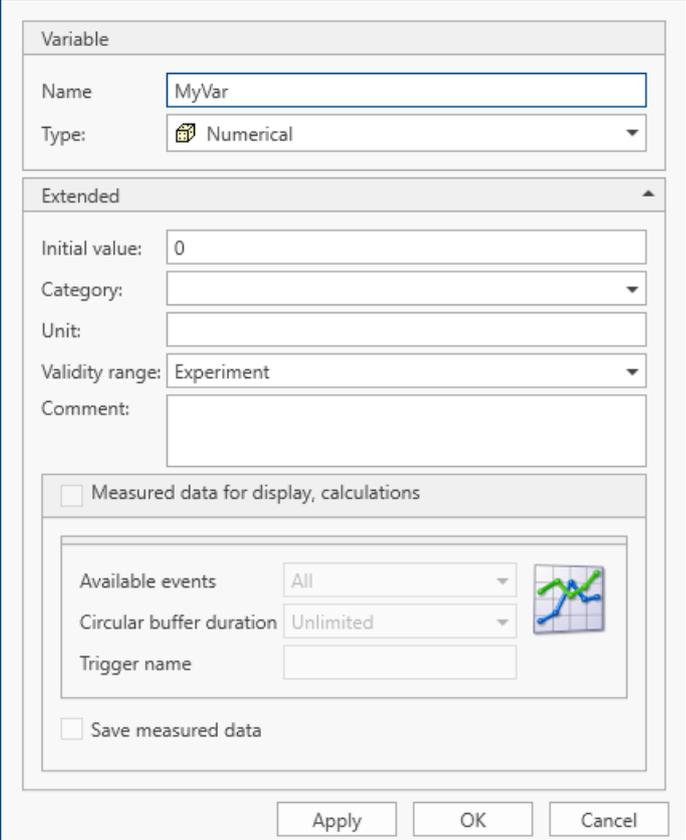
Note

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](#)¹¹⁹⁸". A dialog opens which helps you in setting up the variable.



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	<p>Here, enter a name for your variable. All alphanumeric characters are permitted.</p> <p>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.</p> <div data-bbox="461 479 774 631" data-label="Image"> </div> <p>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.</p> <p><i>Display in the Data Browser</i></p>						
Type	<p>For your user-defined variable, you can choose among various types:</p> <table border="1"> <thead> <tr> <th>Typ</th> <th>Description</th> </tr> </thead> <tbody> <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> </tbody> </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	<p>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"¹¹⁹⁸), the value entered here is written in the variable.</p> <p>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.</p>

Parameter	Description
Category	<p>Assigning a category gives you the ability to structure the variables in the Data Browser.</p> <p>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.</p> <p>Custom variables created in custom main category</p> <p>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.</p> <p>If you wish to create a category within the category "<i>User-defined variables</i>", do the following: write a "\" before the category name.</p> <hr/> <p> 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.</p> <p> Category "\"MeasurementPoint_1". This creates a category "MeasurementPoint_1" within the category "<i>User-defined variables</i>". It contains the variable.</p> <hr/> <p>This applies, for example, to variables created by means of the command: "Load variable"¹¹⁹⁴ or to generating a user-defined variable by means of the Data Browser.</p>
Unit	You can assign a unit to your variables.

Parameter	Description												
Validity range	<p>The setting Validity Range determines the availability of your variable. What is saved is always only the variable itself, not the current value!</p> <table border="1"> <thead> <tr> <th>Validity Range</th> <th>Description</th> </tr> </thead> <tbody> <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> </tbody> </table> <p>Any "higher-level" variables are not overwritten if a variable of the same name but from a different validity range is loaded.</p> <p> 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.</p> <p> Please also observe the notes on the action: "Load/Refill variable" in regard to the "Validity range".</p>	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.												

12.3 Data file format - Saving measured data and export/import formats

With imc WAVE, 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.

12.3.1 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 ⁷⁵³	Compressed format for many fieldbus variables.
CANAllyse-MDF ⁷⁵⁴	CANAllyser-MDF format for CAN-Log data.

Note

"imc FAMOS ZIP" and "CANAllyse-MDF": Applicable with devices of the [firmware group A](#) ¹⁴⁶ (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 WAVE.

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 WAVE (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](#)" ⁷⁸⁶.

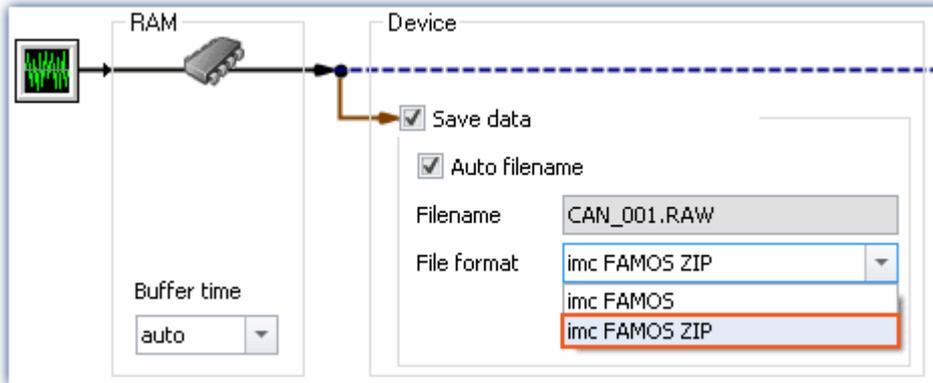
imc FAMOS ZIP

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

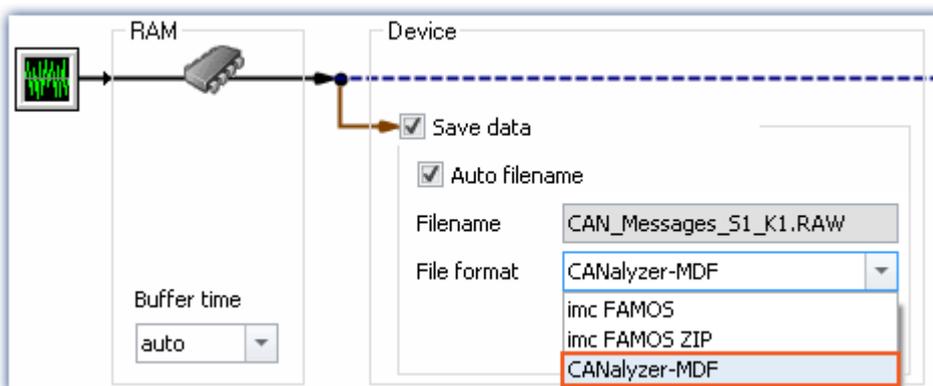
The compression of the data is limited by the computing power of the basic system. An **overload due to too many individually compressed channels can lead to a restart of the system!** The **number of channels** is the most important factor. Typical measurements with an imc BUSDAQ-4 showed that up to **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.

12.3.2 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 byte* + 1 x 2000 x 2 byte = 5,5 KB => 8 KB 8 KB

1536 byte* + 1 x 2000 x 2 byte = 5,5 KB => 8 KB 8 KB

1536 byte* + 1 x 2000 x 4 byte = 9,5 KB => 16 KB 16 KB

Next events:

512 byte* + 1 x 2000 x 2 byte = 4,5 KB => 8 KB => 99 x 8 KB 792 KB

512 byte* + 1 x 2000 x 2 byte = 4,5 KB => 8 KB => 99 x 8 KB 792 KB

512 byte* + 1 x 2000 x 4 byte = 8,5 KB => 16 KB => 99 x 16 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)

12.3.3 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

Import a parameter set file in order to configure the measurement device. You can save multiple parameter sets and import a particular one, 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.

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. An additional documentation possibility is a [report](#)¹⁸¹. Separate reports can be created with various configurations.



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.

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.



Note

Importing variables

For importing variables' values, instead of the command: "*Import parameter*", the command "[Load/Refill variable](#)"¹¹⁹⁴ 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](#)"⁷⁶²).
- **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](#)"⁷⁶²)
- If only one row exists in a target table, 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	<p>For all parameters (all table descriptions), the settings are imported. The assignment is made on the according to the name.</p> <p>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.</p> <p>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.</p>
Uniformly per module type	<p>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").</p> <p>Only one parameter set per module type is required.</p>
Multiple-device operation (Firmware group A)	<p>For all channels/variables, the settings are imported. The assignment is made according to the connector designation and the device serial number.</p> <hr/> <p>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.</p> <p> Cannot be used with devices of firmware group B because the connector does not distinguish between the main channel and the monitor channel.</p>
Single-device operation (Firmware group A)	<p>For all channels/variables, the settings are imported. The assignment is made according to the connector designation.</p> <hr/> <p>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.</p> <p> Cannot be used with devices of firmware group B because the connector does not distinguish between the main channel and the monitor channel.</p>
...	Select a user-defined mapping. See " Mapping - Modifications of the default behavior ^[775] ".

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 ^[775] ".

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	<p>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.</p> <p>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></p> <ul style="list-style-type: none"> • Confirm with "Yes" <p>When you close EXCEL, another message appears: <i>"Do you wish to save changes?"</i></p> <ul style="list-style-type: none"> • Select "Don't save" <p>The following formats are not appropriate since the automatically save with the separator character ";":</p> <ul style="list-style-type: none"> • CSV (separator character) (*.csv) • CSV (Macintosh) (*.csv) • CSV (MS-DOS) (*.csv) <p>Known problems:</p> <p>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.</p>
Editing a csv-file by means of OpenOffice or LibreOffice	<p>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".</p> <p>If you save the file after having made changes, a message appears asking for the data saving format.</p> <ul style="list-style-type: none"> • Click on "Keep current format" <p>Known problems:</p> <ul style="list-style-type: none"> • 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.

12.3.3.1 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 ⁷⁶⁰).
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. <ul style="list-style-type: none"> • Basic Settings⁷⁶⁶: Here you find prepared export settings, for instance, one where only all values of the analog channels are exported. • Manual Settings⁷⁶⁷: 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: <ul style="list-style-type: none"> • 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. <ul style="list-style-type: none"> • 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 ⁷⁶²).

Manual Settings

Tab	Description
-----	-------------

Datapool

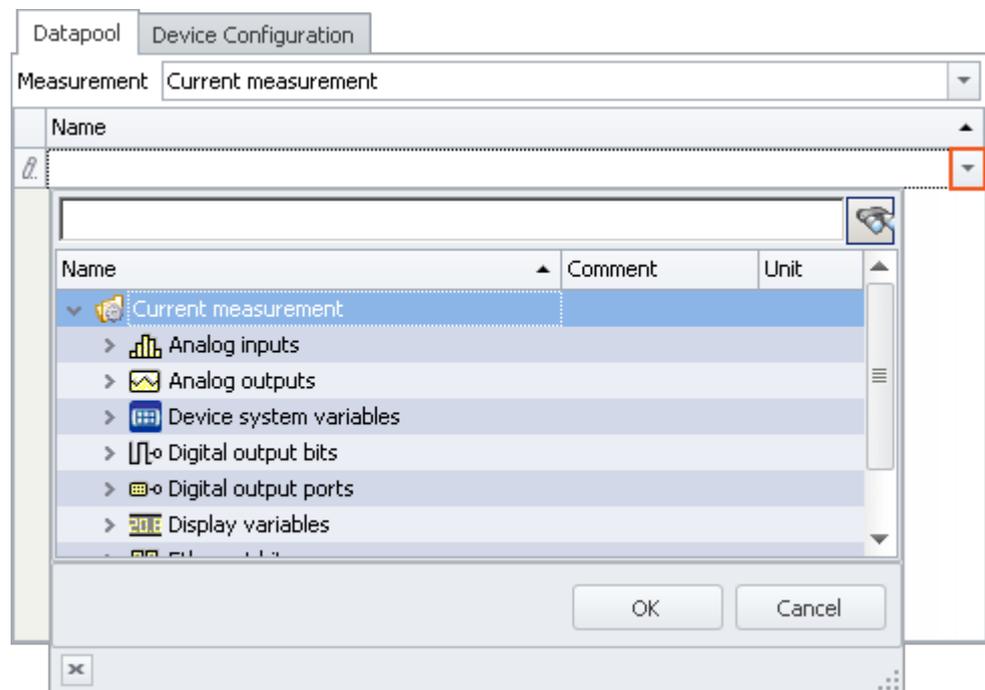
Selection of the variables to be saved.

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](#)" () to update the list of variables it contains.



List of the variables to export

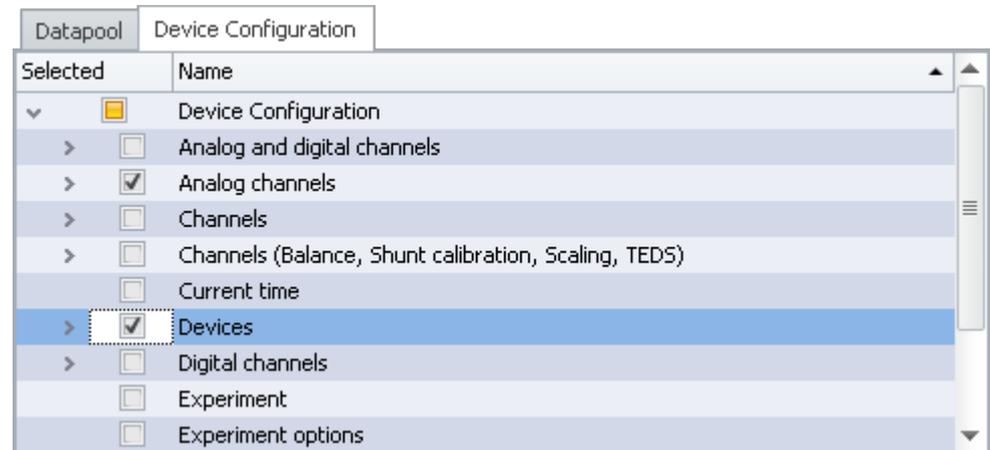
Select all desired variables (multi-selection is also allowed) and click on the button *OK*. All selected variables are added to the list of variables to export.

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.

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

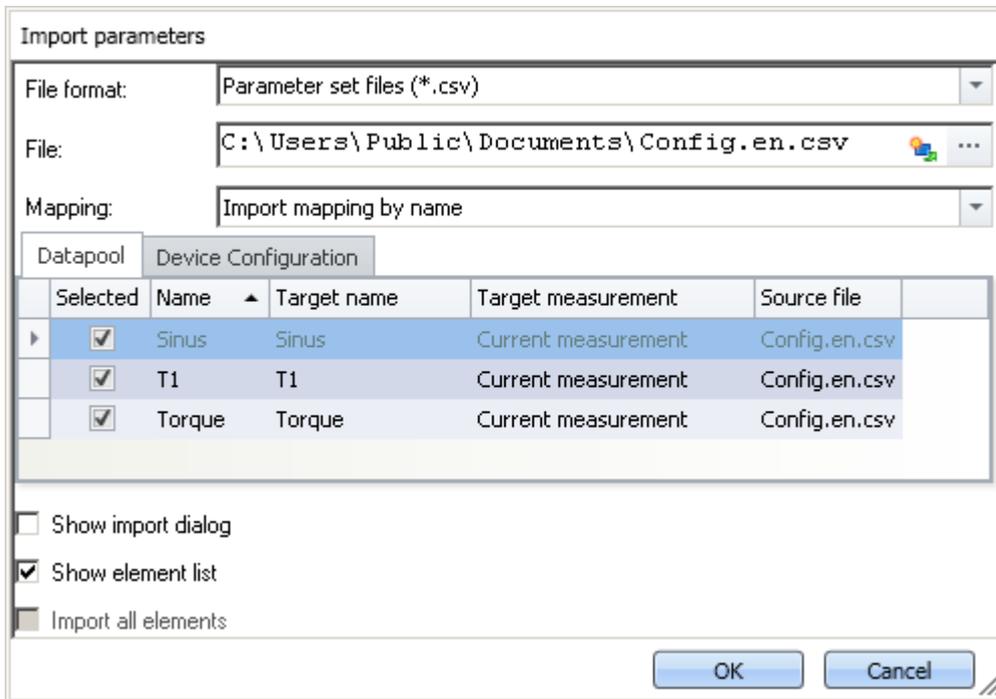
You perform balancing for Channel_001 with the measurement range 10 V. Next, you perform another balancing for the 5 V measurement range and export the result also to the file Balance.en.csv; the file contains both balancing values.

Upon importing, the appropriate balancing settings are imported in accordance with the measurement range set.

12.3.3.2 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**" 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**".

Option	Description
File format	Selection of the file format. For more information, see the description of the parameter set (File formats).
File	Specification of the file to import, including the pathname.

Option	Description
Mapping	<p>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.</p> <p>An unsuitable assignment can cause either no parameters or incorrect ones to be imported!</p> <p>For more information, see further below under: "Mapping for import".</p>
Show export dialog and Show element list	<p>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:</p> <ul style="list-style-type: none"> • Show export dialog: File options (open-dialog) • Show element list: Datapool (variables) and device configuration. <p>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.</p>
Import all elements	<p>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.</p> <p>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.</p>

Tab	Description																														
Datapool	<p>Selection of the variables to be saved.</p> <p>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.</p> <table border="1" data-bbox="459 1256 1434 1485"> <thead> <tr> <th colspan="2">Datapool</th> <th colspan="4">Device Configuration</th> </tr> <tr> <th></th> <th>Selected</th> <th>Name ▲</th> <th>Target name</th> <th>Target measurement</th> <th>Source file</th> </tr> </thead> <tbody> <tr> <td>▶</td> <td><input checked="" type="checkbox"/></td> <td>Sinus</td> <td>Sinus</td> <td>Current measurement</td> <td>Config.en.csv</td> </tr> <tr> <td></td> <td><input checked="" type="checkbox"/></td> <td>T1</td> <td>T1</td> <td>Current measurement</td> <td>Config.en.csv</td> </tr> <tr> <td></td> <td><input checked="" type="checkbox"/></td> <td>Torque</td> <td>Torque</td> <td>Current measurement</td> <td>Config.en.csv</td> </tr> </tbody> </table>	Datapool		Device Configuration					Selected	Name ▲	Target name	Target measurement	Source file	▶	<input checked="" type="checkbox"/>	Sinus	Sinus	Current measurement	Config.en.csv		<input checked="" type="checkbox"/>	T1	T1	Current measurement	Config.en.csv		<input checked="" type="checkbox"/>	Torque	Torque	Current measurement	Config.en.csv
Datapool		Device Configuration																													
	Selected	Name ▲	Target name	Target measurement	Source file																										
▶	<input checked="" type="checkbox"/>	Sinus	Sinus	Current measurement	Config.en.csv																										
	<input checked="" type="checkbox"/>	T1	T1	Current measurement	Config.en.csv																										
	<input checked="" type="checkbox"/>	Torque	Torque	Current measurement	Config.en.csv																										

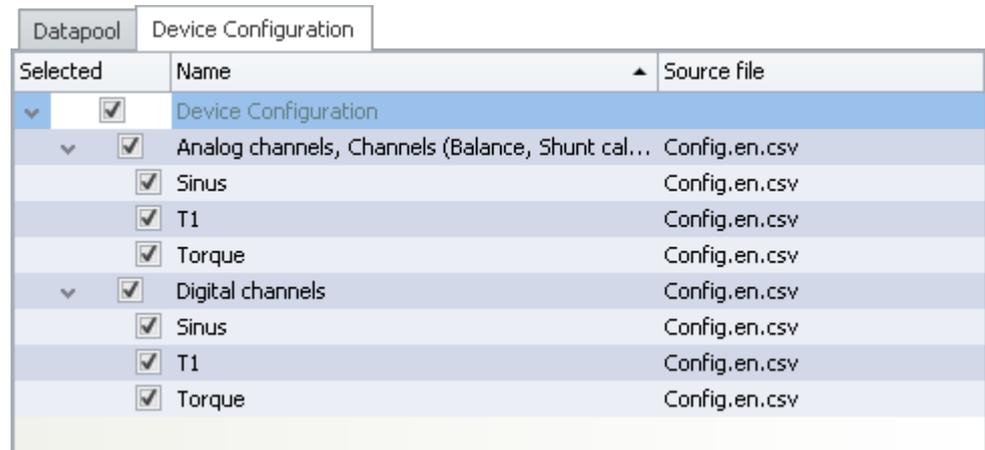
List of variables to import

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.



Selected	Name	Source file
<input checked="" type="checkbox"/>	Device Configuration	
<input checked="" type="checkbox"/>	Analog channels, Channels (Balance, Shunt cal...	Config.en.csv
<input checked="" type="checkbox"/>	Sinus	Config.en.csv
<input checked="" type="checkbox"/>	T1	Config.en.csv
<input checked="" type="checkbox"/>	Torque	Config.en.csv
<input checked="" type="checkbox"/>	Digital channels	Config.en.csv
<input checked="" type="checkbox"/>	Sinus	Config.en.csv
<input checked="" type="checkbox"/>	T1	Config.en.csv
<input checked="" type="checkbox"/>	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	0.1000001 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	<p>For all parameters (all table descriptions), the settings are imported. The assignment is made on the according to the name.</p> <p>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.</p> <p>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.</p>
Uniformly per module type	<p>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").</p> <p>Only one parameter set per module type is required.</p>
Multiple-device operation (Firmware group A)	<p>For all channels/variables, the settings are imported. The assignment is made according to the connector designation and the device serial number.</p> <hr/> <p> 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.</p> <p>Cannot be used with devices of firmware group B because the connector does not distinguish between the main channel and the monitor channel.</p>
Single-device operation (Firmware group A)	<p>For all channels/variables, the settings are imported. The assignment is made according to the connector designation.</p> <hr/> <p> 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.</p> <p>Cannot be used with devices of firmware group B because the connector does not distinguish between the main channel and the monitor channel.</p>
...	Select a user-defined mapping. See " Mapping - Modifications of the default behavior ^[775] ".



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](#)"^[774]).
- 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.

12.3.3.3 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 <ol style="list-style-type: none"> 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, <ol style="list-style-type: none"> 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>	<p>All contents of the table description and/or all variables of the channel type: "analog inputs" are exported.</p> <p>In this case, there is no table description of the name "analog inputs".</p> <p>However, there are variables of the channel type "analog inputs"</p>
<code>SELECT * WHERE Caption = 'analog inputs' OR Caption = 'analog channels'</code>	<p>All contents of the table description and/or all variables of the channel type: "analog inputs" or "analog channels" are exported.</p> <p>In this case, there is a table description of the name "analog channels".</p> <p>And there are also variables of the channel type "analog inputs"</p>
<code>SELECT * WHERE Caption = 'analog channels'</code>	<p>All contents of the table description and/or all variables of the channel type: "analog channels" are exported.</p> <p>In this case, there is a table description of the name "analog channels".</p> <p>However, there is no variable of the channel type "analog channels".</p>
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>
```

12.4 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. Furthermore, it is possible to save the monitor channels of imc EOS and imc ARGUS*fit* in the [envelope type format](#)⁷³².

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 WAVE 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.

12.5 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](#)**^[1172] and **[Format Converter](#)**^[1157].

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.

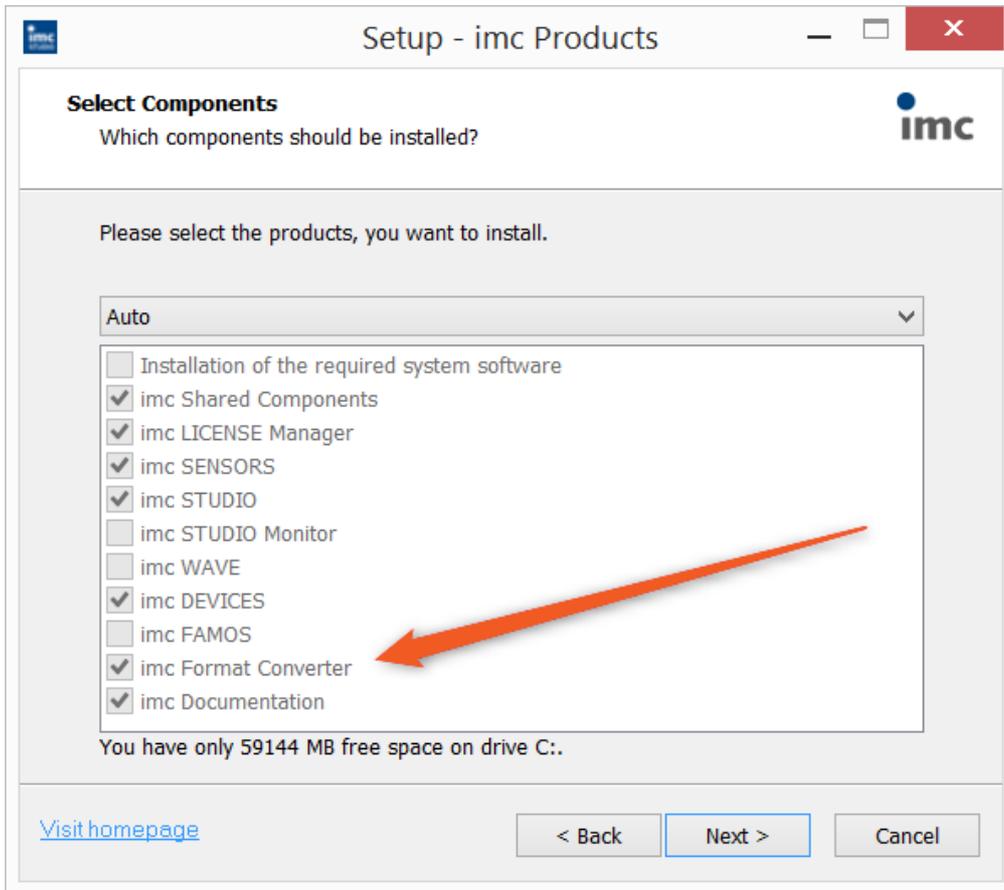
Export formats

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 ATEX, ASAM ATEX 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)

12.5.1 Installation

The installation file is located on the installation medium and is included in the installation by default.



Separate installation

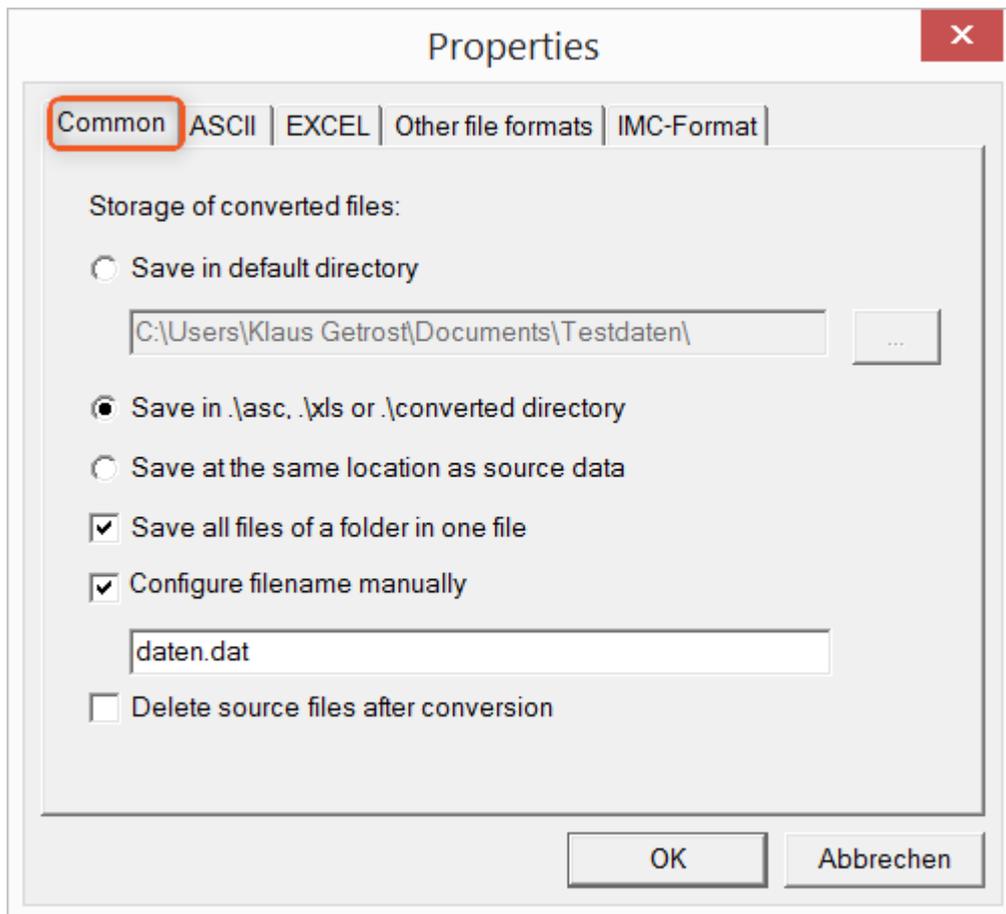
Select the installationvariant: "User-defined" and select there only the imc Format Converter.

12.5.2 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 re saved on the same **level of the tree diagram as the source data, without subfolders**.

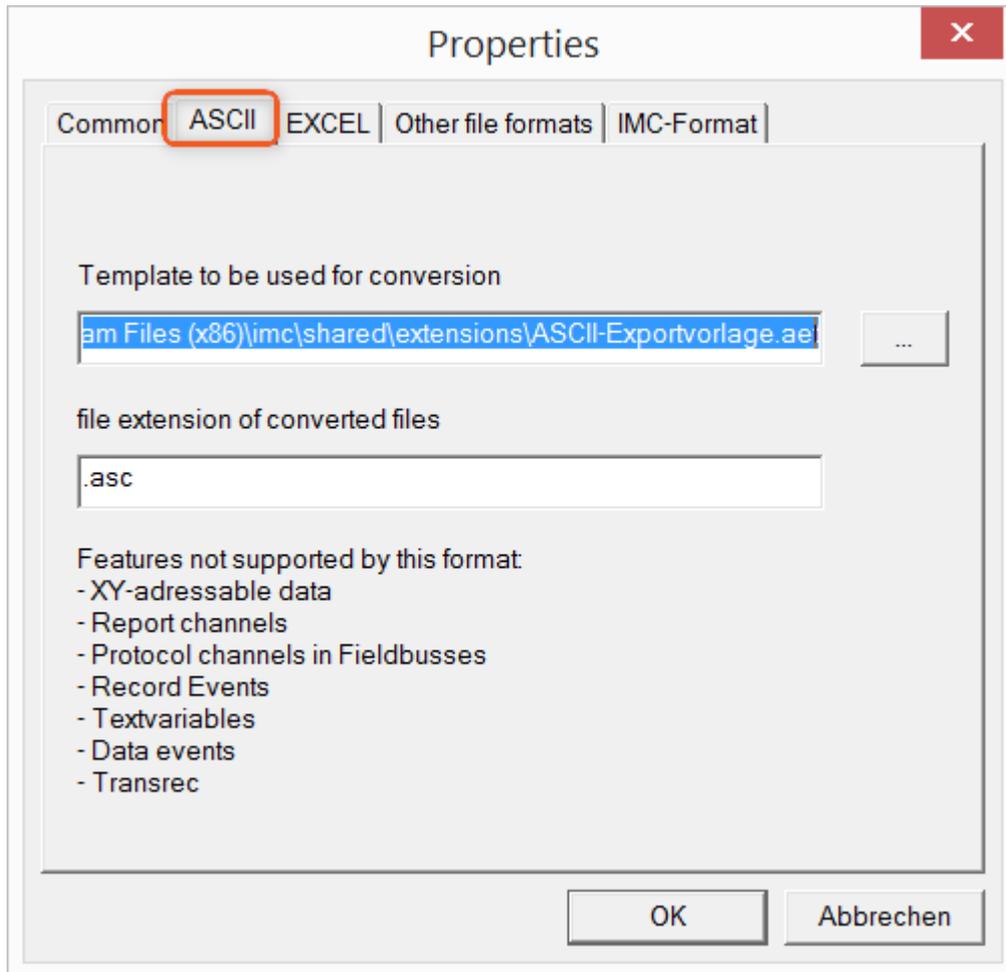
Manner of data saving:

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.

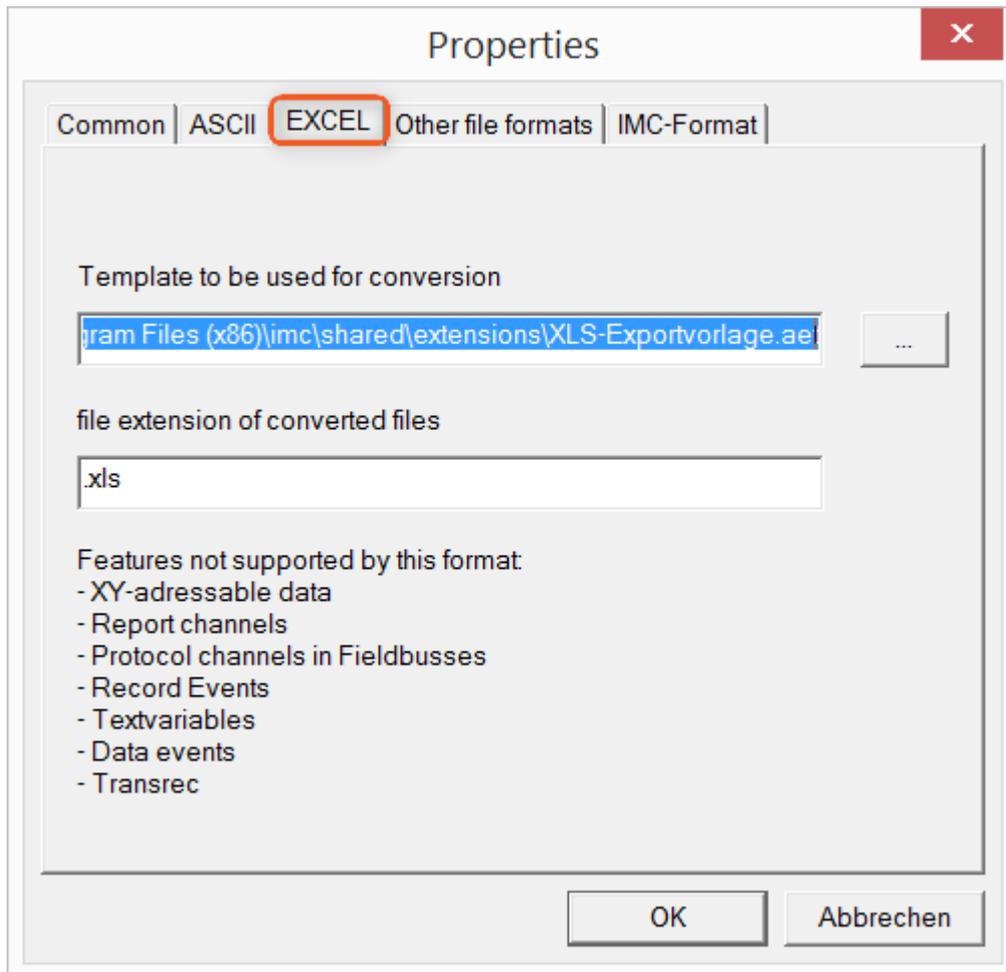
Special variable types which can **not** be exported as ASCII data are listed on this tab page.

! 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](#)⁷⁸⁹. By default, the file extension is "XLS", but can be arbitrarily specified here.

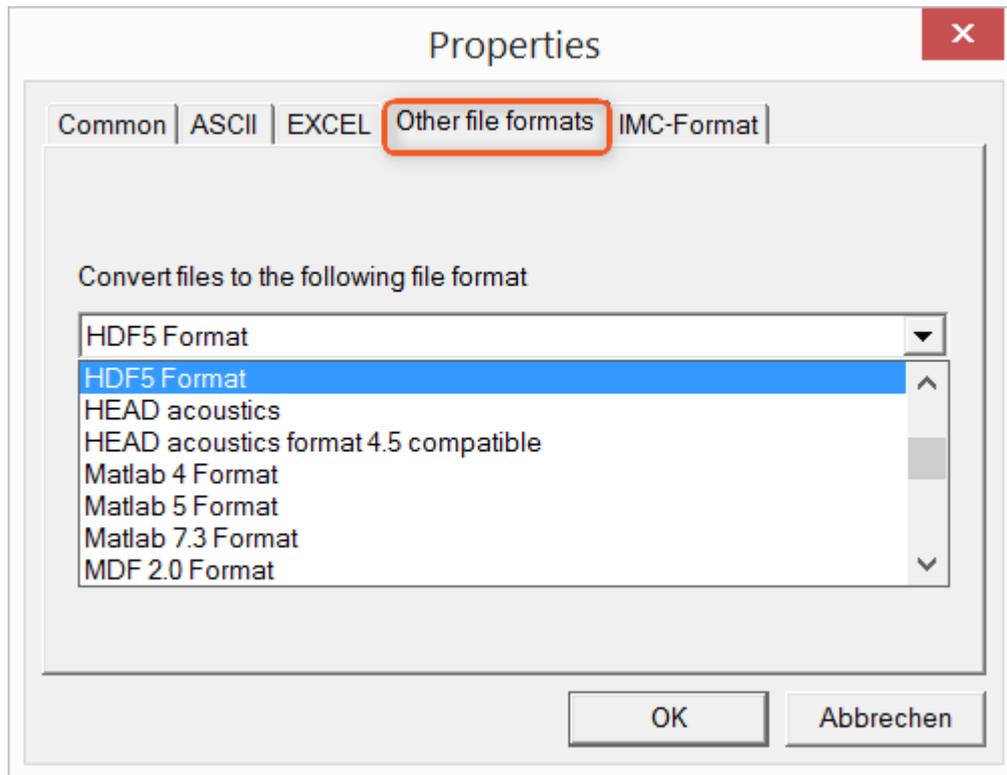
Variable types which are **not** exported correspond to those for [ASCII-export](#)⁷⁸⁹.

! 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



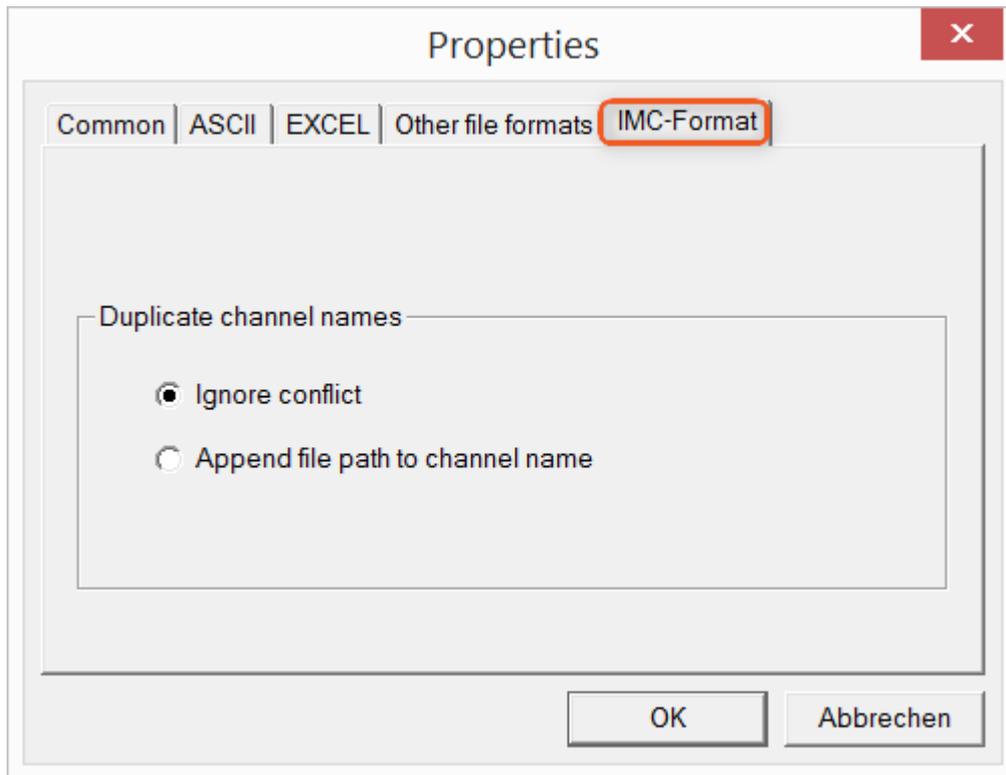
Convert files to the following file format:

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](#)⁷⁸⁸" to accomplish this would be:

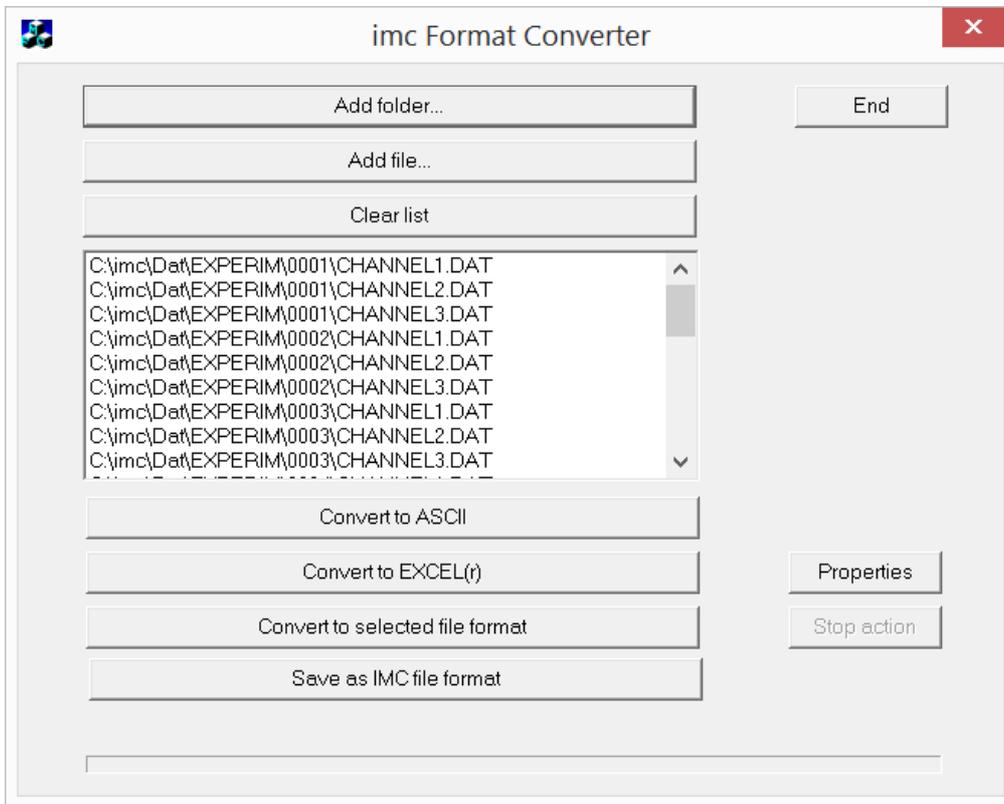
Save at same location as source data = active

Save all files of a folder in one file = active

Delete source files after conversion = active

12.5.3 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...

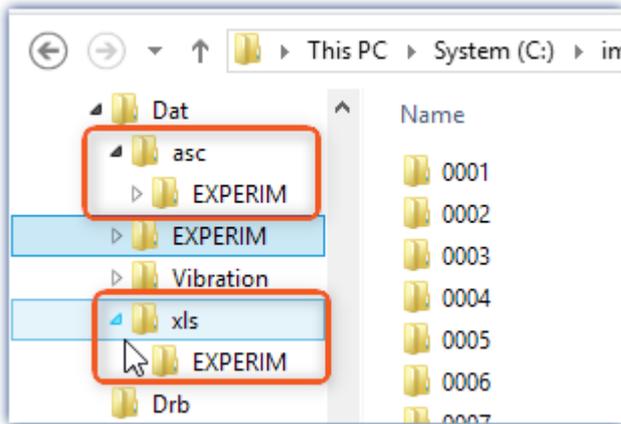
Selection of individual files.

Clear list

All entries are deleted from the Files list.

Convert to ASCII/EXCEL(r)

The standard variant for converting data to [ASCII](#)⁷⁸⁹ or [EXCEL](#)⁷⁹⁰-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](#)⁷⁹¹". 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](#)⁷⁸⁸.

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 page](#)⁷⁹².

12.5.4 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](#) ⁷⁹⁵.

Command line options

	Description	Example
-a	conversion to "ASCII"	<code>imcFrmtCvrt.exe -a <files or paths></code>
-e	conversion to "EXCEL"	<code>imcFrmtCvrt.exe -e <files or paths></code>
-k	conversion to "Other data formats"; The format selected in the settings is used, e.g. MDF	<code>imcFrmtCvrt.exe -k <files or paths></code>
-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.	<code>imcfrmtcvrt -i -s d:\setup.xml</code> <code>imcfrmtcvrt -i d:\setup.xml <files or paths></code>
-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	<code>imcfrmtcvrt -i -h d:\setup.xml <files or paths></code>
-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	<code>imcfrmtcvrt -i -h -l d:\setup.xml d:\Logfile.log <files or paths></code>

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 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
```

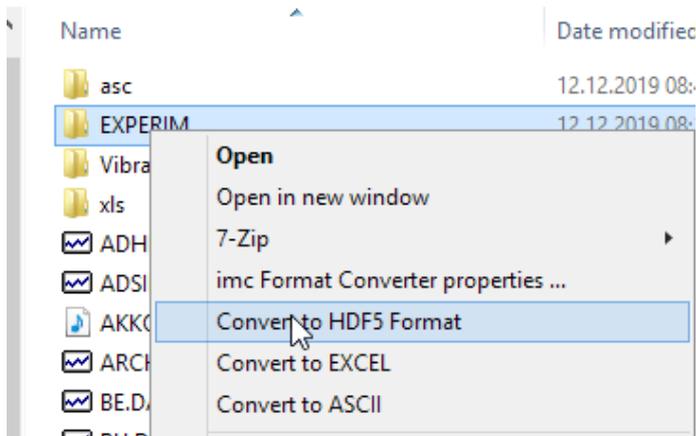
All files specified in the file: "*d:\daten.xml*" are converted.

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*".

12.5.5 Conversion via the Windows Explorer

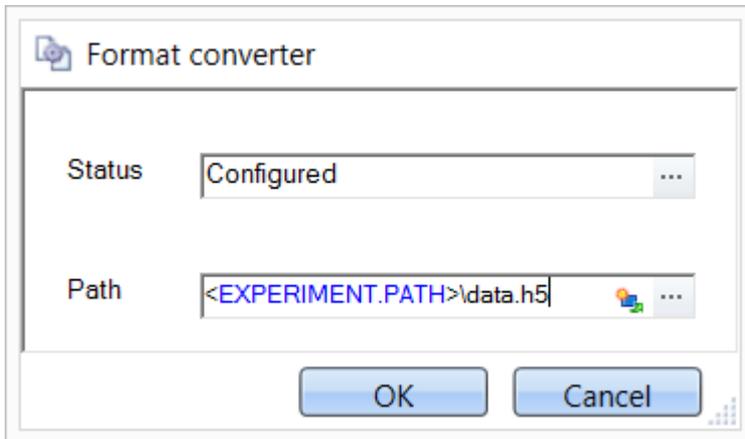
Installation of the imc Format Converter expands the Windows Explorer's context menu. The current templates for export of [ASCII](#)^[789] and [EXCEL](#)^[790], as well as the export format last selected on the page "[Other file formats](#)^[791]" can then be applied directly to a folder or files:



12.5.6 Format Converter as imc STUDIO command

Calling

Via the Sequencer: "External calls" > "Format converter":



Configuration of the command: Format Converter

Status

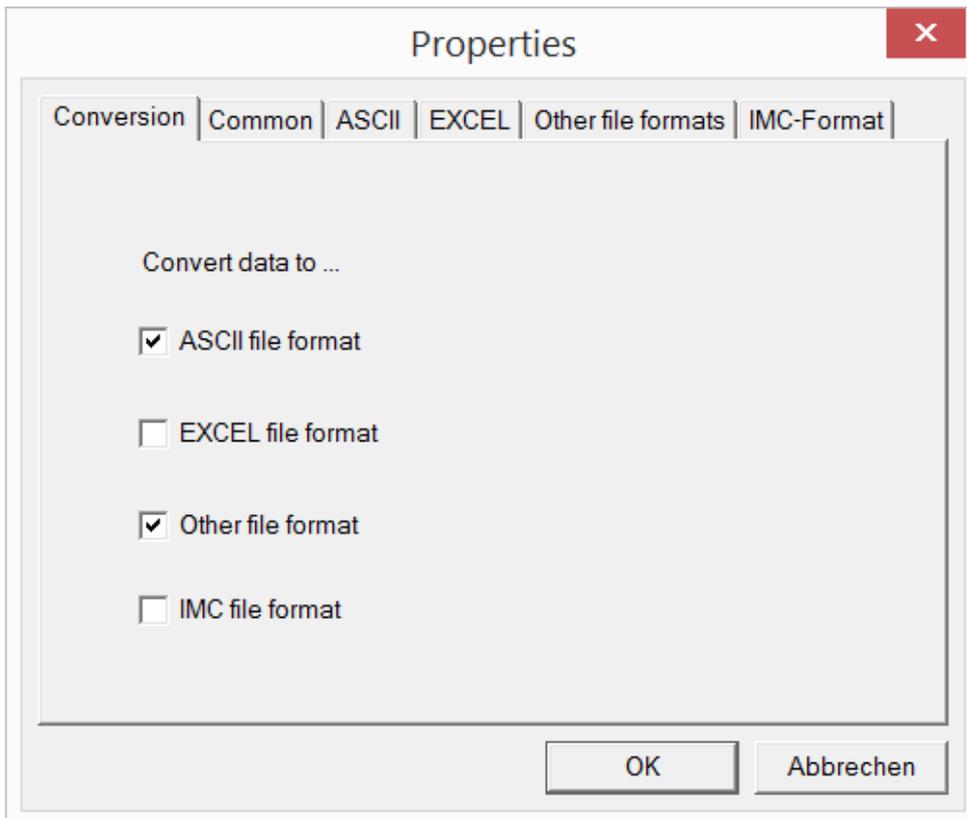
Immediately upon opening the command's configuration, the status is set to "not configured".

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](#) ⁷⁸⁸ 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](#)" ⁷⁸⁸ (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](#)⁷⁸⁸: Storage location and whether individual files or a multi-file is to be created.
- [ASCII](#)⁷⁸⁹: Selection of the export template and specifying of the file extension.
- [EXCEL](#)⁷⁹⁰: Selection of the export template and specifying of the file extension xls or.xlsx.
- [Other file format](#)⁷⁹¹: Selection of the format and if necessary its formatting options.
- [IMC-file format](#)⁷⁹²: Save the data to a file in the imc format, or as a copy to a different memory location.



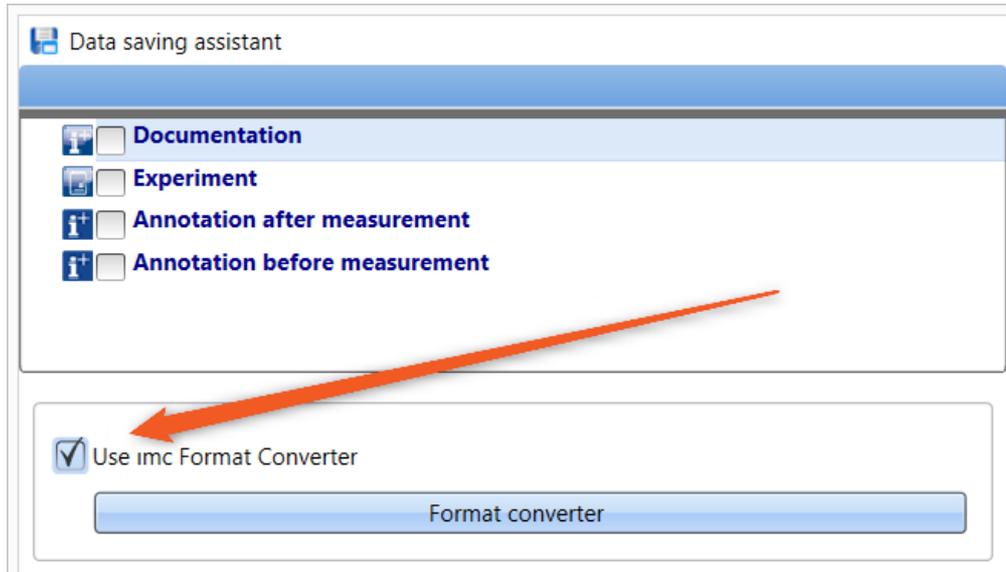
Setting the Format Converter

12.5.7 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](#)⁷⁹⁷.

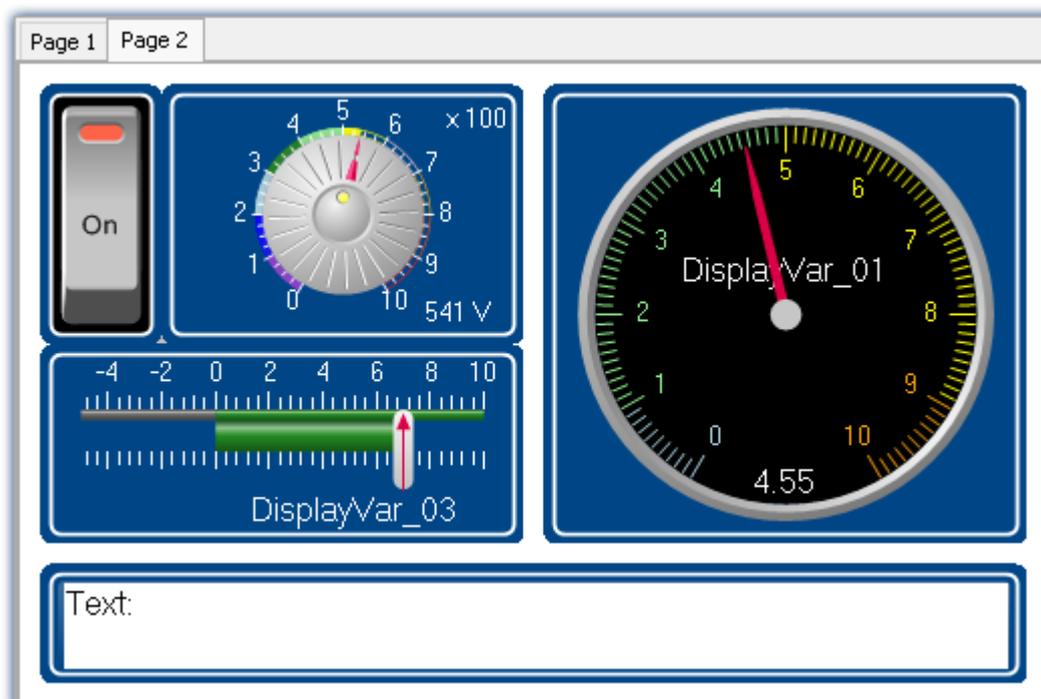


Format Converter as component of the Data Saving Assistant

13 Panel

Panel is the imc WAVE 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](#)^[1120]), 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](#)^[1120] optimized for on-screen display
- [Report pages](#)^[1120] optimized for printouts

Widgets

Various Widgets are available as instruments and controls. The amount of Widgets varies by the product configuration. Even the [curve window](#)^[850], 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 \(commands\)](#)^[848] to Widgets such as switches and buttons.

Operation

For the [positioning of Widgets](#)^[821] and the [assignment of channels and variables](#)^[1133], a variety of operation techniques can be used, such as Drag&Drop or context menus.

The Widgets can be operated by keyboard or mouse.

13.1 Tool windows

The following tool windows are available for the purpose of configuring the Panel pages:

Tool Windows	Description
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 ^[1133] for a Widget.
Properties ^[802]	The Properties window changes its contents in response to the object selected. If no Widget is selected, the page properties ^[1122] are displayed.

13.1.1 Properties (general)

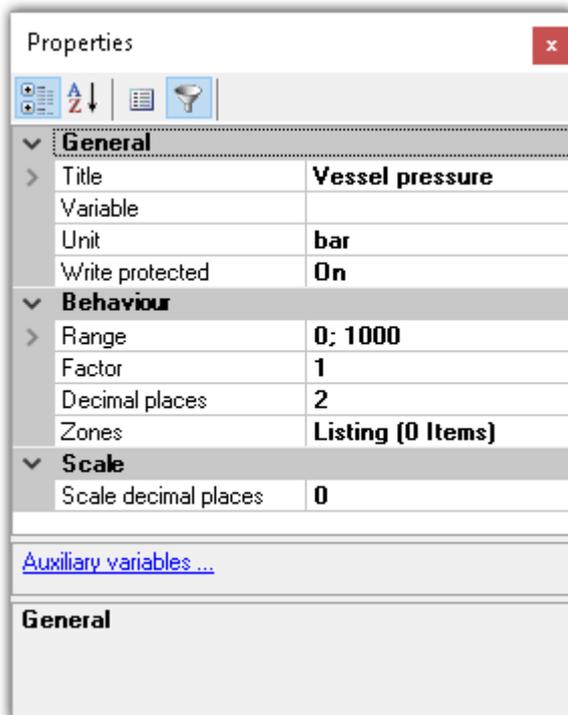
In the tool window "**Properties**" (via the context menu of the selected Widget, (see [context menus of the Widgets](#) ^[822])), the individual Widgets or the Panel pages can be adapted. The content of the Properties window depends on the respective element.

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.

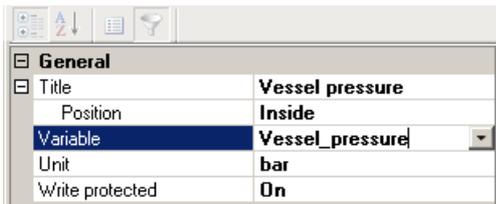


Display of all properties.



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)



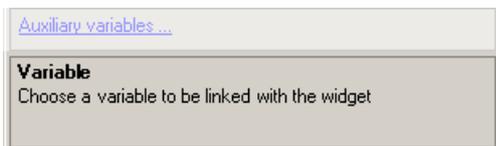
Opens a dialog (for instance Zones-dialog or event)

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.

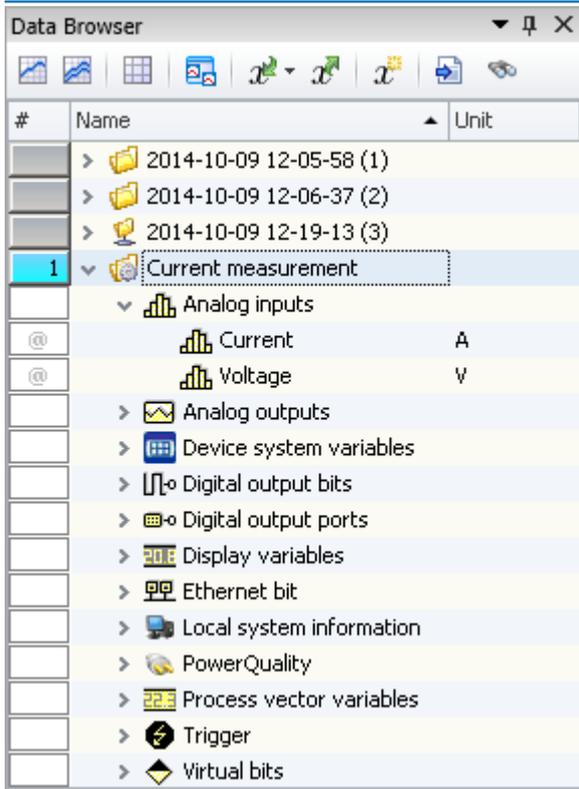
13.1.2 Data Browser

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](#)^[1134] to a Panel page and link them with Widgets. You can also [navigate through measurements](#)^[810] and [assign measurement numbers](#)^[809].



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](#)"^[171] in the documentation on the component Setup.



Data Browser in the Panel

Structure

The Data Browser has a [toolbar](#)^[811] 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 " ^[809]).
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 " ^[807]). 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 " ^[173]).



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: " <i>Setup - Device Configuration</i> " > " <i>Configuring Channels and Variables</i> " > " Channel table ".
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 ".
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 ".
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	<p>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.</p> <p>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".</p>

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	<p>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".</p> <hr/> <p>Notes on circular buffer memory</p> <p> 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.</p>
 Saved measurements	<p>The measurements are arranged as list entries on the same level as Current Measurement.</p> <p>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".</p> <p>(year-month-day hour-minute-second)</p>

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

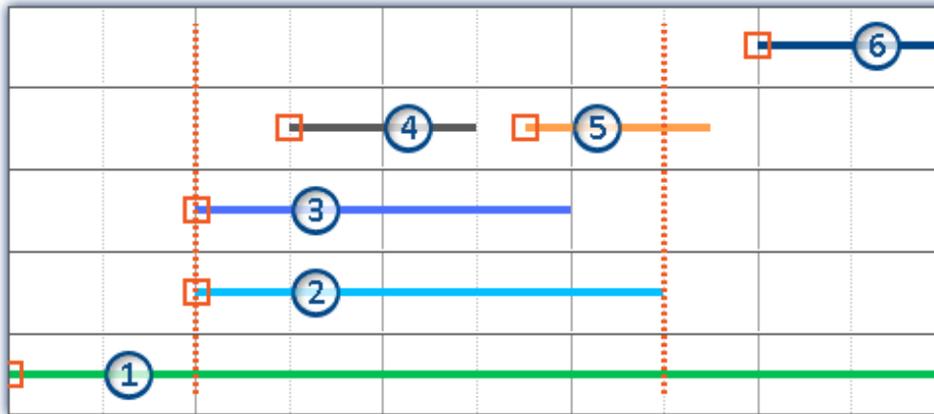
When a defined event is selected, the system will make every attempt to display as many channels as possible. The channels displayed have some time overlap.

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: 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

13.1.2.1 Toolbar

Toolbar: Variable

Menu item	Description
 Show individually	All selected channel variables (e.g. analog inputs) are opened in separate "free-floating" curve windows ^[1102] . All other selected variables (e.g. Display variables) are added to the window " Current values " ^[816] .
 Show together	All selected channel variables (e.g. analog inputs) are opened in a shared "free-floating" curve window ^[1102] . All other selected variables (e.g. Display variables) are added to the window " Current values " ^[816] .
 Show current values	Opens the dialog Current values ^[816] , 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 " ^[811])
 Export variable	Exports variables' values to a file. (See " Export variable " ^[1192])
 Create user defined variable	Creates user-defined variables, including single value- and text-variables. (More info is provided in the section " User-defined variables " ^[749])
 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.

13.1.2.1.1 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*".

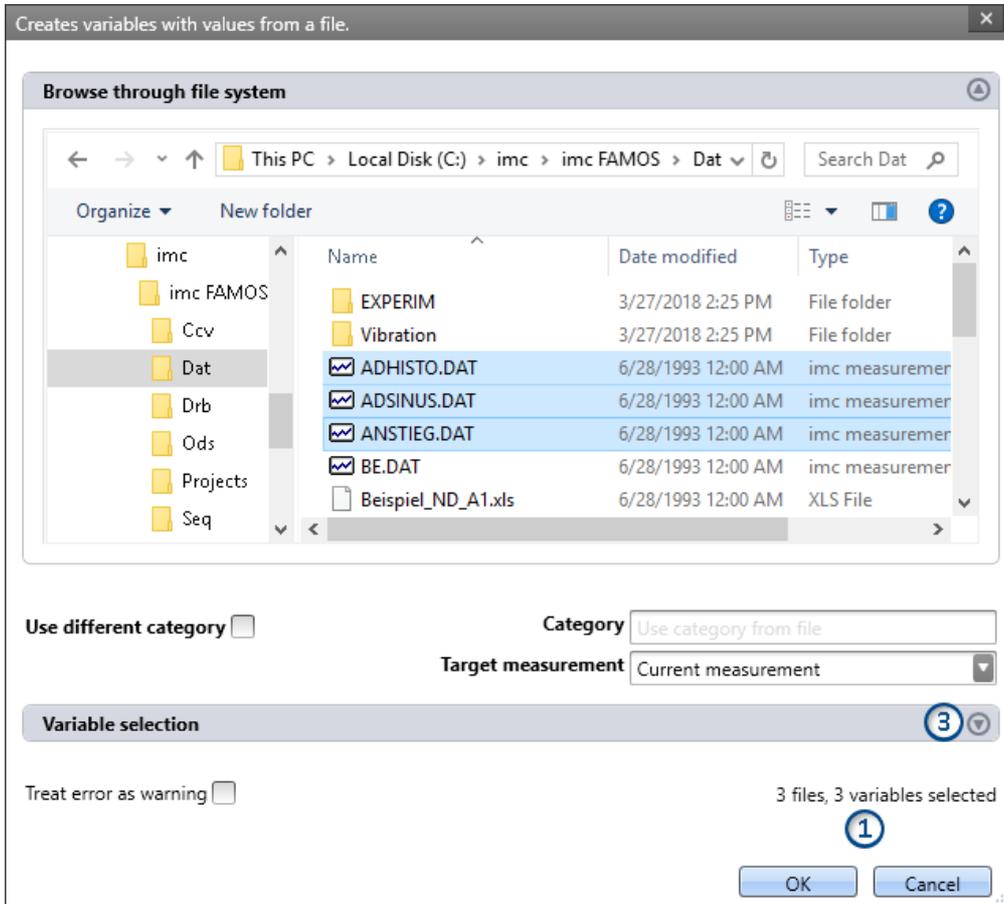
In contrast to imc FAMOS, there are variables which cannot be overwritten, e.g. channels or pv-variables. The reason is that, for instance, these variables belong to the specific device, or possess specific data formats and other properties which must be retained. If the user attempts to overwrite them, 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	<p>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.</p> <ul style="list-style-type: none"> • 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". <p>If you wish to create a category within the category "User-defined variables", do the following: write a "\" before the category name.</p> <hr/> <p>Category: "MeasurementPoint_1". A category called "MeasurementPoint_1" is created in parallel with the other categories, such as "Analog Inputs". It contains the variable.</p> <p> Category "\"MeasurementPoint_1". This creates a category "MeasurementPoint_1" within the category "User-defined variables". It contains the variable.</p> <hr/> <p>This applies, for example, to variables created by means of the command: "Load variable" or to generating a user-defined variable^[749] by means of the Data Browser.</p>
Target measurement	Description
Current measurement	<p>Load: The variable appears in the Data Browser under "Current measurement".</p> <p>Refill: The variable overwrites the value of any variable existing under "Current measurement".</p>
Last completed measurement	<p>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.</p>
Measurement number Measurement#<No>	
Fixed measurement name	
Other options	Description
Treat error as warning	<p>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.</p> <p>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.</p>

13.1.2.2 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	<p>Only in the Numbering column.</p> <p>Opens an input box for entering the desired lowest measurement number for the subsequent selection.</p> <p>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.</p>
 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 " ^[749])
 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	<p>Opens the Editor for changing the properties of user-defined variables. The dialog matches that of "<i>Create variable</i>". With some changes, the variable is newly created. In these cases, the current value is discarded. A corresponding warning is issued first.</p> <p>When a device or system-variable is selected, the dialog opens. Here you can view the variable's properties, but not make any changes.</p> <p>(More info is presented in the section "User-defined Variables"^[749])</p>
 Show individually	<p>All selected channel variables (e.g. analog inputs) are opened in separate "free-floating" curve windows^[1102].</p> <p>All other selected variables (e.g. Display variables) are added to the window "Current values"^[816].</p>
 Show together	<p>All selected channel variables (e.g. analog inputs) are opened in a shared "free-floating" curve window^[1102].</p> <p>All other selected variables (e.g. Display variables) are added to the window "Current values"^[816].</p>
 Find	Displays the Search box. See the matching description for the toolbar ^[811] .
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 ").
 Unload measurement(s)	Unloads loaded, saved measurement(s) (see " Operation - Loading/unloading measurements ").
 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 ").
 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	Loads the appropriate experiment settings for the measurement.
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 ").
Copy name	Copies the name of the measurement to the Clipboard.

13.1.2.3 Current values - Inspecting/changing variables

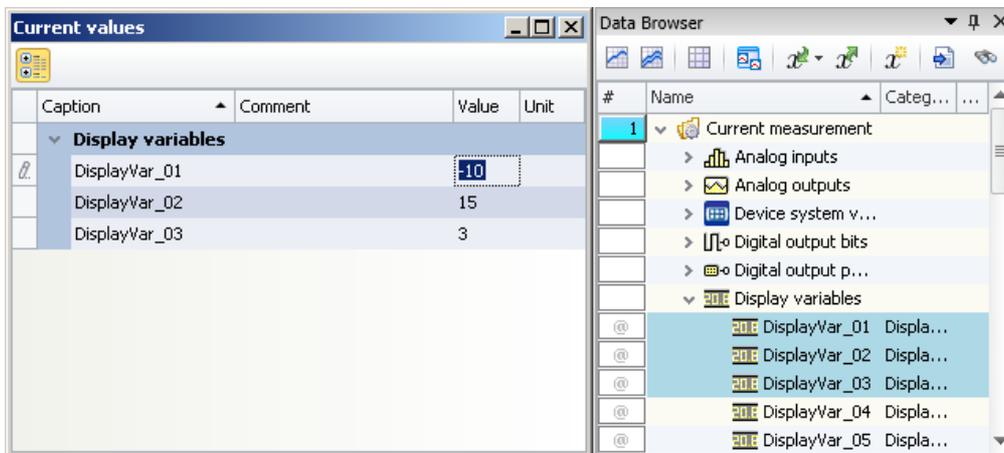
Note

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](#)"-Button ( / ) in the toolbar of the tool window Data Browser.

Subsequently, a dialog opens, as shown below:



When a **variable** is **deleted** from the Data Browser, it is automatically deleted from the Current Values-dialog.

*Inspecting or interactively altering variables
Example: three Display variables*

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

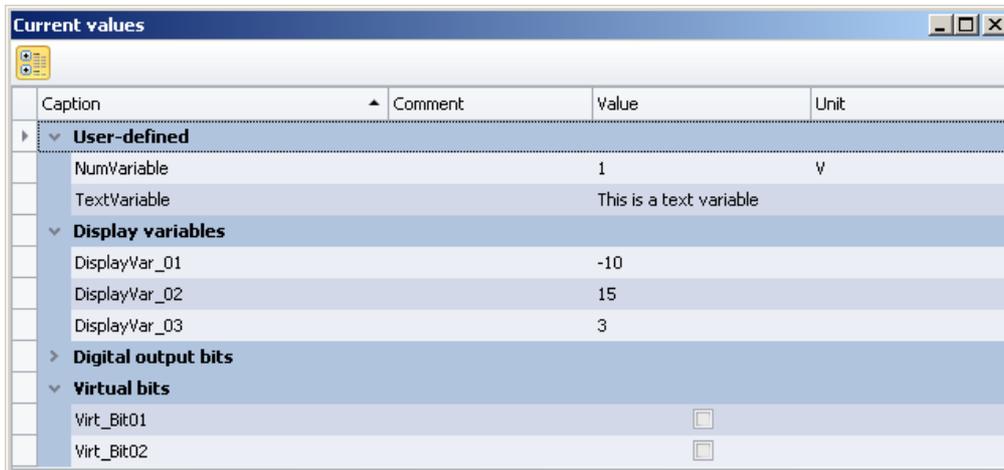
Structure and configuration

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.



Dialog: Current Values
Sample variables

13.2 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 ⁽¹¹²⁰⁾)
 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:	

Menu item	Description
 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: <ul style="list-style-type: none"> • Portable Network Graphics (*.png) • Windows Bitmap (*.bmp) • JPEG (*.jpg)
 As PDF	Saves the current Panel page in PDF format.
 Show Page on Monitor	<ul style="list-style-type: none"> • 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.
 Zoom Zoom (All pages)	Zooms the Panel page for the purpose of better display of, for example, Report-pages (see Zooming Panel pages ^[1128]).
Page Options:	
Page Layout for Printing	<p>Opens the dialog: "<i>Page Setup</i>"</p> <p>Here, you can change the setup and printout settings for Report-pages (including paper size, source, orientation, margin).</p>
 Lock Page	Lock/Unlock the current page. Individual pages can be protected against changes (see Lock and unlock page ^[1127]).
 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	<ul style="list-style-type: none"> • 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	<p>Skin specially adapted to printout.</p> <p>Widget-colors and the background colors of the current page align to the selected skin.</p>

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 ^[1127]).
Adapt Page Size	See Adapt/automatically adapting the size of pages and Widgets ^[1125] .
 Properties	Opens the Properties window of the Panel page ^[1122] .

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 ^[826])
 To Background	Moves the selected Widget all the way to the background, beneath all other Widgets. (see: Widget-levels - Foreground/Background ^[826])
Align	Aligns all selected Widgets to a line. (see: Align and Cascade ^[827])
 Cascade	Widgets are in stacked at the upper left corner of the Panel page according to their respective Widget-levels. (see: Align and Cascade ^[828])
Group	Groups Widgets together (see: Grouping Widgets ^[830])
 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	<ul style="list-style-type: none"> 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 ^[848])
 Properties	Opens the Properties window of the Widget ^[830] .

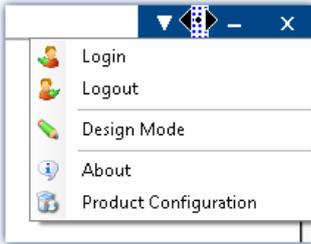
13.3 Design mode

In imc WAVE, you can edit the pages you personally made. The default pages provided are protected against changes. How to add pages is described in the chapter: "[Insert - Dialog / Report](#)"

To edit Widgets on a Panel page, the Panel must be in "Design Mode".

Activating Design-mode

Open the context menu over the bar on the top. In this menu, select the item "Design Mode".



Next, you can modify the page to give it your desired design. [New Widgets](#) are added to the page by means of the context menu.

13.4 Widgets - operation and properties

Widgets are placed ([inserted](#)) on the Panel page.

They offer a variety of [context menus](#) and [properties](#).

In general, there are multiple ways to operate the Widgets:

- [Move and change size](#)
- [Select](#)
- [Editing of Widget-elements](#)
- [Move to background/foreground \(Widget-levels\)](#)
- [Align and Cascade](#)
- [Docking](#)
- [Grouping](#)

Some Widgets have special functions, such as:

- link to variables (see the chapter: [Variable linkage](#))
- Extra variables (multiple scales/pointers and calculations of variables, e.g. Difference)
- [Zones](#)
- [Linking commands with switches](#)
- Use of [placeholders](#)

The structure

Each Widget consists of different [elements](#):

- Boxes displaying the **title** or the current **value**
- **Scales**
- **Numbers**

- **Meters, bar meters, ...**

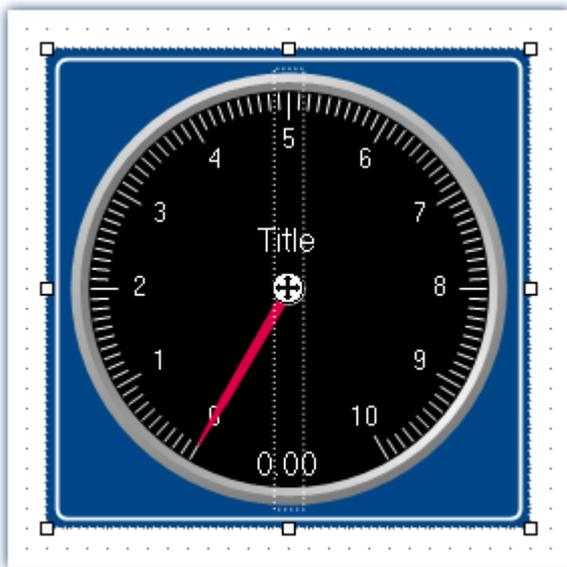
Some of these element have their own [context menu](#)^[822] and can be [edited](#)^[824]. Among other things, they can be moved, or their size and color can be changed. Some of these changes can also be made through the use of the Widget's [properties](#)^[830].

13.4.1 Insert Widget

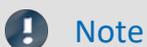
There are various ways to create a Widget on the page:

- Open the [context menu](#)^[818] 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](#)^[804] to Panel page and select the desired Widget (see: [Variable linkage by Drag&Drop](#)^[1134])

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](#)^[1127]).

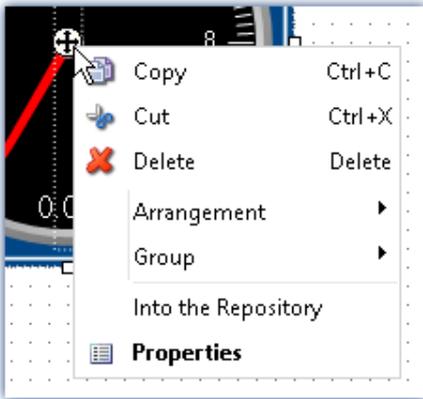
Commonly used Widgets

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.

13.4.2 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.

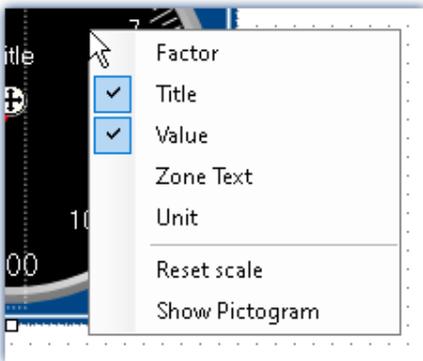
By this means, you can also open the [Properties](#) ⁸³⁰ without using the tool window.

Other context menus

Some Widgets also have additional context menus:

- to show or hide [Widget-elements](#) ⁸²⁴, such as boxes, and for resetting some properties
- to edit selected [Widget-elements](#) ⁸²⁴

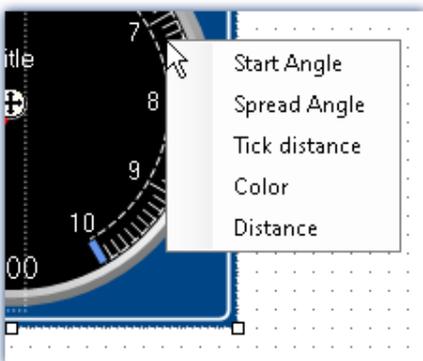
Here, using the example of a *standard meter*.



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](#) ⁸²⁴, for example.



Context menu for the Widget element

To open this context menu, select the [Widget-elements](#) ⁸²⁴ and right-click over the background or the control.

A context menu appears, using which the [Widget-elements](#) ⁸²⁴ can be edited.

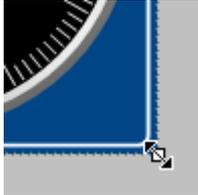
13.4.3 Operation

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](#)^[1127]).

Selection

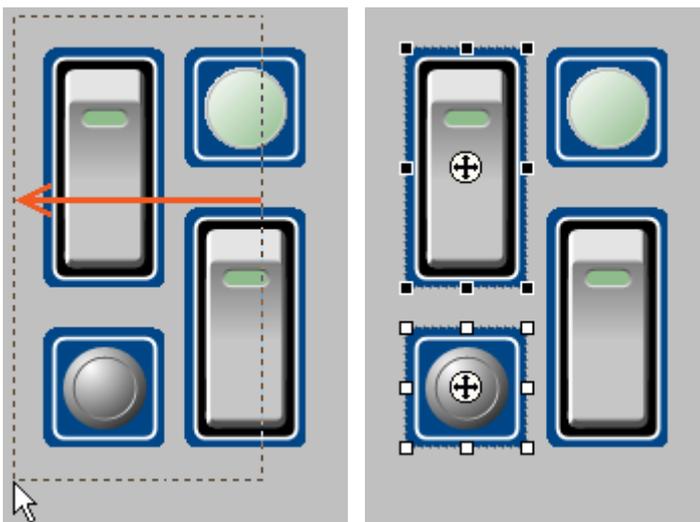
On a single page, you can select either one or multiple Widgets. The operations below can be performed as desired in any order.

- 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:

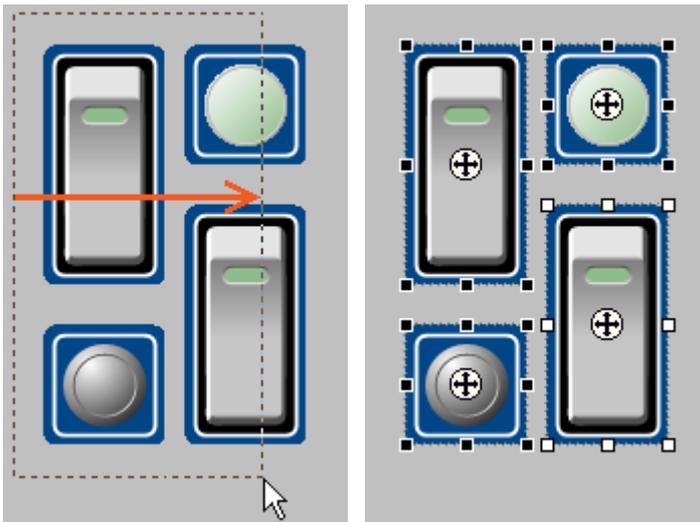
To select multiple Widgets, use the mouse to drag a rectangular frame around the desired Widgets. The starting point for dragging must be in an empty area of the page:



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.



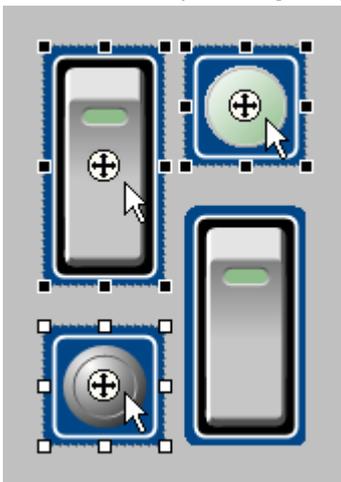
Dragging the frame from left to right

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](#)⁸³⁰. 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:

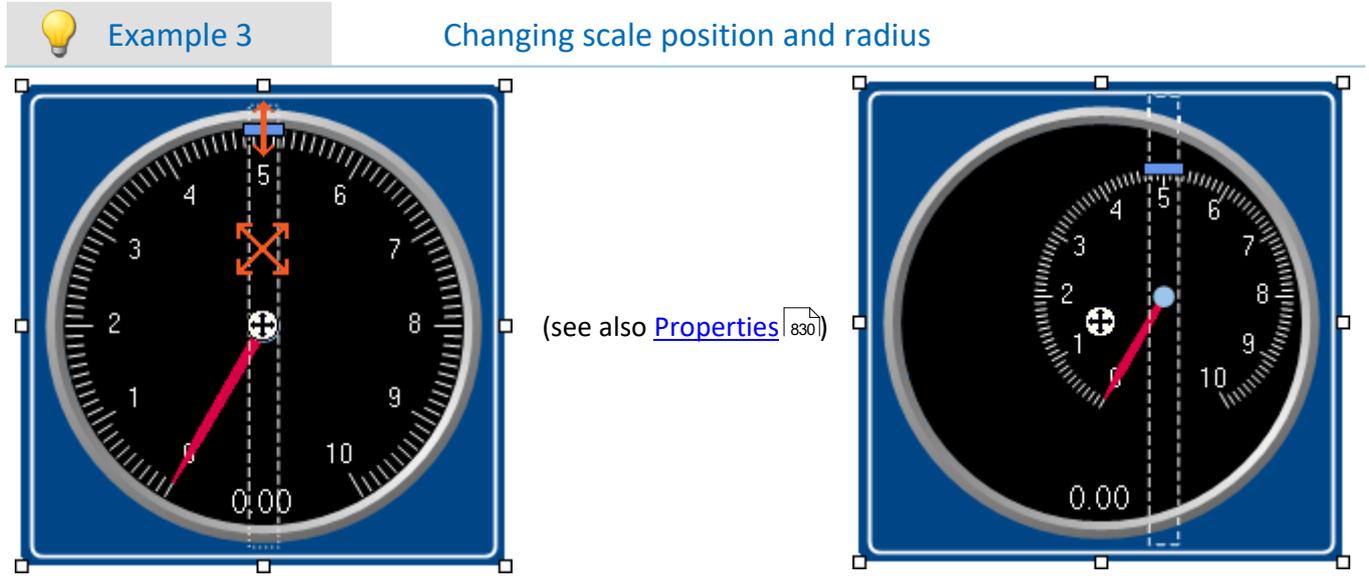
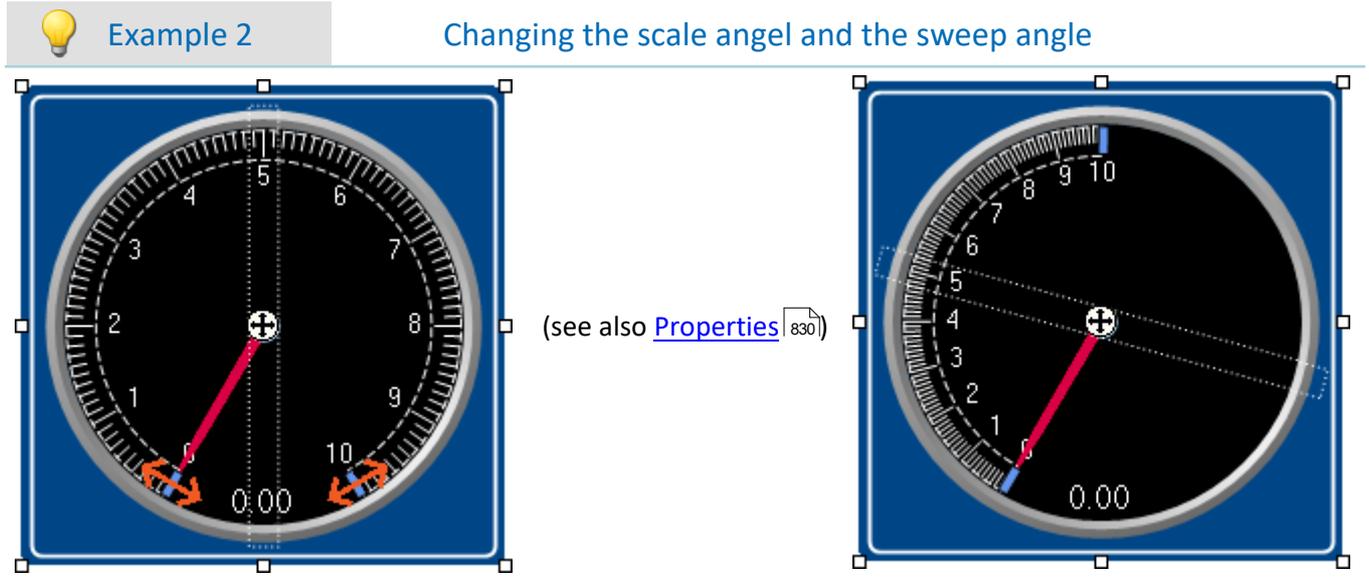


Example 1

Move and change the size of boxes



Move and change the size of boxes



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](#) ⁸²⁹.

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.

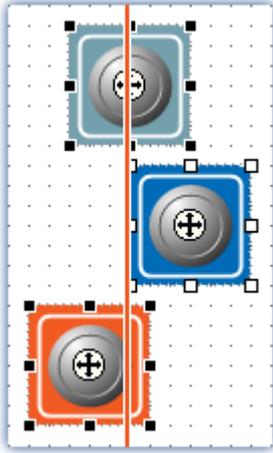
To do this, open the context menu of the "crosshairs" (⊕) within the selected Widget.

- Select *Arrangement*:

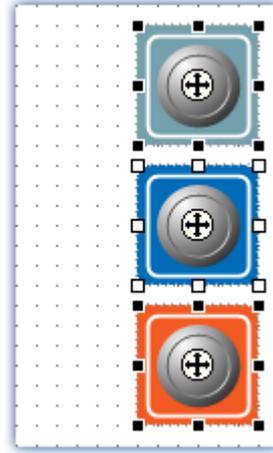
Action	Description
☰ Align top	Alignment to the upper edge of the last Widget selected
☱ Align left	Alignment to the left edge of the last Widget selected
☲ Align Right	Alignment to the right edge of the last Widget selected
☵ Align Bottom	Alignment to the bottom edge of the last Widget selected

 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.

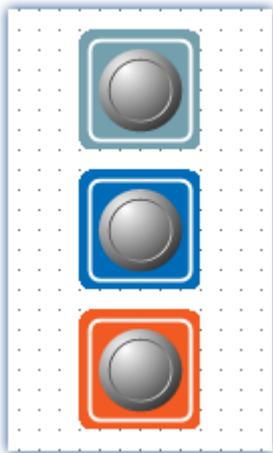
To do this, open the context menu of the "crosshairs" (⊕) within the selected Widget.

- Select *Arrangement*:

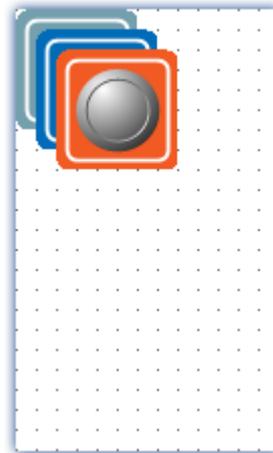
Action	Description
 Cascade	Selected Widgets are stacked in the upper left corner of the Panel page according to their respective Widget-levels.

 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":

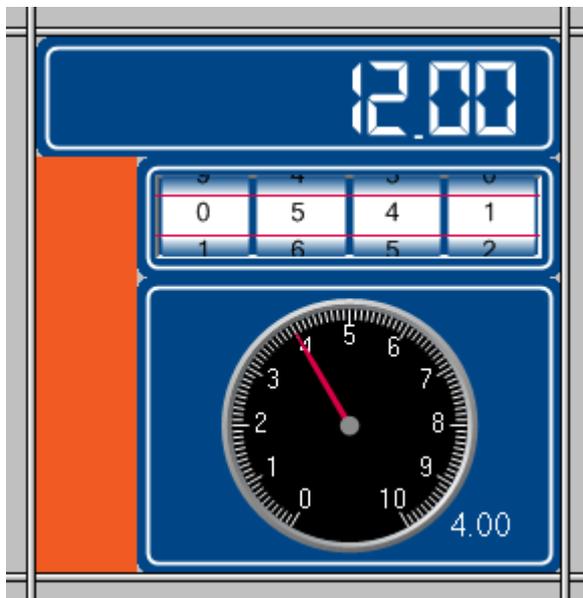
Action	Description
 Free	The Widget is not permanently docked to any edge, but can be freely moved. The size can be varied toward any side.
 Top	The Widget is permanently docked to the upper edge. The size can only be varied toward downward.
 Left	The Widget is permanently docked to the left edge. The 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.

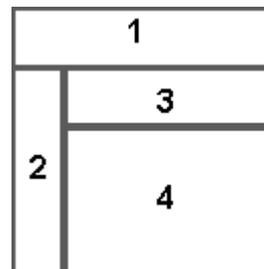


Example

Example for docking in a cell of the split Panel page



Widget-levels:



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*:

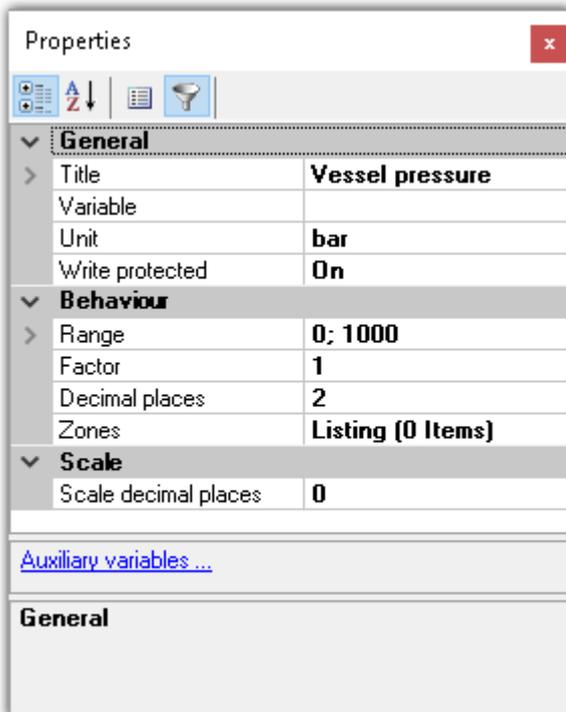
Action	Description
 Create Group	Selected Widgets are joined to a group.
 Dissolve Group	Dissolves the grouping. The Widgets can then each be edited separately.
 Enter Group	Opens the existing group. The properties of the individual Widgets can be edited separately.
 Leave Group	Closes an open group.

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.

13.4.4 Properties - Widget

If you have selected a Widget, its properties can be adjusted in the [tool window](#) ⁸⁰¹ **Properties** (via the context menu of the selected Widget, see "[context menus of the Widgets](#)" ⁸²²).

The content of the Properties window depends on the respective Widget selected. For info on the window structure, see "[Properties \(general\)](#)" ⁸⁰².



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	<p>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.</p> <p>The times are defined as follows:</p> <table border="1"> <tr> <td>• very fast</td> <td>50ms</td> <td>20Hz</td> </tr> <tr> <td>• fast</td> <td>200ms</td> <td>5Hz</td> </tr> <tr> <td>• standard</td> <td>1s</td> <td>1Hz</td> </tr> <tr> <td>• slow</td> <td>10s</td> <td>0.1Hz</td> </tr> <tr> <td>• never</td> <td>--</td> <td>--</td> </tr> </table> <p>Reading and outputting of the new values may be delayed in proportion to the utilization of the PC's resources.</p> <p>With the option "<i>Refresh rate of newly created Widgets</i>", you specify the default refresh rate of newly created Widgets.</p>	• very fast	50ms	20Hz	• fast	200ms	5Hz	• standard	1s	1Hz	• slow	10s	0.1Hz	• never	--	--
• very fast	50ms	20Hz														
• fast	200ms	5Hz														
• standard	1s	1Hz														
• slow	10s	0.1Hz														
• never	--	--														
Status	<p>Switches the Widget active/passive.</p> <ul style="list-style-type: none"> • 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	<p>Source for the title</p> <ul style="list-style-type: none"> • 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 ^{r1138} .															
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. <ul style="list-style-type: none"> • Show: Displayed in the export or printout • Hide: Omitted from the export or printout
Write protected	Protects against changed values. <ul style="list-style-type: none"> • 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) <ul style="list-style-type: none"> • 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). <ul style="list-style-type: none"> • 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 ⁸³⁷ .
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
----------	-------------

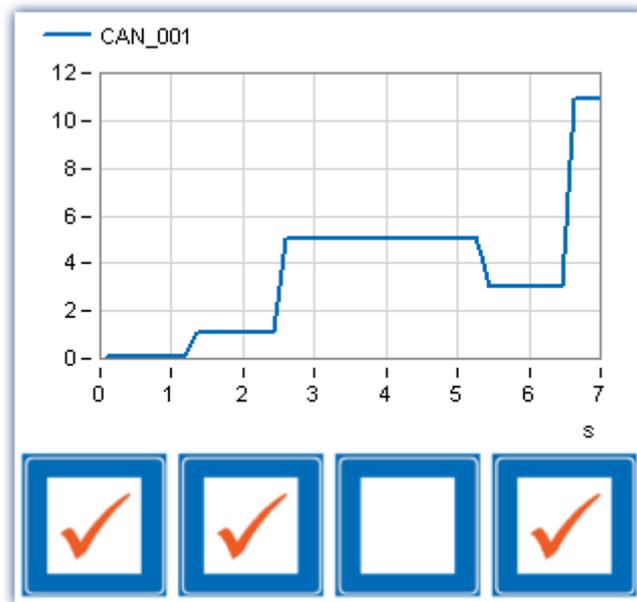
Displaying variables' individual bits

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.

Bitmask



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.

Number of decimal places for display of the value. (See also "Scale decimal places")

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

Graphical display of the zones in text boxes. (Input, output > Numeric ...)

Edit box color

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](#) ⁸⁴⁸ which are carried out at state transitions (called events).

Property	Description
	<p>Factor with which the value is to appear. The available selection options depend on the Widget and on the unit.</p> <p>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>", ...</p> <p>Additionally, the following selection options are available:</p> <ul style="list-style-type: none"> • 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. <hr/> <p>Example: The variable's value is "<i>0.1</i>" and the unit "<i>V</i>".</p> <p> If you select "<i>milli</i>" as the factor, then "<i>100 mV</i>" is displayed.</p> <p>If you select "<i>1</i>" as the factor, no conversion calculation is performed; the displayed output is "<i>0.1 V</i>".</p> <hr/>
Factor	<p>Unit conversion calculations</p> <p>The following procedure is applied by the Widgets "<i>Table</i>" and "<i>Numeric input</i>" belonging to the groups: Automotive, Industrial and Designer.</p> <ol style="list-style-type: none"> 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. <p>The same applies to many other units.</p> <hr/> <p> 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).</p> <hr/>
Increment	Increment size for Widgets with buttons for up and down.

Property	Description
Range	<p>The display range, or the range of valid entries; e.g., for a standard meter, the min- and max-values for the scale.</p> <p>From Variable</p> <p>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"^[270].</p> <p>All other variables automatically use the values defined, even if "<i>From Variable</i>" is selected.</p> <p>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".</p> <p>Defined values: Maximum and Minimum</p> <p>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.</p> <p>Infinite</p> <p>If the range is to be indefinite, you can enter "<i>Infinite</i>" or "<i>-Infinite</i>" (not possible with scale-Widgets).</p>
Maximum	Maximum value of range
Minimum	Minimum value of range
Scale center	<p>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.</p> <p>As the midpoint, any value can be specified or the fixed minimum or maximum value.</p>
Switching behavior	<p>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.</p> <ul style="list-style-type: none"> • 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	<p>Here you can format the value display. Some Widgets also support placeholders^[1217].</p> <p> In cases of parallel use with zone-texts, see the notes regarding: "Zone dialog"^[847]</p>
Zones	<p>Displays existing zones. See "Zone dialog"^[837]</p> <p>See also the display settings: "Zone representation" and "Edit box color"</p>
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	<p>Smallest step between two values. The values in between are not displayed.</p> <p>E.g. standard meter with an "interval" of 0.5: The pointer always jumps at interval steps: 0; 0.5; 1; 1.5; ...</p>

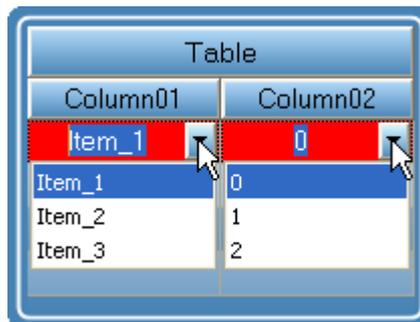
Property	Description
Scale angle	Opening and starting angle for the scale display. <ul style="list-style-type: none"> • Start angle: 0° -> Horizontal rightwards • Spread angle: clockwise
Scale decimal places	Number of decimal places for the scale display. The count of decimal places is automatically adjusted when the range is specified by the variable.
Scale position	<ul style="list-style-type: none"> • 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.

13.4.4.1 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

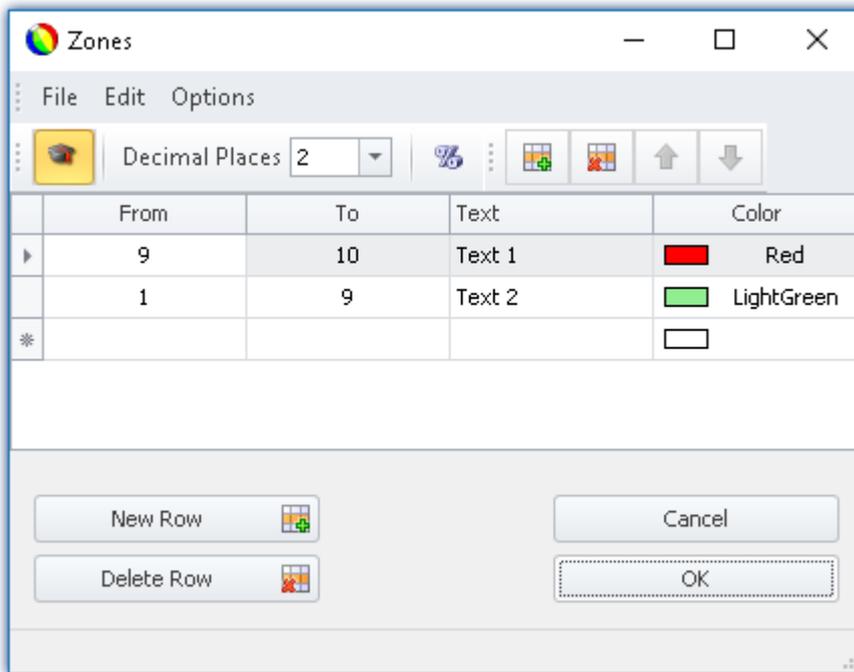


Example of a table with list boxes

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](#)⁸⁴⁰ depend on:

- the Widget (control elements require different settings than display elements)
- the Widget property "[Write protected](#)"⁸³⁰ (preset values are not required)
- the option "[Advanced display](#)"⁸⁴⁴ (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](#) ⁸⁴⁴ 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).

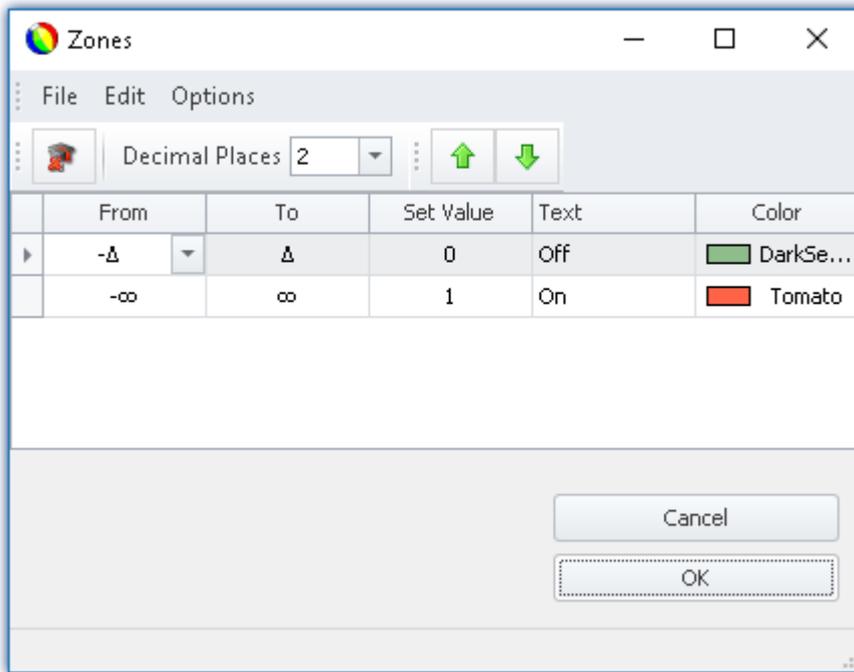
Moving the zones

The top zone has top [priority](#) ⁸⁴².

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](#)" ⁸⁴⁴.



Zones of a switch with "Advanced display"

The switch should be "Off" when near "Zero" and for all other values "On".

13.4.4.1.1 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](#)⁸⁴⁴" 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](#)⁸⁴⁴", **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](#)⁸⁴⁴" 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 in the column **Set Value** and have not activated the **Advanced display**, the other values adjust themselves.

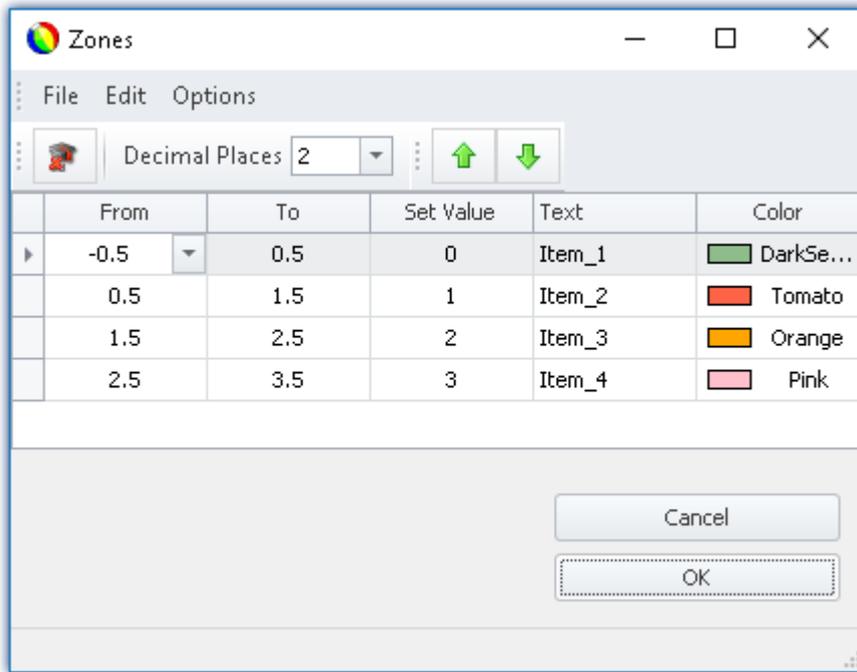
Set Value	Text	From (not displayed)	To (not displayed)
0 > 1	Entry_1	-0.5 > -∞	0.5 > 1.5
1 > 2	Entry_2	0.5 > 1.5	1.5 > 2.5
2 > 3	Entry_3	1.5 > 2.5	2.5 > 3.5
3 > 4	Entry_4	2.5 > 3.5	3.5 > ∞

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	∞

13.4.4.1.2 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").

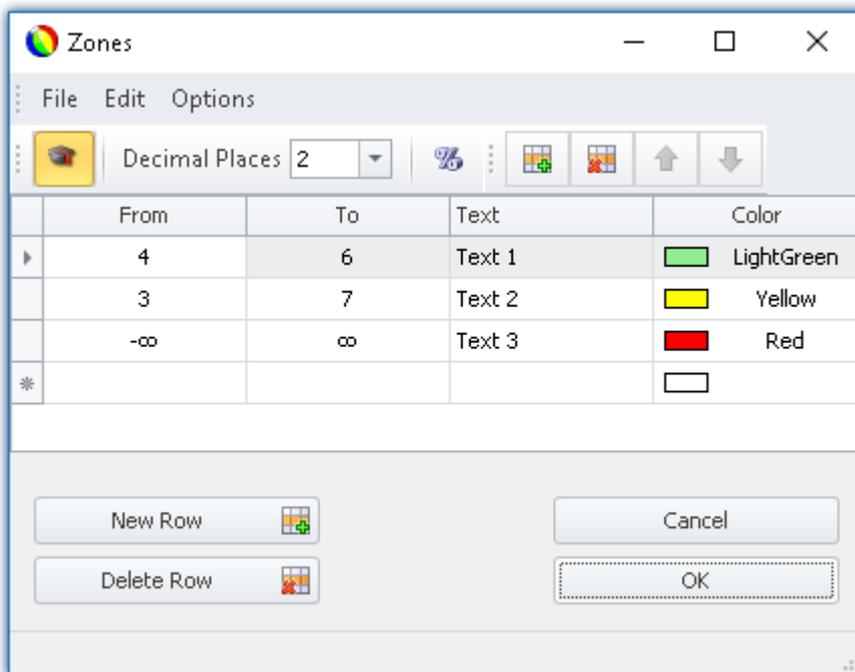


This amounts to five zones:

- 1: 4 to 6
- 2: 3 to 4
- 3: 6 to 7
- 4: -infinity to 3
- 5: 7 to infinity

One can make these into three zones:

- 1: 4 to 5 ("green")
- 2: 3 to 7 ("yellow")
- 2: -infinity to infinity ("red")



13.4.4.1.3 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)

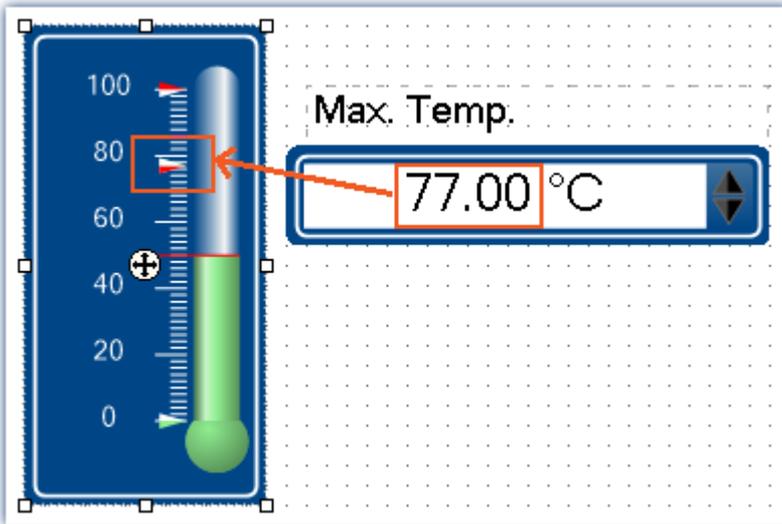
Activating Advanced Display

In order to activate "*Advanced display*", use the entry: "Extended display"  (e.g. via the menu entry).

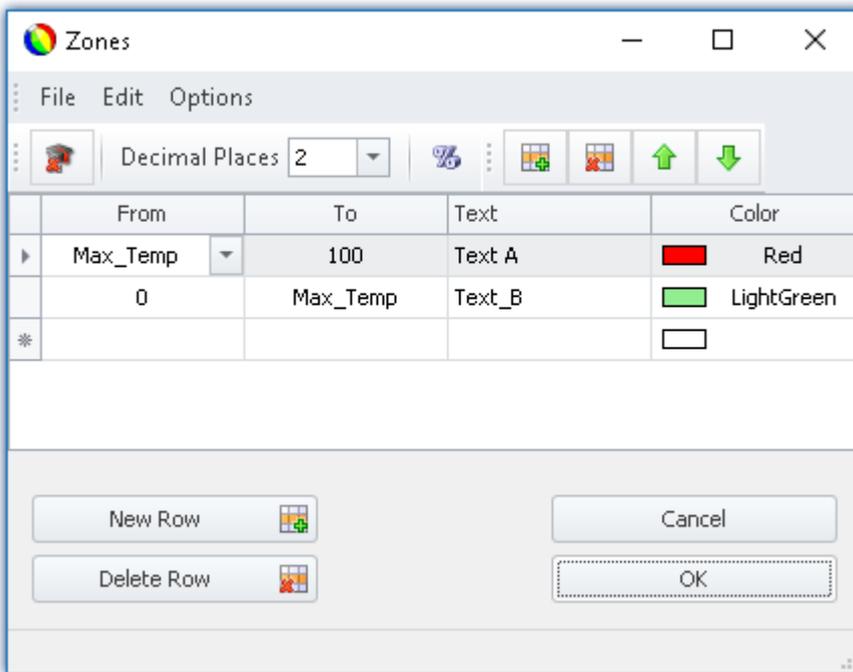
 **Example**

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%)



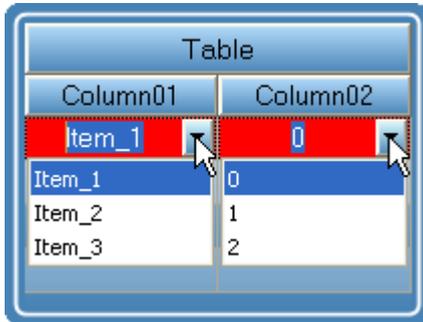
The range changes to: 0-50

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.

13.4.4.2 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**:

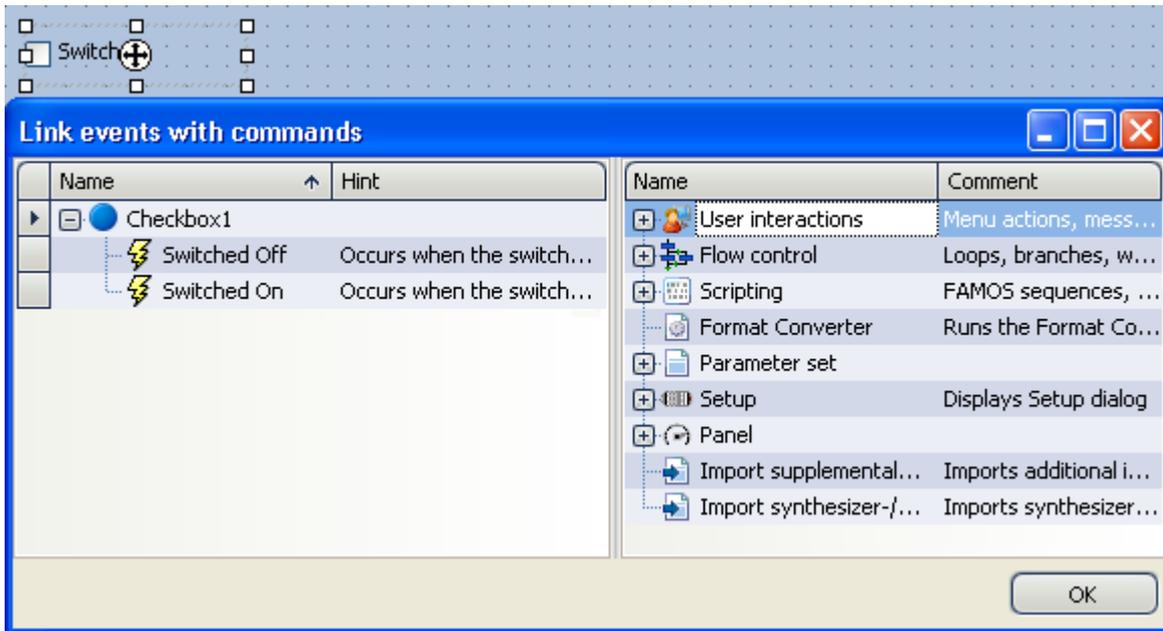
- ⚡ **Switched off** (from On to Off)
- ⚡ **Switched on** (from Off to On)

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**.
- A **sequence of commands** can be created in various imc STUDIO plug-ins and is described separately (see the chapter "[Sequencer, events and commands](#)^[1140]").



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

13.5 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 .

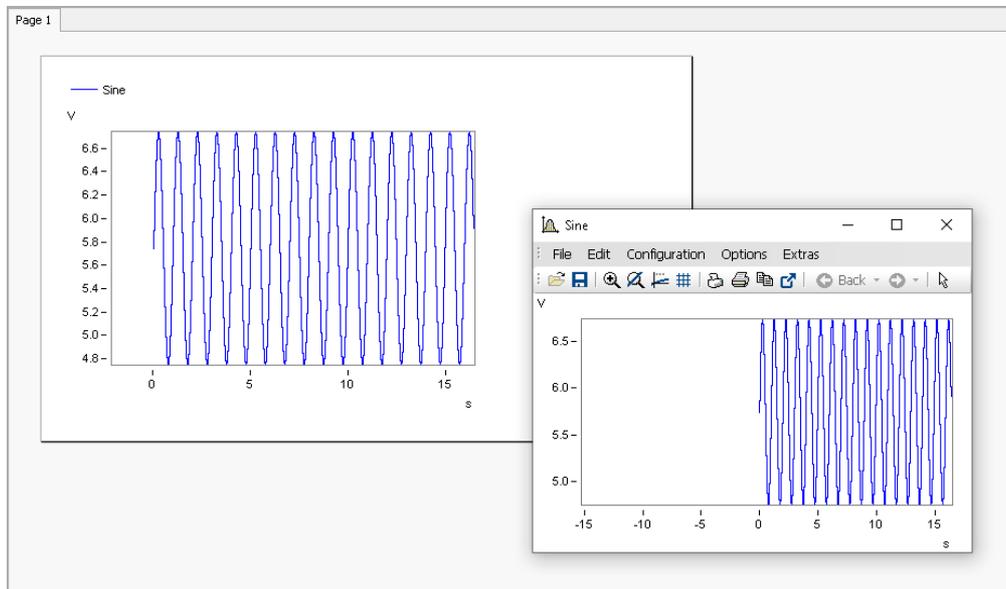
Function	Description
symbols	Curves can be marked with symbols to help differentiate between curves in a monochrome display.
periodic waveforms	An option is available for periodic waveforms in which all periods are compared with each other. This helps to identify trends and aberrations.
Color maps	A "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..

13.5.1 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 ⁸⁵²	Display variants of the curve window
Legends ⁹⁶⁷	All curve windows can include a legend, with the channel (variable) name, line color etc.
Map ⁹⁷⁰	Map lets you place a picture in the background. It is additionally possible to layout the picture in specified coordinates.
Color palette ⁹⁸³	A channel's color can be governed by a reference signal's amplitude.

Display

Display Legend Map Color palette Text

Display type

Default Table Polar plot

Stack y-axes Bar meter

Waterfall diagram Color map

Last value as number 3D

Scale of x-axis

Chosen x-unit Days/hours/minutes: relative

Date/time: absolute Third/octave labeling

Reference for dB-display: Labeling

Coordinate system size: Show x = 0 (trigger)

Number trimming (useful when scrolling)

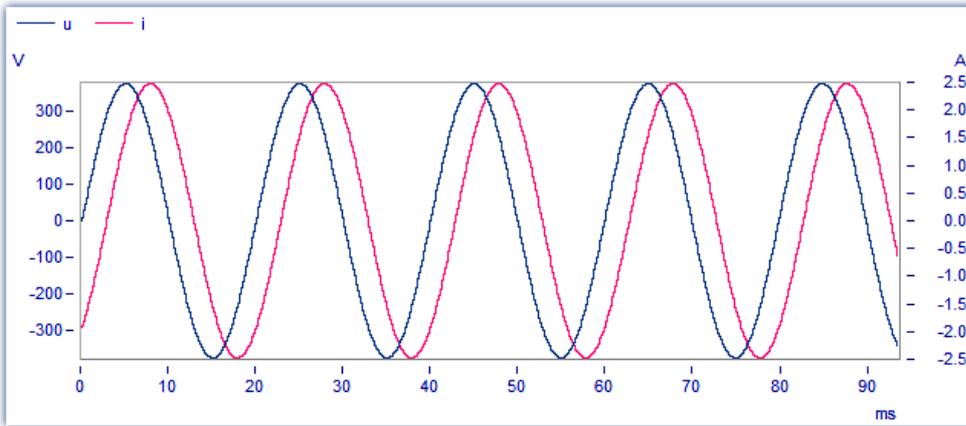
Grid

OK Cancel Apply changes Set as default Topic

Display type

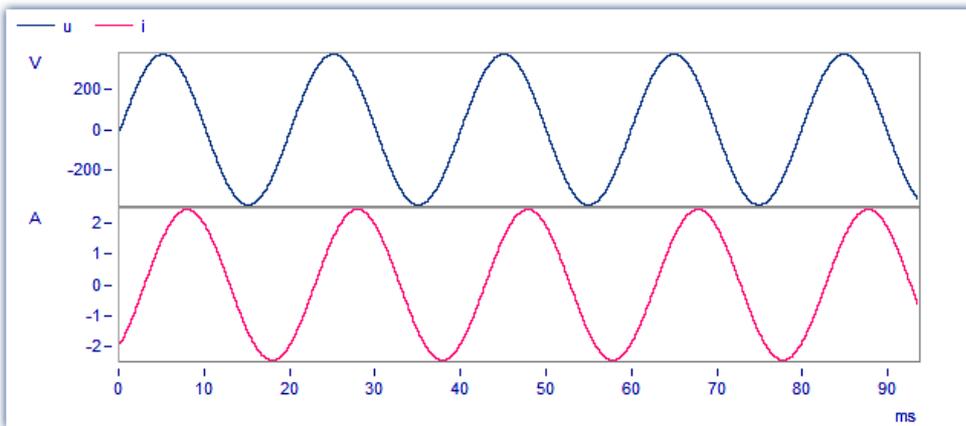
Default

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.



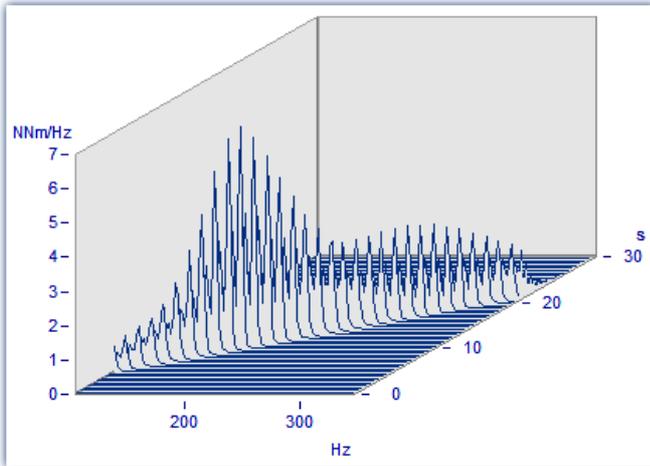
Y-axes stacked

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

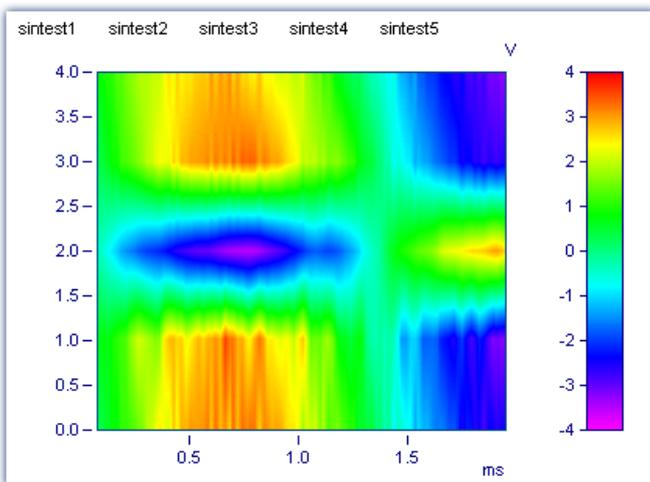
Another alternative is the waterfall diagram, which is especially suited to comparing similar curves. The curves are drawn in a diagonal procession from the front of the screen, with the curves at the back (partly) obscured by those at the front, creating the illusion of a three-dimensional perspective.



See "[Waterfall Diagrams](#)"⁸⁶²".

Color Map

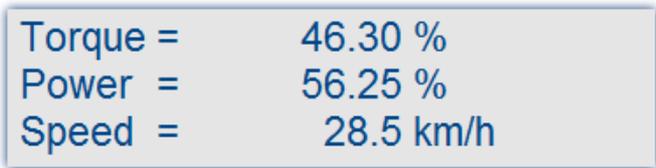
This is a bird's-eye-view of a colored, three-dimensional region. Multiple data sets are each plotted along the horizontal direction, stacked over each other along the y-direction, while their amplitude values are represented by different colors. This style of display is analogous to a geographical map where the landscape contours are indicated by shading or coloring.



See "[Color Map Display](#)"⁸⁸⁸".

Last value as number

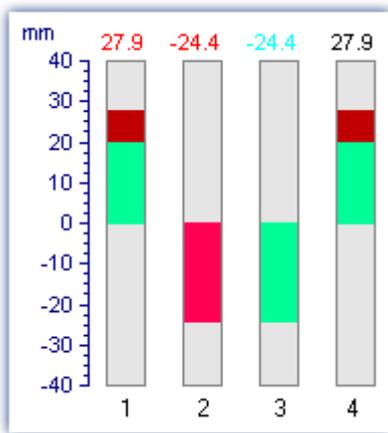
Particularly for online display of measured data, in this mode the last value is displayed as a number. The last value in a waveform is always the most recent measurement value in online display. This mode of display is particularly suited for slowly changing quantities. For example, the amplitude of 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](#)".

Bar meter

Particularly for online display of measured data, in this mode the last value is displayed in a bar meter.



Section "[Bar gauge](#)".

Table

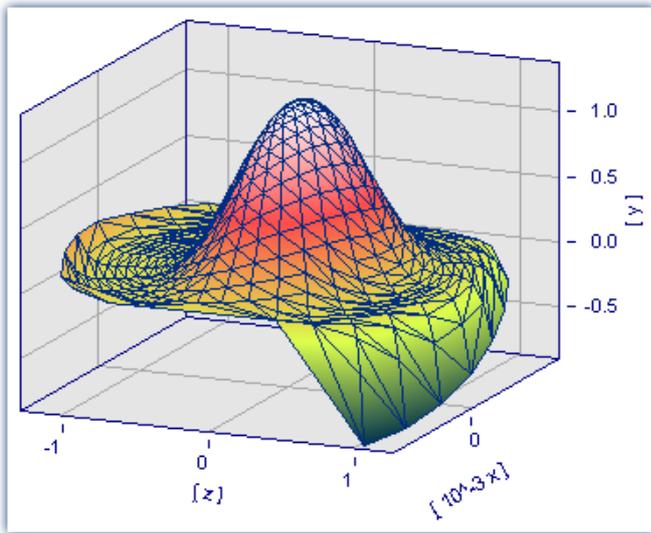
The readings are displayed in tabular form, in chronological order.

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](#)".

3D

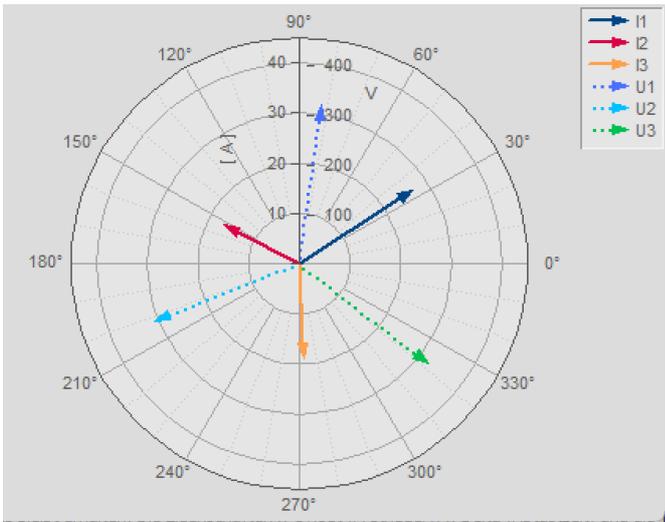
3-D display of waveforms having x,y,and z coordinates with various color map surface patterns, and rotating perspective in all dimensions.



Section on [3D display](#) ¹⁹⁰⁰

Polar plot

Complex data sets can be displayed as a polar diagram.



Complex single values as vector diagram



Section on ["Polar plot"](#) ¹⁹⁰⁰

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.

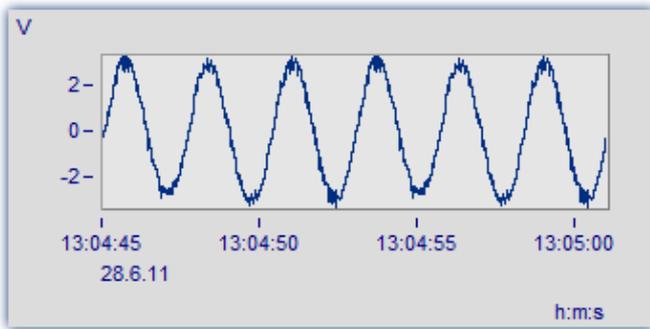
The generation time of a waveform along with the x-coordinate in "s" determines the resulting axis labels.

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

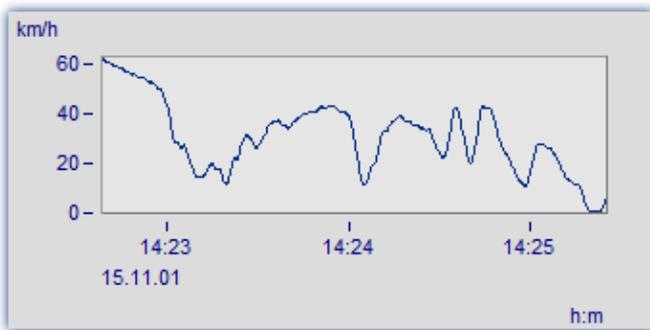
1.1.92, 13:00:00 to 2.1.92, 13:00:00

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.



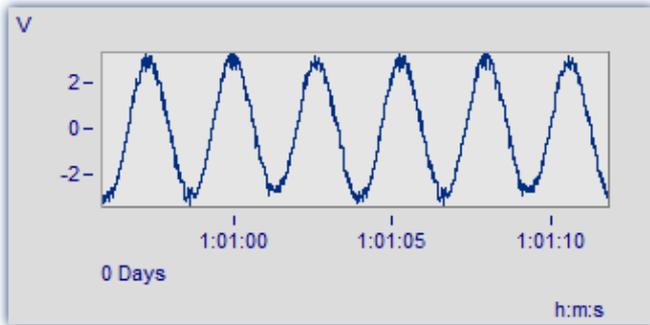
The segment between 14:23 and 14:25 comprises a few minutes.



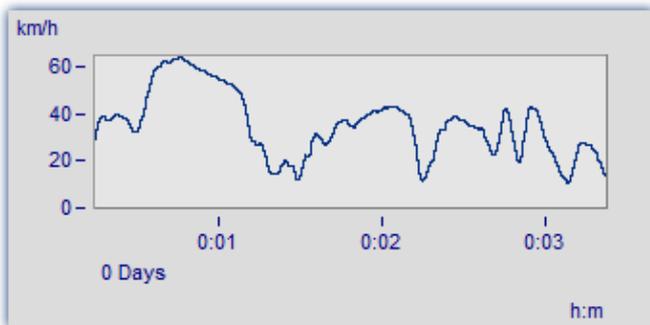
The segment between 4 am and 8 pm encompasses a few hours.

Days/hours/minutes: Relative

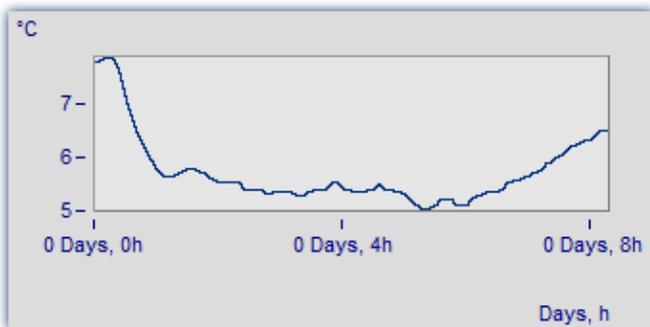
If you wish to view the whole duration of a long-term measurement, then display of relative time in days, hours, minutes and seconds is recommended. The selected unit depends on the interval. Possible variations of this labeling are as follows:



A very narrow section is shown, significantly less than one minute



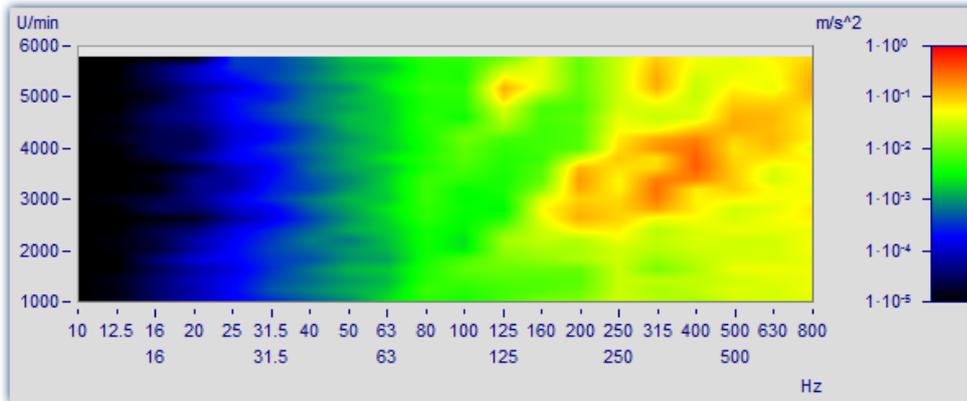
This section is 3 minutes long. The display no longer includes decimal places after the seconds.



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](#) 

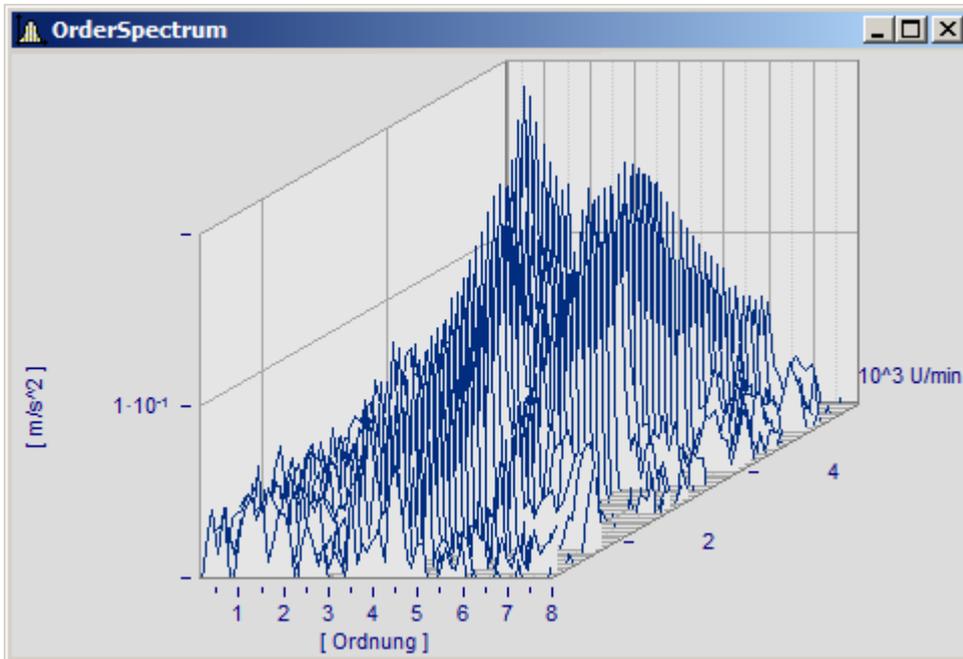
Remarks

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.

13.5.1.1 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.

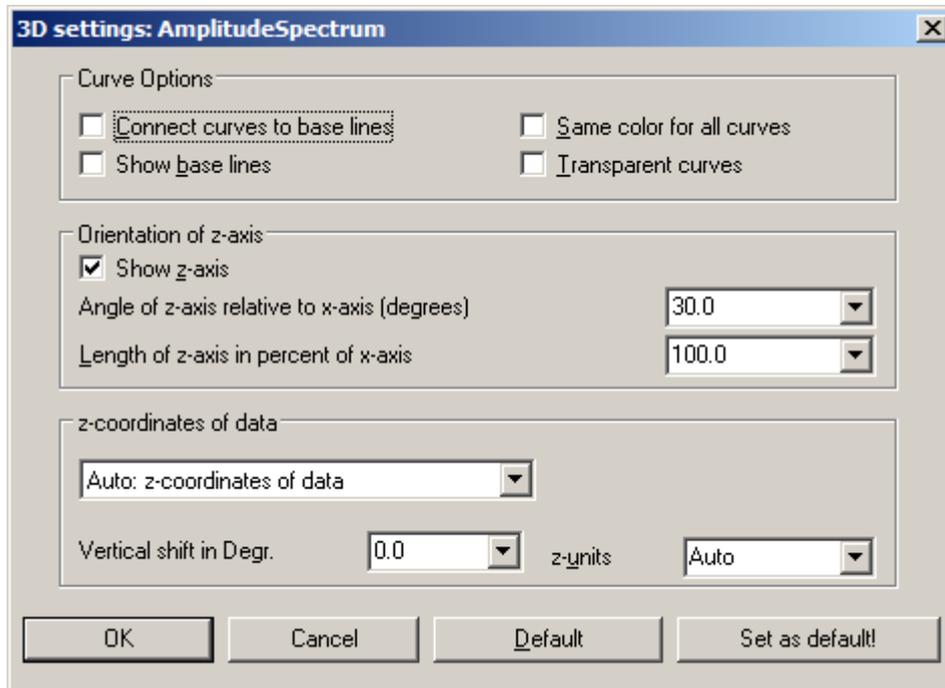
A waterfall display is drawn from back to front. When a new curve is drawn, the area under the curve is filled in with the color of the screen background, covering all lower parts of curves behind it.

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

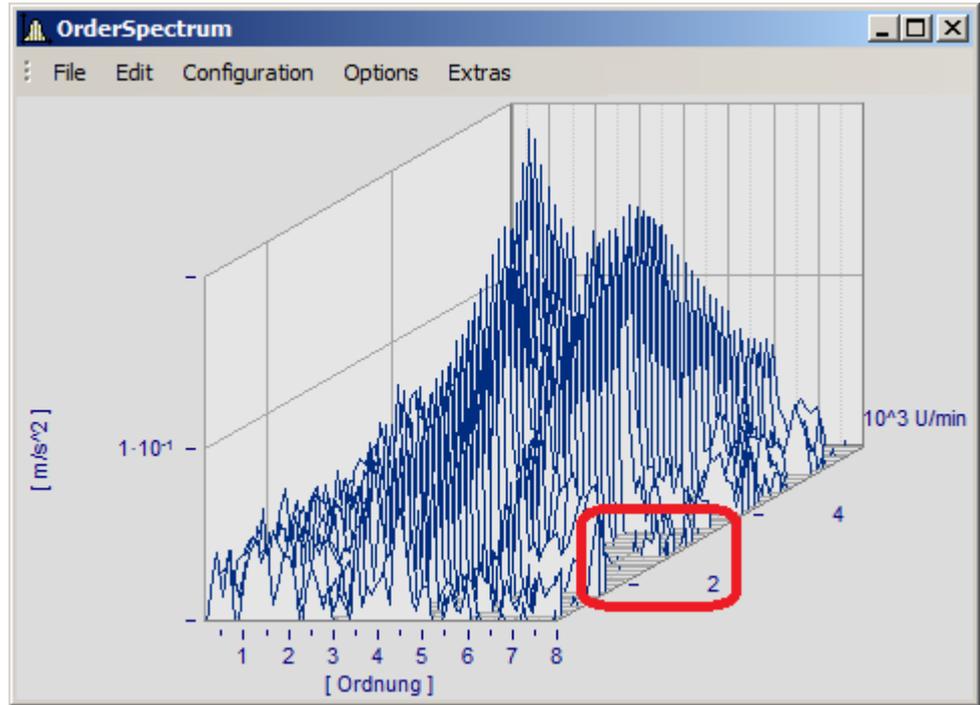
- Call the dialog for setting the type of display using the menu option *Configuration/ Display..* Select *Waterfall* and end the dialog with *OK*.
- The curves displayed in the window are shown in a waterfall diagram with the effective presettings.
- A dialog is available to change the settings of the waterfall diagram. Select menu option *Configuration/ 3D...* The following dialog appears:



Curve options

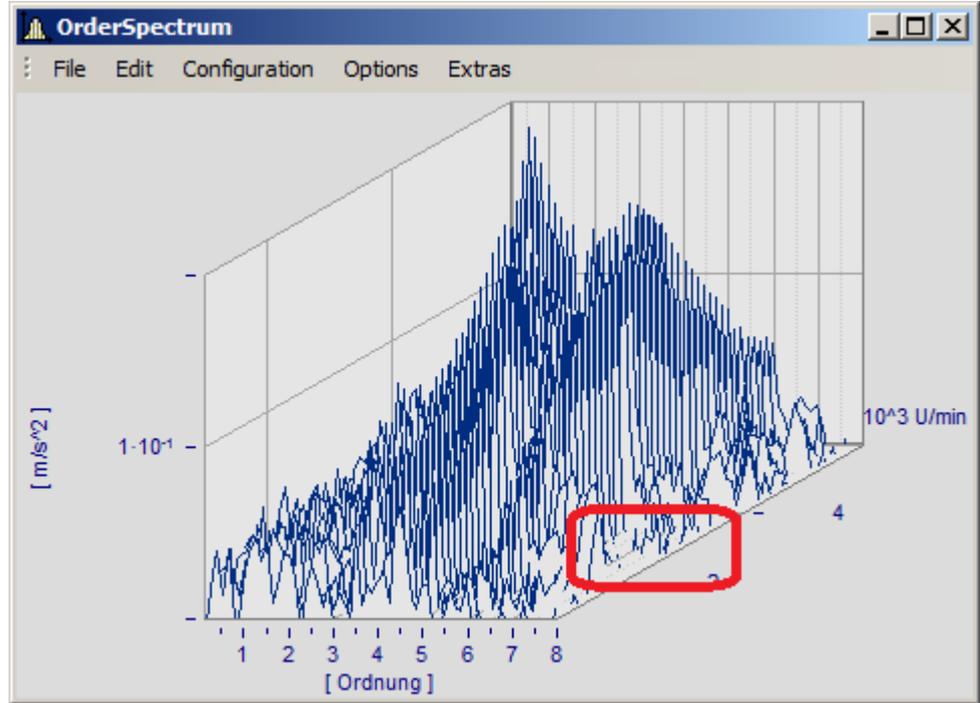
Enter here the settings for the appearance of the curves in the window.

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	<p>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"</p> <p>The waveforms thus displayed must have unique and increasing z-coordinates.</p> <p>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.</p>
Ditto with events: relative to first event	<p>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.</p> <p>The other, related options listed make reference to other shots (events). E.g., the last, the first displayed, or the last displayed.</p>
	<p> The z-coordinate of each waveform absolutely must be greater than the last. Otherwise the display will not include all waveforms!</p>
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](#)'  935

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.

13.5.1.2 Last value as number

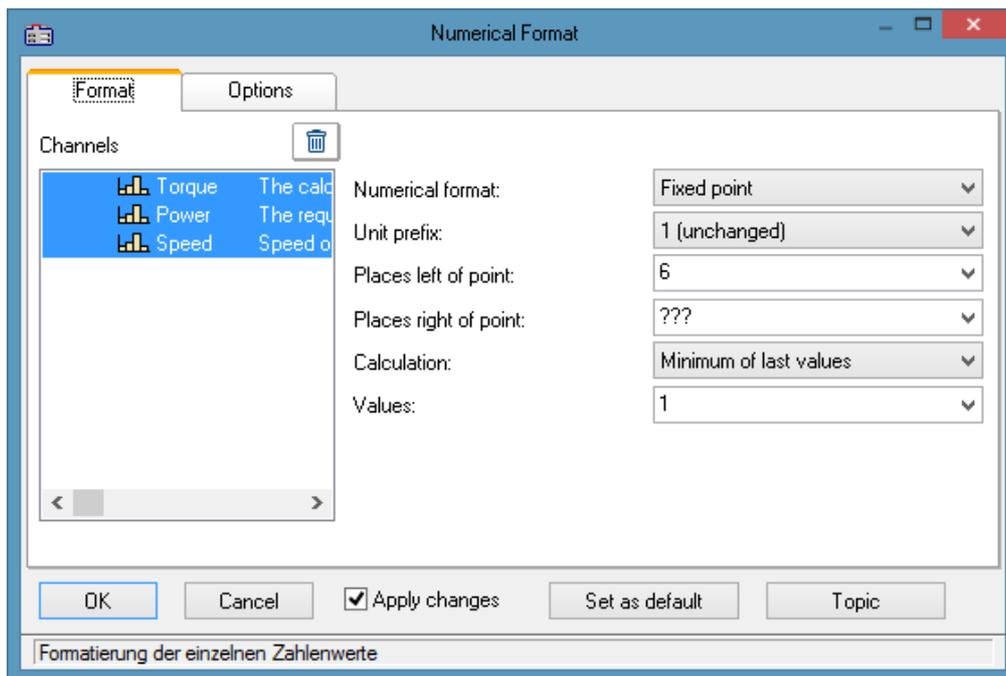
Torque =	46.30 %
Power =	56.25 %
Speed =	28.5 km/h

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)

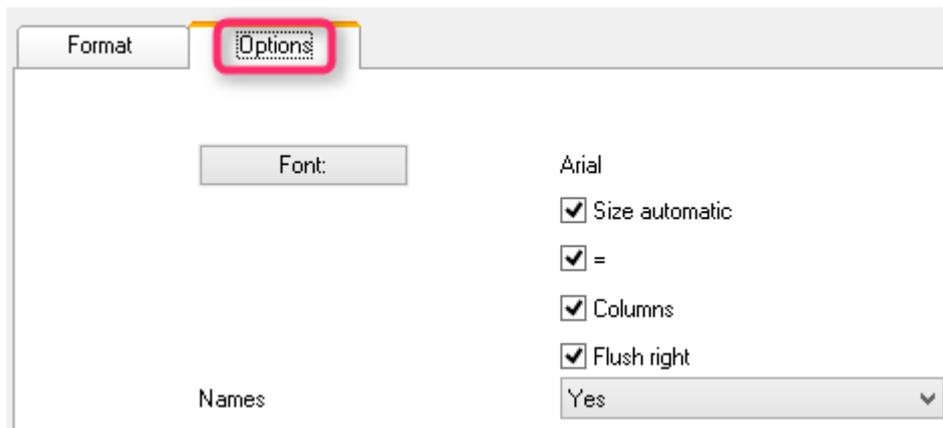
All settings made here only affect waveforms in the *Waveforms* list.

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 border="1" data-bbox="486 555 1433 1003"> <thead> <tr> <th>Numerical format</th> <th>Description</th> </tr> </thead> <tbody> <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-axis⁸⁵².</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 "Scale⁹³⁶".</td> </tr> </tbody> </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 ⁸⁵² .	Time relative-, Date/time fixed	The formatting of the time and date is specified by means of placeholders, see the chapter " Scale ⁹³⁶ ".
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 ⁸⁵² .										
Time relative-, Date/time fixed	The formatting of the time and date is specified by means of placeholders, see the chapter " Scale ⁹³⁶ ".										
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	<p>The numerical value displayed can be calculated from the most recent values:</p> <table border="1" data-bbox="486 318 1436 676"> <thead> <tr> <th data-bbox="491 324 774 369">Calculation</th> <th data-bbox="782 324 1431 369">Description</th> </tr> </thead> <tbody> <tr> <td data-bbox="491 380 774 481">Maximum of the last values</td> <td data-bbox="782 380 1431 481">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 data-bbox="491 492 774 571">Minimum of the last values</td> <td data-bbox="782 492 1431 571">The minimum is calculated from the last n values in the waveform.</td> </tr> <tr> <td data-bbox="491 582 774 672">Average of last values</td> <td data-bbox="782 582 1431 672">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> </tbody> </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	<p>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.</p> <p>When 1 is specified, the result of calculating the minimum, maximum and mean value is always the last value in the waveform.</p>								

Options (Curve Window Defaults)



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 style *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](#)¹⁰⁰⁸.

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*

13.5.1.2.1 Single value display

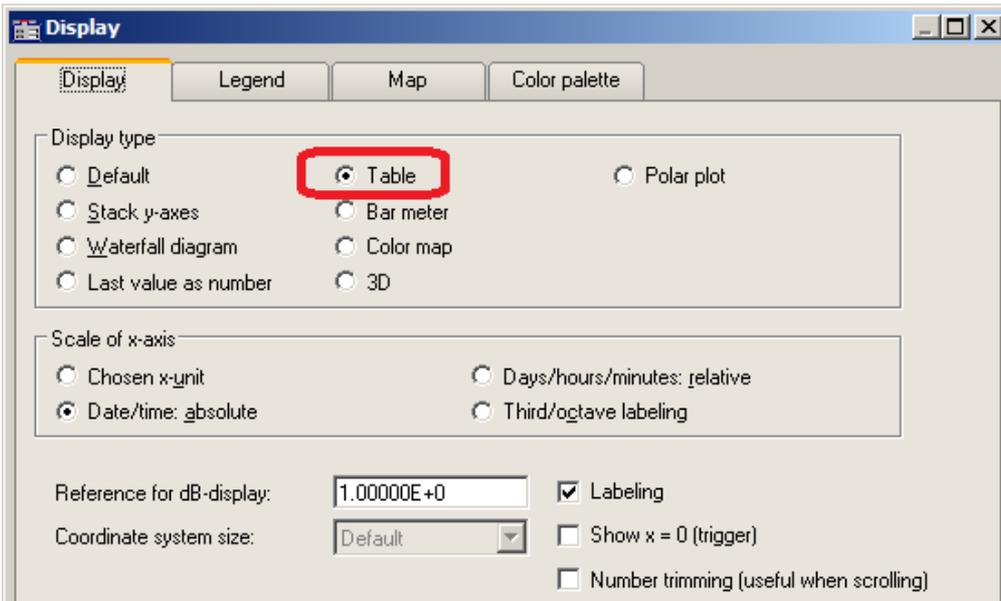
The display mode *Last value as number* is automatically activated if real waveforms with a length of 1 are displayed.

13.5.1.3 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.

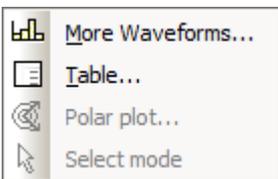
Operation

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:

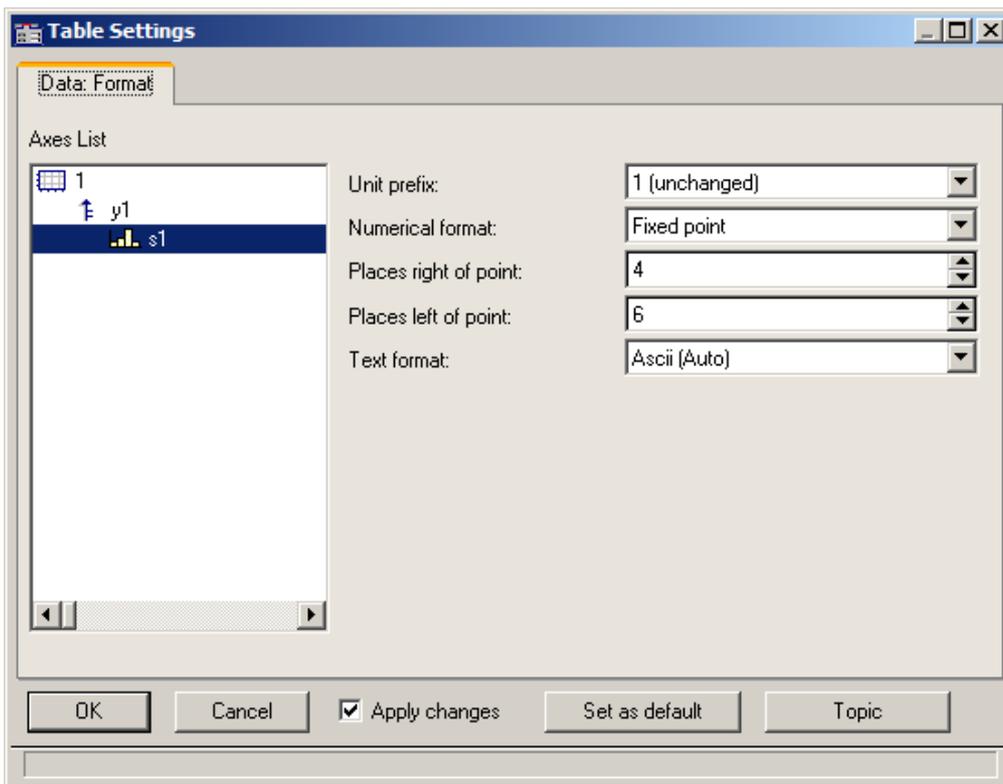
t ["order"]	OrderSpectrum [m/s ²]
0.125	0.0092
0.125	0.0157
0.125	0.0161
0.125	0.0101
0.125	0.0083
0.125	0.0150
0.125	0.0194
0.125	0.0107
0.125	0.0142



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.

13.5.1.3.1 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.

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.

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](#) ⁸⁷⁶ 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.

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)	<p>Display of hexadecimal values in 3-number sets.</p> <table border="1"> <thead> <tr> <th>t [s]</th> <th>Hex_3er_Messages_S1_K2</th> </tr> </thead> <tbody> <tr> <td>2.70843</td> <td>64 00 00 00 00 00 00 00 fe ff</td> </tr> <tr> <td>2.72640</td> <td>65 00 00 00 00 fe ff 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 fe ff</td> </tr> <tr> <td>2.82641</td> <td>65 00 00 00 00 fe ff 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 fe ff</td> </tr> <tr> <td>2.92641</td> <td>65 00 00 00 00 fe ff 00 00</td> </tr> <tr> <td>2.99396</td> <td>66 00 00 00 24 00</td> </tr> </tbody> </table>	t [s]	Hex_3er_Messages_S1_K2	2.70843	64 00 00 00 00 00 00 00 fe ff	2.72640	65 00 00 00 00 fe ff 00 00	2.79394	66 00 00 00 22 00	2.80843	64 00 00 00 00 00 00 00 fe ff	2.82641	65 00 00 00 00 fe ff 00 00	2.89394	66 00 00 00 23 00	2.90843	64 00 00 00 00 00 00 00 fe ff	2.92641	65 00 00 00 00 fe ff 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 fe ff																				
2.72640	65 00 00 00 00 fe ff 00 00																				
2.79394	66 00 00 00 22 00																				
2.80843	64 00 00 00 00 00 00 00 fe ff																				
2.82641	65 00 00 00 00 fe ff 00 00																				
2.89394	66 00 00 00 23 00																				
2.90843	64 00 00 00 00 00 00 00 fe ff																				
2.92641	65 00 00 00 00 fe ff 00 00																				
2.99396	66 00 00 00 24 00																				

Flexray-message	<p>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.</p> <table border="1"> <thead> <tr> <th>t [s]</th> <th>Flexray_Messages_S1_K2</th> </tr> </thead> <tbody> <tr> <td>33.81970</td> <td>65 0000 00: 00 00 00 00 fe ff 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 fe ff</td> </tr> <tr> <td>33.91970</td> <td>65 0000 00: 00 00 00 00 fe ff 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 fe ff</td> </tr> <tr> <td>34.01970</td> <td>65 0000 00: 00 00 00 00 fe ff 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 fe ff</td> </tr> </tbody> </table>	t [s]	Flexray_Messages_S1_K2	33.81970	65 0000 00: 00 00 00 00 fe ff 00 00	33.88721	66 0000 00: 2d 00	33.90171	64 0000 00: 00 00 00 00 00 00 fe ff	33.91970	65 0000 00: 00 00 00 00 fe ff 00 00	33.98723	66 0000 00: 2e 00	34.00171	64 0000 00: 00 00 00 00 00 00 fe ff	34.01970	65 0000 00: 00 00 00 00 fe ff 00 00	34.08724	66 0000 00: 2f 00	34.10171	64 0000 00: 00 00 00 00 00 00 fe ff
t [s]	Flexray_Messages_S1_K2																				
33.81970	65 0000 00: 00 00 00 00 fe ff 00 00																				
33.88721	66 0000 00: 2d 00																				
33.90171	64 0000 00: 00 00 00 00 00 00 fe ff																				
33.91970	65 0000 00: 00 00 00 00 fe ff 00 00																				
33.98723	66 0000 00: 2e 00																				
34.00171	64 0000 00: 00 00 00 00 00 00 fe ff																				
34.01970	65 0000 00: 00 00 00 00 fe ff 00 00																				
34.08724	66 0000 00: 2f 00																				
34.10171	64 0000 00: 00 00 00 00 00 00 fe ff																				

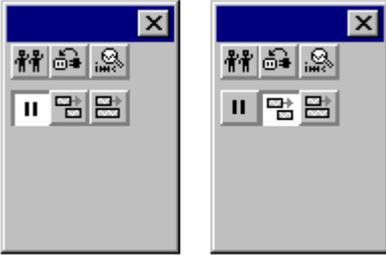
Multi-channel displays

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](#)¹⁰⁰⁸

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](#)¹⁰⁰⁸.

13.5.1.4 Bar meter

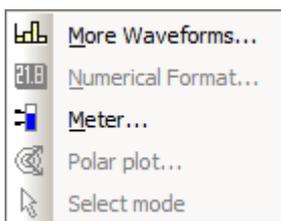
In addition to the numerical data display *Last value as number*, there is also a data display style having the form of a vertical meter bar. The height of the bar is determined from the last value in the waveform. This representation style is particularly useful for online viewing, as it offers a convenient 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:

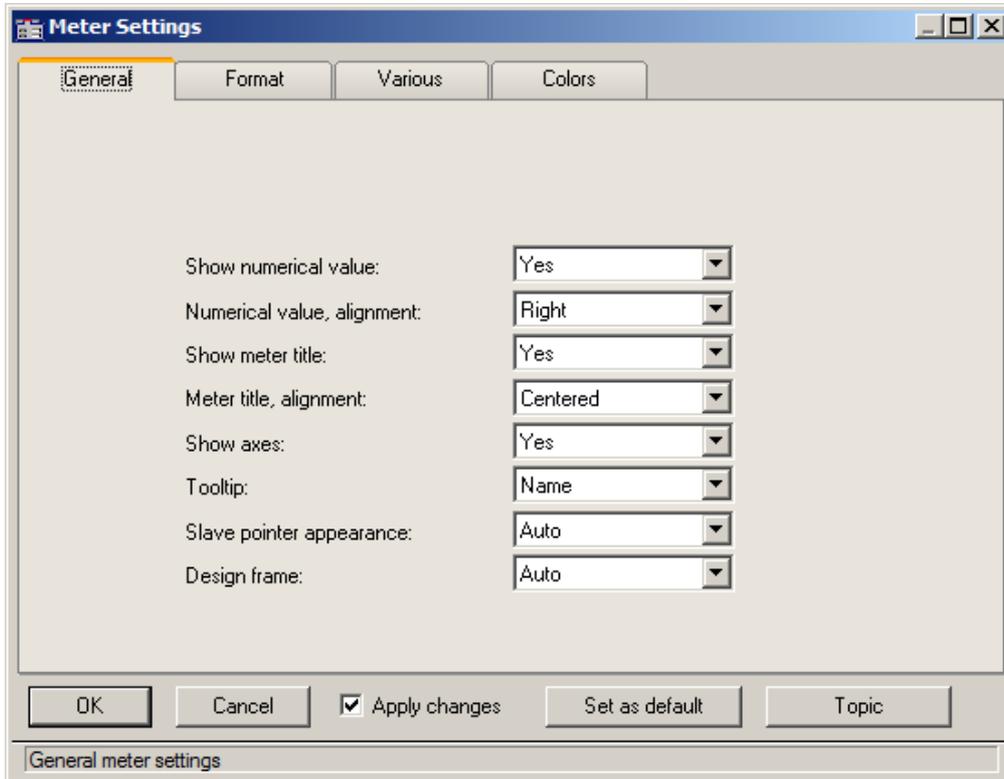
Select *Meter...* to open the *Meter Settings* dialog. This dialog consists of several sub-dialogs, each of which contains controls for different settings.

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.



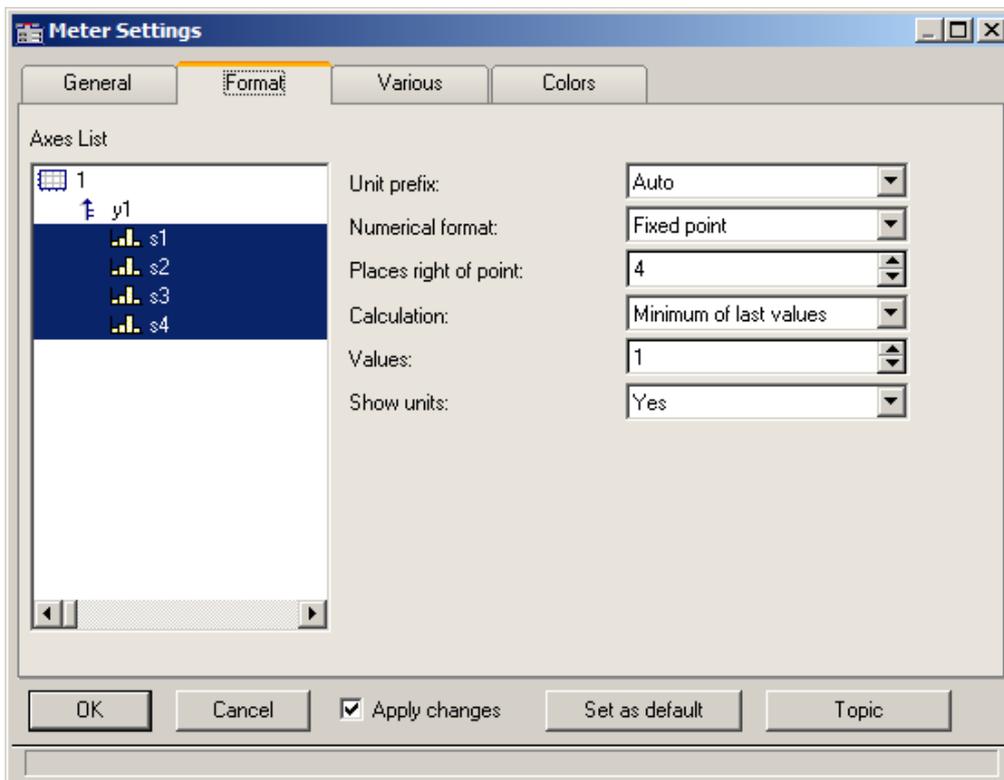
Settings	Description
Show numerical value	This setting governs whether to display the numerical value of a signal together with its bar meter representation. The numerical value is written above the bar.
Numerical value, alignment	This controls the horizontal alignment of the numerical value (in case 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	<p>With <i>auto</i>, each channel has its own frame; with <i>Common frame</i>, not.</p> 

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.

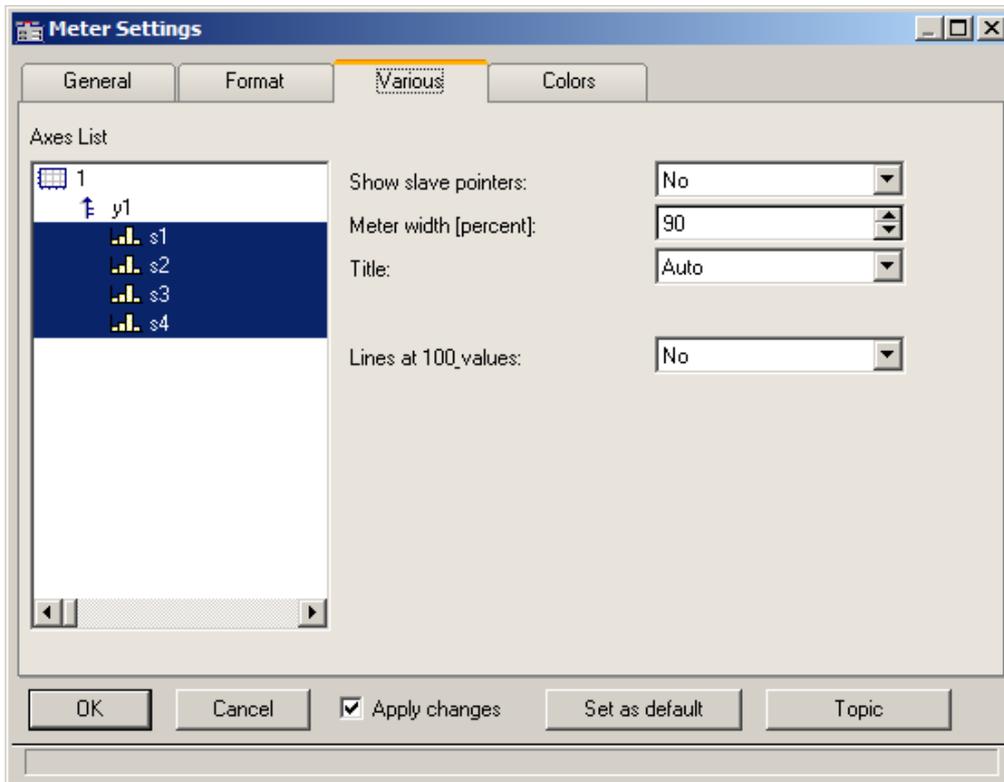
On this and the following sub-dialogs, the first step is to select all the channels to which the settings made should apply in common. Multi-selection can be performed as described in the instructions for the [axes list](#)⁹²⁶. If multiple, selected channels have mutually different settings for a property, the setting indicated for that property will be "???".



Settings	Description
Unit prefix	<p>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.</p> <p>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.</p> <p>This setting is unnecessary if the numerical value of the signal isn't set to be displayed.</p>
Numerical format	<p>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.</p> <p>This setting is unnecessary if the numerical value of the signal isn't set to be displayed.</p>
Places right of point	<p>Zero to fifteen places after the decimal point can be specified.</p> <p>This setting is unnecessary if the numerical value of the signal isn't set to be displayed.</p>
Calculation	<p>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.</p>
Values	<p>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.</p>
Show units	<p>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.</p>

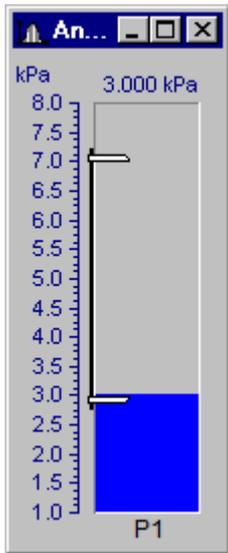
Settings: Various

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	<p>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.</p> <p>Fixed title: Enter here an arbitrary designation for a bar meter. Up to 20 characters are allowed.</p>

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.

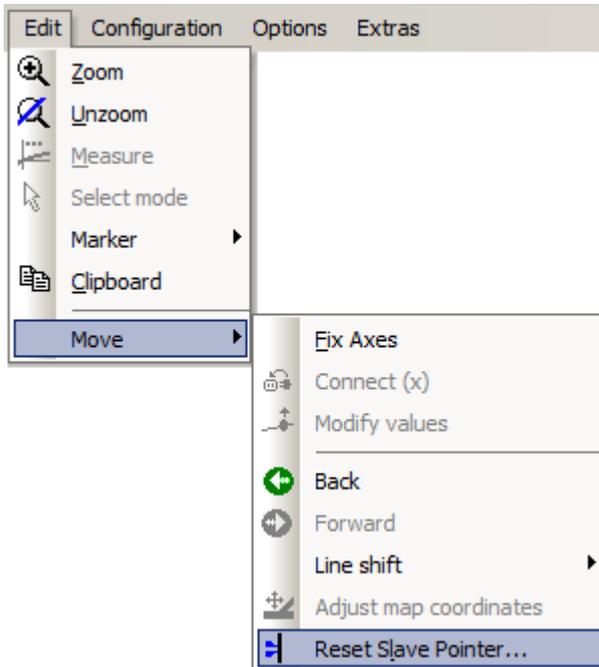
To avoid this effect from occurring, compute the average *online* and then display the resulting *virtual channel* in a bar meter. The slave pointers will then agree with the displayed maximum and minimum values.

Slave pointers are accurate to 6 significant figures, i.e. they use 4 Byte representation.

Slave Pointer: Reset

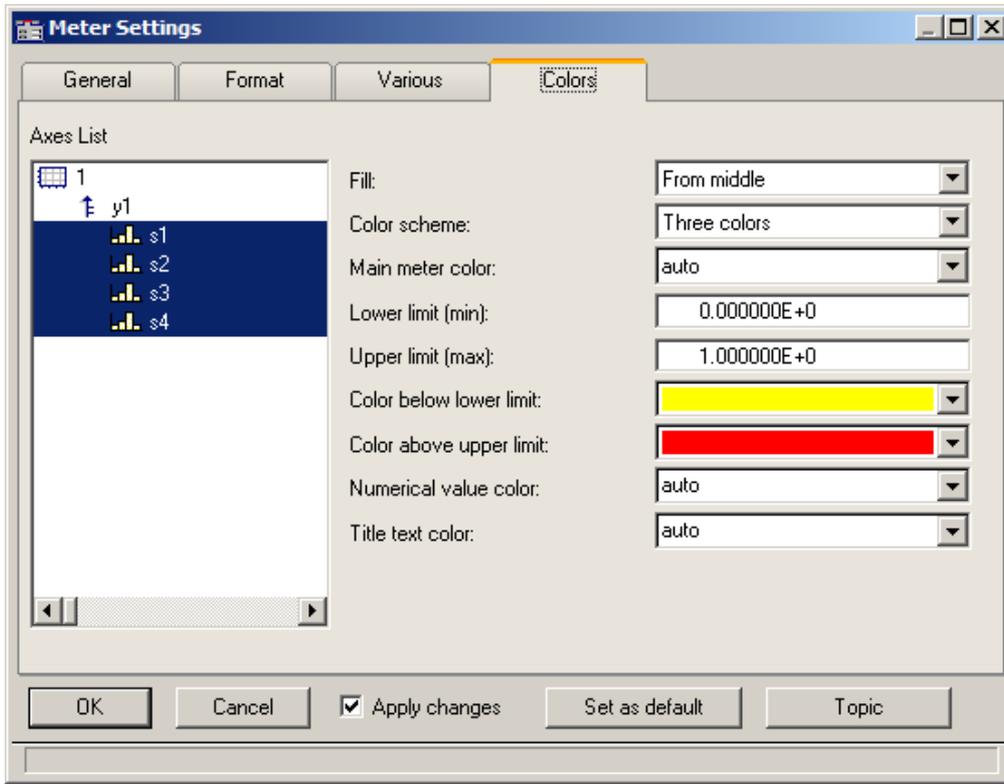
Resetting a slave pointer causes *all* slave pointers in that curve window to be reset. When you reset a slave pointer, it slides back to the current measurement value. Slave pointers have to be reset manually in each window. When a new measurement is started, this does *not* automatically reset the 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.

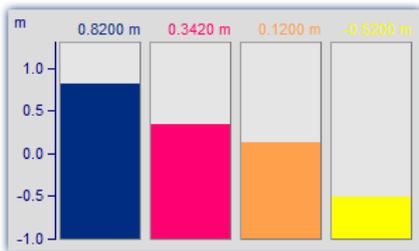


Fill

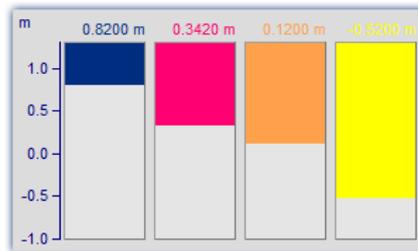
This setting determines from where the meter bars grow with increasing signal measurement values, as represented visually by a column filling with color.

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).

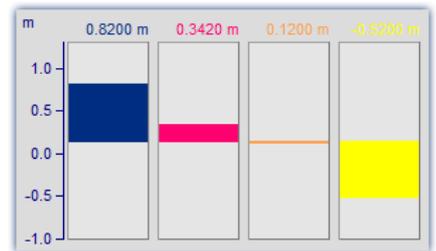
From below



From above



From $y = 0$

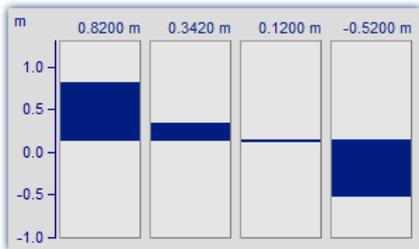


Alternating up and down from the 0 line of the axis

Color scheme

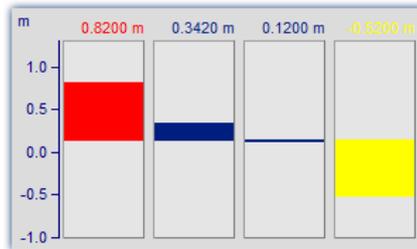
The bar can have only one color, three colors in succession or up to three colors at once. Three colored bars are used to indicate whether the measurement signal is currently within a tolerance range, which can be specified by the user in terms of an upper and a lower threshold value.

One color



A one-colored bar has the same color at all times.

Three colors



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 \leq Lower limit
- The 2nd color, if: Lower limit $<$ Value $<$ Upper limit
- The 3rd color, if: Value \leq Upper limit

Three colors



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 \leq Lower limit
- The 2nd color, in the range: Lower limit $<$ fill height $<$ Upper limit
- The 3rd color, in the range: Upper limit \leq 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 ¹⁰⁷⁸ 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.

13.5.1.5 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](#)⁹⁶².

Prerequisites

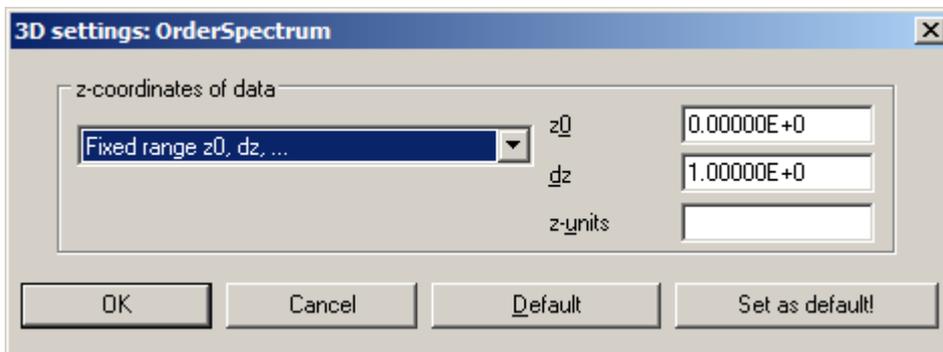
- Color video card with more than 256 colors, in other words 16 bit or 24 bit colors.
- Color printer of DIN A4 format.
- A fast PC (≥ 586 , 100MHz) with plenty of RAM (≥ 32 MB)
- Since the graphics display is based on large bitmaps (pixel graphics), large displays in particular require much storage 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](#)¹⁹⁰⁰ of the curve window, select an appropriate z-coordinate (see below):

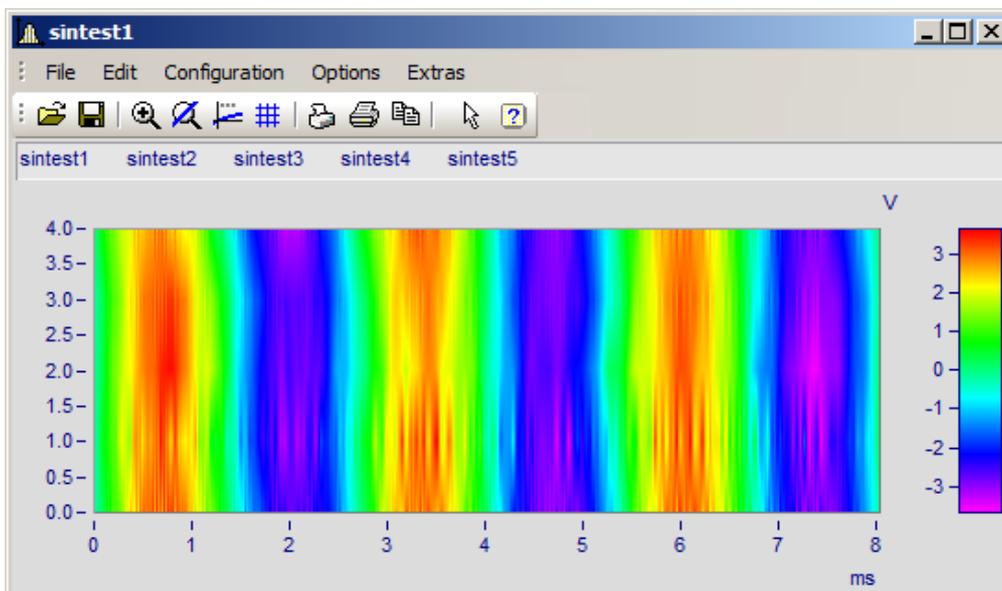


Reference

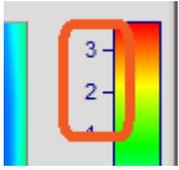
[Waterfall diagrams z-coordinate](#)

More on the topic of [Waterfall diagrams z-coordinate](#)⁸⁶²

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](#) .

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.

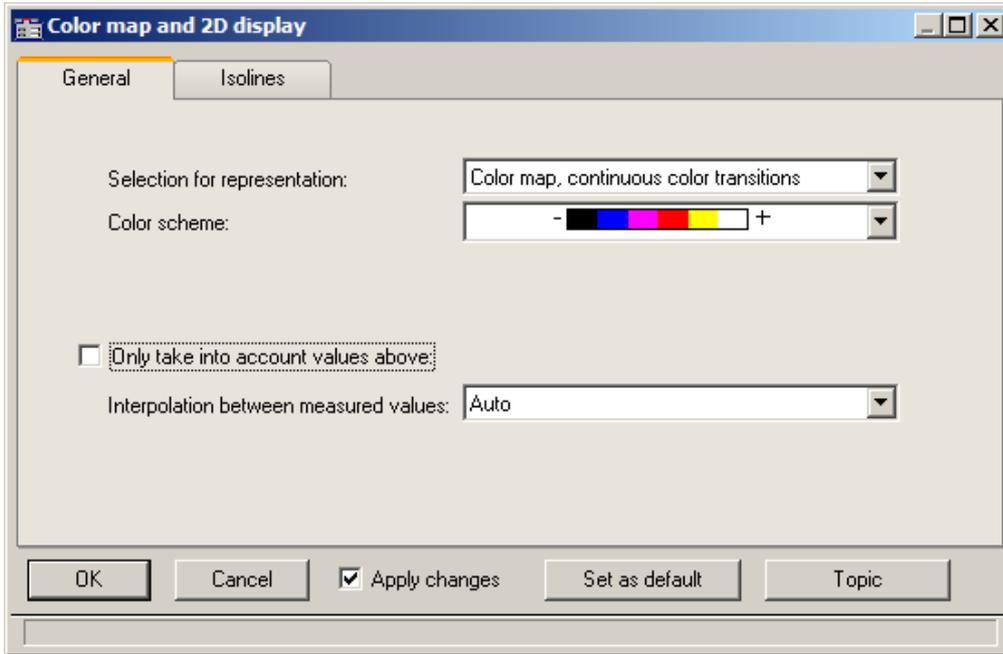
We thus achieve the illusion of a colorful landscape depicted on a map with green signifying valleys, the mountains brown and the snowy peaks white (in our case the colors follow the order of the physical spectrum).

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.

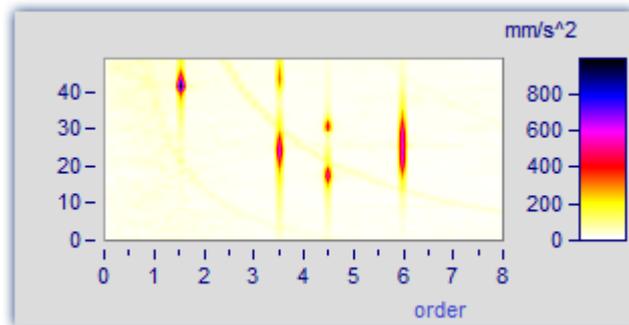
13.5.1.5.1 Color map dialog, "General"

The options for color map display are set in the dialog called by means of the menu item *Configuration/Color Map*:



Selection for representation

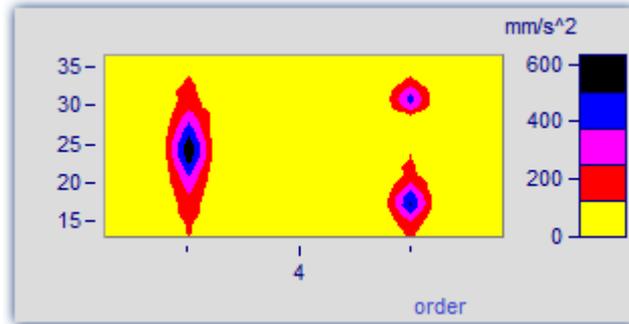
Representation	Description
auto	Default colors from magenta to red, or from white to black for B/W display. No other options.
Color map, continuous color transitions	Interpolated color transitions. For each amplitude to represent, an appropriate color is calculated. The color scheme can be selected.



Representation	Description
----------------	-------------

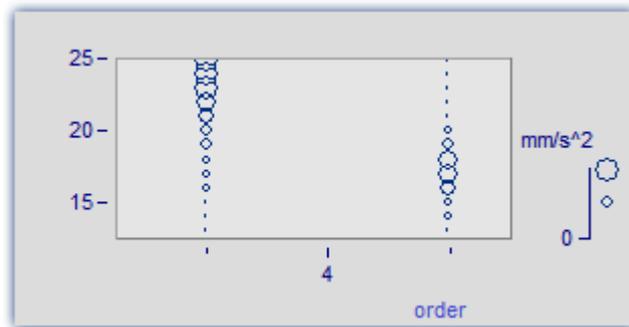
Color map, graduated colors

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 *Color of the symbol* corresponds to that of the first curve and can not be changed here.
- ◻ Under *Symbol selections*, 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 *Fill color* and *Frame color* can be adjusted. For the frame, the setting *auto* is available, which represents the color of the first curve. For *transparent*, 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.

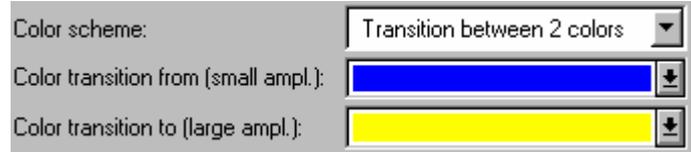
Color scheme	Description
--------------	-------------

auto	For color output devices, the colors Magenta to Red are used; for b&w printers, the shades from White to Black.
------	---

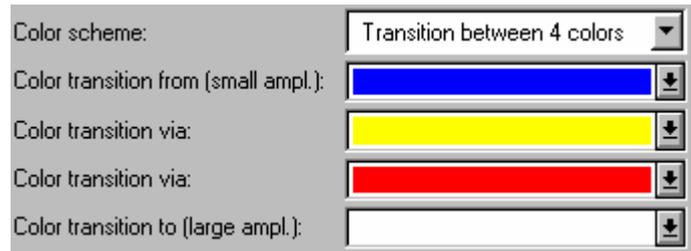
Color scheme	Description
--------------	-------------

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.



Dialog with 2 colors.



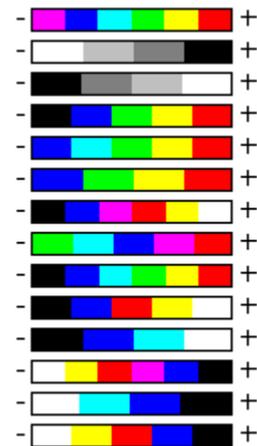
Dialog with 4 colors.

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.

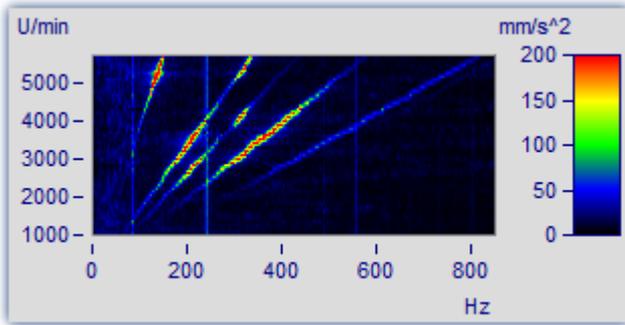
Fixed color schemes

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

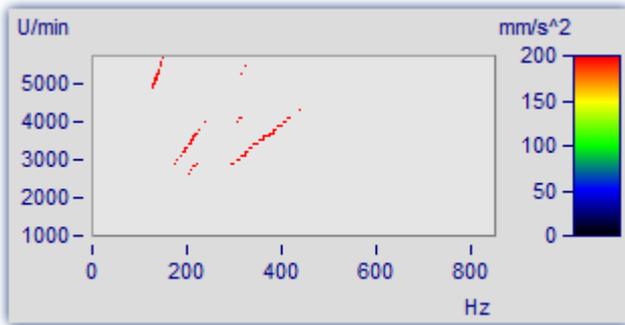
The options depend on what display type is selected.



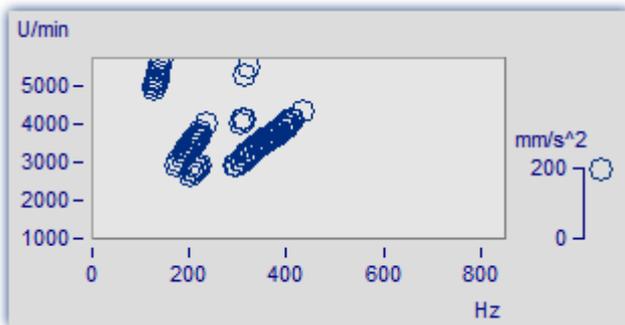
If this option box is not check-marked, all readings are taken into account (default).

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.

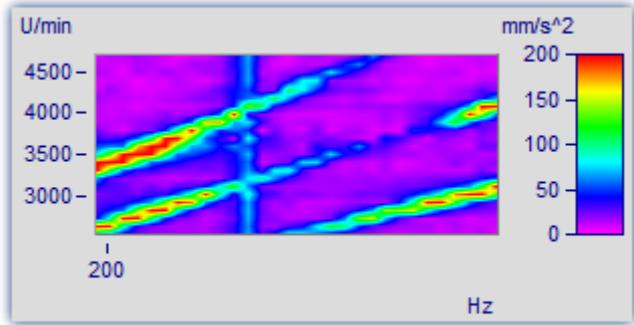
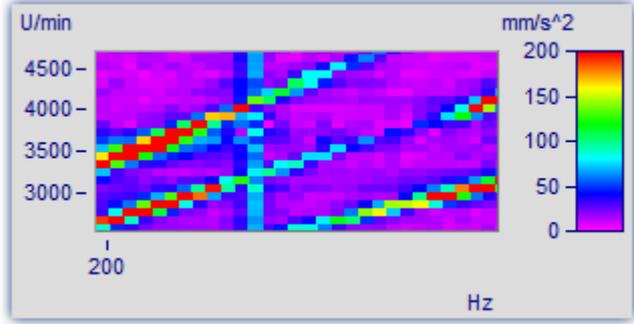
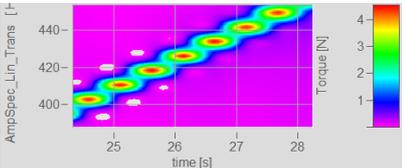


The same graph in the mode *Symbols*, size corresponding to amplitude (*Campbell*)

Interpolation between measured values

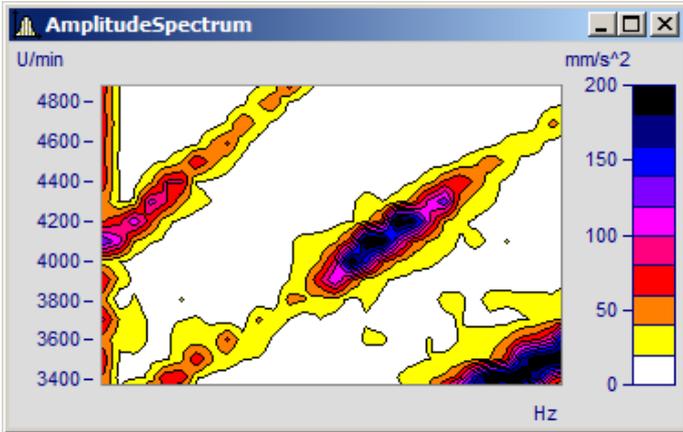
The options depend on what display type is selected.

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. 

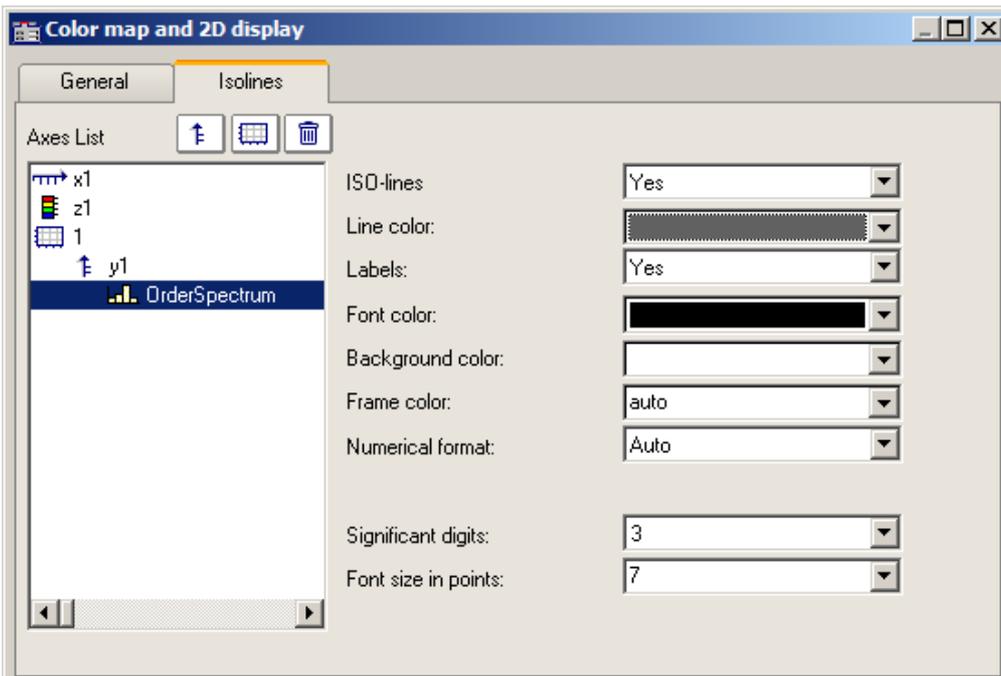
13.5.1.5.2 Color map dialog, ISO-lines

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.



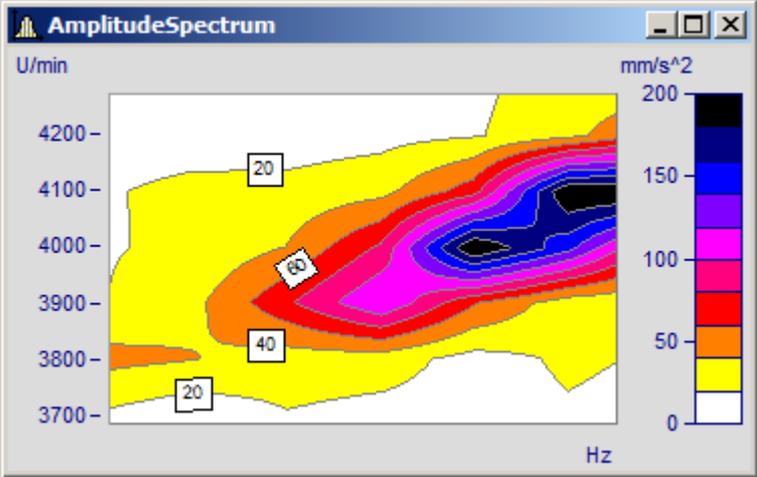
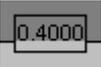
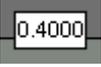
ISO-lines can only be set for [color map display with graduated colors](#) ⁸⁹².

The "Isolines" index card of the Color map settings dialog



The following options can be selected:

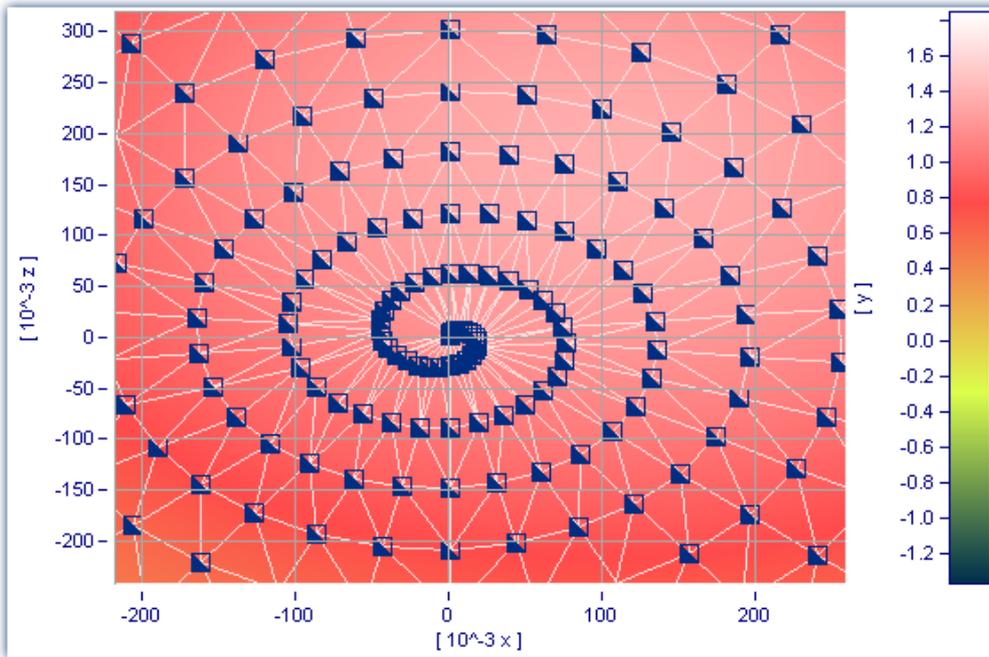
Options	Description
ISO-Line	Display of the isolines. This option is only available for the display types <i>Color maps</i> , <i>graduated colors</i> .
Line color	The color of the isolines is set here.

Options	Description
Labels	<ul style="list-style-type: none"> • <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.
	
<p>All further options only pertain to the labeling of the isolines.</p>	
Font color	Determines the color of the label text.
Background color	<p>Determines the color of the label background.</p> <ul style="list-style-type: none">  <ul style="list-style-type: none"> • <i>auto</i>: Transparent background, but covering the contour line.  <ul style="list-style-type: none"> • <i>transparent</i>: The label background is totally transparent. The contour line between the colors is visible as well.  <ul style="list-style-type: none"> • <i>Fixed color</i>: The label background is given a fixed color and is thus not transparent.
Numerical format	<ul style="list-style-type: none"> • <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	<p>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.</p> <p>For <i>automatic numerical format</i>, the same order of magnitude is set as for the labeling of the z-axis.</p>
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](#) 862

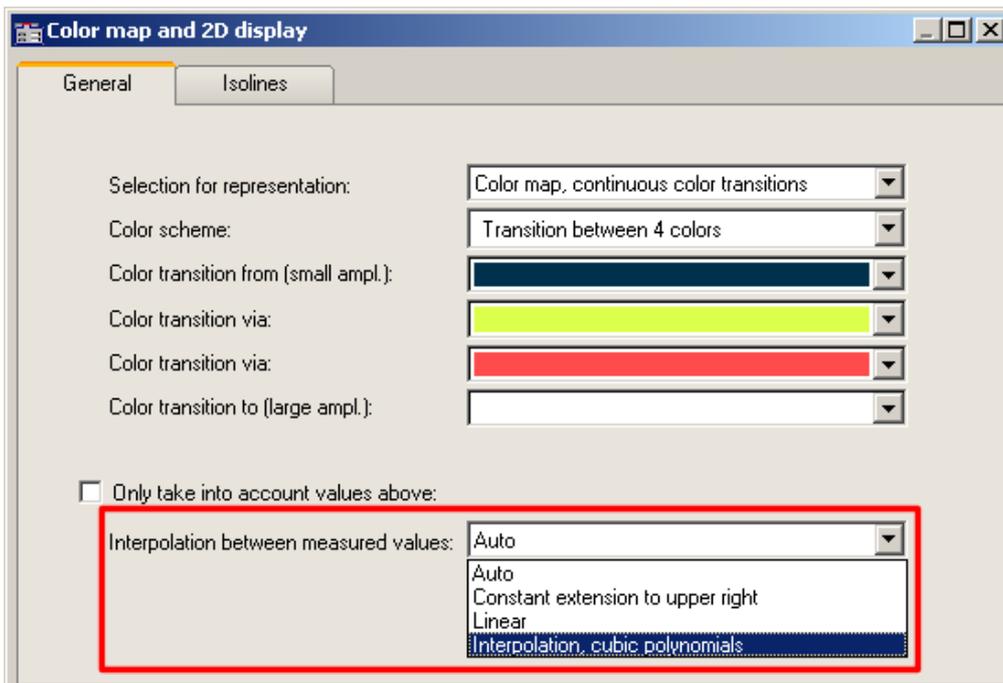
13.5.1.5.3 Waveforms with associated x-,y-,z-variables

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

13.5.1.6 3D display

Another display style for datasets with [associations of x, y, and z variables](#)^[931], for segmented waveforms is 3D display.

In this display style, the perspective can be freely rotated, and the [Axis Navigation Bar](#)^[1101] 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.

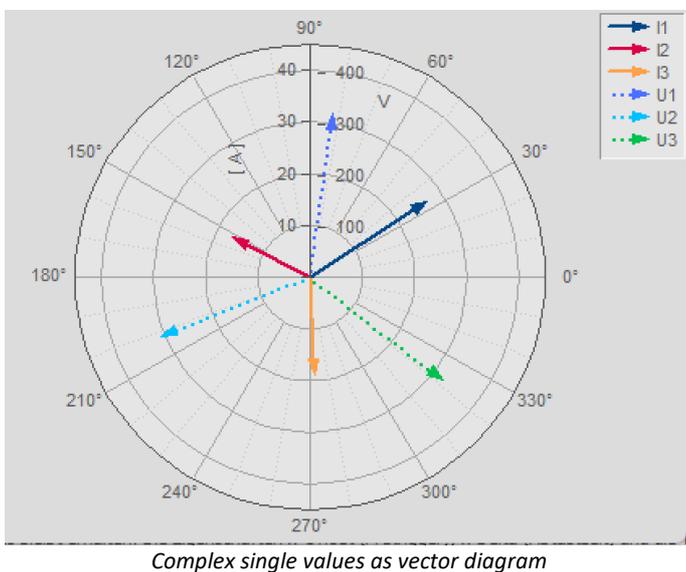
Other curve window functions such as *zooming*, *setting markers* or *line-shifting* are not available here.

For more on using 3D-display, see the section on [Configuration / 3D](#)^[989].

13.5.1.7 Polar plot

Complex data sets can be displayed as a polar diagram. To do this, select from the menu *Configuration\Display* the the display type *Polar plot*.

Example: Pointer diagram

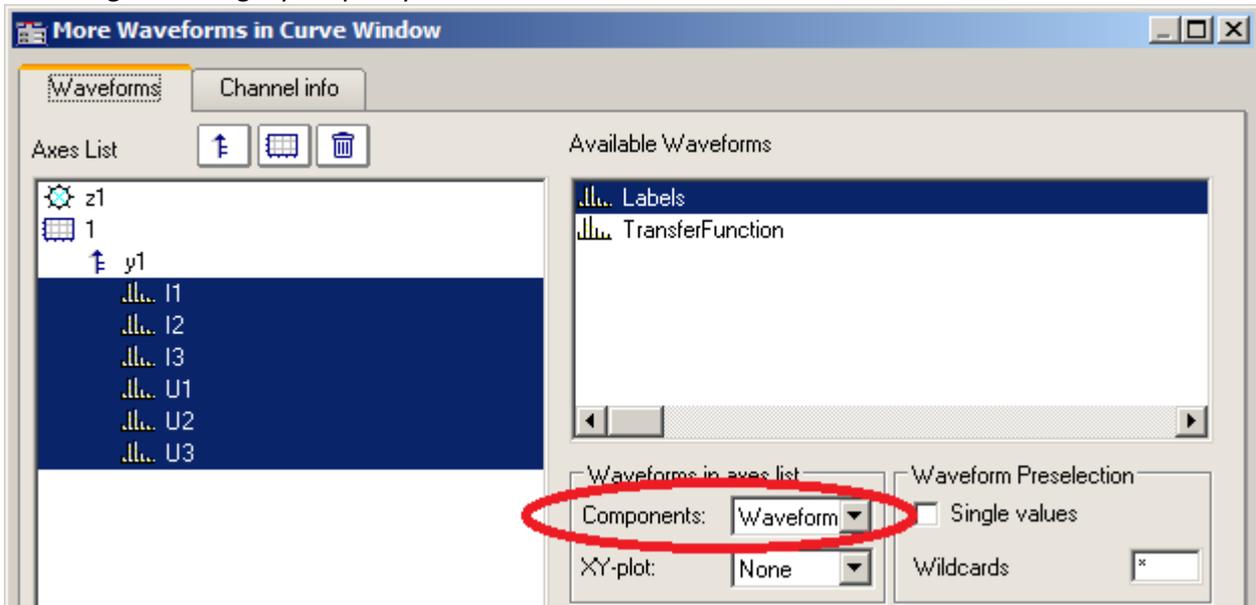


In imc FAMOS you can create a complex single value using the function `Cmpl()`.

The pointer graph above is created as follows:

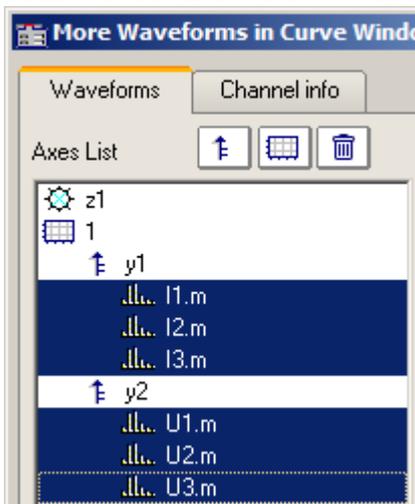
1. Create the desired single values:
 $I1 = \text{Cmpl}(27'A', 33'Degr')$
 $I2 = \text{Cmpl}(17'A', 153'Degr')$
 $I3 = \text{Cmpl}(19'A', 273'Degr')$
 $U1 = \text{Cmpl}(327'V', 82'Degr')$
 $U2 = \text{Cmpl}(315'V', 202'Degr')$
 $U3 = \text{Cmpl}(331'V', 322'Degr')$
2. Display the single values in a curve window and select the display style *Polar diagram* 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 *Waveforms in axes list\Components: Waveform*. The arrows will then be drawn according to the angle you specify.



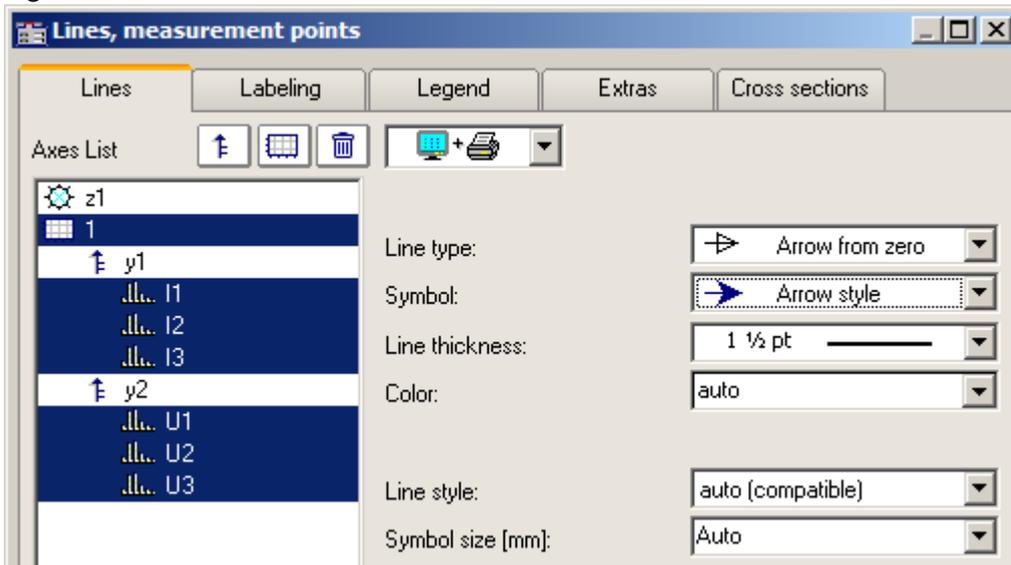
Observing each component of a complex waveform

4. As in a normal curve window, all variables are assigned to an axis. The current which are comparatively low in magnitude therefore appear in the zero point. Create one axis each for current and voltage: right mouse button for context menu -> "More Channels"



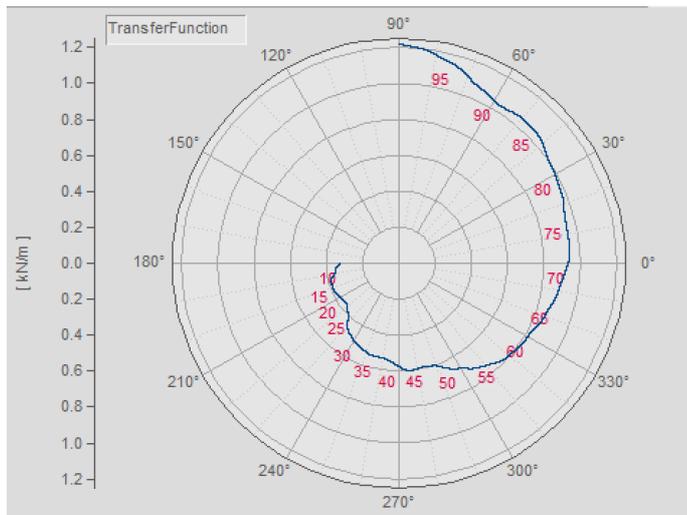
5. Using the Lines dialog, you can now adapt the display of the arrows.

Right mouse button: "Lines"



Dialog lines: Display with arrows

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`

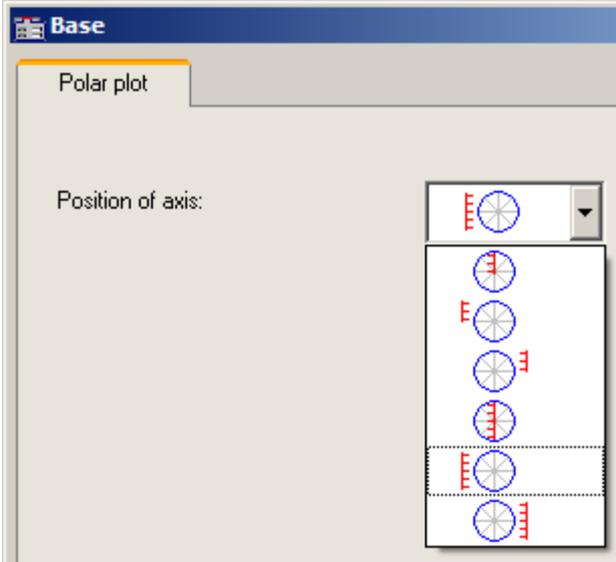
`TransferFunction = compl (m, p)`

`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.

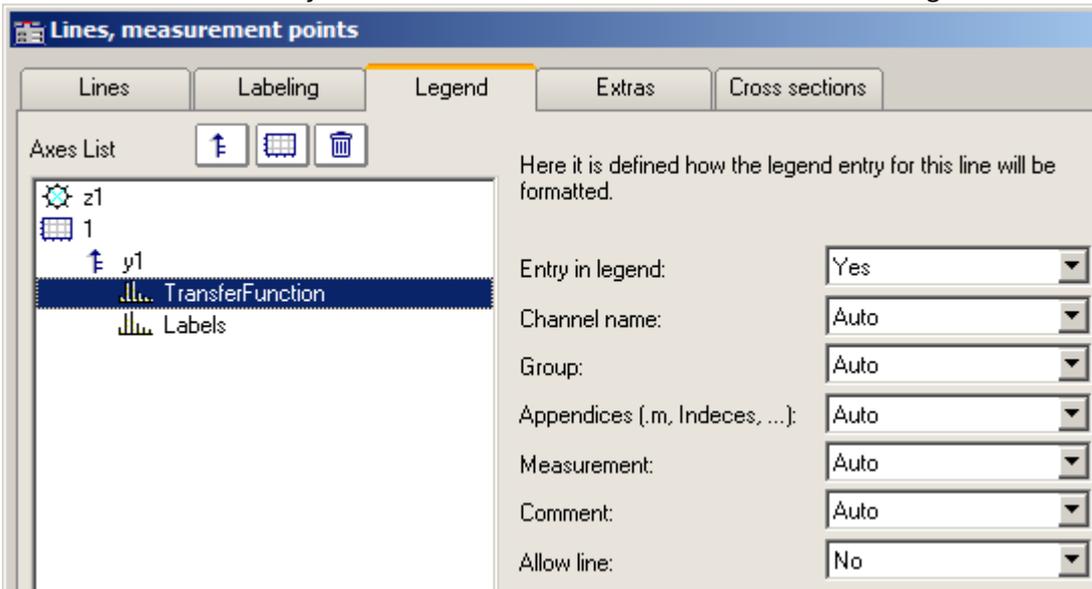
2. Display the data set *TransferFunction* in a curve window and select the display style *Polar plot* in the menu *Configuration\Display*.
3. Right-click the mouse to open the context menu and there select *More Channels...* Select the variable and under *Waveforms in axes list\Components: Waveform*. 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*.



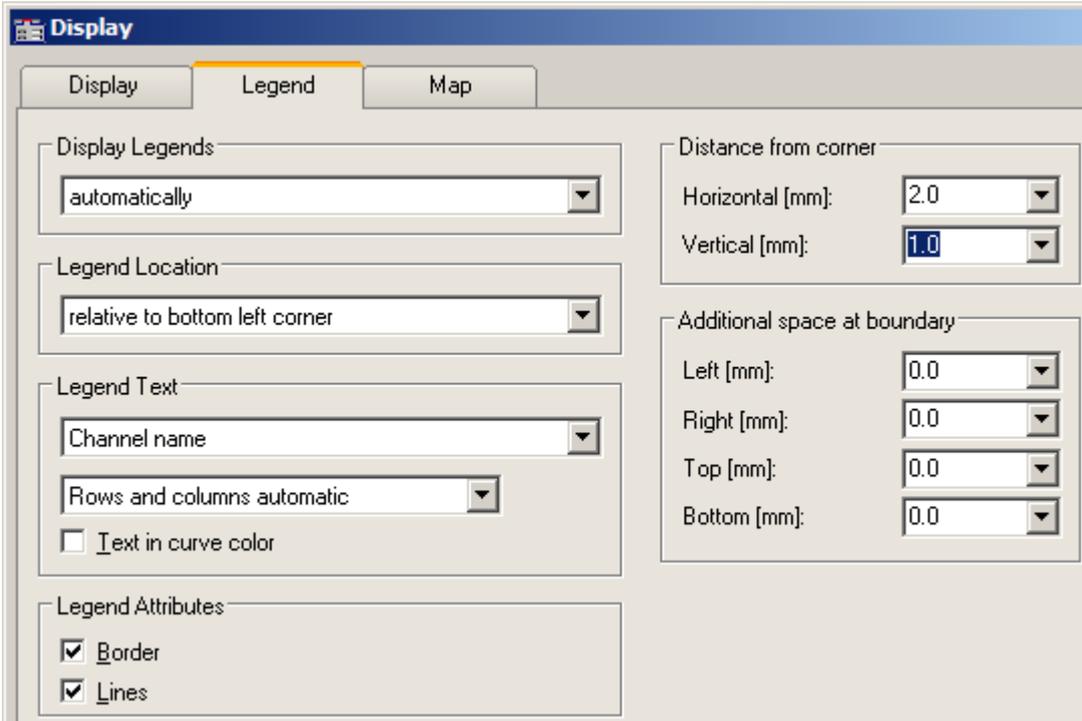
Position of axis

- 5. Labeling the values:** The individual values could be labeled in any curve window 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 an image of *TransferFunction* reduced by a factor of 500. Select the right mouse button: *More Channels...* and assign *Labels* to the same axis. Click the button *Topic* and select *Lines*. 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*.
- 6. Legend:** The legend's position can be moved. First of all, 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*. Close the dialog.



Legend display on/off for each channel separately

- Open the dialog *Configuration \ Legend*. Position the legend in the upper left corner at some distance from the edge.



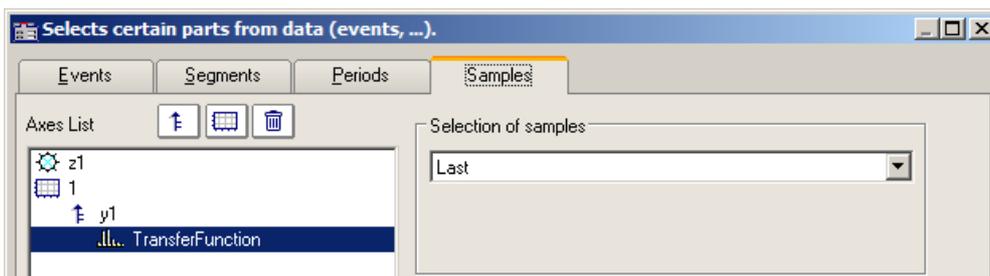
Position of legend

Note

Notes on online display

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...](#)⁹³¹.

If one wishes to display the complex data as a [pointer diagram](#)⁹⁰⁰ like above, then besides the procedure outlined above the following step is also necessary: Since only the channel's current value, in other words, the last one, is to be displayed, select *Configuration \ Events, segments periods...* Go to the page *Samples* and select *Last* under *Selection of samples*.



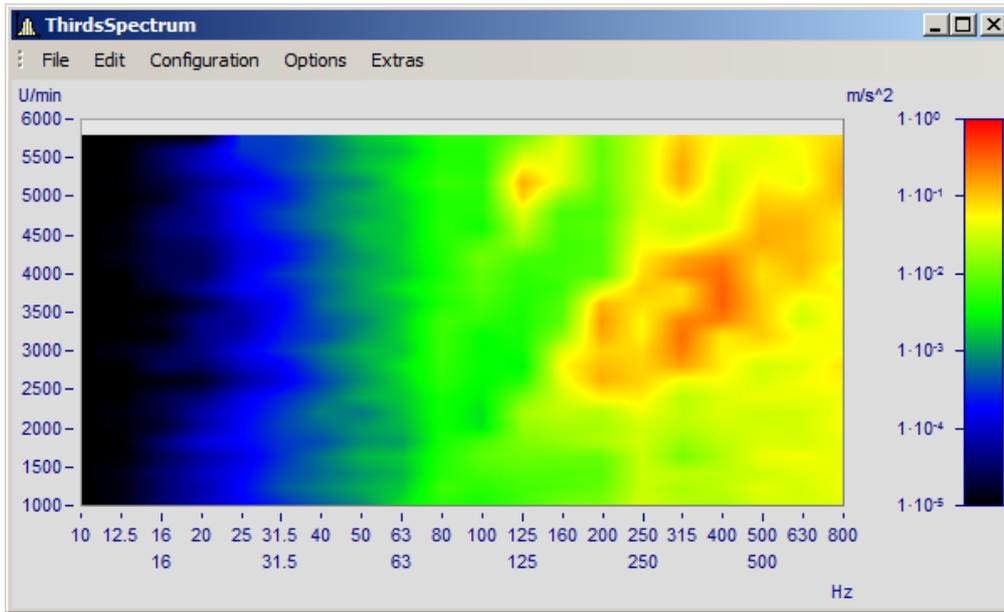
Displaying only the last sample

13.5.1.8 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.

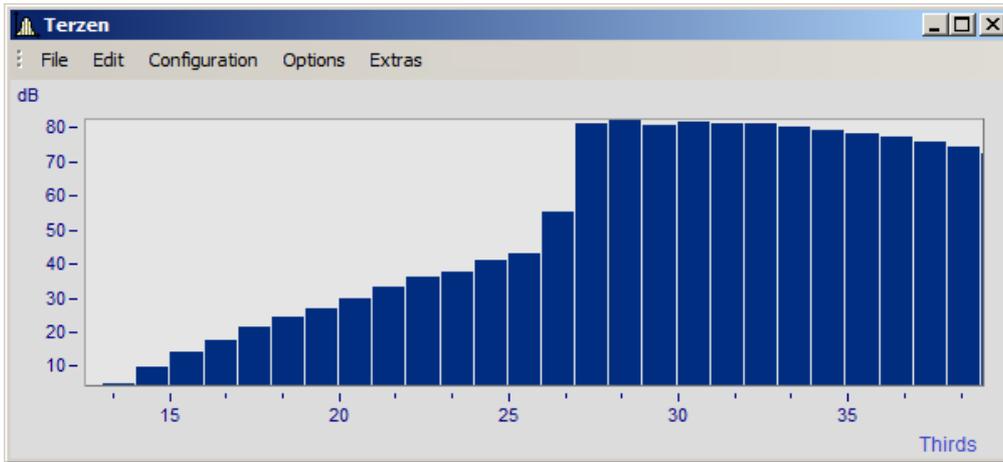
x-scaling	Center frequency (Hz)
..	..
-3	0.5
-2	0.63
-1	0.8
0	1
1	1.25
2	1.6
3	2
4	2.5
5	3.15
6	4
7	5
8	6.3
9	8
10	10
11	12.5
..	..
20	100
30	1000
40	10000
41	12500
43	20000
..	..

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.

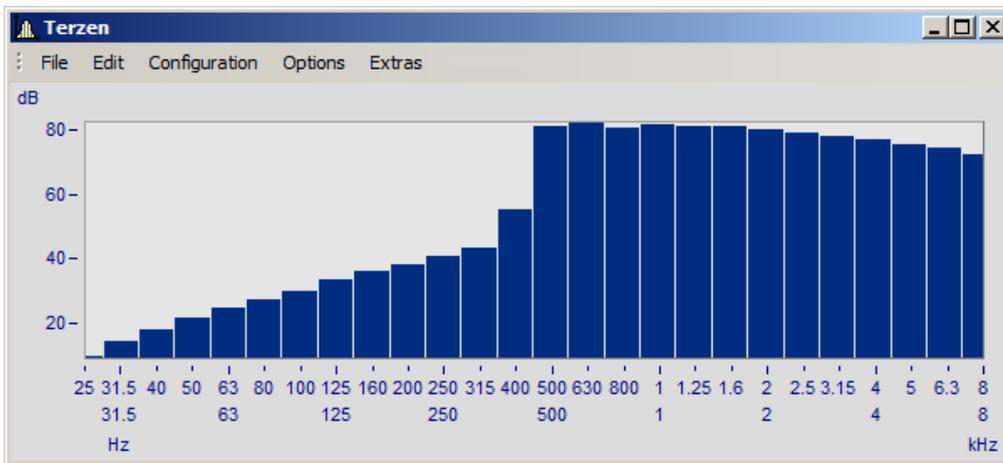
This means that the various bandwidths are expressed as a multiple of a 1/3-octave. Therefore, the delta-x of the x-axis is set to the following values for frequency scaling:

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

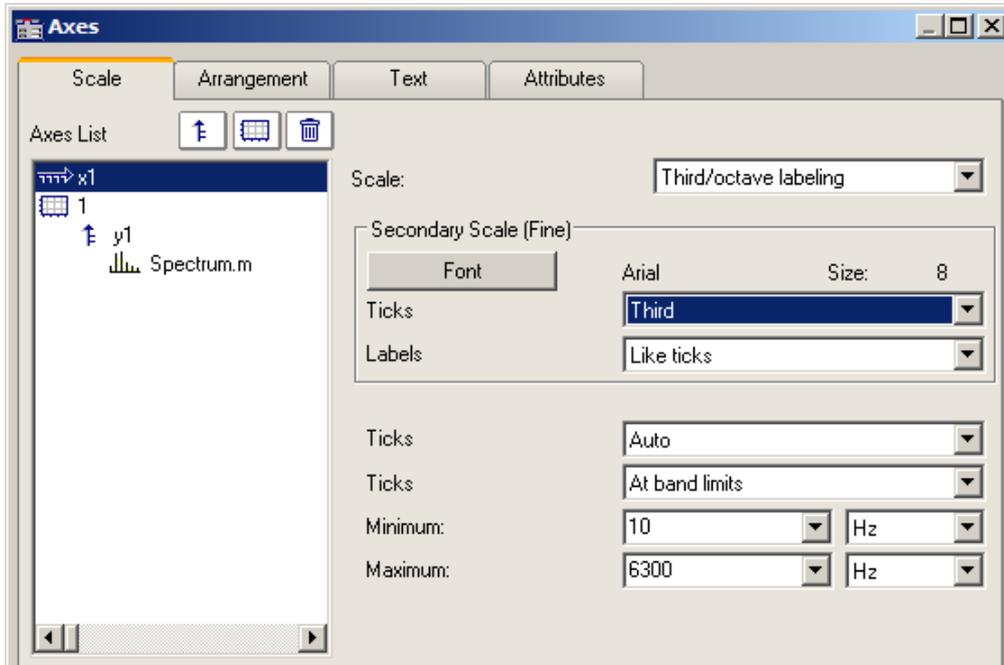


In 1/3-octave / octave labeling

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: <ul style="list-style-type: none"> • 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	<p>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.</p>
<div style="display: flex; justify-content: space-around; margin-top: 10px;"> <i>Ticks centered</i> <i>At band limits</i> </div>	

Minimum, Maximum	<p>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.</p> <p>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.</p> <p>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.</p>
------------------	--

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.

13.5.1.9 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.

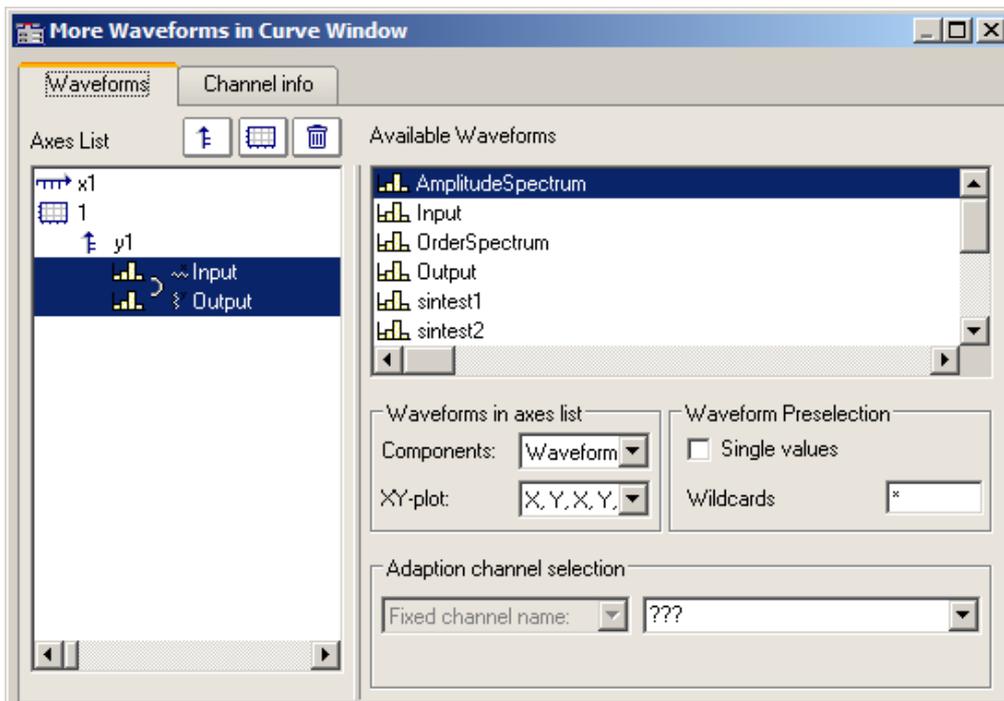
The curve window is also able to display XY-plots, meaning plotting one waveform's values in correlation to another waveform's values.

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

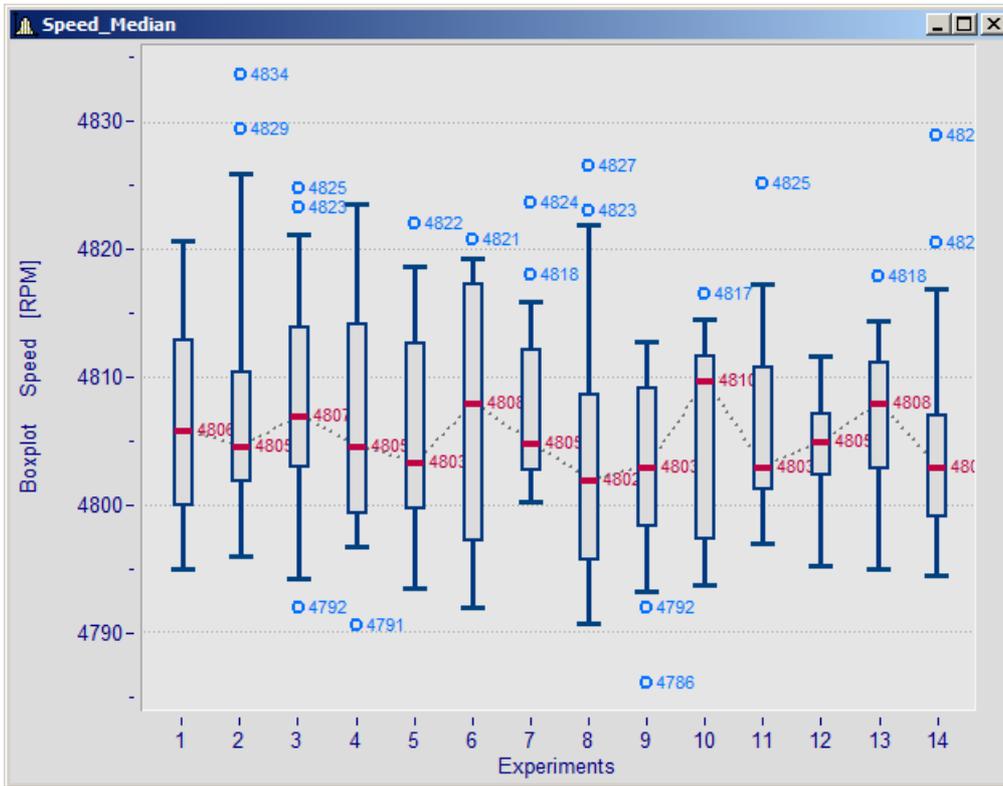
- In all XY-plots, the measurement cursors follow the parameters during operation in "Measure" mode. See chapter section '[Measure](#)'.
- Computing a polar plot requires less computation if the complex waveform is provided in rectangular coordinates. To increase the speed of the graphic display for 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.

13.5.1.10 Boxplot

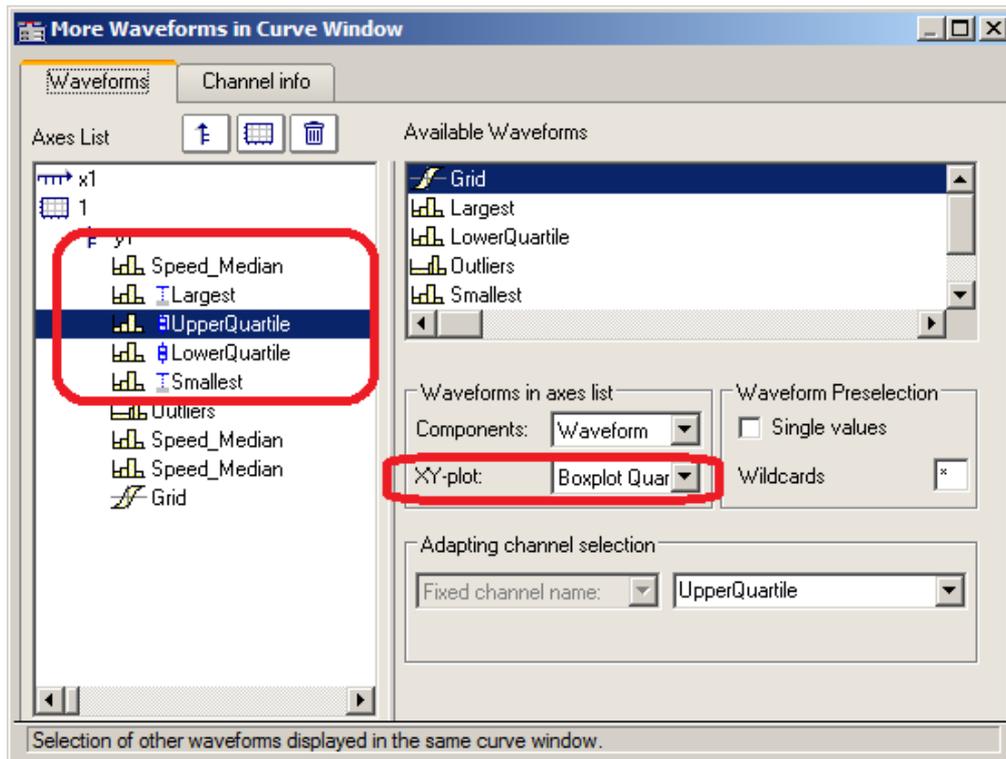
The Boxplot (also called Box Whisker Diagram) depicts a statistical distribution by displaying the minimum value, the maximum value, bottom quartile, top quartile and the median.

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...

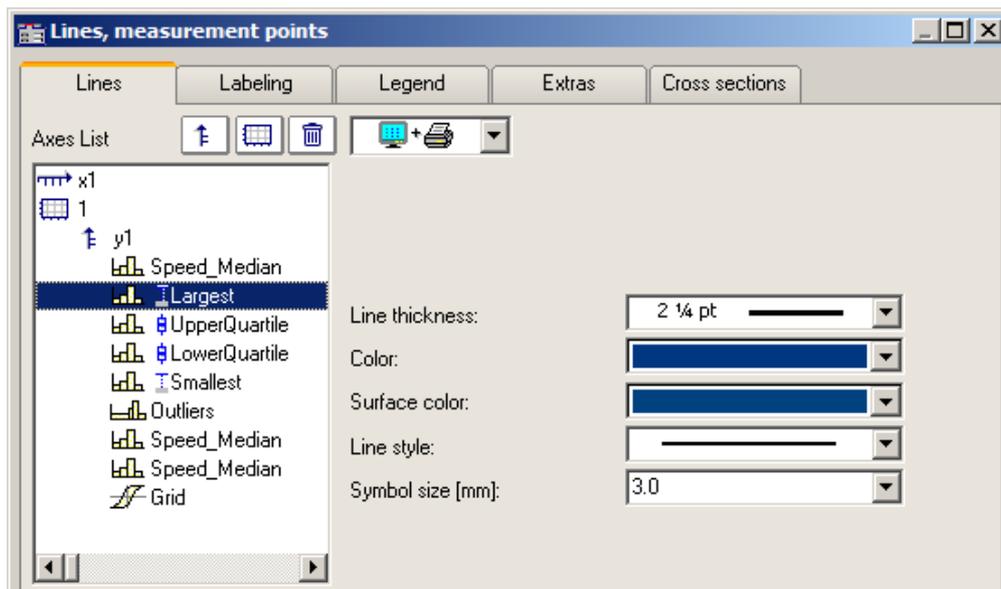


Making the settings:



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.

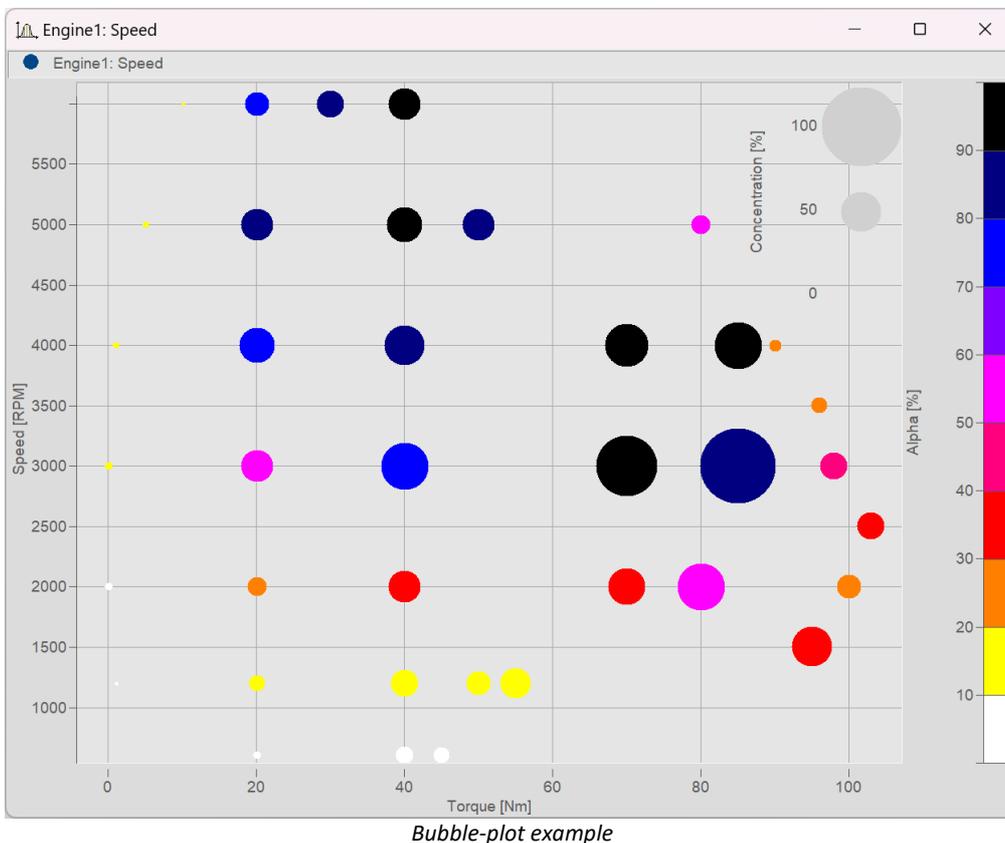
Remarks

- 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.

13.5.1.11 Bubble Plot

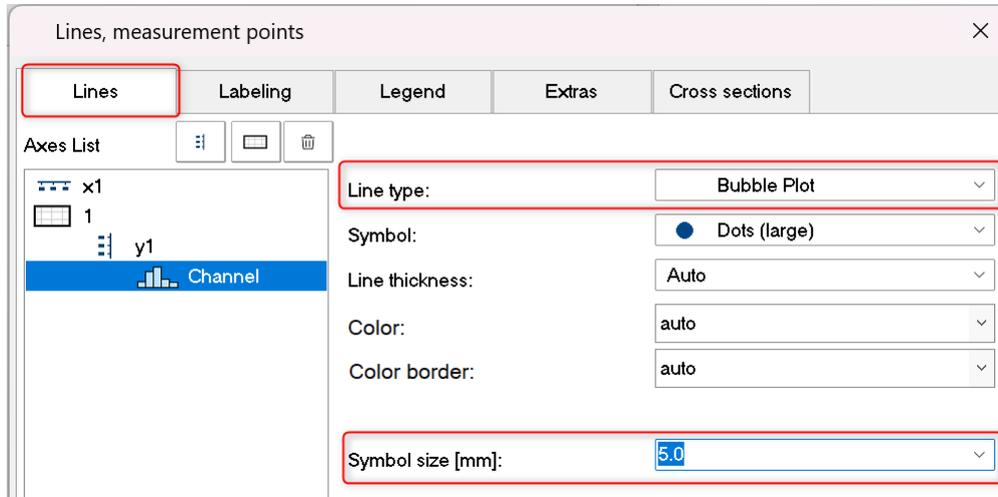
In a **Bubble Plot** (also called "Bubble Chart"), symbols are displayed which represent three (or optionally four) numerical values. Each point in the diagram is symbolized by a disk shape (bubble) in a typically Cartesian coordinate system. One data set (or in the case of XY, two data sets) determines 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.

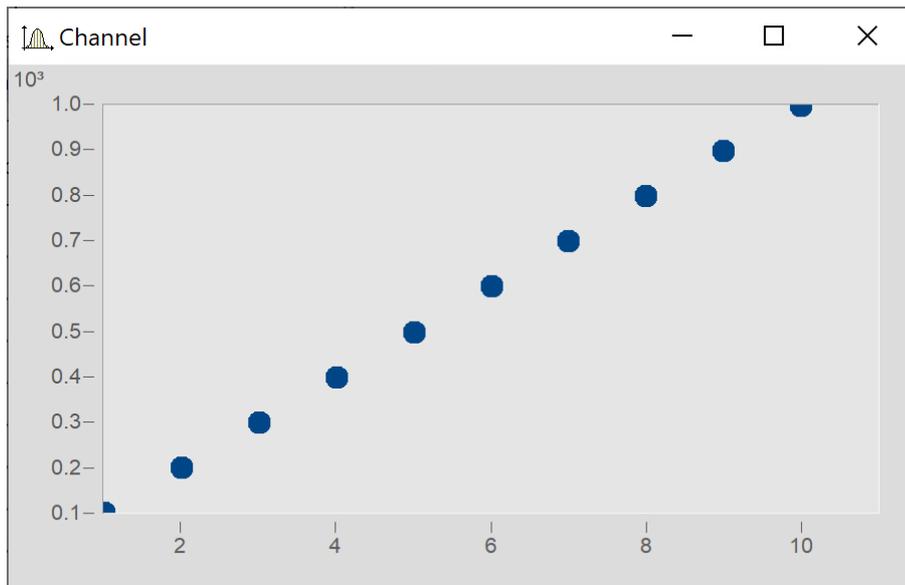


Settings:

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]*:

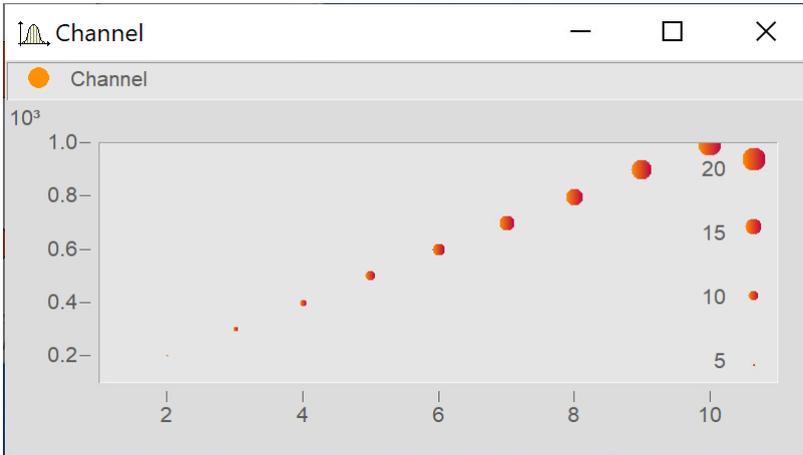


Depending on the *Symbol* selected, all values are displayed as *Dots (large)* etc. in the same size.

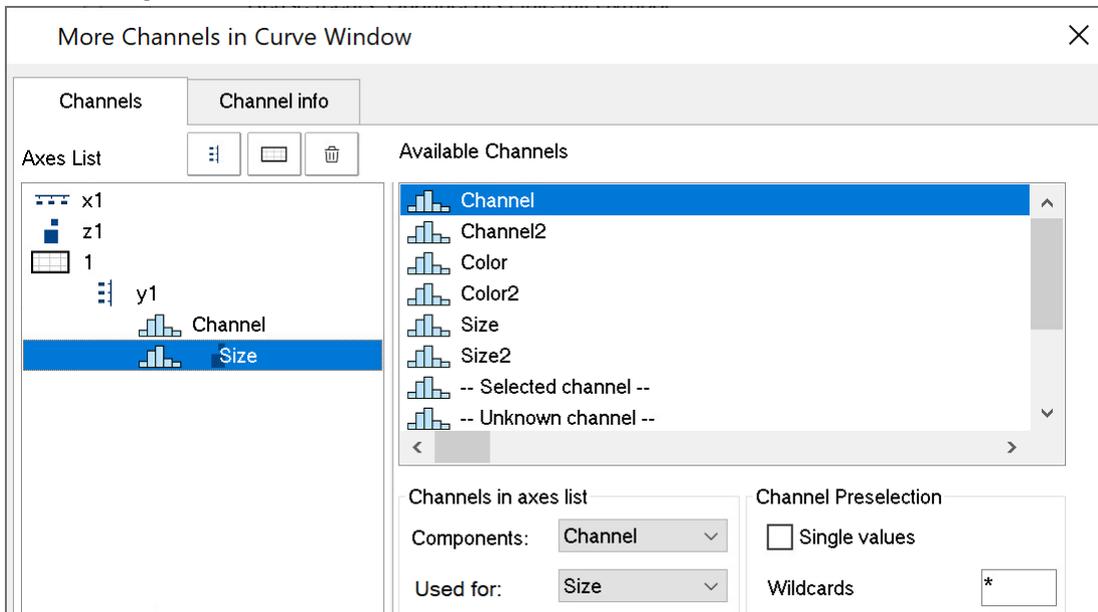


Size of bubbles corresponding to values of another data set

Next, the variable "Size" is intended to determine the size of the bubbles, as illustrated below:

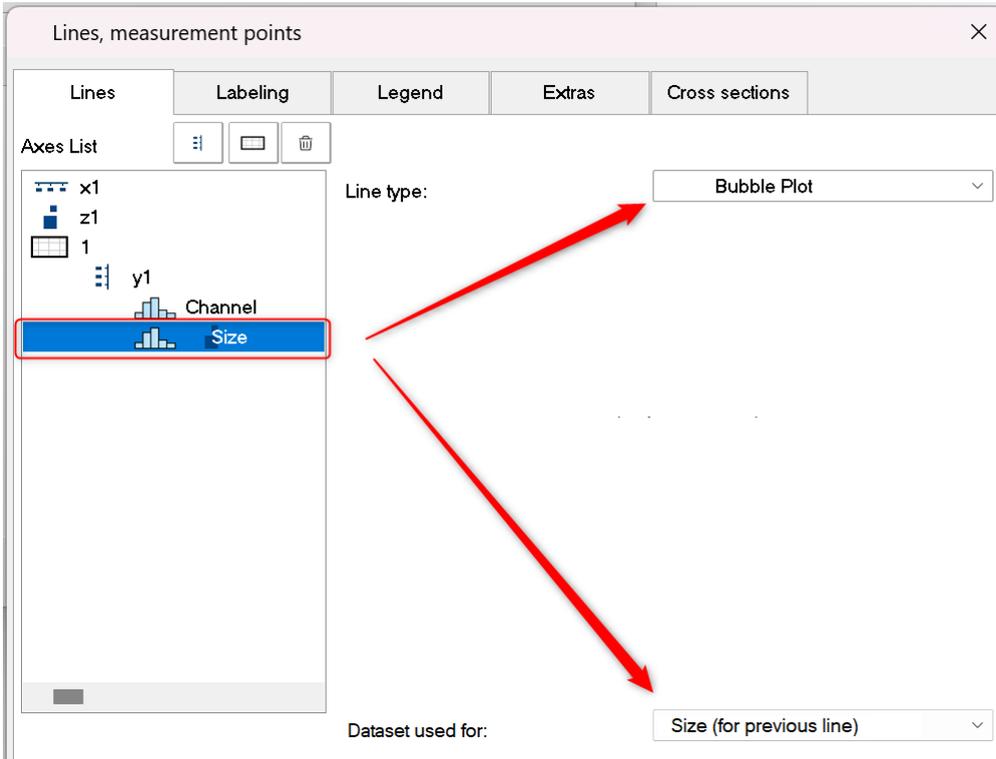


In the dialog **More Channels in Curve Window...** add the variable "Size".



Go to the dialog **Lines** by clicking on the button *Topic* and set the *Line type* for the variable "Size" also to *Bubble Plot*.

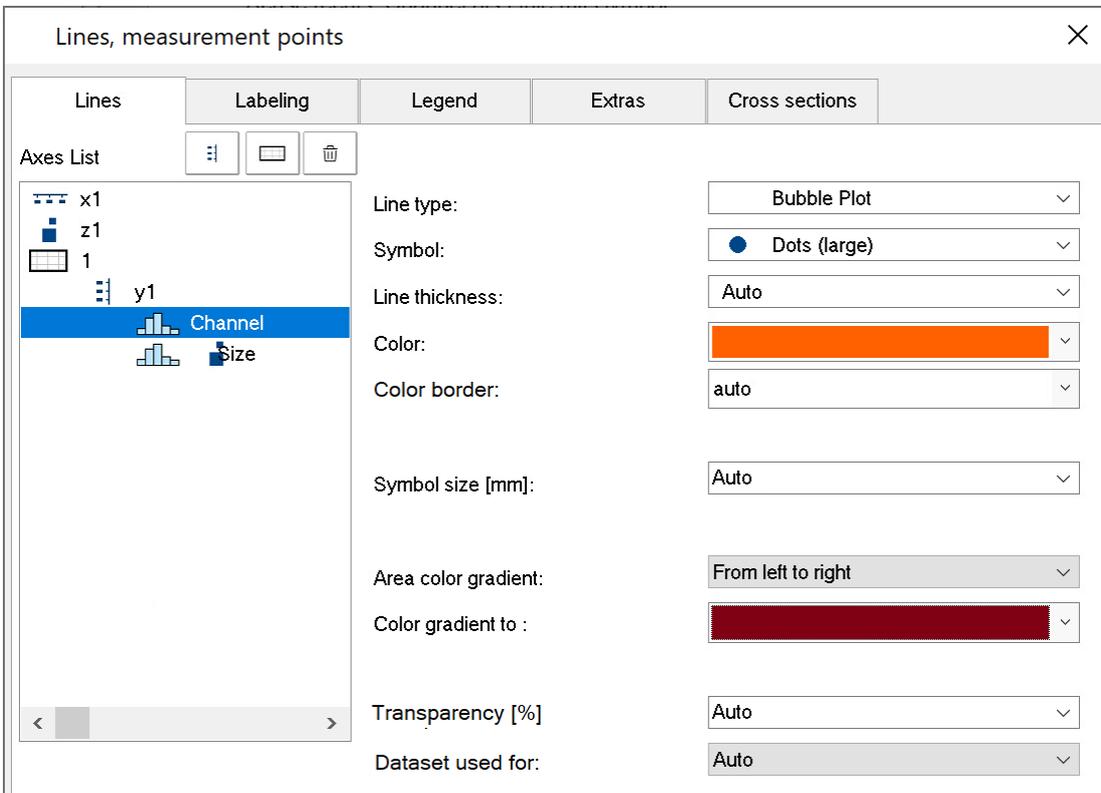
Next, under *Dataset used for:* select *Size (for previous line)*:



Next, in the dialog **More Channels...** set *Used for:* to *Size*

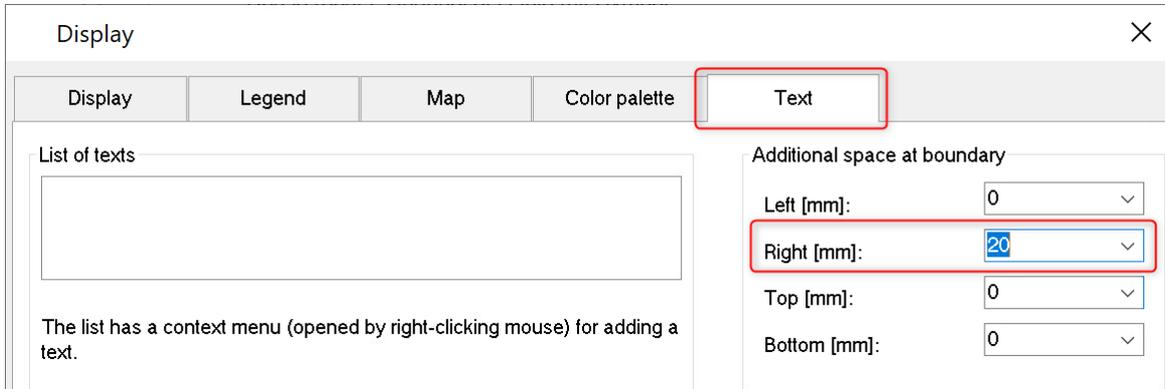
Bubbles with color gradient

Coloring the symbols with a color gradient:

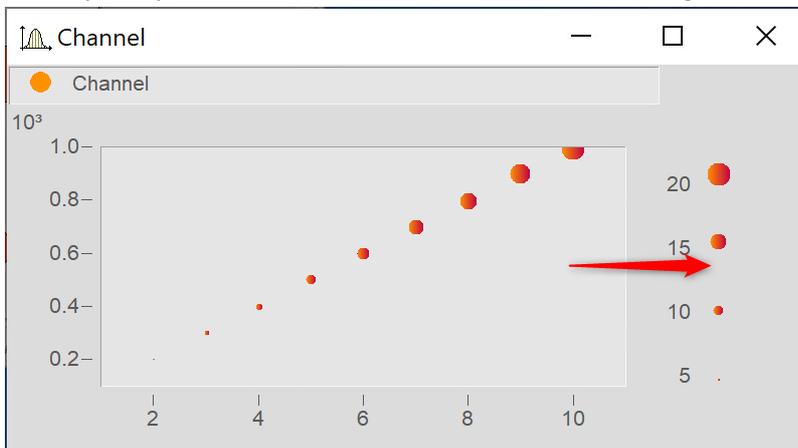


Adjusting the boundary

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:

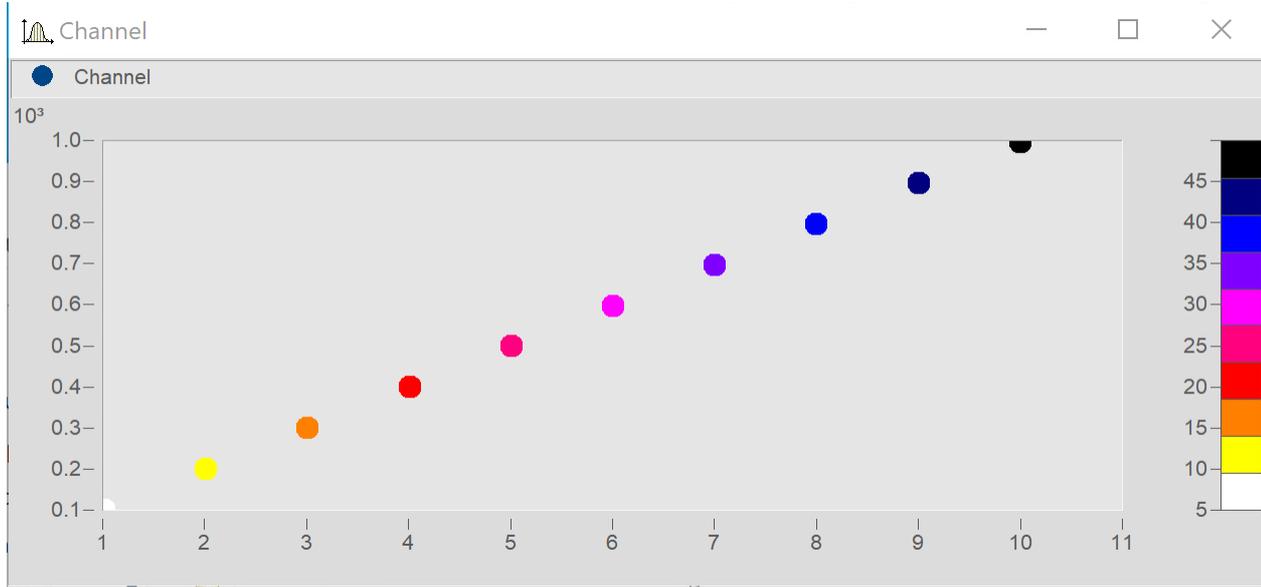


Subsequently, use the mouse to move the axis to the edge:



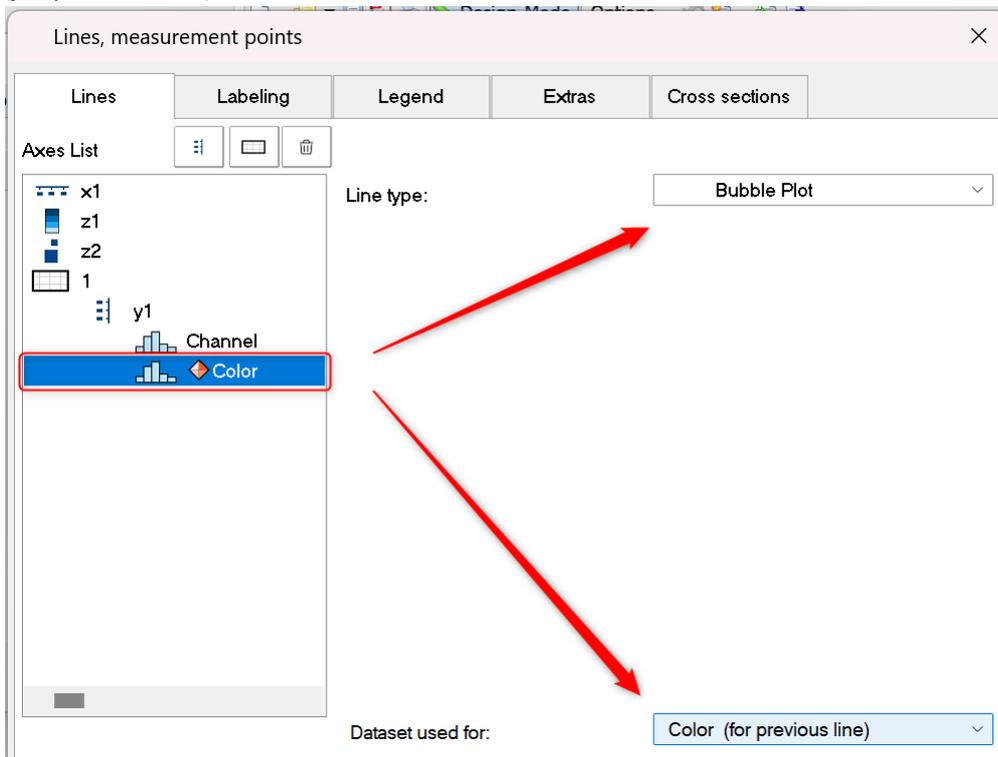
13.5.1.11.1 Additional Bubble Plot displays

Bubble Plot with color palette

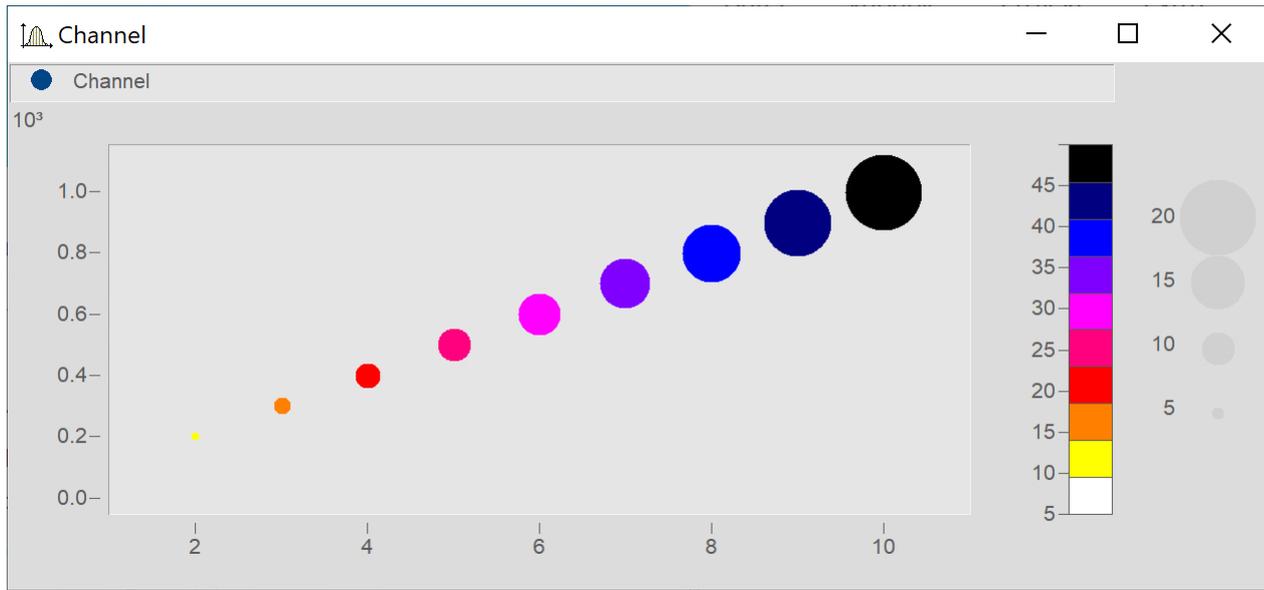


Matches "[Size of bubbles corresponding to values of another data set](#)"⁹¹⁸, except that the second data set is used for the color palette.

In the dialog **Lines**, set the color-determining channel to *Line type: Bubble Plot* and *Dataset used for to Color (for previous line)*.

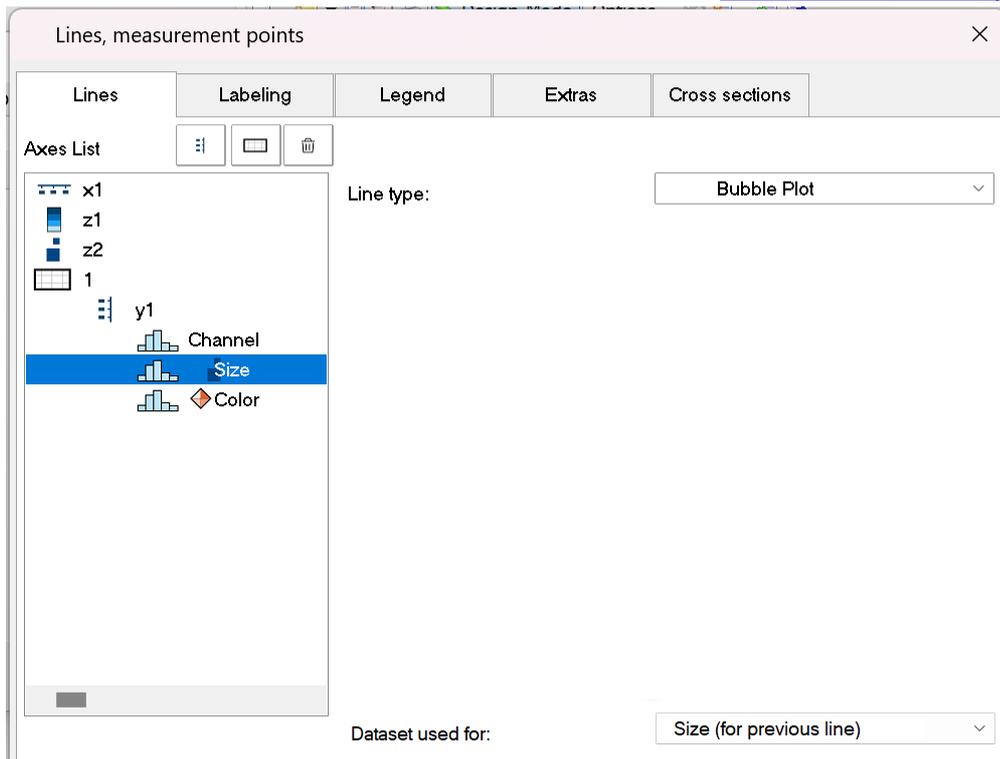


Bubble Plot with size channel and color palette



Matches "[Size of bubbles corresponding to values of another data set](#)"⁹¹⁸ 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](#)"⁹²¹.

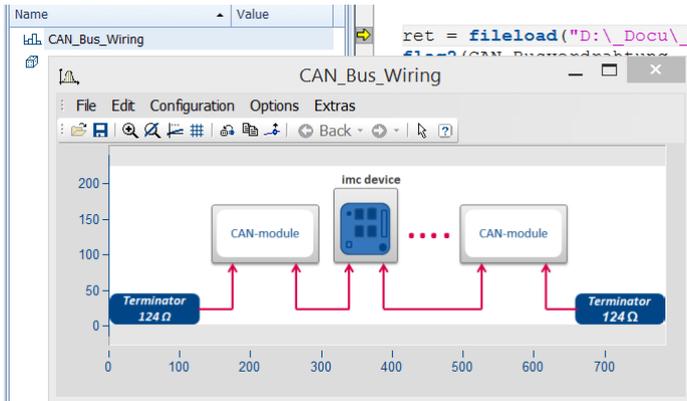


In the dialog **Lines**, set all three data sets to *Line type: Bubble Plot*.
 Make the following settings for the respective channels: *Datset used for: Color (for previous line) and Size (for previous line)*.
 Expand the edge (see [here](#)⁹²⁰) and use the mouse to arrange the scaling axes for the variables Color and Size.

13.5.1.12 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`^[962] 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*^[946].



The color stages of the pixels are displayed exactly for the `Line type setting`^[955] "Steps". For the line type "Lines", the color gradients are interpolated.

If a data set has `multiple events`^[1000] 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.

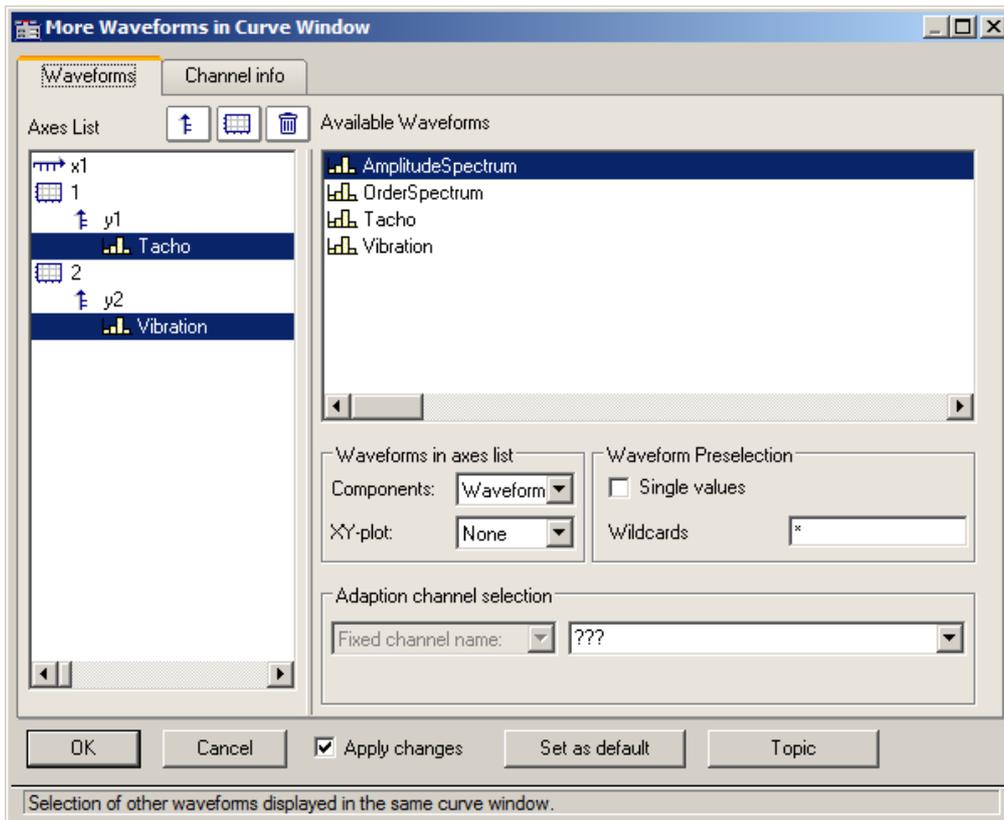
13.5.2 Showing data in curve window

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.

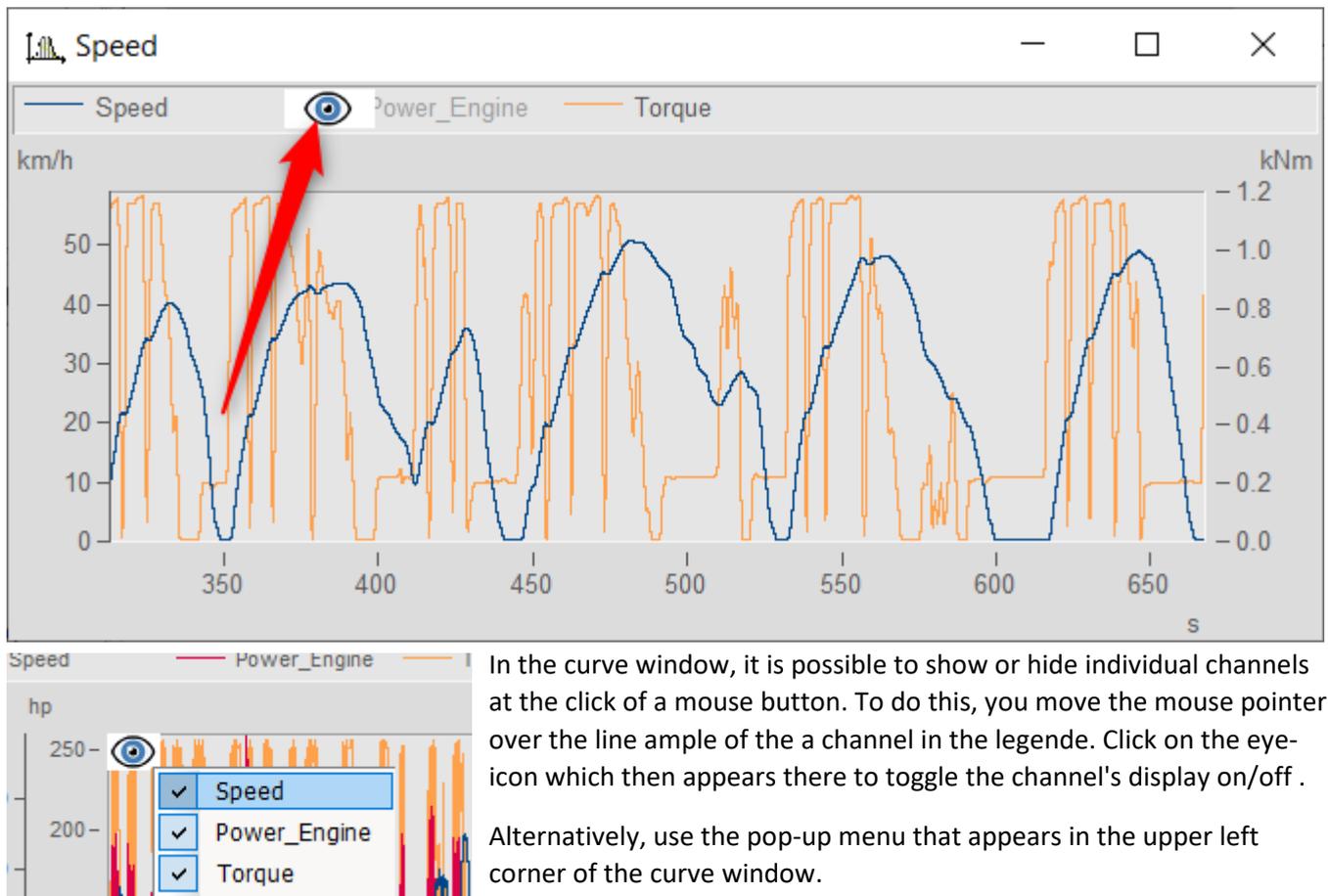
Open the dialog by right-clicking the mouse in the curve window and selecting *More Channels.....*

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

In the curve window, it is possible to show or hide individual channels at the click of a mouse button. To do this, you move the mouse pointer over the line ample of the a channel in the legende. Click on the eye-icon which then appears there to toggle the channel's display on/off .

Alternatively, use the pop-up menu that appears in the upper left corner of the curve window.

13.5.2.1 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

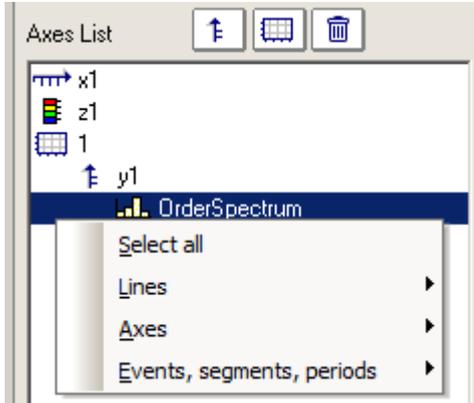
Icon	Description
	A new y-axis is created. To do this, click briefly on this button, which changes the mouse pointer to a suitcase symbol. Click on the target position desired. The new y-axis is inserted above the row over which the mouse pointer is located.
	A new coordinate system is created.
	The lines selected in the axes list (coordinate systems, axes, waveforms) are deleted. Use the button as an alternative.

Symbols used in the axes list

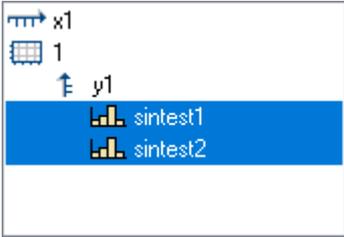
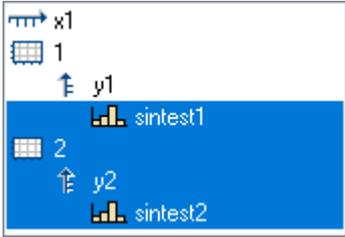
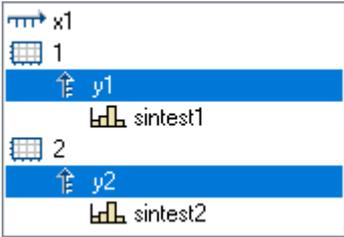
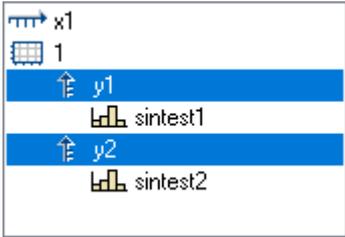
Icon	Description
	x-axis ; always at beginning of axes list. All following coordinate systems are scaled on the same x-axis
	z-axis ; only in <i>3D, color map and waterfall diagram</i> . The z-axis can be scaled only with a fixed z0 and delta-z .
	Indicates an independent coordinate system . All further 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

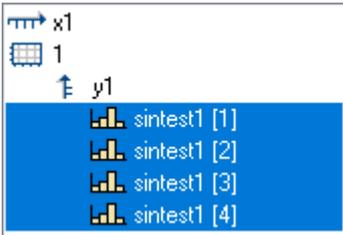
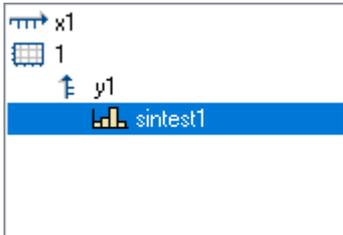
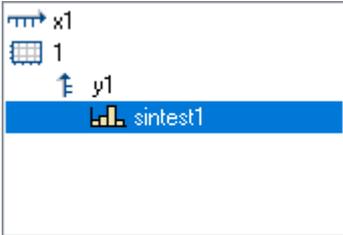
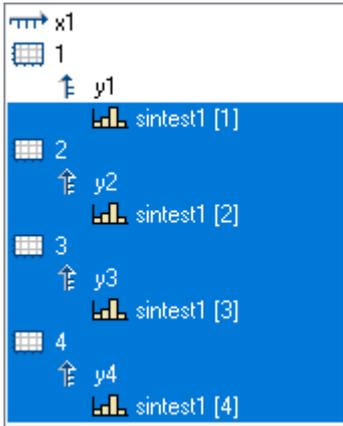
Icon	Description
	A complex waveform in magnitude/phase- or real/imaginary display

Context menu in Axes List



The following menu items are available:

Menu item	Description
Select all	Selects all lines in the list.
Lines	<p>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.</p> <p>The following items affect the selected lines.</p>
Individual coordinate systems	<p>This produces a <i>stacked</i> representation of curves.</p> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">  <p><i>before</i></p> </div> <div style="text-align: center;">  <p><i>after</i></p> </div> </div>
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	<p>All axes are incorporated into only one coordinate system. Obsolete coordinate systems are deleted.</p> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">  <p><i>before</i></p> </div> <div style="text-align: center;">  <p><i>after</i></p> </div> </div>

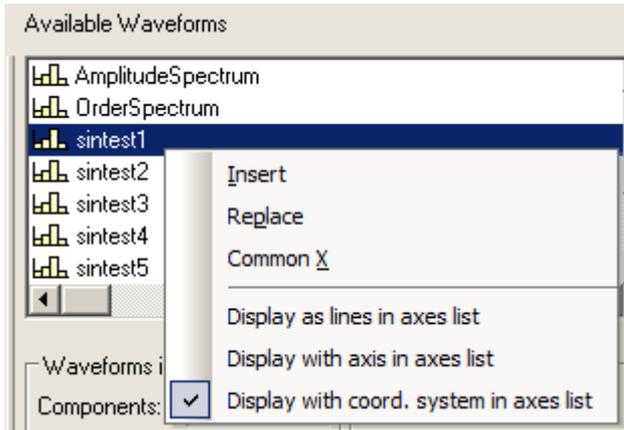
Menu item	Description
Individual coordinate systems	A separate coordinate system is displayed for every (y-) axis.
Events, Segments, Periods	The following items are relevant only to special data types (segmented and multi-shot waveforms)
Combine	<p>Various lines are attached to each other, if the numbering permits. The range of possibilities is dictated by the exigencies of the corresponding dialogs for selecting structural elements.</p> <div style="display: flex; justify-content: space-around; align-items: flex-end;"> <div style="text-align: center;">  <p><i>before</i></p> </div> <div style="text-align: center;">  <p><i>after</i></p> </div> </div>
Separate with same layout	<p>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.</p> <div style="display: flex; justify-content: space-around; align-items: flex-end;"> <div style="text-align: center;">  <p><i>before</i></p> </div> <div style="text-align: center;">  <p><i>after</i></p> </div> </div>
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.

13.5.2.2 Available channels

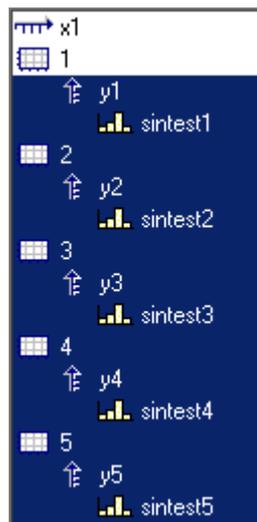
This list contains the waveforms which can be displayed in the curve window. Waveforms selected from this list can be fetched into the axes list using drag&drop (starting from the left edge of the list, the mouse pointer has the shape of a hand).

Context menu in available channels

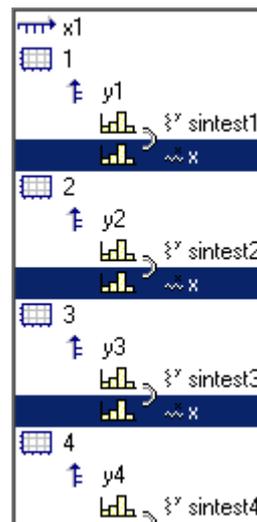
Right-clicking mouse in the *Available Channels* list to open the context menu:



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: <ul style="list-style-type: none"> • 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 replaced 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.



before



after

13.5.2.3 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 ";"

13.5.2.4 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.

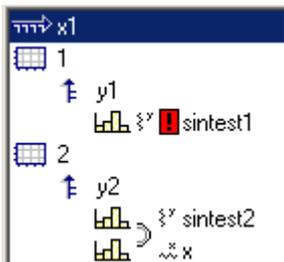
Component	Description
???	The settings for the selected waveforms do not match.
Waveform	Complete display of variables according to type. XY-Waveform: the entire waveform "Y over X" Complex data: polar plot
.x .y	The specified components of XY-waveforms are displayed
.r .i	Real or imaginary part of a 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 its 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.

13.5.2.5 Adapting channel selection

The dialog portion *Adapting channel selection* is needed in conjunction with the [Data Browser](#)⁸⁰⁴.



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.

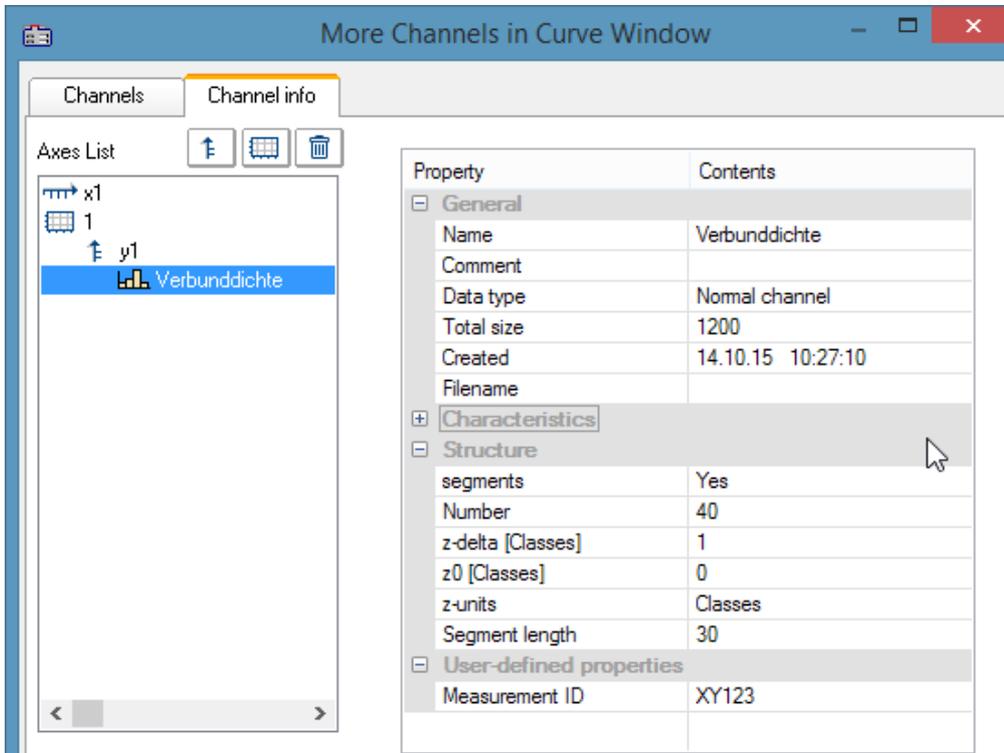
Selected channel

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*.

13.5.2.6 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 under [Configuration / Arrangement / Channel properties](#)¹¹⁰⁰⁷¹ or via the corresponding symbol in the *Navigation Toolbar*.

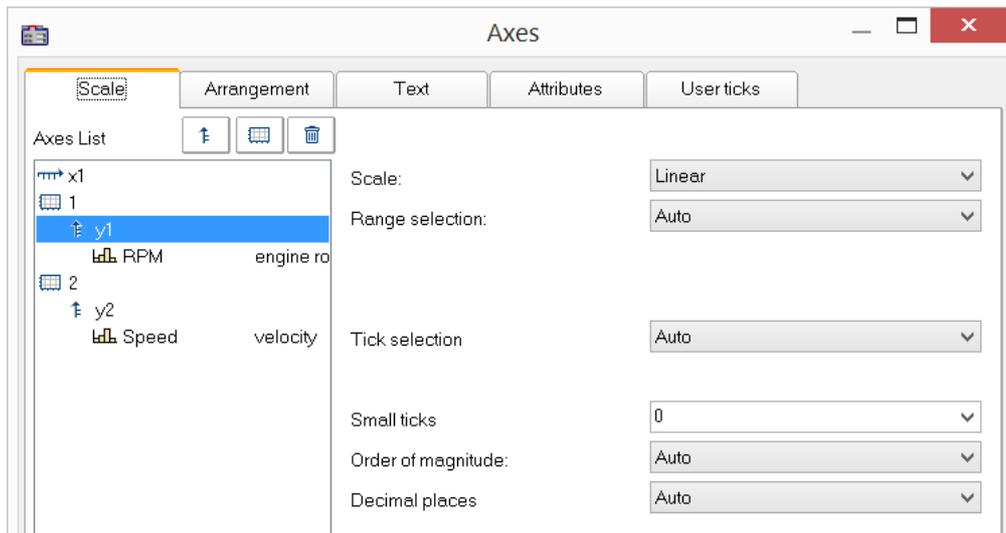


13.5.3 Axis configuration

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.

13.5.3.1 Scale

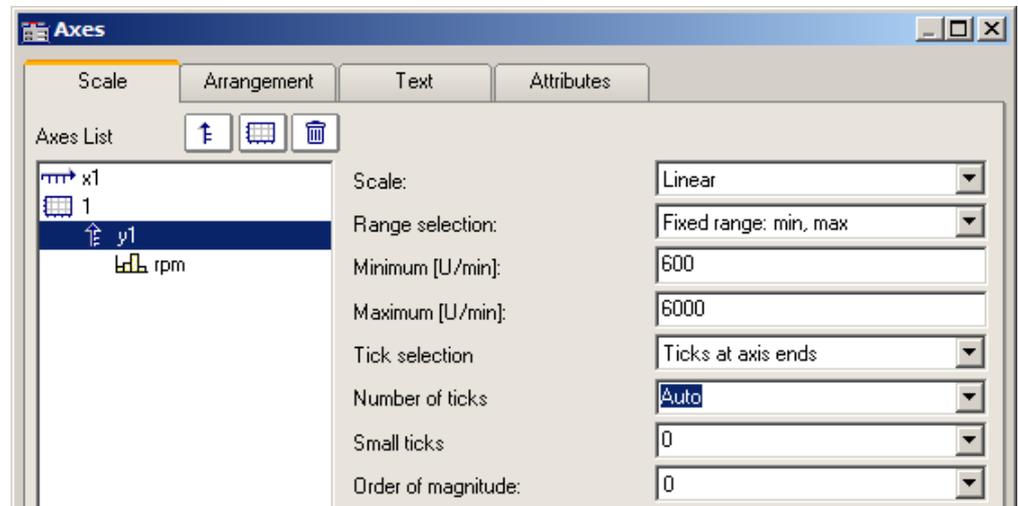
Scaling of the axes. An axis' domain can be defined in a number of ways.

Scale

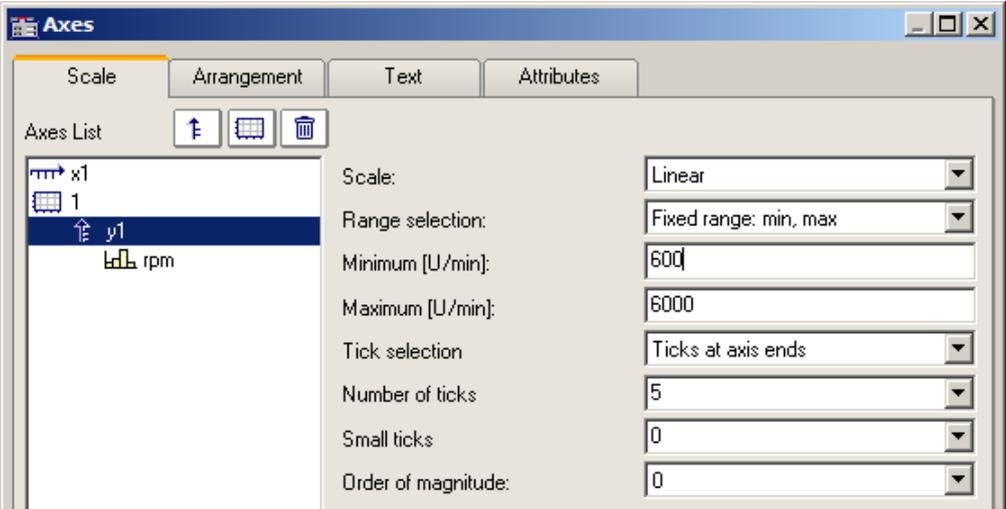
The axis can be scaled as *linear*, *logarithmic* or in *dB*. In Date/time display, the scaling is always linear. When the mode *Rounding* is selected for logarithmic labeling, powers-of-ten will be written to the axis as long as the range displayed is large enough.

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.

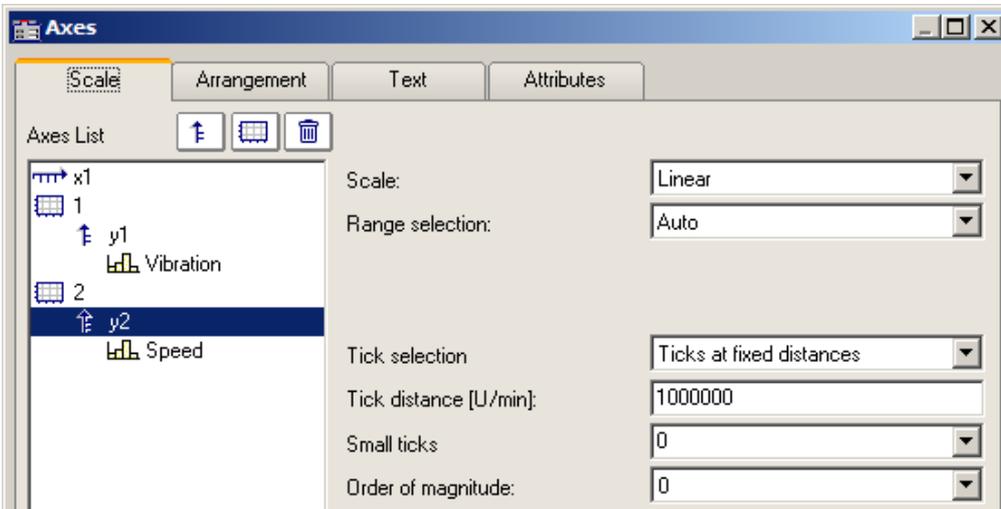


This menu option is not enabled when the option *Tick selection* is set to *Ticks at fixed distances*.

Range selection	Description
<p>Fixed range: min, max</p>	<p>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.</p>  <p>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.</p>
<p>Auto</p>	<p>Determining the range is automatic. The entire curve is displayed in the curve window.</p>
<p>Automatic with zero</p>	<p>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.</p>
<p>Like previous axis</p>	<p>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.</p>

Tick selection

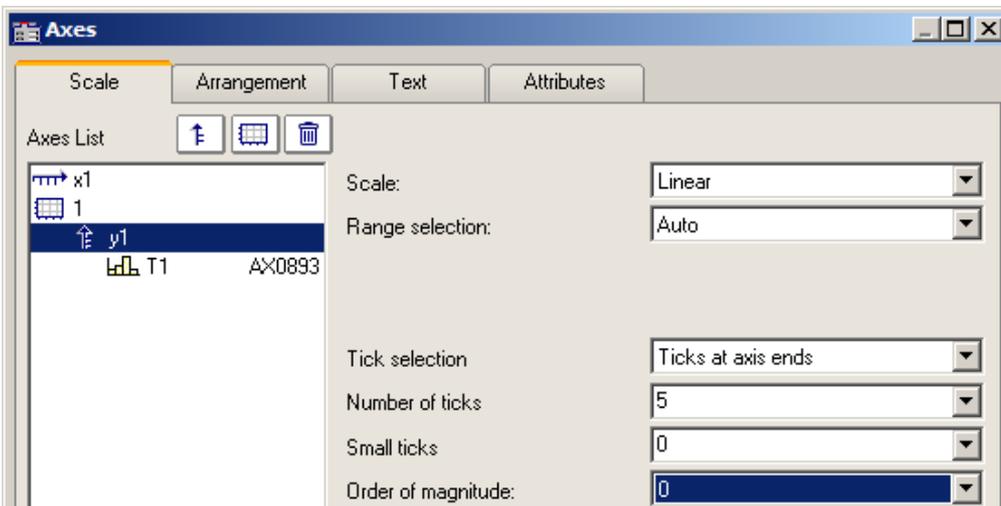
The main ticks, along with which the axes labeling appears in the curve window, can be placed anywhere on the axis. If you select the option *Ticks at fixed distances* in the *Tick selection* control, the additional control *Tick distance* becomes available. Here you have the ability to set the amount 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.



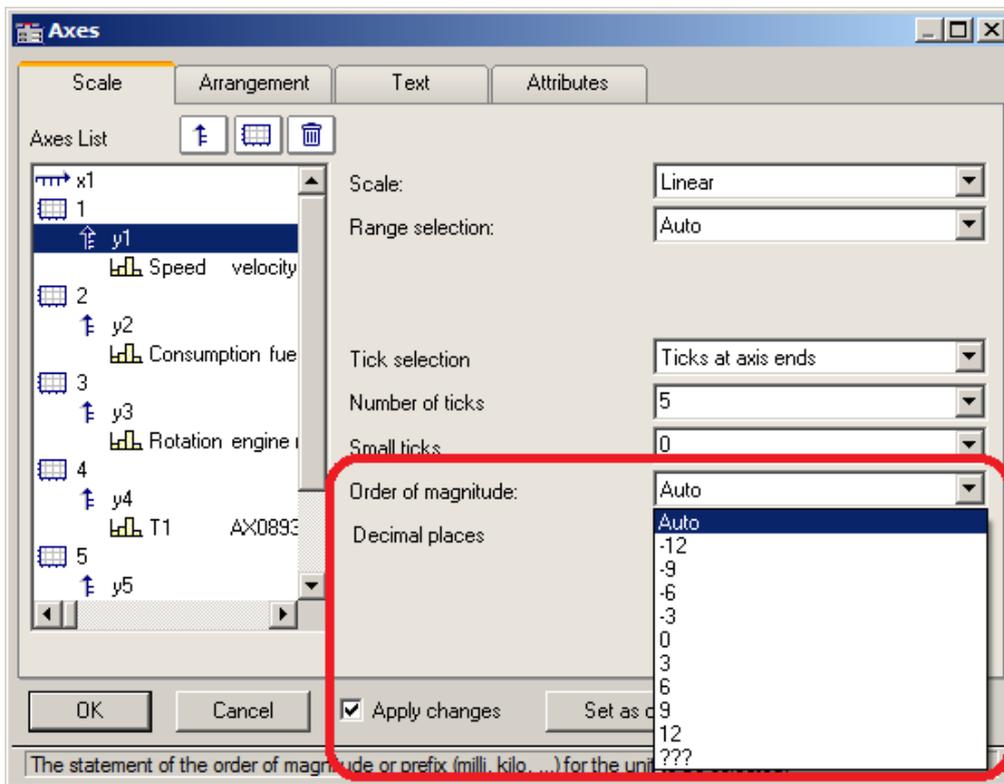
The number of ticks per unit is calculated by subtracting one (1) from the number of markings and dividing the difference by the displayed interval in units. For example, with an interval of 3ms and seven markings specified, two ticks per unit (ms) will be placed on the axis.

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).



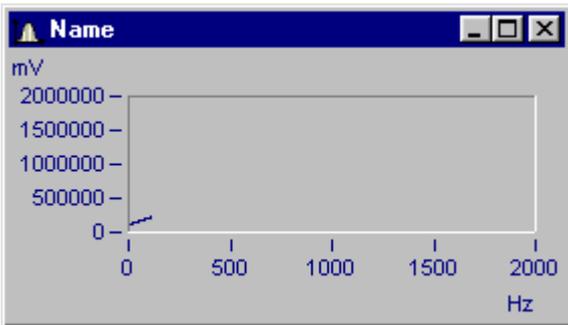


Example

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



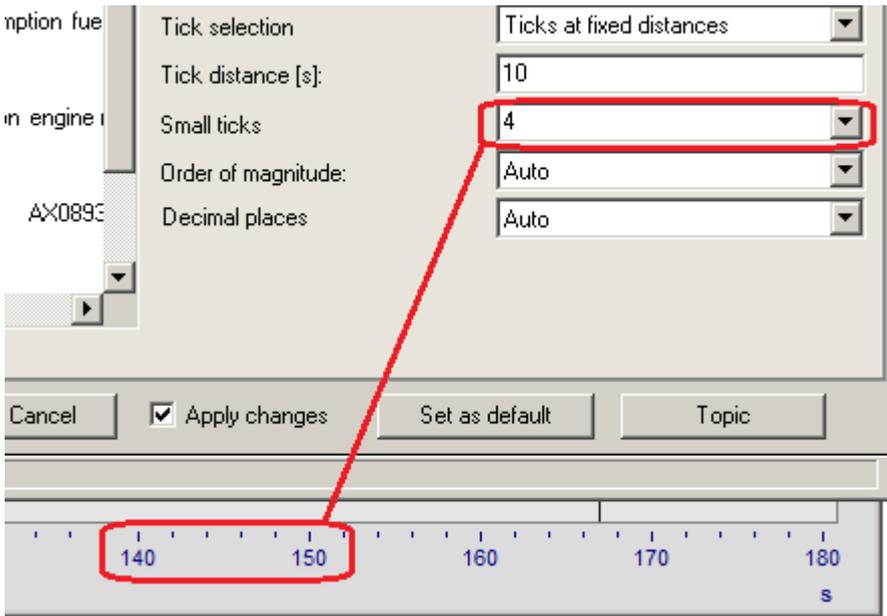
Curve window with inappropriate scaling

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](#)⁹⁸⁹.

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.sssss 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.

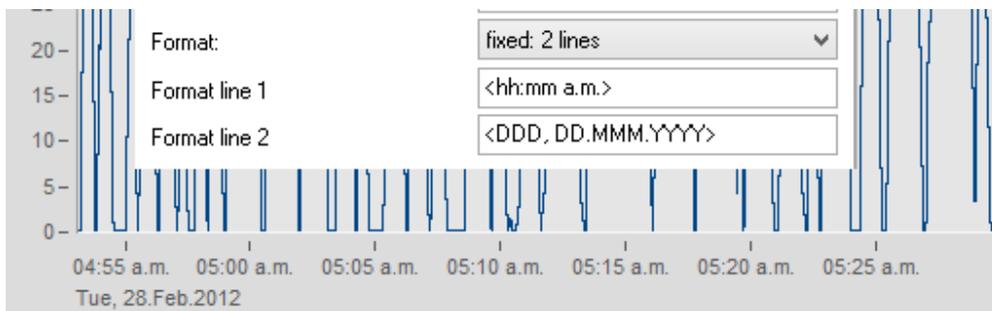
**Example**

<hh:mm:ss.ss>

<D> Tage

<o:mm:ss>

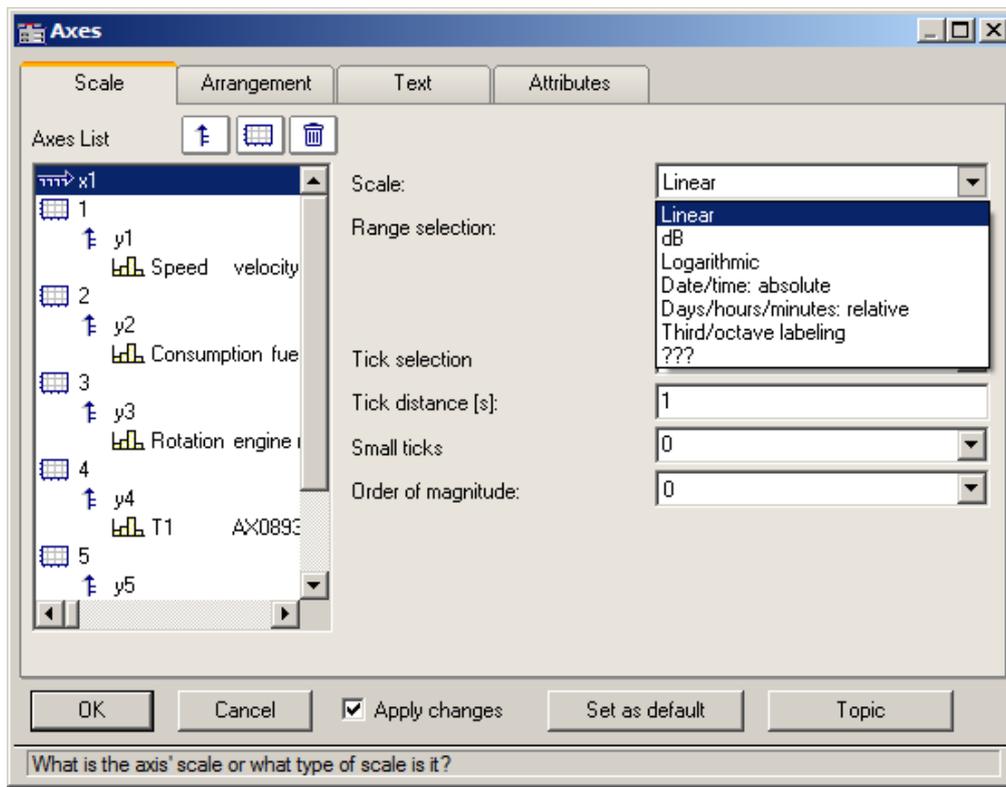
<o> Std, <mm:ss>



Example: line 1: <hh:mm a.m.> line 2: <DDD, DD.MMM.YYYY>

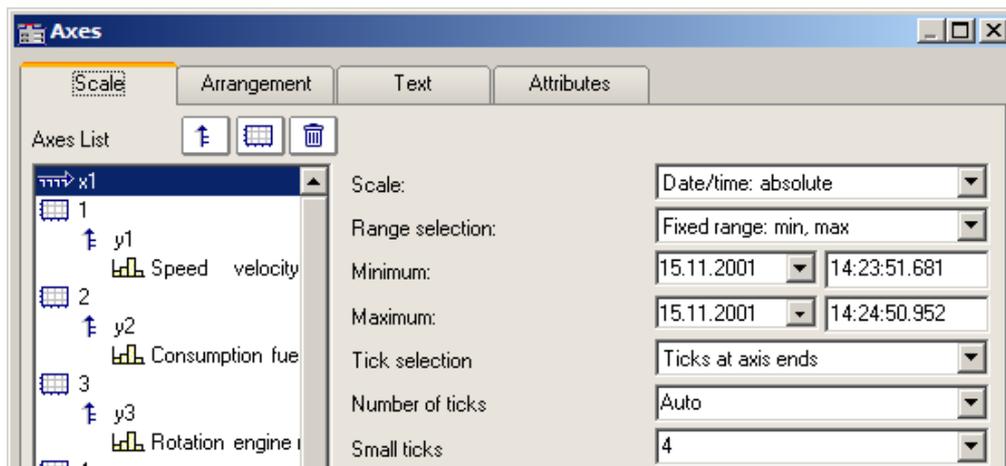
13.5.3.1.1 Particularities of the x-axes

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.

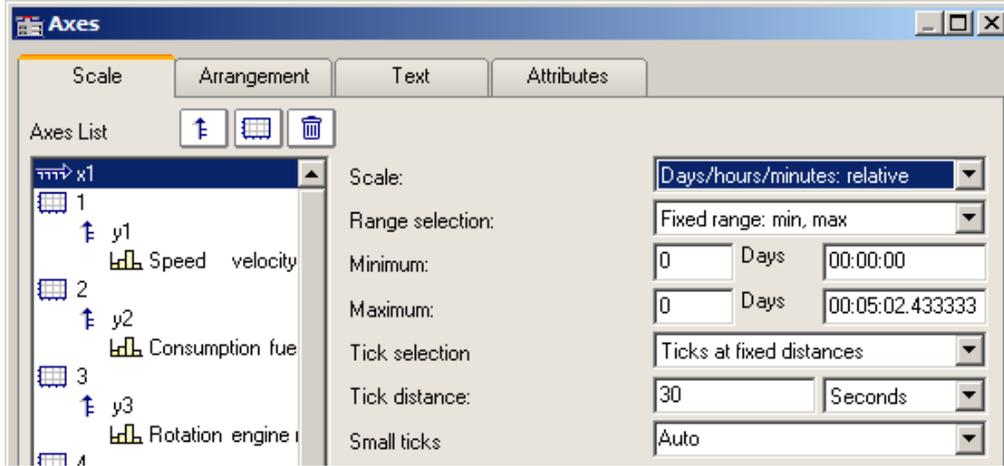


The distance between ticks can be set to be determined automatically or to be fixed. With a fixed setting, the unit can be anything from seconds to days.

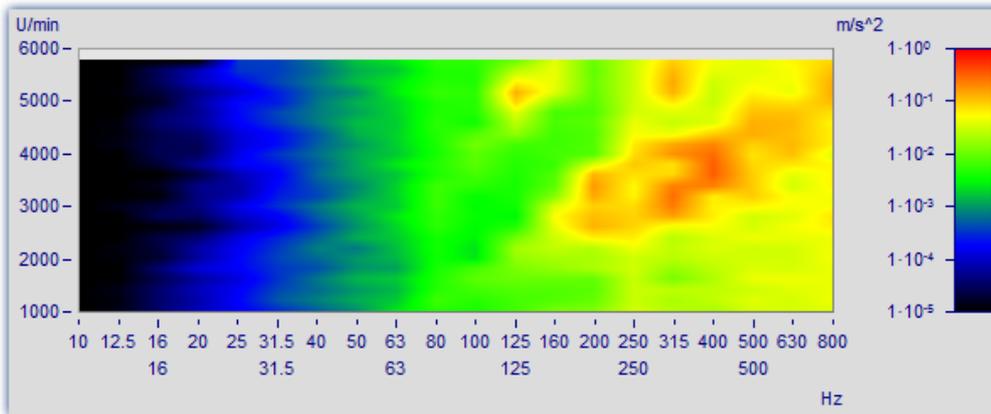
The channels are displayed in reference to absolute time. The absolute time of the measured points is usually derived from the sum of the specified absolute trigger time and the relative time for a point measured after the measurement's start.

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.



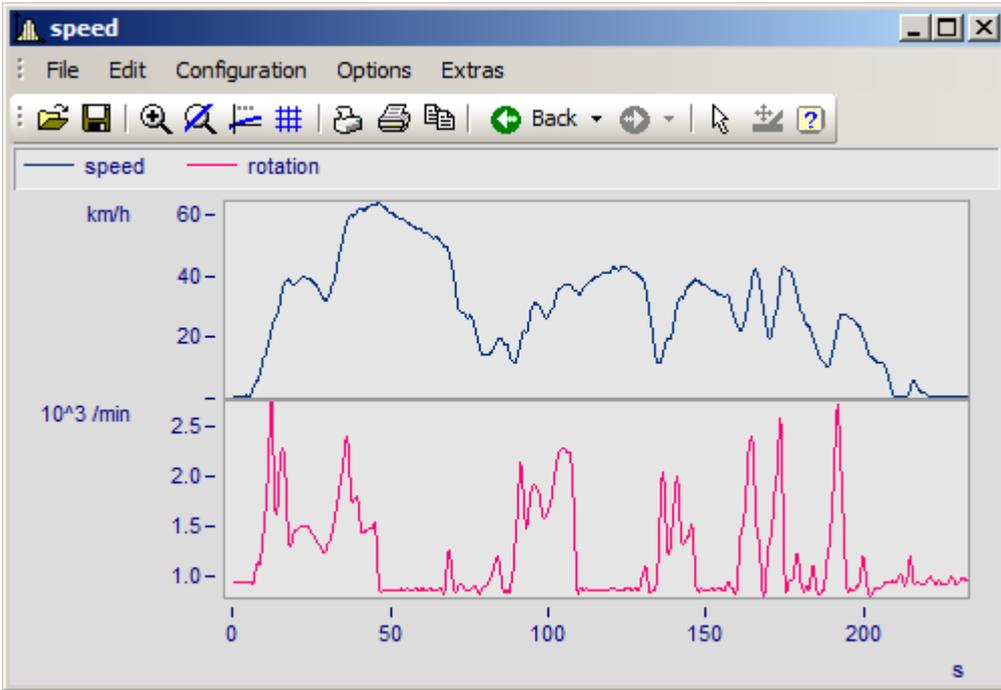
Third/ octave labeling: If a waveform contains a 1/3-octave or octave spectrum and the waveform's x-axis is scaled in 1/3-octaves, then the x-axis can be drawn with numerical values of the 1/3-octaves and octaves, in accordance with DIN.



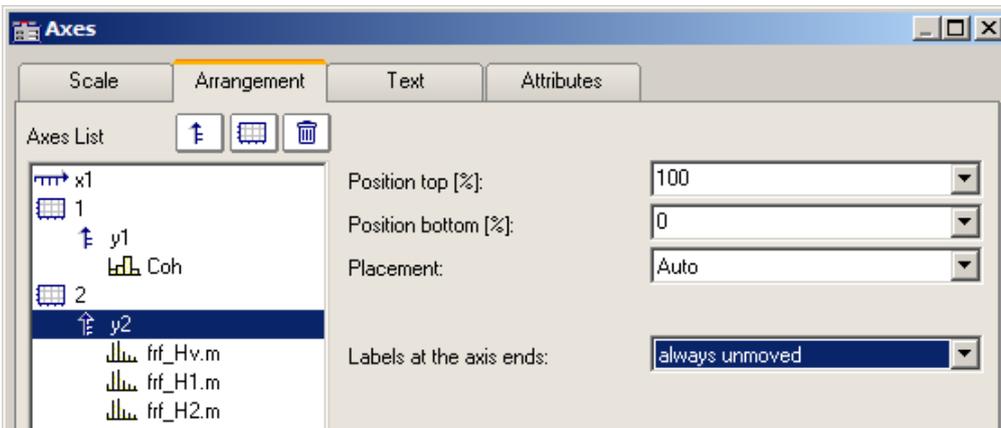
For details, see the dedicated section on "[Third / octave labeling](#)".

13.5.3.2 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. <div style="border: 1px solid gray; padding: 5px; width: fit-content;"> Placement: Left Labels at the axis ends: Left Auto Right ??? </div>

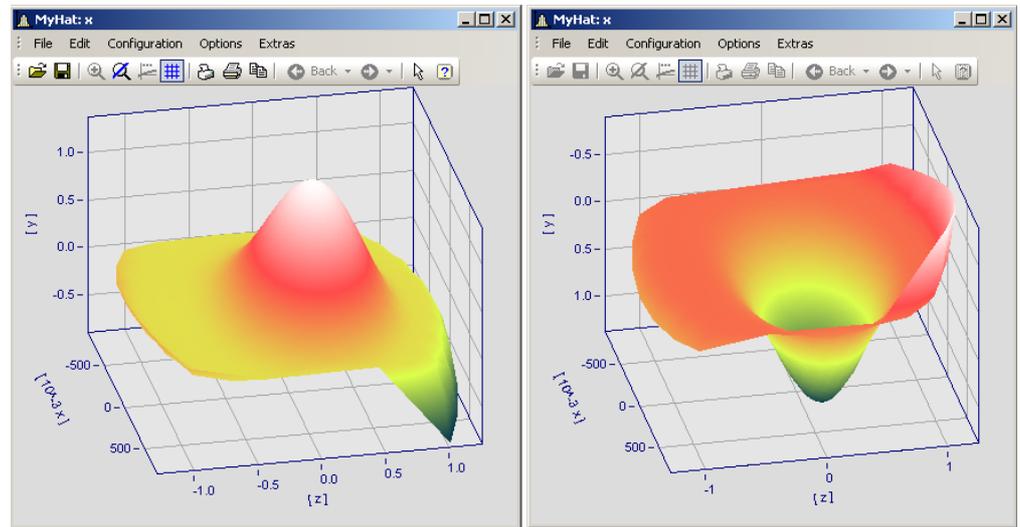
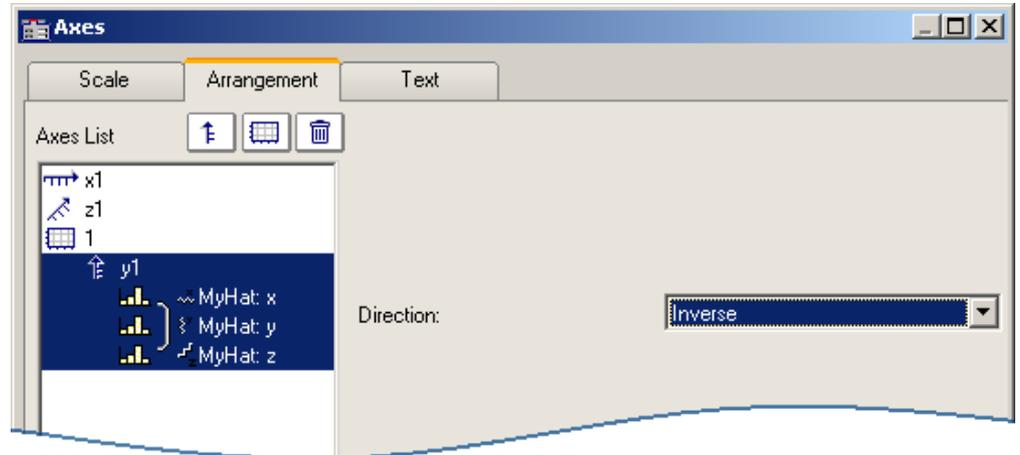
Settings	Description
Labels at the axis ends:	<p>Alongside automatic orientation of the labeling, this control also offers <i>always unmoved</i> and <i>move if necessary</i>.</p> 
Axis width	<p>The width of the Y- and X-axis can be specified manually. In particular when using "User Ticks⁹⁴⁹", this helps in providing sufficient room for the labeling.</p>
Resolution	<p>Setting the scaling of the y-axis</p> <p><i>auto</i>: individually</p> <p><i>same resolution as x-axis</i>: Generates a constant aspect ratio. In consequence, the ensures among other things that for an RGB-image⁹²³ the image is displayed without distortion.</p> <p>Resolution <input type="text" value="same resolution as x-axis"/></p> <p>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.</p>

Settings	Description
----------	-------------

Inverting the axis direction

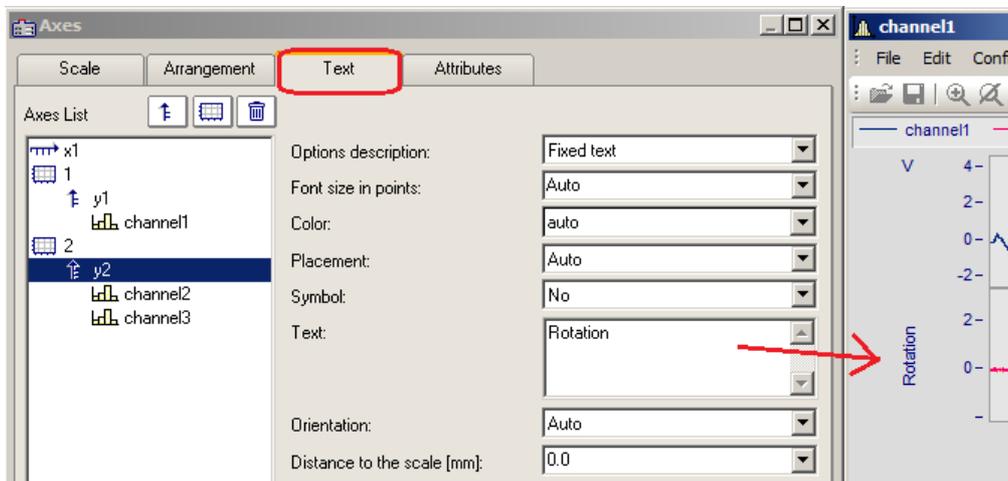
With 3D display, you can invert the direction of the Y-axis.

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

13.5.3.3 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: <ul style="list-style-type: none"> • <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 not allowed. • <g*b> for Greek letters. Also in the exponent: $A\langle e \rangle - \langle g * a \rangle t \langle /e \rangle \Rightarrow A^{at}$. • 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.

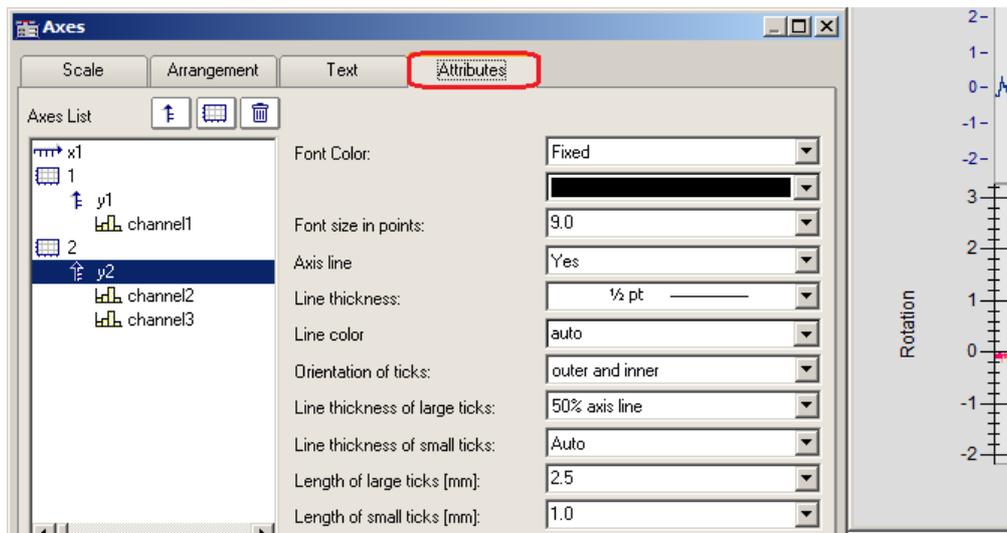


Reference

Further texts

Further texts can be added on the tap [Display/Text](#) and in chapter [User ticks](#).

13.5.3.4 Attributes



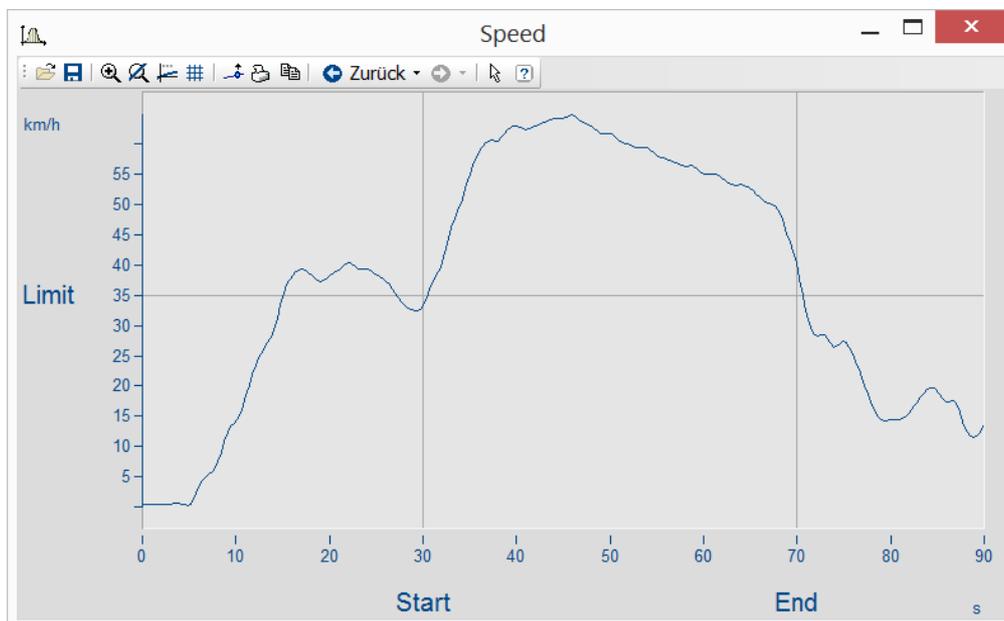
Settings for the axis

The control **Font Color** allows the color of the first data set to be applied, or a freely selected color to be set.

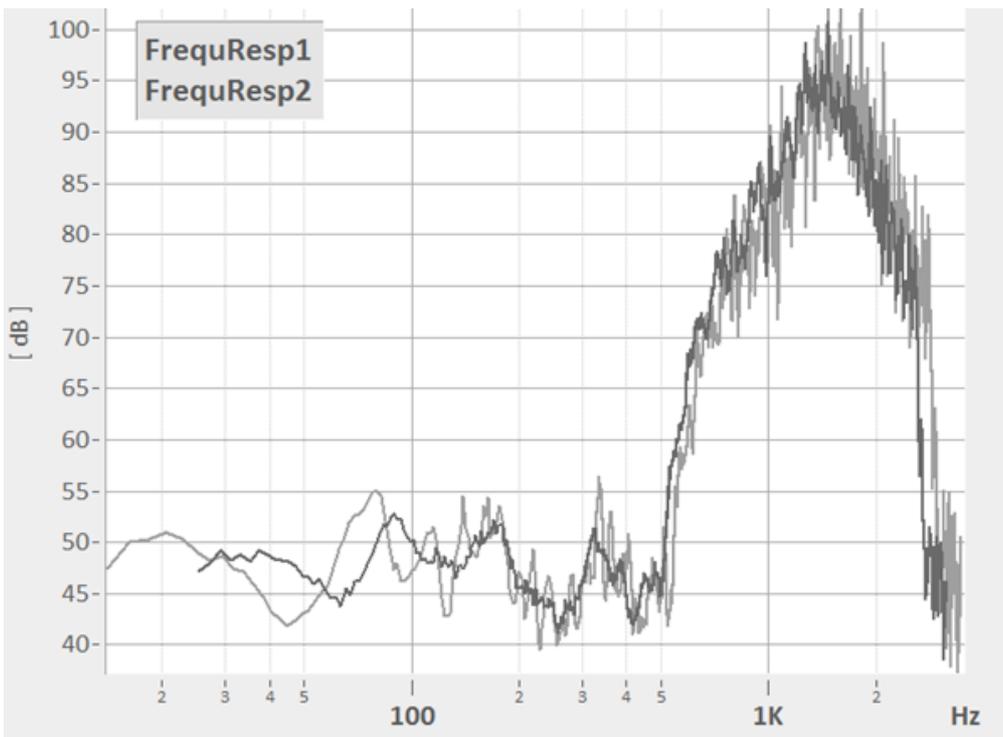
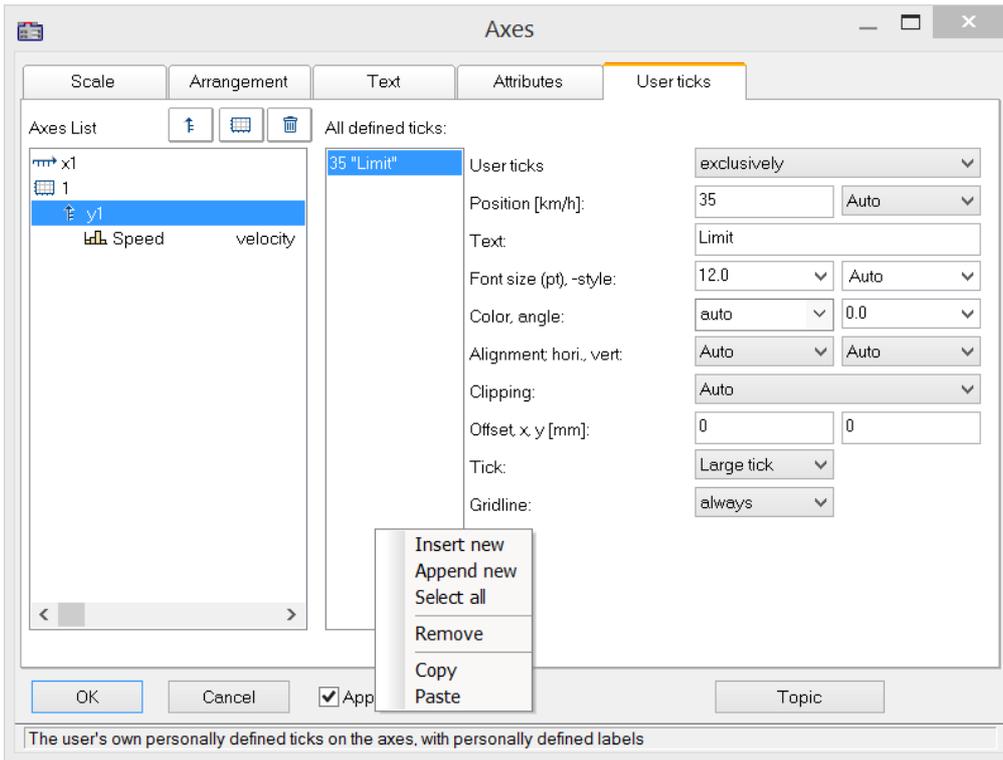
Additionally, it is possible to add an **Axis line** while specifying its **Line thickness** and individual **Line color**.

The **Line thickness** and **length** of the **ticks** can be set separately for both the small ticks (between numbers) and large ticks.

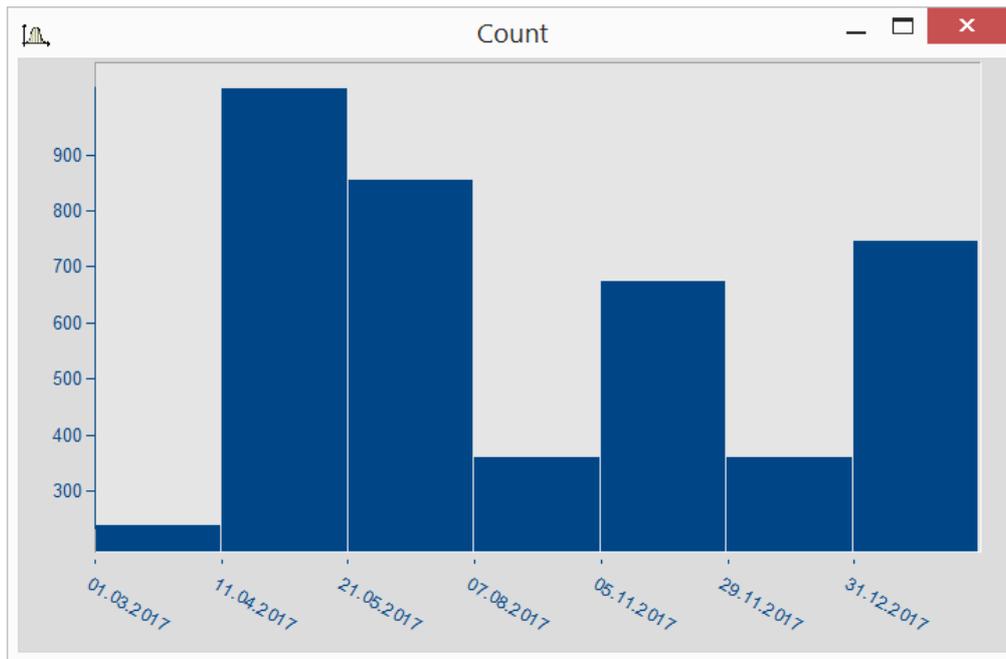
13.5.3.5 User ticks



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.



Any formatting, e.g. for log. Display



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(<i>exclusively</i>) or complements(<i>additionally</i>) 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 " ^[946] .
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 " ^[949] .
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](#)"^[987] and on the tab [Axes/Text](#)"^[947].

13.5.3.5.1 Label for raw data

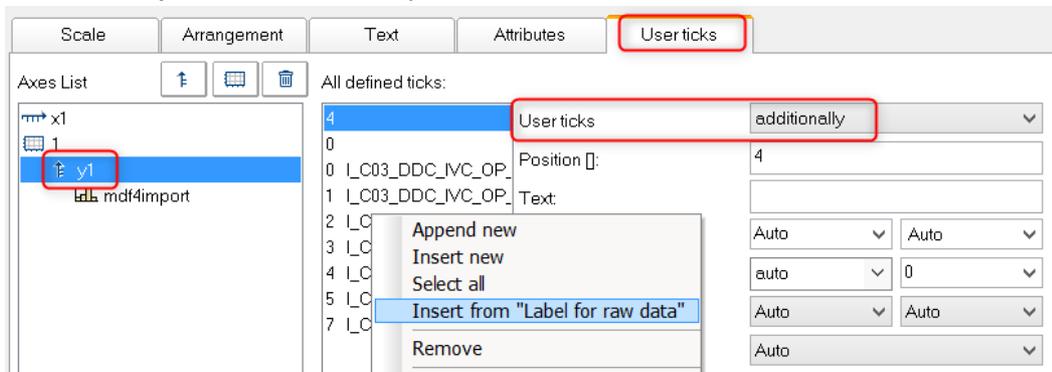
Some signals contain integers representing certain states or error codes. For instance, such a number can denote the gear in an automatic transmission (0=N, 1=D, 2=R, 3=P, etc.). Such information is typically recorded as an integer value.

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.

13.5.3.6 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.



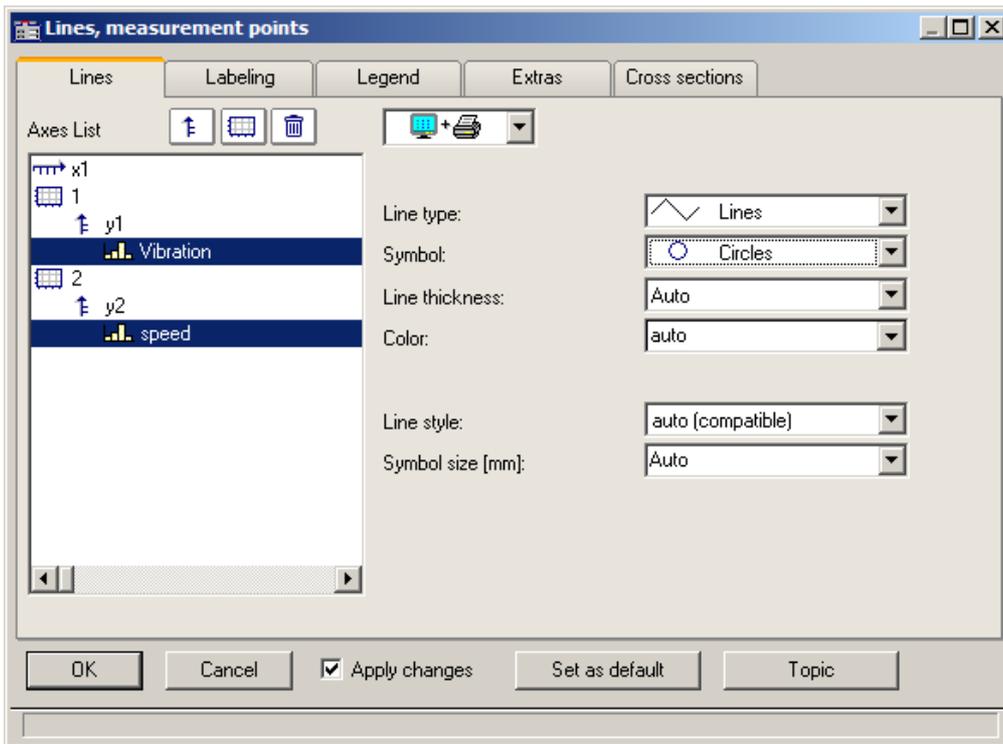
13.5.4 Line configuration

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*.

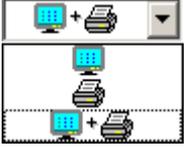
The dialog for setting line properties then appears:



13.5.4.1 Lines

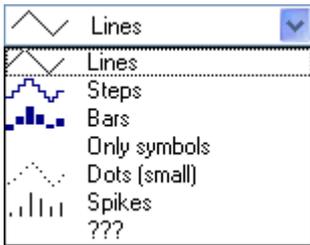
Appearance of lines in the screen display and in printouts

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.

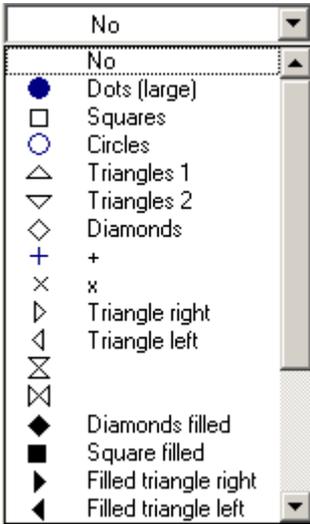
When displaying an [RGB-image](#)⁹²³, there is a distinction between *Lines* and *Steps*. For *Lines*, the pixels are interpolated between the samples (original image points) with color gradients. With *Steps*, the interpolated pixels remain constant. With any RGB-image, all line types other than *Lines* 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.



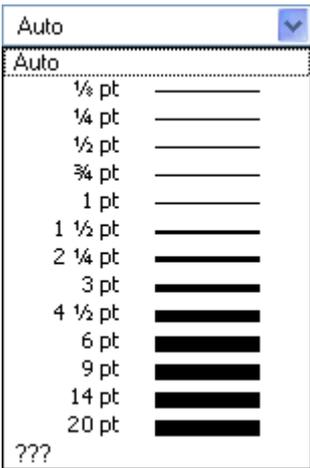
The symbol type desired is selected from this pop-down list. Symbols can be displayed either together with a line or all alone.

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](#)^[852], 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](#)"^[964] tab.

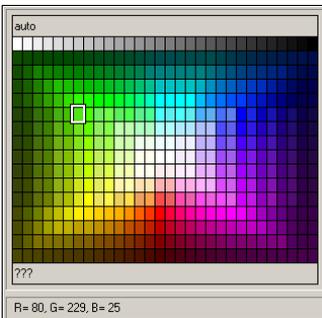
Line thickness



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](#)"^[1078].

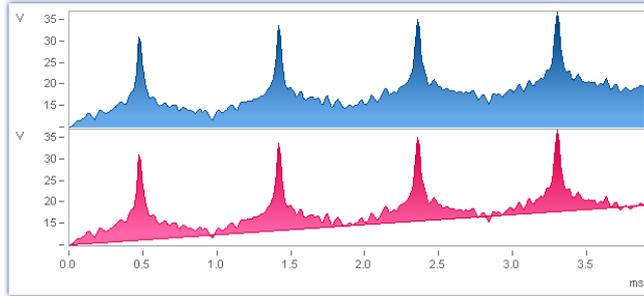
It is also possible to select a fixed color from the palette provided, which cancels all automatic settings.

The color can be different for the printout and for the on-screen view.

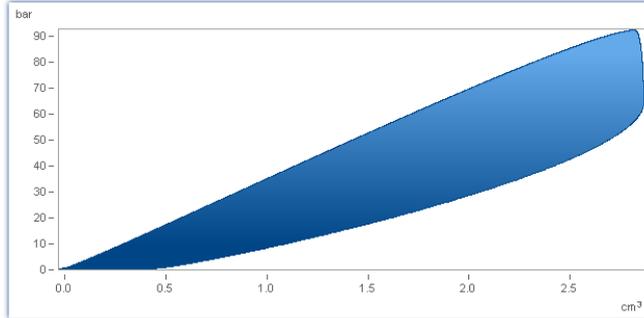
Fill area

Fills the area below a line.

Fill area: to y=zero
 Area color gradient: From top to bottom
 Color gradient to: auto

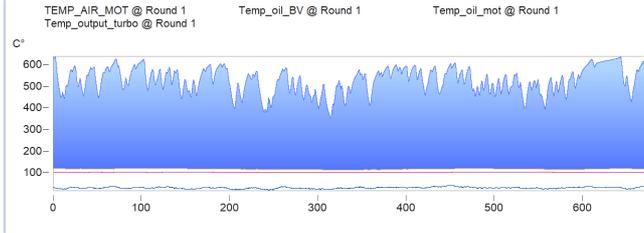


Fill area: inward
 Area color gradient: From bottom to top
 Color gradient to: auto



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
 Area color gradient: No
 Color gradient to: inward

no: as previously

to y=zero: Fills area below the graph up to the zero line.

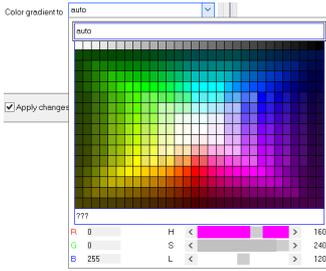
to bottom: Fills area to the bottom edge of coordinate system.

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.

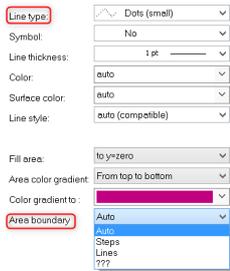
Area color gradient: From top to bottom
 Color gradient to: From bottom to top

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.



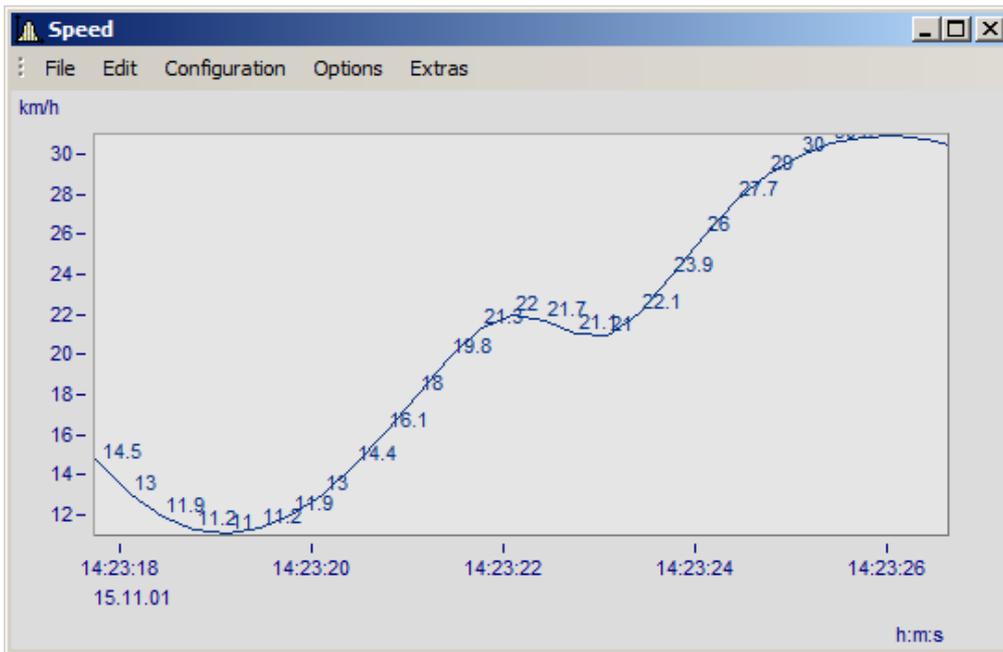
If **Line type = Dots** or **only symbols**, there is the additional option **Area boundary**: This determines whether the edge of the area is interpolated linearly or appears in discrete steps.

If multiple curves are displayed with overlap, the order for each curve is:

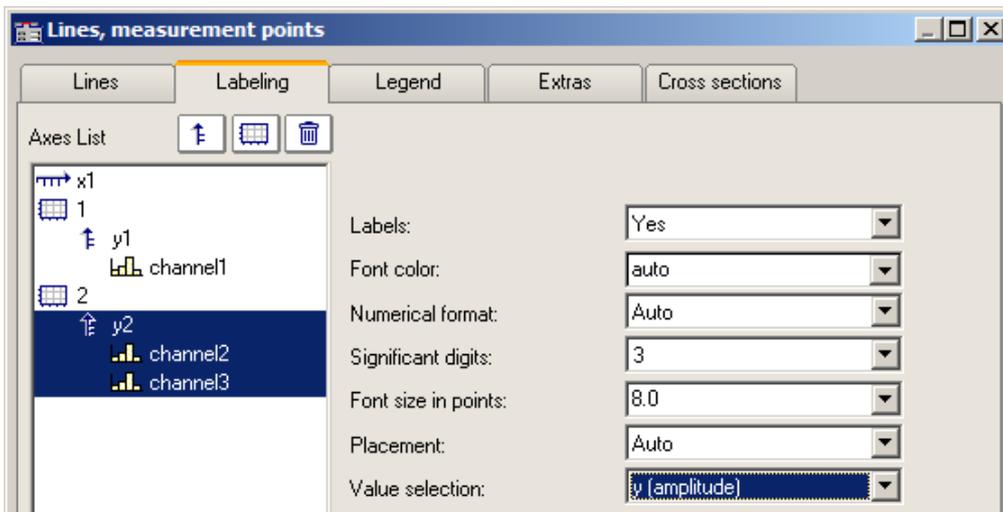
1. first the area
2. then measurement uncertainty
3. then lines

It is possible to generate a **pure** area without any borderline by setting **Line type = Only symbols** and **Symbol = empty**.

13.5.4.2 Labeling

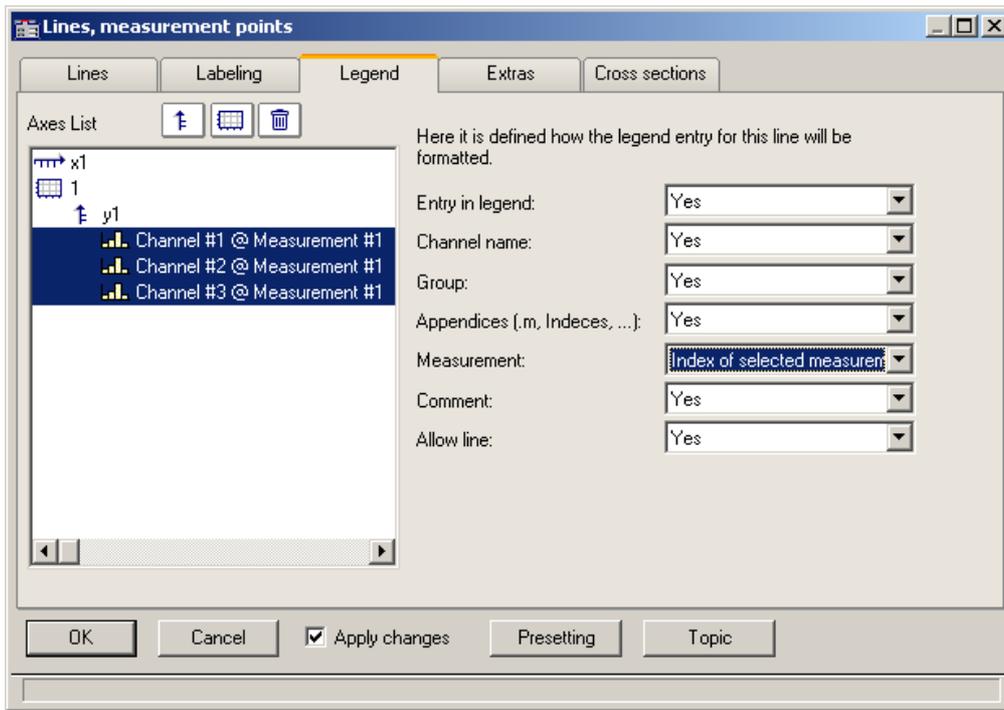


Here you can set whether or not the measurement points are labeled with text stating their numerical values. For the **labeling** of the numerical values, the **color**, **format**, **number of valid digits**, **font size** and **position** can be set.



Using **Value selection**, you can set which values are displayed: y, x, parameter, magnitude and phase

13.5.4.3 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

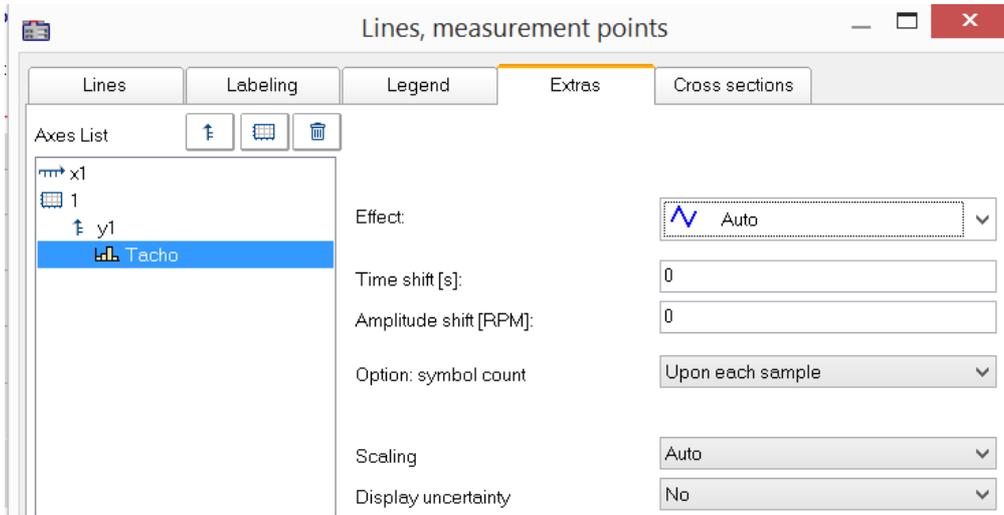
If you display only one data set in the curve window, the legend is only displayed if you have set the control for [Display\Legends\Display Legends](#)⁹⁶⁷ to *always*.

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](#)⁹⁶⁷.

13.5.4.4 Extras

On the page *Lines\Extras*, special display options are available:

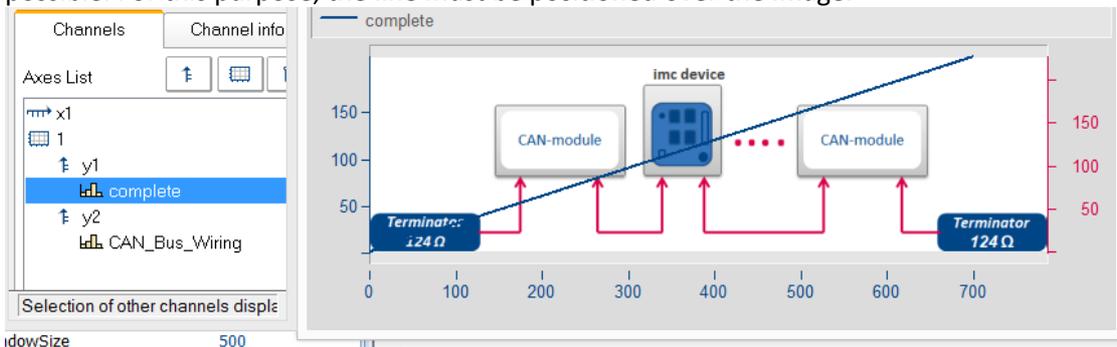


- [Time-/Amplitude-Shift](#)⁹⁶³: Shifts the line in the curve window without changing the values of the variable.
- [RGB-image](#)⁹⁶²: Display of an image made of segmented data.
- Special [color map display](#)⁹⁶³
- Reduction of the [symbol display](#)⁹⁶⁴
- [Measurement uncertainty](#)⁹⁶⁵

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.

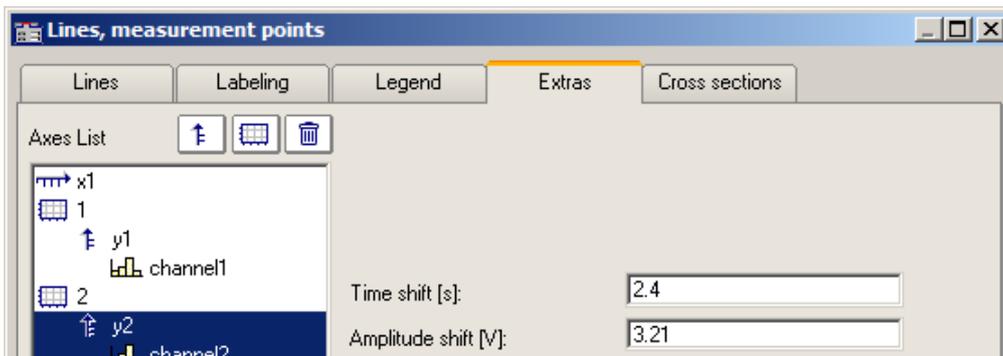
The property applies to each line separately, which means that superpositioning of an image with a curve is possible. For this purpose, the line must be positioned over the image.



References

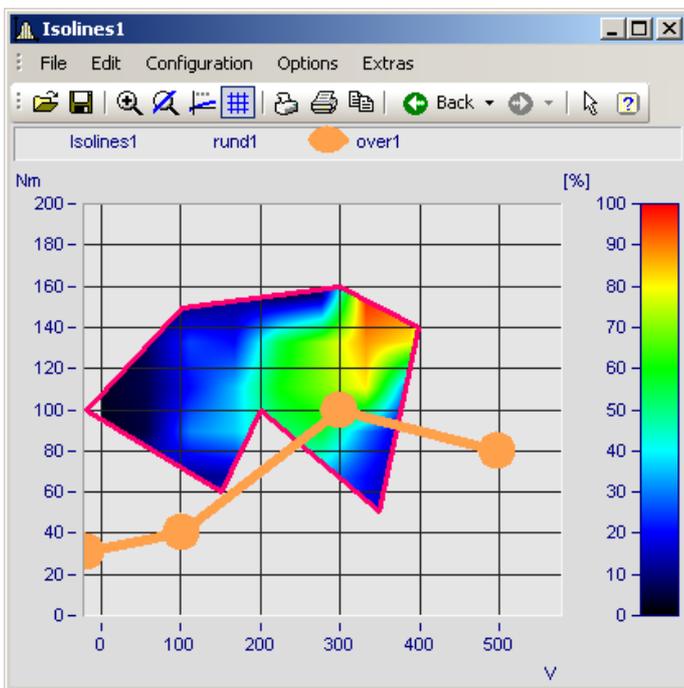
- General info on RGB-images is presented in the chapter [Display variants of the curve window](#)⁹²³
- Info on aspect ratio and resolution is in the chapter [Axes configuration/Arrangement](#)⁹⁴⁶

In display mode *Default*, you can manually specify the parameters of the Line-Shift function here. For more info, see the section [Line-Shift](#)¹⁰⁶¹.



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.



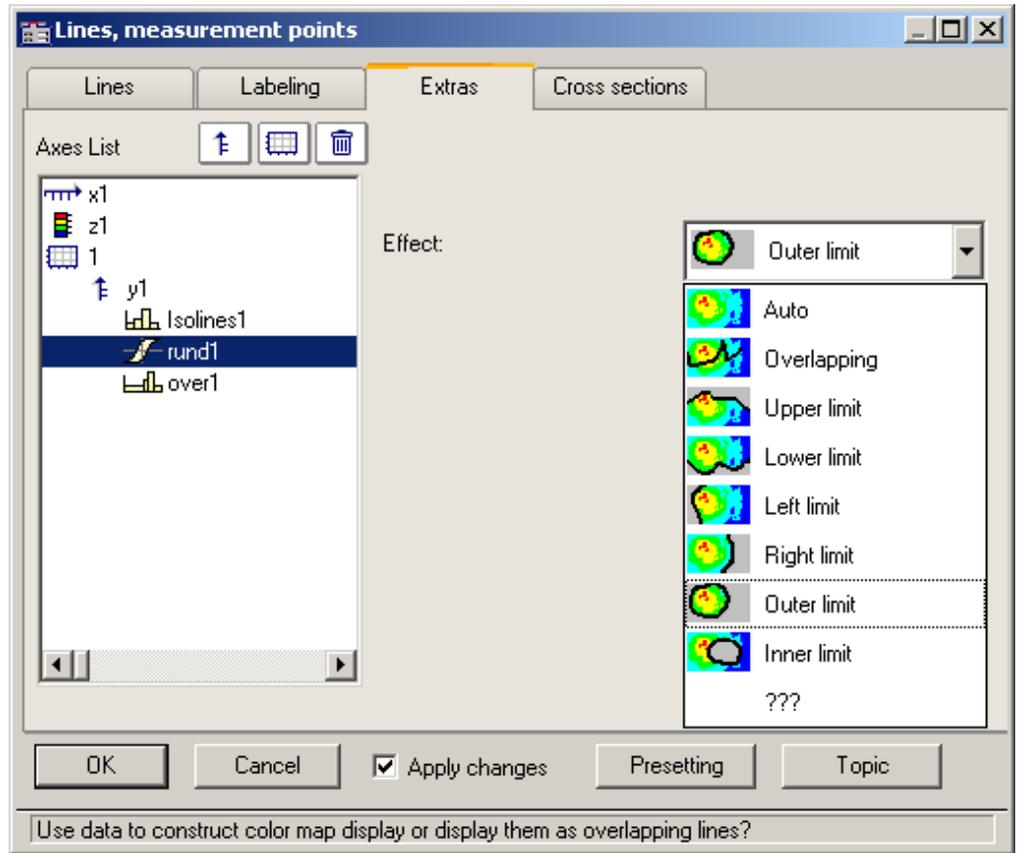
In this picture you see a function for constructing the color map, and a function for superimposing a line over the color map.

Settings	Description
----------	-------------

Effect

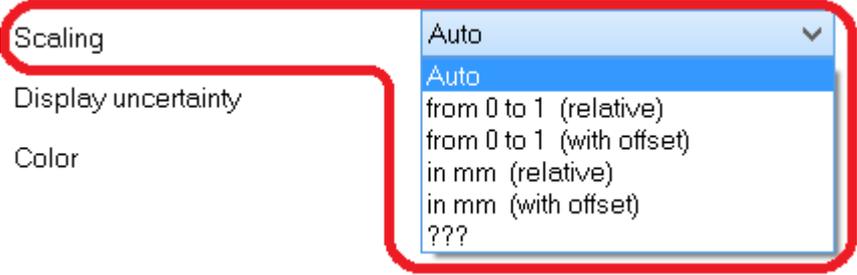
The available *Effects* are 1) *Overlapping* for superimposing a line over the color map, and 2) various margin functions affecting the construction of the color map.

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

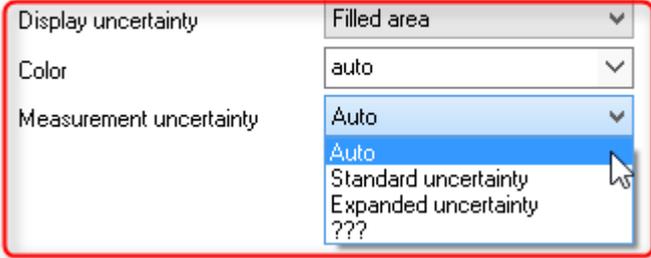
The number of symbols can be defined with "*Option: symbol count*" and "*Number of symbols*". Symbols are activated on the [line tab](#) ⁹⁵⁶.

Settings	Description
Scaling	

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 range 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	

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.

13.5.4.5 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](#)¹⁰⁶⁰.

13.5.5 Additional display options

13.5.5.1 Display

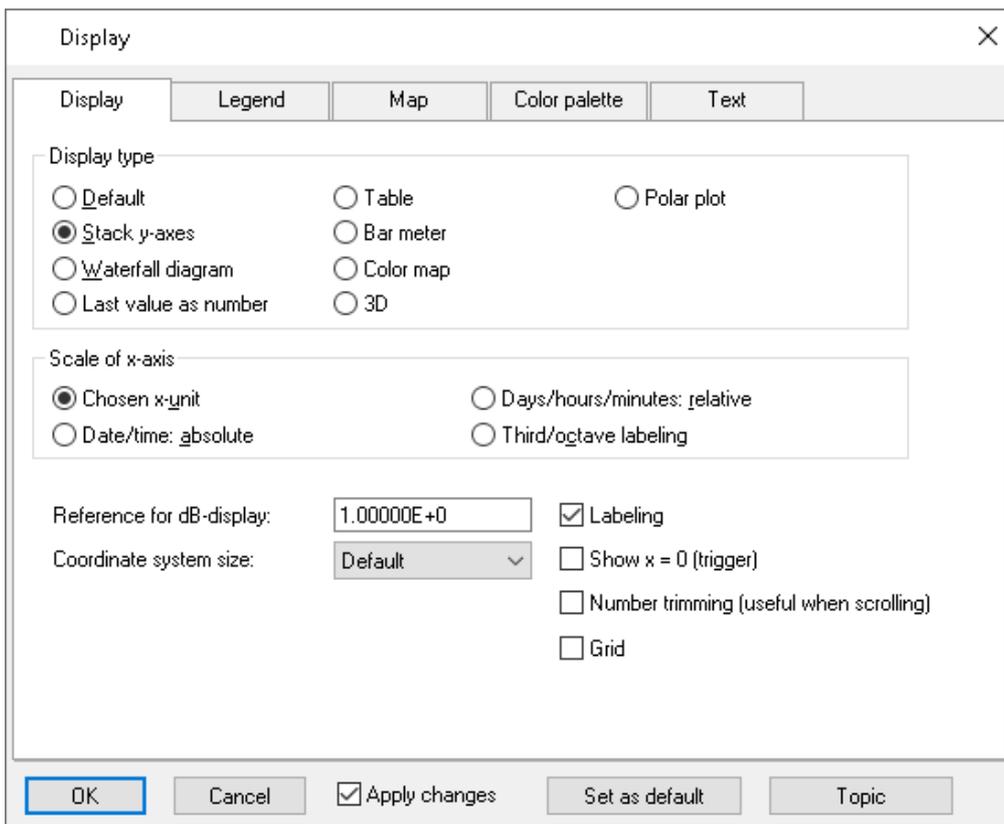
Function

The display mode of a curve window can be defined in a variety of ways.

Calling the dialog

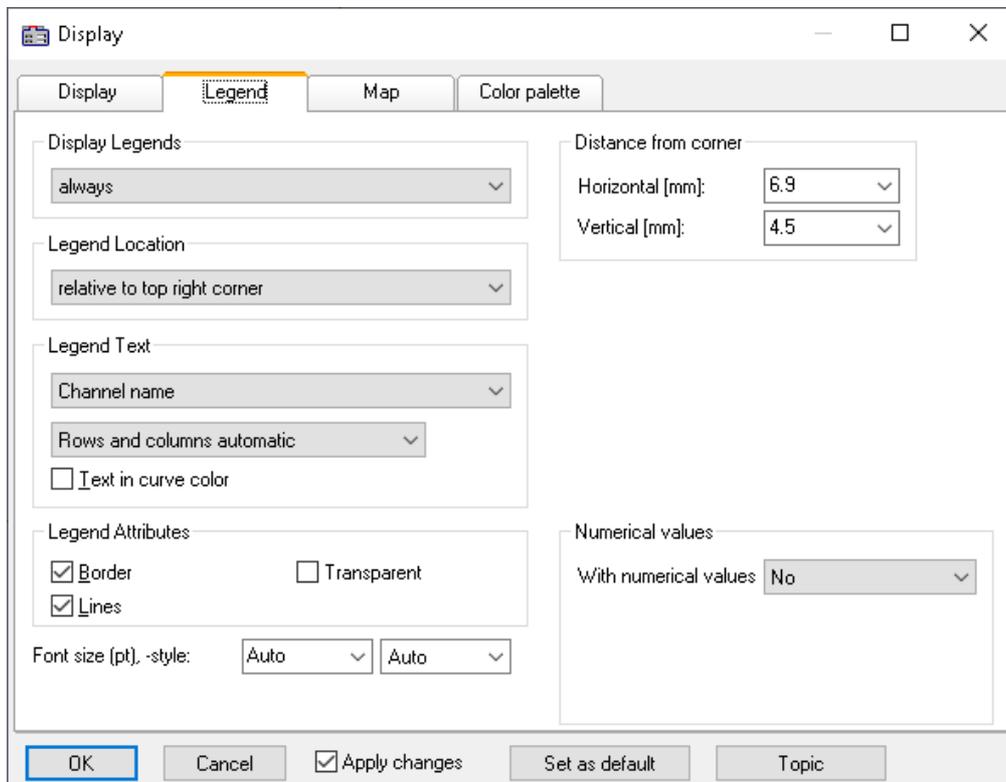
Select menu option "*Configuration*" > "*Display*" in the curve window. A dialog appears in which the curve window display can be defined.

Tap	Description
Display ⁸⁵²	Display variants of the curve window
Legends ⁹⁶⁷	All curve windows can include a legend, with the channel (variable) name, line color etc.
Map ⁹⁷⁰	Map lets you place a picture in the background. It is additionally possible to layout the picture in specified coordinates.
Color palette ⁹⁸³	A channel's color can be governed by a reference signal's amplitude.



13.5.5.1.1 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

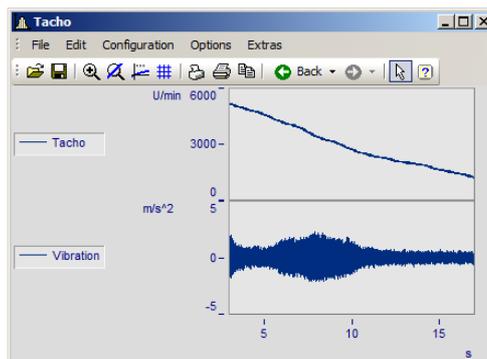
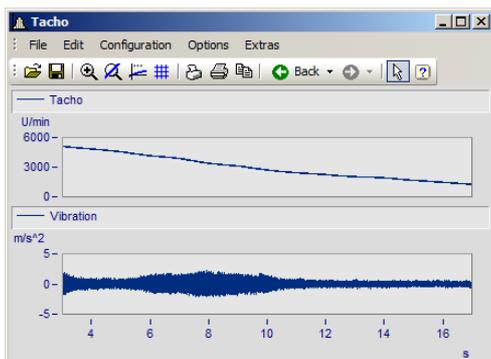
Specify here the conditions under which a legend is to appear. The following options are available:

- *Automatically*
- *Always*
- *Never*
- Only if more than one curve is present

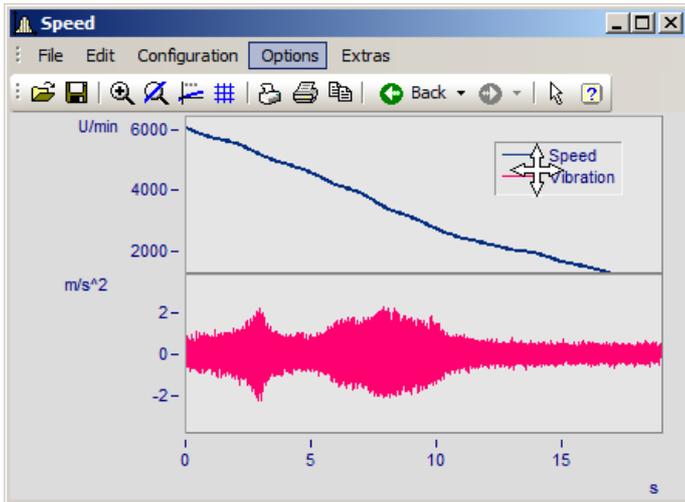
Legend Location

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

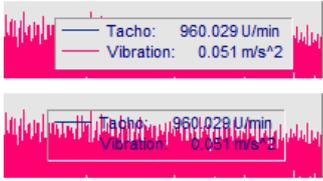
Specify here the text which is to appear in the legend, as well as the format and color of the text. The following options are available:

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 ¹⁰⁹⁷ .
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.

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".

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.

OK, Cancel, Set as default

For details on operation, see the chapter [Confirmation bar](#) 1008

13.5.5.1.2 Map

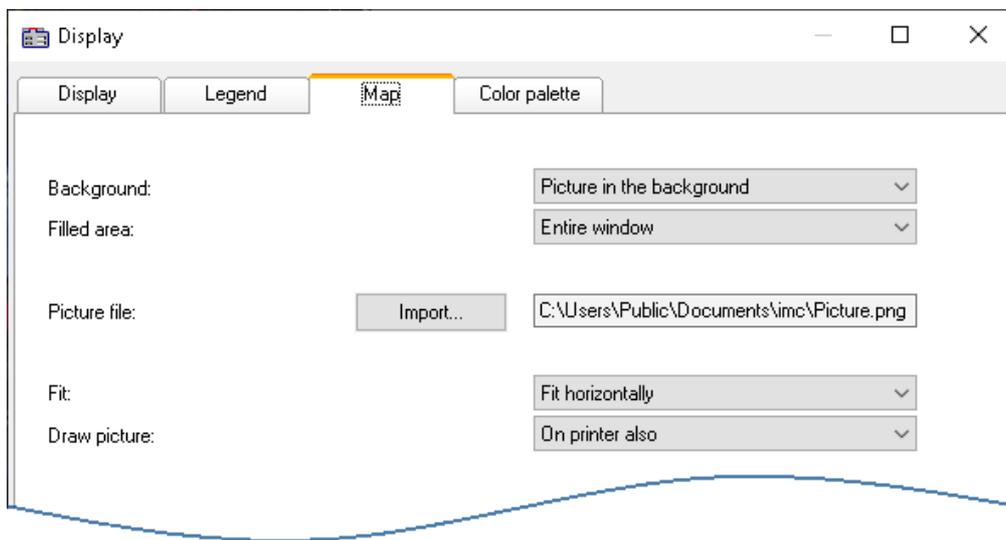
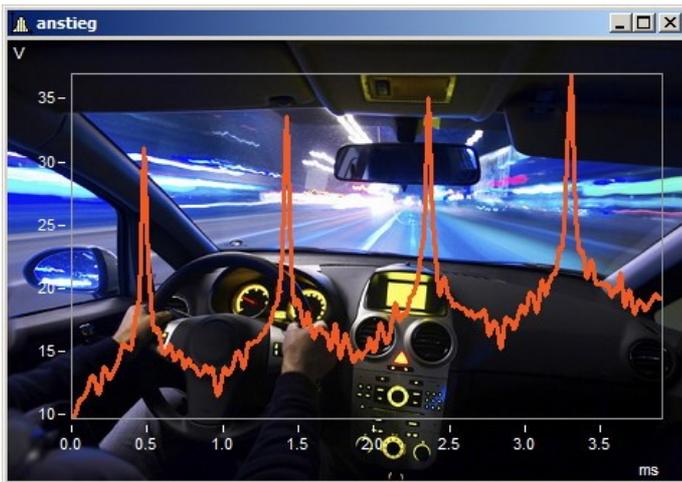
Function

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*.

13.5.5.1.2.1 Background picture



Settings	Description
Background	Select Background to place a picture in the " <i>Background</i> "
Filled area	" <i>Coordinate system</i> " or " <i>entire curve window</i> "
Fit	" <i>auto</i> " adjusts both the picture's horizontal and vertical dimensions. " <i>Retain size</i> " does not adjust the size but centers the picture. " <i>Fit horizontally</i> " and " <i>Fit vertically</i> " 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.

13.5.5.1.2.2 Map

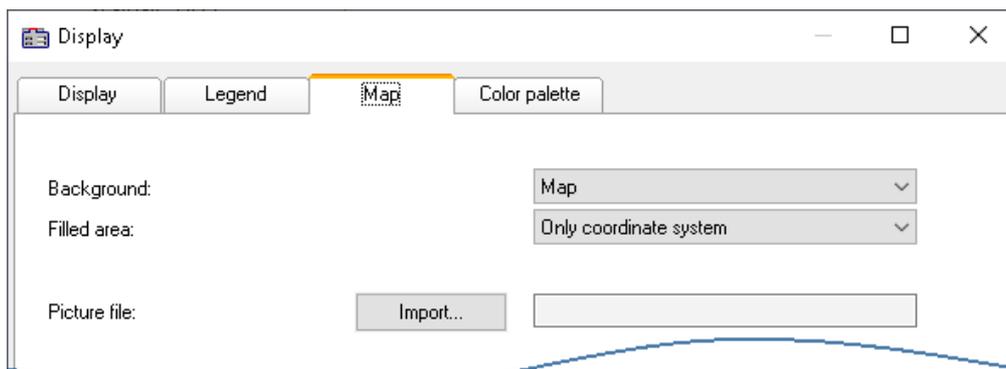
Positioning a map as the background picture

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

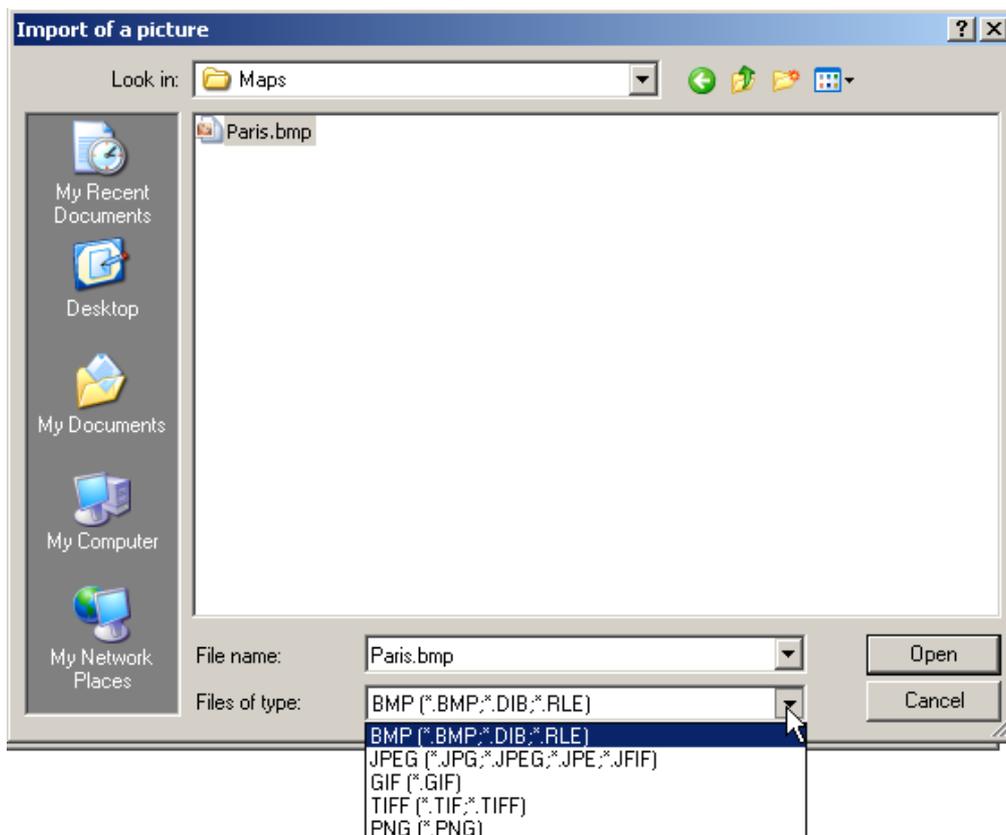
The following example illustrates how GPS data are linked with a map.

Import of a picture



Select the "Map" page and set the background to "Map"

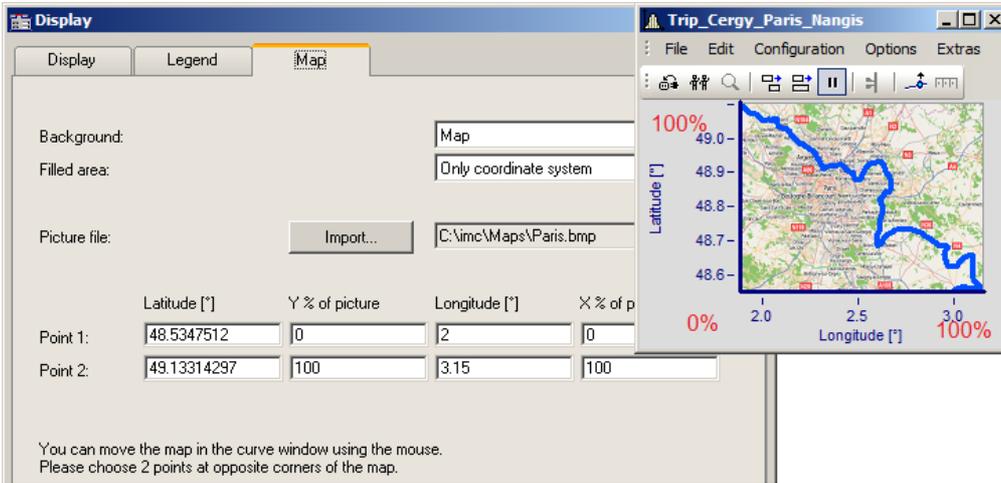
Click on the button *Import* and select a *picture file*. Specify its format.



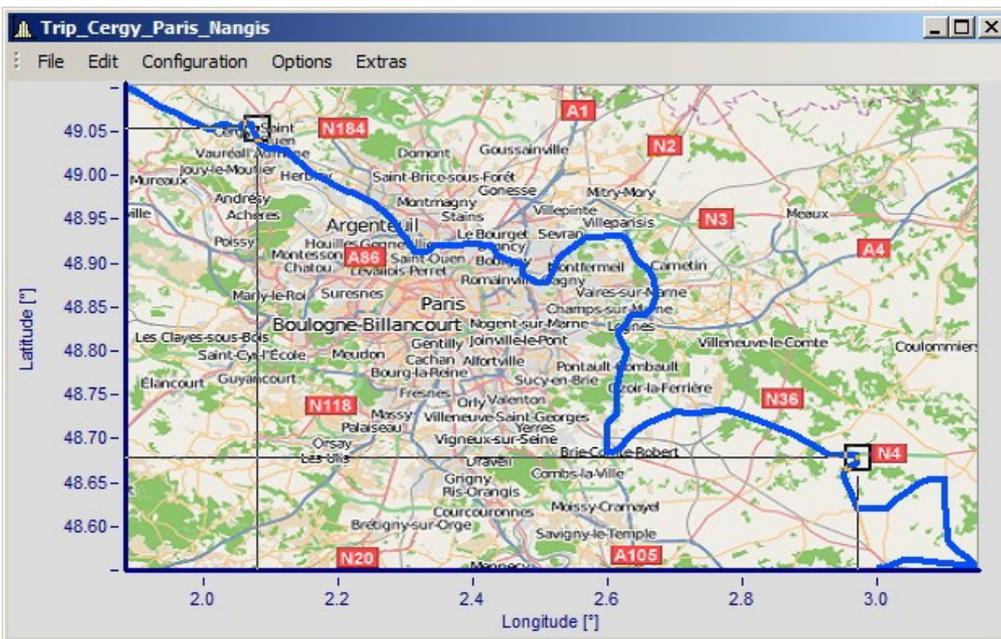
Selecting a picture file. All customary graphics formats are available.

Adapting maps and Map Mode

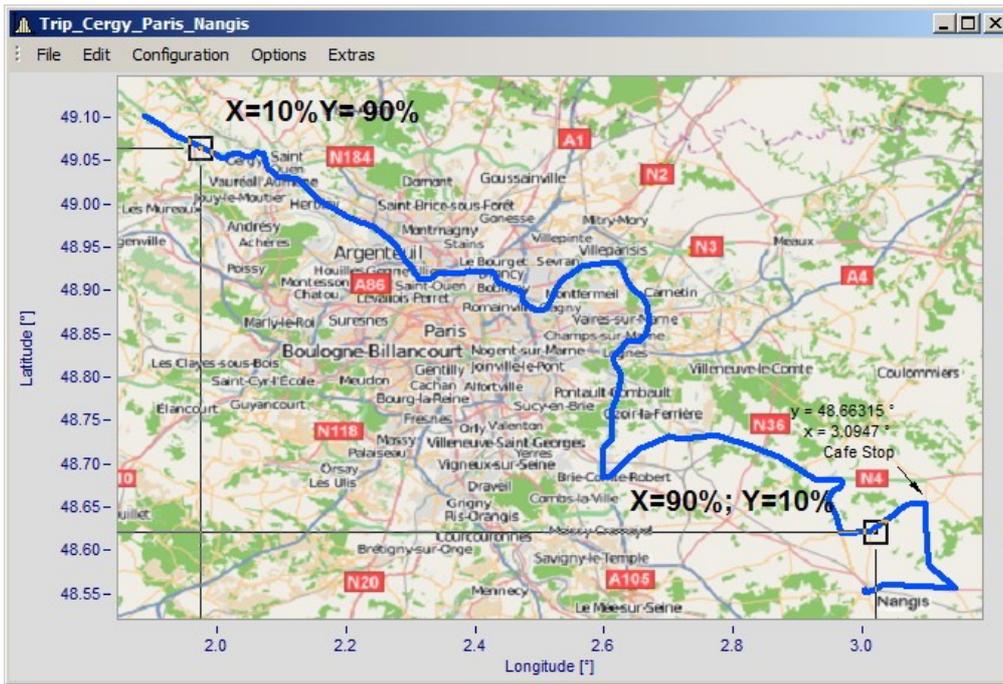
To position a map, you must specify two points. This accomplished by making entries in the input boxes. The curve window is then in Map Mode.



Specifying points: Point 1 at bottom left, Point 2 at top right



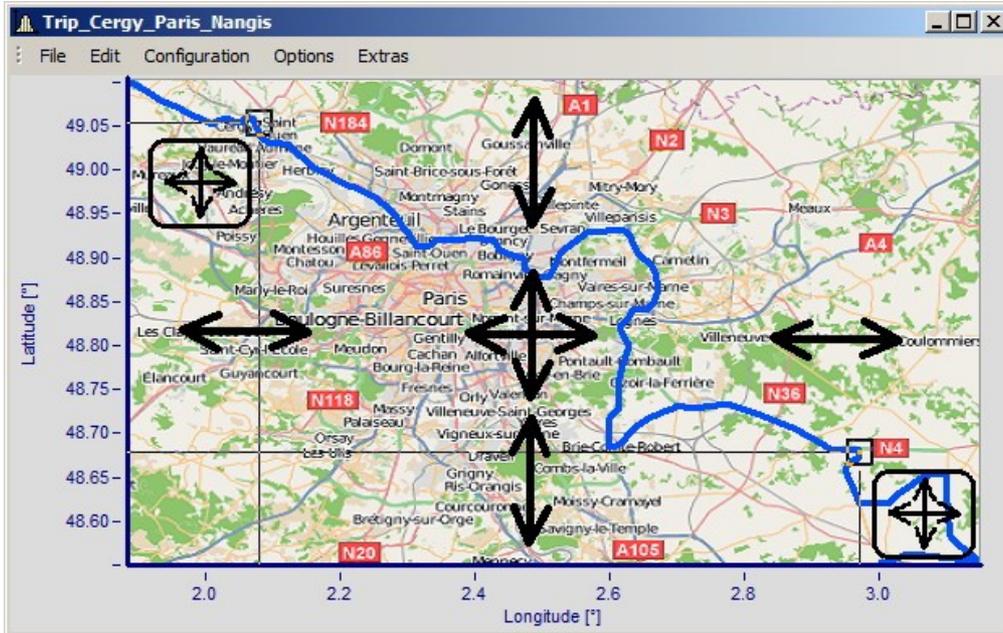
The points are entered into the curve window while the Map dialog is open. Move the points by clicking within the squares and dragging the points using to the mouse to the correct positions.



By moving, the points' X- and Y-positions are updated.

The mouse pointer changes appearance in response to the position in the curve window.

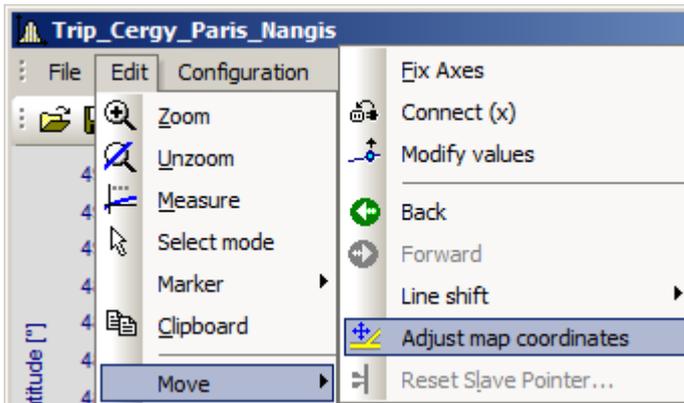
- Drag the entire map by clicking in the center of it.
- Stretch the map by clicking either at the right, left, top or bottom of the map.
- Move the position points by clicking within the squares.



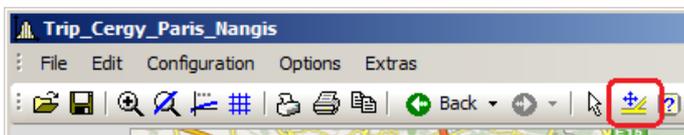
Adjusting the map and points in response to the mouse position

Activating Map mode retroactively

Once the map dialog has been closed, you can use menu commands to adjust the map: *Edit\Move\Map*.



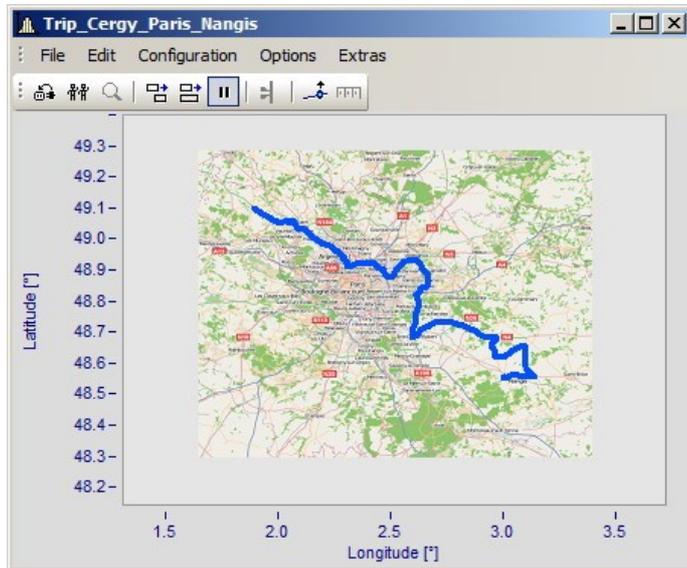
Alternatively, activate the mode by means of the toolbar. To do this, adapt the [Toolbar](#)¹¹⁰⁹⁰.



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](#) ⁹⁷⁰.

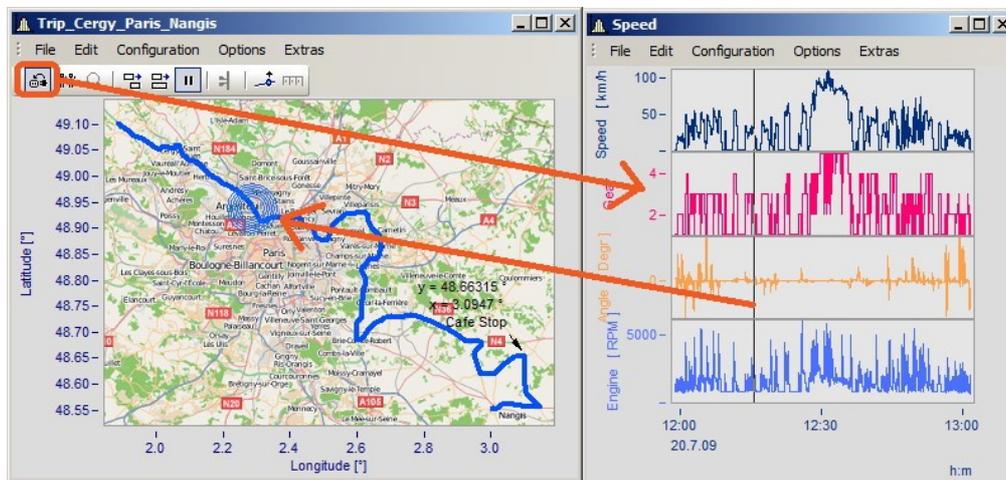
Constraints:

- *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.

13.5.5.1.2.3 Map (from Internet)

Automatic loading of a map from the Internet

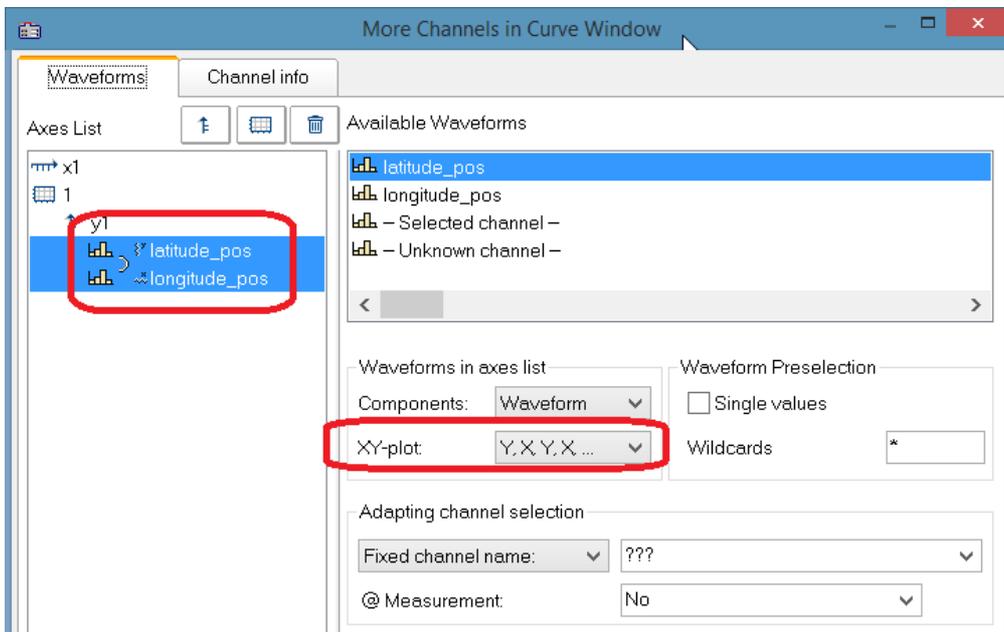
On the following conditions, a picture can be correctly displayed as a map:

- selected curve window configuration: *Display \ Default*
- linear axes
- only one coordinate system
- a data set with plausible position data is displayed.

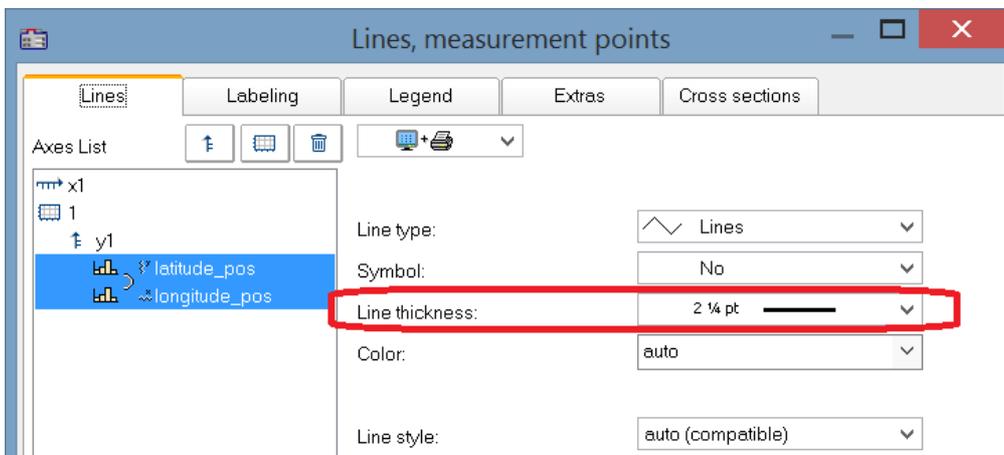
The following example shows how GPS data are automatically superimposed on a map from the Internet.

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":

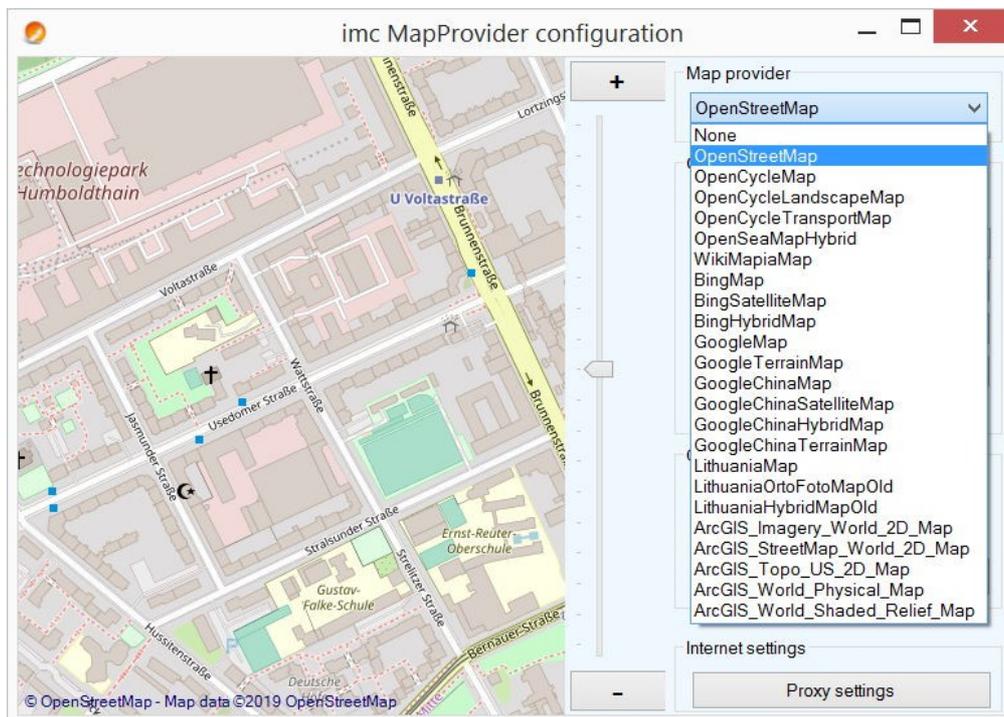
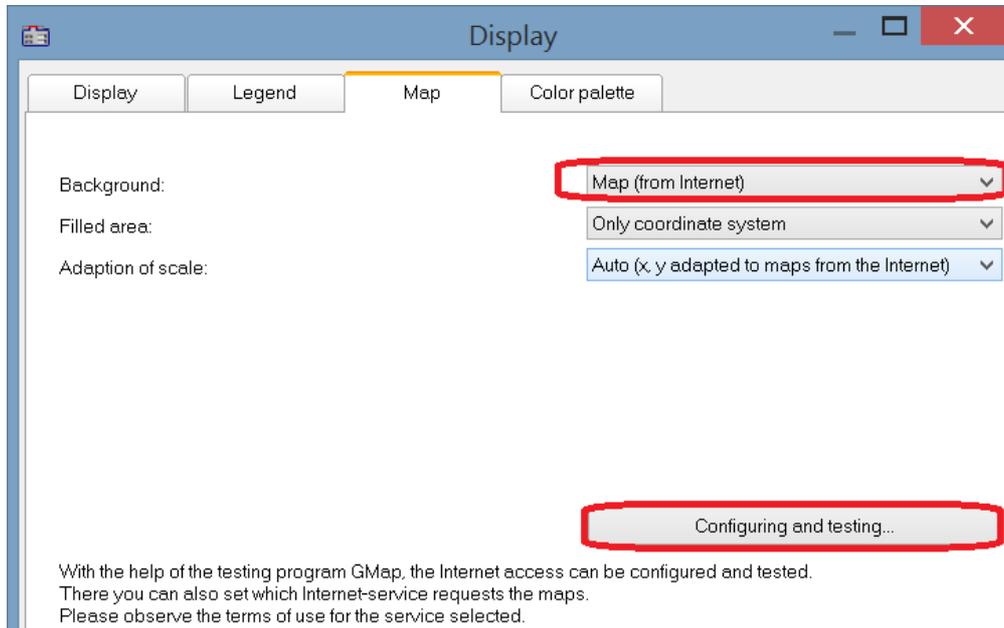


It is advisable to draw the lines more thickly:

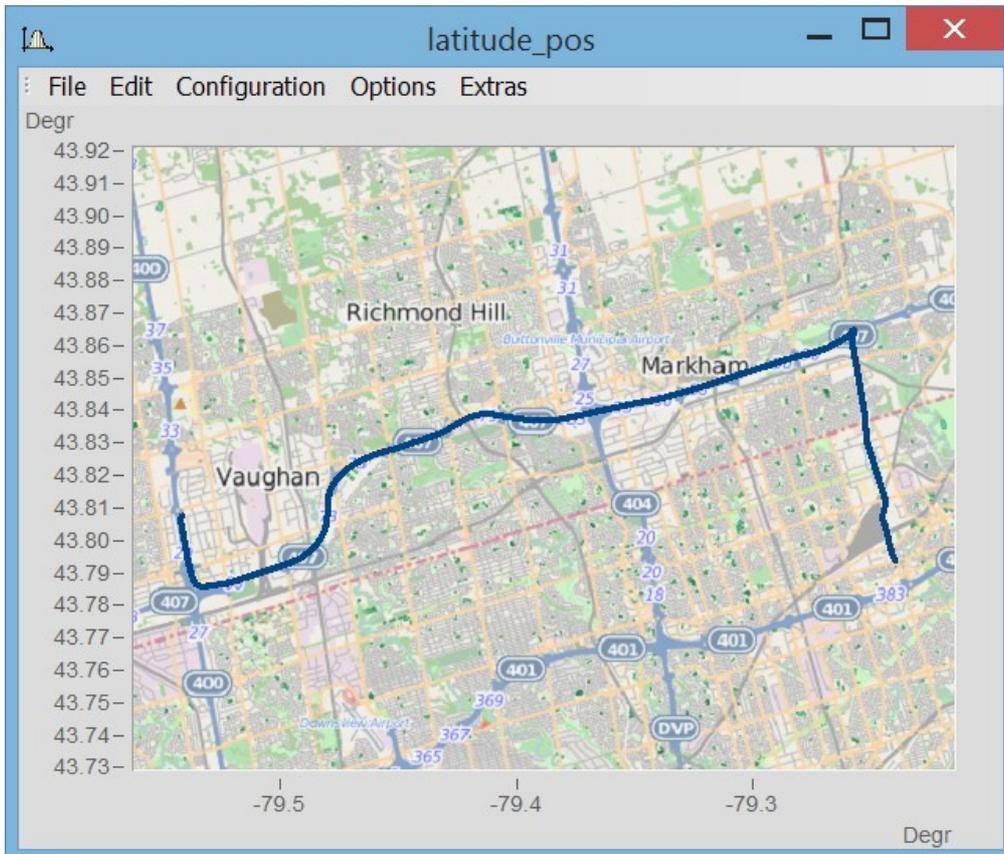


Selecting the map service

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.

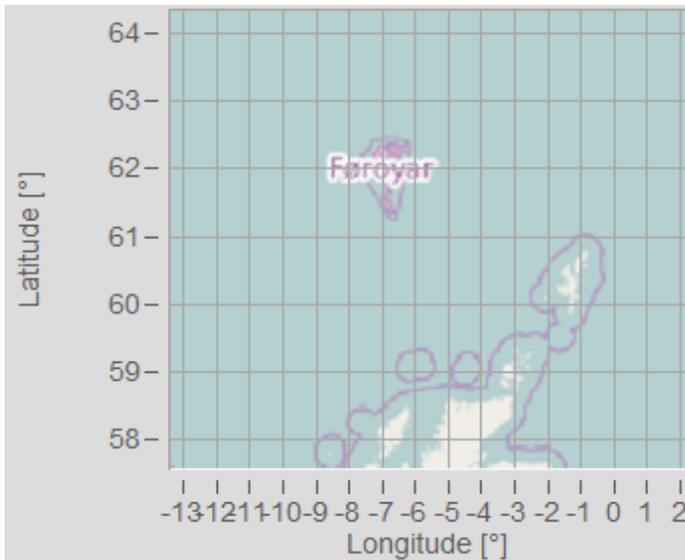
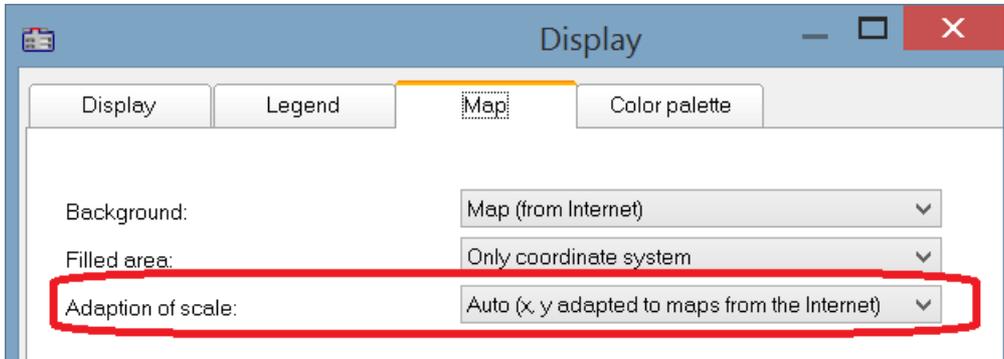
1. Open the folder. By means of *%programdata%*, Windows opens the folder *ProgramData* and then go to *imc\Common\Settings*.
2. The file *AdditionalMapProvider.config* is an XML file with the following structure:

```
<?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>
```

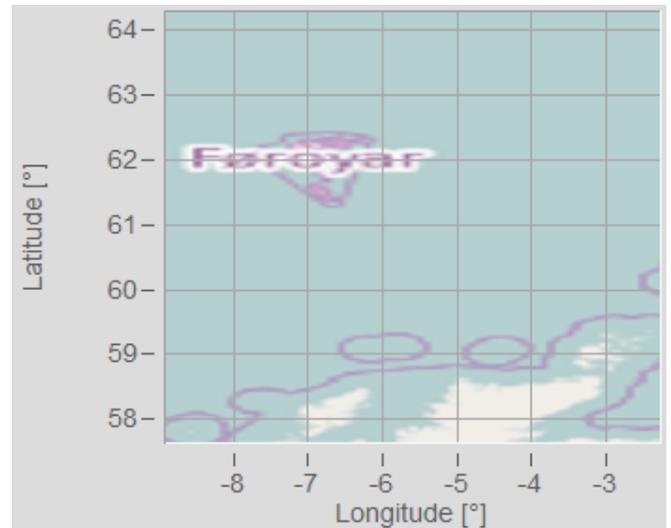
Replace the lines `<MapProvider Name="... TileThreads="2"/>` with the information pertaining to your map provider.

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

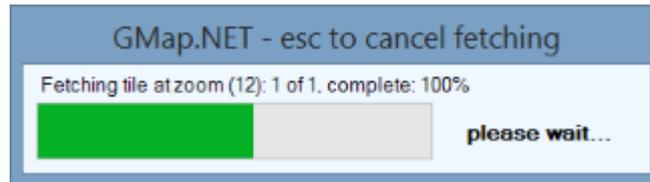
This projection named after Gerhard Mercator (1512-1592) causes imprecision in the curve window, since the coordinates are displayed equidistantly in it. The imprecision increases with the distance from the Equator and the size of the map.

More map service option

Offline cache

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

Select the map section by its coordinates and zoom to the desired section.

Place

Select the map section by entering a location.

Internet Settings

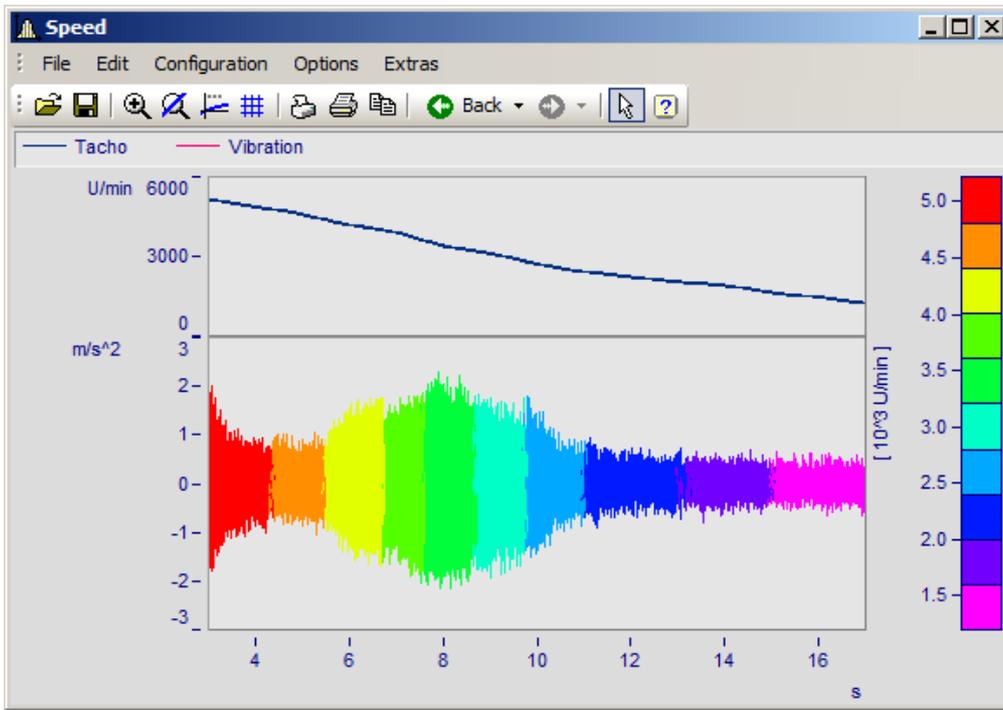
Here you can specify a proxy-server.

13.5.5.1.3 Color palette

A channel's color can be governed by a reference signal's amplitude.

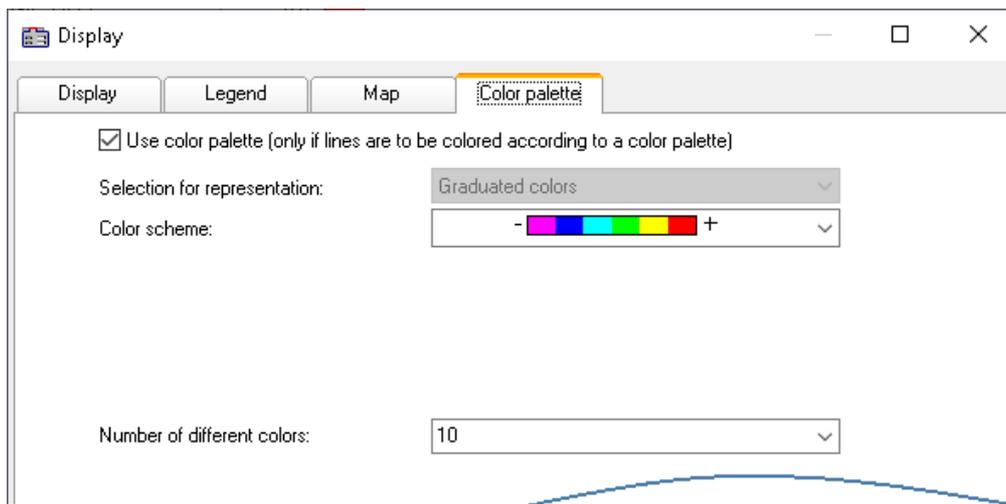
Note

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

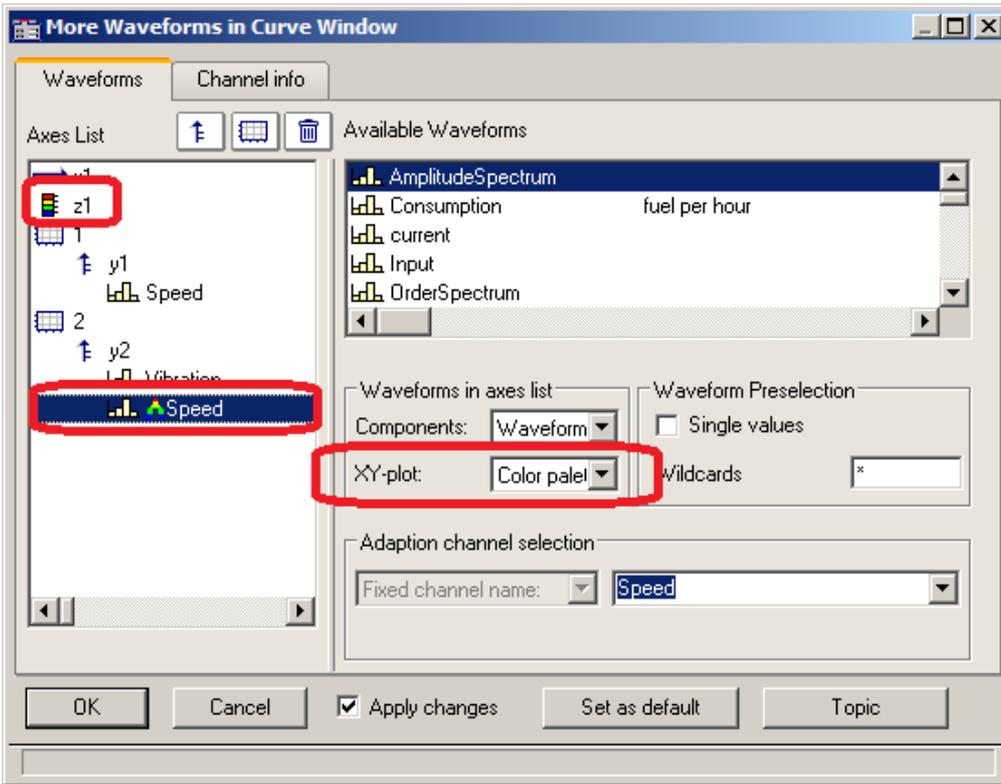
Select the page *Color palette* (not *Display type Color map*) in the dialog called by *Configuration\Display*.



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.

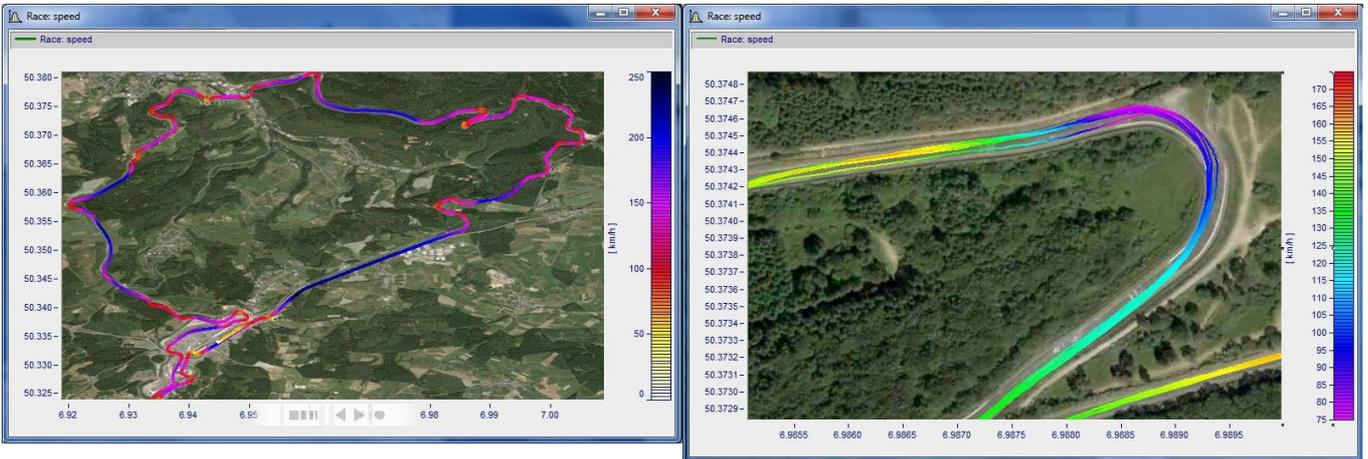
The same settings options for the color scheme and amount of colors are available as for [Color map](#) ⁸⁹¹.

The assignment is performed in the dialog *More Channels...*, which is called from the curve window's context menu.



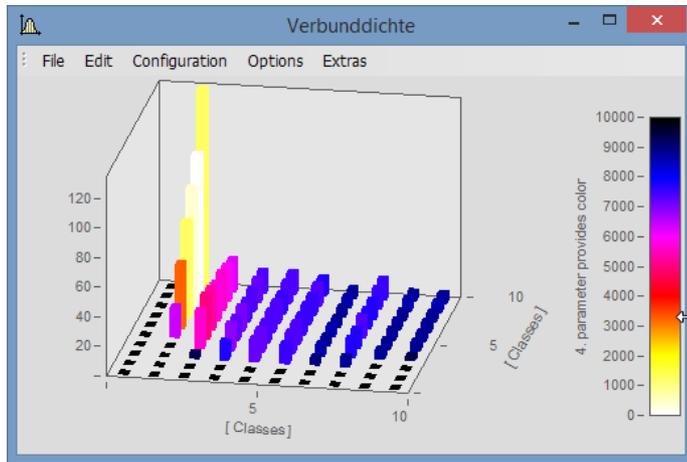
The reference channel is assigned to the same y-axis as the channel to be colored. Select a reference channel and set the *XY-plot* to *Color palette*.

Example:



Color palette as 4th parameter in 3D

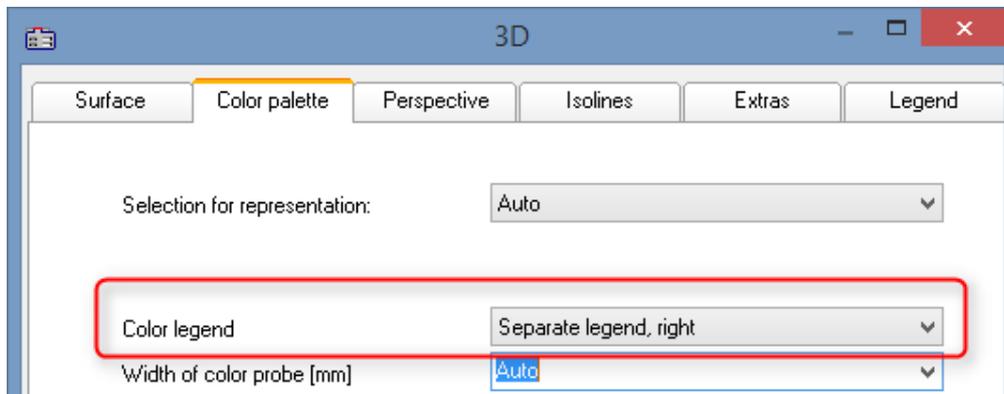
A 3D display can be colored according to a fourth parameter. Thus, 4 dimensions (4D= 3D + color) can be displayed. See also [here](#)⁹⁹⁴.



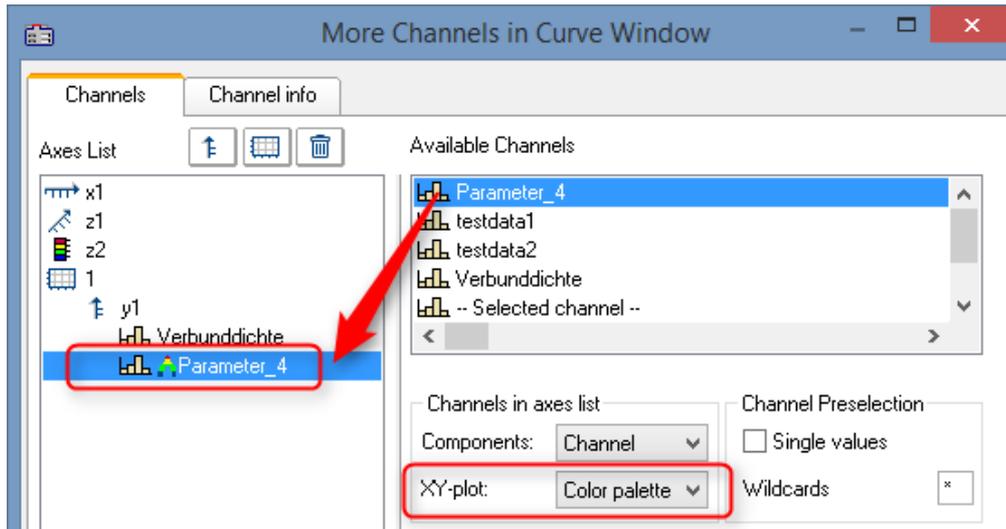
3D display with colors governed by a fourth data set

Procedure:

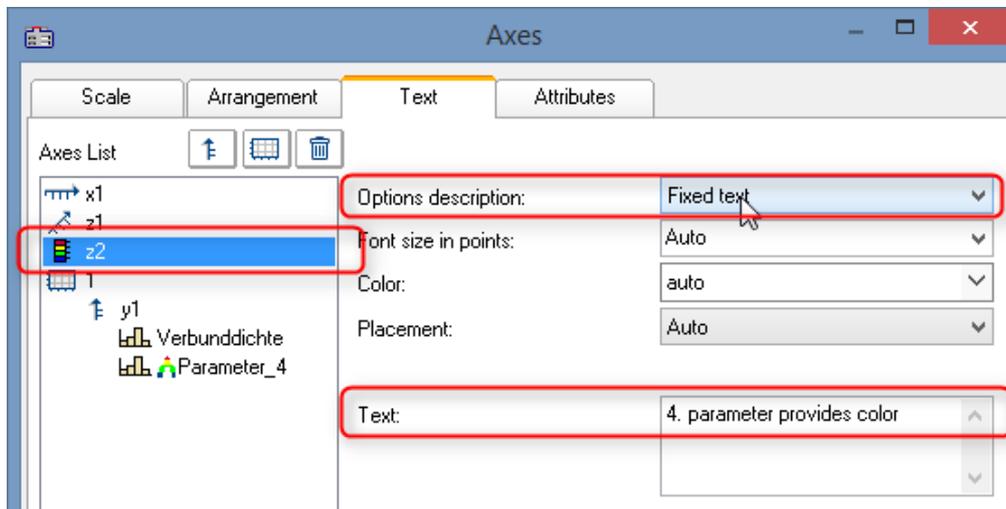
1. Set 3D display with **Separate legend right** or **left** in [Color palette](#)⁹⁹³. With the settings *Auto*, *No* and *Integrated into y-Axis*, it is not possible to generate a colored overlay.
2. Under *More Channels...*, place the fourth variable directly below the 3D data set.
3. Under *XY-plot*, 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"



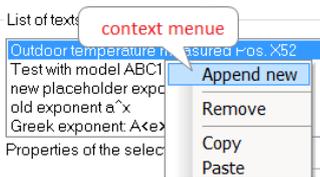
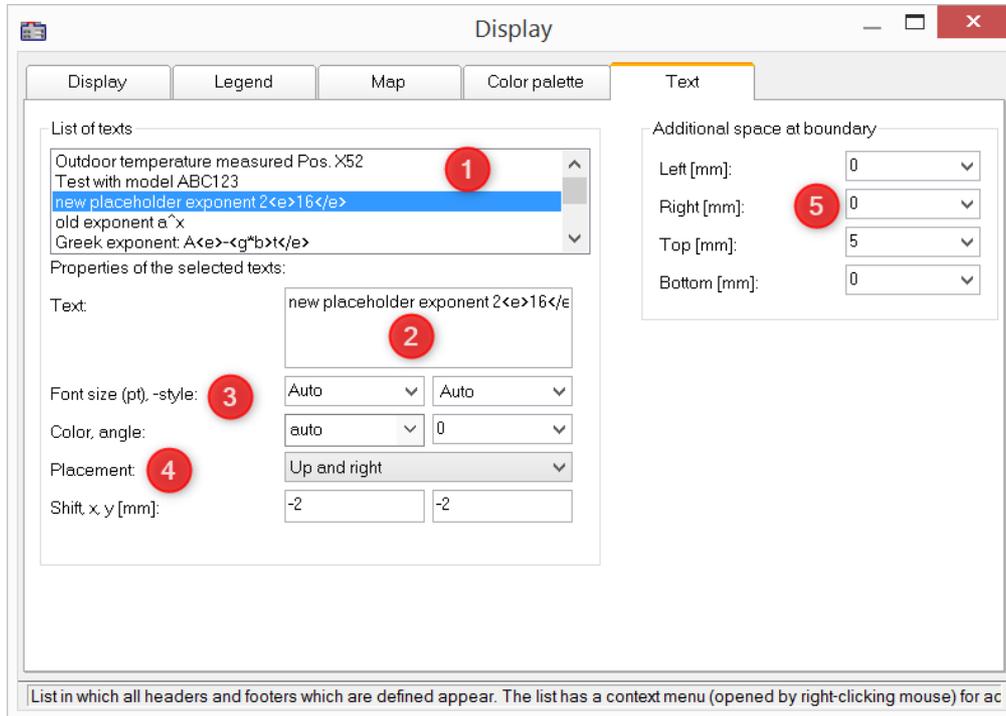
Under "Configuration\More Channels...", assign the color-coded variable



Under "Configuration\Axes\Text" with "Fixed text", state the origin of the coloring.

13.5.5.1.4 Text

The Text page of the Display dialog enables any desired texts to be positioned, for example as titles, headers, footers, or general comments.

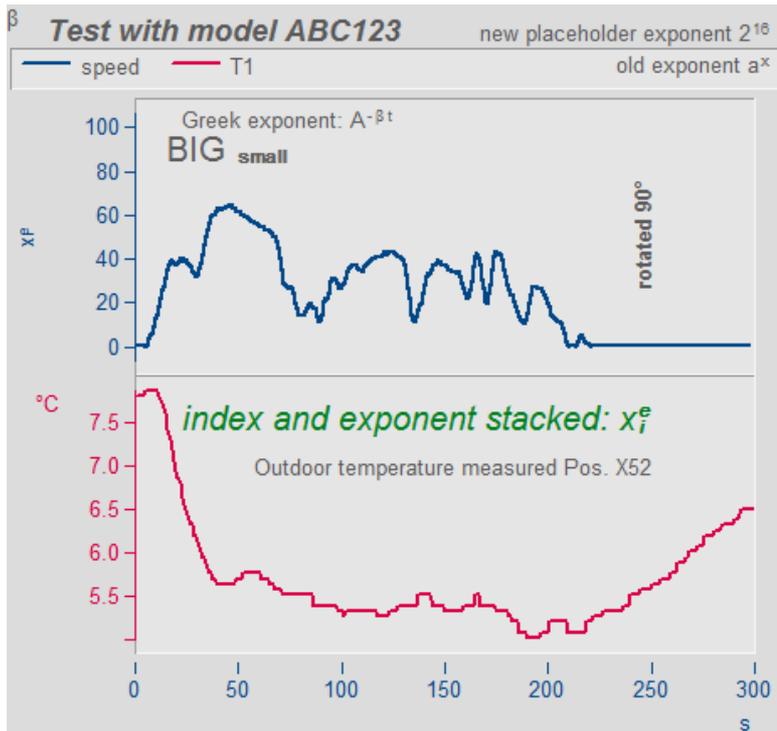


By **right-clicking** in the *List of texts* (1), you open a context menu for managing the entries.

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](#)⁹⁴⁷.

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](#) ⁹⁴⁷.

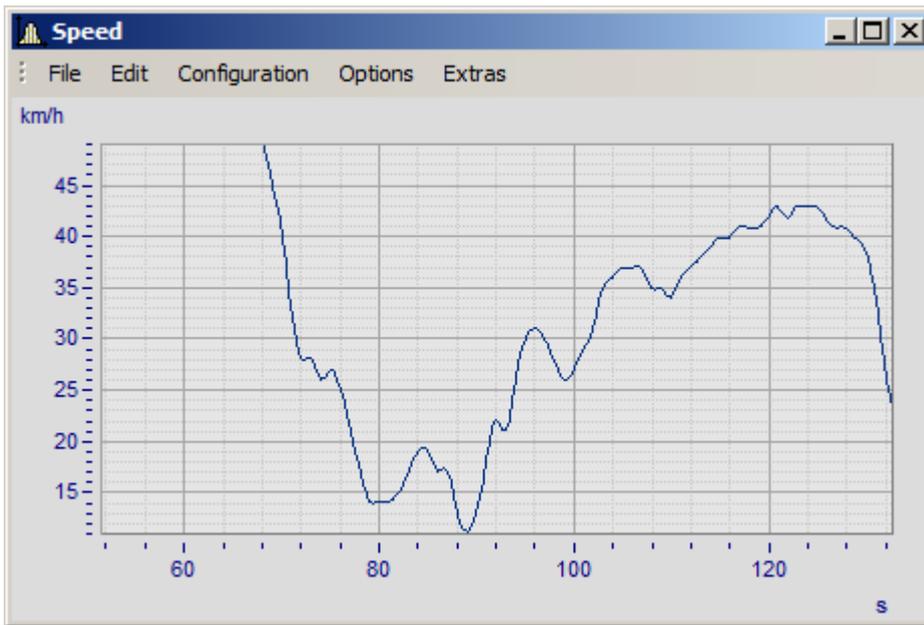
Texts can also be defined as [user ticks](#) ⁹⁴⁹. Here you can also find the function "[Label for raw data](#)" ⁹⁵².

13.5.5.2 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](#)⁸⁵²). Settings for the secondary grid are made in the Axes dialog by means of the control [Small ticks](#)⁹⁴⁰.



Various line widths and types can be specified for the clipboard or print-out, see menu [Opt./Clipboard Settings](#)¹⁰⁷⁴.

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

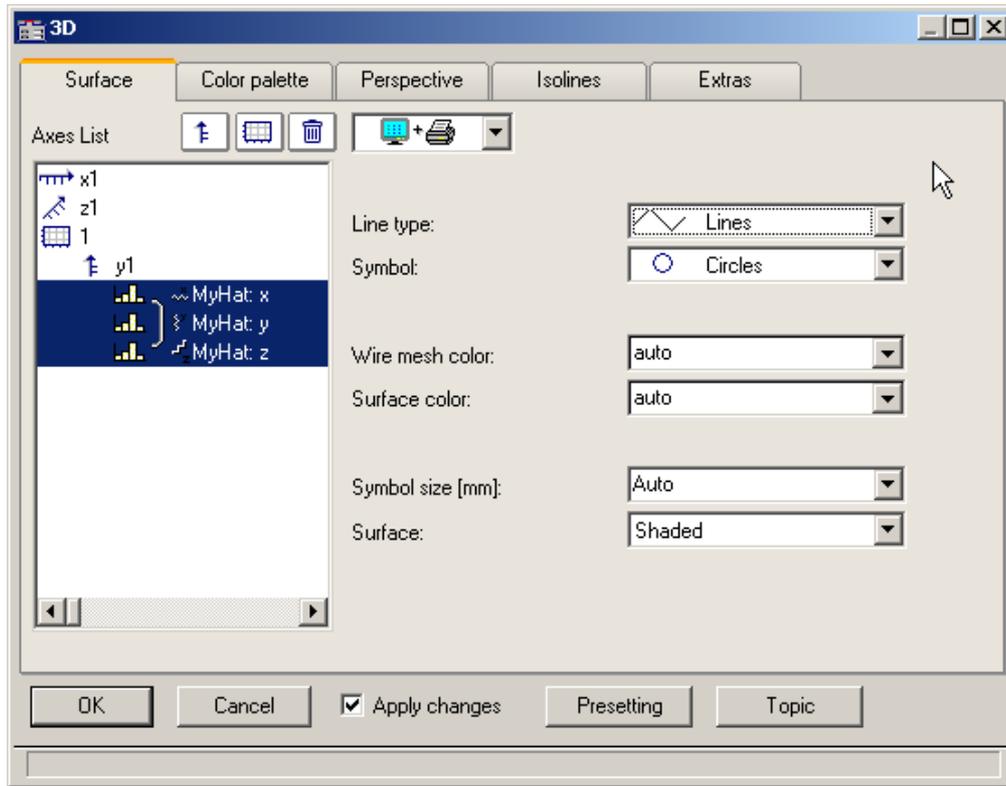
- The color of the grid lines can be determined in the *Colors...* dialog.
- A grid is not drawn if labeling is not visible.
- The menu option *Grid* is marked with a checkmark when grid display is selected.

13.5.5.3 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.

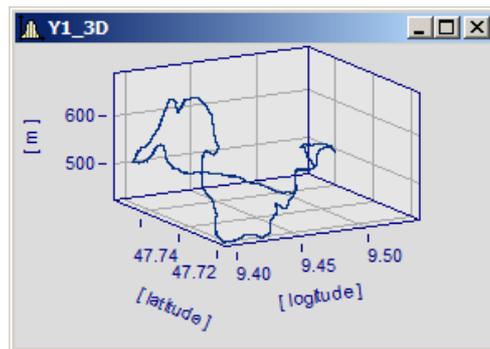
13.5.5.3.1 Surface

In the dialog page *Surface* there are additional settings for 3D-display besides the settings described under [Lines](#)⁹⁵⁵.

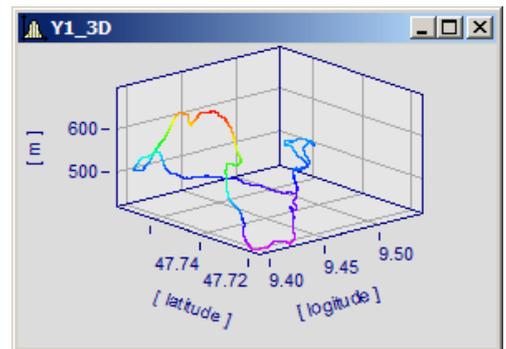


Settings	Description
Line type	In 3D-display, the only available line types are straight lines, no lines, or interpolated lines connecting measured points. For bars, see here ⁹⁹² .
Symbol and Symbol size	See the section on lines ⁹⁵⁵ .
Wire mesh color	Here you can set the color of the grid, or of the 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 ⁹⁹³).
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 ⁹⁹³).

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.	



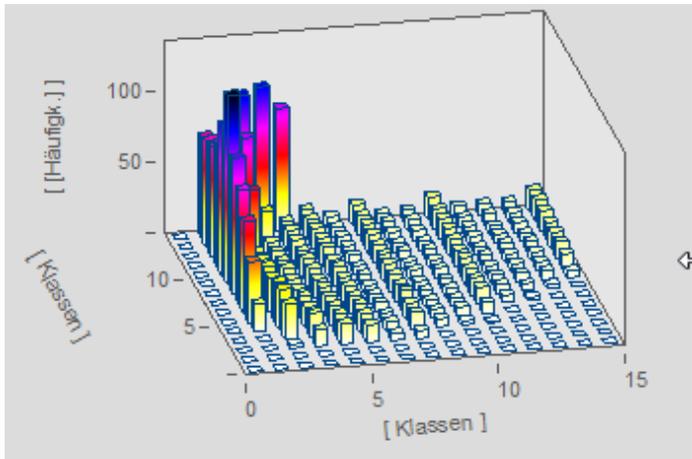
Monocolor space curve



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.



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

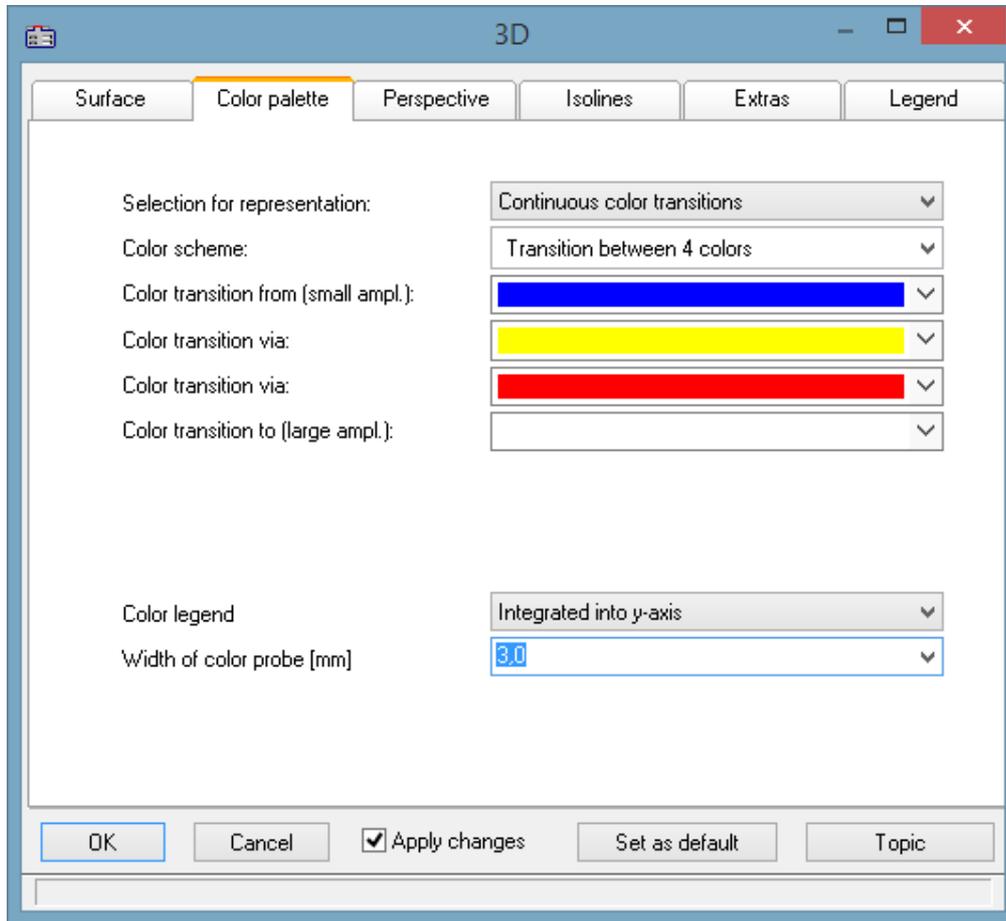
Note

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: <ul style="list-style-type: none"> • <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.

13.5.5.3.2 Color palette

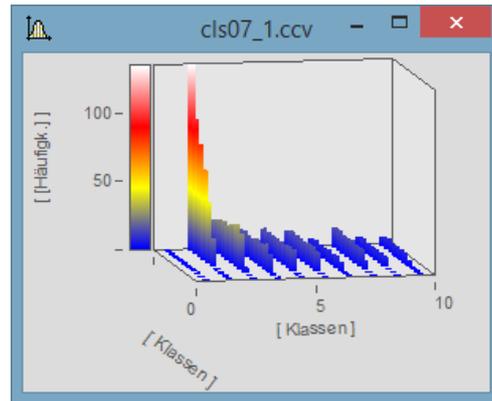
You can make a choice in the *color palette* tab to select a color for the surface. There are two illustration facilities like the continuous color transitions. You can find more information in the chapter [Color map dialog, "General"](#)



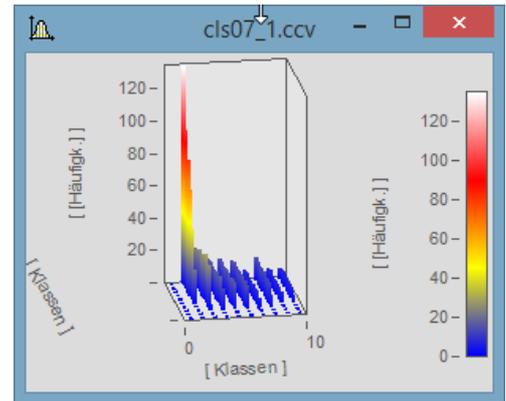
Color legend

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



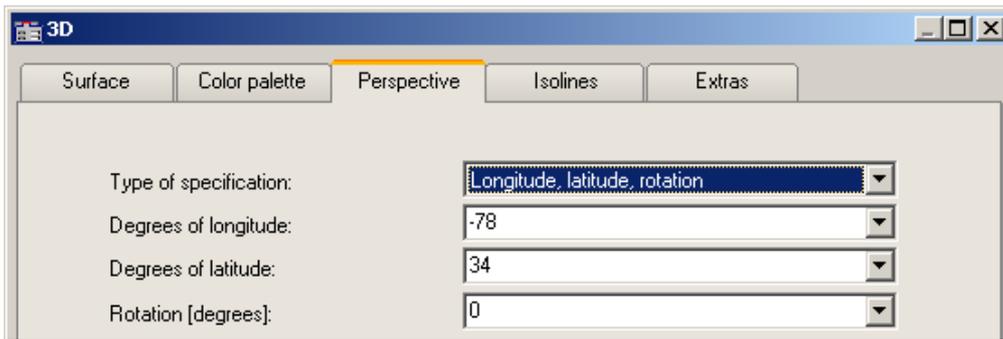
Color legend separate right



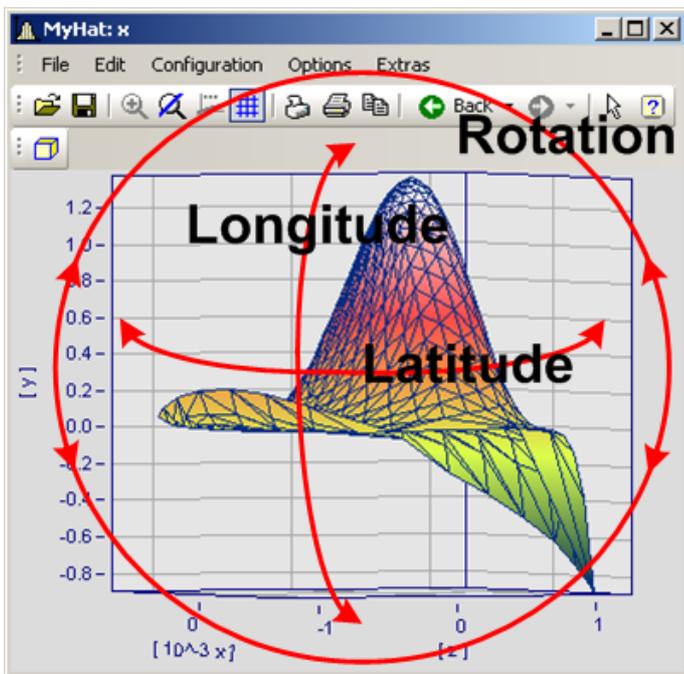
Reference

One way to color the 3D display according to a fourth variable is presented [here](#)⁹⁸⁵.

13.5.5.3 Perspective



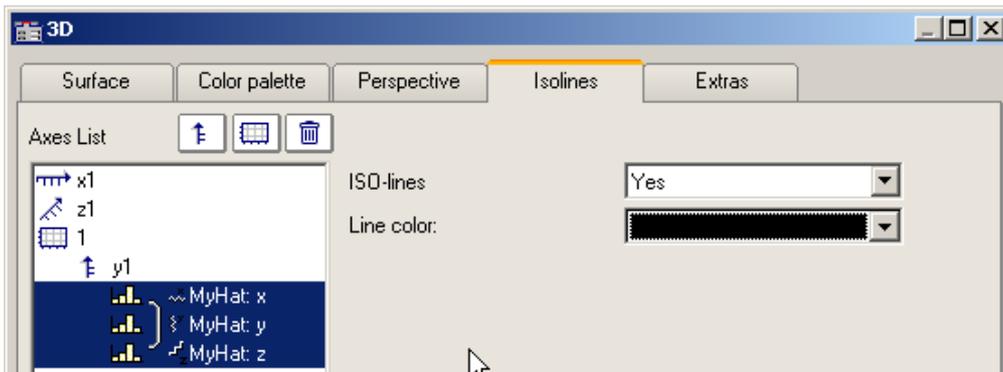
In the dialog page *Perspective*, you can set the viewing perspective as an angle between -180° and $+360^\circ$. There are three ways to make the specification: *Angle of z-axis*, *Longitude and latitude* and *Longitude, latitude and rotation*.



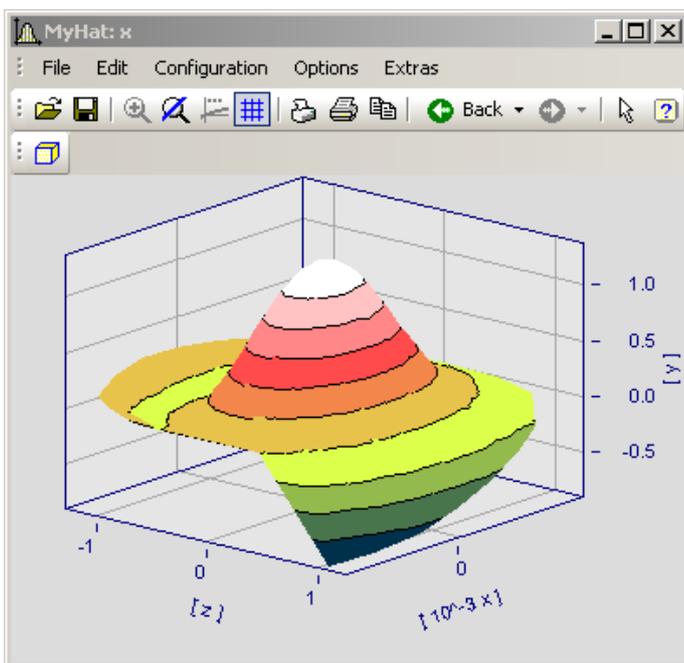
The perspective *Longitude* and *latitude* offers the best mouse control. This view behaves like a survey of the earth from outer space. This is the recommended perspective. If you wish you can additionally specify the rotation.

The three perspective parameters are related as shown below.

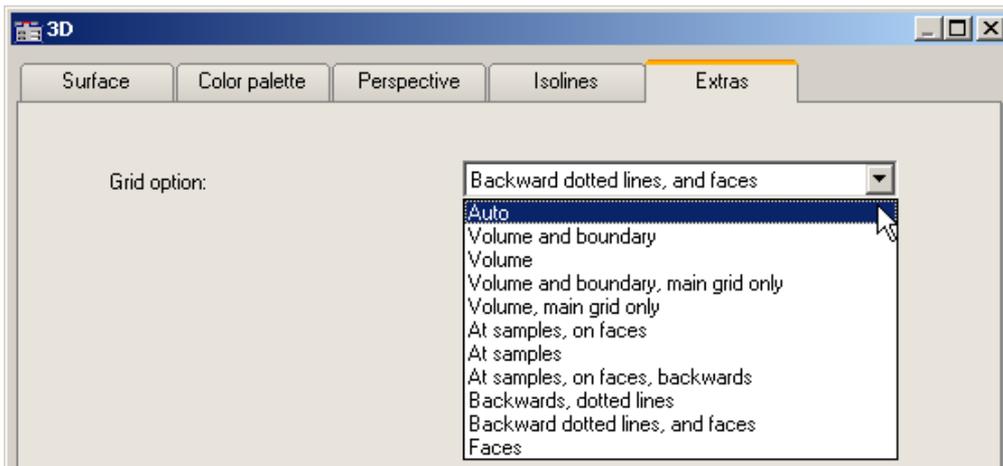
13.5.5.3.4 Isolines



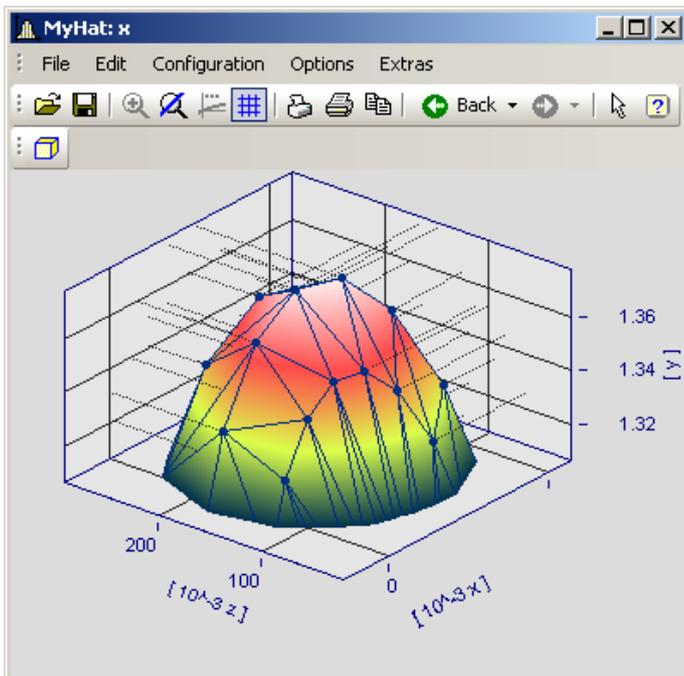
In the dialog page *Isolines* you can select whether to have isolines superimposed on the toned surface, like in color map display. This function can only be activated if graduated colors are set on the [color palette](#)⁹⁹³ page.



13.5.5.3.5 Extras

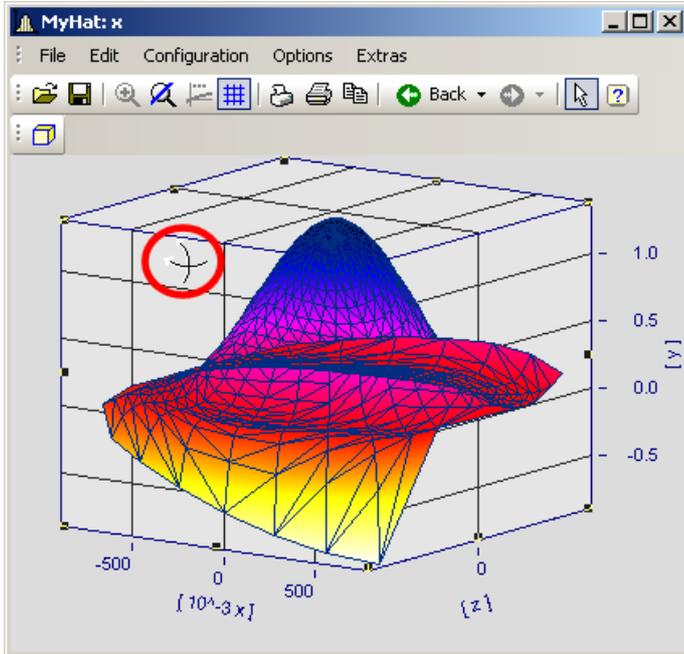


The dialog page *Extras* offers access to various grid options. This grid is then displayed in the space, in addition to the standard coordinate grid, for better legibility of a measurement point's axis values.



13.5.5.3.6 Rotate

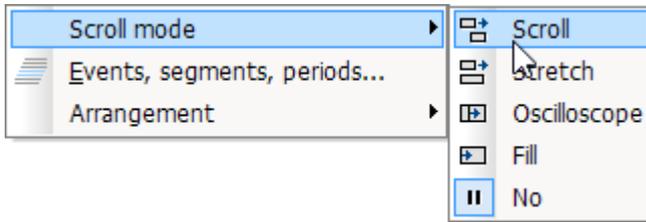
For spatial manipulation of a 3D-display, bring the mouse cursor in the coordinate system, which makes its shape change. If you then click on the mouse and start moving it, the cursor shape changes, as shown in the picture, and the 3D-display can be freely rotated by moving the mouse around.



Cursor for free rotation of a 3D-display

13.5.5.4 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  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.

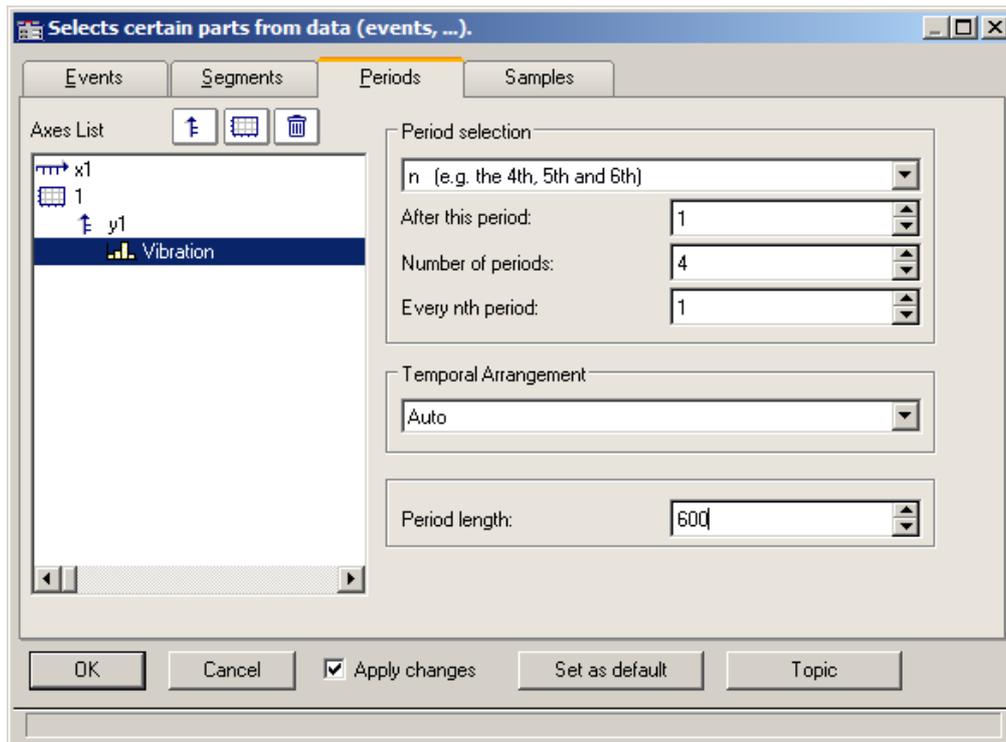
13.5.5.5 Events, segments, periods

Function

With normal waveforms, it is possible to perform a [comparison of signal periods](#)^[1005] 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.

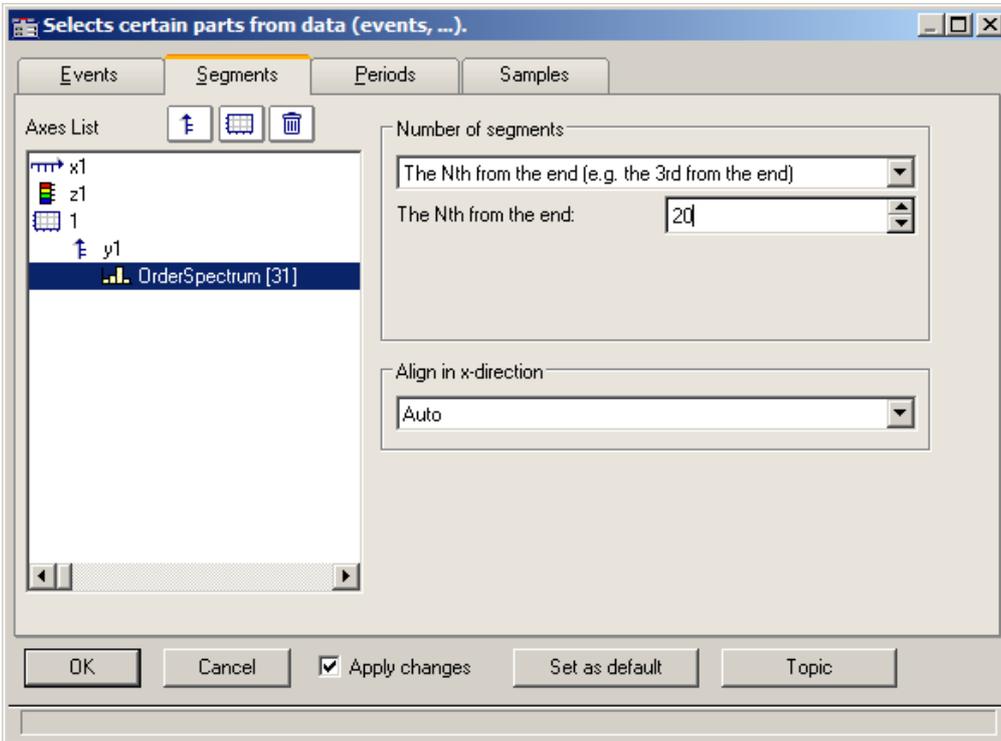
When you have specified the amount, then proceed to the **Temporal Arrangement**:

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:



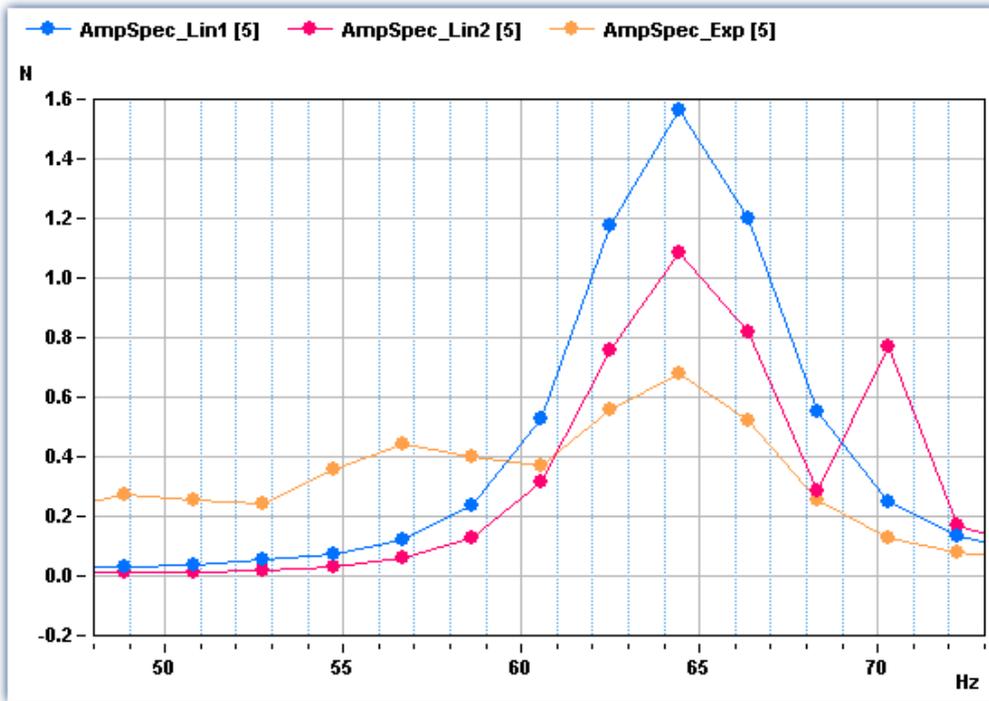
First you must make your selections in the Axes list on the left, as only the selected data indicated there are affected by settings performed with the instruments on the right.

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:

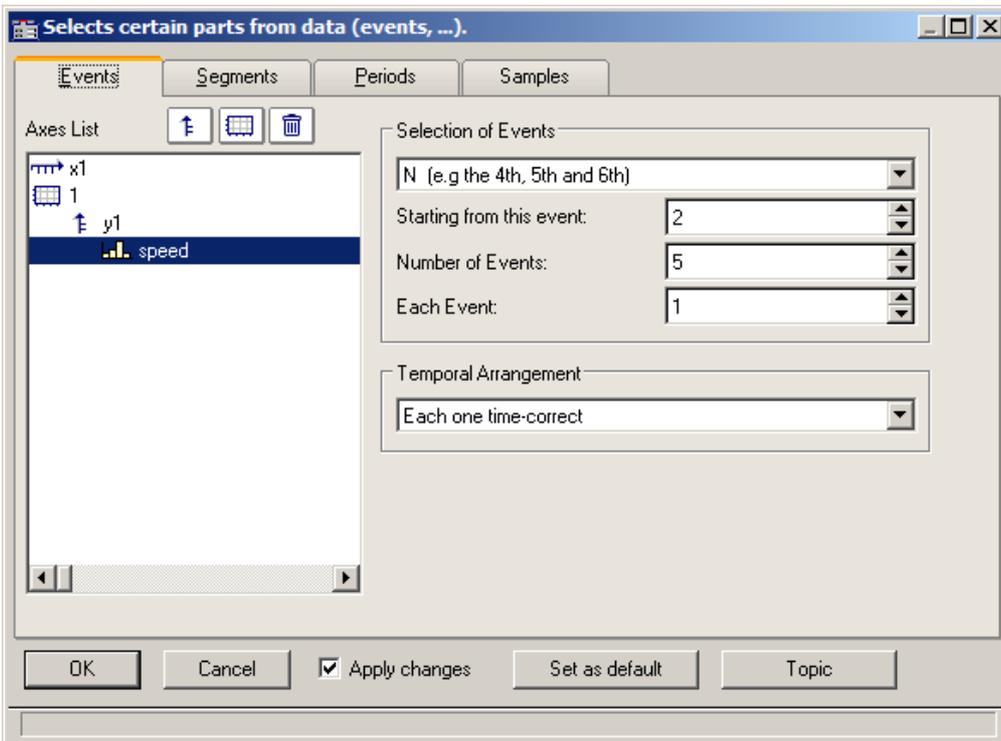


Legend with segment indices

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**:

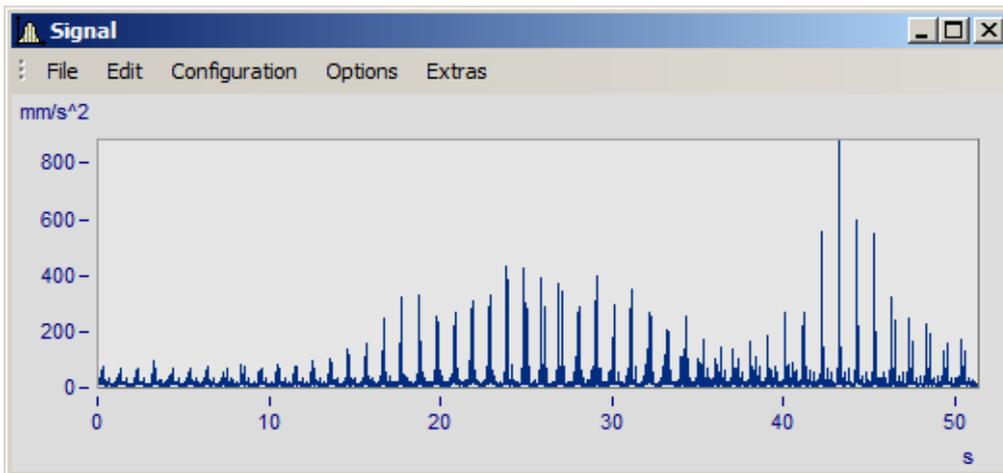
Settings	Description
Auto	This delivers an automatic representation, and is certain to achieve some result. When in doubt, then, use this option.
Each one time-correct	Each event is represented with its genuine time-coordinates. With the curve window's time representation set to "relative", the individual 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.

13.5.5.5.1 Period comparison

Periodic signals occur in a variety of applications, such as rotating machinery, where the measured signal almost repeats itself at each revolution.

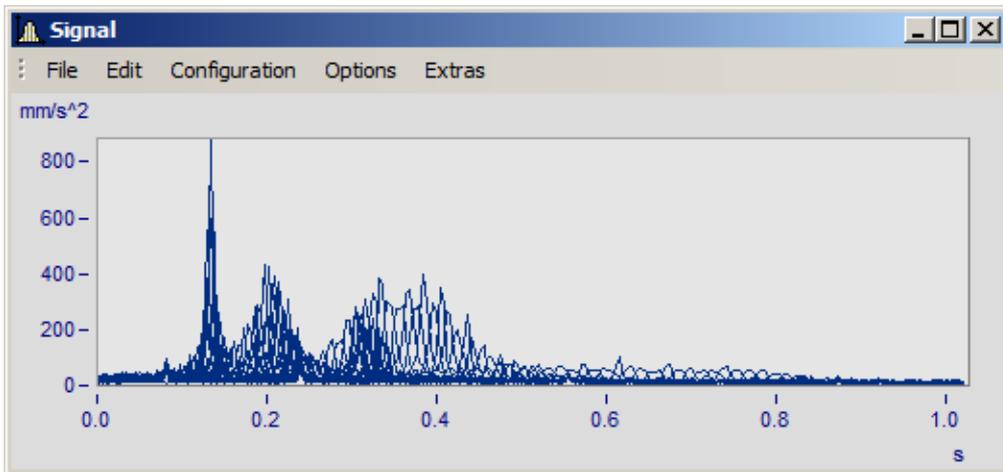
In the curve window's default display mode, entire waveforms appear plotted over time. This means that all periods are drawn in a series. In such a display, it is not always possible to compare individual periods in terms of their maximum values, for instance, or even other properties.

Example:

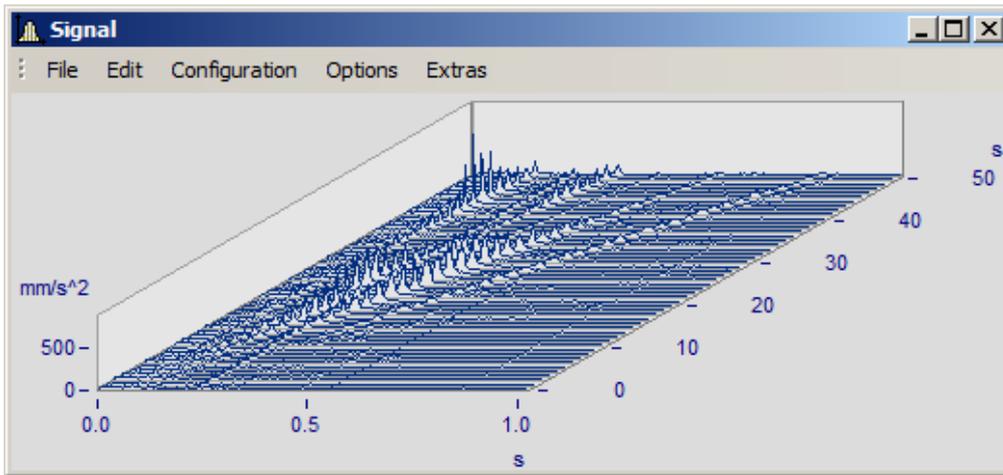


A waveform with 50 periods of 1s duration each is shown

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.

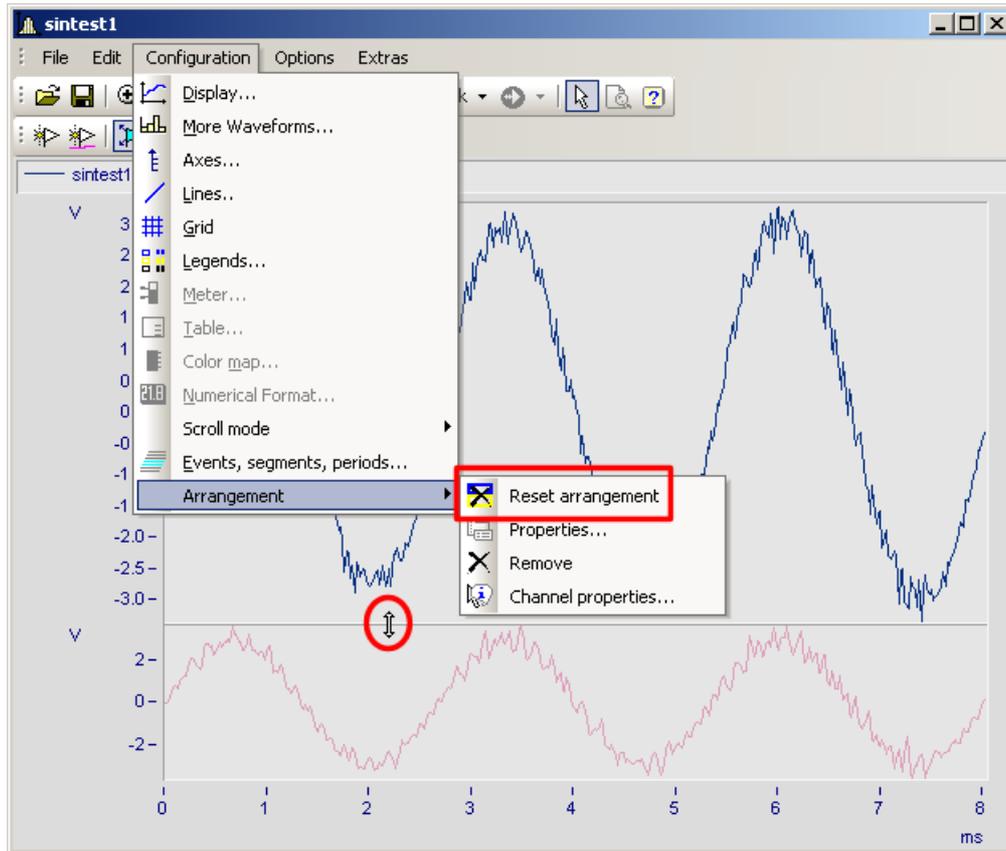


Waterfall diagram: the mountain-range-shaped surfaces clearly illustrate trends emerging across the accumulating signal periods.

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.

13.5.5.6 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](#) ⁽¹⁰²⁶⁾ 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.

13.5.5.7 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

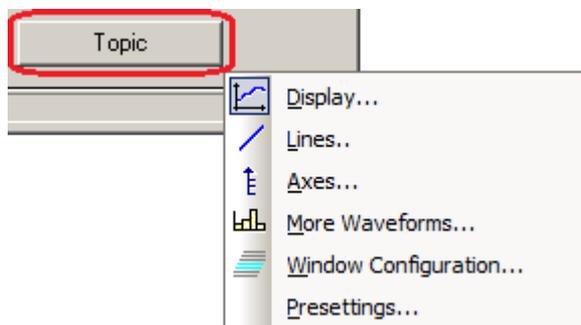
The dialog is closed and any changes made are discarded. This also applies to changes made and displayed in the curve window using the option *Apply changes*.

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.



13.5.6 Working with the curve window

Menu item	Description
 Zoom <small>f1010</small>	Enlarges a section of the curve window.
 Unzoom <small>f1011</small>	Displays the entire curve.
 Measure <small>f1012</small>	A measurement value window and measurement cursors for measuring curves are displayed.
 Select-mode <small>f1026</small>	In this mode, it is possible to select and edit legends, coordinate systems, axes, curves and markers.
Marker <small>f1028</small>	Proceeds to the list of marker functions.
 Clipboard <small>f1052</small>	Copies the graphics in the curve window into the Windows clipboard.
 Graphic-export... <small>f1052</small>	The graphic of the curve window is saved as image or as a PDF file.
 To MS WORD <small>f1054</small>	The curve window is embedded as an OLE object in Microsoft WORD.
Move <small>f1057</small>	Proceeds to a submenu listing various curve viewing functions.

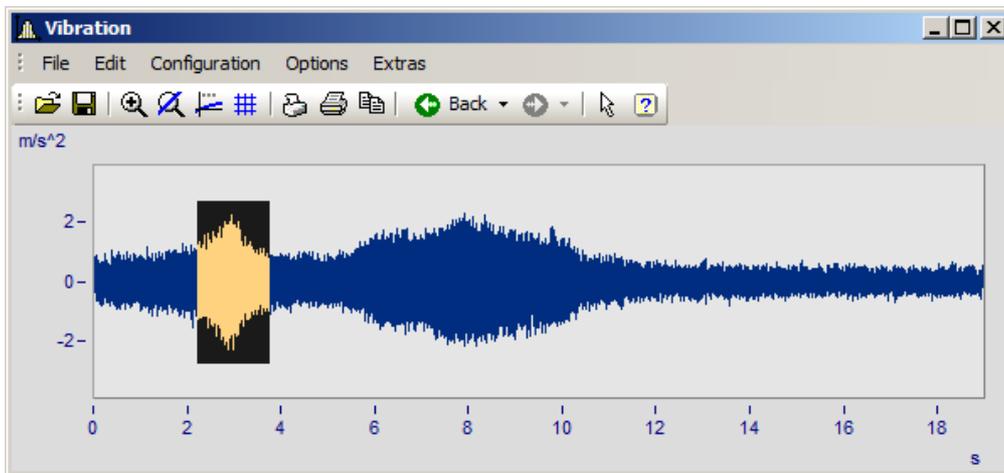
13.5.6.1 Zoom

Function

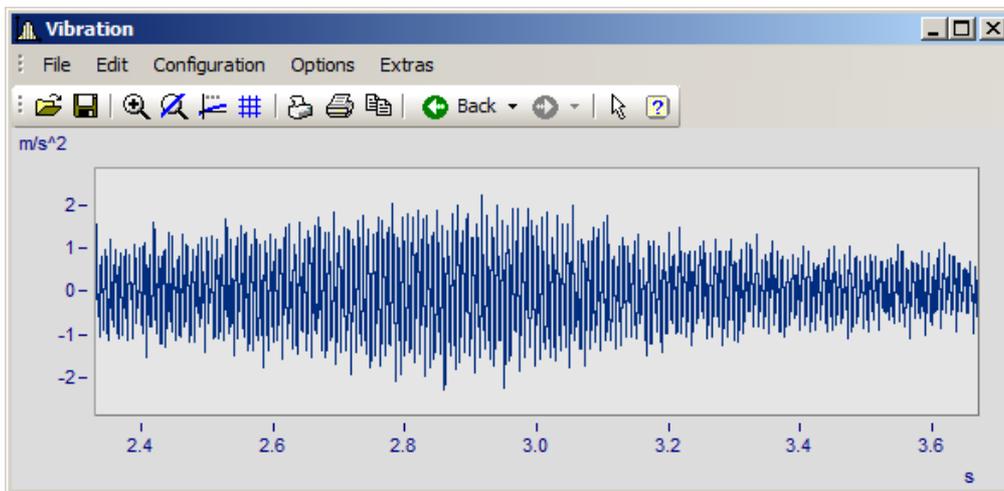
Any region of a displayed curve can be enlarged for viewing (zoomed). Zooming can be done in both x- and y-directions at the same time. The zoomed region is a rectangular area within the coordinate system, with size and position determined by the user.

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](#)¹⁰⁸¹ 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](#)⁹³⁵). 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](#)⁸⁵² 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.

13.5.6.2 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.

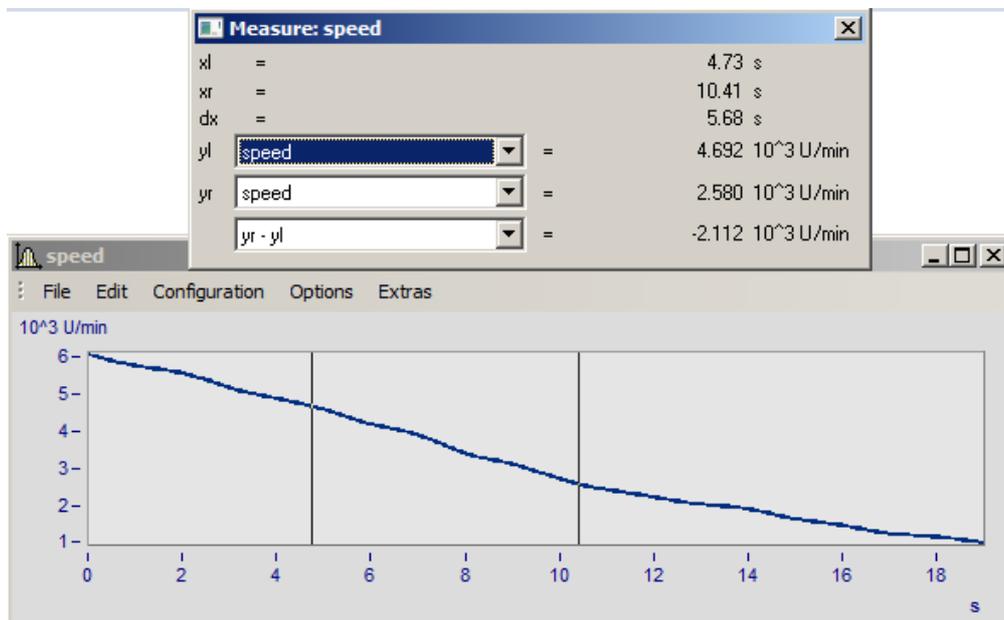
13.5.6.3 Measure

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

- Select *Edit/ Measure* and the *Measure* window appears. 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

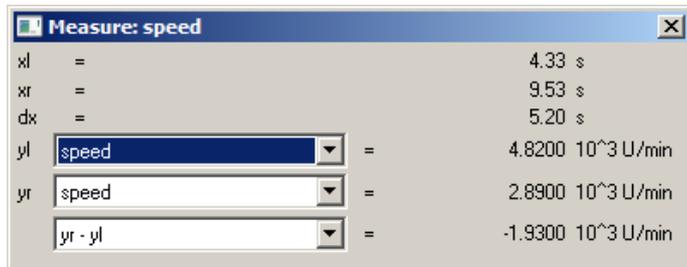
- The crosshairs may cross each other.
- In normal time displays, the horizontal line of a measurement cursor is visible only as long as the curve lies within the coordinate system and is defined for the x-coordinate of the measurement cursor.
- Zooming in XY or locus plots may result in the measurement 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](#)⁹⁵⁴ menu.
- The horizontal measurement cursor can be switched off in the [Options/ Presettings](#)¹⁰⁸¹ 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.

13.5.6.3.1 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.

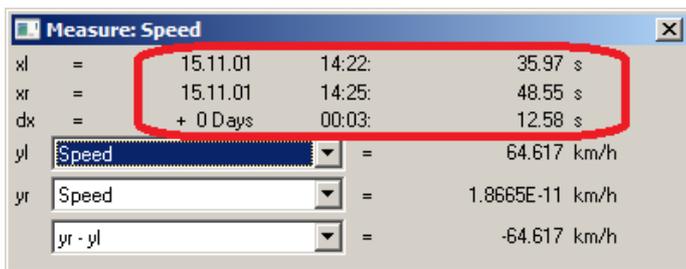


In the selection lists *y1* and *yr*", a waveform displayed in the curve window can be selected for measurement, which is accomplished using the measurement cursors. The list box at the bottom of the dialog contains the following entries as available choices of calculated values:

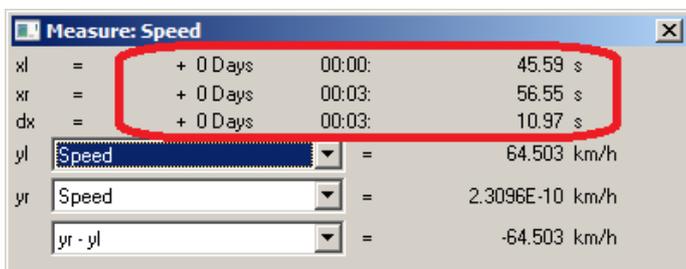
Symbol	Description
x1	x-coordinate of left measurement cursor
xr	x-coordinate of right measurement cursor
dx	Difference between the x-coordinates, xr-x1
xr/x1	Quotient of the x-coordinates of the measurement cursors
y1	y-coordinate of the specified curve at the intersection with the left measurement 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 per 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 = $\text{Sum}(Y_n * \Delta x)$ Available only for equidistantly sample waveforms.
y-Min	Minimum between the measurement cursors. Available only for equidistantly sample waveforms.
y-Max	Maximum 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.
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.
x-coordinate of inflection point	Inflection point according to the following formula: $y_l > y_r \rightarrow$ x-position of minimum slope $y_l < y_r \rightarrow$ 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:



Measure window with date and time view



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](#)^[955] applies to it.

By means of the contextmenu command "[Cursors freely movable](#)^[1016]", 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](#)^[1016]", the horizontal measurement line jumps either to the associated amplitude or to the selected position.

13.5.6.3.2 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

If you select this function, both measurement cursors can be moved freely in the curve window without any linkage to particular curves.

Send Curve Segment to imc FAMOS!

This option sends the curve section located between the vertical lines of the measurement cursors to imc FAMOS. This new waveform receives a name which appears in the variable list and by which it is designated as per the transfer options set using the menu [Options/Transfer options](#)^[1087] in the curve window.

When multiple curves are displayed in the window, the section between the measurement cursors of each of them will be transferred to imc FAMOS if the pertinent transfer option is currently activated.

Note that the original data sets are overwritten if you do not specify a new name when using Transfer Options.

Note that this export function duplicates the waveform in the course of the export procedure, unlike other export functions. Shifting the position of the measurement cursors after exporting a curve section will not affect the exported waveform.



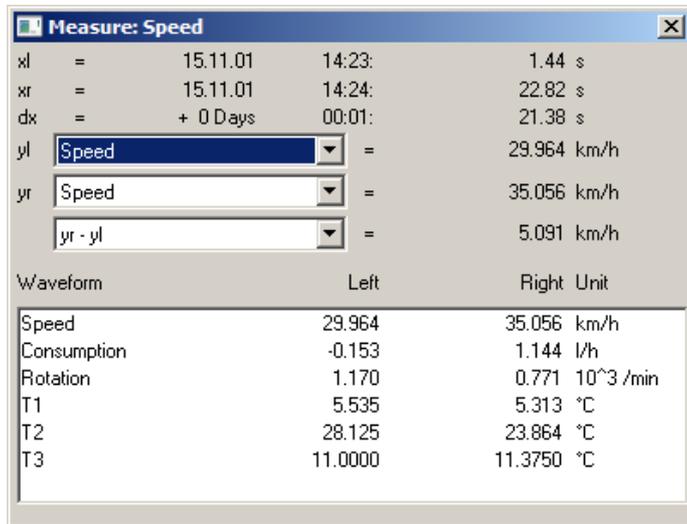
More about [Transfer options](#)^[1087]

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.

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 Channels List, 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. If you wish to close the list mode, select the same menu item again. 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.

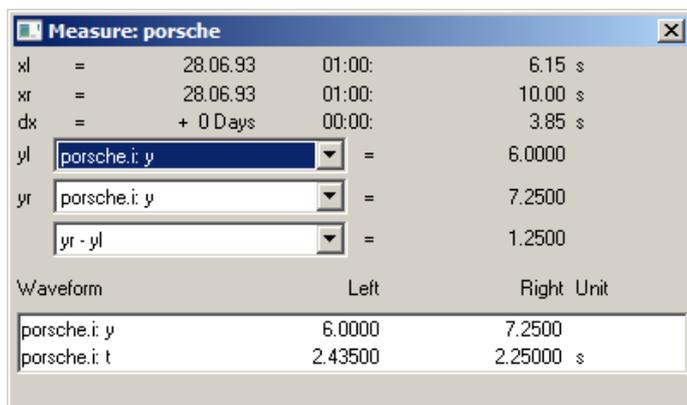
Remarks

The column *right* and *left* above the list is assigned to the measurement cursor which is controlled by the corresponding mouse button.

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.



Reprocessing signals...

This menu item allows signals to be reprocessed, for example, to undergo smoothing or peak clipping. For a detailed description, see the separate chapter [Reprocessing signals](#)^[1020].

Place marker with left click; Place marker with right click

Set a marker at the measurement cursor positions. Subsequently, a dialog for [marker settings](#)^[1039] appears.



Reference

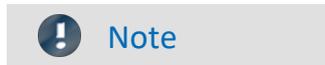
More about "[Markers](#)"^[1028] and "[Marker Definition](#)"^[1039]

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.



Note

As of imc FAMOS 7.0 / imc STUDIO 5.0R3, this function is no longer shown.

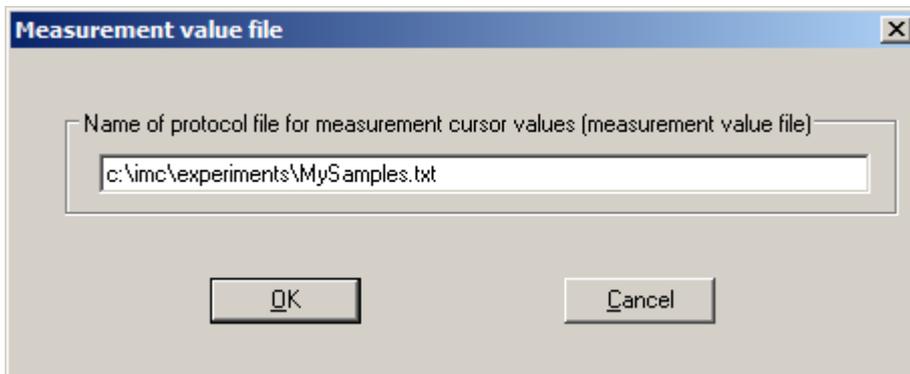
By means of an entry in the Registry, it can be made visible again:

```
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*.

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.

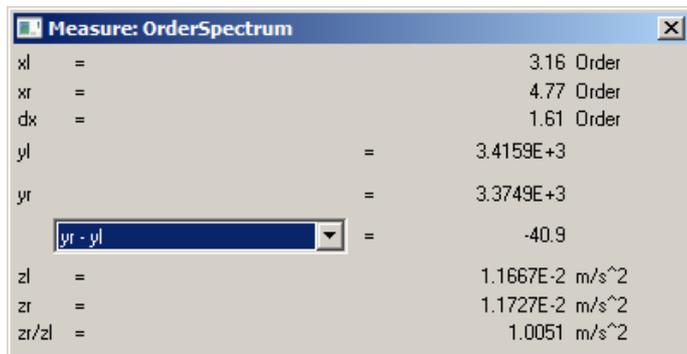


Remarks

- The *Measure* mode (i. e. Measure window is open) is ended whenever a major change in the display of the curve window is made, e.g. when numerical value display mode is selected.
- The menu option *Measure* in the curve window is checkmarked if the curve window is in the *Measure* mode. If the checkmark is removed, the measurement value window is closed.
- 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.

13.5.6.3.3 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.



Parameter	Value	Unit
xl	3.16	Order
xr	4.77	Order
dx	1.61	Order
yl	3.4159E+3	
yr	3.3749E+3	
yr - yl	-40.9	
zl	1.1667E-2	m/s ²
zr	1.1727E-2	m/s ²
zr/zl	1.0051	m/s ²

13.5.6.3.4 Reprocessing signals

Function

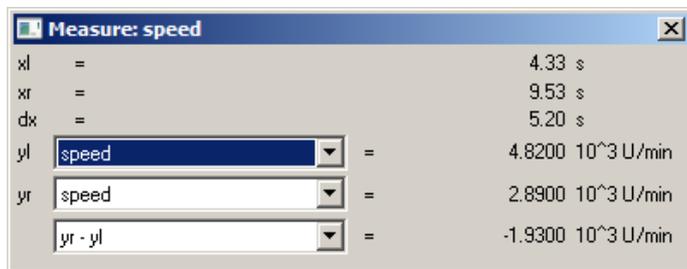
Here it is possible to edit, for instance, to smooth clip peaks from, simple signals. Thus, invalid readings can be removed from a signal and replaced with plausible ones, or distorted readings can be corrected, for example, by eliminating offsets or drifts.

These functionalities are only available off-line (in imc FAMOS).

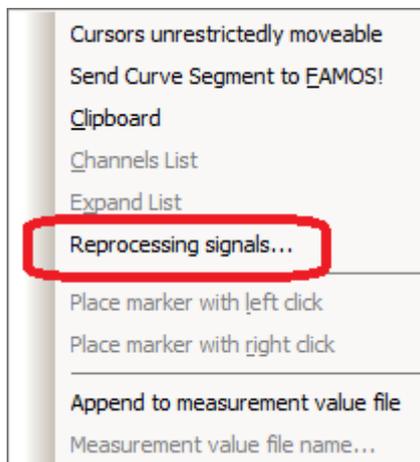
Mouse Operation

A channel is displayed in a curve window. A region is staked out using the measurement cursors, e.g. the region around a signal peak representing interference to be deleted. 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.

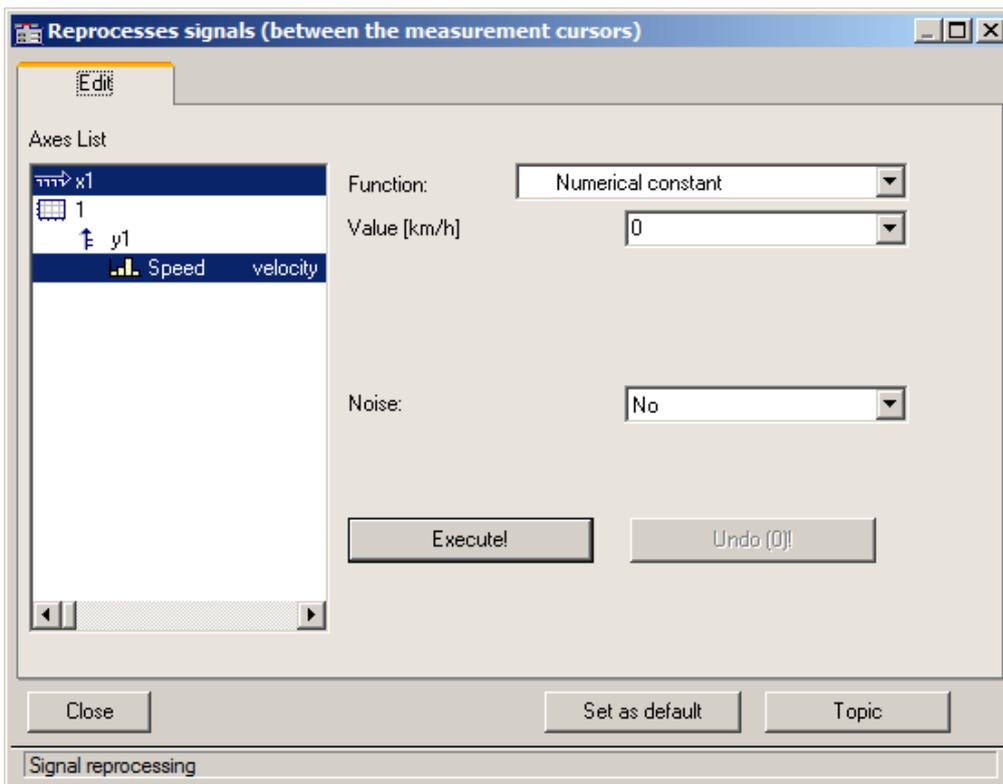
First, the Measurement value window is opened:



Next, this window's context menu is opened: move the mouse pointer over the measurement value window and right-click the mouse; the menu illustrated below then appears:



Select *Reprocessing signals....* The following dialog is called:



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 $-10V .. +10V$ 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. So if you wish to reprocess another curve window's signal also, first close the *Reprocessing signal* dialog in the first window.
- 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](#)"^[1008] for more information on the operation of the *Set as default* button.
- If data in imc FAMOS are edited by means of the *Reprocessing signals..* dialog, the display in imc FAMOS' data editor is not updated. The Data Editor should be closed during this time; it can be called again afterwards.

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](#)^[1023] and [Boundary interval](#)^[1024].

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.

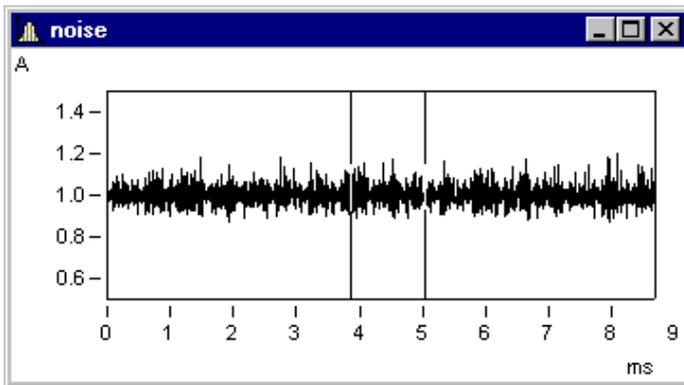
Edit signal	Description
Clip signal above	All readings in the region delineated are clipped to keep the curve below a specified maximum value.
Clip signal below	All readings in the region delineated are clipped to keep the curve above a specified maximum value.
Edit trend	Description
Add a numerical 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

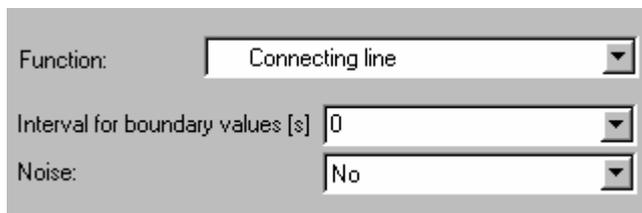
Many functions use the boundary values at the edges of the delineated region; for instance, all values within the delineated region are bridged by a connecting line or spline.

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):



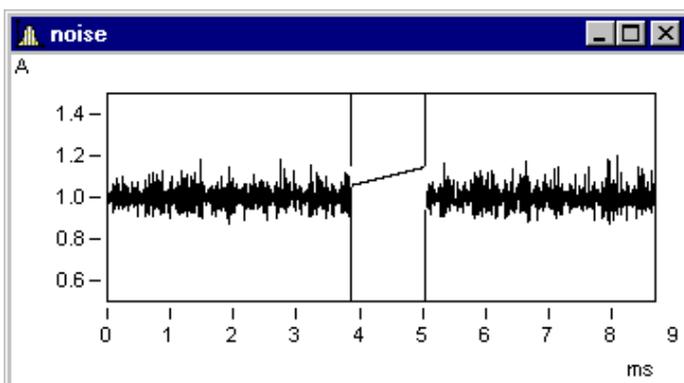
In the dialog *Reprocessing signals...*, the following settings are made.



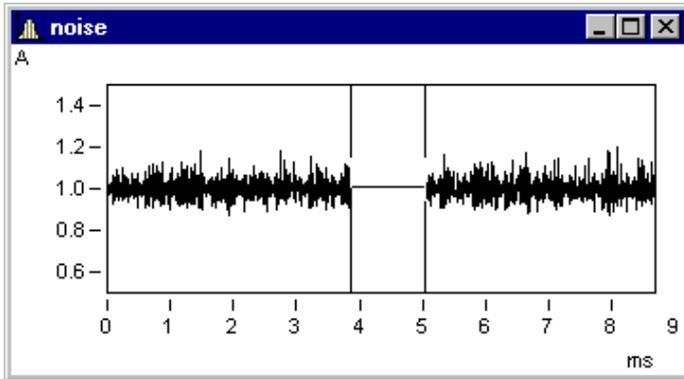
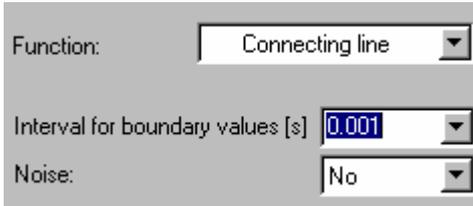
The dialog box shows three settings:

- Function: Connecting line
- Interval for boundary values [s]: 0
- Noise: No

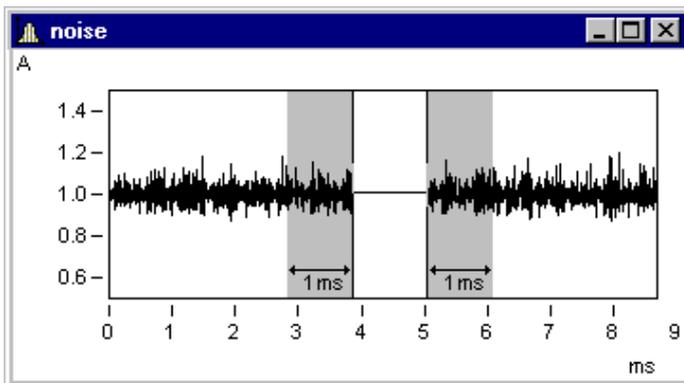
The following connecting line results:



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.



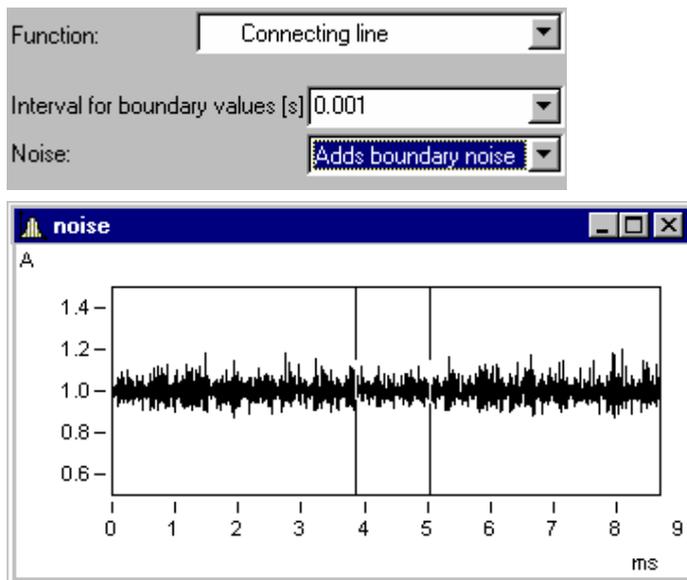
Remarks

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

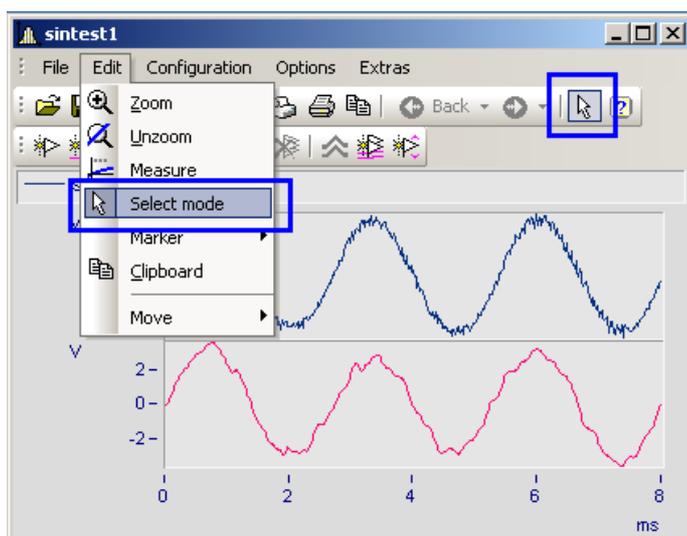
To the sample signal above, it is now possible to add noise. Toward this end, the noise in each of the boundary intervals is determined. This noise, cyclically repeated, is then added to the delineated region. Between the noise signal on the left and that on the right side, a smooth transition is accomplished 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:



13.5.6.4 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.



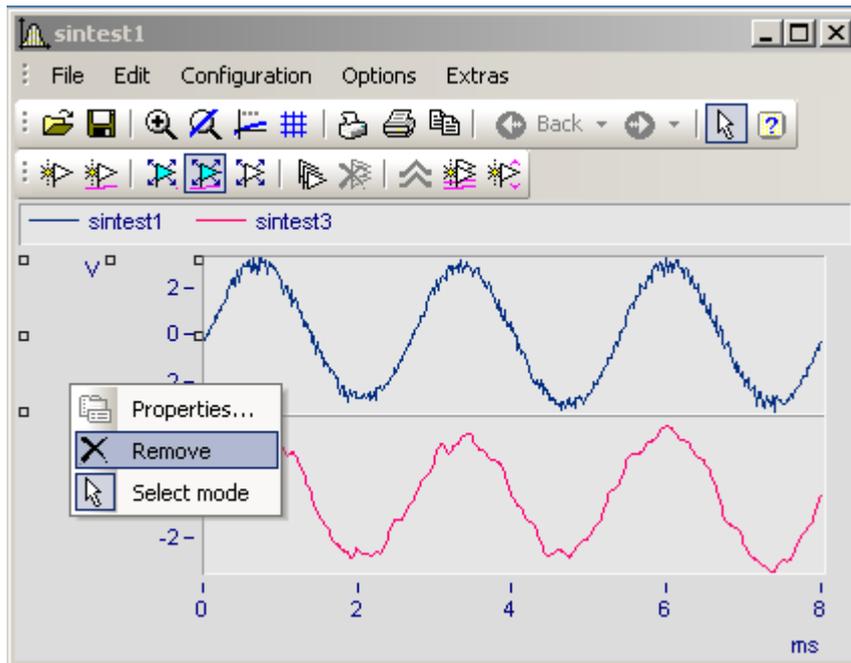
If you are in *Select mode*, the current object is highlighted. This is indicated by the presence of small empty squares at the object's edges. Any lines not selected appear dulled in color.

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.

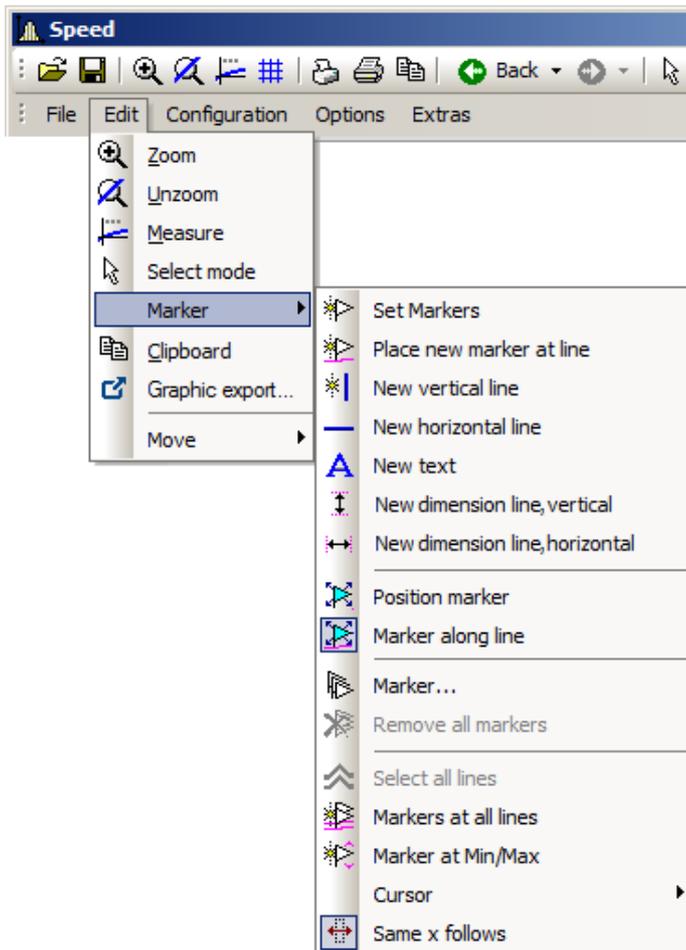
Among other things, the context menu contains the function *Delete*. This is only available if objects which can be deleted are available, such as a second y-axis. If such an object is deleted, any associated objects are also deleted. In the example, the lines in the deleted y-axis would also be deleted.

Example

You select the upper y-axis and deleted it. In the process, the sintest1 lines with which it is associated are also deleted.



13.5.6.5 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.

The image shows a software interface with a 'Defaults' menu on the left and a 'Property' table on the right. A red arrow points from the 'Standard marker' option in the menu to the 'Property' table.

Defaults Menu:

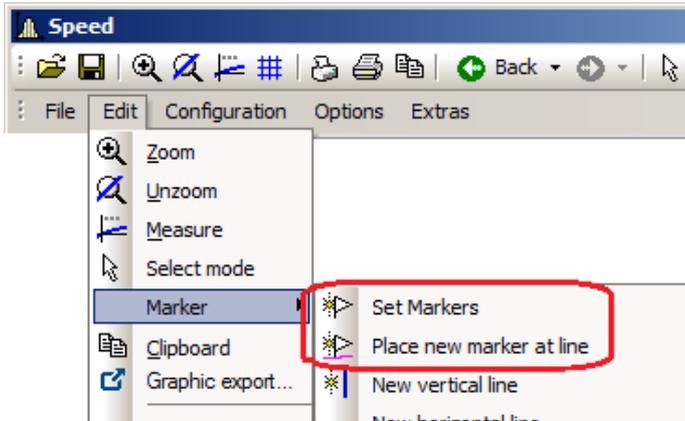
- Standard marker
- Vertical line
- Horizontal line
- Text
- Vertical dimension line
- Horizontal dimension line
- Order line
- Harmonics Cursor
- Min-Marker
- Max-Marker
- Marker at all lines, y
- Marker at all lines, x
- Marker for color map

Property Table:

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	

13.5.6.5.1 Set markers

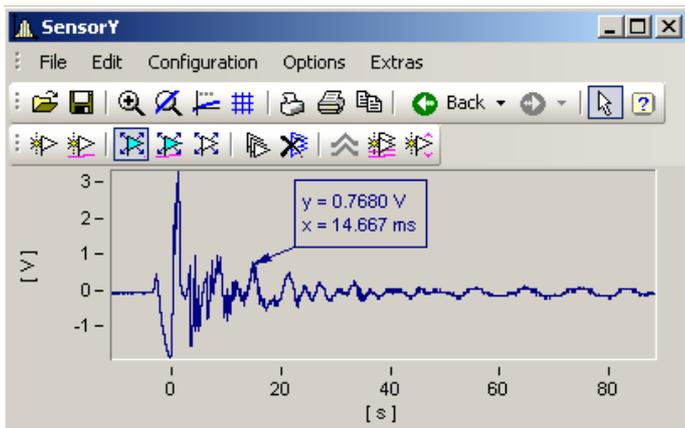
There are two ways of setting a marker, which are accessed either from the Marker toolbar or via the menu item *Edit/Marker*.



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](#)¹⁰³⁹.



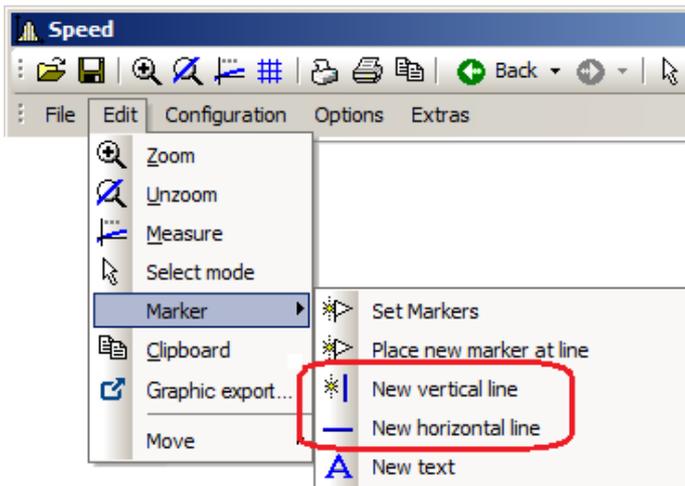
Markers can be placed anywhere in the curve window. Zoom more intensely if a marker should be placed near a line. Accuracy depends on the resolution. To place a marker exactly at a certain point in the curve, select the function *Place new marker at line*.

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.

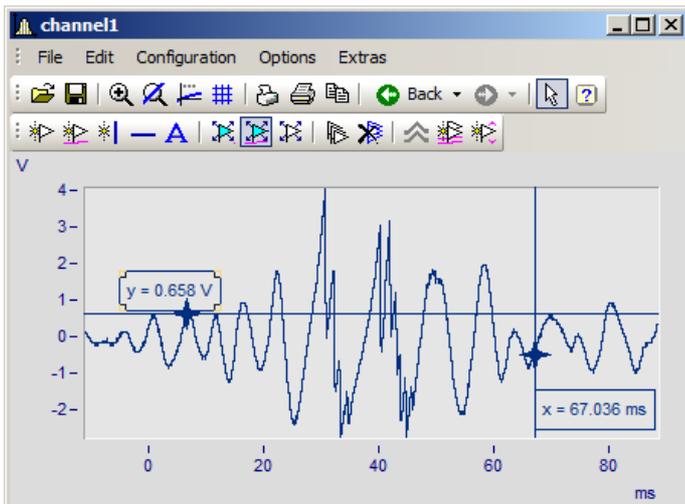
Both modes can be exited by a click of the right mouse button.

13.5.6.5.2 New line

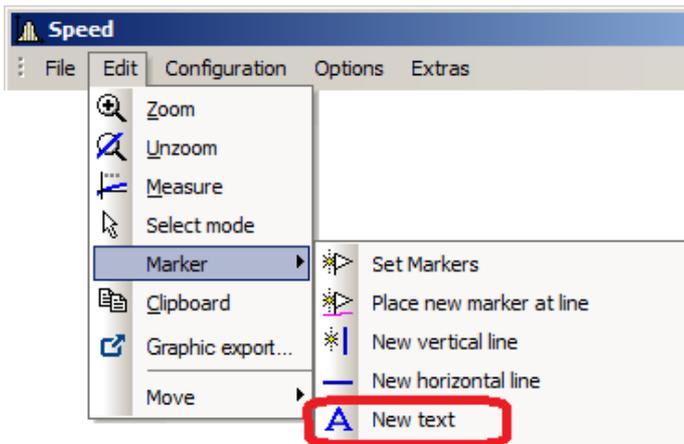


Vertical line new and Horizontal line new

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.



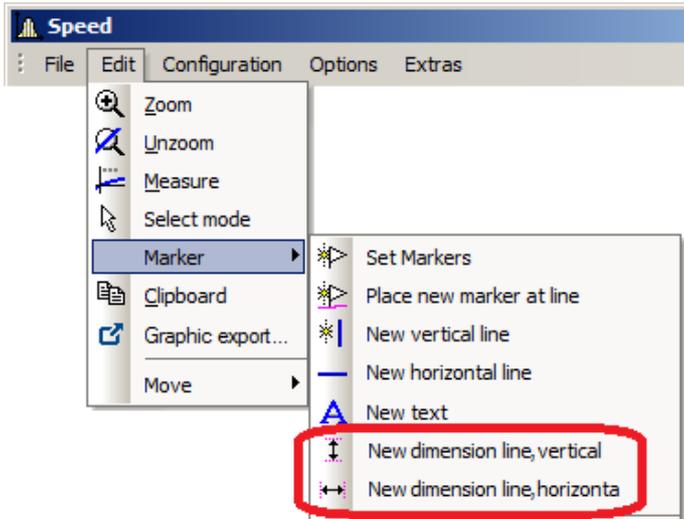
13.5.6.5.3 New text



New text

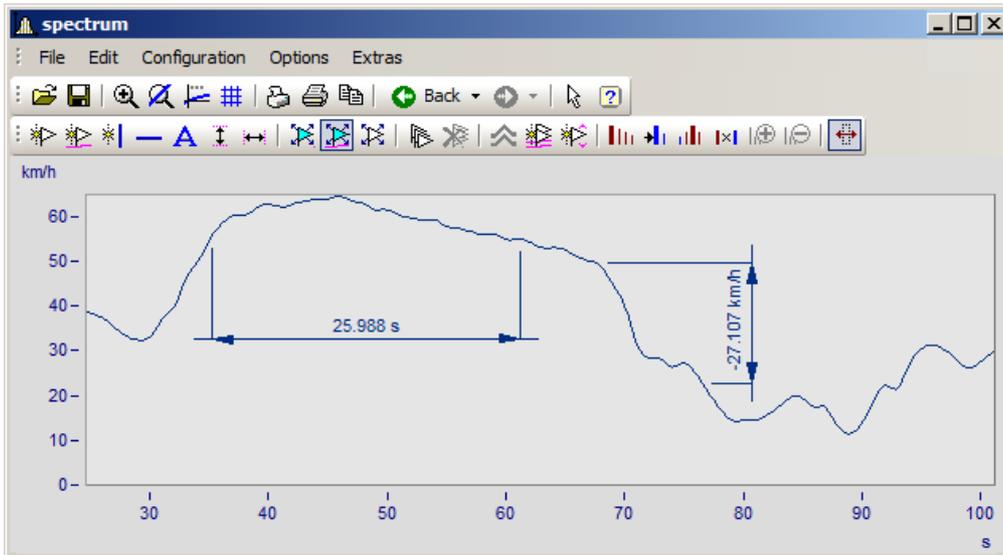
A text box without an arrow can be placed freely. Otherwise, the same settings are available as for markers.

13.5.6.5.4 New dimension lines

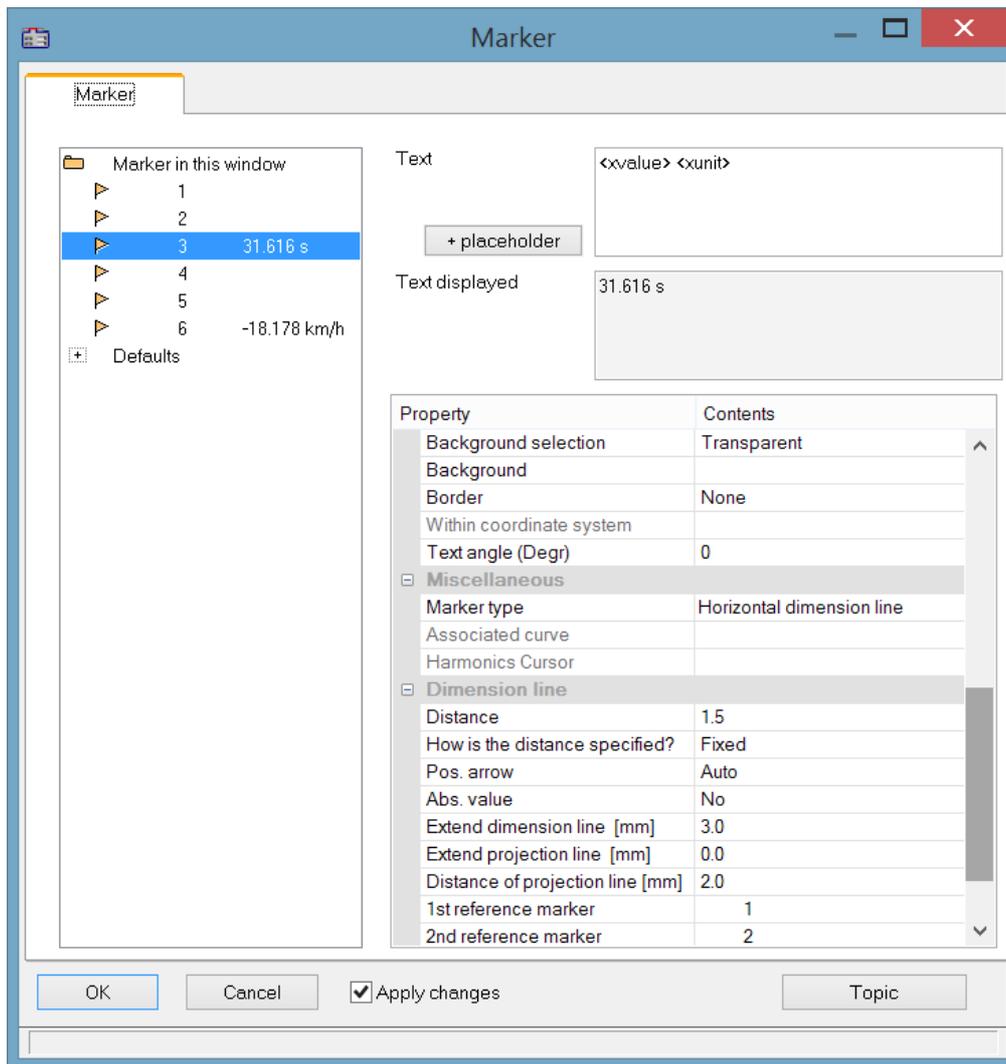


New Dimension Line: vertical and New Dimension Line: horizontal

Select this entry to create a vertical or horizontal dimension line between two points in the curve window where you click the mouse.



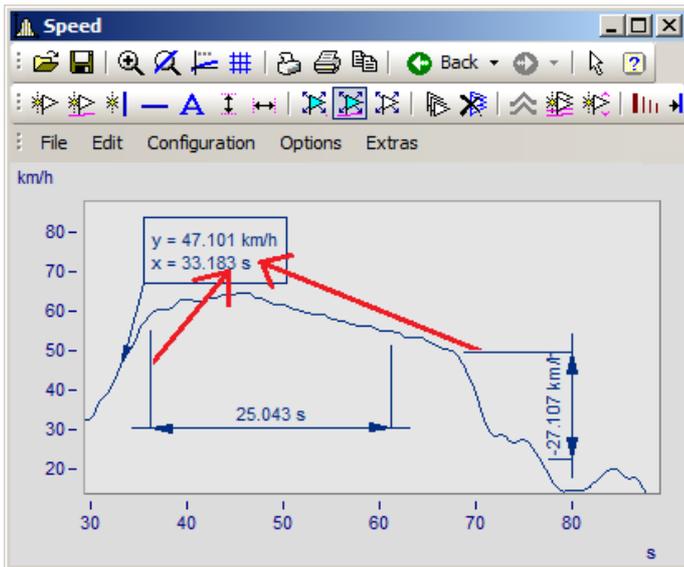
You can change the width of the dimension lines using drag&drop. The placement of the legend as well as of the line is positioned by drag&drop. Making settings for the markers is done as shown below:



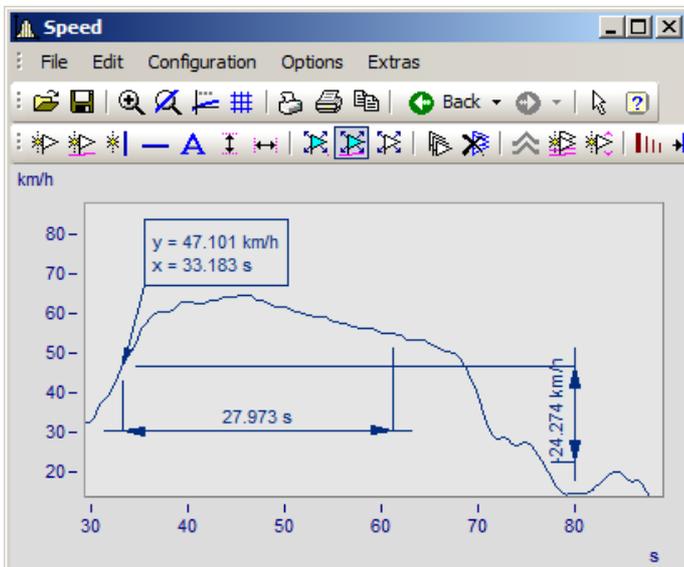
Position

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.

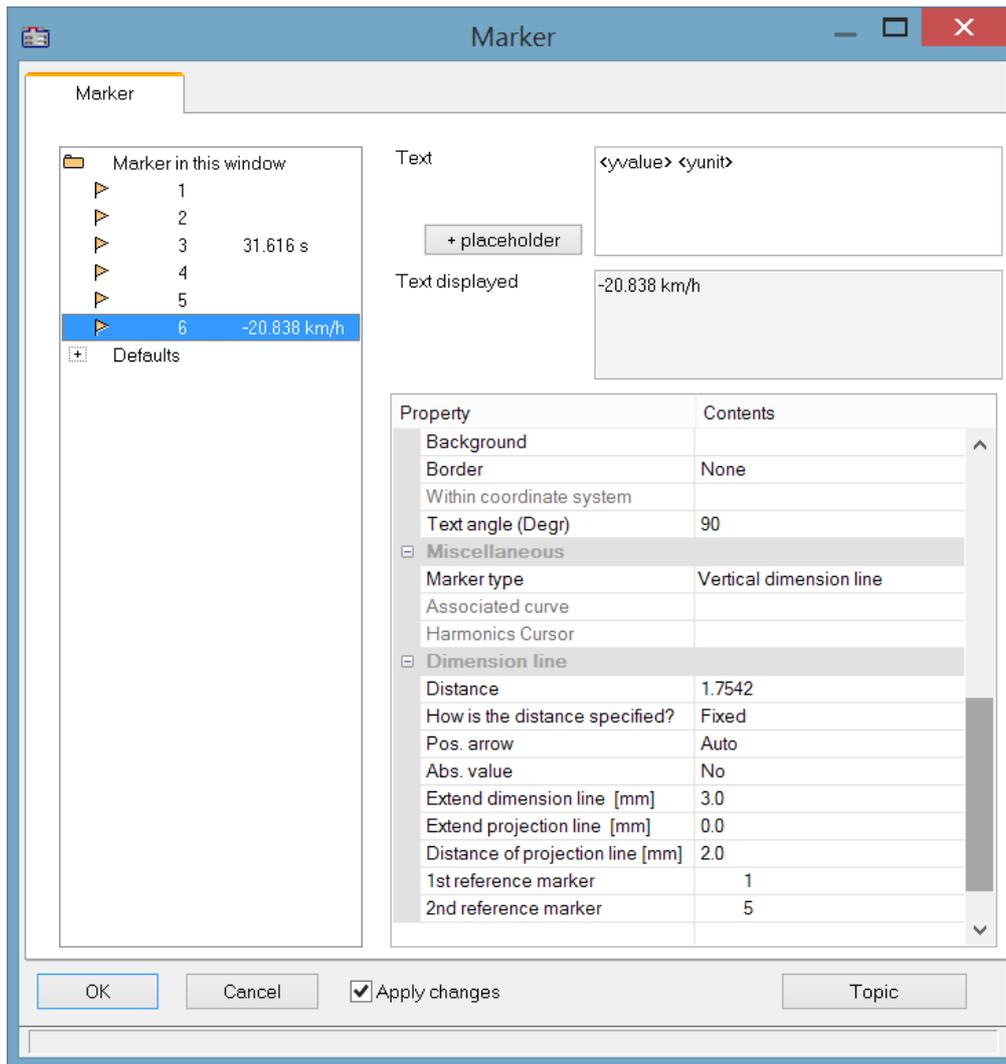
The example below shows how two dimension lines are linked to a marker. When the marker is subsequently moved, the dimension lines are adapted automatically.



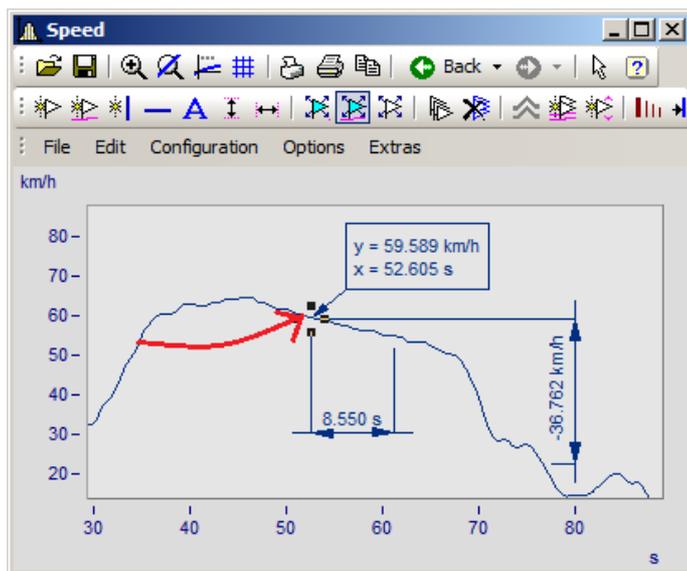
The marker and dimension lines are not yet linked. The extension lines are dragged to the marker.



Now the marker and the dimension lines are linked with the marker.

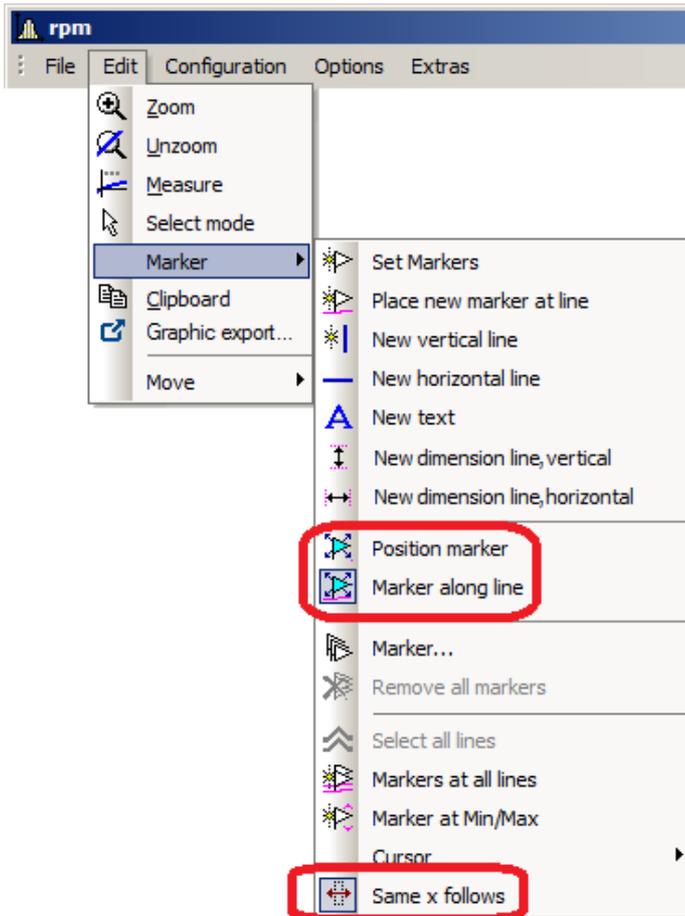


In the definitions dialog, the reference marker for the dimension lines is now linked with the marker.

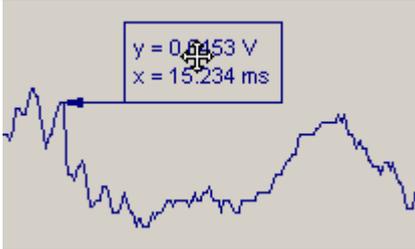
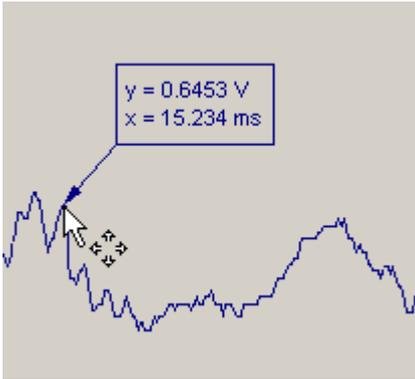


Now, when the marker is moved, the dimension lines are updated.

13.5.6.5.5 Move markers



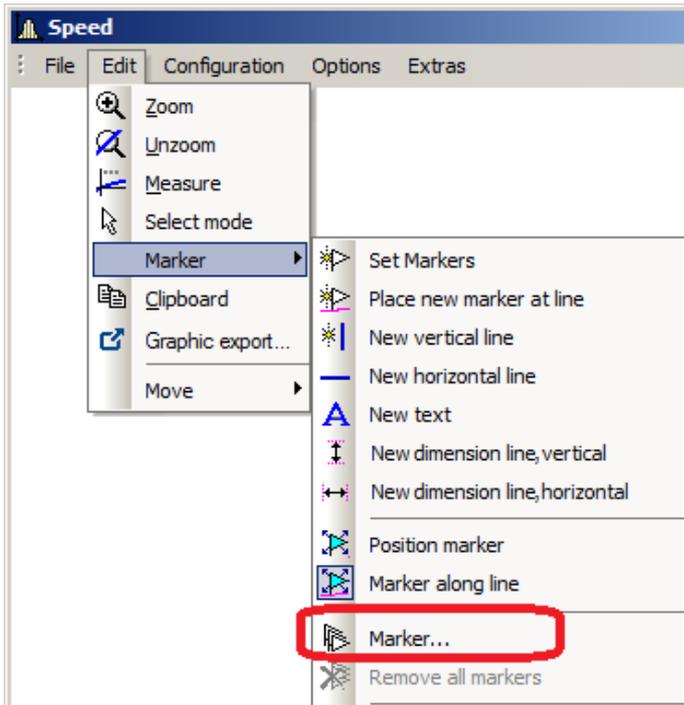
There are different ways to change the marker position:

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 closest to the cursor's location.
Move label	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 marker	Existing markers can be moved subsequently. To do this, use Drag&Drop to move the arrowhead point. 
 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

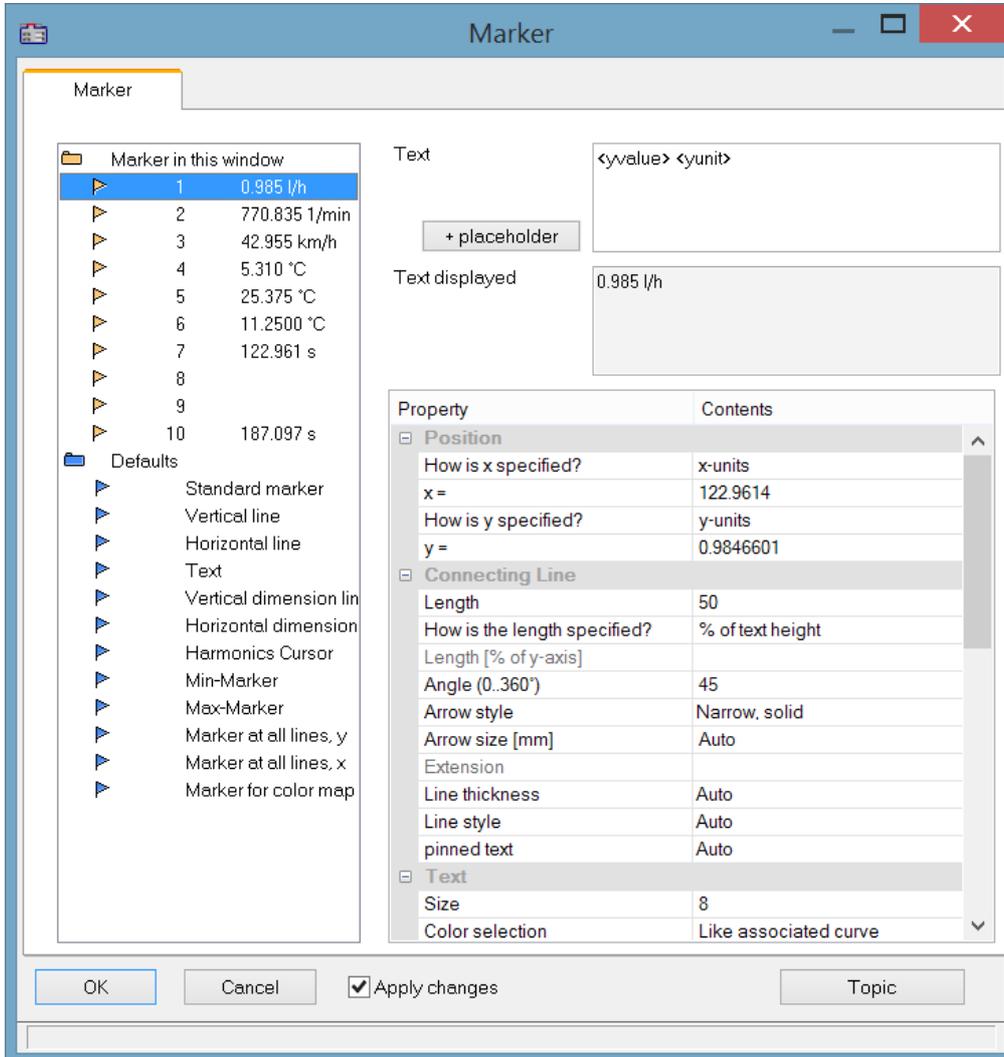
13.5.6.5.6 Marker definition



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	<p>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.</p> <p>For a length of 0, no line is drawn.</p> <p>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.</p>
Angle: (0...360°)	<p>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.</p>
Arrow	<p>The line between marker and text may have an arrowhead at the end of the marker. The size and type can be selected:</p> <ul style="list-style-type: none"> • none • wide • narrow • wide, solid • narrow, solid • large • large, solid • circle • Point • Diagonal dash • Star • Default <p>The size of the arrow is oriented to the size of the symbol, see also "Clipboard settings".</p>
Line thickness, Line style	<p>Settings like for Lines. At printout, the line appears in the thickness of the cursor-line, see Clipboard Settings.</p>
Text corner	<p>Position of the text box (upper right, upper left, lower right, lower left)</p>

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](#)¹⁰⁹⁷ 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

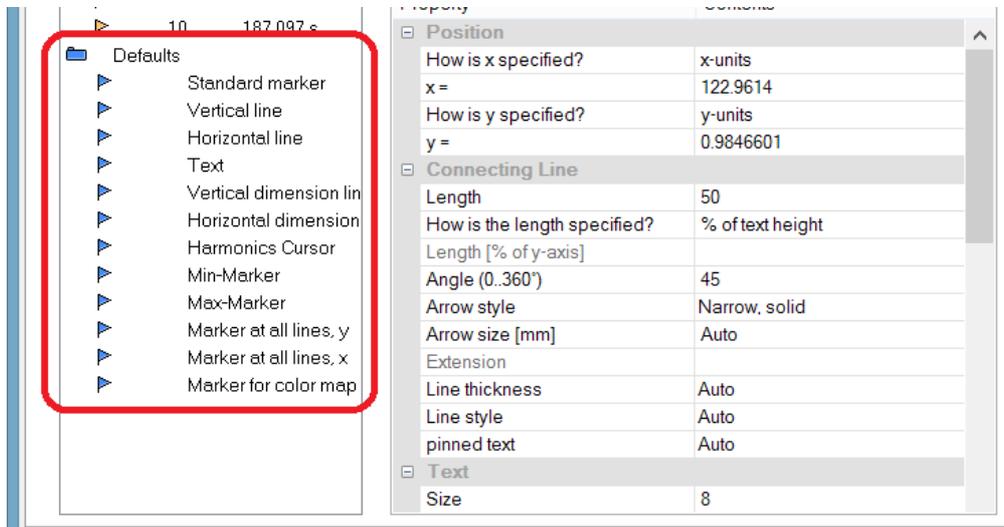
Settings	Description
Size	The size of the text can be defined in points. In general, a 12-point font can be read quite easily. The font specified in "Clipboard settings..." is the font used for printing. A true-type font should be selected. The standard font set for the curve window is also used for the monitor screen (see " Presettings " ^[1087]).
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: <ul style="list-style-type: none"> • <i>none, simple, pointed (trapezoid-shaped), double border</i>
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. <div data-bbox="454 840 778 990" data-label="Image"> </div>
Angle text (Degr)	Angle of the text box up to ± 90 degrees. <div data-bbox="454 1052 695 1249" data-label="Image"> </div>

Delete

All lines selected in the marker list are deleted.

Edit defaults

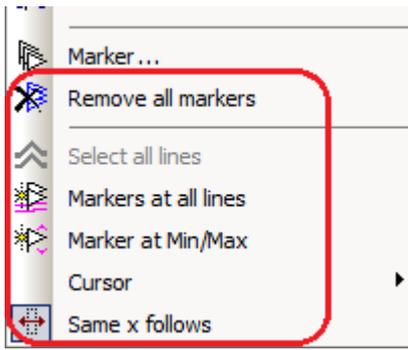
The **Defaults** for the *Marker types* are listed at left links and can be edited at any time using the Properties table.



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.

13.5.6.5.7 Extra marker functions

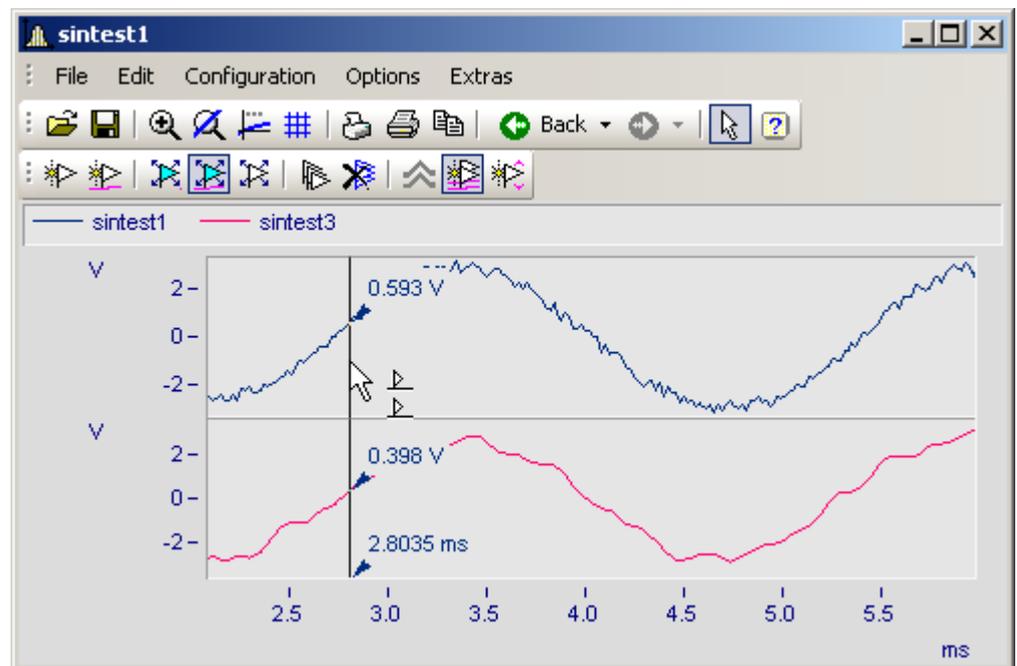


There are four additional marker functions available:

Menu item	Description
Remove all markers	With this function, all markers set in the curve window can be deleted at once.
Select all 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 and 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](#)¹¹⁰³⁷¹".

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.

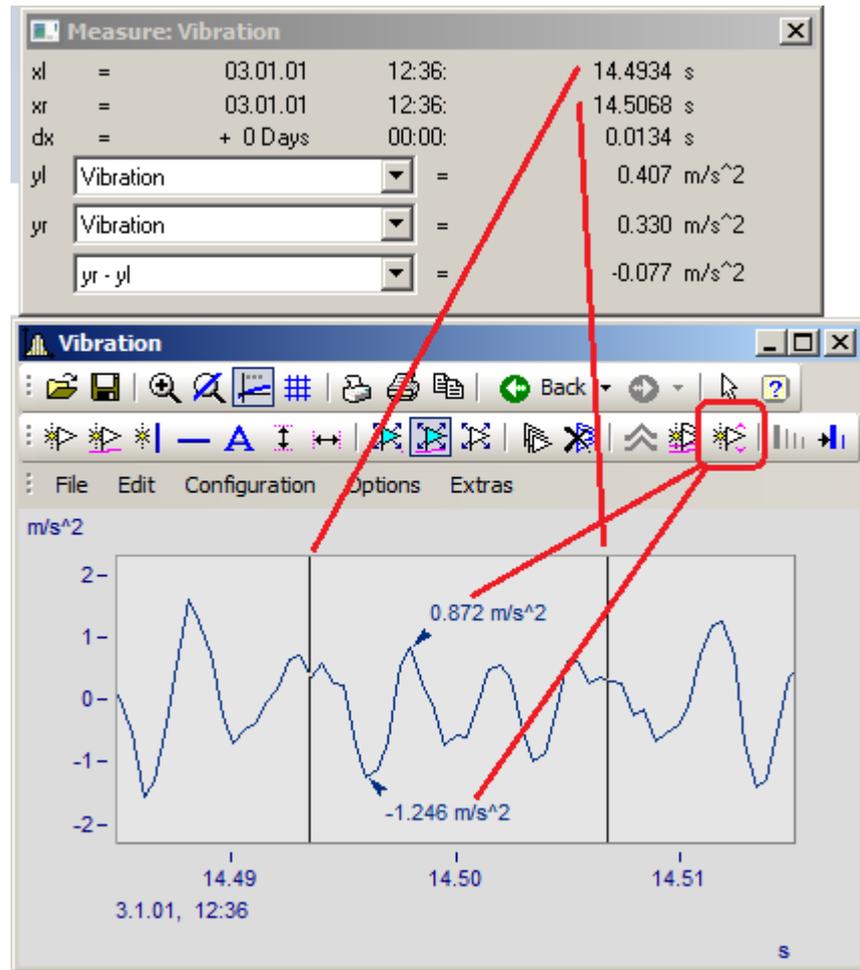
Menu item	Description
-----------	-------------

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.

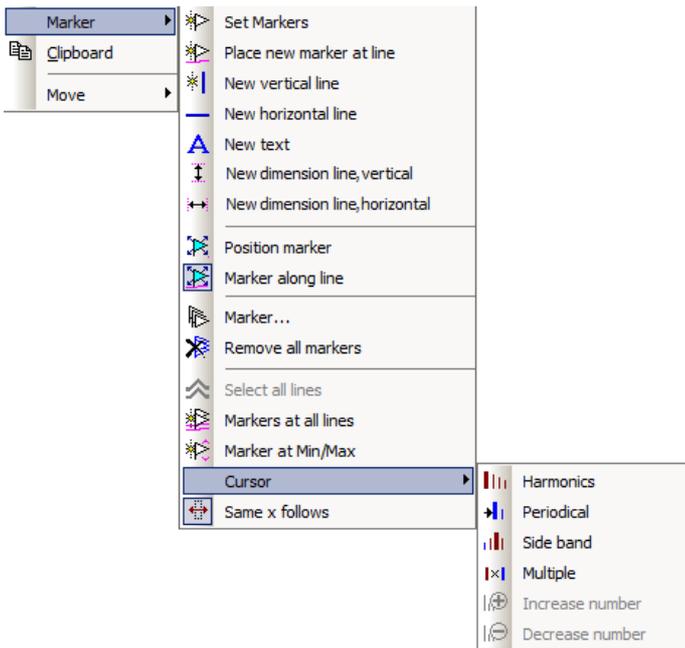
The function only affects the selected time segment. For this reason, zoom on to the region before setting the marker.

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

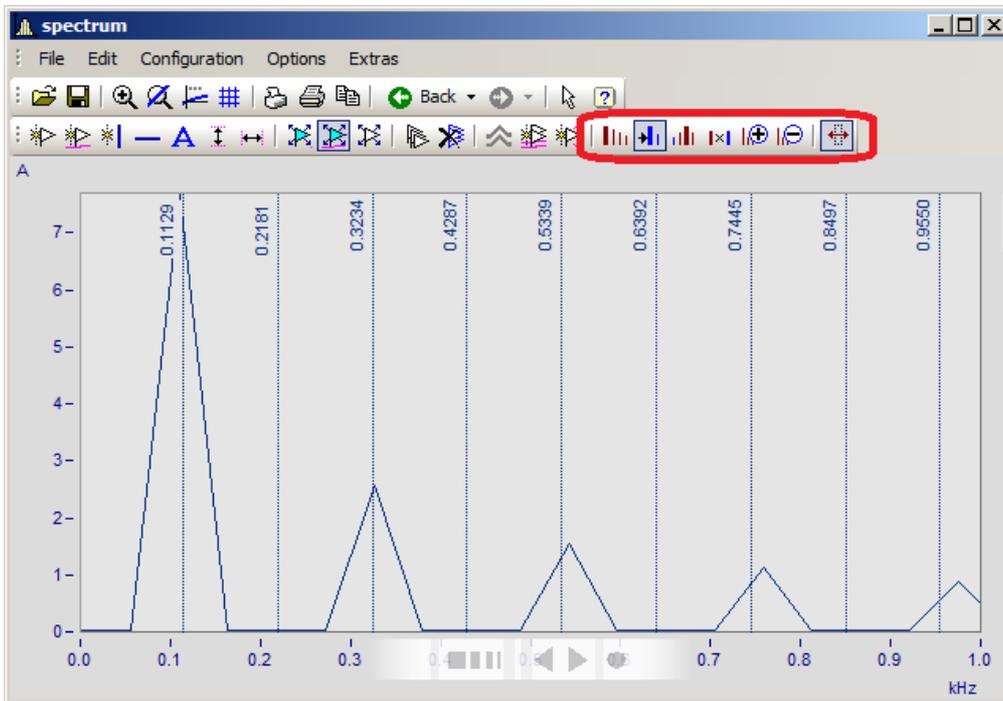


13.5.6.5.8 Harmonics cursor



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.

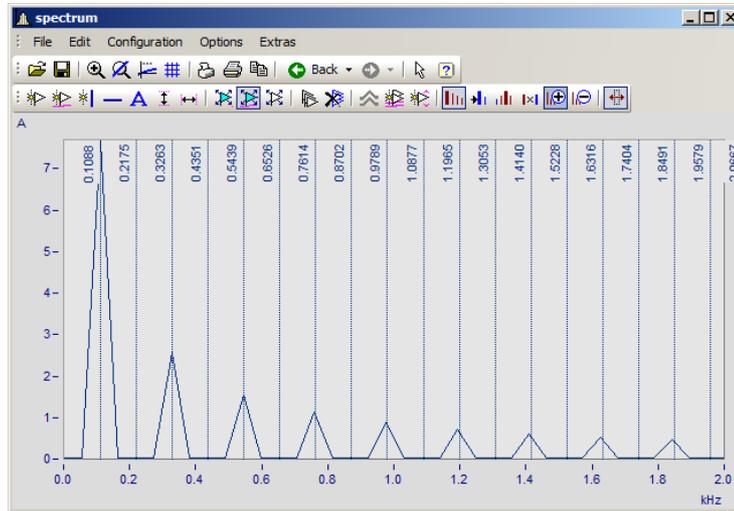


The following types are available:

Menu item	Description
-----------	-------------

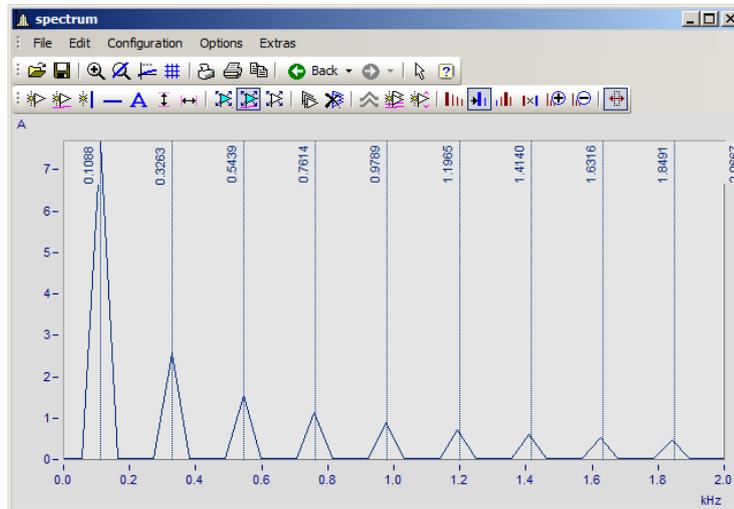
Fundamental oscillations with harmonics. 10 equally spaced markers are inserted. Place the fundamental oscillation. The subsequent harmonics follow the fundamental oscillation by the respective multiple. The fine adjustment is performed by moving the upper harmonics.

 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.

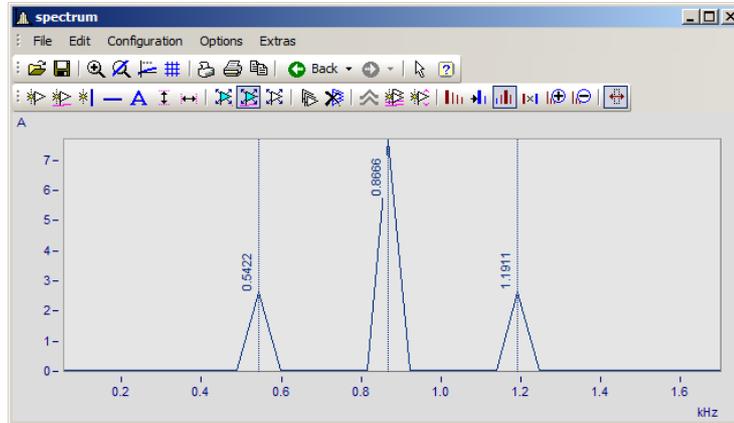
 Periodic



Menu item	Description
-----------	-------------

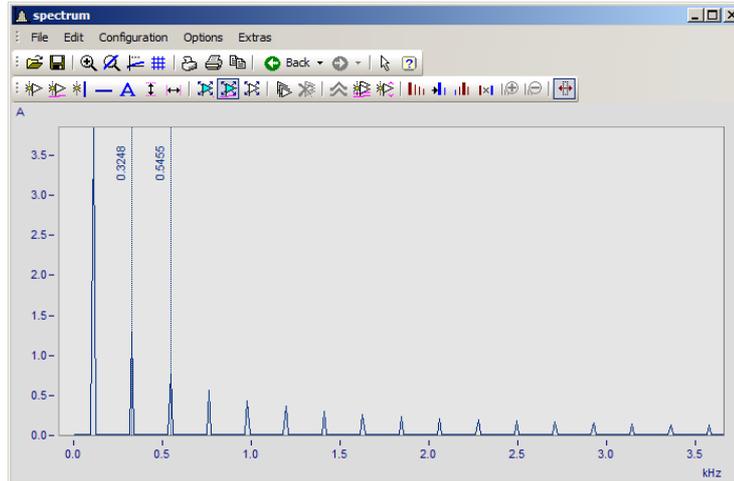
Fundamental oscillation with side bands. 10 equally spaced markers are inserted. Position the fundamental oscillation. The harmonics which follow the fundamental oscillation by the respective multiple. Fine adjustment is performed by moving the upper harmonics.

 Side band

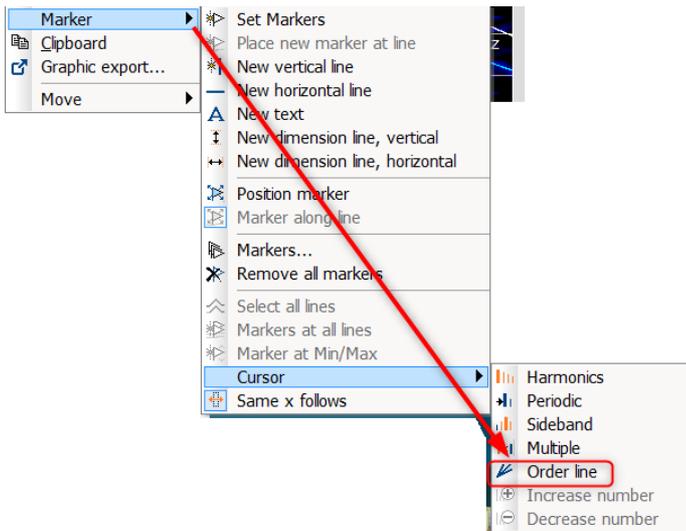


Distances at a fixed ratio. Two frequency lines which can be freely positioned.

 Multiple



13.5.6.5.9 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.

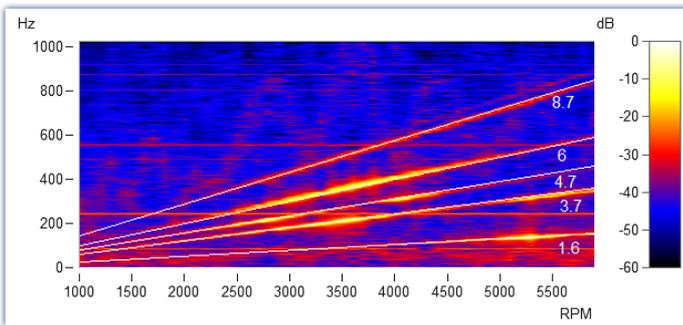
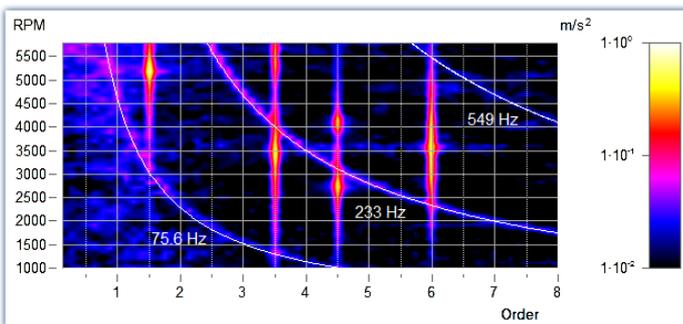


Fig. Amplitude spectrum with frequency over RPMs

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

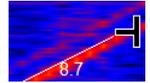
Hyperbola in the order spectrum

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](#)¹⁰²⁸. 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.

13.5.6.6 Export

13.5.6.6.1 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..](#)^[1072], the graphic can be saved as a file in an image or PDF format. The [default settings](#)^[1086] 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.

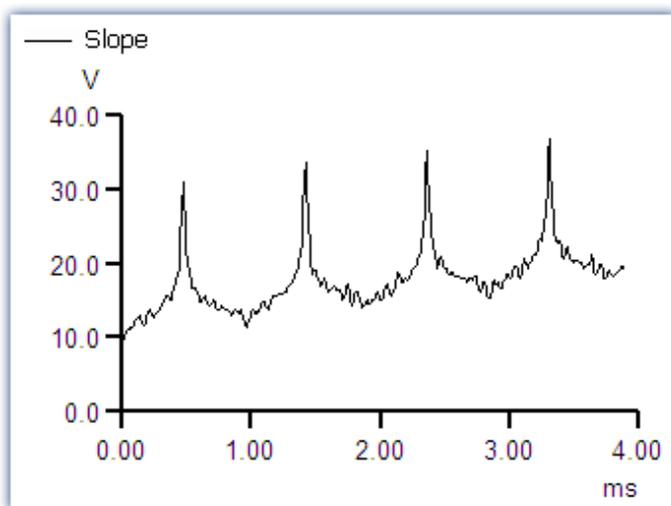
When one application sends graphics to the clipboard, another application can read and retrieve these graphics from the clipboard. In general, data are not damaged when they are read from the clipboard, which means that applications can retrieve the same graph as often as desired. Existing graphics are 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](#)"¹⁰⁶⁹). 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](#)'¹⁰⁶⁷ 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.

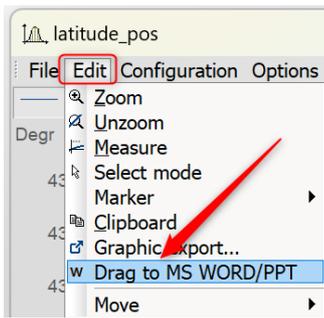


Reference

Clipboard Settings

Section '[Clipboard Settings](#)'¹⁰⁷⁴

13.5.6.6.2 To MS WORD (OLE)



Users with an **imc FAMOS Enterprise** or **Runtime** license can embed the curve window into a WORD or PowerPoint document.

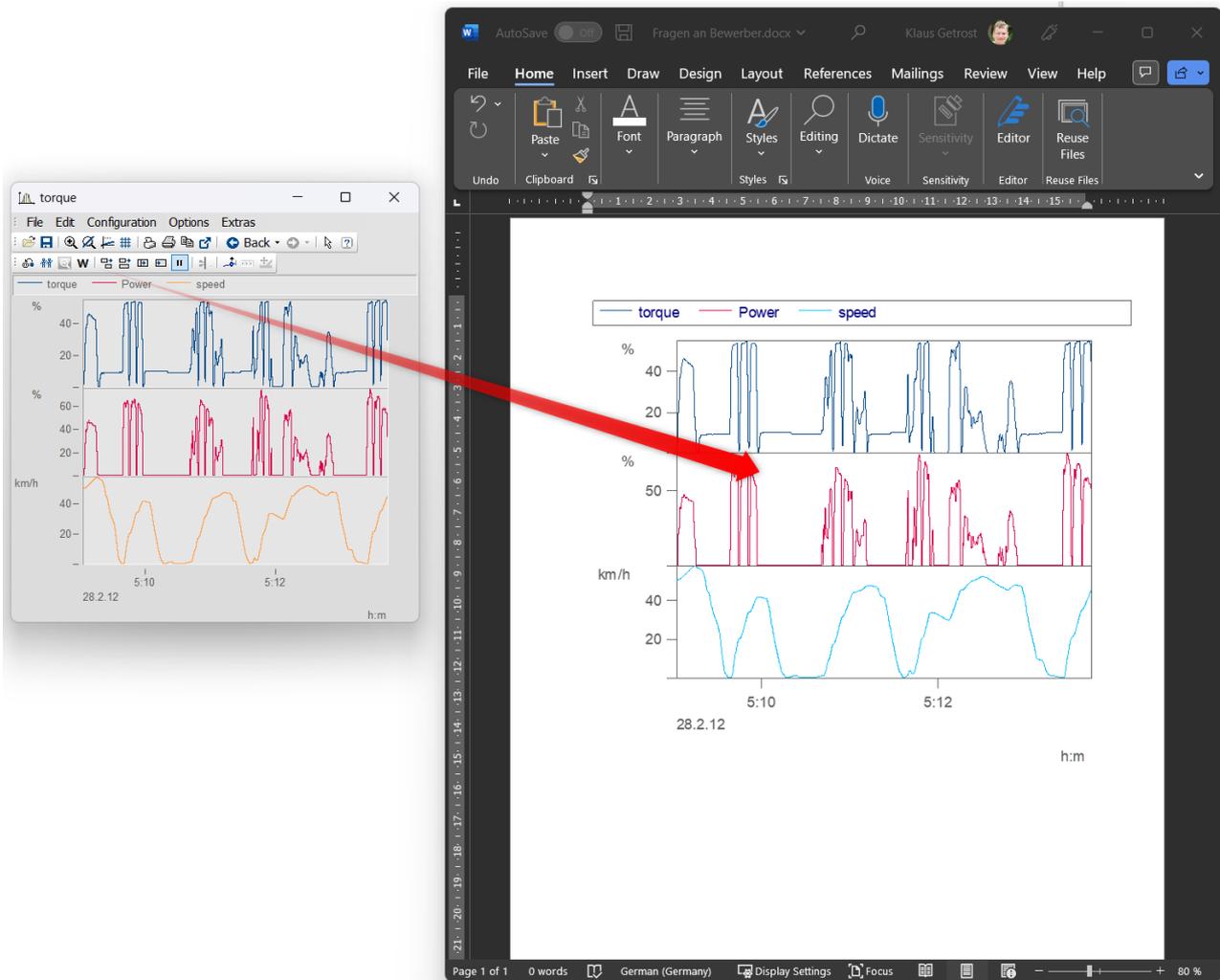
The **transfer**¹⁰⁵⁴ is done by **Drag&Drop** via the menu of the curve window or its toolbar.

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**¹⁰⁵⁵ if at least an **imc FAMOS reader** is installed. Otherwise, the curve window is displayed as an image and cannot be edited further.

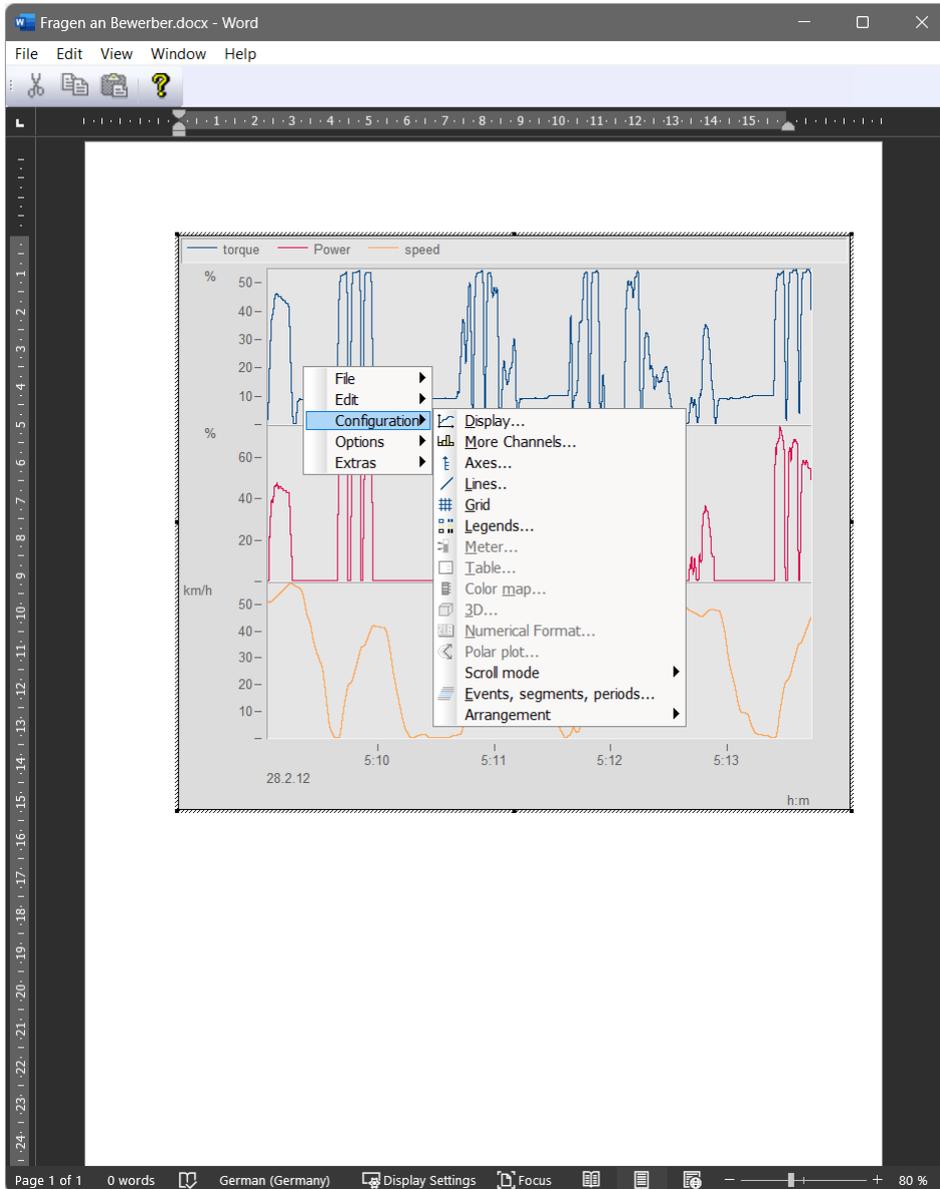
Transfer

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** :



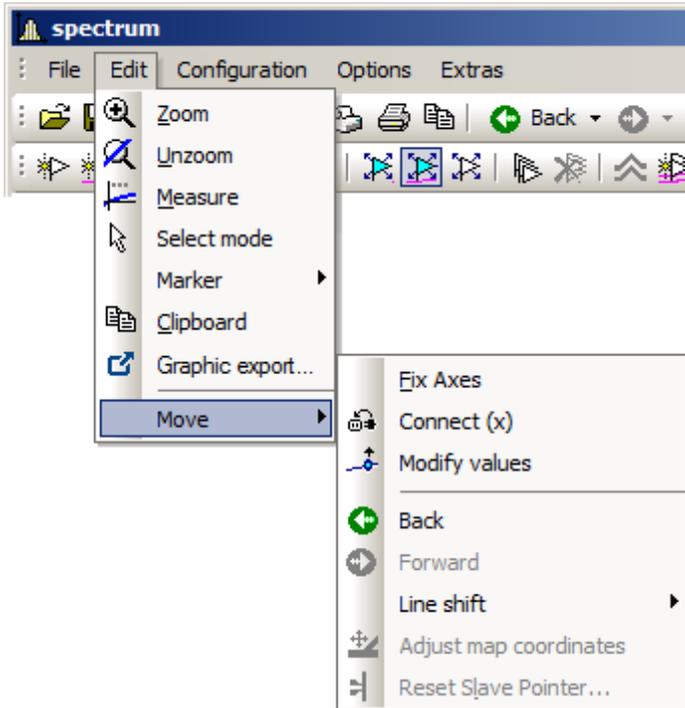
Edit

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.



13.5.6.7 Move

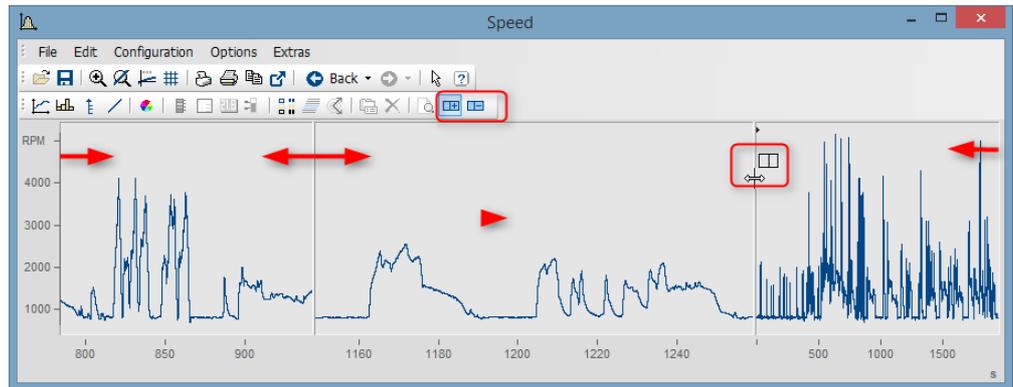
The menu item *Move* contains various curve window functions for working in the curve window:



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 ¹⁰⁵⁸ .
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 ¹⁰⁶⁷ .
Reset Slave Pointer...	For resetting the slave pointer. For more on this topic, see the section Bar meter ⁸⁷⁹ .

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

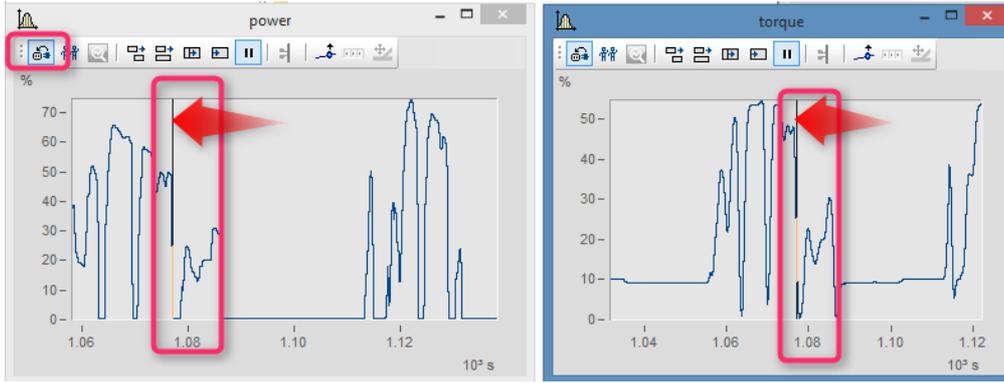


Split-Mode

13.5.6.7.1 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* ¹⁰⁸¹.

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).

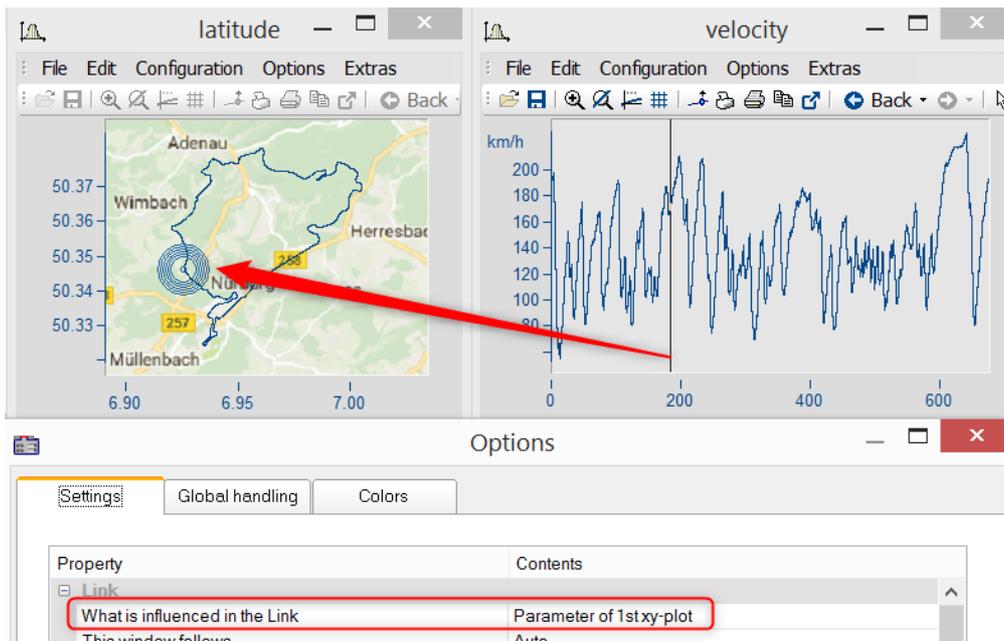
To undo an x-linkage, simply click on the button again.

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.

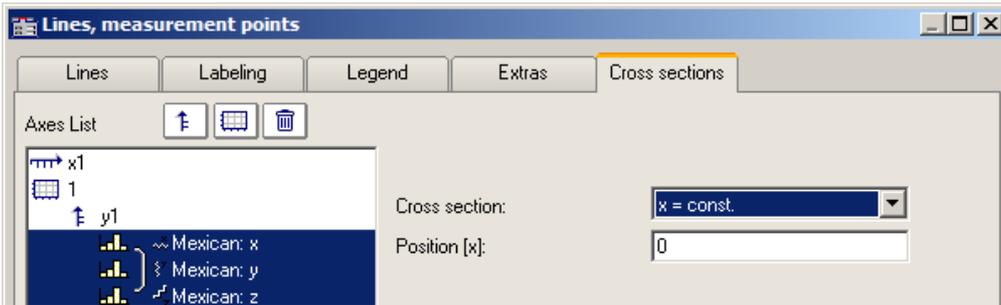
For this you have to set "*What is influenced in the link -> Parameter of 1st xy-plot*" ¹⁰⁸².



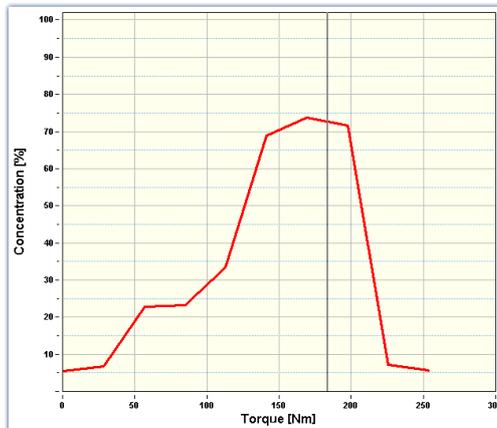
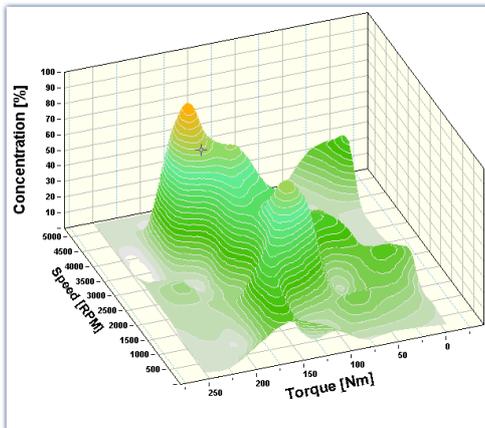
Connect with 3D

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.



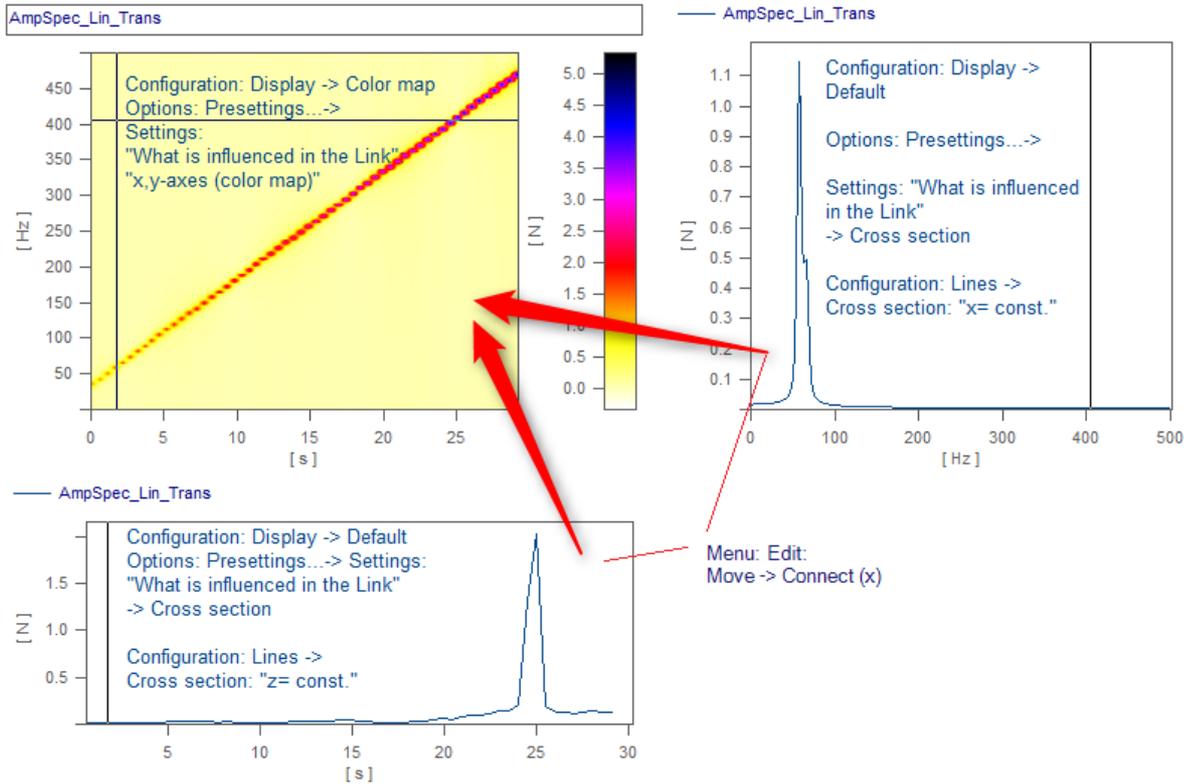


Example

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](#)¹⁰⁸¹, under *Settings*, it is necessary to set the *Link*-property "What is influenced in the Link" accordingly:

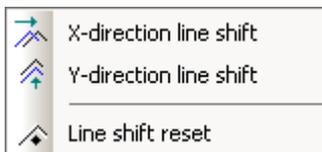
- On the color map: "x-, y-axis (color map)"
- On the right and bottom curve window: "Cross section"
- All curve windows show the same amplitude spectrum from example project "FA70 Spectral analysis"



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.

13.5.6.7.2 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.



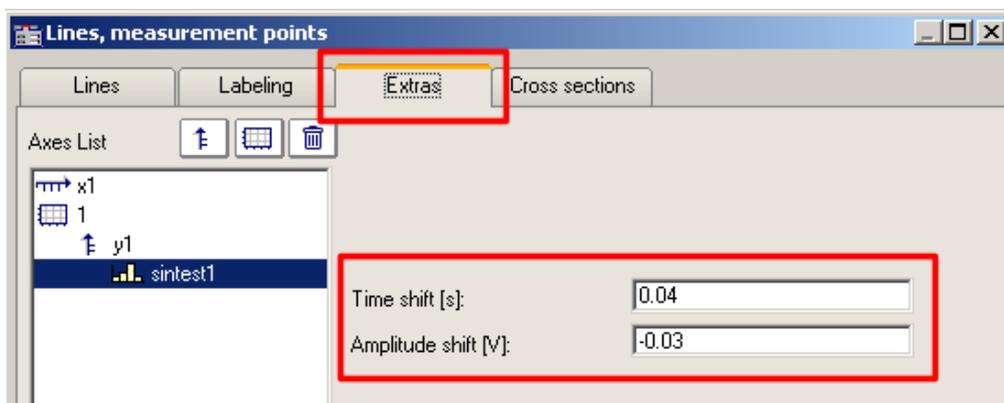
To do this, select one of the two functions either using the menu *Edit/Move/Line-Shift* or from the *Line-Shift toolbar*.

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](#)¹¹⁰⁸³¹" has been set to "yes".

13.5.7 Ribbon

13.5.7.1 "File" menu

Menu item	Description
 Load <small>1064</small>	Loads a curve window configuration from a file.
 Save as <small>1066</small>	The curve window configuration (e.g. the axis scaling) is saved in a file.
 Transfer to imc FAMOS! <small>1067</small>	The waveform in the curve window is transferred to imc FAMOS.
 Report Generator <small>1067</small>	Calls the Report Generator. The report's layout can have your own personal, creative design.
 Print <small>1067</small>	Prints out the curve window.
Printer setup <small>1069</small>	Sets up the printer used for printing out the curve window's contents.
Overview Window <small>1070</small>	Displays an overview window.
 Twin Window <small>1071</small>	Generates a new curve window with the same appearance.

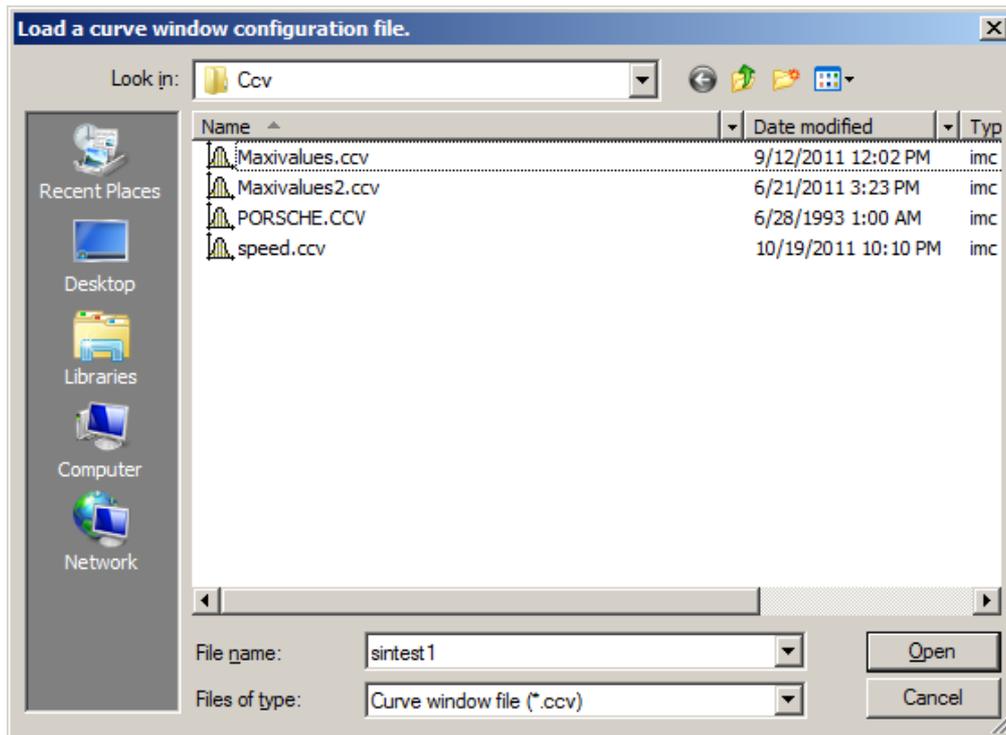
13.5.7.1.1 Load configuration

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.

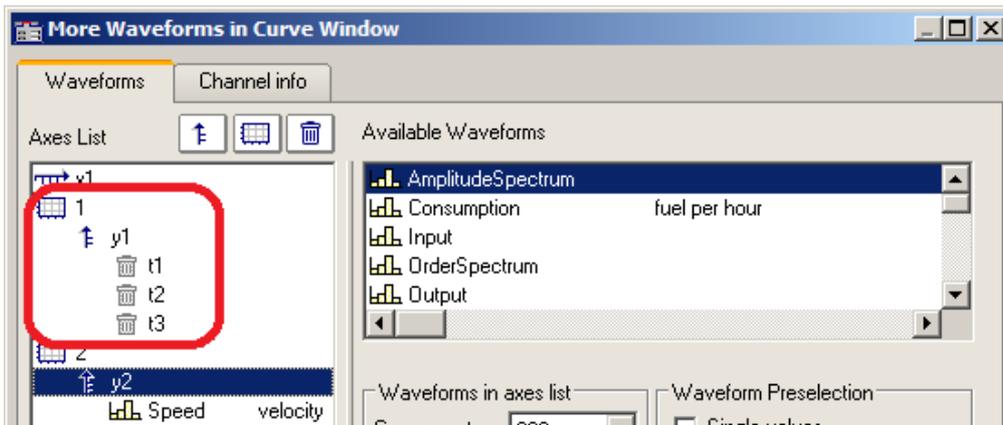


- Select file format "CCV".
- Select the desired file and end the dialog by clicking on *Open*.

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](#)^[1100], you can directly transfer a configuration from one curve window to another.

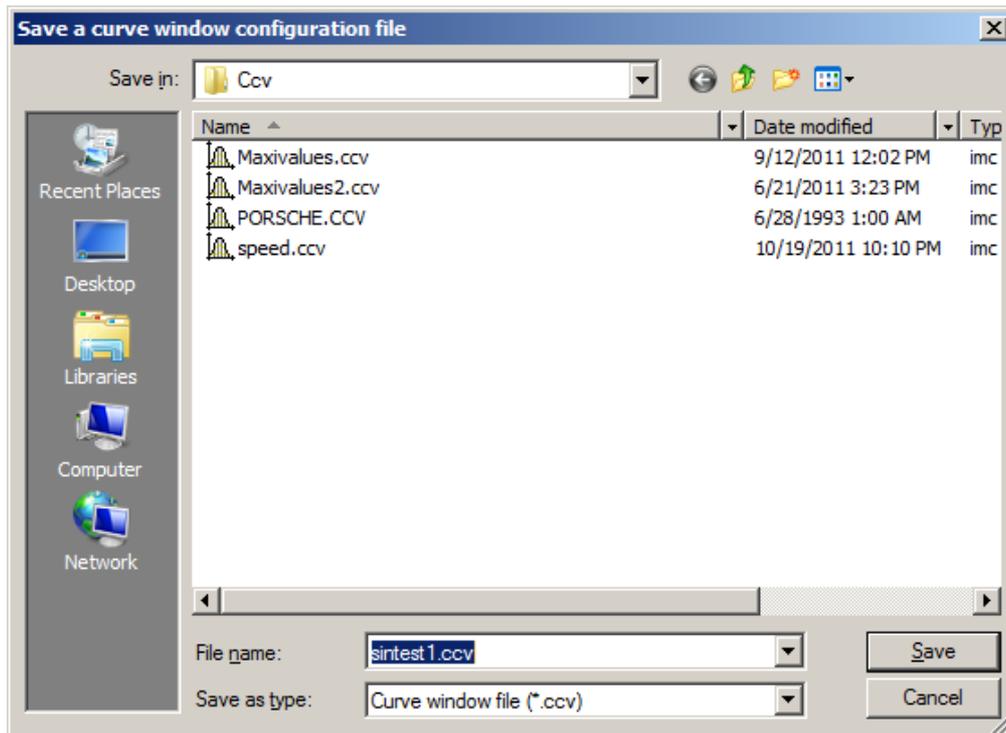
13.5.7.1.2 Save configuration

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](#) you can directly transfer a configuration from one curve window to another.

13.5.7.1.3 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](#)" 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](#)" for more information.
- Waveforms can also be transferred to imc FAMOS using [Drag&drop](#).

13.5.7.1.4 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.

13.5.7.1.5 Print

Function

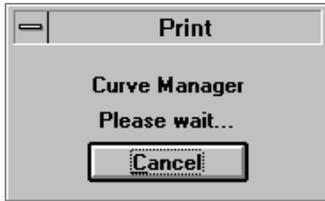
Hardcopy of a curve window can be created on any printer or plotter supported by Windows. The hardcopy is always made to an entire page. The graphics are centered on this page.

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](#)¹⁰⁶²".
- 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](#)¹⁰⁶⁷" 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](#)¹⁰⁷⁴", "[Printer Setup](#)¹⁰⁶⁹"

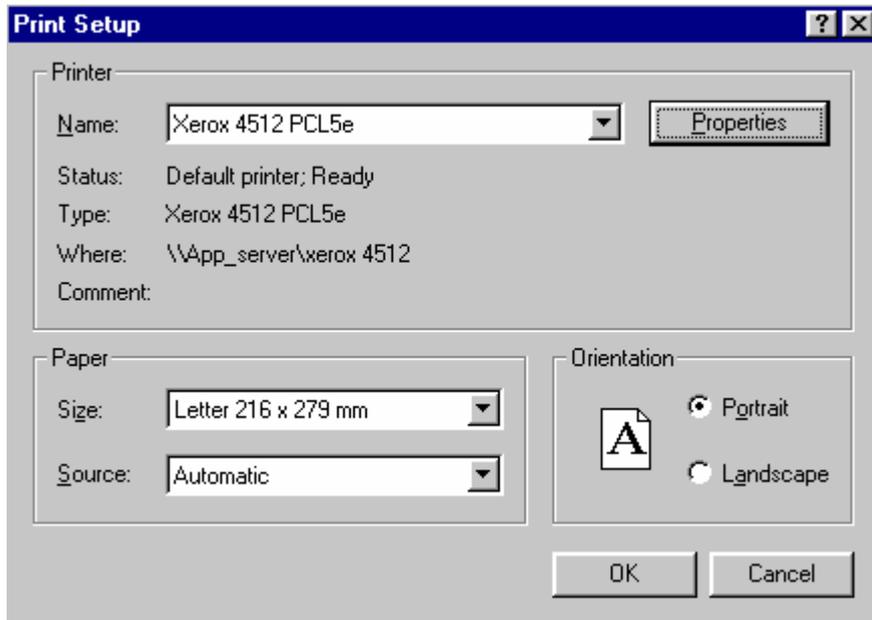
13.5.7.1.6 Printer setup

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.

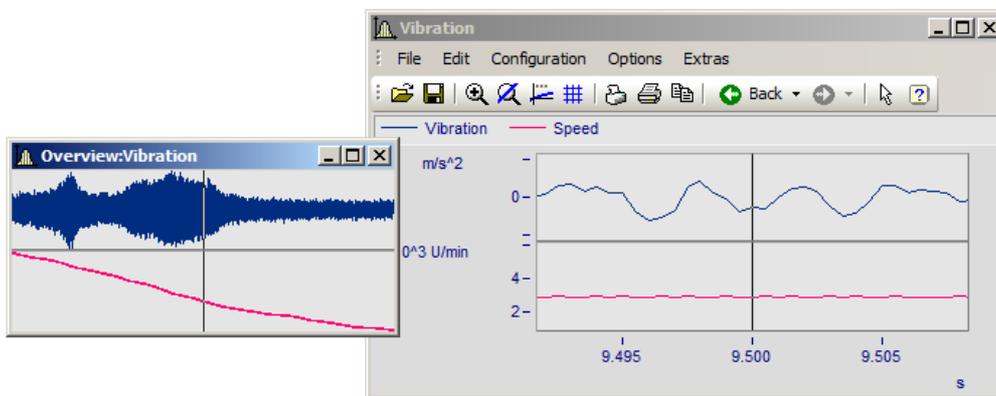
The settings apply to all of the computer's imc curve window's and remains operative upon the program's next start.

13.5.7.1.7 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.

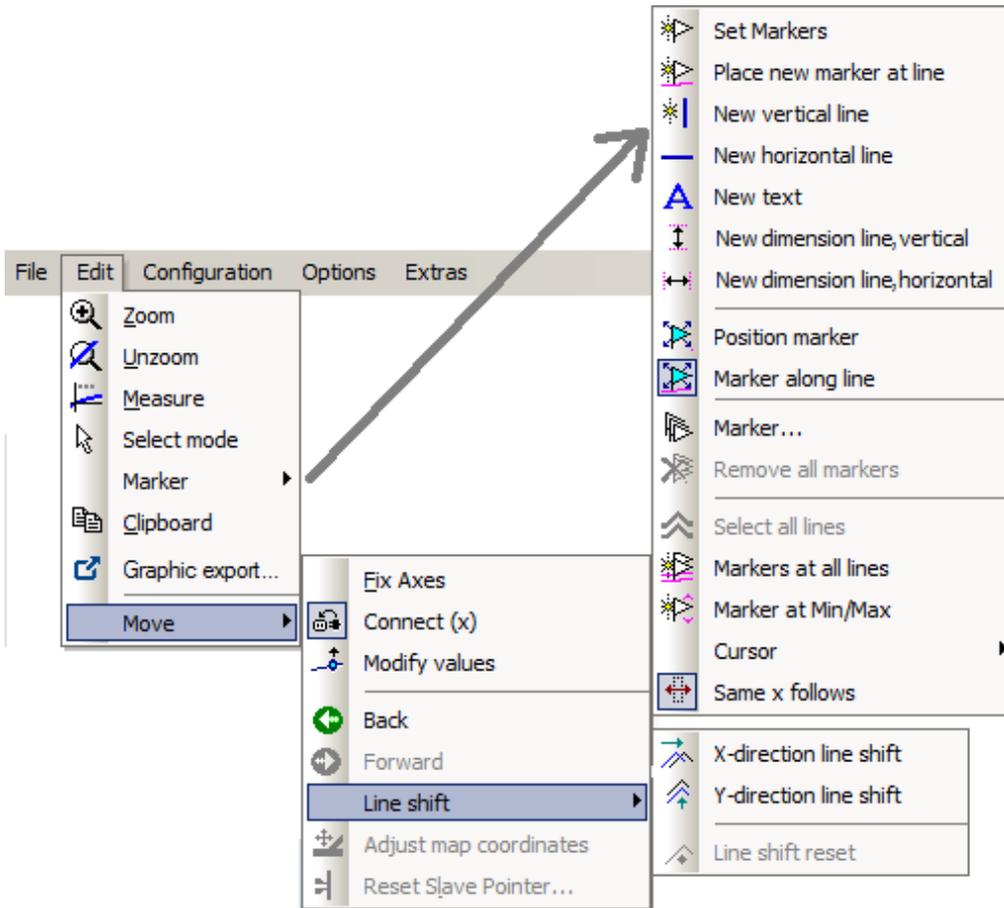
13.5.7.1.8 Twin window

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](#)⁸⁶⁸).
- Display of a Waterfall graph in 3D, or a color map.
- Simultaneous display of the exact same data both in an overview and zoomed.

13.5.7.2 "Edit" menu

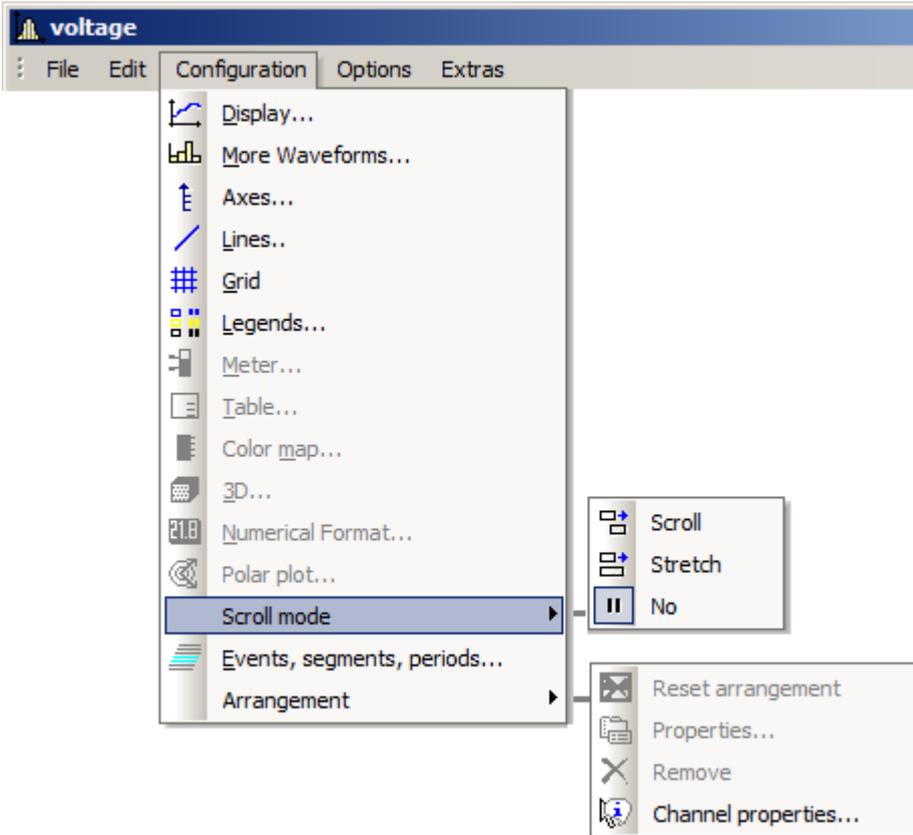


Menu item	Description
Zoom ¹⁰¹⁰	Enlarges a section of the curve window.
Unzoom ¹⁰¹¹	Displays the entire curve.
Measure ¹⁰¹²	A measurement value window and measurement cursors for measuring curves are displayed.
Select-mode ¹⁰²⁶	In this mode, it is possible to select and edit legends, coordinate systems, axes, curves and markers.
Marker ¹⁰²⁸	Proceeds to the list of marker functions.
Clipboard ¹⁰⁵²	Copies the graphics in the curve window into the Windows clipboard.
Graphic-export... ¹⁰⁵²	The graphic of the curve window is saved as image or as a PDF file.
W To MS WORD ¹⁰⁵⁴	The curve window is embedded as an OLE object in Microsoft WORD.
Move ¹⁰⁵⁷	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](#)' ¹¹⁰¹.

13.5.7.3 "Configuration" menu



Menu item	Description
 Display ⁸⁵²	The structure of the curve window and other attributes, such as the "Freeze" mode and time/date display can be selected here.
 More Channels ⁹²⁴	Selects additional waveforms to be displayed in the current curve window.
 Axes ⁹³⁵	Sets x- and y-axis parameters.
 Lines ⁹⁵⁴	Sets line (measurement cure) parameters.
 Grid ⁹⁸⁹	A grid can be added to the coordinate system.
 Legends ⁹⁶⁷	Sets the display of legends in the curve window.
 Meter ⁸⁷⁹	Sets properties of bar meter display.
 Table ⁸⁷³	Sets properties of tables display.
 Color map ⁸⁸⁸	Sets properties of color map display.
 3D ⁹⁰⁰	Sets properties of 3D display.
 Numerical format ⁸⁶⁸	Sets properties of the display mode "Last value as number".
 Polar plot ⁹⁰⁰	Sets properties of polar plot.
 Scroll mode ⁹⁹⁹	In the curve window, a particular section of the data set can either "scroll past" or continuously "grow" from the beginning.
 Events, segments, periods ¹⁰⁰⁰	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 ¹⁰⁰⁷	Restores the curve window arrangement, access to properties and deletion of selected objects.

13.5.7.4 "Options" menu

Menu item	Description
Clipboard settings ¹⁰⁷⁴	Settings for graphics created for output or export with the clipboard are specified here.
 Colors ¹⁰⁷⁸	Changes the colors in the curve window.
Presettings ¹⁰⁸¹	Presettings such as the standard configuration directory and drives for storage of temporary files are made here.
 Printer preview ¹⁰⁸⁸	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.

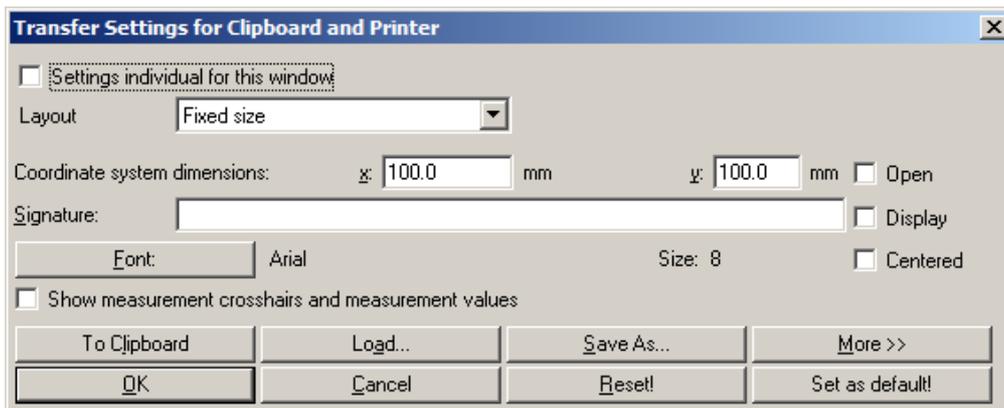
13.5.7.4.1 Clipboard settings

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](#) ¹⁰⁷² 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.

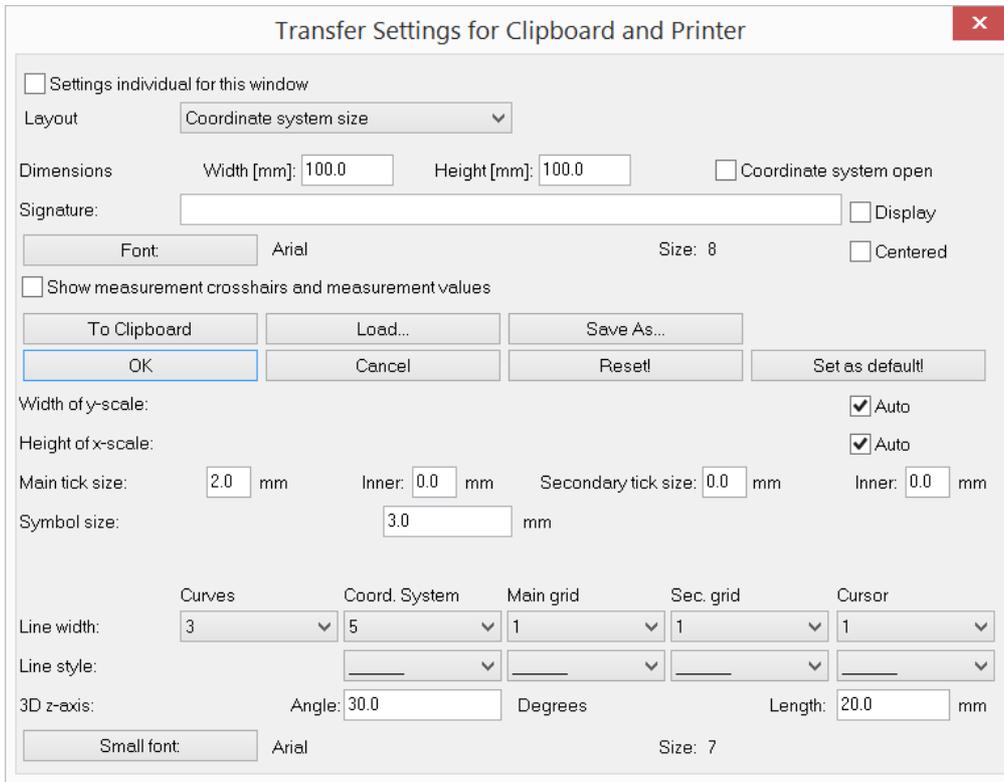
Settings	Description
<i>Layout</i>	<p>Proportions as on screen (the default setting!) saves the curve window to the clipboard with the proportions which accord with the settings currently applied on screen.</p> <p>The layout Coordinate system size creates a graph in which the coordinate system exactly applies the x and y values specified by the setting <i>Dimension</i>. Observe also the notes on the Table^[1075].</p> <p>Total size creates a graphic in which the outer edge of the curve window is determined by the x and y values.</p>
<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>	<p>Select here the font for the axes scaling. The MS-Windows standard dialog appears when the "Font" button is selected.</p> <p>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.</p>
<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](#)^[938] 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:



Settings	Description
<i>Width of y-scale</i>	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.
<i>Height of x-scale</i>	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.
<i>Size of ticks</i>	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.
<i>Symbol size</i>	Lines can also be distinguished by accompanying symbols, see the menu items Configuration/ Axes ⁹³⁵ and Configuration/ Display ⁸⁵² . 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>	<p>The width can be specified for the following lines:</p> <ul style="list-style-type: none"> curves, coordinate system, main grid, secondary grid, measurement cursor <p>Any specified line thickness ⁹⁵⁵ for an individual waveform overwrite the settings saved in the Clipboard.</p> <p>Specify the line width with a setting between 1 and 100. A very fine line width can be achieved with the lowest settings.</p> <p>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.</p> <p>Line widths are generally set to between 1 and 5.</p>
<i>Line style</i>	<p>Various line styles can be selected for the following:</p> <ul style="list-style-type: none"> coordinate system, main grid, secondary grid, measurement cursor <p>The following line types can be selected:</p> <ul style="list-style-type: none"> broken, dense dots, coarse dots, dense dashes, coarse dashes, alternating dots and dashes <p>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.</p>
<i>z-axis angle</i>	<p>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.</p>
<i>Length of z-axis</i>	<p>The length of the z-axis can be specified in mm for the waterfall diagram, with up to one place after the decimal.</p>
<i>Small font</i>	<p>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.</p>

Remarks

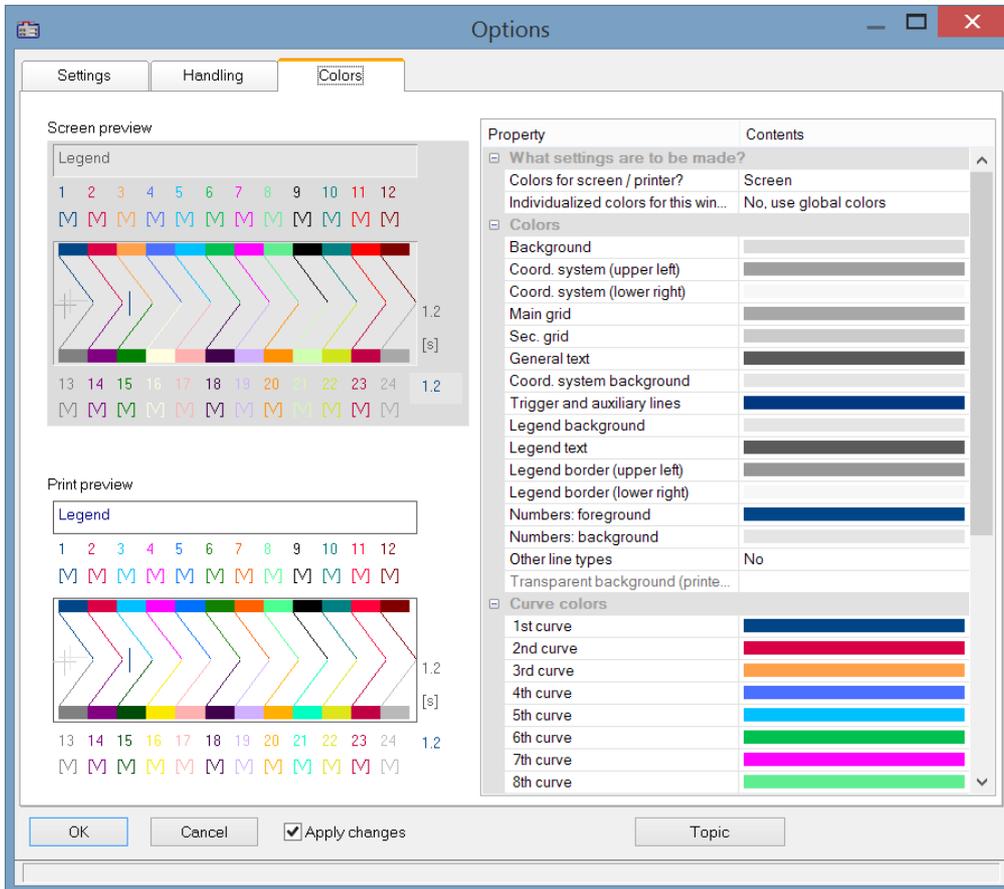
- The settings in this dialog apply to all curve windows.
- The fonts available for selection depend on the printer set for the curve window.
- When graphics are transferred to the clipboard, it is important that the printer set for the curve window is also the printer with which the graphics will later 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.

13.5.7.4.2 Colors

Function

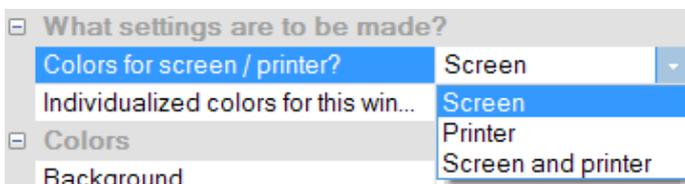
All colors used in the graphic interface of the curve window can be defined as desired in a special dialog. Single windows can use an individual color scheme.

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..](#)¹⁰⁷⁹.

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.

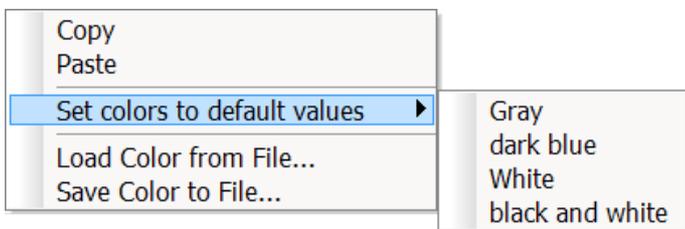
A dialog appears in which the colors can be defined. This dialog contains a list of the graphic elements in the curve window.

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

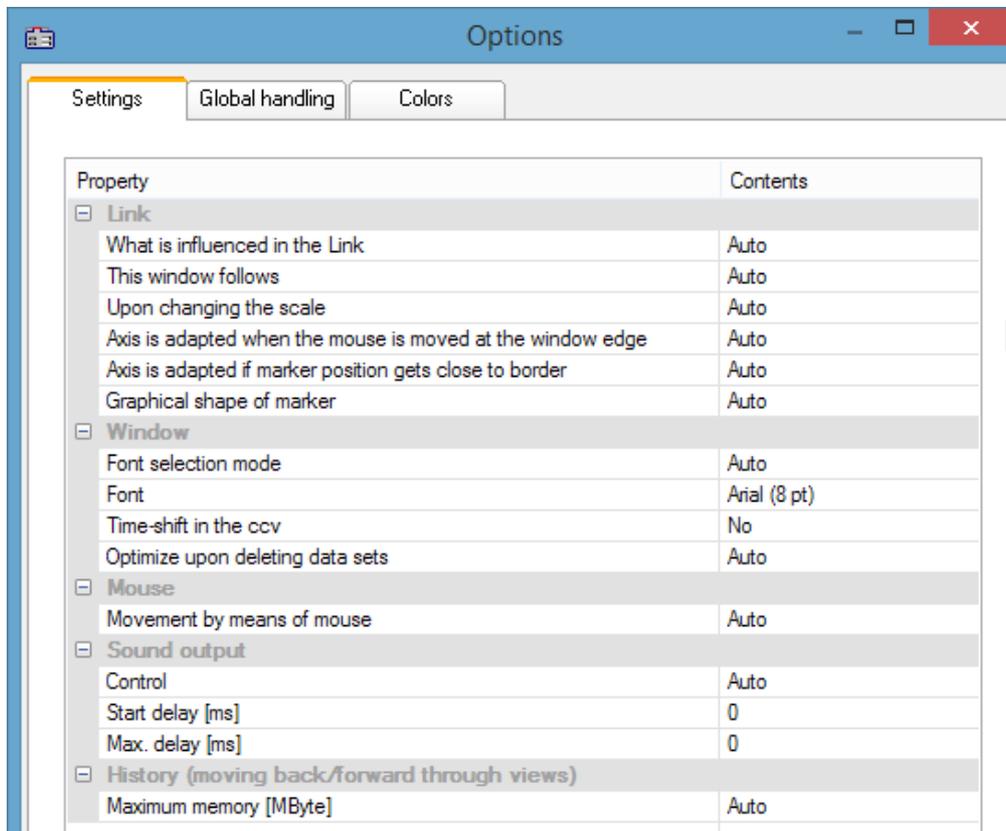
See also

"[Curve window colors on the Panel page](#)"^[1103] - Which colors are used for the on-screen display and which for the printout.

13.5.7.4.3 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



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 [map](#)^[977]
x-, yAxis (color map): [both the x-, and the y-direction are linked](#)^[1058].
y-axis: follows the y-direction in color maps
Cross section: creates a cross-section referenced to the linked axis on the color map.

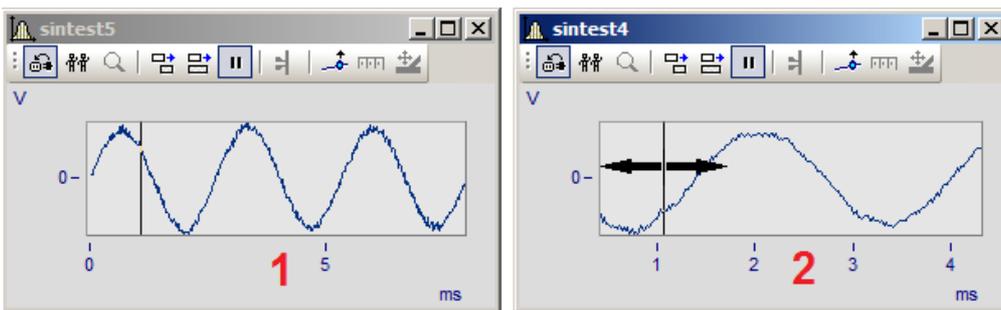
This window follows **auto:** mostly Line follows
Axis follows: The curve and the axis are moved, but the link pointer remains stationary.
Line follows: The curve and axis remain stationary while the link pointer moves.

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 [map](#)^[977]
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 [map](#)^[977] 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 ^[1096] .
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 ^[1058] 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.

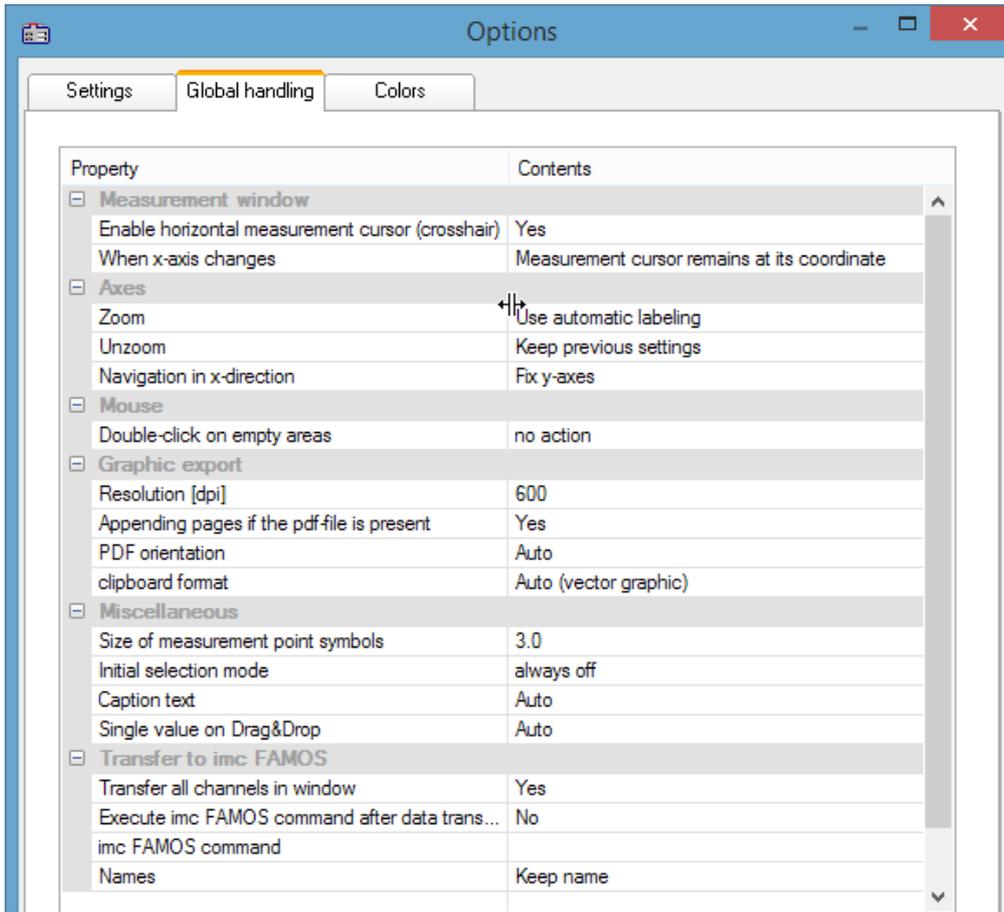
Vector graphic: The interior of the coordinate system generates a vector graphic.

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](#)^[1086]. If the setting there is for export of the curve window as a [Bitmap to PDF](#)^[1086], 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	<i>yes/no</i> : display of the horizontal measurement cursor (yes: crosshairs; no: vertical line)
When x-axis changes	<i>Sets whether the measurement cursor remains at the data set coordinate or at the pixel position when the zoomed region is changed.</i>
Mouse	
Double-click on empty areas	Enables Select-mode ^[1026] 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	<i>Use automatic labeling; Place labels at end, use rounding; Place labels at end, use no rounding; Keep previous settings</i>
Rezoom	See Axes Scale ^[936]
Navigation in x-direction	Determines whether after navigating ^[1107] 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	<p>Bitmap: The entire graphic is converted to PDF as a bitmap. Resolution in dpi. (Default before Version imc FAMOS 7.3)</p> <p>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.</p> <p>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.</p> <p>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.</p> <p>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^[1084].</p> <p>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!</p> <p>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.</p> <p>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>".</p>
Miscellaneous	
Size of measurement point symbol	0.5 -10 mm. Size of the symbols ^[956] representing the measured points, e.g. squares, circles...
Initial selection mode	Default setting for Select-mode ^[1026] .
Caption text	<p>Auto: Name of first dataset.</p> <p>Filename: The filename of the loaded ccv file can be shown in the caption of a free-floating curve window.</p>
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](#)¹¹⁰¹⁶.

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.

13.5.7.4.4 Printer preview

This item is only enabled in the Data Browser's Report view.

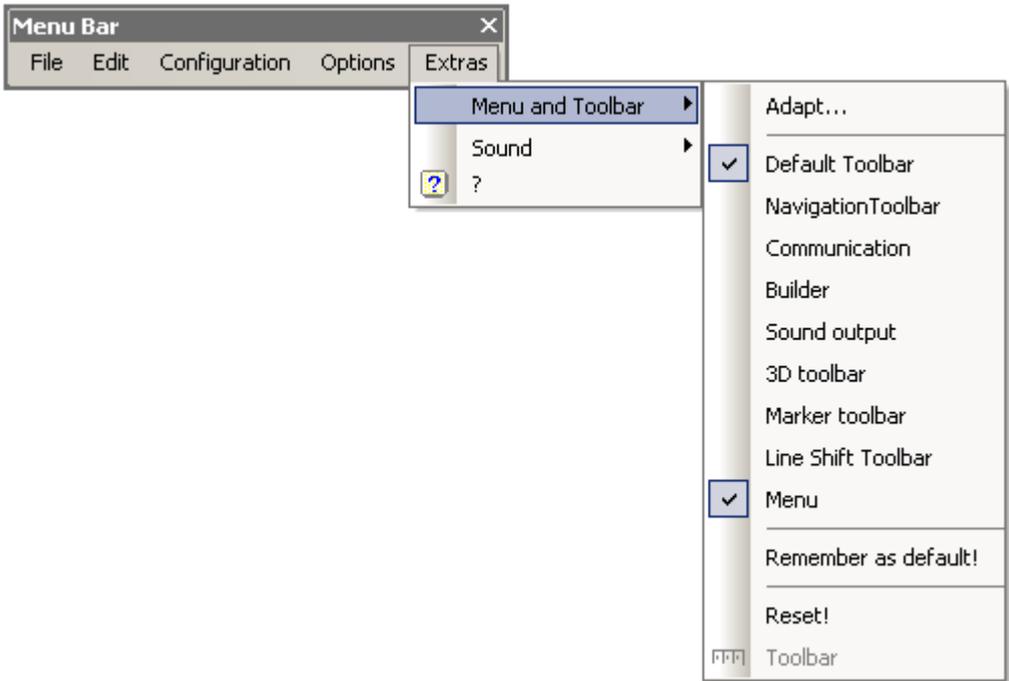
The *Print Preview* is activated by default for curve windows which are incorporated into a report in the Data-Browser. This means that the colors are displayed which the user set in the [Colors](#)¹¹⁰⁷⁸-dialog in the *Options* menu as the color scheme for the printer.

With this option, it is possible to switch to the monitor screen's color scheme.

13.5.7.5 "Extras" menu

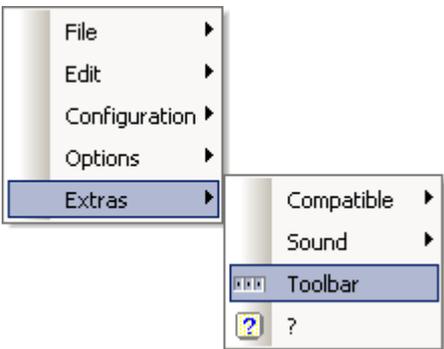
Menu item	Description
Menu and Toolbar ¹¹⁰⁸⁹	Options pertaining to the menu and toolbar.
Sound ¹¹⁰⁹²	Activates the sound output and access to the functions of the Sound-Toolbar.
?	Displays the curve window's Help.

13.5.7.5.1 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

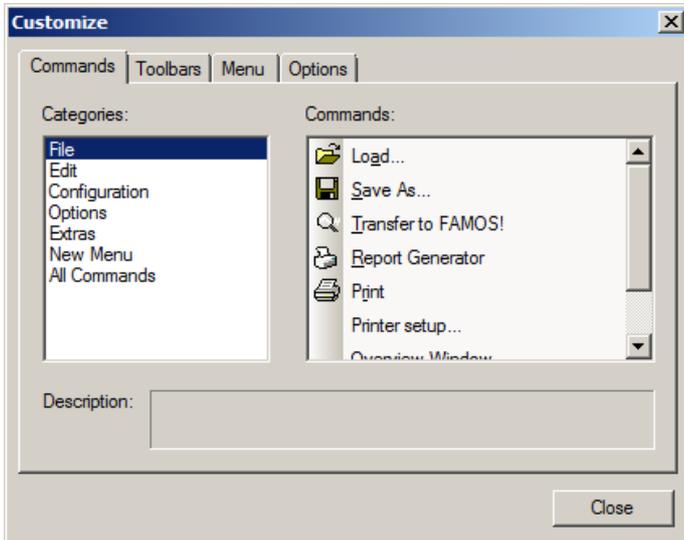


As a curve window integrated in the Data-Browser's Report view, the entire Toolbar only appears when a curve window is selected. To cancel this behavior so that the Toolbar does not appear, the function "Toolbar" can be selected. This is only available from within the Data-Browser. To make the toolbar visible again, select "Toolbar" under "Extras" in the curve window's context menu.

13.5.7.5.1.1 Adapt / Customize

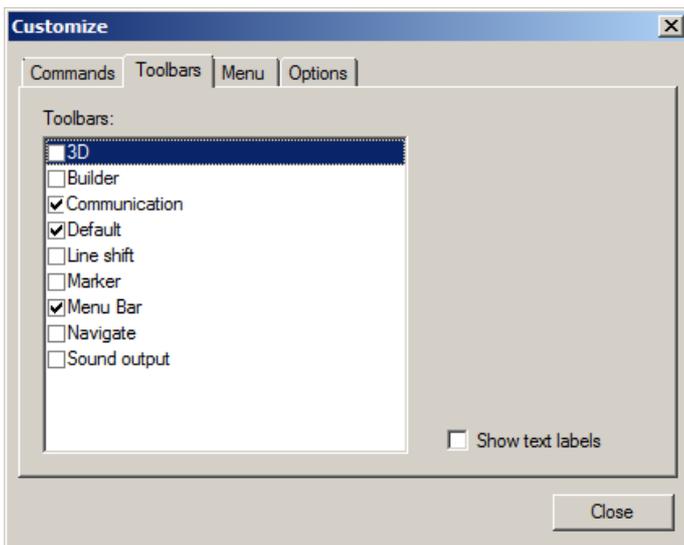
The curve window's menu and toolbar can be adapted. Right-clicking in either the menu or the toolbar opens the context menu. Alternatively, select the menu item *Extras / Menu and Toolbar*.

The menu item *Adapt...* opens the following dialog:

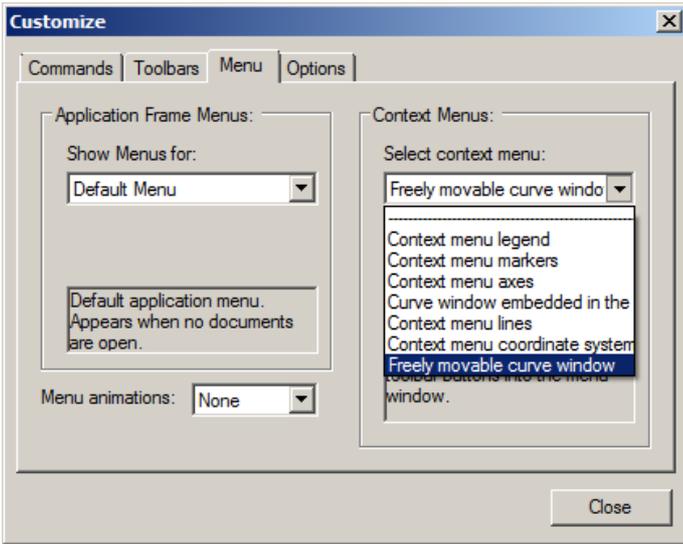


A command can be dragged from the list at right and dropped on a menu or toolbar. Pop-down menus open by themselves when the mouse is moved over them. The list at left contains the different command topics.

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.

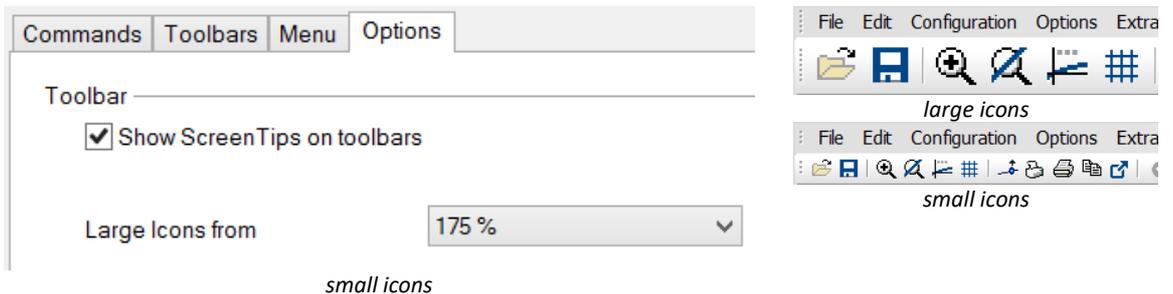
- normal curve window which is free-floating and freely movable and can also be maximized (popup-window)
- a curve window without title bar (child window) embedded in a dialog.

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*:

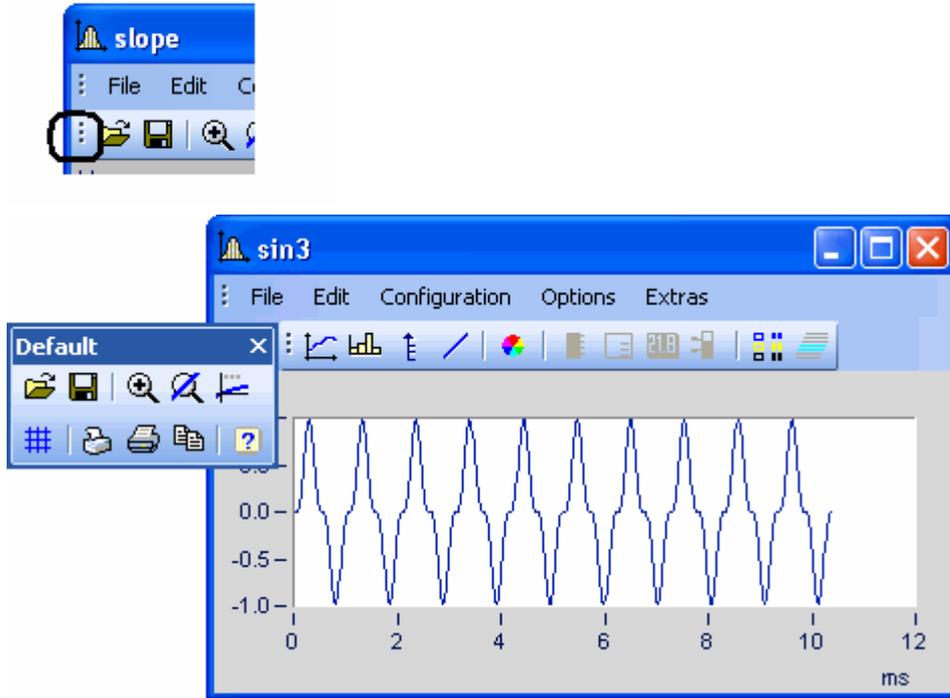
Options	Description
Show tips on toolbars	A description of the icon is displayed when the mouse is moved over the window.
Large buttons	For clearer display at fine screen resolutions such as 4K, the icons can be automatically enlarged from a certain scale.



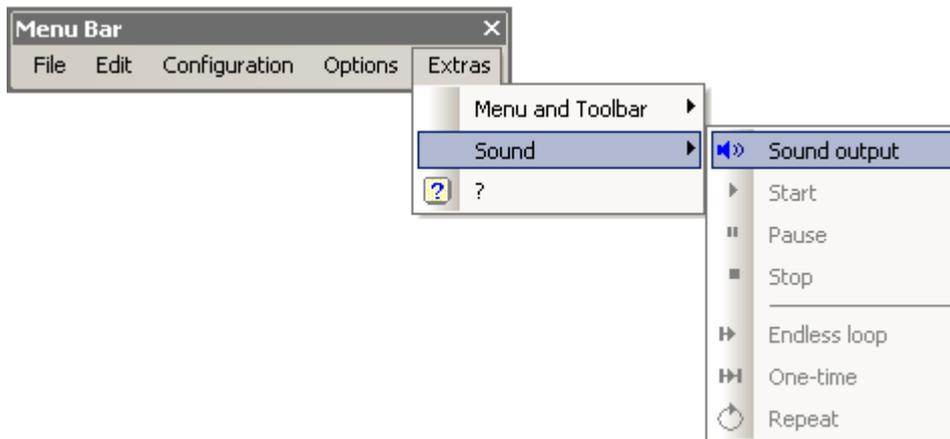
Changes of the menu are permanently saved when the dialog *Customize* is closed. The particular toolbar arrangement saved depends on the display type:

- 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:



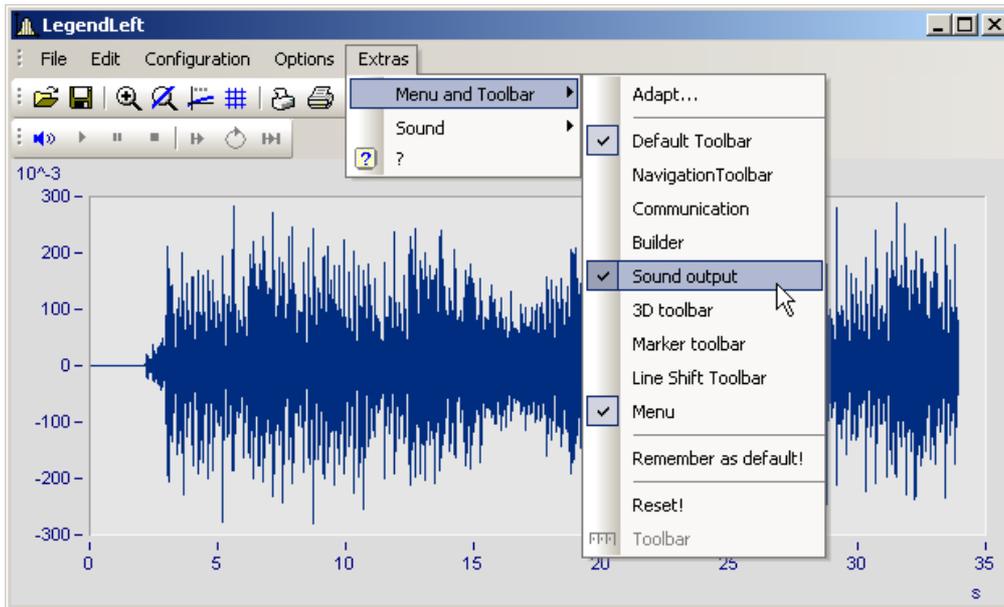
13.5.7.5.2 Sound



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.

13.5.7.5.2.1 Sound toolbar

For working with the sound output, you should select the associated toolbar from the menu *Extras / Menu and Toolbar / Sound output*.



Sound output toolbar

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 ⁽¹⁰⁹⁵⁾).
 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 ⁽¹⁰⁹⁵⁾
 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.

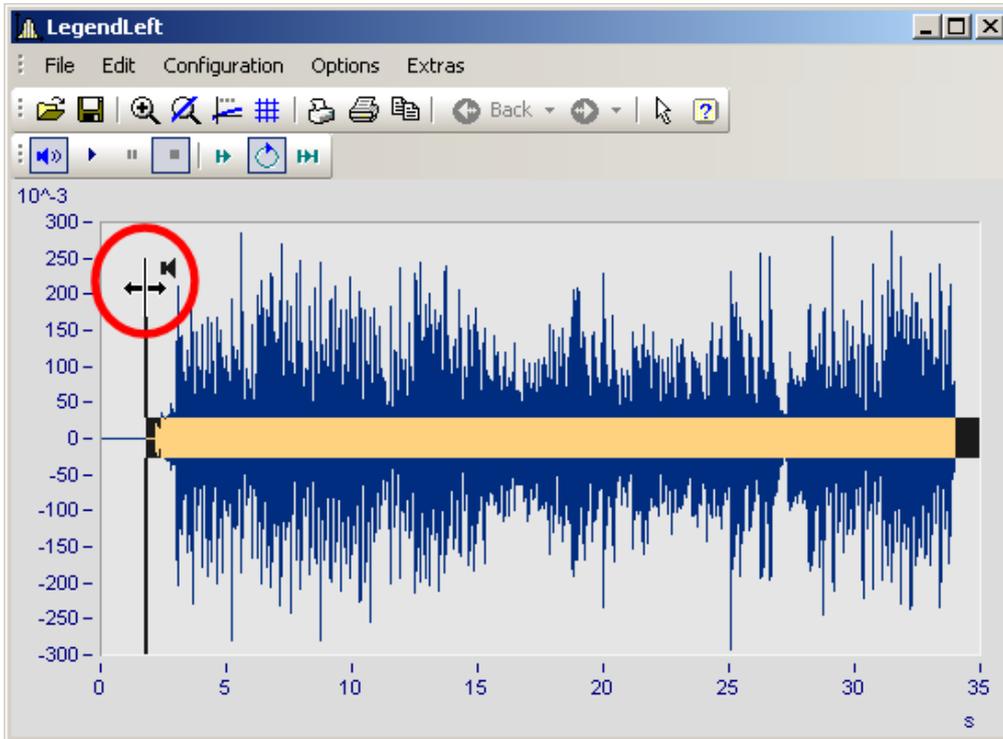
When the volumes of various signals are to be compared, all y-axes must be set to the same scaling. If the y-axes are scaled automatically, all signals have the sound card's maximum level.

The Sound Kit functions have not settings options for volume.

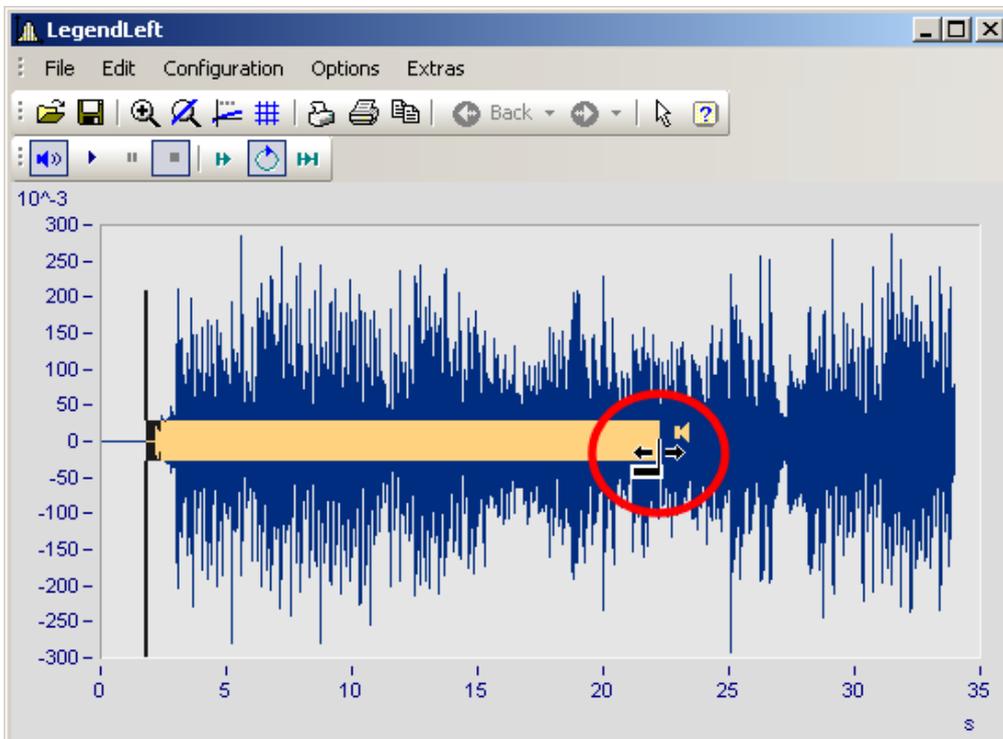
13.5.7.5.2.2 Sound clip

In sound output, it is possible to freely select (take a clip from) the playback region.

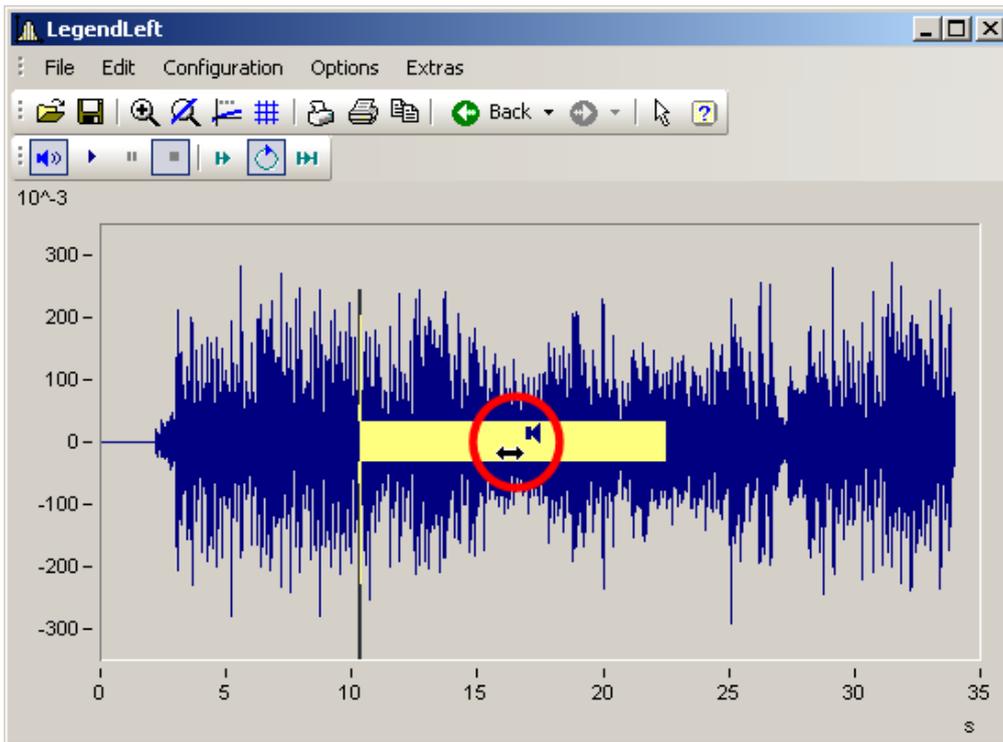
Grab the start position at the beginning of the Playback strip and move it to the position desired.



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.



13.5.7.5.2.3 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](#)^[1083], 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.

13.5.8 Information and tips

13.5.8.1 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*a>`.

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.

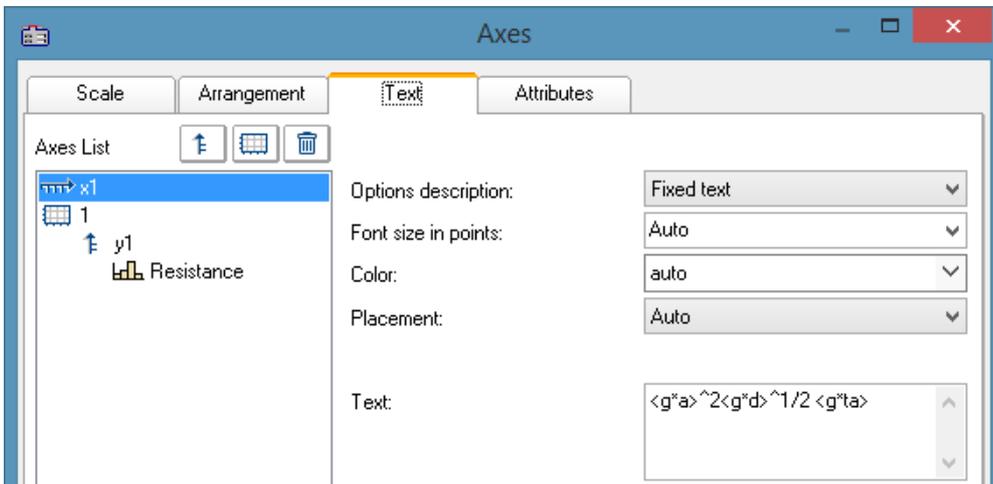
Placeholder	Unicode	Meaning	Placeholder	Unicode	Meaning
'a'	0x3b1	Alpha	'm'	0x3bc	My
'b'	0x3b2	Beta	'n'	0x3bd	Ny
'g'	0x3b3	Gamma	'x'	0x3be	Xi
'G'	0x393	uppercase Gamma	'X'	0x39e	uppercase Xi
'd'	0x3b4	Delta	'p'	0x3c0	Pi
'D'	0x394	uppercase Delta	'ph'	0x3c6	Phi
'e'	0x3b5	Epsilon	'ps'	0x3c8	Psi
'et'	0x3b7	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

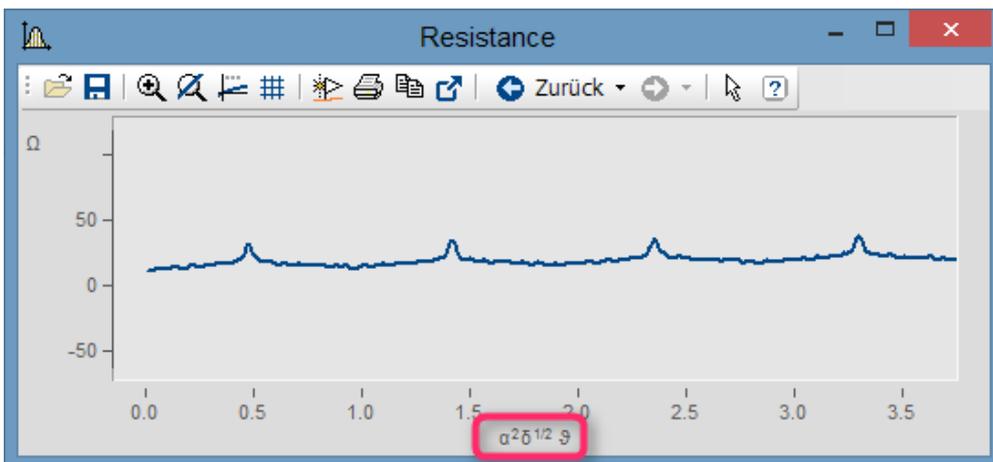
curve window

with



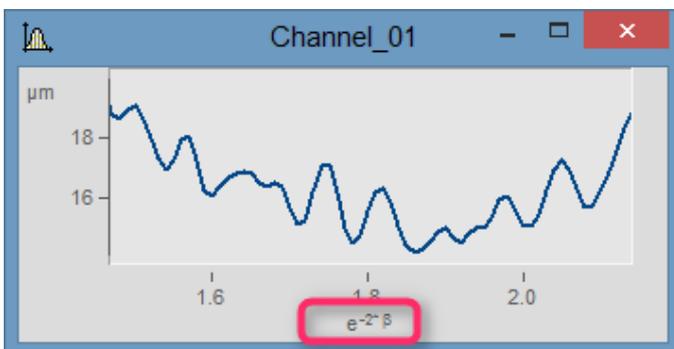
Greek characters in "Axes\Text": $\alpha^2 \delta^{1/2} \gamma$

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: $e^{-2 \cdot \gamma \cdot b}$

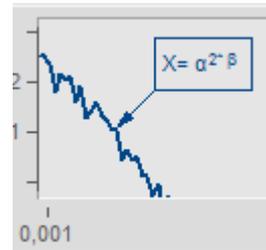
 Example

markers

with

Greek characters in marker-definition:

the result is:



Greek characters in markers

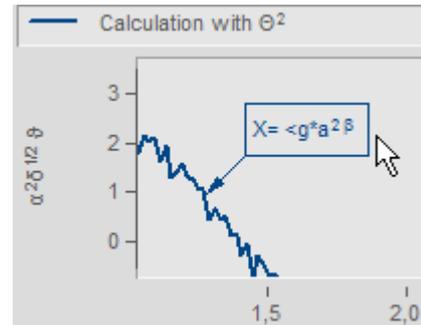
 Example

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".

13.5.8.2 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.

13.5.8.3 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).

13.5.8.4 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**.

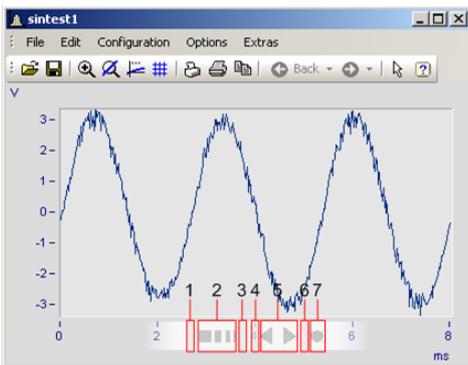
Remarks

- 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.

13.5.8.5 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

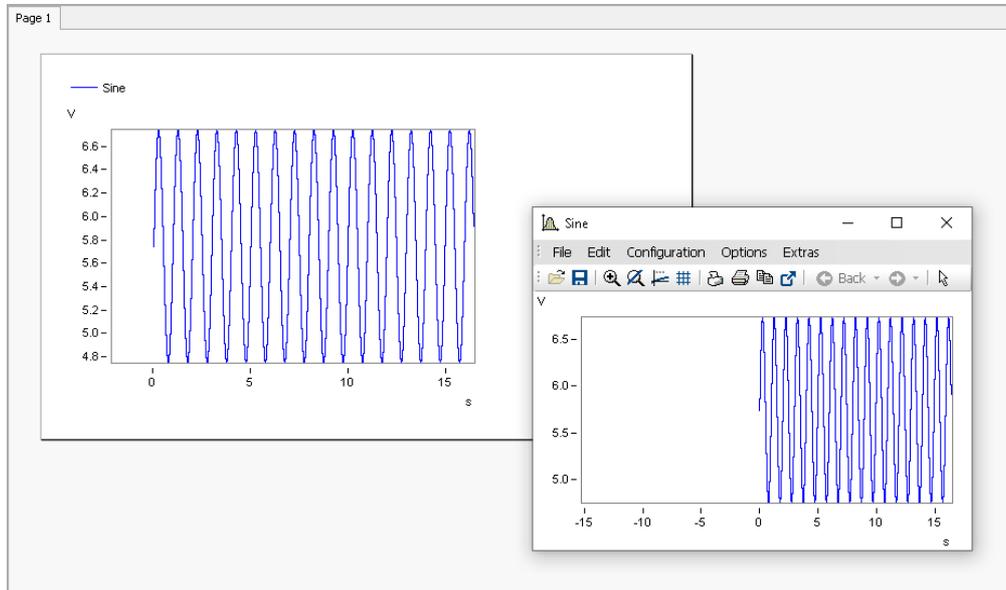
Use of the keyboard arrow keys

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.

13.5.9 Curve windows in imc STUDIO

13.5.9.1 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:

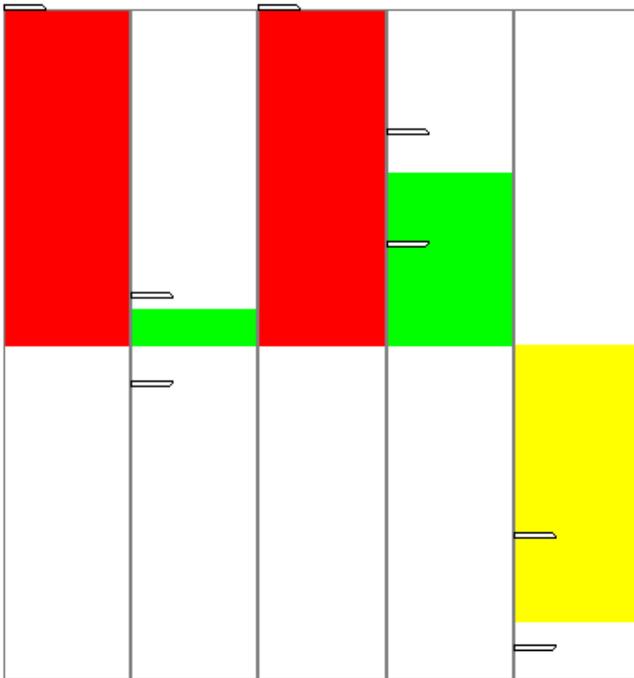
- Double-click in the tool window Data Browser on the desired channel
- or select the desired channels and click on the [Show](#) -button ( / ) in the symbol bar of the tool window Data Browser.

13.5.9.2 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.

13.5.9.3 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](#)^[1078]", 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	<ul style="list-style-type: none"> • On the monitor screen: Screen color • In the printout: Screen color

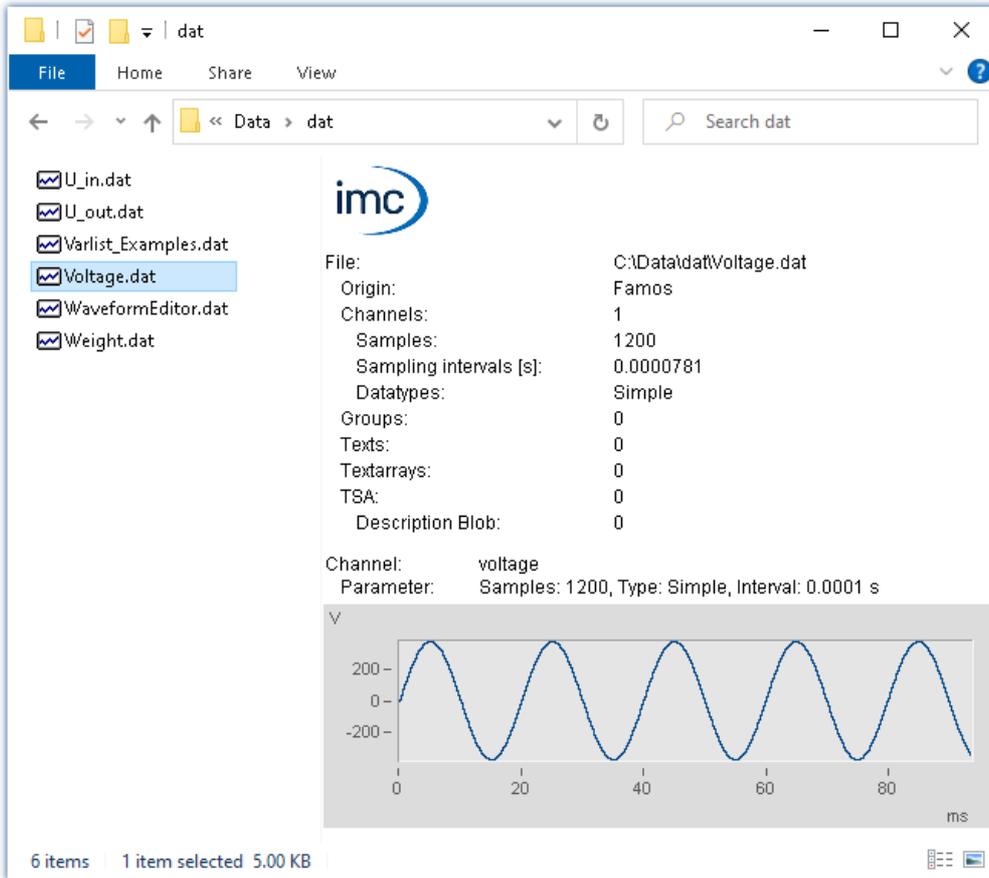
Page	Colors
Report-page	<ul style="list-style-type: none">• On the monitor screen: Printer color (The Report-page is designed for the printout or for the PDF. In consequence, the printer colors are displayed here.)• In the printout: Printer color

**Note****Curve window on Report pages**

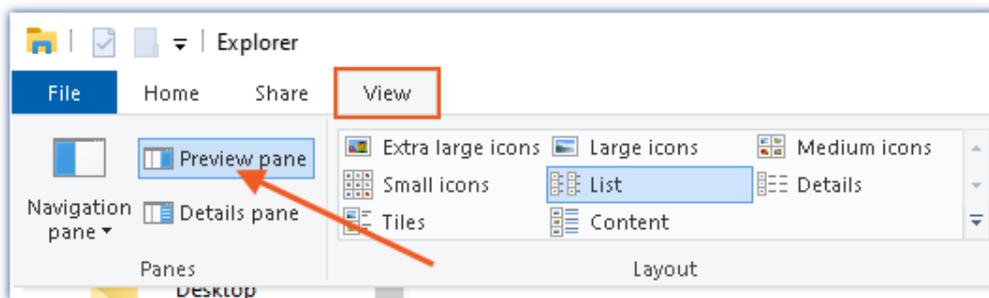
Curve windows on Report pages are set so that the global colors are used for the printout. Furthermore, the background is always white, even when a different color is set.

13.5.10 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	<ul style="list-style-type: none"> • 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	<p>Infos:</p> <ul style="list-style-type: none"> • 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	<p>Only one data set is always displayed. When multiple files are selected, only the first is displayed.</p> <p>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.</p>
Scroll bar	<p>When the file contains multiple channels, a scroll bar appears, by means of which it is possible to navigate through the various channels.</p>

13.6 Special Widgets

Here, special Widgets are described.

13.6.1 Text input for report channel

This Widget currently has no function.

13.6.2 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 WAVE (" VRTC ₁₉₅₁ "). This time is used for all components of imc WAVE.
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"	<p>Displays the moment in time at the last change of the state: Armed, Triggered, Stopped. I.e. when:</p> <ul style="list-style-type: none"> • the measurement is started, • the trigger fires, • the trigger event ends and • the measurement is stopped. <p>Thus, stopping the measurement also resets the clock after a previously concluded triggered measurement.</p>

Clock properties

Properties	Description														
Appearance	Here you specify the appearance of the clock; e.g. with clockhands , or a digital readout .														
Mask	<p>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.</p> <table border="1"> <thead> <tr> <th>Mask</th> <th>Description</th> </tr> </thead> <tbody> <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> </tbody> </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	<p>In imc STUDIO, it is possible for differing time zones and clock readings to coincide.</p> <ul style="list-style-type: none"> • 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. <p>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.</p>														

13.6.3 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

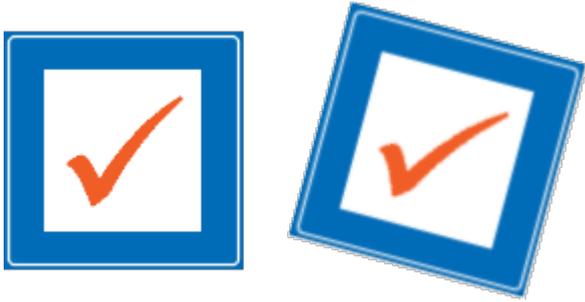
If you wish to display a background image over the entire page, you can use the page's property: "[Background image](#)".

Properties

Properties	Description												
image	<p>Available file types: png, jpg and bmp</p> <p>If an image is selected, the text: "(<i>Bitmap</i>)" appears under this property. In order to delete the image, delete the text.</p> <p>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.</p>												
Size mode	<p>The image can be stretched or displayed without any zoom factor:</p> <table border="1"> <thead> <tr> <th>Size mode</th> <th>Description</th> </tr> </thead> <tbody> <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 ("<i>Background color</i>").</td> </tr> </tbody> </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 (" <i>Background color</i> ").
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 (" <i>Background color</i> ").												
Background color	The color displayed wherever the Widget is not totally filled by the image.												

13.6.4 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 ¹⁸³⁷ . 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.

13.6.5 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

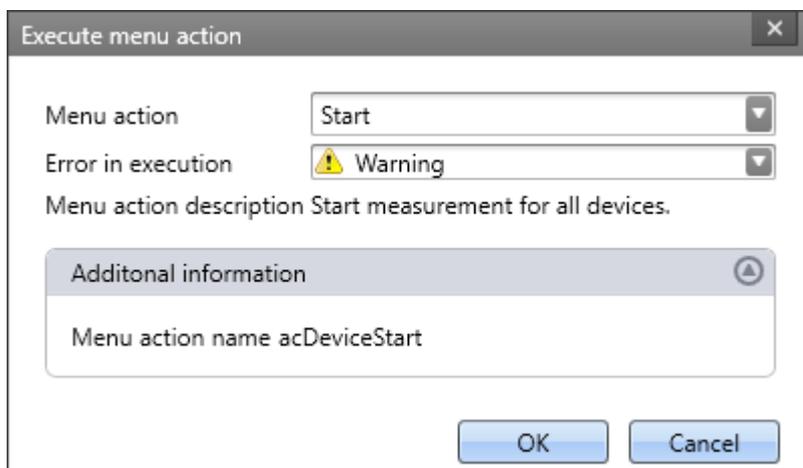
This Widget adapts itself to the status of the menu action. E.g. when the action is currently not present, the button can not be clicked. Some menu actions are attached to specific user roles, device components or loaded imc STUDIO-components.



Reference

Command - Execute menu action

For Sequencer-routines, there is a [command of the same name](#)¹¹⁹⁰. 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. <ul style="list-style-type: none"> • 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.

13.6.6 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](#)^[1113].

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](#)^[1116]" 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. <ul style="list-style-type: none"> • 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" > "Direction^[1113]", 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) <p>Useful property: "Automatic table length^[1113]"</p>

Action	Description
Adding/removing columns and rows	<p>By means of the properties, you can specify the column and row counts.</p> <p>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.</p> <p>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.</p>
Moving columns	Use Drag&Drop to move the columns to the desired position
Table length in relation to the channel length	<p>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.</p> <div style="border: 1px solid orange; padding: 5px; margin-top: 10px;">  <p>When multiple channels are displayed simultaneously, please ensure that the point counts and update rates of the different channels are the same.</p> <p>Do not use another column to display single values.</p> </div>
Order/read-out direction	<p>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.</p> <p>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.</p>
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 " ^[1114]

Titles, captions and texts

Text-settings	Description
Change text of column header (title)	<p>Change the title using the property: "Column" > "Appearance" > "Title".</p> <p>When the title is changed, the source^[831] 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^[831]").</p>
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	<p>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".</p> <p>Change the contents in the title column cell by means of the property: "Cell" > "Appearance" > "Cell title".</p> <p>When the title is changed, the source^[831] 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^[831]").</p>
Alignment of the texts	<p>The alignment of the texts (right/left/centered) can be set by means of various properties:</p> <ul style="list-style-type: none"> 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](#)^[1115] 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](#)⁸³⁷". 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.

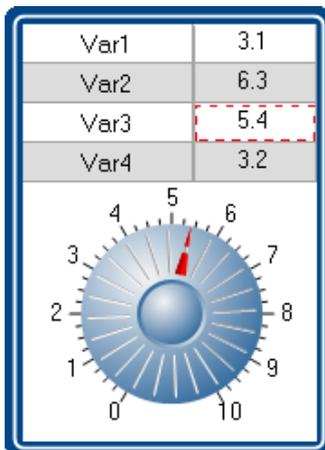
Accept only Zone-values as entries

"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.

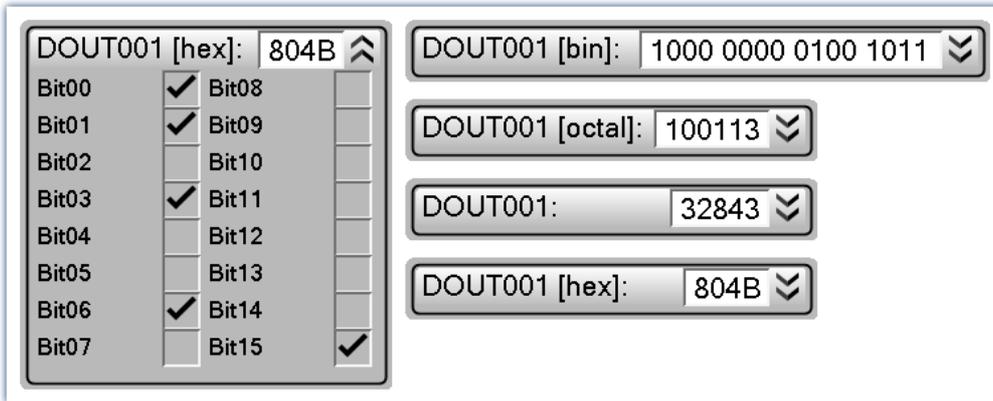
The Widget is linked with the selected cell of the column, which is highlighted by a **red frame**. The Widget displays the value of the selected cell. Additionally, the Widget can also be used as the entry for this cell.

13.6.7 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](#)¹¹¹⁹.

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	<div style="display: flex; align-items: flex-start;"> <div style="flex: 1;"> </div> <div style="flex: 2; padding-left: 10px;"> <p>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.</p> </div> </div> <p style="text-align: center;"><i>Port-input with bit-list</i></p>

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	<p>By means of the context menu: "<i>Hide selected bits</i>" or "<i>Arrange bits</i>"</p> <p>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:</p> <table border="1"> <thead> <tr> <th>Bit</th> <th>Value</th> </tr> </thead> <tbody> <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> </tbody> </table> <p>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).</p> <p>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).</p>	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 border="1"> <thead> <tr> <th>Decimal</th> <th>Hexadecimal</th> </tr> </thead> <tbody> <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> </tbody> </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 processed directly. Numbers which are too large can cause unexpected results.														
	<table border="1"> <thead> <tr> <th>Decimal</th> <th>Hexadecimal</th> </tr> </thead> <tbody> <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> </tbody> </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														

13.7 Pages

13.7.1 Insert - Dialog / Report

The following ways to add a page are available:

- via the page title's "[context menu](#)"^[817]

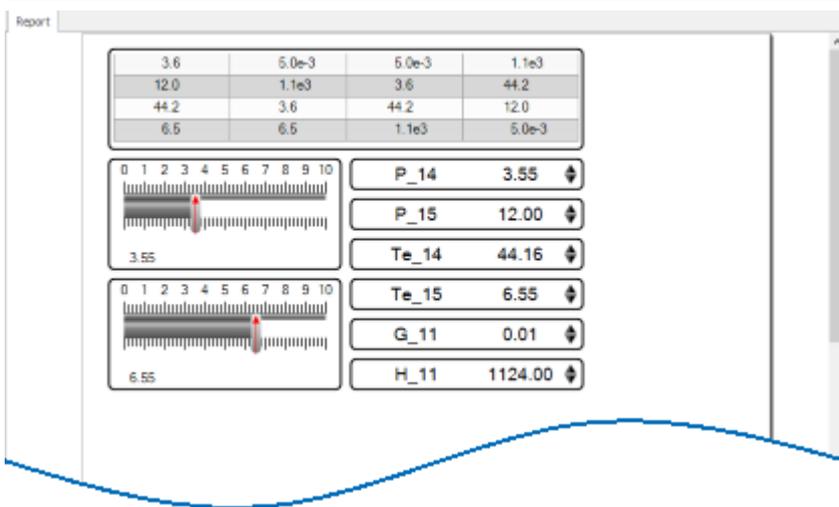
A page can be generated as a "Dialog-page" or as a "Report-page".

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](#)^[817] 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).



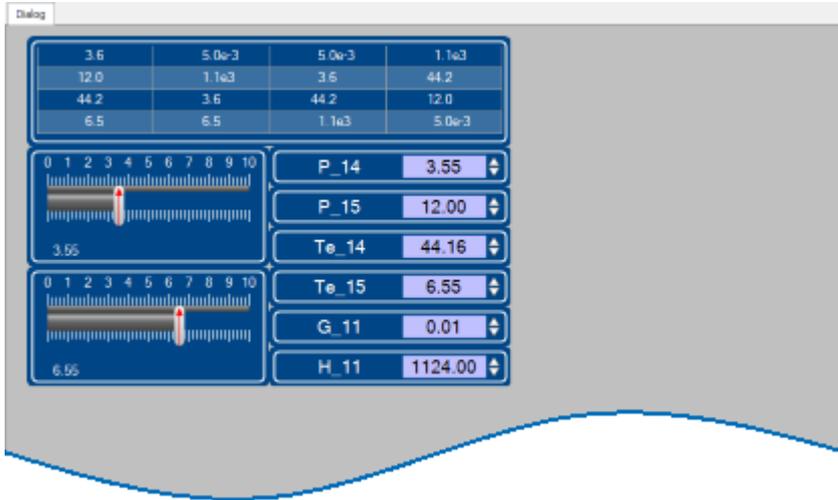
Example of a Report-page

To **print out the page in the page format set**, select the command: "Print Panel page". By means of **PDF-export**, you can generate a corresponding PDF in that format.

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](#)¹¹²³.

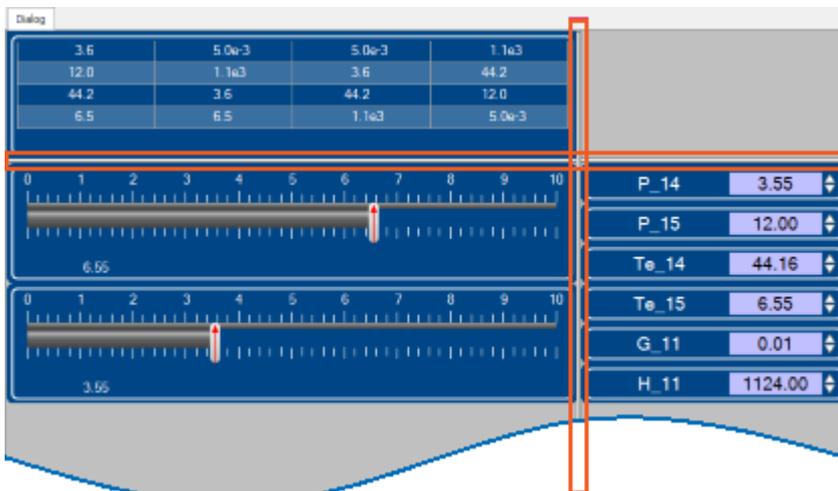


Example of a Dialog-page

After the program is started, the Panel contains a default Dialog-page.

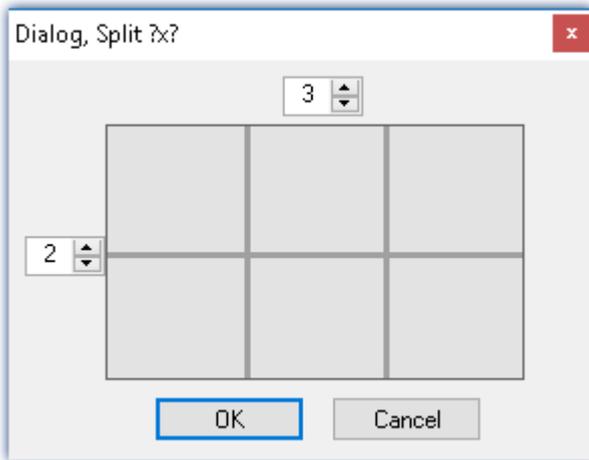
Divided dialogs, e.g. "Dialog 2x2"

These dialogs are partitioned into regions by movable sliders. The size of the [docked](#)⁸²⁹ 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:

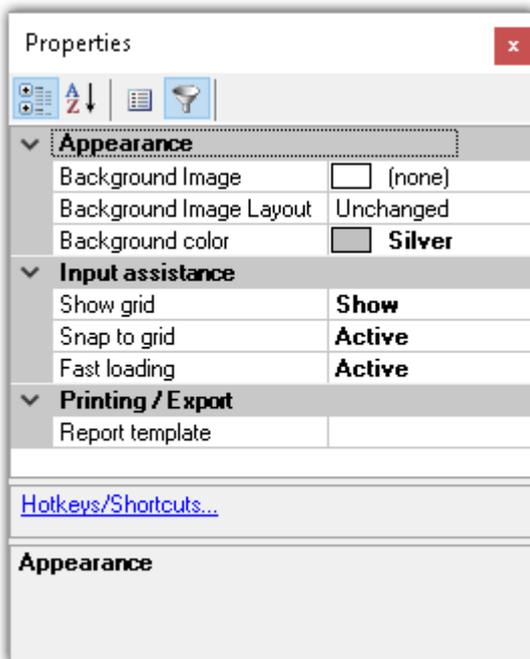


Partitioned format

13.7.2 Properties - Pages

With this tool window, you can set the page's properties. The Panel must be in [Design Mode](#)⁸²⁰ 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 ^[1138] .

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
Page size	Size of the Panel page (<Width>; <Height>) The page size adjusts itself to the setting for the Zoom ^[1128] 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 ^[1128] , 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	<p>Placement of the background image in reference to the page size:</p> <ul style="list-style-type: none"> • 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	<p>Display of the background color:</p> <ul style="list-style-type: none"> • 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 " <i>Print Preview</i> ", 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	<p>Activates/deactivates the alignment of the Widgets to the grid.</p> <ul style="list-style-type: none"> • 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	<ul style="list-style-type: none"> • 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.

13.7.3 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](#)¹¹²⁷.

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.

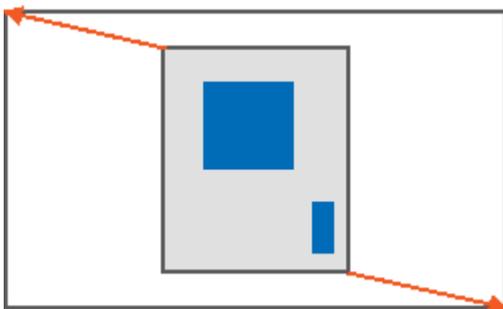
Use of the Zoom function

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)

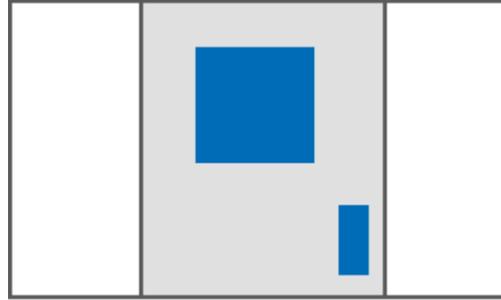
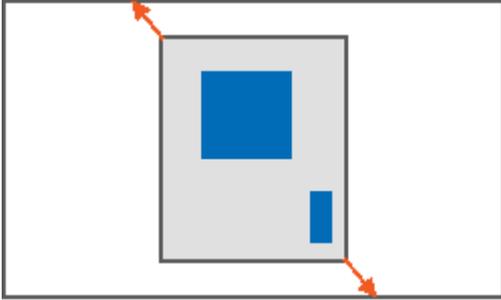


Result

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).

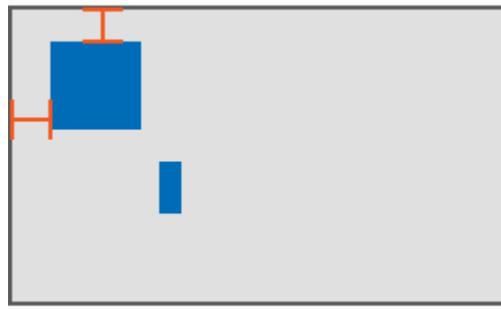
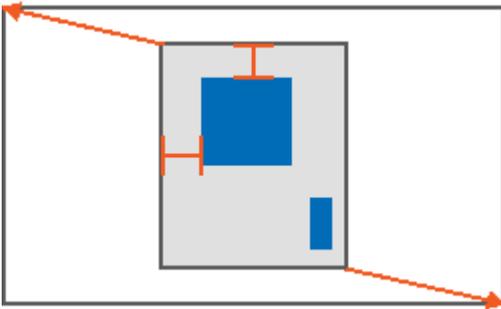


Result

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.

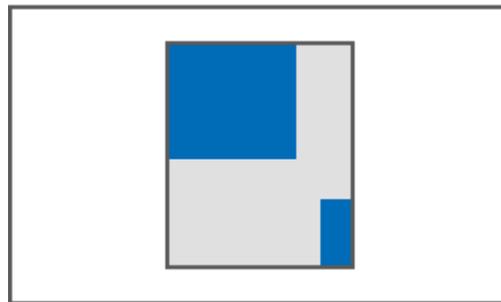
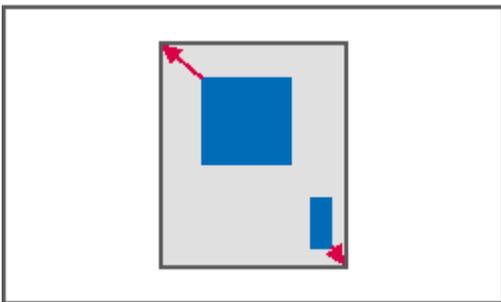


Result

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

13.7.4 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 context menu of the Page-tab.

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](#)"⁸²⁰, 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.

13.7.5 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). <ul style="list-style-type: none"> • 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)

13.7.6 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* (

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

13.7.7 Fullscreen

Panel pages can be displayed on other monitors in a full-screen mode.

Display of a page on a monitor (no "genuine" fullscreen, without protection)

Panel pages can be displayed in **fullscreen on individual monitors**. The imc WAVE 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). So for example, you could see **different pages** on the main monitor and on the second monitor.

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	<ul style="list-style-type: none"> • 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 ^[1185] " is executed, imc STUDIO jumps to the selected page. When a Panel page is defined as a target , then it can also be displayed in fullscreen mode on any desired monitor. In this case, imc STUDIO does not change the main window.

Title Bar - Overview of functions

The fullscreen window has its own title bar (menu). A **variety of functions** can be called from this menu.

	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.
	This button minimizes the program .
	Thus button closes the program .
Data Browser	This button  causes the free-floating Data Browser ^[804] to be shown.

Additional functions can be accessed via the context menu or via the button  in the expanded menu.

Menu item	Description
 Login	Logs in a user
 Logout	Logs a user out
 Design Mode	<ul style="list-style-type: none"> 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](#)"^[1125]".

13.7.8 Page printout or print to PDF

Panel pages can be printout out as hard copy or to a PDF-file. imc WAVE offers a variety of possibilities for this purpose: via the menu ribbon or via the commands ("[Print Panel page](#)"^[1170] / "[Export Panel page](#)"^[1170]).

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 WAVE 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](#) ⁷ 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](#) ⁸¹⁷ over the tab for the respective Panel page)

Menu item	Description
	Opens the dialog: " <i>Page Setup</i> "
Page Layout for Printing	Here, you can change the setup and printout settings for Report-pages (including paper size, source, orientation, margin).

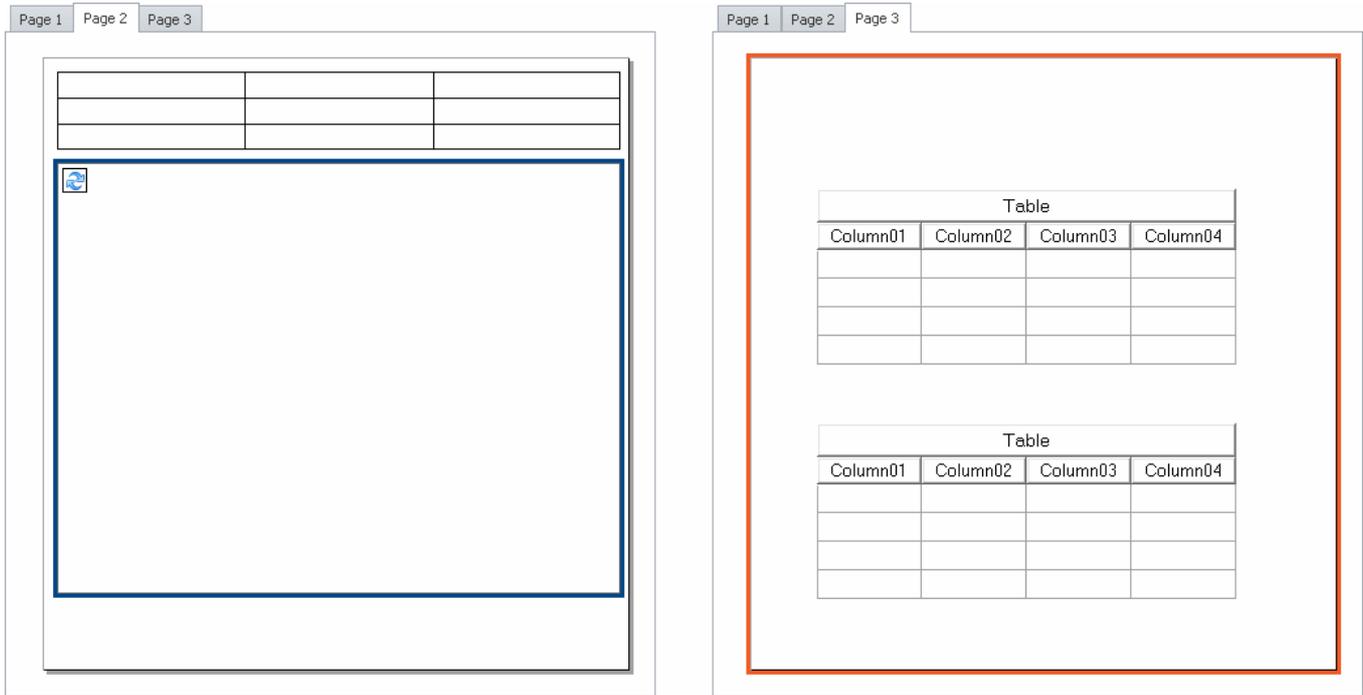
Report template - Header

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.

13.8 Variable linkage

To link a Widget with a variable, there are two methods:

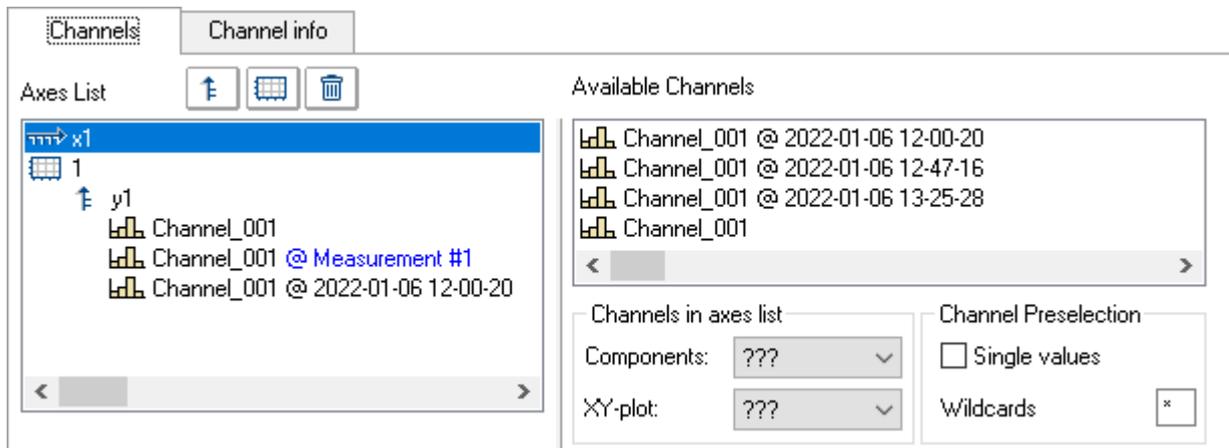
Variable linkage by Drag&Drop ^[1134]	Move the variable using Drag&Drop from the Data Browser ^[804] to the Widget or the Panel page.
Variable linkage by Widget Properties ^[1134]	Open the " Properties " ^[830] window of the Widget and there click on " Variable ".

Even **without existing variables**, it is possible to set up a linkage (see: "[Variable linkage to not yet existing variables](#)"^[1136]).

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	<p><VariableName>@<Measurement Name></p> <p>e.g. Channel_001 In this way, the Widget always indicates the current measurement.</p> <p>or Channel_001@2022-01-06 12-00-20 in this way, the Widget is always linked with this concrete measurement</p>
Symbolic name	<p><VariableName>@Measurement#<Measurement number></p> <p>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.</p> <p>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.</p>



Properties of the curve window. Example: three times the Channel_001.

1. Current measurement
2. Selected measurement (Measurement number 1)
3. Saved measurement with the defined name

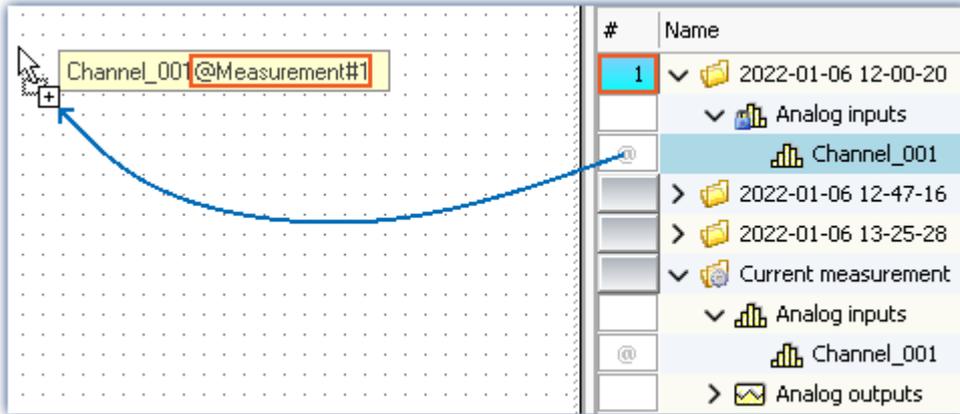
Browsing data

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](#)"^[809]).

Variable linkage by Drag&Drop

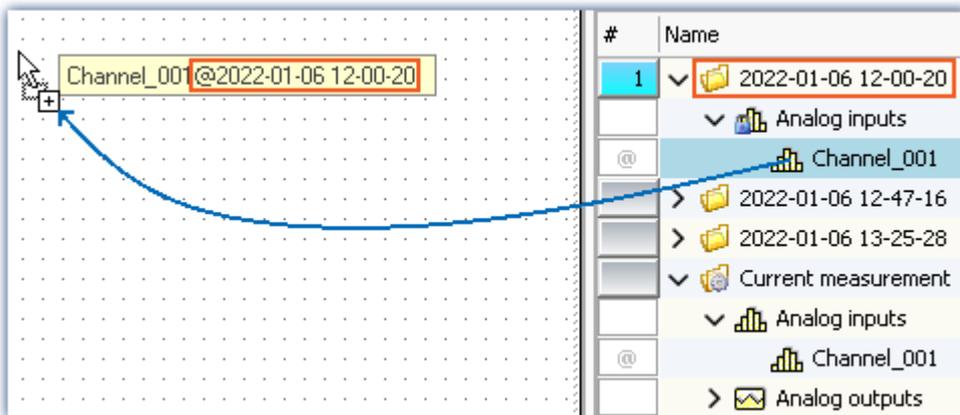
If you have opened a measurement in the Data Browser, the "@" Symbol () appears in the **Numbering column**.

- 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](#) ⁸⁵⁰.

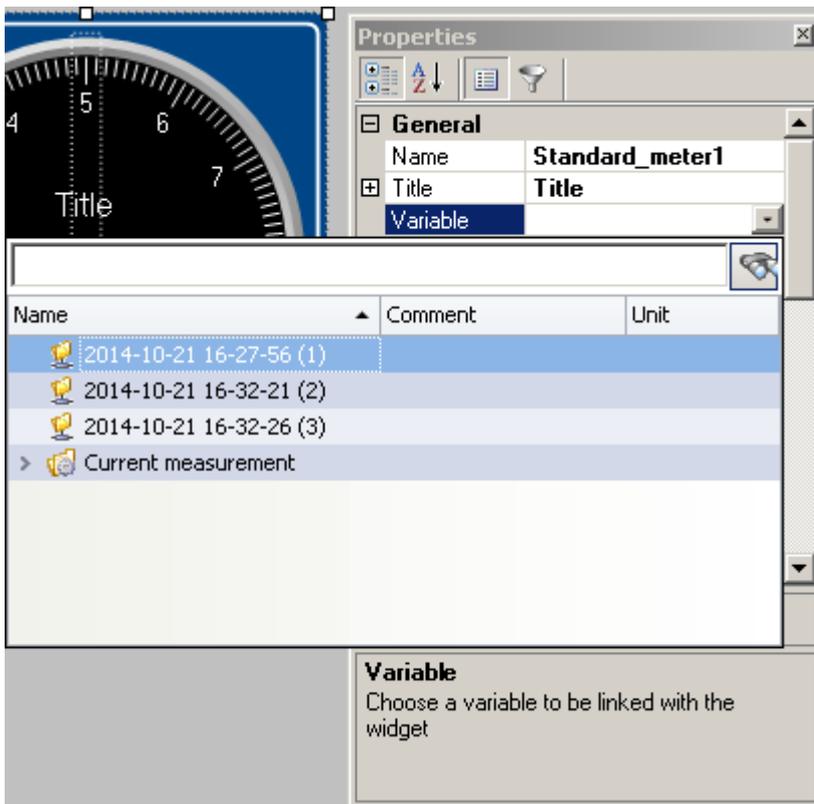
You can establish the link with the variable by means of the Widgets's properties.

- 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

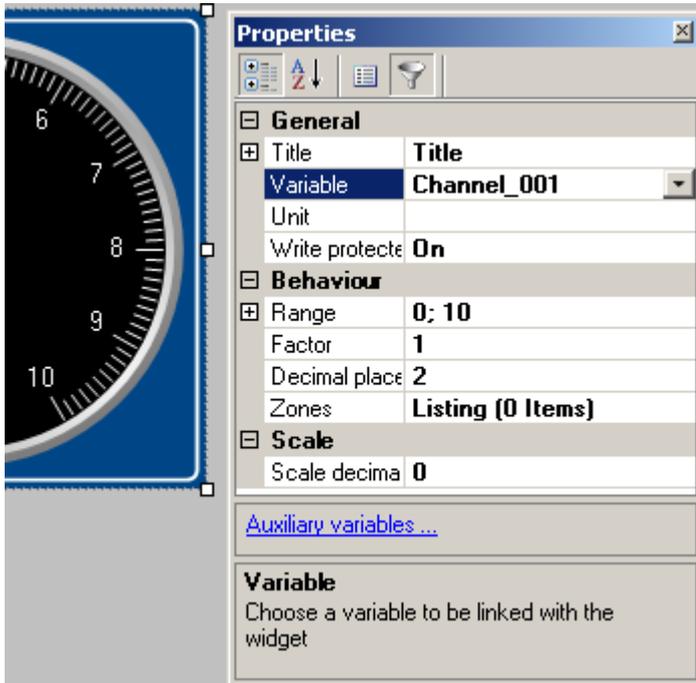


- Select the desired variable

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](#) ⁽¹¹³⁴⁾)

13.9 Information and tips

13.9.1 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.

13.9.2 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 WAVE from the virus scan.

See: Recommended virus scanner [settings](#)³⁵

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](#)"⁸³¹

13.9.3 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.

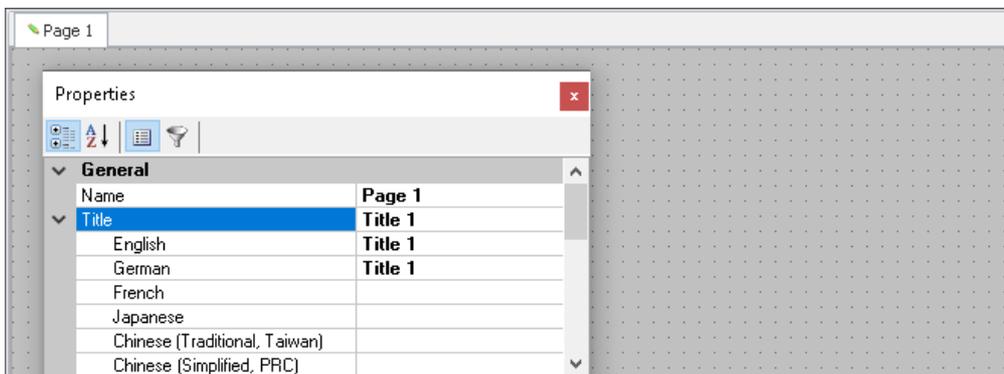
Option: Deactivating multi-language text input

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**.



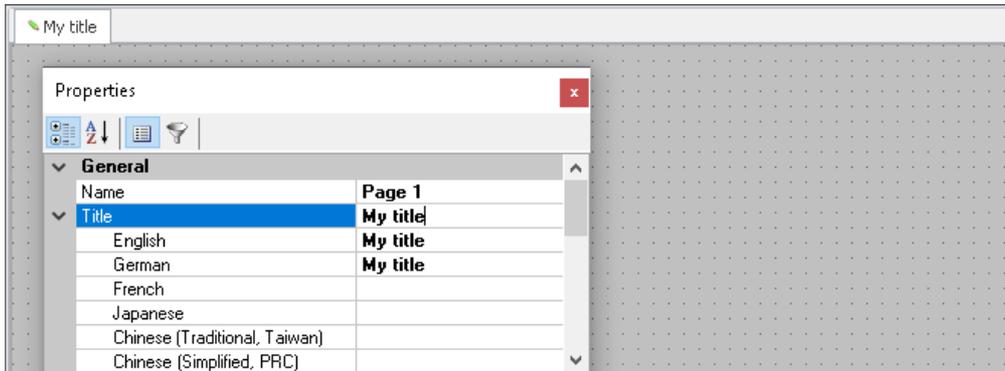
Example - Title of the Panel page

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*".

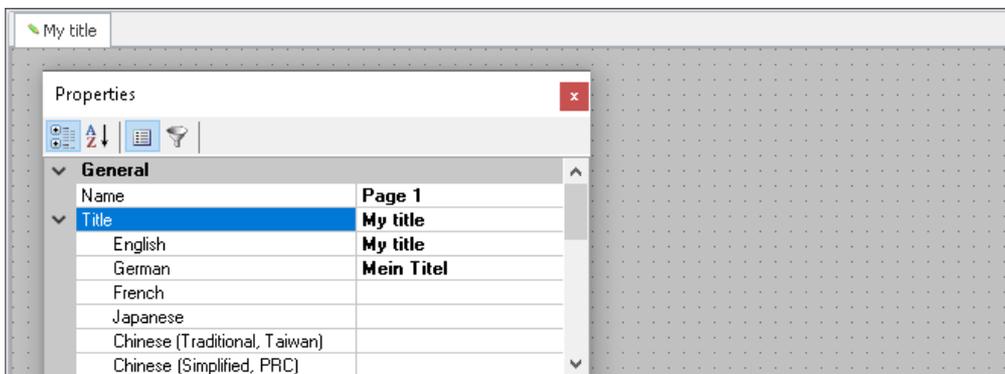
Case 1: You change the title in the main branch (all existing titles are the same):

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.



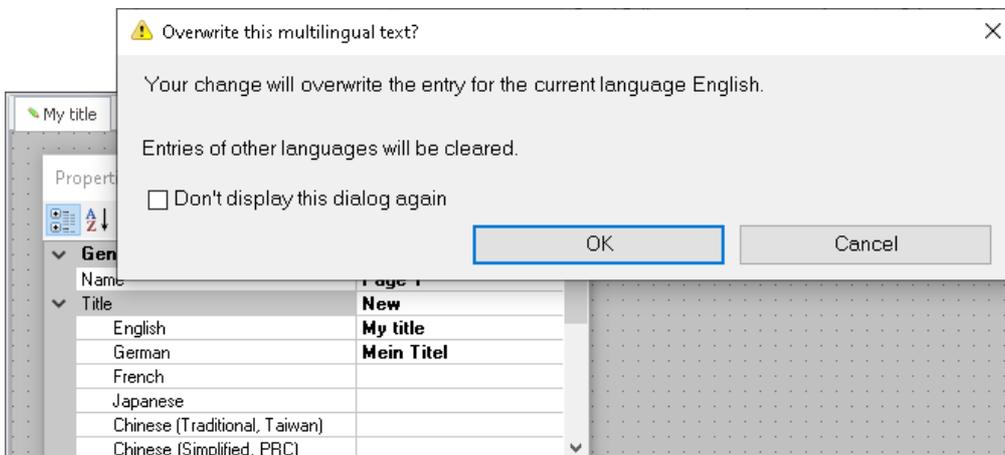
Fall 3: You change the title in the main branch while the titles in the other languages are all different:

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.



14 Sequencer, events and commands

Using imc WAVE, 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 WAVE components:

Component	Description
Panel	You can link events with commands by means of the " Event Dialog ^[1142] ". The commands are executed when the event occurs. Here, you can use selected commands.
Command batches	On the " Command batches ^[92] "-page.

The sequence

Name	Hint
Checkbox1	
Switched Off	Occurs when the switch has been switched off
#01 Execute menu action: Stop (Stop measureme...	Execute menu action: Stop (Stop measureme...
#02 Run imc FAMOS sequence	...
#03 Browse in workspace - Panel: Panel/Report	
Switched On	Occurs when the switch has been switched on
#01 Execute menu action: Start (Start measurem ...	Execute menu action: Start (Start measurem..
#02 Browse in workspace - Panel: Panel/Messung	

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 WAVE

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 WAVE.

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 WAVE.

Many application cases are described in the context of imc STUDIO. Some of these are also applicable to imc WAVE.

Chapter overview

Synopsis	Section
Creation and configuration of a sequence of commands. Adding, moving and making settings for commands.	Creating a sequence of commands ^[1145]
Events relating to Panel-switches	Event Dialog ^[1142]
Description of all commands.	Command Reference ^[1148]

14.1 Sequence table

The sequence table contains multiple columns.

Status	Command	Comment	Enable	Stop on error
Done	Set variables		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Columns - Command sequence table

Column	Description
--------	-------------

Status In this column, the commands' status is indicated.

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 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 command was concluded successfully.
- **Error:** The command was concluded by an error.

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 ^[1141] 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 ^[1141] With this option, the [sequence is halted](#) ^[1144] 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](#) ^[1144] if an error occurs.

Link events with commands					
Status	Name	Comment	Enable	Stop on error	
▼	Button1				
▼	Clicked	Occurs when the ...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	#01 Run imc FAMOS sequence:	...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	#02 Export parameters		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	#03 Show Panel page as dialog:		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	#04 Export Panel page		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

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.	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	#01 Run imc FAMOS sequence	...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	#02 Export parameters	▼	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	#03 Show Panel page as dialog:	▼	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	#04 Save Panel page as PDF	▼	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
>	Project	Project-specific events		
▼	Sequencer	Experiment-specific events of the Sequencer		
	Device_AfterCheckConfiguration	After checking the configuration of a device	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Device_AfterRequestConnect	After trying to connect to a device	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Device_AfterRequestDisconnect	After trying to disconnect from a device	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
▼	Device_BeforeCheckConfiguration	Before checking the configuration of a device	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	#01 Import variable	▼	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Device_BeforeCreateDiskStart		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

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.

14.2 Event Dialog

Certain **events** can be **linked with commands**. The commands are performed **upon occurrence of the event**.

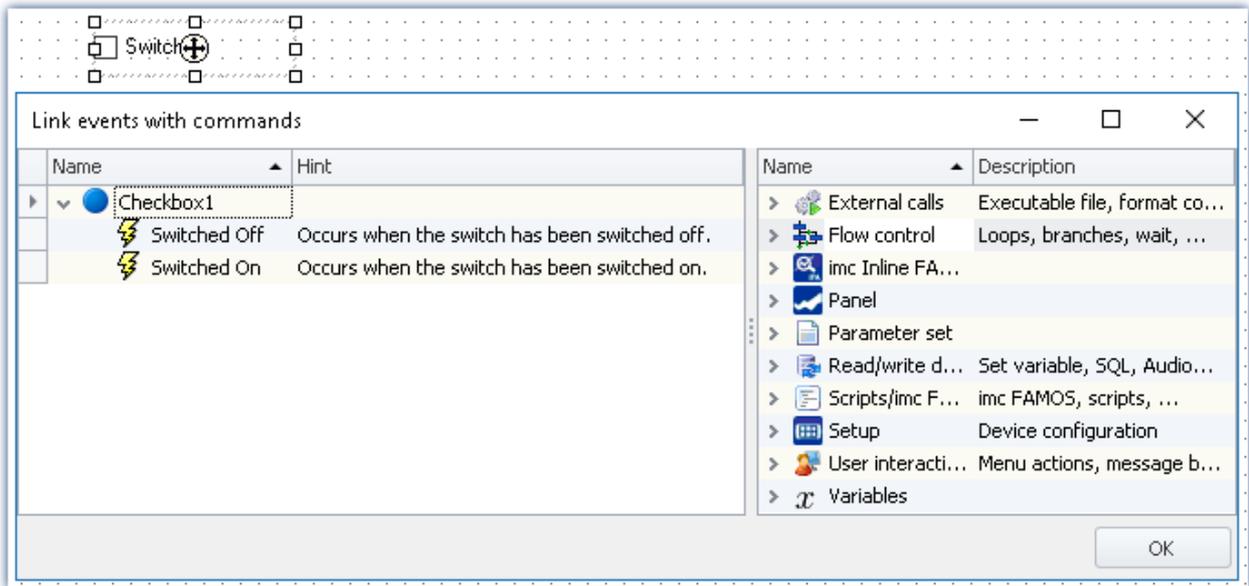
For configuring the event, the "event dialog".

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*

14.3 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](#)¹¹⁹⁷" 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.

14.4 Creating a sequence of commands

The following examples are illustrated in the sequence table. However they also apply to the "[Event dialog](#)"¹¹⁴².

To add a command to the sequence table, there are two options:

by Drag&Drop	<ul style="list-style-type: none"> • Use Drag&Drop to move the command from the "Commands" tool window to the desired position. • Release the mouse button.
by the context menu ¹¹⁴⁷	<ul style="list-style-type: none"> • Open the context menu of the desired position and select a command.

You can place a command at different positions.

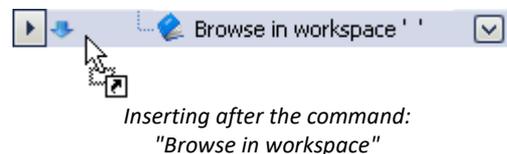
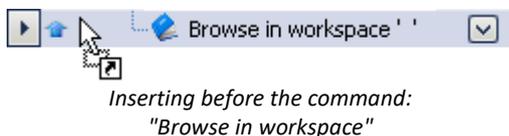
	Before the current position
	Below or within the current position: events (following also called group)
	After the current position

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.

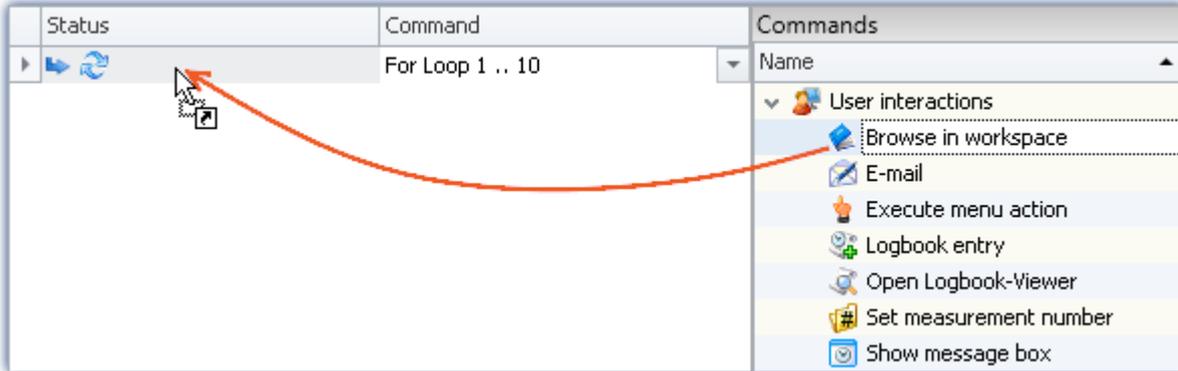
Adding commands before or after existing commands

Commands can be added either **before** or **after** an existing command. To do this, move the mouse either a little upward/downward.



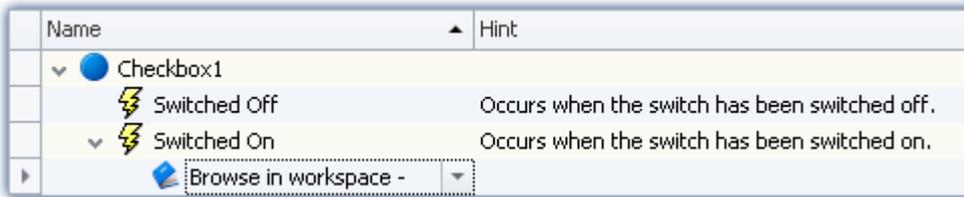
Adding commands to a group

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](#)" ¹¹⁴⁸.

14.5 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 ^[1145] 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.

14.6 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](#)^[1147]" 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.

14.7 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

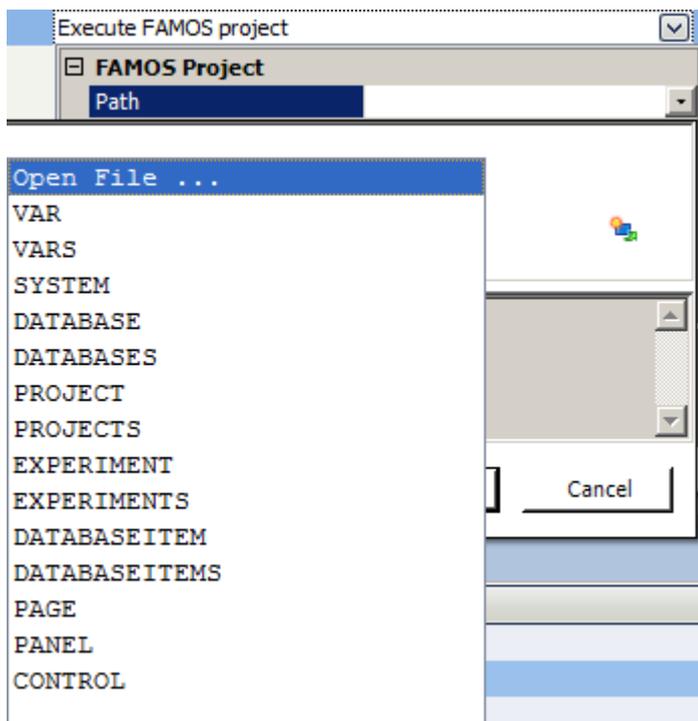
Note

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.

14.7.1 Data analysis and Scripting

14.7.1.1 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*.

14.7.1.2 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](#)".

14.7.1.3 Run Python code file

The Python command supplies functions which provide a bridge to the Python programming language. imc WAVE 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

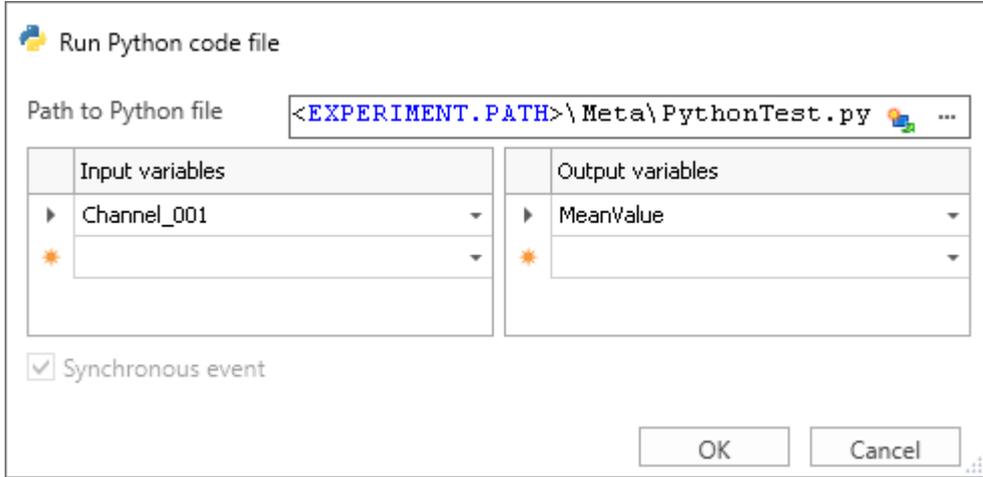
Software	Compatible version	Installation	Notes
Python 64-bit	3.11, 3.10, 3.9, 3.8	required	Support is provided exclusively for the Python reference implementation from the " <i>Python Software Foundation</i> " (CPython) in one of the versions listed below; these can be downloaded and installed from https://www.python.org/ .
NumPy 64-bit	1.23, 1.20, 1.19	required	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.



Properties of the command "Run Python code file"

Parameter	Description
Path to Python file	<p>Path to the executable Python code file (*.py).</p> <p>Ideally, the file is saved in the Meta-folder. The content of this folder is also included when performing Save As/Export.</p> <p><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.</p>
Input variables	<p>The variables entered are passed to the Python-sequence. The variable names must match those in the Python-sequence. The spelling is case sensitive.</p> <p>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.</p>
Output variables	<p>The variables calculated by Python are assigned to imc WAVE via the table. The variable names must match those in the Python-sequence. The spelling is case sensitive.</p> <p>If the variable does not yet exist in imc WAVE, it will be created.</p> <p><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.</p>

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 WAVE 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](#)"^[173])

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.

Format for transfer 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']
```

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 WAVE

The data type of the imc WAVE-variable generated is determined automatically from the data type of the Python-variable.

Here, the following rules apply:

Python data type	imc WAVE 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 WAVE 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 WAVE 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:

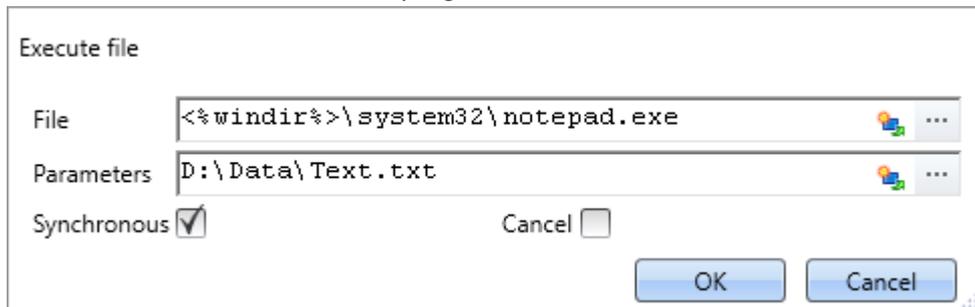
```
imcVariables = ["Channel_001", "Channel_002", "Channel_003"]
```

```
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'])
```

14.7.2 External calls

14.7.2.1 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.

14.7.2.2 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](#)"⁷⁸⁶

Export formats

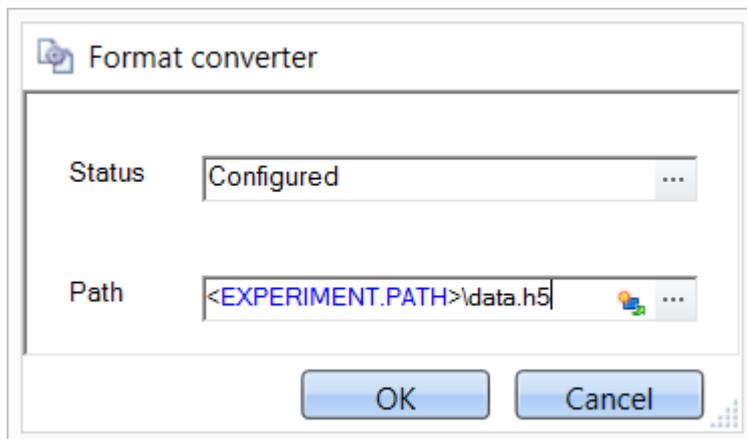
The same export formats are supported which are also offered by imc FAMOS. These include:

- 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 TAFFmat and TEDAM, DIAdem TDM and TDMS
- Binary Universal File Format
- Universal File Format (UFF)

14.7.2.2.1 Format converter as command

Calling

Via the Sequencer: "External calls" > "Format converter":



Configuration of the command: Format Converter

Status

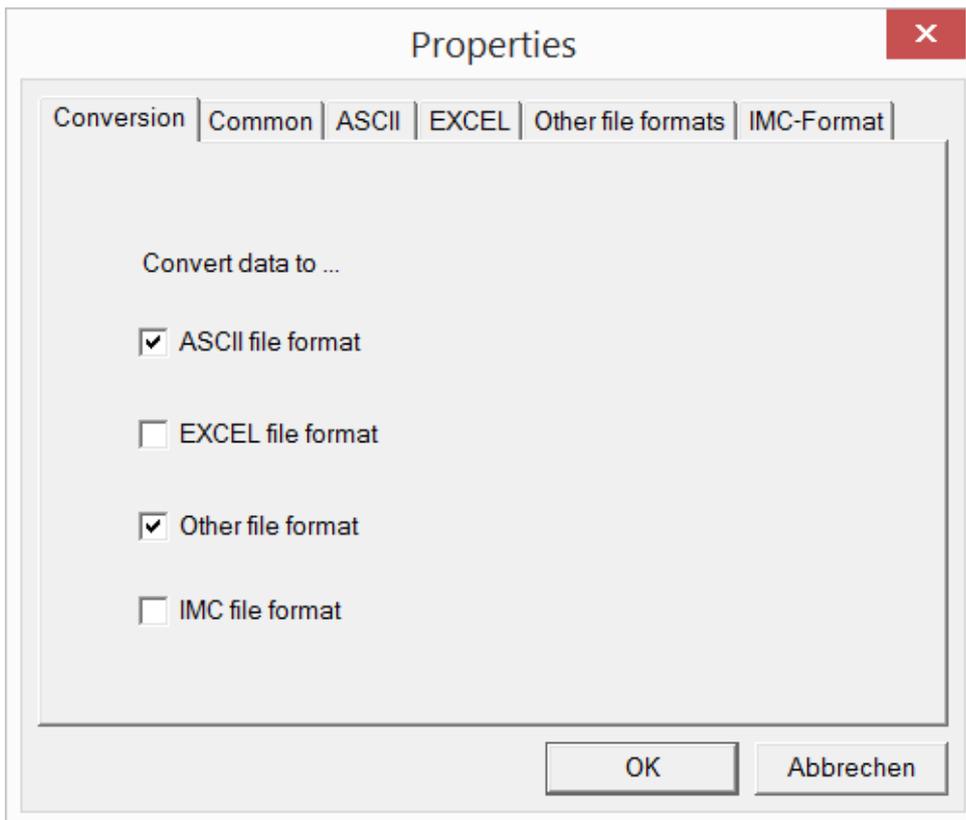
Immediately upon opening the command's configuration, the status is set to "not configured".

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.



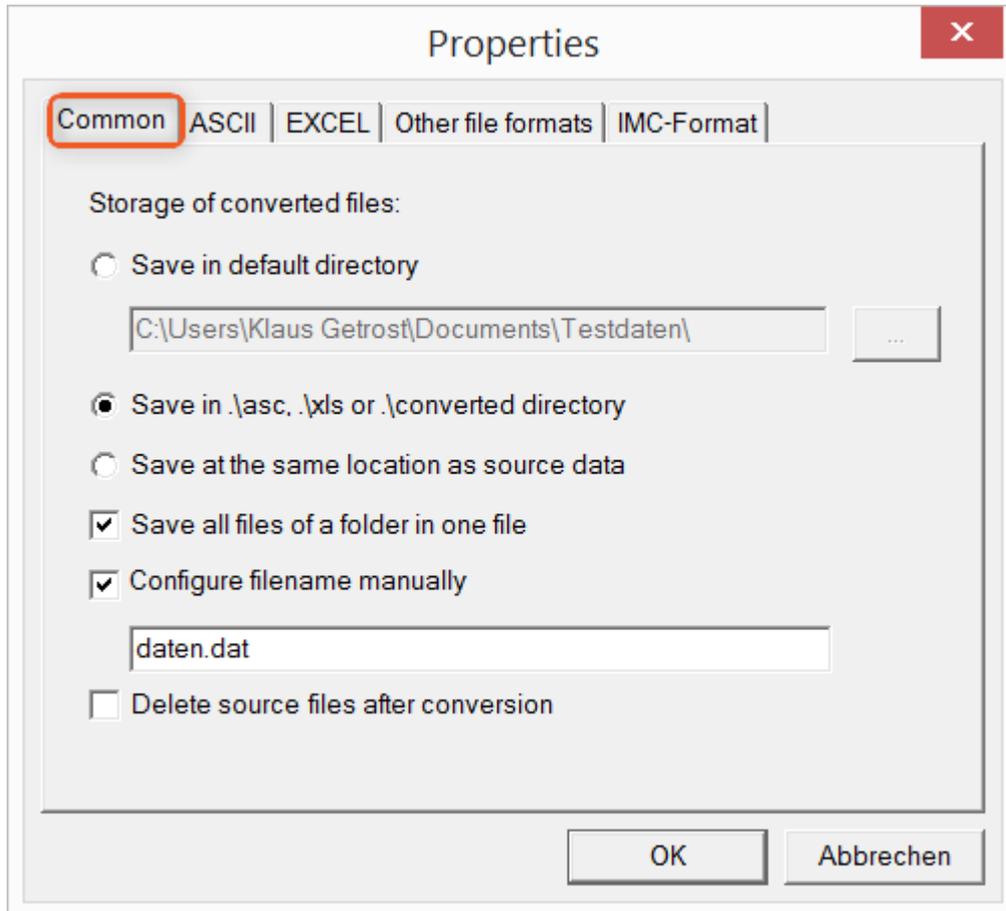
Setting the Format Converter

14.7.2.2 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 re saved on the same **level of the tree diagram as the source data, without subfolders**.

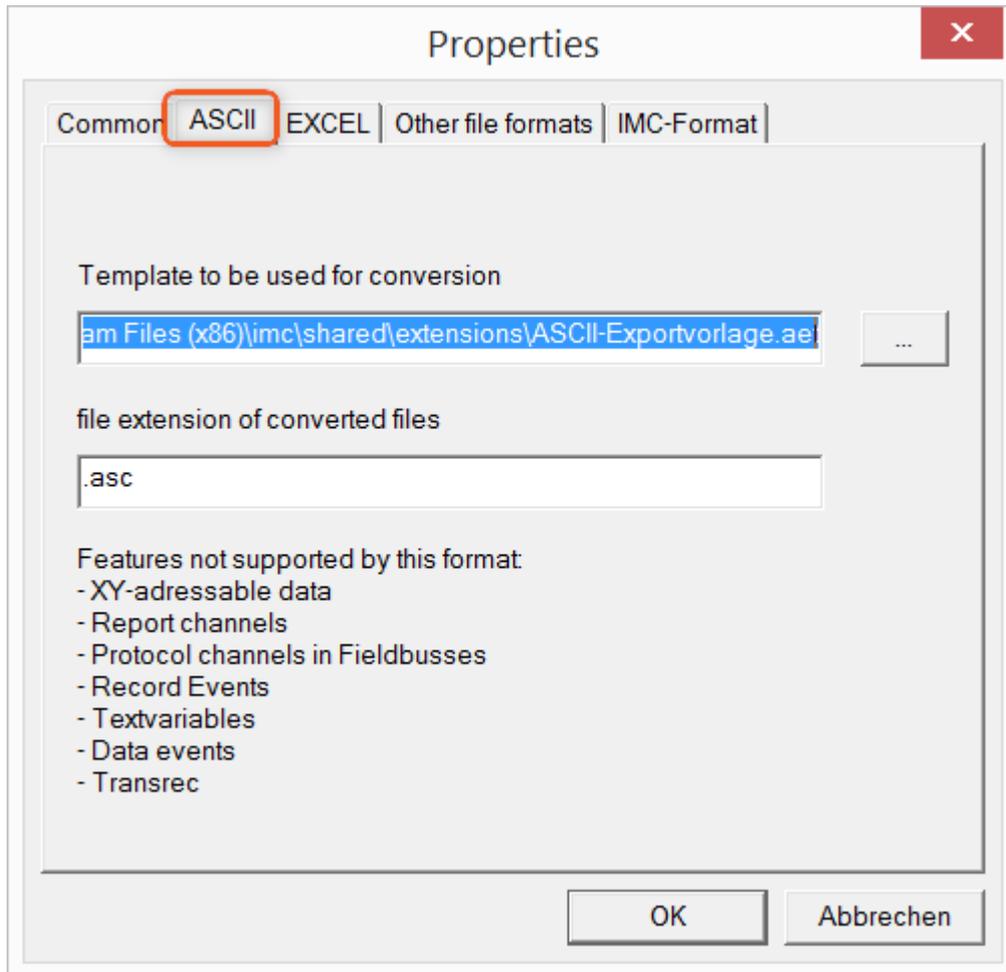
Manner of data saving:

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.

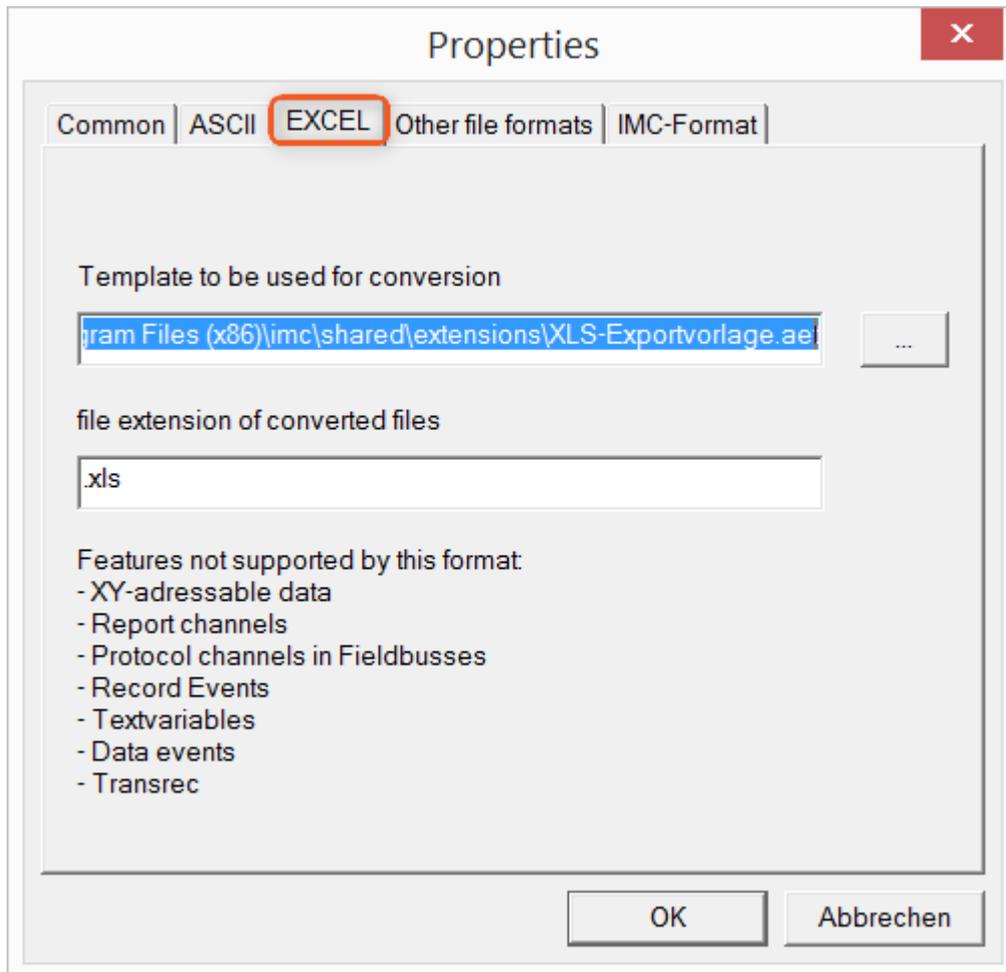
Special variable types which can **not** be exported as ASCII data are listed on this tab page.

! 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](#)^[1160]. By default, the file extension is "XLS", but can be arbitrarily specified here.

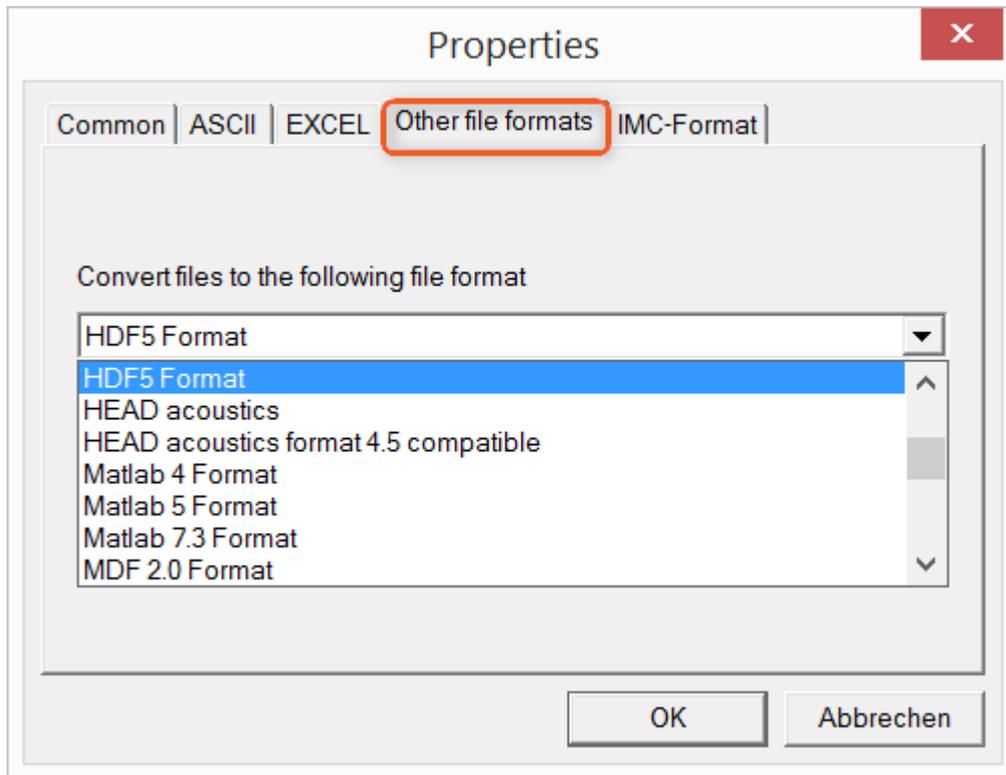
Variable types which are **not** exported correspond to those for [ASCII-export](#)^[1160].

! 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



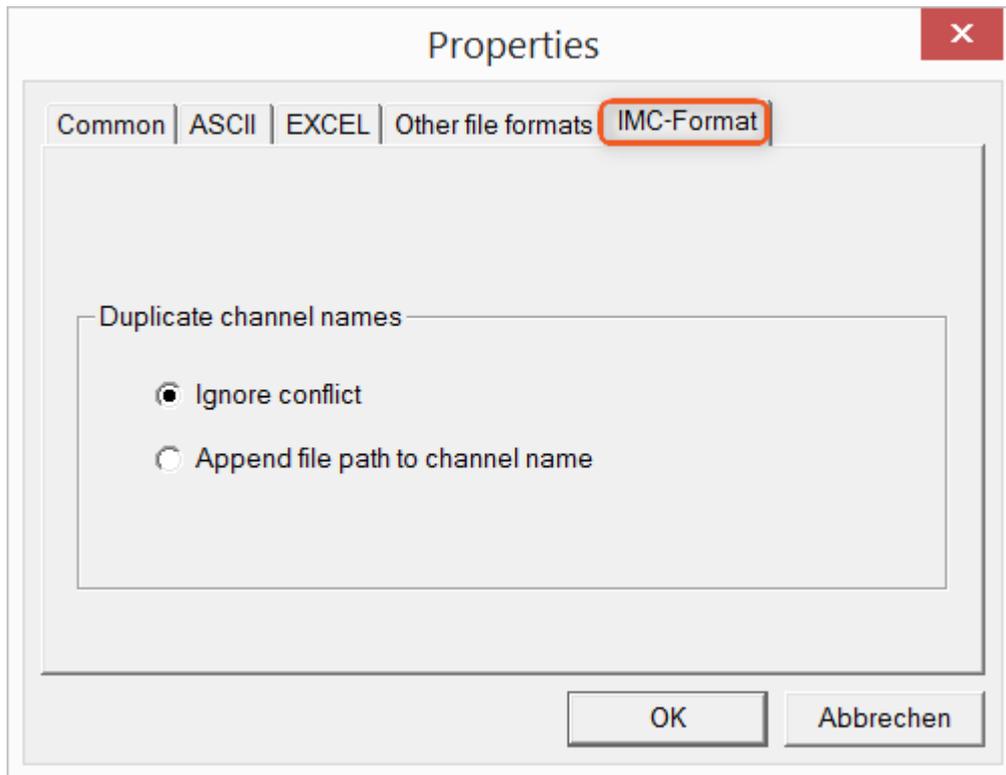
Convert files to the following file format:

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

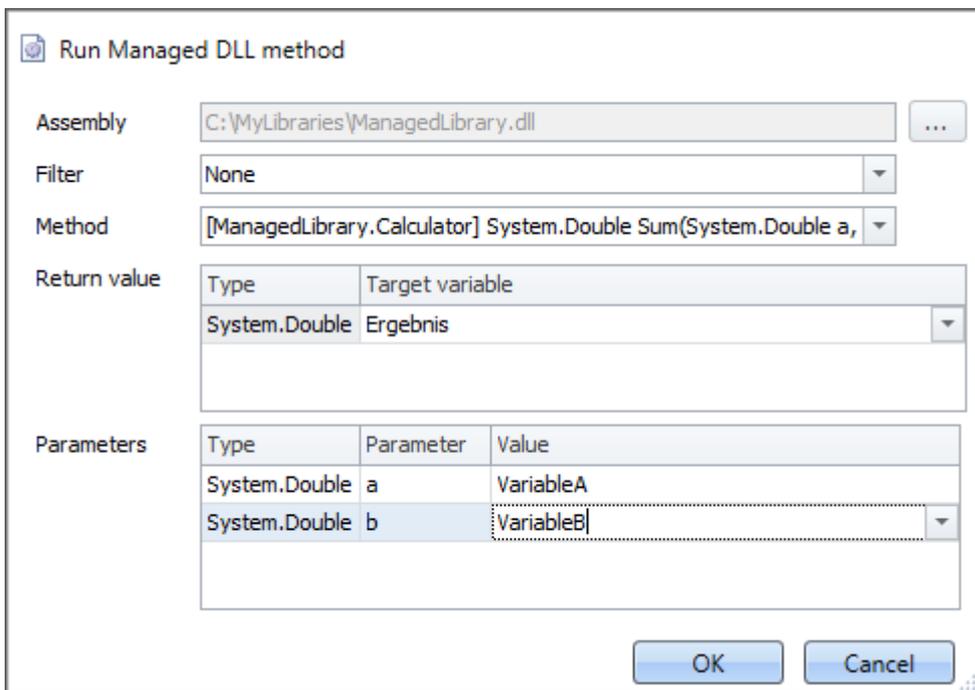
14.7.2.3 Run Managed DLL method

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.

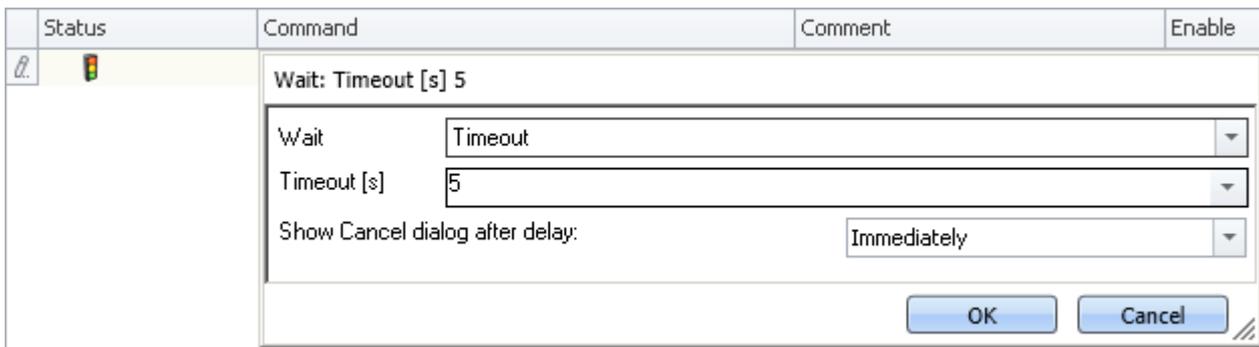
14.7.3 Flow control

14.7.3.1 Stop Sequencer

The command is not used in imc WAVE.

14.7.3.2 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*.



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 "==".

14.7.4 imc Inline FAMOS

14.7.4.1 Import imc Inline FAMOS source code

The command is not used in imc WAVE.

14.7.5 Measurement

14.7.5.1 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](#)"^[335]

set - to a fixed measurement name

The dialog box titled "Set measurement number" contains the following fields:

- Measurement number:** A dropdown menu with the selected option "set - to a fixed measurement name".
- Symbolic measurement name:** A text field with the label "Measurement#" and the value "1".
- Measurement name:** An empty text field.

At the bottom right, there are "OK" and "Cancel" buttons.

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 with the selected option "set - to a fixed index".
- Symbolic measurement name:** A text field with the label "Measurement#" and the value "1".
- Sorting of measurements:** A dropdown menu with the selected option "Time - ascending".
- Index:** A text field with the value "1".

At the bottom right, there are "OK" and "Cancel" buttons.

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.

14.7.6 Panel

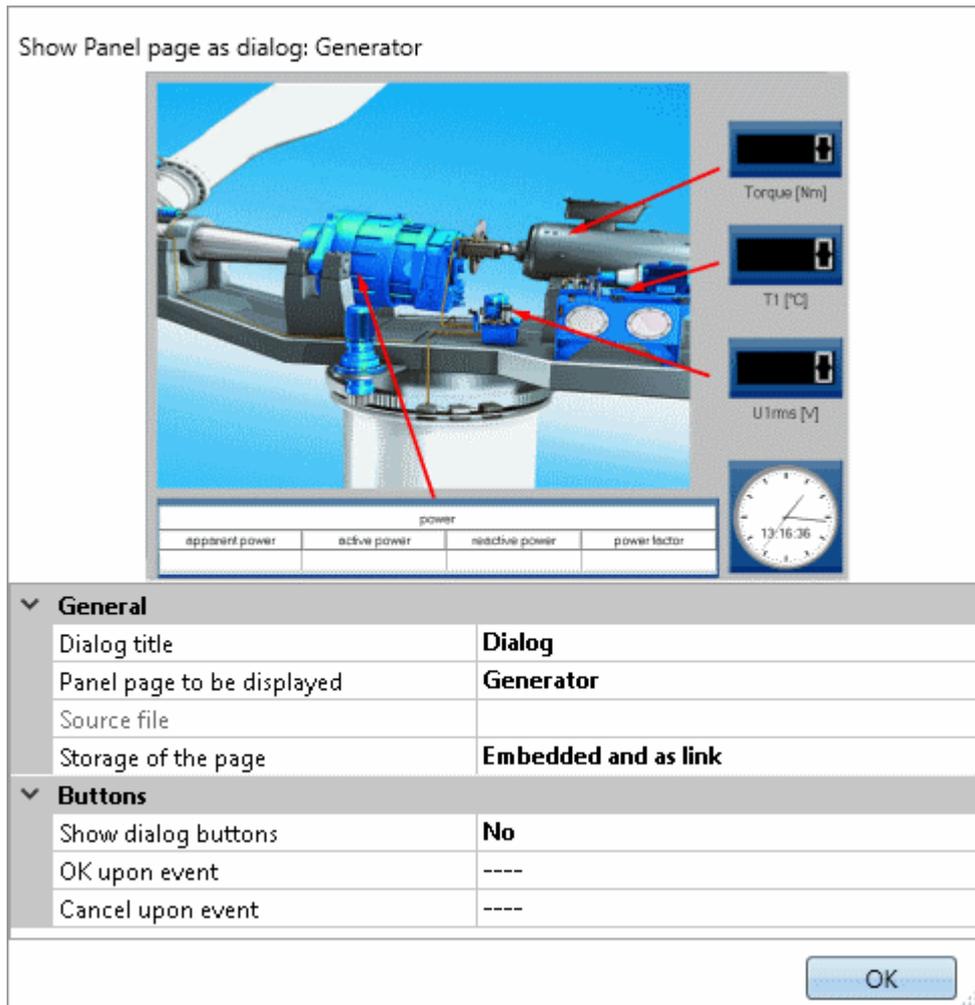
14.7.6.1 Show Panel page as a dialog

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.



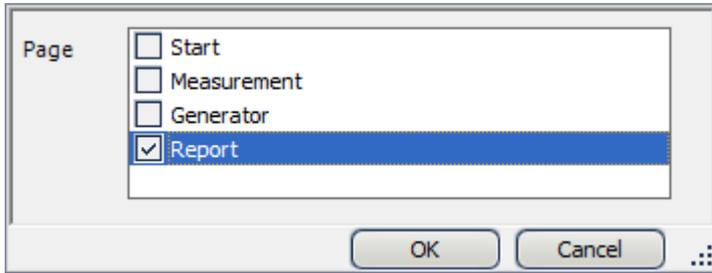
Properties of the command "Panel-page as dialog"

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	<p>Here you can define to which changes to the Panel pages the system responds.</p> <p>Embedded: This page is embedded into the command. There is no connection to the page in the Panel.</p> <ul style="list-style-type: none"> • If this page is deleted, the command continues to function. • Changes to the page are not applied for the command. <p>As link: Only a reference to the existing Panel page is saved.</p> <ul style="list-style-type: none"> • 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. <p>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.</p> <ul style="list-style-type: none"> • 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	<p>Separate "OK" and "Cancel"-buttons in the dialog can be shown or hidden.</p> <p>Yes: The buttons are displayed upon execution. The parameters for the OK- and Cancel-buttons are shown for the purpose of configuring the button.</p> <p>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".</p>
OK upon event	<p>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.</p> <p>For this purpose, the switch must be present on the page.</p>
Cancel upon event	<p>Analogous to "OK upon Event". In this case, the dialog is closed with "Cancel".</p>
OK button	<p>This field is only enabled if "Show dialog buttons" is set to "Yes".</p>
Visible	<p>Shows/hides the button.</p>
Active	<p>The button can be deactivated in particular situations.</p> <p>Always: The button can be actuated at all times.</p> <p>If all inputs are valid: The button can only be actuated if all inputs are valid. For this purpose, embedded Setup pages are required.</p> <p>For details on this topic, see the chapter: "<i>Setup-Layout</i>" > "<i>Using Mandatory Boxes</i>".</p>
Label	<p>The button's caption.</p>
Cancel button	<p>This field is only available if "Show dialog buttons" is set to "Yes".</p> <p>For the Cancel-button, the same parameters are available as for the OK-button (Visible, Active, Label).</p>

14.7.6.2 Print Panel page

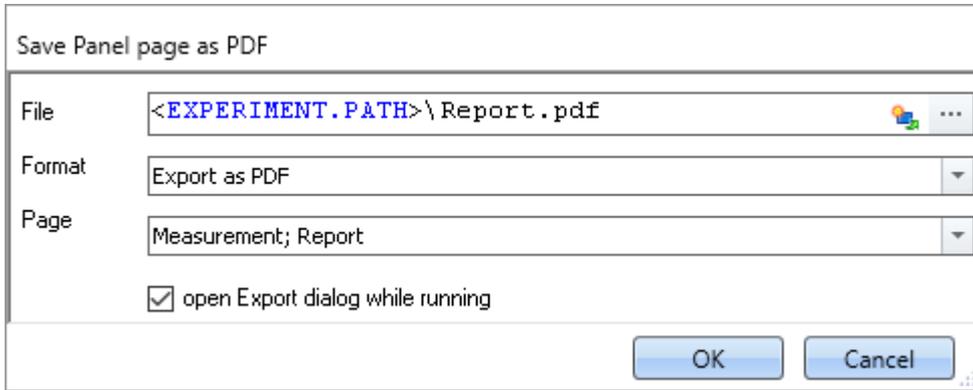
You can print out Panel pages. By this means, it is easy to compose reports. To do this, create Panel pages which are displayed as a report.



Example for printing out a Panel page

14.7.6.3 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 border="1" data-bbox="523 1355 1394 1733"> <thead> <tr> <th>Format</th> <th>Description</th> </tr> </thead> <tbody> <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> </tbody> </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.

14.7.6.4 Import Panel page

You can import a Panel page which was previously exported in DBV format.

14.7.6.5 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.
```

14.7.7 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](#)"

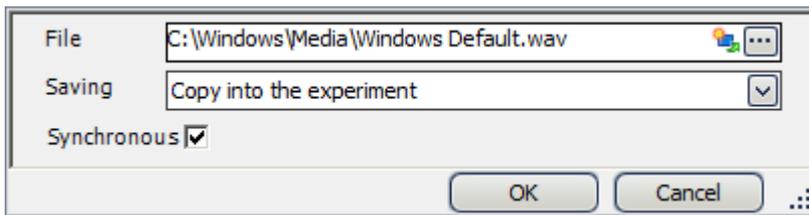
 Note

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.

14.7.8 Read/Write data

14.7.8.1 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.

14.7.8.2 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

For purposes of traceability, the measurement settings (in force for the current experiment) are saved along with the corresponding measurement results in the database. If this option is active, the current experiment's settings are exported in parallel with the data export.

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

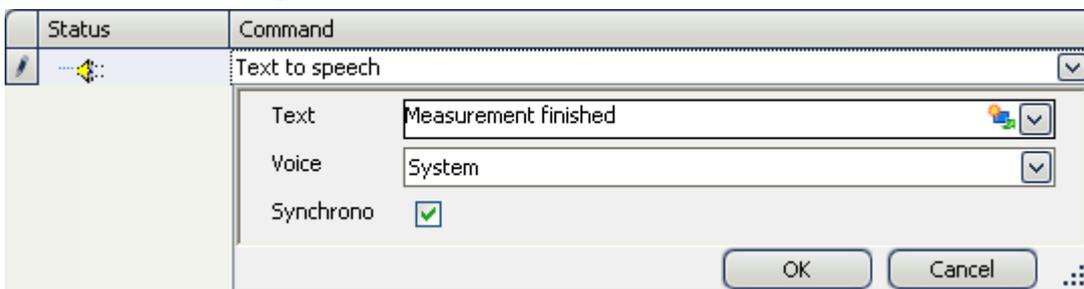
Here you set which button is "pre-selected". If the command is executed, the selected action can be run by clicking on the Enter-key.

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.

14.7.8.3 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.

14.7.9 Setup

14.7.9.1 Measure a user-defined characteristic curve

Characteristic curves of sensors provide the ability to convert measured electrical readings into physical values by referencing a plot of the relationship between those variables – the characteristic curve. Using this command, a characteristic curve is saved in a dat-file. This file can then later be imported as a supplemental file for imc Online FAMOS, for example.

Reference

Sensors, Characteristic Curves and TEDS

A description of the sensors and how to generate their characteristic curves using the command is presented in the section:

"Setup pages - Configuring device" > .. > "Sensors, Characteristic Curves and TEDS" > ["Create a new sensor characteristic curve"](#)

14.7.9.2 Execute device action

By means of this command, you can perform a variety of device actions which are also available via the action buttons on the Setup pages. For instance, you can perform bridge balancing or a shunt calibration.

The list of available actions is unfiltered. For this reason, you also find actions here which have no effect unless you make additional settings on the Setup pages, or which have no effect as they are located in the list.

Parameter	Description
Enumerator Class	In an enumerator-class, the association with "object-groups" is defined (e.g. analog channels, Display variables, devices, ...). See also: " Enumeration Classes ^[178] ".
Action column	List of possible actions to be performed (corresponding to the buttons on the Setup pages)
Action	Some action columns contain various possible actions to be performed. For instance, with the action column: "Balance action", it is possible to perform a taring or bridge balancing.

Filter type Here you define the target for which the action is to be performed. In many cases, multiple targets are available, e.g. all analog channels in the case of a balancing action.

All rows: The action is performed for all targets possible. In some cases this can lead to error messages being posted if the particular action cannot be performed for certain targets.

Selection: Here, select particular targets on which the action is to be performed. If any new targets are added, they are not already automatically selected.

Filter: Create a filter according to whose criteria the targets are selected. Here are some examples:

Summary	Filter formulation	Interne designation
all active channels	[Channel status] Equals Active	([eStatus] = 1)
Only one balancing type	[Balance action] Equals Bridge	([eBalanceAction] = 2012)
Only one particular module type	[Module type] Equals Bridge, current and voltage amplifier and temperature measurement unit "UNI2-8"	([eModuleType] = 47)
Only channels belonging to a defined coupling	Coupling Is any of (Full bridge, Half bridge, Quarter bridge)	[eCoupling] in (5, 3, 6)

As an illustration, the following examples from practical applications



Example

Balancing 1

Perform a balancing action: for all channels currently selected

Parameter	Setting
Enumerator Class	Channel
Action column	Balance action
Action	Currently selected
Filter type	All rows (here the use of a filter is recommended: see further below)

For all analog channels, whichever of the balancing types "Bridge", "Taring" or "Manufacturer's calibration" which is set on the Setup page: "Balancing" is performed.

Note: Since passive channels will report an error in consequence of this action, it is advisable to define a filter for all "Active" channels:

```
[Channel status] Equals Active
([eStatus] = 1)
```



Example

Balancing 2

Perform bridge balancing: for all active channels of the module type: "Bridge, current and voltage amplifier and temperature measurement unit "UNI2-8""

Parameter	Setting
Enumerator Class	Channel
Action column	Balance action
Action	Bridge
Filter type	Filter: Only active channels and only channels belonging to the amplifier: "UNI2-8" (([eModuleType] = 47) And ([eStatus] = 1)) (here, it is recommended to use an additional filter: Only channels for which a suitable coupling is set: see further below),

For all analog channels having the appropriate amplifier, a bridge balancing operation is performed.

Note: Since not all channel settings allow bridge balancing, it makes sense to modify the filter even more. Depending on the amplifier used, the coupling may need to be set to full bridge, half-, or quarter-bridge:

```
(([eModuleType] = 47) And ([eStatus] = 1) And [eCoupling] in (5,3,6))
```

**Example****Shunt calibration**

Perform a shunt calibration: for all channels selected

Parameter	Setting
Enumerator Class	Channel
Action column	Shunt calibration
Action	--
Filter type	Selection

**Example****Starting measurement**

Start a measurement: for all devices

Parameter	Setting
Enumerator Class	Device
Action column	Device control
Action	Start
Filter type	All rows (starting a specified device is also possible)

All devices are started.

Note: The difference from the menu action is that correct preparation (reconfiguration) is always performed previously.

Instead of starting measurement this way, please use the command: "[Execute menu action](#)¹¹⁹⁰".



When the action "Start" is performed using the command "Execute device action", no calculations or analyses performed on the PC side, such as imc Inline FAMOS-calculations, are started. Only the devices are started.

14.7.9.3 Import MFB configuration

The command enables the **import** of Fieldbus-configurations. The following Fieldbusses are supported:

- **CAN:** *.cba; *.dbc
- **ARINC:** *.idb
- **ECAT-IF:** *.imcecatif.xml
- **AFDX:** *.xml

**Reference****Description of the configuration files**

The files are generated using the Fieldbus-Assistant. For a detailed description, please refer to the associated chapters of the manual.

Parameter	Description
Import file	Specification of the file to import, including the pathname.

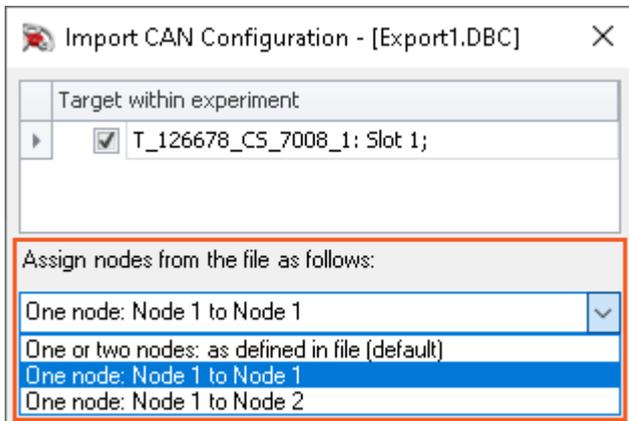
Parameter	Description
Mapping	<p>Either specify the device so that no additional prompt will appear upon execution of the command, or wait to select the device until when you run the command.</p> <p>Show mapping table: Upon execution, a prompt appears which requests you to specify the device to use as the target. You can predefine the mapping table.</p> <p>Use mapping table: Specify the desired device by means of the additional parameter "<i>Mapping table</i>".</p>
Mapping table	Here, select the desired device. The target is a device having an appropriate module.

Menu item: "Import/Export":

By means of this menu item, you can also import MFB-configurations. Here, the same options are available for your use.

CAN-import - Node selection

If the configuration only contains information on one node, you can select to which node to perform the import.



Selection	Description
One or two nodes: as defined in file	<p>The file's content is processed sequentially. The first node (belonging to the file) is placed on the first node (in the Assistant). If there is any second node in the file (only cba), then this is set on the second.</p> <p>If there is no second node in the file, the configuration of the second node is deleted from the Assistant.</p>
One node: Node 1 to Node 1	<p>The file contains only one node - The content of the file is imported to the first node in the Assistant.</p> <p>The configuration of the second node remains intact.</p>
One node: Node 1 to Node 2	<p>The file contains only one node - The content of the file is imported to the second node in the Assistant.</p> <p>The configuration of the first node remains intact.</p>

14.7.9.4 Reassignment of module positions

The module addresses are re-assigned. This matches the working of the function "Reassign" in the dialog "Module Properties". The existing module numbers are overwritten according to a gapless re-numbering process.

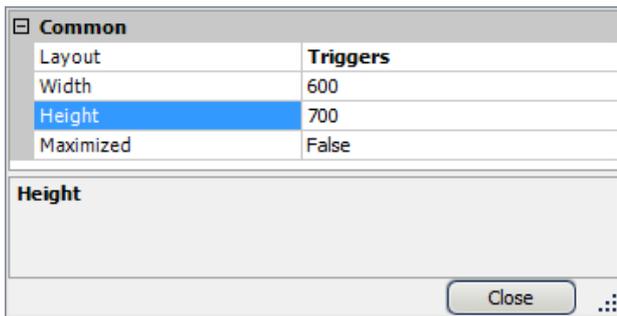


Reference

For more information on the module position and re-assignment, see the chapter: "[Module properties](#)".

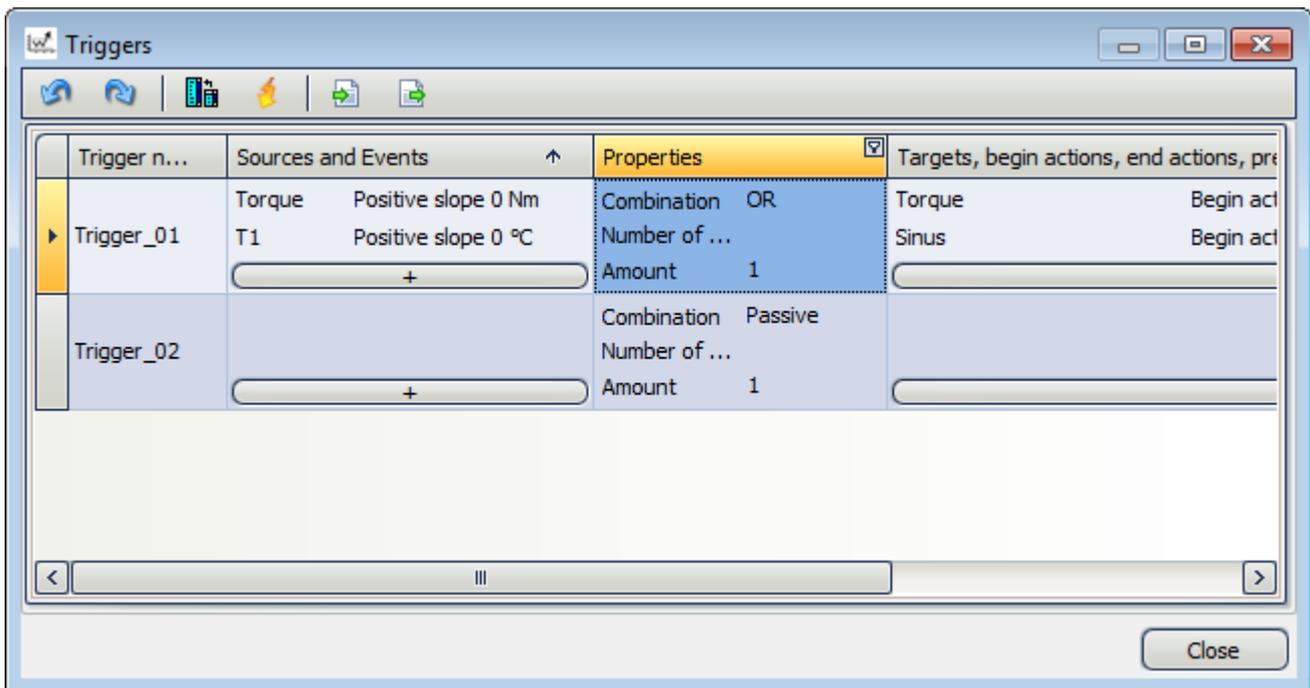
14.7.9.5 Display Setup dialog

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

14.7.9.6 Import synthesizer-/controller configuration

This command enables **import** of synthesizer and controller configurations for the **Synthesizer-module**.



Reference

Description of the configuration files

The files are generated by means of the Synthesizer-Assistant. A detailed description is presented in the Synthesizer user's manual.

Parameter	Description
Import file	Specification of the file to import, including the pathname.
Mapping	<p>Either specify the device so that no additional prompt will appear upon execution of the command, or wait to select the device until when you run the command.</p> <p>Show mapping table: Upon execution, a prompt appears which requests you to specify the device to use as the target.</p> <p>Use mapping table: Specify the desired device by means of the additional parameter "<i>Mapping table</i>".</p> <p>auto: The system searches for an appropriate target for the import. Only devices having a suitable module can be the target.</p> <ul style="list-style-type: none"> • The import file contains a statement of the source device. If this device is selected, it is used. • If that device is not selected, the system uses a device among the ones which are selected which has the same number of slots. • If no such device is available either, the system selects the next best one. <p>Assignment of the slot is performed in a similar way. If the same numbers are available, they are used. Otherwise the assignment procedure uses other numbers in numerical succession.</p>
Mapping table	Here, select the desired device. The target is a device having an appropriate module.

14.7.9.7 Load/Delete synthesizer signal file

Using imc STUDIO you can load files for the Synthesizer module into the device, or delete them from the device. By means of an index you can import various segment-files. With an appropriate device, you can play back these segments in succession.

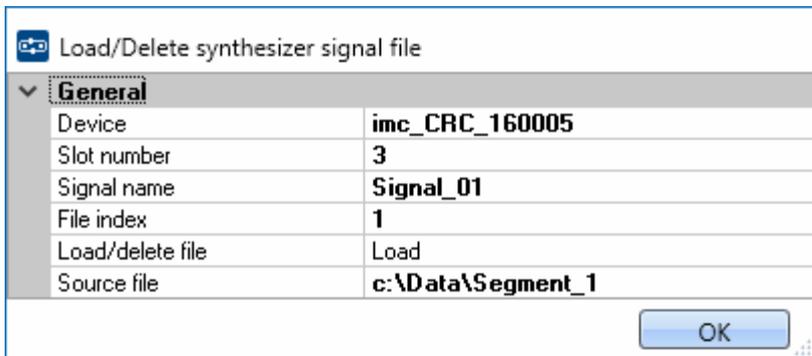
In the Properties you select the Synthesizer output. The outputs are distinguished by their respective slot numbers and signal names.

Parameter	Description
Device	Here, select the target device. The device requires a Synthesizer-module
Slot number	Synthesizer slot in the device
Signal name	See the Synthesizer-configuration for the signal name.
File index	A signal's segments can be changed during the measurement. By means of the parameter, the segment is assigned an index in reference to which it is accessed. The index indicates the number of the segment which is selected by means of the pv-variable <SignalName>_Slot<Number>_NextFile.

 For an detailed description, see the section "*Managing the segment files*" in the "*Synthesizer Manual*".

Load/delete file	<ul style="list-style-type: none"> • Load: Transfers the source file to the device • Delete: Deletes the file having the index specified from the device
------------------	--

Source file	Here, select the file you wish to import.
-------------	---



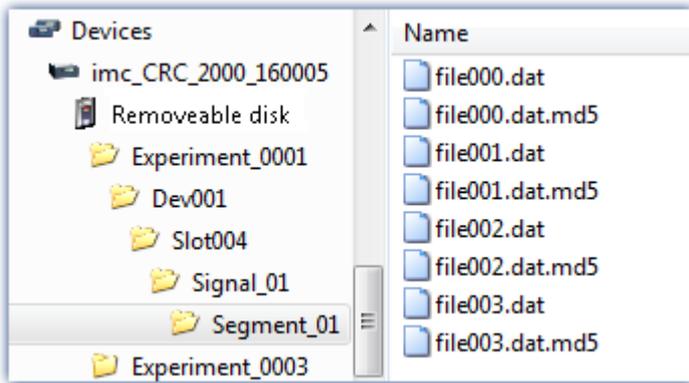
Assignment of the segment to the synthesizer output

It is thus possible to transfer multiple segments in succession:

Status	Command	Comment	Enable	Sto...
	Load/Delete synthesizer signal file	Signal_01 : Index 1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Load/Delete synthesizer signal file	Signal_01 : Index 2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Load/Delete synthesizer signal file	Signal_01 : Index 3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Transfer of three segments

The segments are set up in the device's data carrier as follows:



File structure of segments on the device disk

Note

The data are saved on the device drive in a special format. It is **not** possible to copy the data manually!

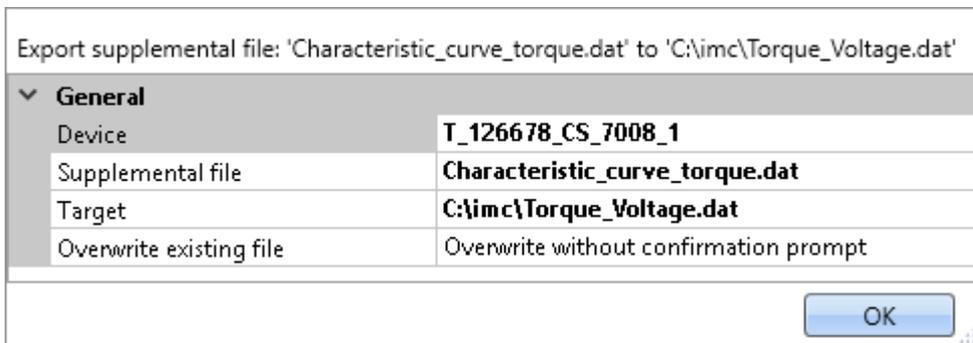
14.7.9.8 Export supplemental file

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	<p>Here, define the system's behavior for handling the files if the target file already exists.</p> <ul style="list-style-type: none"> • 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". <p>This option is relevant for automated routines.</p>

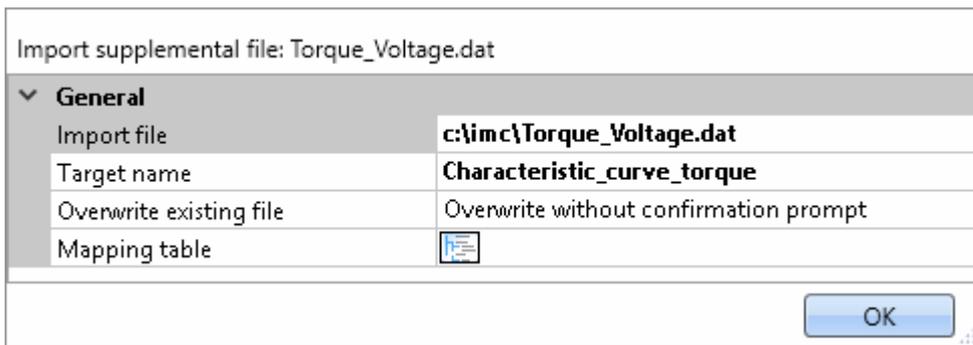
14.7.9.9 Import supplemental file

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".



Example of a characteristic curve used in imc Online FAMOS with the Charact() function

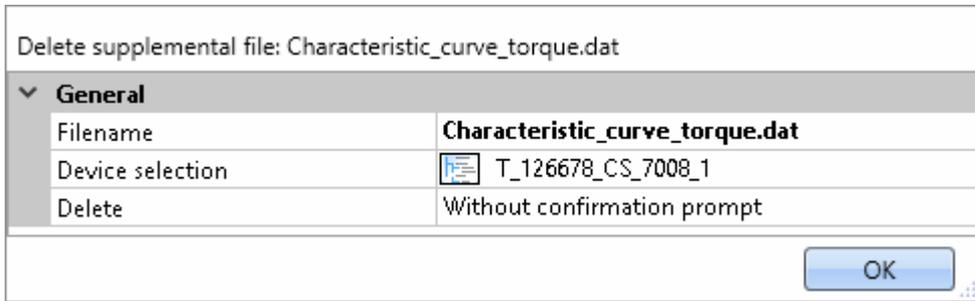
Parameter	Description
Import file	Specification of the file to import, including the pathname.
Target name	Optional entry. Here, specify a deviating name if the filename in the device is not supposed to match the original filename.
Overwrite existing file	<p>Here, define the system's behavior for handling the files if the target file already exists.</p> <ul style="list-style-type: none"> • 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". <p>This option is relevant for automated routines.</p>
Mapping table	Here, select the import target. The target may be a device . it is always only possible to make one selection.

14.7.9.10 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".



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	<p>Define here whether a confirmation prompt is to appear.</p> <ul style="list-style-type: none"> • 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". <p>This option is relevant for automated routines.</p>

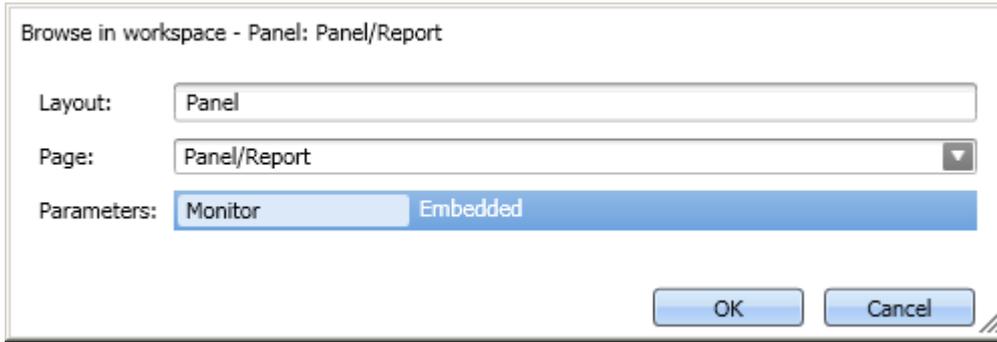
14.7.10 User interaction

14.7.10.1 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	<p>If target page: Panel</p> <ul style="list-style-type: none"> • View: Panel Embedded Mode: Jumps to the Panel. If necessary, exits fullscreen view. <p>If target page: Panel page</p> <ul style="list-style-type: none"> • 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 Monitor⁸¹⁷).

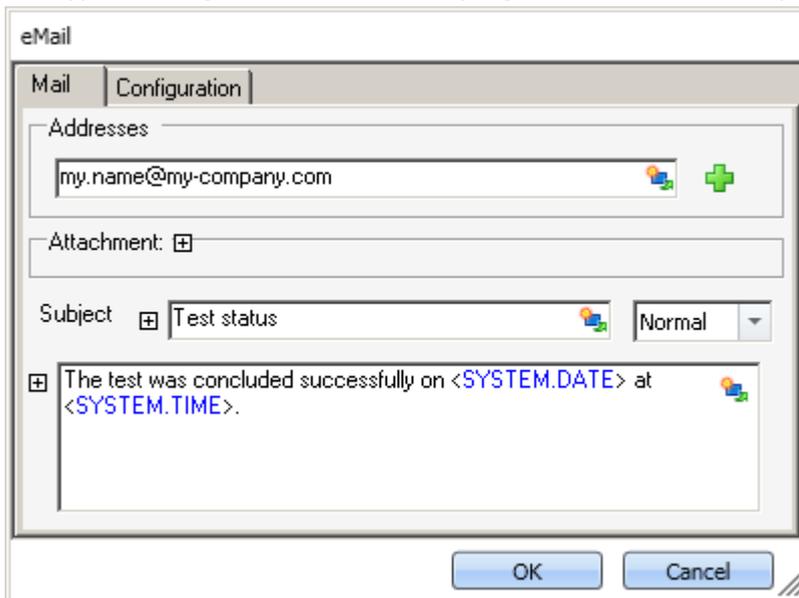
14.7.10.2 E-mail

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

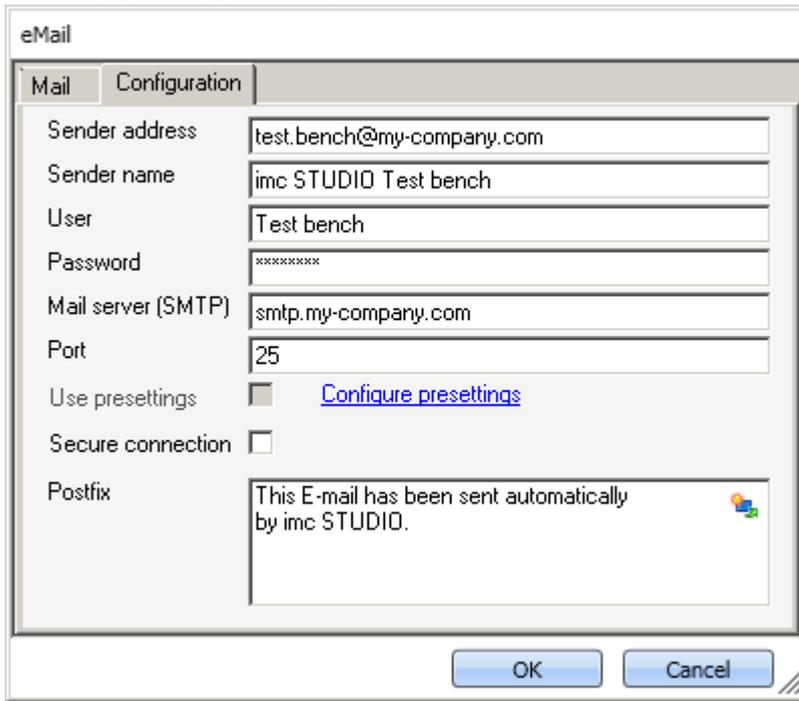
By right-clicking, or by clicking on the placeholder symbol (), you can apply a variety of placeholders.

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.

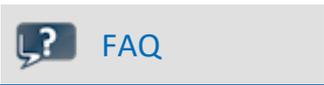
Configuration

On the page *Configuration* you set the access data for the E-Mail provider.



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 presettings	In the imc STUDIO option, it is possible to define a presetting for the E-Mail command (Ribbon <i>Extra > Options</i>). If "Use presettings" 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.



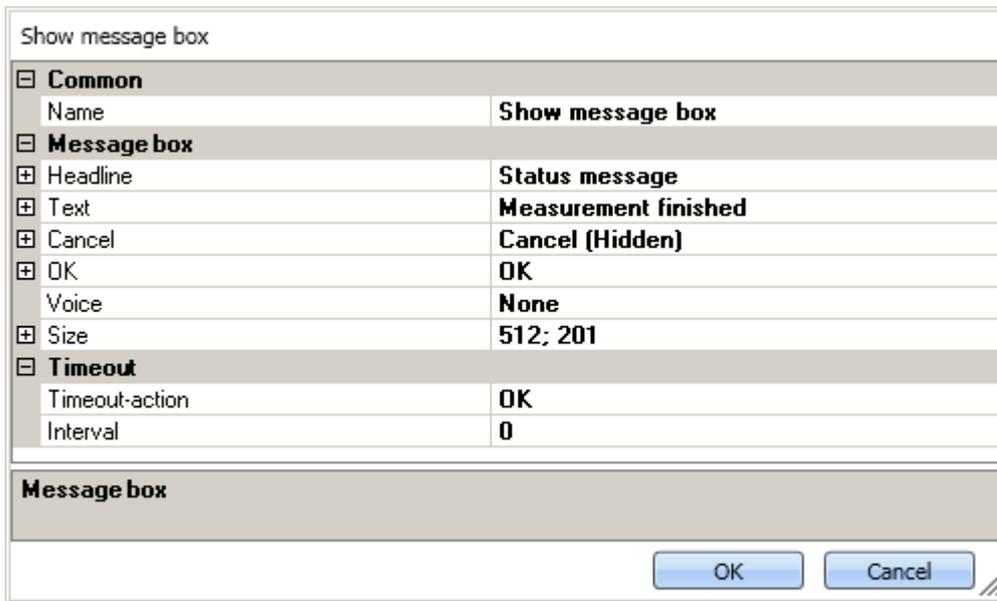
Why do e-mails not get sent, even though the settings are correct?

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.**

14.7.10.3 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.



Show message box

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. <ul style="list-style-type: none"> • 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.

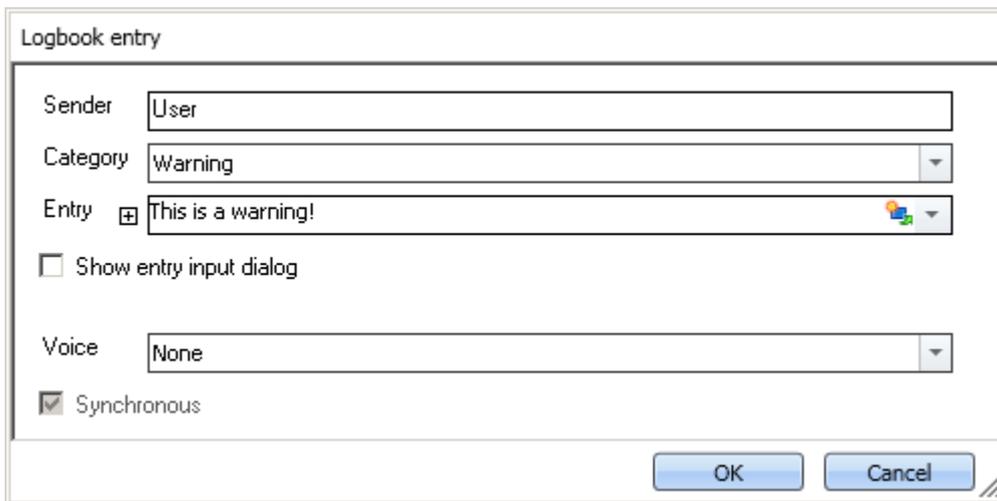
14.7.10.4 Open logbook-viewer

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](#)¹²¹⁴.

The Logbook-Viewer can be started with defined filter settings. For example, messages of the category: [Information](#)¹²¹⁵ and all [Duplicates](#)¹²¹⁵ can be hidden.

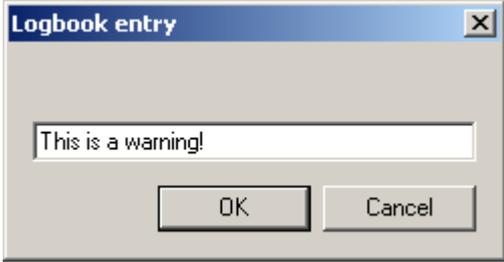
14.7.10.5 Logbook entry

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.

Parameter	Description
Category	<p>There are four different categories:</p> <ul style="list-style-type: none"> • 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	<p>Enables a text input during runtime.</p>  <p><i>With the option: "Show entry input dialog"</i></p>
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

14.7.10.6 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"

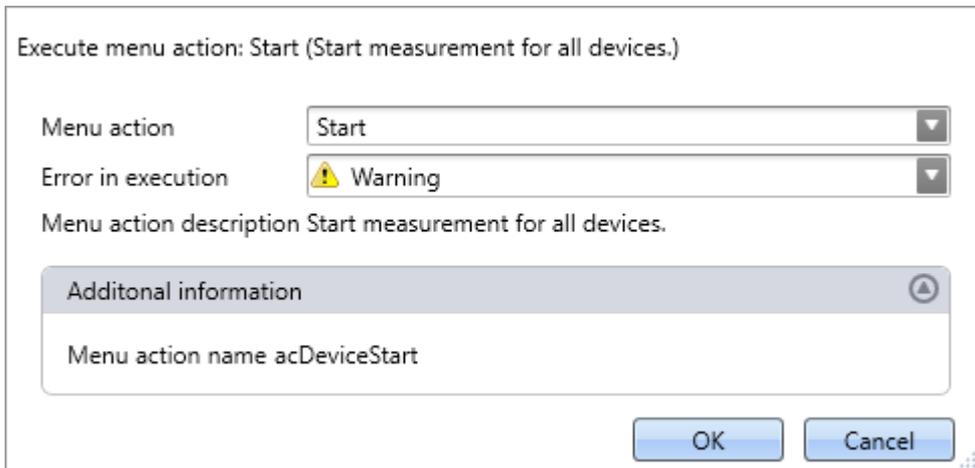
The sequence does not wait for the function behind the action to be concluded. For the sequence, the action is "done" when the action is acknowledged.

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.



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. <ul style="list-style-type: none"> • 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.

14.7.11 Variables

14.7.11.1 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)	<p>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.</p> <p>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.</p>
Parameter - Mapping instruction	Description
Mapping instruction	<p>Here you can select pre-defined export settings.</p> <ul style="list-style-type: none"> • 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	<p>Here you specify which measurement's variables to export. The option can only be used in conjunction with a mapping instruction.</p> <ul style="list-style-type: none"> • 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	<p>Here, the variables to be exported are selected and entered in the Variables list.</p> <ul style="list-style-type: none"> • 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. <p>Selected and entered variables appear in the Variables List.</p>

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 <ul style="list-style-type: none"> • 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 ^[124] . imc Format Converter provides a variety of export format templates. Storage location for the templates: "C:\ProgramData\imc\Common\Def" Note: With the help of imc FAMOS, you can create your own export format templates (*.aet) and use them in imc WAVE. 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: <ul style="list-style-type: none"> • 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: <ul style="list-style-type: none"> • Export in RAW/DAT-format, as NO Key • Export to *.aet files for the placeholder %FILECOMMENT%

14.7.11.2 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 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.

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*".

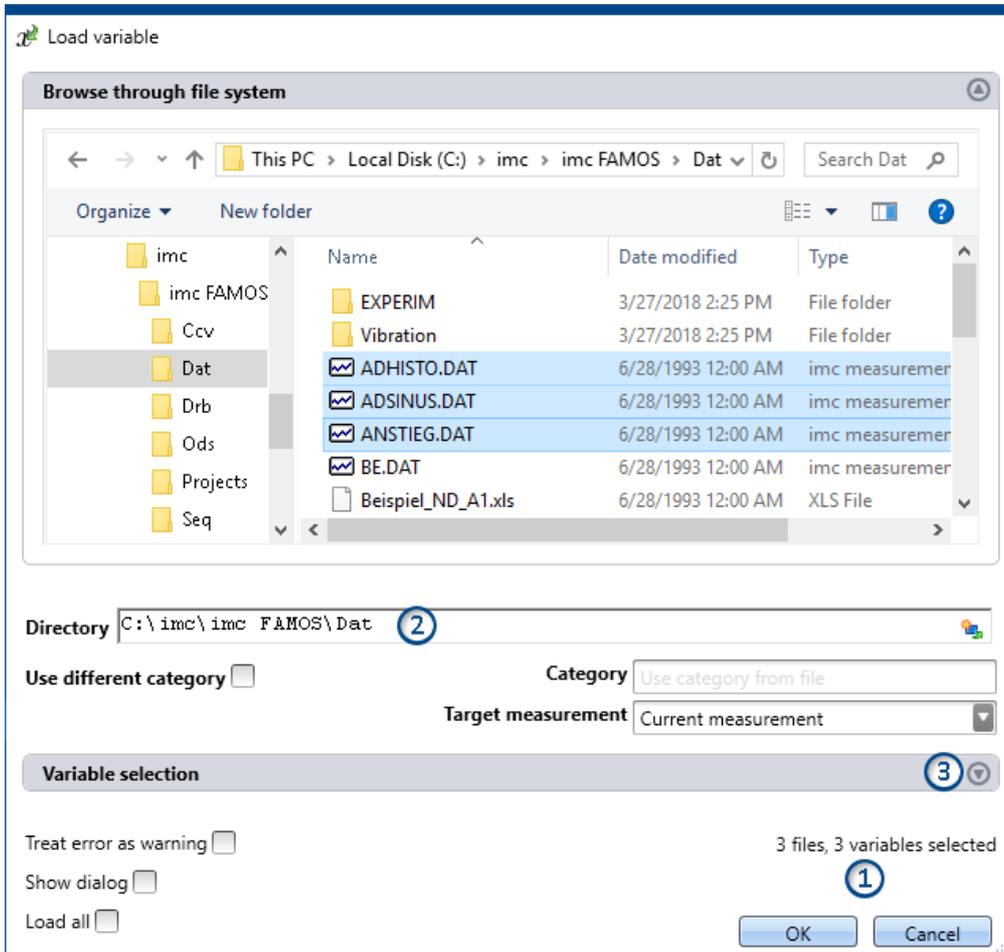
In contrast to imc FAMOS, there are variables which cannot be overwritten, e.g. channels or pv-variables. The reason is that, for instance, these variables belong to the specific device, or possess specific data formats and other properties which must be retained. If the user attempts to overwrite them, 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

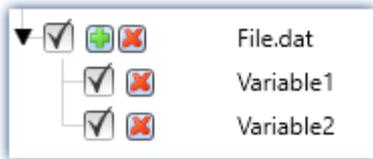
Utilizing a placeholder

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 they 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 variable](#) [749] by means of the Data Browser.

Target measurement	Description
Current measurement	<p>Load: The variable appears in the Data Browser under "<i>Current measurement</i>".</p> <p>Refill: The variable overwrites the value of any variable existing under "<i>Current measurement</i>".</p>
Last completed measurement	<p>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.</p>
Measurement number Measurement#<No>	
Fixed measurement name	
Other options	Description
Treat error as warning	<p>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.</p> <p>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.</p>
Show dialog	<p>If this option is activated, the settings dialog appears when the command is executed.</p> <p>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.</p>
Always overwrite existing variables	<p>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.</p>
Load/Import all	<p>When this option is activated, then when the command is run, all variables are imported in accordance with the file selection:</p> <ul style="list-style-type: none"> • 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*".

14.7.11.3 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

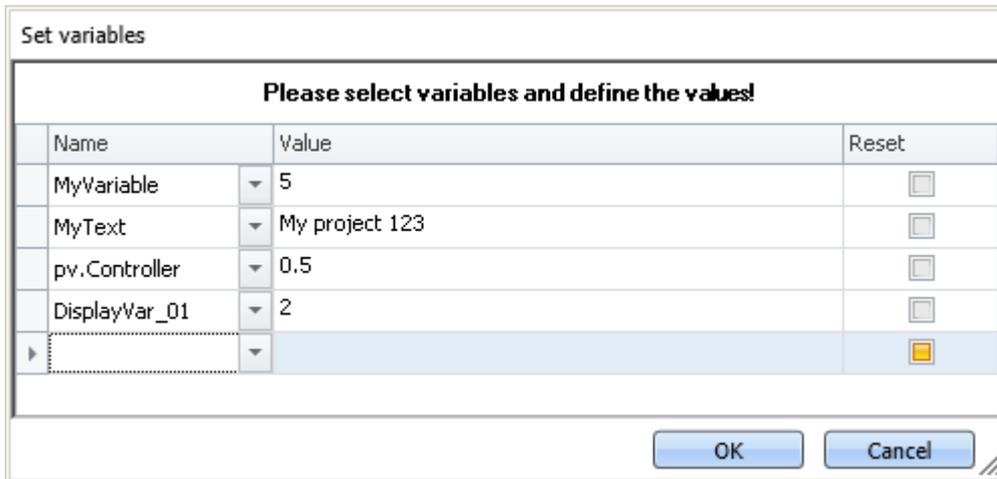
Fault condition - Unable to delete variable

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.

14.7.11.4 Set variables

The command **Set variables** assigns values to variables.



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](#) ¹²¹⁷ 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".

15 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 Assistent](#)** ^[1172] and **[Format Converter](#)** ^[1157].

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.

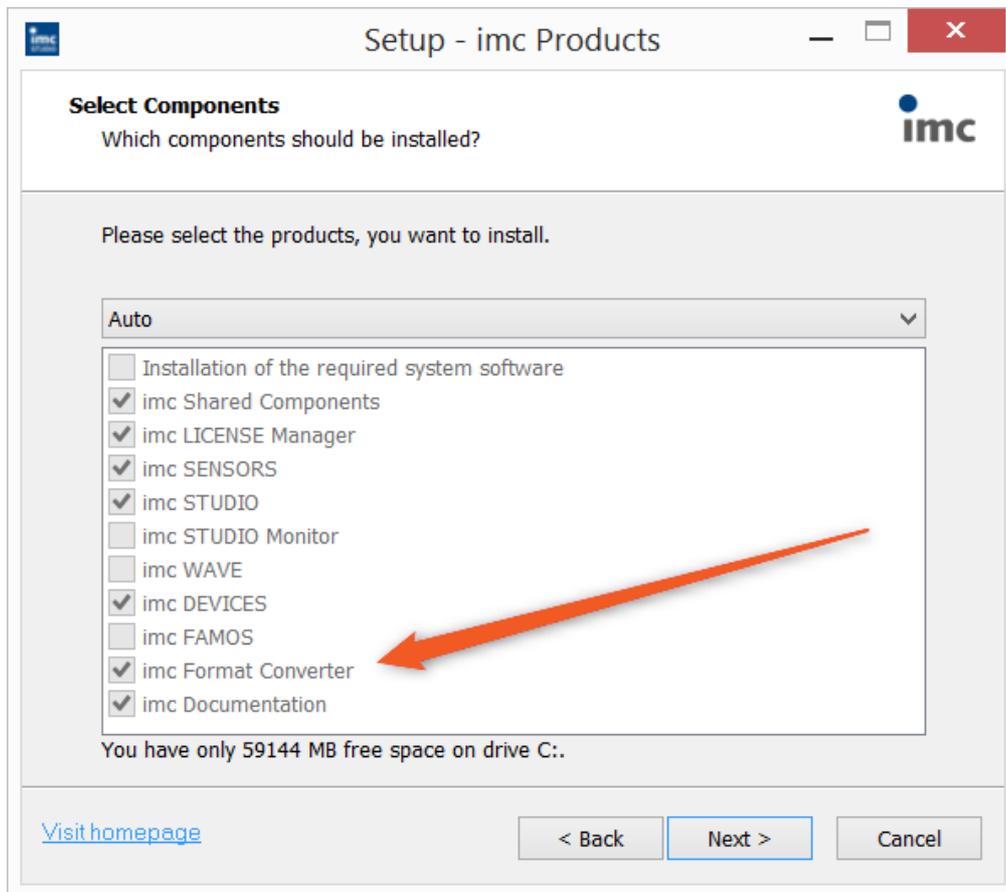
Export formats

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 TAFFmat and TEDAM, DIAdem TDM and TDMS
- Binary Universal File Format
- Universal File Format (UFF)

15.1 Installation

The installation file is located on the installation medium and is included in the installation by default.



Separate installation

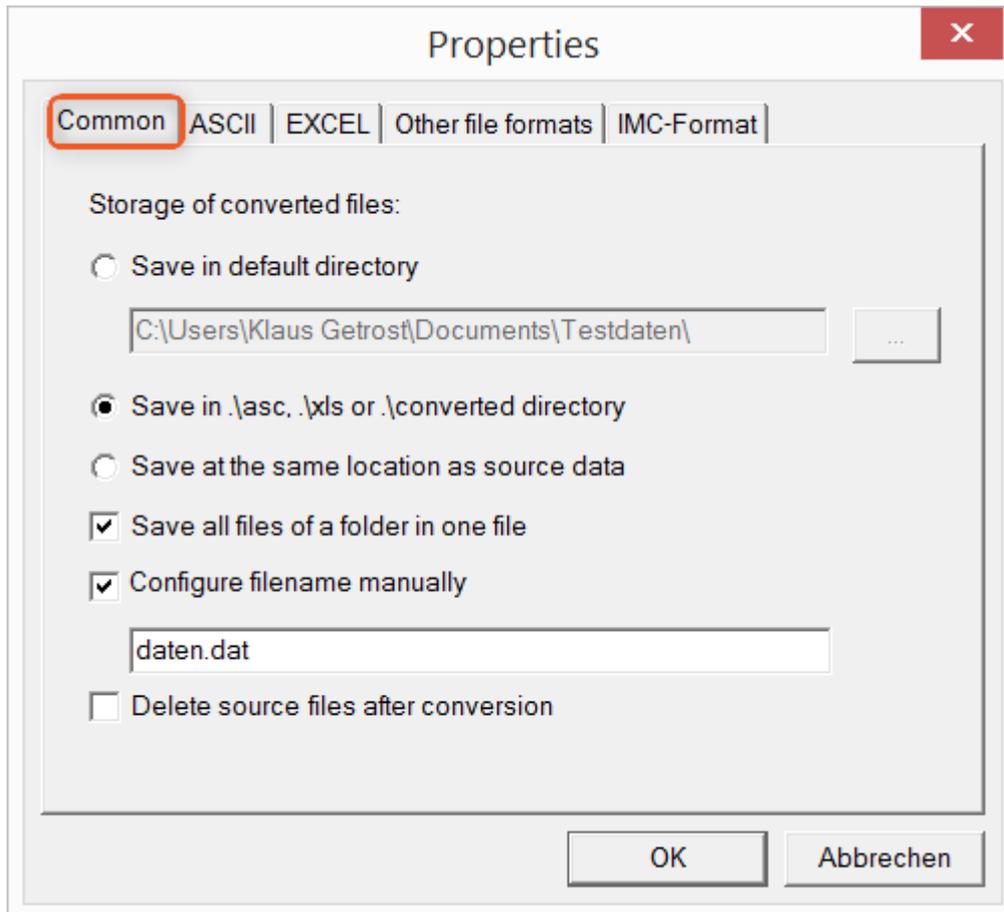
Select the installationvariant: "*User-defined*" and select there only the imc Format Converter.

15.2 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 re saved on the same **level of the tree diagram as the source data, without subfolders**.

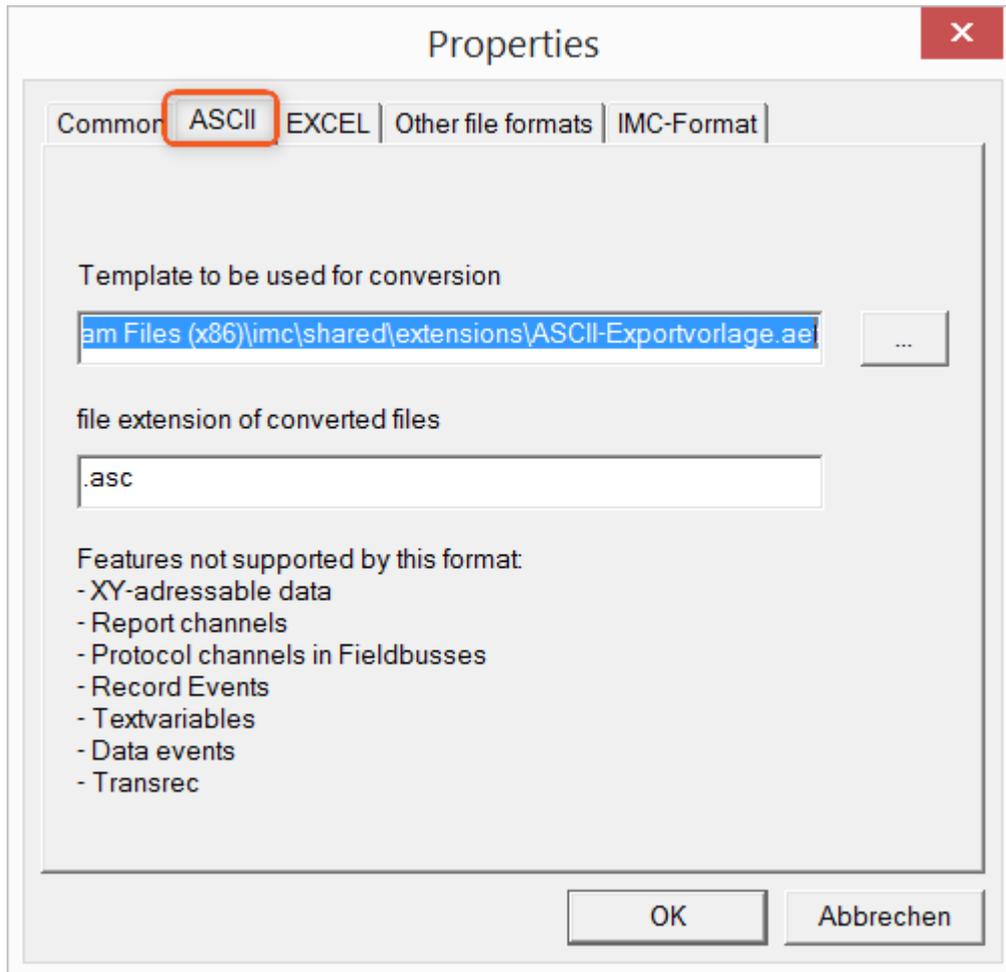
Manner of data saving:

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.

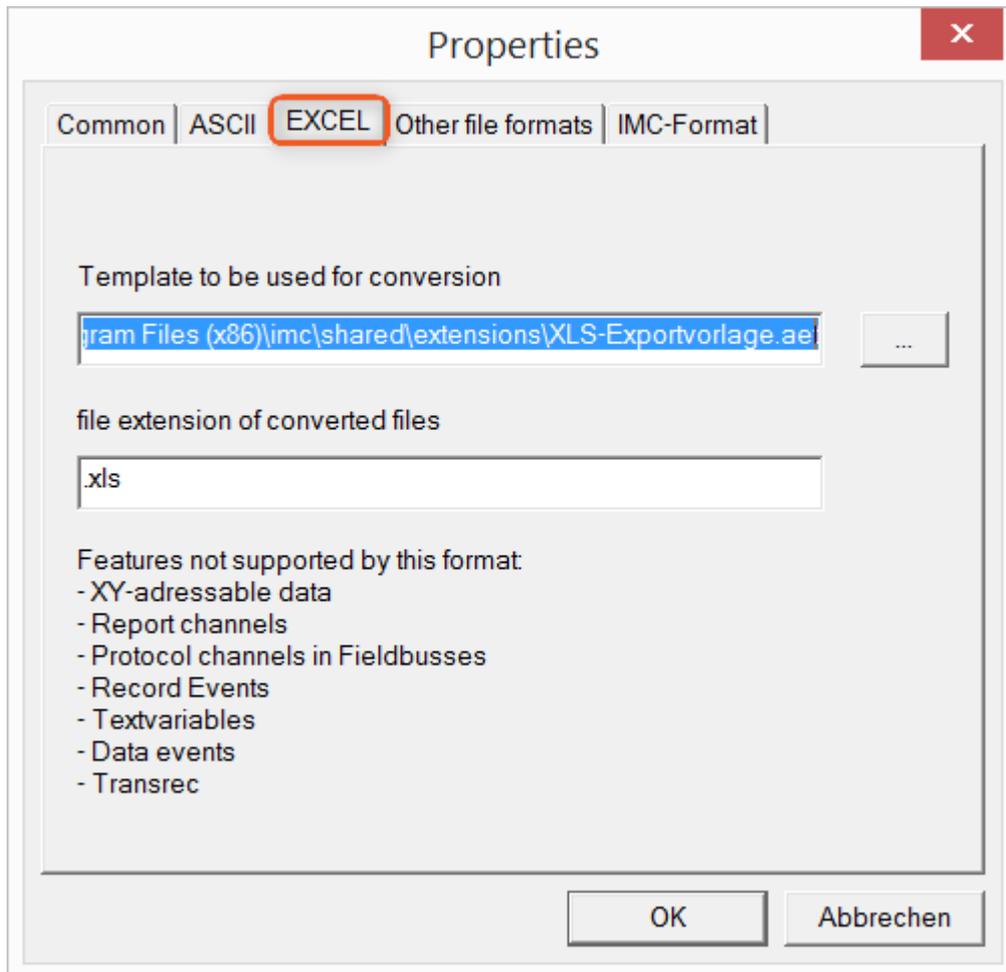
Special variable types which can **not** be exported as ASCII data are listed on this tab page.

! 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](#)⁷⁸⁹. By default, the file extension is "XLS", but can be arbitrarily specified here.

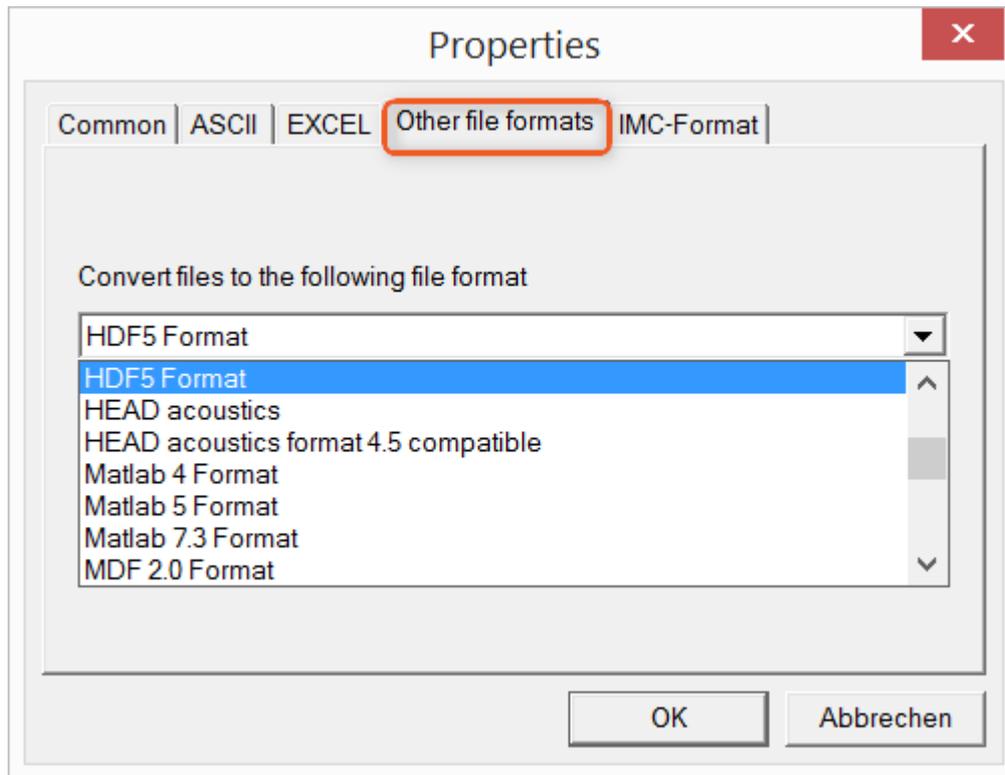
Variable types which are **not** exported correspond to those for [ASCII-export](#)⁷⁸⁹.

! 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



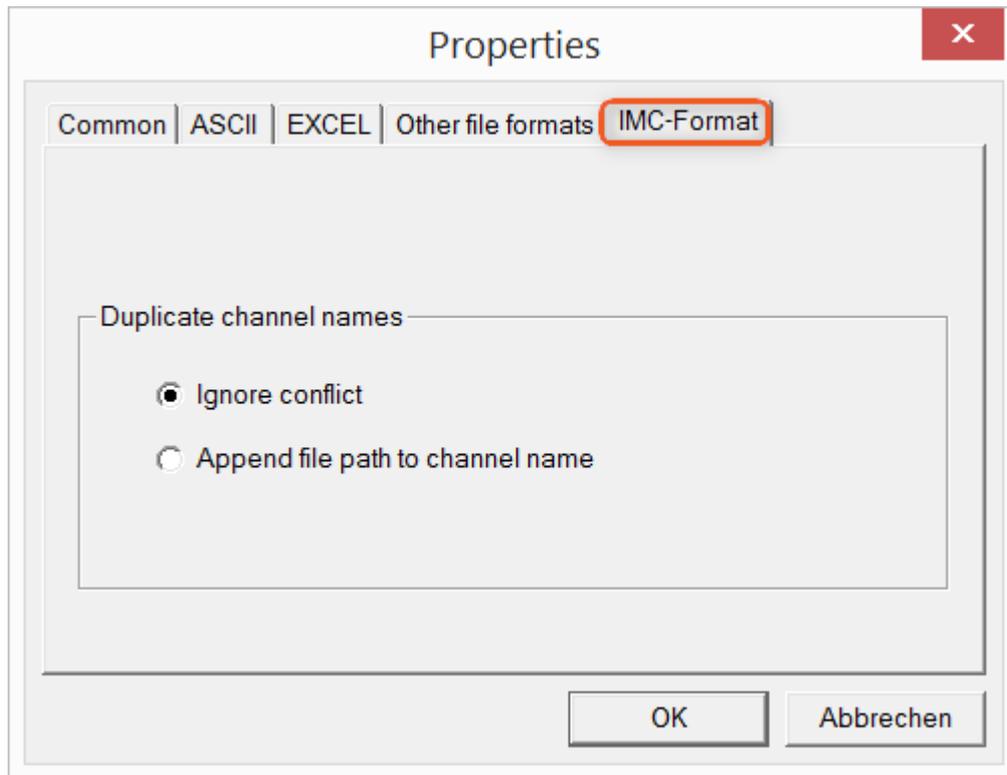
Convert files to the following file format:

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](#)⁷⁸⁸" to accomplish this would be:

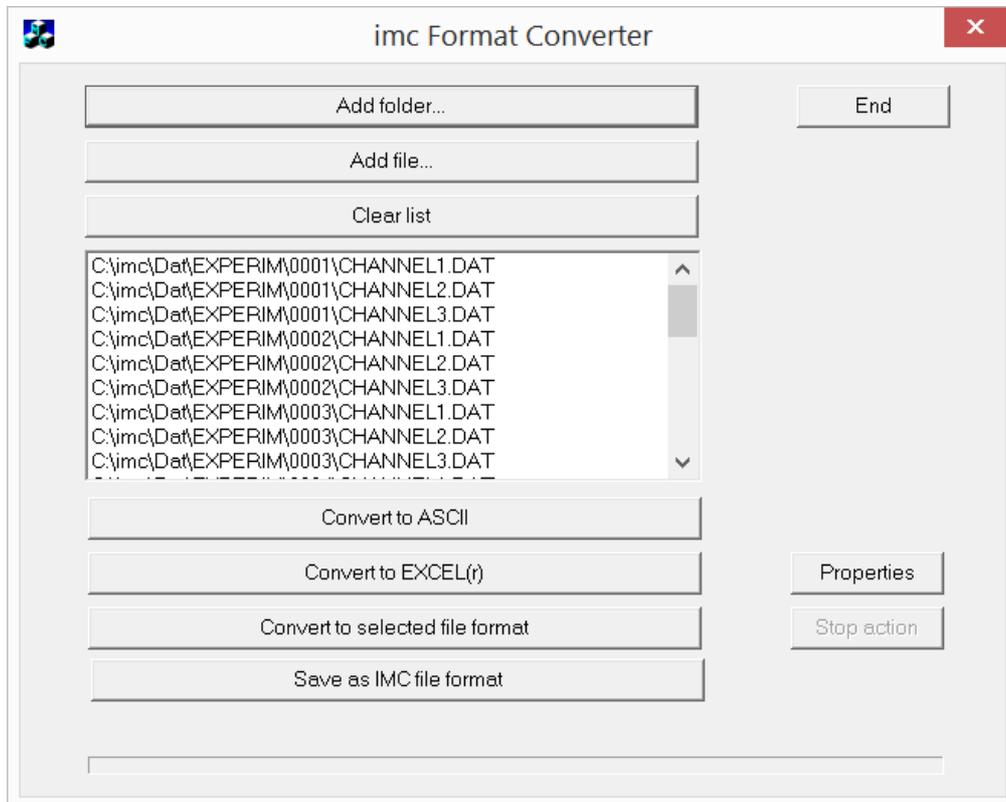
Save at same location as source data = active

Save all files of a folder in one file = active

Delete source files after conversion = active

15.3 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...

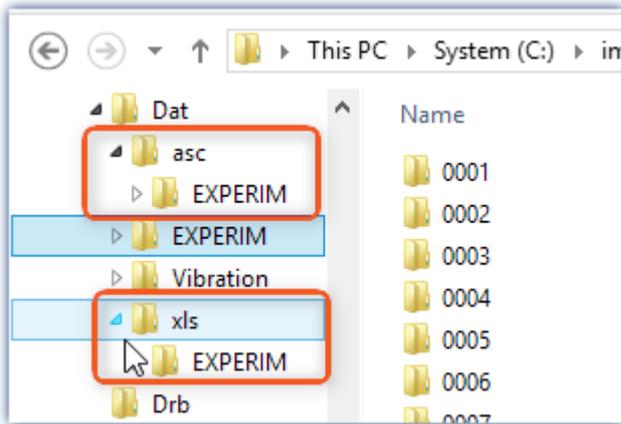
Selection of individual files.

Clear list

All entries are deleted from the Files list.

Convert to ASCII/EXCEL(r)

The standard variant for converting data to [ASCII](#)⁷⁸⁹ or [EXCEL](#)⁷⁹⁰-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](#)⁷⁹¹". 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](#)⁷⁸⁸.

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 page](#)⁷⁹².

15.4 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](#)⁷⁹⁵.

Command line options

	Description	Example
-a	conversion to "ASCII"	<code>imcFrmtCvrt.exe -a <files or paths></code>
-e	conversion to "EXCEL"	<code>imcFrmtCvrt.exe -e <files or paths></code>
-k	conversion to "Other data formats"; The format selected in the settings is used, e.g. MDF	<code>imcFrmtCvrt.exe -k <files or paths></code>
-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.	<code>imcfrmtcvrt -i -s d:\setup.xml</code> <code>imcfrmtcvrt -i d:\setup.xml <files or paths></code>
-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	<code>imcfrmtcvrt -i -h d:\setup.xml <files or paths></code>
-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	<code>imcfrmtcvrt -i -h -l d:\setup.xml d:\Logfile.log <files or paths></code>

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 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
```

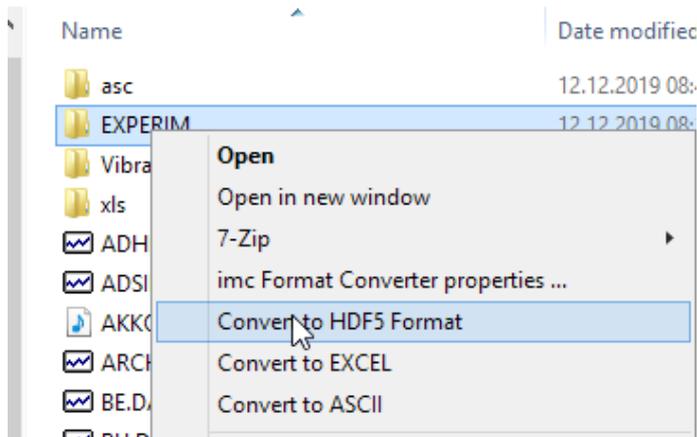
All files specified in the file: "*d:\daten.xml*" are converted.

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*".

15.5 Conversion via the Windows Explorer

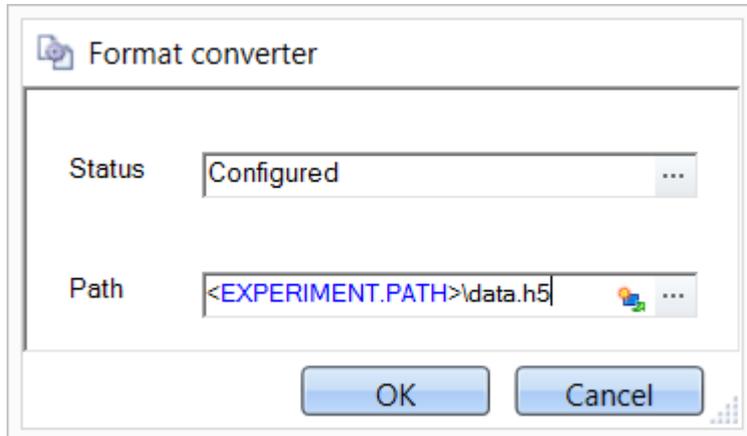
Installation of the imc Format Converter expands the Windows Explorer's context menu. The current templates for export of [ASCII](#)⁷⁸⁹ and [EXCEL](#)⁷⁹⁰, as well as the export format last selected on the page "[Other file formats](#)⁷⁹¹" can then be applied directly to a folder or files:



15.6 Format Converter as imc STUDIO command

Calling

Via the Sequencer: "External calls" > "Format converter":



Configuration of the command: Format Converter

Status

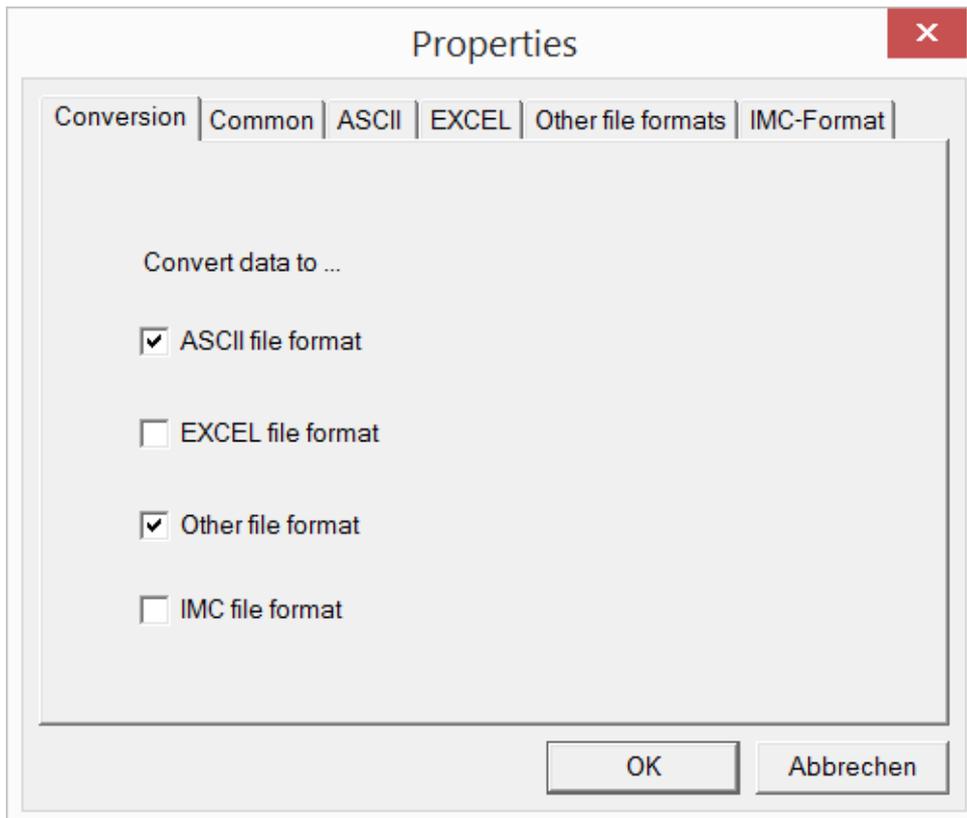
Immediately upon opening the command's configuration, the status is set to "not configured".

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](#) ⁷⁸⁸ 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](#)" ⁷⁸⁸ (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](#)⁷⁸⁸: Storage location and whether individual files or a multi-file is to be created.
- [ASCII](#)⁷⁸⁹: Selection of the export template and specifying of the file extension.
- [EXCEL](#)⁷⁹⁰: Selection of the export template and specifying of the file extension xls or.xlsx.
- [Other file format](#)⁷⁹¹: Selection of the format and if necessary its formatting options.
- [IMC-file format](#)⁷⁹²: Save the data to a file in the imc format, or as a copy to a different memory location.



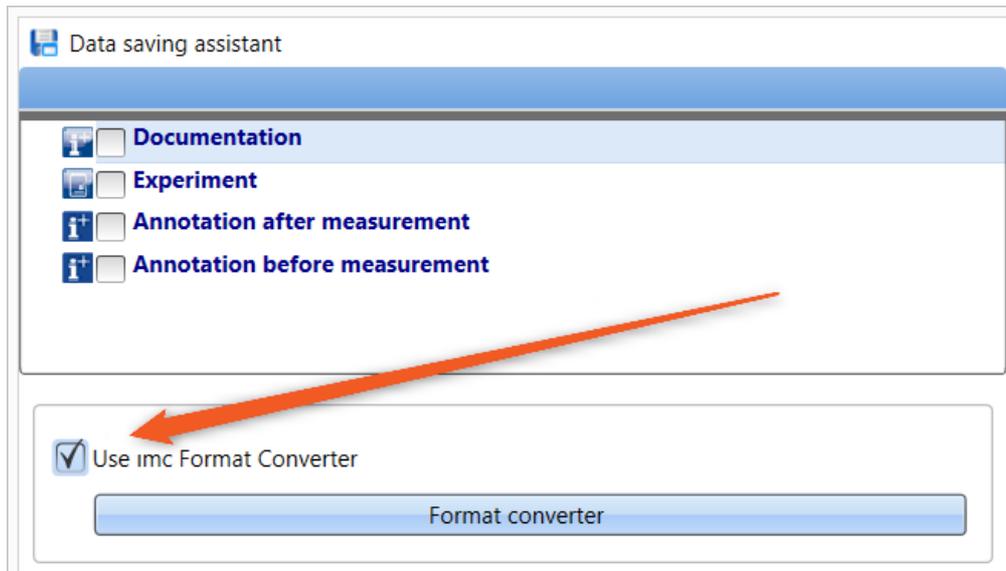
Setting the Format Converter

15.7 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](#)⁷⁹⁷.



Format Converter as component of the Data Saving Assistant

16 Shared Components

Documentation for components used by multiple imc STUDIO plug-ins.

Component	Description
Logbook ¹²¹⁴	In the Logbook, messages 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.
Placeholder ¹²¹⁷	Description of all placeholders.
Options ¹²³⁷	General options

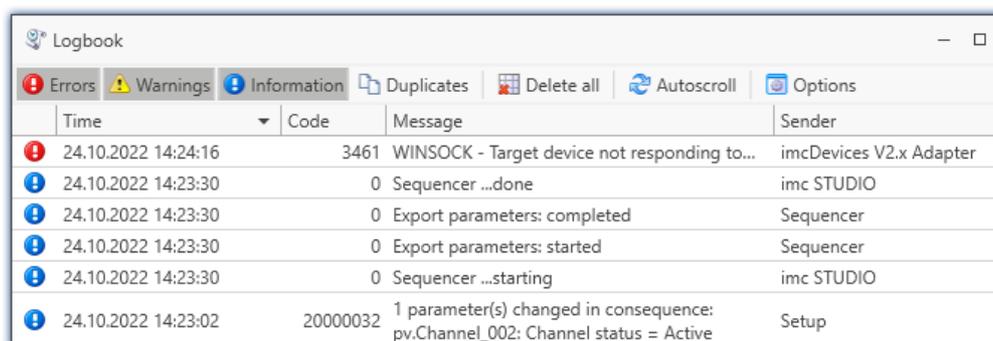
Note

These documents can contain references to products not included in your installation. The links to them will not work in the help.

16.1 Logbook

In the tool window "Logbook", **messages of the categories: "Fatal" (🔴), "Error" (🔴), "Warning" (⚠️) and "Information" (🔵)** 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.

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.



Time	Code	Message	Sender
24.10.2022 14:24:16	3461	WINSOCK - Target device not responding to...	imcDevices V2.x Adapter
24.10.2022 14:23:30	0	Sequencer ...done	imc STUDIO
24.10.2022 14:23:30	0	Export parameters: completed	Sequencer
24.10.2022 14:23:30	0	Export parameters: started	Sequencer
24.10.2022 14:23:30	0	Sequencer ...starting	imc STUDIO
24.10.2022 14:23:02	20000032	1 parameter(s) changed in consequence: pv.Channel_002: Channel status = Active	Setup

Example of logbook entries

Each logbook entry consists of:

Parameter	Description
Symbol for the category	<i>Fatal</i> (🔴), <i>Error</i> (🔴), <i>Warning</i> (⚠️) and <i>Information</i> (🔵)
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 WAVE 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
	<p>Filtering the Logbook according to categories. To do this, click on the Filter-button: "Errors", "Warnings" or "Information"</p>  <p style="text-align: center;"><i>Logbook filter</i></p> <p>Usually, all categories are selected.</p>
Filtering the logbook	<p>Filtering also takes effect on new messages</p> <p> Note that the filter has an effect when a new message appears. For this reason, this message may not be displayed.</p> <hr/> <p>Default setting for the filters</p> <p> You can pre-configure the filter settings using the option ¹²³⁹ ("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.</p>
Show search panel	<p>Search the logbook for entries and text passages</p> <p>Located in the context menu of the title bar.</p> <p>An input appears. Enter the desired search keyword here. All messages are displayed which contain the text entered.</p> <hr/> <p>Search also takes effect on new messages</p> <p> 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.</p>
 Duplicates	<p>Hiding duplicates</p> <p>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.</p> <p>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.</p>
 Copy	<p>Copying the selected message to the Clipboard</p> <p>Context menu "Copy": All information of the entry that can be seen in the logbook is now in the Clipboard.</p>
 Copy with details	<p>Copying the selected message with internal information to the Clipboard</p> <p>For error messages, the stack trace is added to the copied text. This helps our tech support to analyze errors reported to us.</p> <p>Context menu "Copy with details": All of the entry's information is then copied to the Clipboard.</p>

Buttons / Context menu	Description
 Delete all	<p>Deleting the displayed messages</p> <hr/> <p>"Delete all" clears the display</p> <p> 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 <i>"Logbook-Viewer"</i>.</p>
 Autoscroll	<p>Activating/deactivating automatic scrolling</p> <p>When a new message is entered in the logbook, the display in the <i>"Autoscroll"</i> mode automatically jumps to the new entry. If you wish to investigate older messages, you can deactivate this mode.</p> <p>This happens automatically when a message is selected, or click on the <i>"Autoscroll"</i>-button.</p> <p>Re-activate the mode by:</p> <ul style="list-style-type: none"> • pressing <Esc> • left-clicking the mouse over the selected entry while holding down the <Ctrl> key • clicking on the <i>"Autoscroll"</i>-button in the menu <hr/> <p> New messages are not automatically displayed visibly</p> <p>Be aware that when the <i>"Autoscroll"</i> mode is deactivated, you will not be automatically notified when new messages appear.</p>
 Logbook-Viewer	<p>Starting the Logbook-Viewer and opening old logbooks</p> <p>By means of the <i>"Logbook-Viewer"</i>, you can look at current, deleted and saved logbook entries. In addition to the familiar logbook-functions, it is possible to load saved logbooks.</p>
 Options	<p>Options for the Logbook</p> <p>From within the logbook, you can open the Options for the Logbook¹²³⁹.</p>
 Export	<p>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.</p>

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.

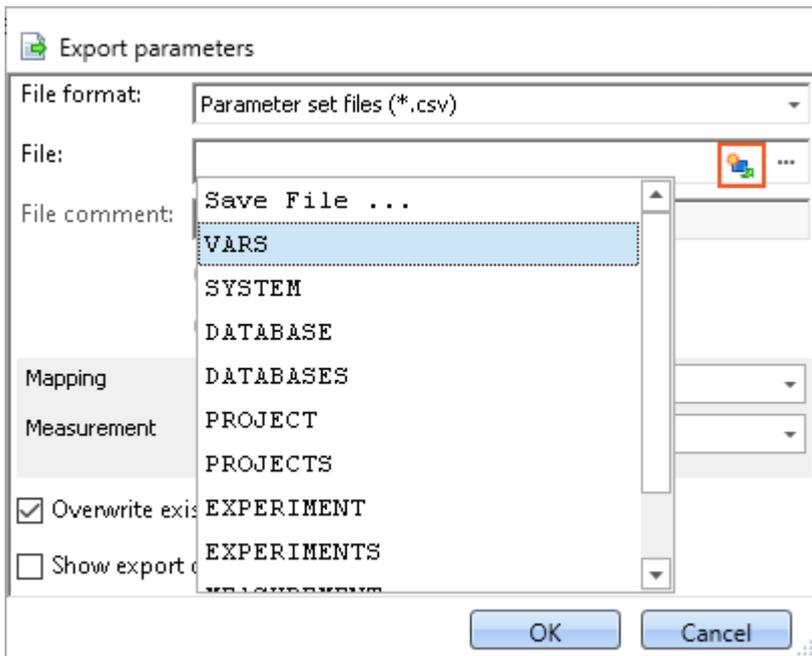
16.2 Placeholders

In imc WAVE , 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](#)^[1148] as well as with [Widgets](#)^[820]. A list of all known placeholders is presented in the section "[list of placeholders](#)"^[1218].

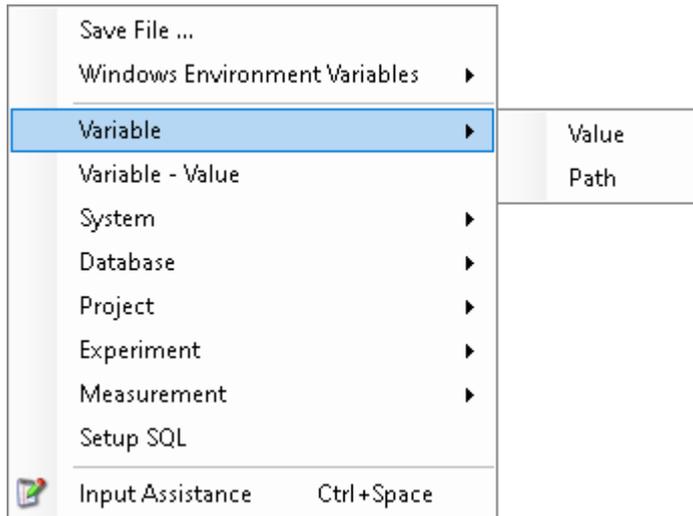
Placeholders are offered where the placeholder symbol () is displayed. Not all placeholders can be used at all positions. To obtain a list of available placeholders, click the left mouse button over the placeholder symbol ()



List of available placeholders for the comand "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](#)^[1218] of known placeholders
- [Formatting](#)^[1233] of the placeholders
- [Environment variables of the OS](#)^[1236] supported by imc WAVE

16.2.1 List of placeholders

The following placeholders are described in this section:

- [CONTROL](#)^[1219]
- [DATABASE/DATABASES](#)^[1220]
- [EXPERIMENT/EXPERIMENTS](#)^[1221]
- [MEASUREMENT](#)^[1224]
- [MEASUREMENTS](#)^[1225]
- [PAGE](#)^[1227]
- [PANEL](#)^[1227]
- [PROJECT/PROJECTS](#)^[1227]
- [SELCONTROL](#)^[1229]
- [SETUP](#)^[1230]
- [SYSTEM](#)^[1231]
- [VAR/VARS](#)^[1232]

16.2.1.1 CONTROL

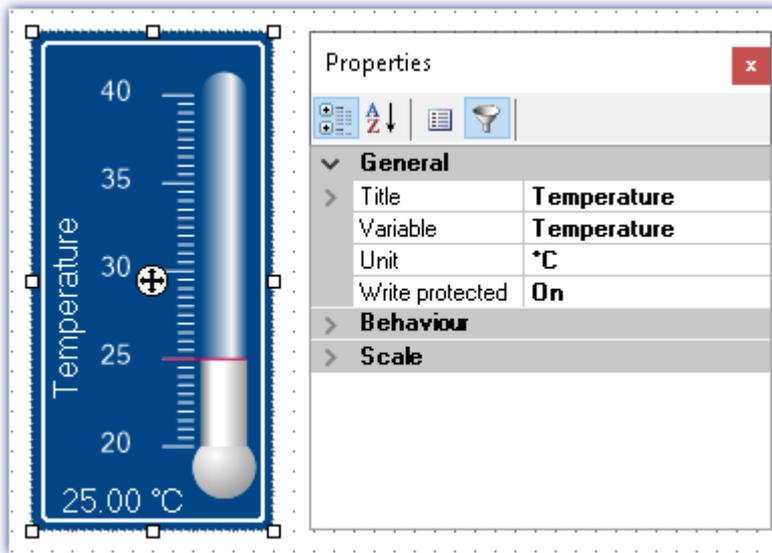
These placeholders are only available for Widgets.

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

16.2.1.2 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

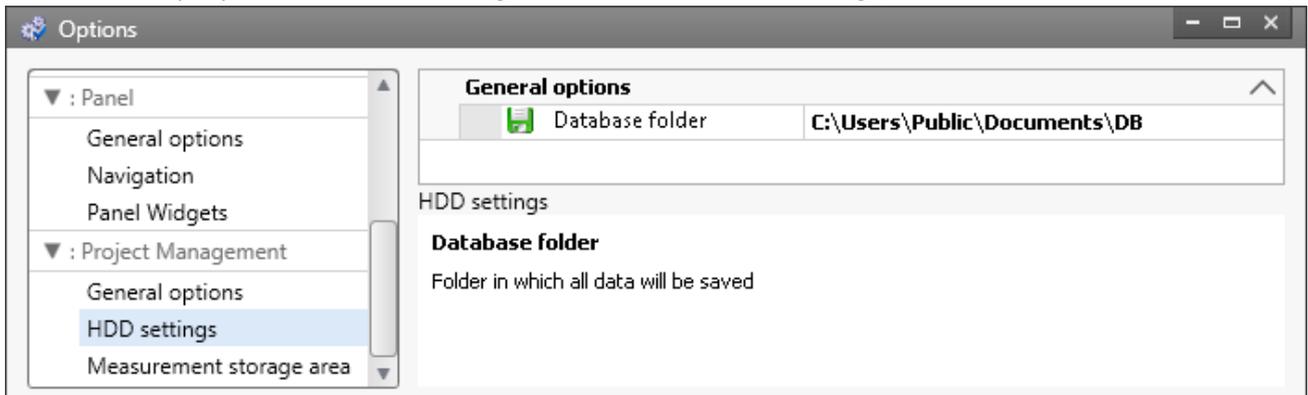
The placeholder *DATABASE* always pertains to the current project database. If you wish to use the placeholder with another database, the placeholder *DATABASES* must be used, e.g.:

<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:



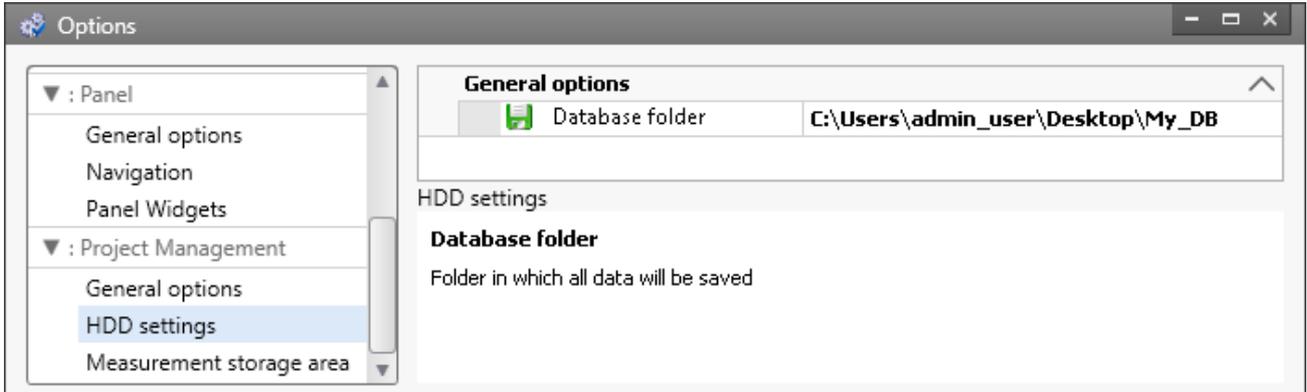
Default database path under Windows 10

- <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

16.2.1.3 EXPERIMENT/EXPERIMENTS

Name	Context menu	Description
CFG	Experiment - Configuration path	Path to the configuration folder (" <i>config</i> " 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

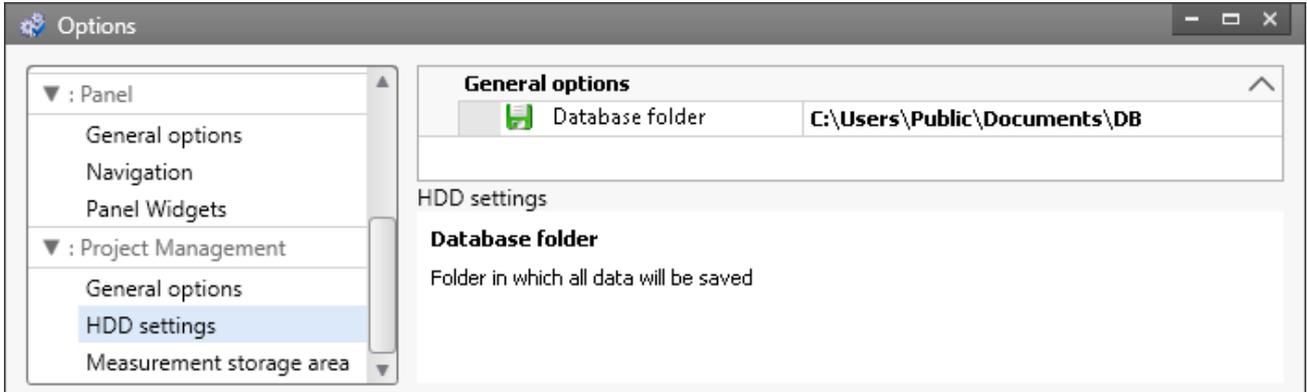
The placeholder `EXPERIMENT` always pertains to the current experiment. If you wish to use the placeholder with another experiment, the placeholder `EXPERIMENTS` must be used, e.g.:

`<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>

16.2.1.4 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.

MEASUREMENT.SQL-Assistant

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

Parameter	Description
Columns	<p>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:</p> <ul style="list-style-type: none"> • 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. <p>If any metadata with the measurement were saved, they are listed here.</p>
Condition	<p>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:</p> <ul style="list-style-type: none"> • 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". <p>If the condition is left empty, the properties/metadata of all measurements are displayed in succession (separated by ',').</p>



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).

16.2.1.5 MEASUREMENTS

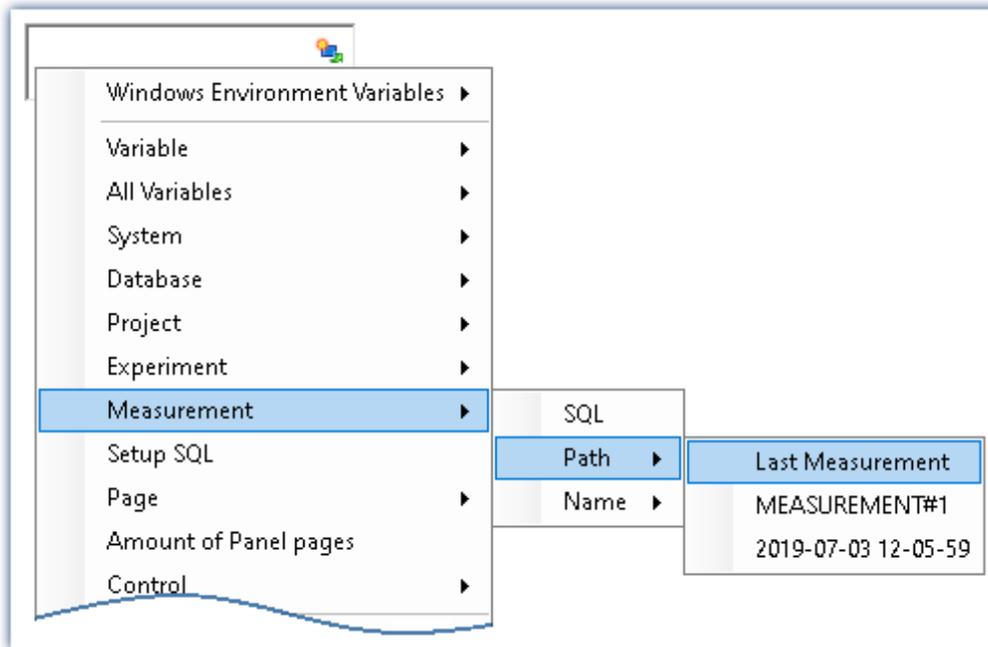
With this placeholder, you can get the path or name of a measurement; e.g. the folder path to the last saved measurement.

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

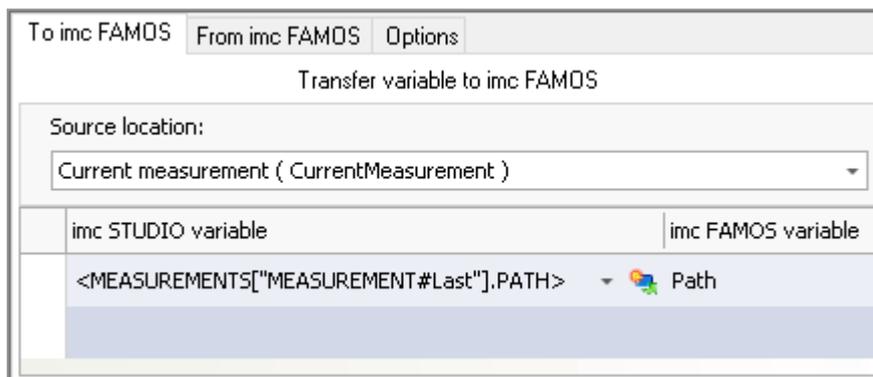
The following MEASUREMENTS-placeholders are available:

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](#)"

16.2.1.6 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`.

16.2.1.7 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`.

16.2.1.8 PROJECT/PROJECTS

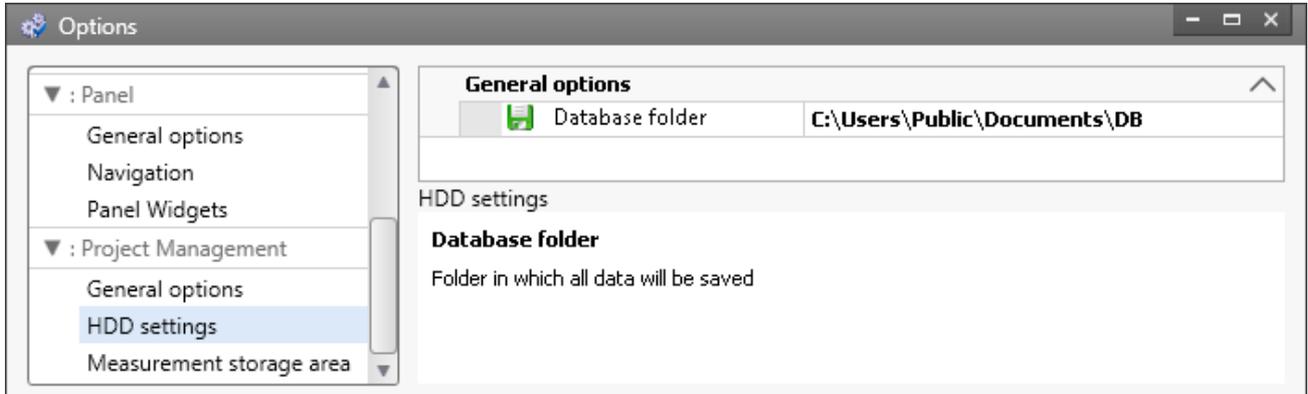
Name	Context menu	Description
CFG	Project - Configuration path	Path to the project's configuration folder (<i>config</i> folder)
NAME	Project - Name	Name of the project
PATH	Project - Path	Path to the project's saving folder
SETTINGS	Project - Configuration file	Settings for the project (*.imcAppSettings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

16.2.1.9 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

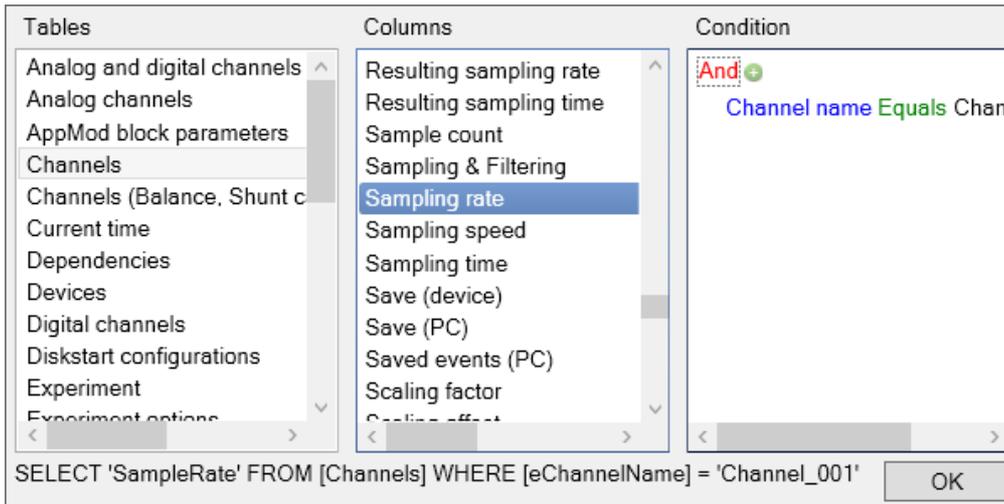
In which case, `<SELCONTROL.VARS>` = Channel_001,Channel_002,Channel_003.

16.2.1.10 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. Therefore, 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:



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.

16.2.1.11 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](#)¹²³⁴ and the [time](#)¹²³⁴ are provided in the section [Formatting](#)¹²³³.

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](#)¹²³³.

16.2.1.12 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](#)".

16.2.2 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](#)¹²³³
- [SYSTEM.DATE](#)¹²³⁴
- [SYSTEM.TIME](#)¹²³⁴
- [VALUE](#)¹²³⁵

16.2.2.1 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.

16.2.2.2 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

16.2.2.3 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.

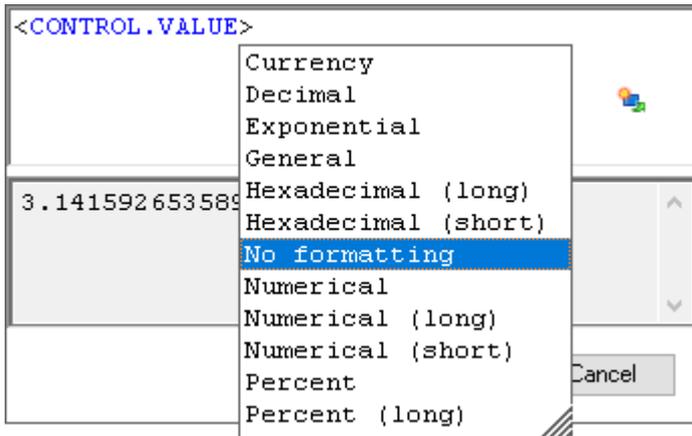
Example

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

16.2.2.4 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`

16.2.3 Environment variables of the OS

In imc WAVE 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*.

16.3 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 WAVE installation). A diskette-symbol indicates the storage location:

Scope	Description
 Application option	Options which apply for the imc STUDIO / imc WAVE 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	<p>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.</p> <p>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".</p>
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	<p>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").</p> <hr/> <p> See also "Setup pages - Configuring device": "Storage options and directory structure" > "Controlling Data Storage" > "Customized storage location for measurement data"^[330].</p>
Custom folder naming for measurement data storage	<p>This specifies the structure below the base path. For instance, folders are set up here according to the settings for the directory structure^[328].</p> <hr/> <p> See also "Setup pages - Configuring device": "Storage options and directory structure" > "Controlling Data Storage" > "Customized storage location for measurement data"^[330].</p>

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.

Project Management

HDD Settings

Options - General	Description
Database folder	Here you set where the " Database " is to be saved.
	 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.

Setup

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.

Device options

Options - General	Description
Selection of the firmware version	<p>If multiple imc DEVICES versions are installed on the PC, a specific version must be selected for the operation of each device. This option controls the selection strategy.</p> <p>If only one imc DEVICES version is installed, this setting has no effect.</p> <ul style="list-style-type: none"> • Manual: With "Manual", the system always asks which firmware version to use when device are selected or an experiment is loaded. • Automatic: "Automatic" avoids firmware updates. The version currently running on the device is used preferentially. • Always use latest: With this selection, the most current firmware version compatible with this imc WAVE version is always used.
Show warning for critical parameter changes	<p>Show a warning when a critical parameter value changes, e.g. when changing the supply provided to all of a module's channels.</p> <p>Disabling this option can be particularly useful when using automated procedures such as Sequencer or Scripting, since the warning otherwise pauses the procedure until it has been confirmed.</p>
Time limit for calling device control software [s]	<p>Timeout in seconds for a call to the device control software. If this time elapses, the device control software automatically restarts. This can cause a parts of the configuration to be lost.</p> <p>If the devices are operated over a very slow connection, this value can be increased to ensure correct functioning.</p> <p>The minimum and the default value is 60 seconds.</p>
Options - Virtual device clock	Description
<p>Virtual device clock:</p> <p>The virtual clock synchronizes during running measurement with the Master-device's clock. This is necessary when functions are being used on the PC which require an exact time track (e.g. Video).</p> <p>In networks under high load, good synchronization (between the PC's virtual device clock and the device) is not always possible. When the demands on the network are too high, imc WAVE posts a corresponding error message when an attempt is made to start measurement.</p> <p>It may be helpful to increase the setting for the Minimum accuracy.</p>	
<p> See also "Setup pages - Configuring device": "<i>Synchronization</i>" > "<i>Clock types</i>" > "VRTC - Virtual clock on the PC".</p>	
Minimum accuracy [ms]	<p>The option defines the maximum time in milliseconds that the PC's virtual clock may deviate from the device clock. This concerns mainly non imc devices, e.g. video. In case of video a value of 10 ms provides synchronisation of measurement data to video images at 100 frames per second. Too low a value prevents measurement start.</p>

Variables

Datapool

Settings for memory use by the imc WAVE Datapool.

Options - Speicher Optionen (PC)	Description
Split up pretrigger data into intervals	<p>If the pretrigger's measured data are located in time across interval boundaries, it is possible to treat the measured data as follows:</p> <ul style="list-style-type: none"> • 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. <hr/> <p> See also: "Setup pages - Configuring device" > "Trigger and Events" > ... > "Pretrigger"^[301]</p>

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](#)"^[334].

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](#)"^[1192] and
- the "[Export current data](#)"^[334] 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.

17 Miscellaneous

17.1 Last Changes

17.1.1 in Doc. Rev. 6.5

Setting Up - Software

Section	Amendments
Supported Measurement Devices	The structure analyzer can now also be used with imc ARGUSfit and imc EOS devices.

Setup - Device (general)

Section	Amendments
Device overview ^[146]	Device overview supplemented with imc Online FAMOS and the pre-processing.
Saving metadata with the channel	<ul style="list-style-type: none"> 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.
Comments on the measurement - Measurement metadata	Using the Comment-function, it is possible to record extra information on a saved measurement.

Setup pages - Configuring device

Section	Amendments
For devices of the firmware group B (imc DEVICEcore) ^[337]	<ul style="list-style-type: none"> Notes on Hot-Plug capability ^[337] have been removed. Information on the file format ^[337] of the internal hard drive of the imc EOS-devices. Notes on protecting the memory card by means of routine formatting ^[338]. Corrected specification of the file system: "FAT32" -> "Large FAT32"
Synchronization - Potential difference with synchronized devices ^[232]	Notes revised: Devices belonging to firmware group B no longer have a yellow ring.
Synchronization varieties ^[199]	The chapter has been restructured so that all information on a synchronization type is located in the same place.
Trigger and events Hysteresis ^[293]	<p>Notes supplemented: "<i>Influence of the hysteresis on the measurement's start</i>"</p> <p>When an event occurs before the start of the first measurement, it is ignored if it is no longer valid by the time of the beginning of the measurement. This change only applies to the time preceding the first measurement. For subsequent measurements, the previous behavior persists.</p>
Balance - tare and bridge	If any changes were made to the device configuration during measurement, taring is no longer possible during the measurement (devices belonging to firmware group B).

Section	Amendments
Breakdown of the start/stop and triggering behavior	imc ARGUSfit adopted in the tables.

Data analysis and signal processing

imc Online FAMOS

Section	Amendments
Function reference: Miscellaneous	Optional parameter described: Target sampling time instead of a reduction factor. Red, Max, Min, Mean, RMS, StDev, NumberOfPulses, HighLowRatio, Sum, Sum2, ExpoRMS, PulseDuration, PulseFrequency, PulsePhase

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.

17.1.2 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	

Setting Up - Connect the device

Section	Amendments
Associated firmware (imc DEVICES)	The term "Firmware group" had been used in two meanings.
The network	The example of a complex network with devices belonging to firmware group B has been expanded.

imc WAVE

Section	Amendments
Monitoring the overmodulation and under range ^[116]	Section added

Setup - Device (general)

Section	Amendments
Transferring an experiment to other devices	Note added concerning devices of Firmware-group B, indicating what constraints to observe when transferring a configuration from one device to another.
Connect and disconnect ^[147]	
Device properties ^[153]	imc REMOTE SecureAccess has been removed from the device properties. The obsolete TLS 1.0 technology is no longer utilized.
Metadata	New function described: Saving metadata with the channel

Setup pages - Configuring device

Section	Amendments
Sampling rate and the Aggregate sampling rate	Info added: Active monitor channels are included in the calculation of the aggregate sampling rate at the reduced rate.
Trigger and events - Hysteresis ^[293]	Info added: Automatic adjustment of the hysteresis to the measurement range
Sampling & Preprocessing	New reduction method for imc ARGUSfit: Monitor channel as envelope type

Section	Amendments
Trigger and events - Sources and events ²⁸⁷	Note added: Channels having the preprocessing " <i>Envelope</i> " can not serve as a trigger source.
Synchronization ¹⁸⁸	imc ARGUS <i>fit</i> -devices can now be synchronized via GPS by means of a connected GPS-mouse.
Storage media in the device - Formatting ³³⁹	Tool for formatting memory cards for imc ARGUS <i>fit</i> -devices described.

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
Envelope type monitor channel (min-/max-channel)	The functioning of the data reduction in envelope-type monitor channels for imc ARGUS <i>fit</i> is described.
Parameter set > Mapping Import parameters > Mapping	<ul style="list-style-type: none"> "<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: <ul style="list-style-type: none"> "<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 ⁸¹¹	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 ¹¹⁰⁵	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
Execute device action	Note added, for the action "Start": No calculations or analyses performed on the PC side, such as imc FAMOS-calculations, are started. Only the devices are started.
Set variables	<ul style="list-style-type: none">• 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 "Load Variable", only user-defined variables having the scope "temporary" can be overwritten. However, all other user-defined variables can also be overwritten.

17.1.3 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.

Setup pages - Configuring device

Section	Amendments
Events and circular buffer ³¹⁷	New section summarizing the settings for Events and Circular buffer.
Trigger and events Hysteresis ²⁹³	Effect of the hysteresis on the measurement start.
Exchanging values among the devices	For exchanging or processing values across devices, there are a variety of approaches and methods. The collection of these is now presented here.
Fieldbusses ³⁶⁸	The description of the fieldbuses has been moved to the section " <i>Setup pages - Configuring device</i> ".
CAN-bus interface ³⁷⁰	Description of the new CAN FD fieldbus assistant for imc ARGUSfit.

Data analysis and signal processing

imc Online FAMOS

Section	Amendments
User interface	In the imc Online FAMOS Editor's toolbar there is now a button " <i>Undo</i> ".
Menu	In the menu, the item " <i>Undo all</i> " has been added.

Panel

Section	Amendments
Preview in Windows Explorer ¹¹⁰⁵	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

17.1.4 in Doc. Rev. 6.2

Setting Up - Software

Section	Amendments
Installation variant "User-defined"	Installation steps for imc STUDIO, imc DEVICES and imc DEVICEcore have been merged.
Notes and troubleshooting	Updated " <i>User Account Control</i> " images.
Unattended installation - Silent installation	In the step-by-step instructions, the information has been added which indicates that the data carrier's content must first be unpacked by the installer.

Setup - Device (general)

Section	Amendments
Transferring an experiment to other devices	In the " <i>Open Experiment</i> "-dialog, the selected can now not only be opened but also transferred to a different device.

Setup pages - Configuring device

Section	Amendments
Sampling & Preprocessing	Detailed description of the " <i>Preprocessing</i> ".
Channel table	Included: Maximum channel count per device in firmware group B (imc DEVICEcore)
Storage media in the device ³³⁷	Separation of the description for firmware groups A and B.
Storage media - Formatting ³⁴⁵	Information on cluster sizes updated.
Filter settings ²⁵⁸	The filter description was extended.

Channel types, variable types, data types and data storage formats

Section	Amendments
Monitor channels	New: Monitor channels for imc ARGUSfit devices
imc Format Converter ⁷⁸⁶	Description of the Format converter included in this chapter.

Sequencer, events and commands

Command Reference

Section	Amendments
Transfer device settings	The command "Transfer device settings" is no longer supported.

17.1.5 in Doc. Rev. 6.0

Setting Up - Software

Section	Amendments
System requirements	The system requirements have been adjusted.

Setting Up - Connect the device

Section	Amendments
Device connection via LAN	By means of the device interface's configurations dialog, you can directly open the PC's adapter settings.

Setup pages - Configuring device

Section	Amendments
Storage	Note added, indicating that when the option " <i>Save trigger events in individual files</i> " is activated, the column " <i>Event time</i> " must be added manually in the Data Browser.
Sampling & Preprocessing	Note added, indicating that results of the function " <i>TransRec</i> " and DI-port-compressed channels are blocked from transfer to the PC.
Curve properties	<ul style="list-style-type: none"> Parameter renaming implemented. The description of the parameter "<i>Range</i>" has been revised - The choice "<i>Auto</i>" now also transfers a range to scale-Widgets. Description supplemented: Setting the expected signal value range in addition to the measurement range
Range & Scaling	New notes on rounded measurement ranges.

Panel

Section	Amendments
Curve window colors on the Panel page ^[1103]	Description adapted: When the colors are used for the display and when for the printout.
Properties - Widget	Range: New behavior described: Channels having a measurement range now also always transfer a display range.

17.1.6 in Doc. Rev. 5.4

Section	Amendments
Miscellaneous	Some missing components and functions were removed from the documentation.

Setting Up - Software

Section	Amendments
Select the products / Installation type	<ul style="list-style-type: none"> When you select the installation variant "<i>Demo</i>", imc WAVE is also selected and installed. New overview table of all installation variants and their respective product contents.
Product configuration / Licensing	Removal of the note indicating that the required licenses are shown summarized at the bottom, and that the Automation requires a separate license.
Error causes upon starting	New: Corrective actions for cases where the logbook appears with error messages directly upon starting imc WAVE.
System requirements	The system requirements have been expanded to include the required and installed OS-components.

Setting Up - Connect the device

Section	Amendments
Connecting via LAN in three steps 	The dialog for configuring the device's IP-address, also returns the IP-address of the PC. Thus the first step has been eliminated, which presented various ways to find the IP-address in Windows.

imc WAVE

Section	Amendments
Playback	Playback and re-calculation of measured data - missing function from Version 5.2 activated

Setup - Device (general)

Section	Amendments
Assistants	The dropdown-box for device selection has been eliminated. Instead, all device tabs are displayed from the very beginning.

Setup pages - Configuring device

Section	Amendments
Additional parameters	New: Display of instantaneous values for 3rd-party device channels, incremental counter inputs, device-variables, etc.
Transfer and storage on the PC 	Information gathered: Data transfer - Access via Windows Explorer
Storage in the device 	
Access via Windows Explorer 	Description added: Access to the saved measurement data in the device for devices of the firmware group B (imc DEVICecore).
Trigger and events - Hysteresis 	New: The event hysteresis of the devices belonging to the firmware group B (imc DEVICecore).
Special modules and third party devices	Combining the documentation under the device configuration

Section	Amendments
Third party device: ChannelLoader ^[462]	Notes on the system's treatment of missing files and relative paths.
Third party device: FunctionSimulator ^[463]	The 3rd-party device " <i>FunctionSimulator</i> " is now available as a complete 3rd-party device which can be activated as desired.

Channel types, variable types, data types and data storage formats

Section	Amendments
imc Data format	Notes on imc2 and imc3 file formats.
Data file format - Saving measured data during the measurement	Note: With which version and which device were the measured data generated?

Panel

Section	Amendments
Operation - Docking	Widget docking is now only possible by means of the Properties and no longer via the context menu.

17.1.7 in Doc. Rev. 5.3

Section	Amendments
Miscellaneous	Some missing components and functions were removed from the documentation.

Setting Up - Software

Section	Amendments
Installation Step by Step > User-defined	Statement clarified, that imc WAVE needs to be activated upon making the product selection. In the picture, imc WAVE is highlighted in color.
Important settings	Another important settings: " <i>Checking and modifying the time zone</i> "
Installation - Step by step	imc WAVE 5.2 and imc WAVE 2022 can now be installed in parallel.
System requirements	The list of operating systems supported has been revised: New: Windows 11

imc WAVE

Section	Amendments
Experiments, projects and the database ^[70]	Section added
Spectrum Analyzer - Derived channels	The calculation functions belonging to Inline Analysis " imc WAVE Structure " ^[114] are now also offered in the " <i>Spectrum analyzer</i> ".
Inline Analysis - imc WAVE Vibration	imc WAVE Vibration ^[110] has been supplemented with the function: " <i>Machine diagnosis</i> " (in accordance with ISO 10816/20816).

Setup - Device (general)

Section	Amendments
Report of the Setup-configuration	The following function is no longer available: Report creation via the Windows Explorer

Setup pages - Configuring device

Section	Amendments
Create a new sensor characteristic curve ^[467]	Chapter on generating a characteristic curve supplemented, including how to generate curves using the command: " <i>Measure a user-defined characteristic curve</i> ".
Data recording with sampling or with time stamp	The description of data recording has been moved from the section on the Fieldbus to the general information portion.
Measurement types	In the chapter, information on the different types of measurements has been brought together: <ul style="list-style-type: none"> • Temperature measurement ^[349] • Bridge measurements with wire strain gauges ^[352] • LVDT ^[358] • Incremental counters channels ^[359]

Section	Bug-fix
List of sensor supported (TEDS/imc SENSORS) ^[470]	Title of the table was wrong: " <i>NOT supported sensor information (TEDS or imc SENSORS)</i> "

Channel types, variable types, data types and data storage formats

Section	Amendments
All	Re-arrangement: the information had previously been distributed among various sections.
Monitor channels	New: Description of the Envelope-type monitor channels
Data types	New: Description of the various data types and the properties: " <i>Segmented</i> " and " <i>Event-based</i> "

Panel

Section	Amendments
Special Widgets	The following Widget is described: <ul style="list-style-type: none"> • DIO
Zone dialog > Other options and settings	Note added: " <i>Zone-texts are ignored when ...</i> "
Properties - Widget	The description for the option "Decimal Places" has been extended.

Sequencer, events and commands

Command Reference

Section	Amendments
Measure a user-defined characteristic curve	Description added
Information and Tips  729	Info about breakpoints and bookmarks in sequences
Export  765 / Import  770 parameters	Description merged with other format descriptions in the chapter " Channel types, variable types, data types and data storage formats "  730

Index

(

- () Opening Closing Bracket 546
- (%) Modulus 544
- (-) Subtraction 543
- (*) Multiplication 543
- (/) Division 543
- (;) Semicolon 546
- (^) Power 543
- (+) Addition 543
- (<) Less than operator 545
- (<=) Less than or equal operator 545
- (<>) Inequality operator 545
- (=) Equal Operator 544
- (=) Equal Sign 544
- (>) Greater than operator 546
- (>=) Greater than or equal operator 546

<

- <auto> (marker) 1039

1

- 1/3-octave, octave labeling 852
- 16-bit integer 262

2

- 24-bit mode 262
- 2-point calibration 281
- 2-point controller 716
- 2-point scaling 249

3

- 32 bit integer
 - pv-variable 741
- 3D 852
 - Color palette 993
 - Display 900
 - Extras 997
 - Isolines 996
 - options 989
 - Perspective 995
 - Rotate 998
 - surface 990
- 3D-bars 990
- 3PDI
 - AudioDevice 461
 - ChannelLoader 462
 - FunctionSimulator 463

4

- 4D
 - 3D + color 983

A

- A2L 413, 458
- A2L import 411
- AAF 258
 - Preprocessing 264
- AAF-filter 258
- ABCRating 547
- Abs 547
- Absolute date, time (curve window) 943
- Absolute time
 - Display for synchronized start 225
- Accessing the data carrier 340
 - Error 348
- AccuLength 548
- AccuMax 549
- AccuMean 549
- AccuMin 550
- AccuRMS 550
- AccuStDev 551
- Acos 551
- Acquired
 - Point 249
- Acquisition time
 - UPS 154
- Acquisition time after power fail (s) 154
- Action 286
 - Setting bits 295
 - Starting data recording 295
 - Stop recording of data 295
- Actions and targets
 - Trigger 295
- activating map mode retroactively 975
- Adapt
 - Menu (curve window) 1090
 - toolbar (curve window) 1090
- Adapt Panel page to window size 1125
- Adapt Panel page to window size (keep aspect ratio) 1125
- Adapt Panel page to window size (keep Widget size) 1125
- Adapt Widgets to the size of the Panel page 1125
- adapting maps 973
- Adaption channel selection
 - More Channels in Curve Window 933
- Add device 42
- Add device interface 343

- Add file
 - Command Load/Refill variable 1196
- Add variable
 - Command Load/Refill variable 1196
- Adding a device (Add) 343
- Addition 543
- Adjusting variable 467
- adjustment 8
- Advanced Configuration
 - Interface configuration 49
- Advanced display
 - Zone 840, 844
- Aggregate sampling rate 267
- aliasing 258
- Aliasing effects 258
- Aligning Widgets 827
- All available 807
- Allow a change of direction 1113
 - Prevent a change 1113
- Allow data export 1174
- Allow discarding of all measurements 1174
- Allow selection of the export folder 1174
- Alphabetical 243
- alternative IP and FW-Update 46
- Amount 298
- Amplifiers
 - Address 158
 - Module address 158
 - Modules 158
 - Permanently installed amplifiers 158
- Amplitude-Shift 963
- Analog channels 238
- Analog inputs 238
- Analog outputs 238, 514
- AND 552
 - Trigger event 295
- Angle (differential, abs, sum) 365
- Angle measurement
 - Incremental counters 365
- antialiasing filter 258
- Anti-aliasing filter 259
 - Low pass 259
- anti-aliasing filter (AAF) 258
- Appearance
 - Property - Widget 832
- Application option 1237
- Arrangement
 - Widgets 827, 828
- Arith. Mean 267
- Arithmetic mean
 - Preprocessing 264
- ARON measuring circuit 530
- Arrangement
 - Axes 945
 - Configuration (curve window) 1007
- Arrangement (Widgets)
 - Align Bottom 827
 - Align left 827
 - Align Right 827
 - Align top 827
 - Cascade 827
 - To Background 826
 - To Foreground 826
- ARXML 413
- As presettings
 - Curve Window 1008
- ASCII
 - imc Format Converter 788, 1159, 1202
 - imc Format Converter Command 1159
- ASCII display (curve window) 873
- Asin 552
- Aspect ratio for RGB image 946
- Assignment 263, 417, 432
- At defined time 234
- Atan2 552
- AudioDevice 461
- auto
 - Buffer time 323, 324
- Auto filename 308
- Auto-IP 49
- Auto-IP (DHCP + APIPA) 46
- Automatic table length 1113
- Automatic timed start 234
- Autosize
 - Property - Widget 833
- Available Channels
 - Context menu 929
 - More Channels in Curve Window 929
- Available events 268, 309, 317
 - Display and calculations 317
- Axes
 - Arrangement 945
 - labeling 852
 - Scaling 935
 - Text 947
- Axes List 926
 - Symbols 926
- Axes Navigation Bar (curve window) 1101
- B**
- Back (curve window) 1057

- Background
 - Property - Widget 832
 - Background color
 - Property - Pages 1123
 - Background image
 - Property - Pages 1123
 - Background image layout
 - Property - Pages 1123
 - Background plain colored
 - Property - Pages 1123
 - background picture (curve window) 970
 - Balance at device startup 272
 - Balance at startup 272
 - Balance values import 773
 - Balancing values
 - balance value after change of device 275
 - ex-/import 275
 - Balancing with setted offset 251, 276
 - Bar meter (curve window) 852, 879
 - Base path for measurement data storage 330, 1238
 - BaseTrigger 285
 - Beeper 514
 - Overflow 500
 - Begin action 295
 - Begin=0 295
 - Begin=1 295
 - End=0 295
 - End=1 295
 - start 295
 - stop 295
 - Binary 1117
 - BitAnd 553
 - Bitmask
 - Property - Widget 833
 - BitNot 553
 - BitOr 553
 - Bits 730
 - BitXor 554
 - Block comment
 - imc Online FAMOS 517
 - Bookmarks
 - imc FAMOS 729
 - boolean variables
 - from files 523
 - IF conditions (OFA/IFA) 523
 - Border
 - Background 832
 - boundaries (curve window) 1024
 - boundary intervals (curve window) 1024
 - Boxplot (curve window) 914
 - Breakpoints
 - imc FAMOS 729
 - Bridge 275
 - Bridge factor N 252
 - bridge measurement 351
 - Bridge mode 252
 - Bridge resistor 247, 252
 - Bridge supply 273
 - Bridge supply (On/Off) 273
 - Bridge unit 253
 - Browse in workspace
 - Layout 1185
 - Page 1185
 - Parameters 1185
 - Bubble Chart 916
 - Bubble Plot
 - change edge 916
 - mit Farbkanal 921
 - Standard 916
 - Buffer time 322
 - auto 323, 324
 - increase 475
 - Bus Decoder 386, 407, 422
 - Button 1114
- C**
- Cable compensation without sense 282
 - Calibrating the amplifiers 242
 - calibration 8
 - Calibration date 242
 - CAN
 - duplicate values 460
 - CAN analyzer 439
 - CAN bus
 - extended-ID and TSA evaluation 396
 - receive a message 396
 - send a message 396
 - Send standard message in extended node 396
 - CAN channel
 - Data formats 380
 - CAN channel not appearing in the imc Online FAMOS 542
 - CAN database import
 - enable 156, 157
 - CAN FD 374
 - Activate 383
 - CAN FD Node (ARGUS)
 - Terminator 420
 - CAN Monitor channel 392
 - CAN protocol
 - CANalyser-MDF format 754
 - CAN send with imc FAMOS 399

- CAN-1 Protocol 385
- CANALYSER 396, 424
- CANALyser-MDF format 754
- CAN-Botschafts-Protokoll 424
- CAN-Bus 370
 - A2L 458
 - A2L import 411
 - Analog CAN channel 403
 - Analoger CAN Kanal 426
 - Assignment 417
 - Assistant 381
 - assistant (ARGUS) 418
 - Balancing a bridge module via CAN 433
 - Botschaft Definition 424
 - Botschaft empfangen 424
 - Botschaft senden 425
 - cabling 371
 - CAN protocol 376
 - CBA import, export 411
 - CCP 455
 - Channel - Error Handling 409
 - Channel - State 407
 - Channel - Validity 407, 408
 - Channel configuration 417
 - CURVE 458
 - DBC import, export 411
 - Diagnostic On CAN 451
 - DiagOnCAN 451
 - Digital CAN channel 406
 - D-SUB - pin configuration 371
 - ECU configuring 441
 - Editing a CAN configuration 382
 - Error handling 376
 - Expert settings 384
 - Format 376
 - GMLAN 457
 - ID 376
 - Initialization 394
 - Intel format 403
 - Intel-Format 426
 - J1939 401
 - KWP2000 446
 - KWP2000 TP2.0 451
 - log message 396
 - Log-channel 407, 417
 - MDF 396, 424
 - Message log 396
 - Messages 375
 - Motorola format 403
 - Motorola-Format 426
 - Number of CAN nodes 375
 - Number of samples per bit 384
 - OBD-2 452
 - Operating (ARGUS) 419
 - Options 416
 - protokolliere Botschaft 424
 - Quick Tutorial 382
 - Receiving a message 396
 - Sample location in bit time 384
 - Sampling interval 417
 - Sampling time 417
 - Send with imc Online FAMOS 521
 - Sending messages 397
 - Sensor initialization 394
 - Shunt calibration 435
 - Slope detection 384
 - Specifications 375
 - Terminator 372
 - Time stamp 376, 417
 - Transfer rate 373
 - UDS 458
 - VAL_BLK 458
 - Validity 385
 - XCP 457
 - Y-adaptor 371
- CAN-Bus (ARGUS)
 - Assignment 432
 - Channel settings 431
 - Sampling interval 432
 - Sampling time 432
 - Time stamp 432
- CAN-Bus (ARGUS) - Channel
 - Channel - Error handling 430
 - Intel 426
 - Motorola 426
 - number format 426
 - replacement value 430
 - Scaling 426
 - Start byte 426
 - Unit 426
 - Validity 429
- CAN-Bus (ARGUS) - Message
 - extended-ID and TSA evaluation 424
 - receive a message 424
 - send a message 424
 - Send standard message in extended node 424
- CAN-Bus (ARGUS) - Node
 - Activate (ARGUS) 419
 - Baud rate 419
 - Clear Node settings 423
 - Connection 419
 - Export Node settings 423
 - Extended format 419
 - Extended+ format 419

- CAN-Bus (ARGUS) - Node
 - format 419
 - Import Node settings 423
 - Initialization 420
 - Log channel 422
 - Node context menu 423
 - node definition 419
 - Sensor initialization 420
 - Standard format 419
 - Validity 422
- CAN-Bus (Menu)
 - Edit 411
 - Extras 416
 - File 411
 - Insert 412
 - Transform messages 411
 - transform receiving- in send-messages 411
 - XCP 411
 - XCPplus 411
- CAN-Bus Definition
 - Baud rate 383
 - CAN FD 383
 - Connection 383
 - Format 383
 - Nodes 383
 - Terminator 384
- CAN-Bus Gültigkeit
 - Acknowledge 422
- CAN-Bus Synchronization
 - CAN-1 Protocol 385
- CAN-Bus Validity
 - Acknowledge 386
 - Deactivate device if no measured value x s 390
 - Designation 392
 - Diagnostic Message (DM) 387
 - Errorframes 390
 - J1939 387
 - Monitor channel 392
 - Support of Wake On CAN 390
 - Wake on CAN 390
- CAN-Bus-Assistant 418
 - import formats 413
- CAN-Bus-Data (curve window) 873
- Cancel Button
 - Show message box 1188
 - Text (Button caption) 1188
 - Visible (Hide) 1188
 - Visible (Show) 1188
- CAN-database 411, 416
- CAN-FD (ISO 11898-1:2015) 370
- CAN-message
 - Sensor delay time 402
- CanMsg 711, 712
- CanMsg.GetData 713
- CanMsg.SetData 714
- CanMsg.Transmit1_S1 714
- Caption in tables 1114
- Cascade Widgets 828
- CBA 413
- CBA import, export 411
- CCP 378, 455
- ccv file
 - loading 1064
 - saving 1066
- CE Certification 8
- Cell title 1114
- Centered
 - Background image 1123
- Certain parts from data (curve window) 1000
- Certificates 8
- CF card (Compact Flash) 341
- CFast Storage medium 341
- Change requests 7
- changing the scale 1081
- Channel
 - CAN-Bus 412
- Channel balance 275
- Channel comment 242
- Channel count 240
- Channel data type 262
- Channel definition (Setup page) 242
- Channel file name 308
- Channel info
 - More Channels in Curve Window 934
- Channel name 242
 - imc Online FAMOS 517
- Channel name assistant 243
- Channel names
 - Fieldbus 369
- Channel number 273
- Channel numbering 158
- Channel playback 462
- Channel status 242
- Channel table 238
- Channel wiring 248
- ChannelLoader 462
- Channels 238
- Channels with invalid properties 416
- Charact 555
- Characteristic curve 465
 - creating 467
- Charakteristic (filter) 258

- Check - Syntax 507
- Check Box 1114
- CiA Draft Standard 370
- Circular buffer memory 475
- Circular buffer operation 321
- Circular buffer time 268, 309, 318
 - Display and calculations 318
- Class-counting 537
- CIHistogram 556
- Clipboard 1074
 - curves 1052
- clipboard format (curve window) 1085
- Clipboard settings 1074
 - Font 1074
 - Line style 1074
 - Line width 1074
 - plot 1074
 - Print 1074
 - printout 1074
 - Ticks 1074
- CILevelCrossing 557
- CIMarkov 558
- Clock 1107
- Clock types
 - DCF77 193
 - GPS 193
 - IRIG-B 193
 - NTP 193
 - PTP 193
 - SyncRTC 193
 - VRTC 193
- Close folder on command 335
- CloseSaveInterval 335, 559
- ClRainFlow 560
- ClRainFlowRes 561
- ClRainFlowTM 562
- ClRainFlowTMRes 563
- ClRangePairCount 564
- ClRevolutionsHistogram 564
- ClRevolutionsMatrix 565
- ClTrueMax 565
- ClTwoChannelHistogram 565
- Cluster size 756
 - FAT32 338
- CodeRange 566
- coldjunction compensation 350
- Color 242, 271
 - Title 831
- Color components 1078
- Color grandient
 - Background color 1123
 - color map (curve window) 852, 983
 - Display 888
 - general 891
 - ISO-lines 897
 - measure mode 1019
 - Waveforms with associated x-,y-,z-variables 899
 - color of Y-axis 949
- Color palette
 - 3D 993
- color palette (curve window) 983
- Color spectrum 888
- color-coding thermocouples 349
- Coloring
 - Property - Widget 832
- Colors (curve window)
 - Copy 1078
 - Curves 1078
 - Screen/Printer 1078
- Column captions 1114
- Column ID
 - CurrentValueDisplay 274
 - eAllocation 263, 417, 432
 - eBalanceAtDeviceStart 272
 - eBridgeEModule 253
 - eBridgeEps 253
 - eBridgeFactor 252
 - eBridgeMode 252
 - eBridgeN 252
 - eBridgeResistor 247, 252
 - eBridgeSupplyOnOffAction 273
 - eBridgeUnit 253
 - eChannelComment 242
 - eChannelMode 247, 254, 262
 - eChannelName 242
 - eCorrection 248
 - eCoupling 247, 252
 - eCouplingsDC 247
 - eCurveColor 242, 271
 - eCurveYAxisMax 271
 - eCurveYAxisMin 271
 - eCurveYAxisOption 271
 - eDataType 262
 - eDuration 261
 - eEnumeratedChannelNumber 273
 - eHistogramSaveInterval 270
 - eHistogramUpdateInterval 270
 - eHysteresis 255
 - eInputRange 255
 - elsolatedThermoCouple 249
 - eLevel 255
 - eMaximum 255
 - eNullImpuls 254

- Column ID
 - ePlugInName 272
 - ePolarization 249
 - eProcessing 266
 - eProcessingPoints 266
 - eRange 248, 249, 253
 - eSampleTime 261
 - eSignal 254
 - eSignalform 255
 - eStartEdge 255
 - eStatus 242
 - eStopEdge 255
 - eSupply 248
 - eUserOffset 250, 255
 - eUserScalingFactor 250
 - eUserScalingFactorENC 254
 - eUserUnit 250, 255
 - eWiring 248
 - eXFormatVariable 263, 417, 432
 - PreprocessedSampleRate 266
 - PreprocessedSampleTime 266
 - SampleCount 261
 - SampleRate 261
 - UserScalingFactor_Reciproce 250
- Column title 1114
- Column value 243
- Combination mode 361
- Combinations of events 295
- Combined event 287
- Combo Box 1114
- Command
 - Delete supplemental file 1185
 - Delete Variable 1198
 - Execute device action 1176
 - Execute file 1156
 - Execute imc FAMOS sequence 1149
 - Export Panel page 1170
 - Export supplemental file 1183
 - Export Variable 1192
 - imc FAMOS project 1148
 - imc FAMOS sequence 1149
 - Import imc Inline FAMOS source code 1165
 - Import supplemental file 1184
 - Load variable 1194
 - Measure a user-defined characteristic curve 1175
 - Open Logbook viewer 1189
 - Panel page - Export 1170
 - Panel page as dialog 1168
 - Panel page entfernen 1171
 - Refill variable 1194
 - Set measurement number 1166
 - Set variables 1198
 - Show Panel page as dialog 1168
 - Stop Sequencer 1165
 - Supplemental file - Delete 1185
 - Supplemental file - Export 1183
 - Supplemental file - Import 1184
- Command line parameters 31
- Command Reference 1148
- Comment 176, 242, 467
 - Property 527
 - Title - Default Source 831
- Comment a line
 - imc Online FAMOS 517
- Comment several lines
 - imc Online FAMOS 517
- Communicator
 - create scroll link 1058
 - Transfer to imc FAMOS 1067
- Compact Flash 341
- comparator conditioning
 - Incremental counter 363
- Comparison operators 502, 520
- Components
 - Product configuration 28
- Computation uncertainty in summations 540
- Conditional AND
 - Trigger event 295
- Configuration (curve window)
 - Arrangement 1007
 - loading 1064
 - Menu 1073
 - saving 1066
- Configuration transfer
 - Curve window 1100
- Connect 147
- Connect (curve window) 1057, 1058
 - cross-section with 3D 1059, 1060
- connect device 40
- Connect XY with 2nd curve window 1058
- Connecting via LAN 40, 49
- Connecting via WLAN 53
- Connection 186
 - to Device 30
- Connection failed - transfer configuration 1238
- Connection status 186
 - Variables 739
- Connection via a direct address 52
- Connector 272
- Consider defined message length 393
- Context menu
 - Available Channels 929
 - Curve window 1100

- CONTROL 1219
- Control commands 501, 502, 518
 - activate 501
 - Array 519
 - Comparison operators 520
 - Create a process vector variable 519
 - Create a variable 519
 - Data array 519
 - deactivate 501
 - Single value 519
 - SingleValueChannel 519
 - Trigger 302
- Control functions 502
- Control of synchronization 229
- Control sync state - PC 230
- Control units 440
- Control units (ECU) 378
- Control units in imc Online FAMOS 445
- Controller functions 716
- Conversion 71
 - Database 18
- Coordinate system size 852
- Coordinates (map) 982
- Copy
 - Logbook 1215
- Copy & Paste
 - Curve window 1100
- Copy data to the PC 338, 342
- Copy with details
 - Logbook 1215
- Correction 248
- Cos 566
- Counter 359
- Counter hysteresis 255
- Counter input range 255
- Counter scaling factor 254
- Counter signal 254
- Counter signal shape 255
- Counter start edge 255
- Counter stop edge 255
- Counter switching level 255
- Coupling 247, 252
- Coupling is DC 247
- Create a new characteristic curve 467
- Create a process vector variable 519
- CreateVChannel 566
- CreateVChannelInt 567
- CRFX
 - Module address 158
- CRFX Modules 241
- Critical channel fill level 475
- Cross sections
 - Lines 965
- CrossCorrelation 567
- Cross-section with 3D
 - connect 1059, 1060
- CRXT
 - Module address 158
- CRXT Modules 241
- CtPID 717
- CtPID.Calc 718
- CtTwoPos 719
- CtTwoPos.Calc 720
- cumulative measurements 362
- Current device configuration
 - Interface configuration 49
- Current measurement 1133
- Current PC configuration
 - Interface configuration 49
- Current value 274
- Currently not reachable
 - Interface configuration 49
- CurrentValue 568
- Cursor
 - Measure window 1016
- CURVE 458
- Curve color 242, 271
- Curve configuration
 - loading 1064
 - saving 1066
- Curve properties (Setup page) 270
- Curve window
 - changing the scale 1081
 - clipboard 1052
 - clipboard format 1085
 - construction 924
 - context menu 1100
 - Date/time absolute 852
 - Days/hours/minutes relative 852
 - Display of synchronized start 225
 - Double-click on empty areas 1085
 - Enable horizontal measurement cursor 1085
 - Fill mode 999
 - Font 1081
 - Global Handling 1085
 - Graphic export 1085
 - graphic export as Vector graphic 1081
 - hiding data 924
 - history 1081
 - import a picture or map as background 972
 - Introduction 850

- Curve window
 - legend 967
 - Line-Shift 1081
 - link 1081
 - Max. delay [ms] 1081
 - Measure 1014
 - Measure window 1014
 - More Channels 924
 - Navigation in x-direction 1085
 - New coordinate systems 926
 - New x-axis 926
 - New y-axis 926
 - Oscilloscope mode 999
 - PDF orientation 1085
 - Rezoom 1085
 - scroll links to tables 1058
 - Scroll mode 999
 - selection mode 1081
 - settings 1081
 - Size of measurement point 1085
 - sound output 1081
 - Start delay [ms] 1081
 - Stretch mode 999
 - Time-Shift in the ccv 1081
 - touchscreen operation 953
 - Transfer to imc FAMOS 1085
 - Undo 1081
 - Updating 1137
 - XY plots 931
 - Zoom 1085
 - Custom base path for measurement data storage 330, 1238
 - Custom folder naming for measurement data storage 330, 1238
 - Customer support
 - Tech support 7
 - Cut sound
 - sound output (curve window) 1095
 - Cut-off frequency 1 (filter) 258
 - Cut-off frequency 2 (filter) 258
- D
- DAC 514
 - Scaling 514
 - DAC does not output every value 542
 - Data format
 - imc2 785
 - imc3 785
 - Data formats
 - CAN-Bus 380
 - Data loss 318
 - Data overflow 312, 322, 475
 - Data rate 476
 - reduce 475
 - Data recording with sampling 472
 - Data recording with time stamp 472
 - Data saving 306
 - Data Saving Assistant 1172
 - Allow data export 1174
 - Allow discarding of all measurements 1174
 - Allow selection of the export folder 1174
 - Default button 1175
 - Default path 1174
 - Delete original files 1174
 - Deleting a measurement 1174
 - Discard measurement 1174
 - Export measurement settings 1174
 - imc Format Converter 1173
 - Measurement comments 1173
 - Parameter set 1173
 - Prompt to confirm discard 1174
 - Setup pages 1173
 - Data storage 328
 - Data stream 306
 - Data transfer 308, 338, 342
 - FTP access 344
 - Password 338
 - Storage medium 338, 342
 - Data transfer (Setup page) 268
 - Data type 262
 - 16-bit integer 262
 - 24-bit mode 262
 - Float (24-bit mode) 262
 - Data types 730
 - Memory requirement 756
 - Data Browser 82, 804
 - All available 807
 - current measurement in the Data Browser 806
 - Event time 807
 - Load variable 811
 - Measurement number 809
 - measurements in the Data Browser 806
 - Refill variable 811
 - renaming measurements 806
 - saved measurements 806
 - Data Browser columns
 - Category 805
 - Comment 805
 - Event time 805
 - Metadata columns (optional) 805
 - Name 805
 - Numbering column (#) 805
 - Unit 805

- Database
 - conversion 71
- Database conversion 18
- Database folder 71
- DATABASE/DATABASES 1220
- Date and time 1107
- Daylight saving time 192
- Days/hours/minutes relative (curve window) 943
- dB 569
- DBC 413
 - J1939 401
- DBC import, export 411
- DC (Coupling is DC) 247
- DCB 416
- DCF
 - Synchronization 200
- DCF77
 - External clocks 200
- D-component 533
- D-Controller
 - imc Online FAMOS 532
- Deactivating the pv-variables 741
- Decimal 1117
- Decimal places
 - Property - Widget 833
 - Zone 848
- Default 709
- Default button
 - Data Saving Assistant 1175
- Default path
 - for export 1174
- Default printer (curve window) 1069
- Default Source
 - Title 831
- Delay 467
- DelayBuffer 569
 - Fill 571
 - Next 572
 - SetSize 573
- DelayBuffer.Fill 571
- DelayBuffer.Next 572
- DelayBuffer.SetSize 573
- DelayLine 574
- Delete
 - experiment 73
 - project 73
 - Set variables 1198
 - Supplemental file 1185
- Delete original files 1174
- Delete sample 1113
- Delete supplemental file 1185
- Delete Variable 1198
 - Show dialog 1198
 - Treat error as warning 1198
 - Variable name 1198
 - Variables list 1198
- Deleting a measurement
 - Data Saving Assistant 1174
- Deleting sensor information from channel 467
- Deleting the process vector backup file 744
- Delta-connection 531
- Delta-x 527
- Delta-z 527
- Derived channels
 - recalculate 85, 86
 - Spectrum analyzer 98
- Description 184
- Design Mode 820
- Designation data folders 328
- Deviation of system time from NTP 741
- Device
 - add 42
 - connect 40
 - connection 30
 - Properties 153
- Device configuration 153
- Device connection via LAN 49
- Device control 187
 - Starting measurement 150, 187
- Device group 146
- Device information
 - Interface configuration 49
- Device Interfaces 47, 49
- Device is known 171
- Device is selected for measurement 186
- Device name 186
- Device overview 146
- Device properties 153
- Device selection 171
- Device SN 186
- Device specification 186
- Device system variables 738
- Device Table 186
- Devices - Documentation 64
- Devices deselected 1238
- DFilt 574
- DHCP 44, 49
 - DHCP ClientID 43, 46
 - DHCP Hostname 46
- Diagnostic Message (DM) 387
- Diagnostic On CAN 451

- DiagOnCAN 451
 - Diff 575
 - differential measurement procedures 362
 - Digital bits
 - Trigger target 295
 - Digital channels 238
 - Digital events
 - Signal = 0 291
 - Signal = 1 291
 - Signal change: 0 -> 1 291
 - Signal change: 1 -> 0 291
 - Digital inputs 514
 - Digital inputs / outputs (bits and ports) 239
 - Digital output
 - Trigger target 295
 - Dimension Lines 1033
 - DIN-EN-ISO-9001 8
 - DIO
 - Widget 1117
 - DirClosed 312, 336
 - Direction 1113
 - Directory structure 328
 - Trigger 329
 - Discard measurement
 - Data Saving Assistant 1174
 - Disconnect 147
 - Disconnect from running measurement 1238
 - DiskFreeSpace 575
 - DiskFreeTime 576
 - DiskRunDir 577
 - Display
 - 3D 900
 - bar meter 879
 - color map 888
 - Connector 154
 - GPS connector 154
 - Number (curve window) 868
 - Type of display 154
 - waterfall diagram 862
 - Display (curve window) 852
 - Display before trigger events 302
 - Display for synchronized start
 - Absolute time 225
 - Curve window 225
 - display options
 - third-octave axes 906
 - Display Setup dialog 1180
 - Display text
 - Zone 1115
 - display uncertainty 962
 - Display update interval
 - Histogram 270
 - Display variables 239, 487
 - Display, calculations of measured data 268
 - Display-GPS connector 154
 - DisplaySetButton 577
 - DisplaySetPage 577
 - Distance (differential, abs, sum) 365
 - Distance measurement
 - Incremental counters 365
 - Division 543
 - DM-services 387
 - DNS-server-address 43
 - Dock
 - Property - Widget 832
 - Dock (Widgets)
 - Below 829
 - Fill 829
 - Free 829
 - Left 829
 - Right 829
 - Top 829
 - Documentation 64
 - Documentation (Setup page) 184
 - Documentation of devices 64
 - Domain 43
 - DotNET (.NET) 1164
 - drag to WORD/PPT 1054
 - Drag&Drop
 - curve window 1100
 - DST 192
 - dual-track encoder 364
 - duplicate values
 - CAN 460
 - Duration 261
 - Duration of events 292
 - Dwell time 286
 - for Trigger events 292
 - max 292
- ## E
- ECU
 - CAN-Bus 412
 - Clear Diag Information 579
 - Cmd Return 579
 - Properties in common 459
 - Read Trouble Codes 578
 - Start Session 578
 - Stop Session 578
 - ECU (A2L) 411, 413
 - ECU configuration 440
 - ECU configuring 441

- ECU Control units 378
- ECU Monitor channel 392
- ECU requirements 378
- ECUClearDiagInformation 579
- ECU-protocol
 - enable 156, 157
- ECU-protocols 378
- ECUReadTroubleCodes 578
- ECUStartSession 578
- ECUStopSession 578
- edge (incremental counter) 366
- Edit
 - Menu (curve window) 1072
 - signals (curve window) 1020
- Edit box color
 - Property - Widget 833
- Editor 1114
- Elastic modulus 357
- Else 706
- Enable
 - ECU-Protokolle 156, 157
 - imc Online FAMOS 156, 157
 - imc Online FAMOS Professional 156, 157
 - imc STUDIO Monitor 156, 157
 - Online-Klassierung 156, 157
 - Online-Ordnungsanalyse 156, 157
- Enable horizontal measurement cursor 1085
- Encoder (Setup page) 254
- Encoder w/o zero impulse 254
- End online mode (curve window)
 - Scroll mode 999
- Entering range 290
- Enumerated channel number 273
- Enumeration classes 178
- Envelope
 - Preprocessing 264
- Envelope type monitor channel 732
- Envelope type monitor channels 736
- Environment variables of the OS 1236
- Equal 580
- Equal Operator 544
- Equal Sign 544
- Error (Logbook) 1214
- Error Message 103 183
- Ethernet bits 239, 486, 737
- Event 286, 298
- Event (curve window) 1000
- Event time
 - All available 807
 - Data Browser 807
- Event type
 - Digital 291
 - Level 290
 - Range 290
- Event-based data set 317, 731
- Events 268, 317
 - Available events 309, 317
 - Property - Widget 833
 - Trigger 287
- Events counting 360
 - Incremental counters 365
- Events from digital signals 291
- Events from virtual bits 291
- EXCEL
 - imc Format Converter 788, 1159, 1202
 - imc Format Converter Command 1159
- Excel-Display (curve window) 873
- Exchange display variables via network 236, 487
- Exchange real-time value 488
- Execute device action 1176
 - Starting measurement 150
- Execute file 1156
- Execute imc FAMOS sequence 1149
- Execute menu action
 - Command 1190
 - Widget 1111
- Execute program 1156
- Exiting range 290
- Experiment 70, 73
 - copy 73
 - delete 73
 - export 73
 - import 73
 - new 72, 73
 - save (as) 73
- Experiment from template 73
- Experiment has changed 1238
- Experiment option 1237
- Experiment template 73
 - export 73
 - import 73
- Experiment, open
 - Automatically upon starting imc STUDIO 31
 - With a specific imc STUDIO installation 31
- EXPERIMENT/EXPERIMENTS 1221
- ExpoRMS 581
- Export
 - experiment 73, 74
 - experiment template 73
 - Interface-configuration 47, 51
 - Logbook 1216

- Export
 - Measure window 1016
 - project 73, 74
 - Export all channels
 - Mapping 775
 - Export all channels and settings
 - Mapping 775
 - Export current data 334
 - Options 335
 - Export measurement settings 1174
 - Export Panel page 1170
 - Export parameters 765
 - Export supplemental file 1183
 - Export Variable 811, 1192
 - Add variables 1192
 - Always overwrite existing files 1193
 - File format 1193
 - File options 1192
 - Filename 1193
 - Folder 1192
 - Mapping instruction 1192
 - Measurement 1192
 - Measurement name 1193
 - Save all in a single file 1192
 - Save to one file per file format 1192
 - Show file options 1193
 - Show variables options 1193
 - Treat error as warning 1193
 - Variable name 1193
 - Variables 1192
 - Variables list 1193
 - ExpressCard Storage medium 341
 - Extended Format
 - CAN 383
 - Extended ID (CAN)
 - TSA evaluation 396
 - Extended+ Format
 - CAN 383
 - Extension for Windows-Explorer 342
 - Extras (curve window)
 - 3D 997
 - Lines 962
 - Menu 1088
 - Menu and Toolbar 1089
- F**
- Factor 250
 - Property - Widget 833
- Factor-offset scaling 249
- Factory calibration 275
- FAMOS-ZIP format
 - Limitations 753
- FAQ synchronize devices 225
- Farbkarte
 - Wirkung (Extras) 963
- Fast loading
 - Property - Pages 1124
- FAT16 756
- FAT32 756
- Fatal (Logbook) 1214
- FFT 583
 - FFTAplitudePhase 584
 - FFTAverage 585
 - FFTInverse 586
 - FFTReallmaginary 586
- fieldbus
 - Analog inputs 238, 263
 - Channel names 369
 - data recording 472
 - Digital inputs / outputs 263
 - Digital inputs / outputs (bits and ports) 239
 - General notes 368
 - Operating methodology 368
- File
 - Menu (curve window) 1063
- File comment 1193
- File format 309, 752, 753
 - CANalyser-MDF 754
 - Header 755
 - imc FAMOS 753
 - imc FAMOS ZIP 753
 - Key 755
- File header 755
- File name for channel 308
- File system 476
 - FAT32 338
- File system FAT16/FAT32 345
- Filesize (maximum) 345
- Fill mode
 - Curve window 999
- FiltBP 587
- FiltBS 587
- Filter
 - instable state 587
 - Setup table 166
- Filter concept 259
- Filter digital
 - Limits 541
- Filter types
 - AAF 258
 - band pass 258
 - high pass 258

- Filter types
 - low pass 258
 - without 258
 - filtering (curve window) 1022
 - Filtering (Setup page) 258
 - FiltHP 588
 - FiltLP 588
 - Firewall 44
 - Security alert 30
 - Firmware update 58
 - blocking 60
 - Logbook 60
 - password 60
 - Firmware version 57
 - Fix Axes 1057
 - Fixed-point (curve window) 868
 - Flags 31
 - Float
 - pv-variable 741
 - Float (24-bit mode) 262
 - floating-point notation 868
 - Floor 588
 - Flush right (curve window) 868
 - Folder naming for measurement data storage 330, 1238
 - Font
 - Property - Widget 832
 - Title 831
 - Font (curve window) 1081
 - Last value as number 868
 - For 708
 - For changed message clock rate 416
 - Format item 243
 - format marker (curve window) 1039
 - Formatkonverter
 - Export formats 1157
 - Formatting 339, 345, 476
 - Formatting (x-axis absolute)
 - Format line 1, 2 936
 - Forward (curve window) 1057
 - Frequency 368
 - frequency bands
 - third-octave display 906
 - Frequency error
 - Synchronization 220
 - From Variable
 - Range 833
 - FTP access
 - Data transfer 344
 - full bridge: 4 active strain gauges 357
 - full bridge: general 355
 - full bridge: Poisson full bridge (strain gauges adjacent branches) 356
 - full bridge: Poisson full bridge (strain gauges opposed branches) 356
 - full bridge: WSG 355
 - Full functionality for 30-day demo 22
 - full-bridge
 - LVDT 359
 - Fullscreen 1129
 - Automatically upon starting imc STUDIO 31
 - Launch 1129
 - Function 266
 - Function Reference
 - imc Inline FAMOS 542
 - imc Online FAMOS 542
 - FunctionSimulator 463
 - FW-Update and alternative IP 46
- 
- Gauge factor 252
 - GearRatio 589
 - General terms and conditions 8
 - GetDateTime 589
 - GetDuration 591
 - GetHistoValue 591
 - GetHistoValue2 592
 - GetLastError 593
 - GetSampleCount 596
 - GetSamplingTime 596
 - Global Handling (curve window) 1085
 - Global RAM 541
 - GMLAN 457
 - GMW 3110 Version 1.5 457
 - GPS
 - connector 154
 - External clocks 204
 - process vector variables 745
 - RS232 settings 747
 - Synchronization 204
 - GPS + DCF Master/Slave
 - Synchronization 205
 - GPS data as XY-plot 978
 - GPS-Channels 239
 - GPS-Display connector 154
 - Graphic export
 - Bitmap 1081
 - Vectorgraphic 1081
 - Graphic export (curve window)
 - Bitmap 1085
 - Vektorgraphic 1085
 - Graphical switch

- Graphical switch
 - Widget 1109
 - Greater 596
 - Greater than operator 546
 - Greater than or equal operator 546
 - GreaterEqual 596
 - Greek characters (curve window) 1097
 - Grid
 - Curve Window 989
 - Grid (curve window) 852
 - small x-ticks (-1 = auto) 852
 - small y-ticks (-1 = auto) 852
 - Grid (Panel)
 - Realign to Grid 1127
 - Snap to Grid 1127
 - Grid size
 - Property - Pages 1124
 - Group
 - Device overview 146
 - Variables 243
 - Group (Widgets)
 - Create Group 830
 - Dissolve Group 830
 - Enter Group 830
 - Leave Group 830
 - Group By Box
 - Setup table 165
 - Grouping
 - Setup table 165
 - Guarantee 8
- H**
- half bridge: 1 active and 1 passive strain gauge 355
 - half bridge: 2 active strain gauges 354
 - half bridge: general 353
 - half bridge: Poisson 354
 - half bridge: strain gauge 353
 - half-bridge
 - LVDT 359
 - Hand-Terminal
 - Typ 154
 - Hard drive 310, 337
 - harmonic cursor 1047
 - Header (file header) 755
 - Headline
 - Show message box 1188
 - Heavy burdening of the processor 1137
 - Height
 - Property - Pages 1123
 - Help and Documentation 64
 - Hexadecimal 1117
 - hiding data
 - curve window 924
 - HighLowRatio 597
 - high-pass (curve window) 1022
 - Histogram / Rainflow (Setup page) 270
 - Histogram display update interval 270
 - Histogram storage interval 270
 - history (curve window) 1081
 - Homepage 29
 - Hostname (DHCP) 46
 - Hotline
 - Tech support 7
 - Hot-Plug
 - FTP access 344
 - Storage medium 340
 - Hyst 598
 - Hysteresis 255, 286
 - for trigger events 293
- I**
- I-component 533
 - I-Controller
 - imc Online FAMOS 532
 - iDiv 599
 - IEEE FLoat
 - pv-variable 741
 - If 706
 - IF conditions
 - boolean variables 521
 - IFA 498
 - Image
 - Widget 1109
 - imc CRONOSflex
 - Module address 158
 - imc CRONOSflex Modules 241
 - imc CRONOS-XT
 - Module address 158
 - imc CRONOS-XT Modules 241
 - imc Data format
 - imc2 785
 - imc3 785
 - imc Help and Documentation 64
 - imc Language Selector 34
 - imc Software License Agreement 9
 - imc Systems 342, 343
 - formatting 345
 - imc FAMOS
 - Always overwrite existing files 721
 - Bookmarks 729
 - Breakpoints 729
 - congestion 729

- imc FAMOS
 - Current measurement 724
 - Debug Mode 721
 - From imc FAMOS 724
 - Last concluded measurement 724
 - many data cutting interaction 729
 - Measurement 724
 - Measurement number 724
 - Open 721
 - Options 724
 - parallel data cutting interaction 729
 - parallel processing of sequences 729
 - Synchronous event 721
 - Timeout 721
 - To imc FAMOS 724
 - Transfer table 724
 - imc FAMOS project 1148
 - imc FAMOS sequence 1149
 - imc Format Converter
 - ASCII 788, 1159, 1202
 - Command line parameters 795, 1209
 - Data saving assistant 799, 1173, 1213
 - EXCEL 788, 1159, 1202
 - Export formats 786, 1157, 1200
 - Installation 787, 1201
 - Save in one file 788, 1159, 1202
 - Sequencer Command 797, 1157, 1211
 - Settings 788, 1159, 1202
 - Stand alone (Operation) 793, 1207
 - Storage location 788, 1159, 1202
 - Windows-Explorer 796, 1210
 - imc Format Converter Command
 - ASCII 1159
 - EXCEL 1159
 - Save in one file 1159
 - Sequencer Command 1157
 - Settings 1159
 - Storage location 1159
 - imc Inline FAMOS 498
 - Import source code 1165
 - imc LICENSE Manager 28
 - imc Online FAMOS 498
 - Class-counting 537
 - Comparison 493
 - Control units 445
 - Data flow schematic 493
 - enable 156, 157
 - FAQ 542
 - Filter digital - Limits 541
 - Global RAM 541
 - Local channels 513
 - Local single value variables 513
 - Local variable 513
 - maximaum stack 538
 - Nesting depth 538
 - pv-variable 741
 - Single value variables 513
 - Supplemental Files 537
 - System requirements 497
 - imc Online FAMOS Professional 502
 - Controller 532
 - D-Controller 532
 - enable 156, 157
 - I-Controller 532
 - P-Controller 532
 - PI-Controller 532
 - PID-Controller 532
 - imc Online FAMOS program is not saved 542
 - imc SENSORS
 - Reset channel's sensor information 467
 - supported sensors 470
 - imc STUDIO Monitor
 - enable 156, 157
 - imc WAVE 66
 - imc WAVE Noise 101
 - imc WAVE Rotation 106
 - imc WAVE Spectrum Analyzer 94
 - imc WAVE Structural Analyzer 132
 - imc WAVE Structure 114
 - imc WAVE Vibration 108
- imc2
 - Data format 785
 - imc3
 - Data format 785
 - imcSyslog 312
 - Immediate
 - Trigger event 295
 - immediate trigger 285
 - Immediately 234
 - Import
 - experiment 73, 75
 - experiment template 73
 - project 73, 75
 - Source code 1165
 - import a picture or map as background (curve window) 972
 - Import imc Inline FAMOS source code 1165
 - Import mapping by name
 - Mapping 775
 - Import of channels according to connector
 - Mapping 775
 - Import of channels according to connector and device serial number

- Import of channels according to connector and device serial number
 - Mapping 775
- Import Panel page 1171
- Import parameters 770
- Import supplemental file
 - Mapping table 1184
- Increment
 - Property - Widget 833
- Incremental counter
 - combined measurement 368
 - comparator conditioning 363
 - edge 366
 - max. number of pulses per rev. 365
 - maximum input range 362
 - sampling rate 359
 - scaling 362
 - start edge 366
 - stop edge 366
 - time measurement 366
- Incremental counter (Setup page) 254
- Incremental counter does not go to zero 368
- Incremental counter inputs 239
- index-channel 364
- Individualized colors for this window 1078
- Inequality operator 545
- Information (Logbook) 1214
- Initialization
 - CAN-Bus 394
 - CAN-Bus (ARGUS) 420
 - Sensor 394, 420
- Initializing a sensor 437
- Inline FAMOS 498
 - Apply 507
 - Block comment 517
 - Channel name 517
 - Check - Syntax 507
 - Comment a line 517
 - Context menu 511
 - Digital inputs 514
 - Line comment 517
 - Menu 509
 - Schematic 499
 - Special characters 517
 - Starting 504
 - Syntax check 507
 - Time Base for Channels 517
 - Virtual channels 513
- Inline Function 98
- Input range 248, 249, 253, 255, 479
- Input variables
 - Python 1149
- Insert sample after 1113
- Insert sample before 1113
- Insert Widget 821
- Inside of range 290
- instable filter functions 587
- Installation 16
 - imc DEVICEcore 24
 - imc DEVICES 24
 - imc FAMOS 26
 - imc SENSORS 25
 - imc Shared Components 24
 - imc STUDIO 25
 - Projects 36
 - Security software 20
 - Silent 37
 - Step by step 20
 - Unattended 37
 - Update 16
 - User account control 19
 - Views 36
 - Virus scanner 20
 - Your own personal programs 36
- Installation type
 - Full functionality for 30-day demo 22
 - Typical functionality incl. imc STUDIO Professional 22
 - User-defined 22, 24
- Integral 599
- Integral2 599
- IntegralFFT 599
- IntegralP 600
- IntegralP2 600
- Interface configuration
 - Device Interfaces 47, 49
- Interface-Configuration 47, 51
 - Export current settings 47, 51
- Internal storage media 337
- Internet Settings for maps 982
- Interval
 - Count 318
 - Property - Widget 836
 - Saving 318
- Interval (Timeout)
 - Show message box 1188
- Interval funktions 600
- IntervalFrom1Level 602
- IntervalFromLevels 603
- IntervalFromPulse 604
- IntervalMax 604
- IntervalMean 605
- IntervalMin 605

- IntervalMult 606
 - IntervalResample 607
 - IntervalRMS 607
 - IP address 43
 - configure 40
 - of the devices 40
 - of the PCs 40
 - IRIG Format 1107
 - IRIG-B
 - External clocks 200
 - Synchronization 200
 - ISO / DIS 11898 370
 - ISO 11519-2 370
 - ISO-9001 8
 - isolated synchronization 232
 - Isolated thermo couple 249
 - Isolines
 - 3D 996
 - Issues
 - Storage medium 340, 348
 - IsSynchronized 608
- J**
- J1939 401
 - CAN 387
 - DBC 401
 - Jiggling display 35, 1137
 - JKFlipFlop 608
- K**
- K-factor 357
 - Kurvenfenster
 - freie Texte 987
 - Text 987
 - KWP2000 378, 446
 - KWP2000 TP2.0 378, 451
- L**
- Label for raw data 952
 - Labeling
 - Lines 960
 - LAN 43
 - Device connection 49
 - Language change 34
 - Language installing 35
 - Last measurement 335
 - Last value as number (curve window) 852
 - Display 868
 - Font 868
 - Launching in Fullscreen mode 1129
 - Layout
 - Browse in workspace 1185
 - LED 514
 - LED6 150, 539
 - Overflow 500
 - LED-flashing during measurement 539
 - legend
 - curve window 967
 - legends (line) 961
 - LEQ 609
 - Less 609
 - Less than operator 545
 - Less than or equal operator 545
 - LessEqual 609
 - Level - Trigger-Event type
 - Negative slope 290
 - Positive slope 290
 - Signal < Level 290
 - Signal > Level 290
 - Level Indicator 1103
 - Liability restrictions 8
 - Licensable components 28
 - License 28
 - Limitations
 - Storage medium 340, 348
 - Limited Warranty 8
 - line color
 - depend on magnitude 983
 - Line comment
 - imc Online FAMOS 517
 - Lines (curve window) 954
 - Color 955
 - Cross sections 965
 - Extras 962
 - Labeling 960
 - Line structure 955
 - Line thickness 955
 - Line type 955
 - Lines 955
 - Printer / Screen: 955
 - Symbol 955
 - Symbol size 955
 - Line-Shift 963, 1057, 1061
 - in the ccv 1081
 - Reset 1061
 - link (curve window) 1081
 - Link with several curve windows 1059, 1060
 - Link XY with 2nd curve window 1058
 - linking a map with time data 977
 - Lissajous figures
 - curve window 913
 - List 243

- literature (CAN-Bus) 370
 - Ln 609
 - Load CAN configuration CBA 411
 - Load dialog (curve window) 1064
 - Load measurement settings 816
 - Load variable 811
 - Add file 1196
 - Add variable 1196
 - Category 814, 1196
 - Command 1194
 - Data Browser 811
 - Load all 1197
 - Show dialog 1197
 - Target measurement 814, 1197
 - Target variable name 813, 1196
 - Treat error as warning 814, 1197
 - Variable name 813, 1196
 - Variables list 813, 1196
 - Local channels 513
 - Local single value variables 513
 - Local system information
 - Variables 738
 - Local variable 513
 - Localtime 192
 - Lock page 1127
 - log channel
 - CAN Log channel 386
 - with/without sended channels 385, 394
 - LogAnd 610
 - Logbook 1214
 - Autoscroll 1216
 - Category 1214
 - Copy 1215
 - Copy with details 1215
 - Deleting 1216
 - Duplicates 1215
 - Export 1216
 - Filtering 1215
 - Firmware update 60
 - Location 1214
 - Logbook-Viewer 1216
 - Options for the Logbook 1216
 - Search 1215
 - Zip file 1216
 - Logbook (curve window) 873
 - Logbook category
 - Error 1214
 - Fatal 1214
 - Information 1214
 - Warning 1214
 - Logbook viewer
 - Opening by command 1189
 - LogNot 610
 - LogOr 610
 - LogXor 610
 - Long date 1107
 - Long name
 - Title - Default Source 831
 - Long time 1107
 - Long-term measurement 475
 - Lower 610
 - low-pass (curve window) 1022
 - LVDT 358
- 
- MAC-address 55
 - Magnitude (curve window) 868
 - maintenance 8
 - Making a device known 171
 - Manage Projects 72
 - Managed DLL 1164
 - Map 972
 - cache 982
 - curve window 976
 - from Internet 978, 979
 - link 977
 - Map provider, Adding- 980
 - mode 973
 - provider 979
 - settings 982
 - stretched 981
 - map coordinates adjust 972
 - Map provider, Adding-
 - Map in curve window 980
 - Mapping 775
 - Export all channels 775
 - Export all channels and settings 775
 - Import mapping by name 775
 - Import of channels according to connector 775
 - Import of channels according to connector and device serial number 775
 - Multiple-device operation (Firmware group A) 775
 - Single-device operation (Firmware group A) 775
 - Mapping table
 - Export supplemental file 1183
 - Import supplemental file 1184
 - Marker
 - Voreinstellungen 1028
 - Marker (curve window) 1028, 1039
 - Delete all markers 1045
 - extra functions 1045
 - Markers at Min/Max 1045

- Marker (curve window) 1028, 1039
 - Markers on all lines 1045
 - move 1037
 - new line 1031
 - Select all lines 1045
 - set 1030
- marker precision 1039
- Matrix (curve window) 1005
- Max 271, 611
- max dwell time 292
- max. delay (sound) 1096
- Max. delay [ms] (curve window) 1081
- Maximum stack
 - imc Online FAMOS 538
- Maximum 255
 - Preprocessing 264
 - Range 833
- Maximum amount of variables in imc Online FAMOS 407
- Maximum input range
 - INC-channels 362
- MDF 424
- MDF format 754
- Mean 267, 612
- Measure (curve window)
 - 1/dx (frequency) 1014
 - dx 1014
 - Slope 1014
 - Slope per decade 1014
 - xl 1014
 - xr 1014
 - xr/xl 1014
 - yl 1014
 - yr 1014
 - yr/yl 1014
 - yr-yl 1014
- Measure a user-defined characteristic curve 1175
- Measure context menu
 - Channels List 1016
 - Clipboard 1016
 - Expand List 1016
 - Export Curve Segment 1016
 - Place marker with left click 1016
 - Send Curve Segment... 1016
- Measure mode
 - Curve 1012
- Measure window (curve window) 1012, 1014
 - color maps 1019
 - context menu 1016, 1020
 - cursor positioning 1016
 - Date/ time display 1014
 - Days/ hours/ minutes display 1014
- Measured value 482
- Measurement 187
 - loading 808
 - Start 150
 - Stop 150
 - unloading 808
- Measurement comments
 - about Data Saving Assistant 1173
- Measurement cursor (curve window) 1012
- Measurement data 70
 - Memory requirements 755
- Measurement folder structure 330
- measurement mode 247, 254, 262
 - Encoder 256
 - temperature 349
- Measurement mode (Setup page) 247
- Measurement modes for incremental counter inputs 360
- Measurement number 1133
 - Assigning 809
 - Locking 809
- Measurement options
 - Setup page 236
- Measurement procedure for incremental counter channels 256
- Measurement process states 699
- Measurement running 1238
- Measurement running - reconnect not possible 1238
- Measurement status 187
 - Variables 739
- Measurement storage area 330
- Measurement variable 467
- MEASUREMENT.SQL 1224
- Median3 613
- Median5 613
- Memory card 337
- Memory cards 146
- Memory requirements 755
- Menu
 - Edit (curve window) 1072
 - Extras (curve window) 1088
 - File (curve window) 1063
 - Options (curve window) 1074
- Menu (curve window)
 - Adapt 1090
 - Configuration 1073
 - Toolbar 1090
- Menu and Toolbar (curve window) 1089
- Mercator-projection 981
- Message
 - CAN-Bus 412

- Message
 - multiple with different length 393
 - Synchronization 402
 - Messunsicherheit 965
 - Meta 72
 - Metadata 73, 173, 176
 - Metadata readout 173
 - Metadata save in the channel 173
 - Metafile
 - transferring curve windows 1052
 - microSD Storage medium 337
 - Min 271, 614
 - Minimum
 - Preprocessing 264
 - Range 833
 - Minimum accuracy [ms] 1240
 - Min-Max
 - Preprocessing 264
 - Mode 247, 252, 254, 262
 - Modify values (curve window) 1057
 - Module 241
 - Address 158
 - exchange 158
 - Properties 158
 - removing 158
 - Module address
 - Assign manually 158
 - Reassign 158
 - Module numbering 158, 241
 - Modulus 544
 - Modulus of elasticity E 253
 - Monitor 1129
 - Analog inputs 238
 - Digital inputs / outputs (bits and ports) 239
 - Incremental counter inputs 239
 - Monitor channel
 - anti-aliasing filter 258
 - CAN 392
 - ECU 392
 - Monitor channels 238, 302, 735
 - Monoflop 615
 - MonoflopRT 615
 - More Channels in Curve Window 924
 - Adaption channel selection 933
 - Available Channels 929
 - Channel info 934
 - Waveform Preselection 930
 - Waveforms in axes list 931
 - mouse wheel
 - curve window 1100
 - move area in curve window 1100
 - Move markers
 - curve window 1037
 - Multi-Event
 - Curve window 1137
 - Multiline comment
 - imc Online FAMOS 517
 - Multi-lingual text input 1138
 - Multilingual title
 - Title 831
 - Multimaster Synchronization 219
 - Multimeter (curve window) 868
 - Multiple triggering 298
 - Memory requirements 757
 - Multiple-device operation (Firmware group A)
 - Mapping 775
 - Multiplication 543
- N**
- Name 186, 242, 467
 - Property 527
 - Property - Pages 1123
 - Property - Widget 831
 - Show message box 1188
 - Title - Default Source 831
 - Names (curve window) 868
 - Navigation bar (curve window)
 - Axes 1101
 - Navigation in x-direction (curve window) 1085
 - Necessary system component
 - Installation 24
 - Negative slope 290
 - Nesting depth
 - imc Online FAMOS 538
 - Net bits 239
 - Network
 - Firewall 30
 - Optimize 56
 - Network drive 310, 314
 - Network drive storage settings 314
 - network name (device hostname) 46
 - New
 - experiment 73
 - experiment from template 73
 - New device configuration
 - Interface configuration 49
 - New experiment 72, 73
 - from template 73
 - NMEA 745
 - NMEA Talker IDs
 - GA, GB, GI, GL 748
 - GN, GP, GQ 748

- No new devices found 151, 1238
 - Nodes
 - CAN-Bus 412
 - Noise 101
 - Trigger 293
 - noise (curve window) 1022
 - Normal data set 730
 - NorthCorrection 616
 - NOT 617
 - NTP
 - External clocks 206
 - Synchronization 206
 - Number (curve window)
 - Display 868
 - Number of folders (max.)
 - Storage medium 477
 - Number of symbols (0 = every sample) 852
 - Number of trigger releases 298
 - Number trimming (useful when scrolling) 852
 - NumberOfPulses 617
 - Numeric 243
 - Numeric Zone
 - Zone 847
 - Numerical zones 1115
 - Nyquist frequency 258
- O**
- OBD-2 452
 - Assistant 453
 - PIDs 453
 - user defined PID 453
 - Octal 1117
 - Octave (1/3) labeling 91
 - octave labeling
 - display options 906
 - OFA 498
 - Offline cache (map) 982
 - Offset 250, 275, 1113
 - Offset (curve window) 1022
 - OK Button
 - Show message box 1188
 - Text (Button caption) 1188
 - Visible 1188
 - Visible (Hide) 1188
 - Visible (Show) 1188
 - OLE 1054
 - OnAlways 701
 - OnCanMessageReceive1_S1 716
 - OnECUCmdReturn 579
 - OnECUCmdReturn_ECU_001 619
 - OnInitAll 521, 700
 - Online classification
 - enable 156, 157
 - Online FAMOS 498
 - Apply 507
 - Block comment 517
 - Channel name 517
 - Check - Syntax 507
 - Comment a line 517
 - Context menu 511
 - Digital inputs 514
 - Line comment 517
 - Menu 509
 - Schematic 499
 - Special characters 517
 - Starting 504
 - Syntax check 507
 - Time Base for Channels 517
 - Virtual channels 513
 - Online-Order tracking
 - enable 156, 157
 - Online-Trigger 302
 - OnMeasureEnd 702
 - OnMeasureStart 701
 - OnPowerOff 705
 - OnSyncTask 703
 - OnTimer 705
 - OnTriggerEnd 302, 521, 704
 - OnTriggerMeasure 302, 704
 - OnTriggerStart 302, 521, 703
 - Open 72
 - imc FAMOS 721
 - Open for different device 179
 - Open source software licenses 33
 - Opening Closing Bracket 546
 - Operable
 - Property - Widget 832
 - Optional components 28
 - Options 1237
 - 3D 989
 - colors 1078
 - Export current Data 335
 - Menu (curve window) 1074
 - Presettings (curve window) 1081
 - Save current Data 335
 - OR 619
 - Trigger event 295
 - Orientation
 - Property - Widget 832
 - Oscilloscope mode
 - Curve window 999
 - OtrAngleAdd 620

- OtrEncoderPulsesToRpm 620
- OtrFrequLine 623
- OtrFrequLine2 624
- OtrFrequLine3 624
- OtrOrderSpectrum 625
- OtrOrderSpectrumP 627
- OtrPulseDuration 629
- OtrResample 630
- OtrResampleAngle 632
- OtrResampleFromRpm 633
- OtrRpmComplexOrder 634
- OtrRpmOrder 637
- OtrRpmPresentation 639
- OtrRpmPresentVector 640
- OtrRpmSpectrum 641
- OtrSynthSin 641
- OtrTrackingLowPass 642
- Output variables
 - Python 1149
- Outputting a waveform to a DAC module 436
- Outside of range 290
- Overflow
 - Beeper 500
 - imc Inline FAMOS 500
 - imc Online FAMOS 500
 - LED6 500
- overload 116
- overmodulation 116
- Overview window 1070
- Overwrite autostart configurations 1238
- Overwrite existing file 1238
- Overwrite this multilingual text? 1138
- P**
- P3 time out 446
- PAGE 1227
 - Browse in workspace 1185
 - Property - Pages 1123
- Page setup 1120
- Page size
 - Property - Pages 1123
- PAGE.NUMBER
 - Formatting 1233
- PANEL 1227
- Panel page
 - Adapt Page Size 1125
 - Deleting by command 1171
 - Dialog 1168
 - Export 1170
 - Zoom 1128
- Panel page as dialog 1168
- PANEL.PAGECOUNT
 - Formatting 1233
- Paper size 1120
- Parameter set 759
 - Mapping 775
- Parameterizing bridges 479
- Parameters
 - Browse in workspace 1185
- partition 339, 345
- Passive
 - Trigger event 295
- Password
 - Data transfer 338
- Password locking
 - Firmware update 60
- Path to Python file 1149
- PC time 1107
- P-component 533
- P-Controller
 - imc Online FAMOS 532
- PDF orientation (curve window) 1085
- Peakhold (curve window) 868
- Peaks 474
- Peaks (curve window) 1020
- Percentage
 - Zone 845
- Period comparison (curve window) 852, 1000, 1005
- Periodic (curve window) 1005
- Perspective
 - 3D 995
- Phase offset correction
 - Synchronization 220
- Physical unit 467
- PI-Controller
 - imc Online FAMOS 532
- Pictogram
 - Property - Widget 832
- PIDcontroller 716
- PID-Controller
 - imc Online FAMOS 532
- Pin configuration
 - CAN 371
- Place (map) 982
- Placeholder
 - CONTROL 1219
 - DATABASE/DATABASES 1220
 - EXPERIMENT/EXPERIMENTS 1221
 - PAGE 1227
 - PANEL 1227
 - PROJECT/PROJECTS 1227
 - SELCONTROL 1229

- Placeholder
 - SETUP 1230
 - SYSTEM 1231
 - VARS 1232
- Placeholder Formatting
 - PAGE.NUMBER 1233
 - PANEL.PAGECOUNT 1233
 - SYSTEM.DATE 1234
 - SYSTEM.TIME 1234
 - VALUE 1235
- Placeholders 1217
- Placeholders for folder naming
 - SETUP.SQL 331
 - STORAGE.FOLDERNAME 331
 - STORAGE.MEASUREMENT 331
 - VARS.VALUE 331
- Places left of point (curve window) 868
- Places right of point (curve window) 868
- Platzhalter
 - MEASUREMENT 1224
- Playback 83
- Plug-in
 - Info 33
 - Version information 33
- Points 266
- Poisson full bridge 356
- Poisson half bridge 354
- Poisson's ratio 357
- polar plot 900
- Polarization 249
- Poll 643
- Polygon needle
 - Property - Widget 832
- Ports 44
- Position
 - Property - Widget 832
 - Title 831
- Positive slope 290
- Potential difference with synchronized devices 232
- Power 543
- Power fail 154
- Power measurements 530
- Power1 530, 643
- Power2 530, 644
- Power3 531, 645
- PPT 1054
- Prepare 149
- Preprocessing 264
- Presettings
 - Curve Window 1008
- Presettings (curve window) 852
 - options 1081
- Pretrigger 286
- Pretrigger 300
- Preview in Windows-Explorer 1105
- Print 1120
 - Curve Window 1067
- Print Panel page 1170
- Print preview
 - Widget 1132
- Printer
 - Setup (curve window) 1069
- Printer preview 1088
- printer settings 1074
- Problems
 - Storage medium 340, 348
- Process configuration 147
- Process vector
 - backup file 744
 - pv-variable 741
- Process vector backup file 744
- Process vector variable 741
 - Create 519
- process vector variables 239
 - GPS 745
- Process vector variables for data acquisition channels 236
- Processing function 266
- Processing points 266
- Product
 - Configuration 28
 - Edition 28
 - Info 33
 - Licensing 28
 - Version information 33
- Product configuration 28
 - Changing 28
- Product improvement 7
- Program
 - execute 1156
- Program add
 - Installation 36
- Project 70, 73
 - delete 73
 - export 73
 - import 73
- Project has changed 1238
- Project option 1237
- PROJECT/PROJECTS 1227
- Prompt to confirm discard 1174
- Properties
 - Device 153

Properties

- log file 153

Properties of moldules 158

Properties of virtual channels

- Comment 527

- Delta-x 527

- Delta-z 527

- Name 527

- x-offset 527

- x-unit 527

- y-unit 527

- z-offset 527

- z-unit 527

PROPS

- Metadata readout 173

Protocol display (curve window) 873

Protokollkanal 424

- CAN Protokollkanal 422

Pt100 350

PTP

- External clocks 210

- PTP-Master only 217

- Synchronization 210

Pulse spikes in the signal 474

pulse time 367

PulseDuration 645

PulseFrequency 647

PulsePhase 648

pulses number max. 365

Push-button

- Switching behavior 833

pv.State.ExternalPower 741

pv.State.SyncTimeDeviation 741

pv-variable 741

- As monitor channels 745

- Create 519

- Deactivating 741

- Name 741

- Process vector 741

- restore 744

- Restoring 744

- Update rate 744

pv-variables 239

PWM mode (INC4) 367

Python 1149

Q

Quality Management 8

quarter bridge

- 120 Ohm WSG 353

- WSG 353

R

Radius

- Scale position 836

RAM buffer time 322

Range 271

- Property - Widget 833

Range - Trigger-Event type

- Entering range 290

- Exiting range 290

- Inside of range 290

- Outside of range 290

Range & Scaling (Setup page) 249

RangeMax 650

RangeMin 650

raw 753

Read sensor information 242, 466

Ready for measurement

- Interface configuration 49

ReadyForPowerOff 651

Realign to Grid 1127

RealTimeClock 192

recalculate derived channels 85, 86

Receiver

- GPS 745

Reconfigure 149

RecordEvent 652

RecordText 653

Red 654

ReduceDataRate 655

Reduction

- Preprocessing 264

Refill variable 811

- Add file 1196

- Add variable 1196

- Command 1194

- Data Browser 811

- Import all 1197

- Show dialog 1197

- Target measurement 814, 1197

- Target variable name 813, 1196

- Treat error as warning 814, 1197

- Variable name 813, 1196

- Variables list 813, 1196

Refresh rate

- Property - Widget 831

Released 295

Remote Frames (RTR) 370

Removable data carrier 310

repair 8

ReplaceFirstValues0 655

- ReplaceFirstValuesN 656
- Report
 - Setup 181
- Report Generator 1067
- Report template 1132
 - Property - Pages 1124
- Report-page 1120
- Reprocessing signals (curve window) 1020
- ReSample 656
- Reset
 - Set variables 1198
 - Slave Pointer 879
- Reset channel's sensor information 467
- Reset Slave Pointer... 1057
- Resistor 247, 252
- Restart measurement automatically 231
- restore 657
 - pv-variable 744
- Restoring pv-variables 744
- Resulting sampling rate 266
- Resulting sampling time 266
- Rezoom (curve window) 1085
- RGB image
 - Aspect ratio 946
 - Resolution 946
- RGB-image 923
- Ring (circular) buffer storage 309, 318
- Ring buffer memory 475
- RMS 528, 657
 - Preprocessing 264
- Rosette1 658
- Rosette2 660
- Rotate
 - 3D 998
- Rotation 106
- Round 660
- Router 43, 52
- RPM 368
- RS232 settings
 - GPS 747
- RSFlipflop 661
- RTC 192
- Run Python code file 1149
- RunAutoBalance 662
- RunAutoShuntCalibration 662
- RVDT 358
- S**
- Sample count 261
- Sample delay 474
- SamplesGate 663
- sampling
 - data recording 472
- Sampling & Preprocessing (Setup page) 261
- sampling rate 261
 - Incremental counter 359
- sampling theorem 258
- Sampling time 261
- Save as 72
 - experiment 73
- Save CAN configuration CBA 411
- Save current data 334
 - Options 335
- Save data 268
- Save dialog (curve window) 1066
- Save measurement data
 - Saving subsequent to measurement 334
 - Targeted saving 334
- Save trigger events in individual files 329
- Saving 306
 - Cluster size 756
 - Data types 756
 - to the internal drive 310
- Saving interval 318, 321
- Saving pv-variables 744
- Saving subsequent to measurement 334
- Sawtooth 663
- Scale angle
 - Property - Widget 836
- Scale center
 - Range 833
- Scale decimal places
 - Property - Widget 836
- Scale position
 - Property - Widget 836
- Scaled
 - Point 249
- scaling 479
 - Incremental counter 362
 - lines 962
 - x-axis 936
- Scaling factor 250, 254
- scaling for strain analysis 357
- Scaling offset 250, 255
- Scaling with offset 480
- scaling: strain gauges 357
- schematic
 - imc thermo plug 351
 - T4 351
- Script name
 - Property - Pages 1123

- Scroll mode (curve window) 999
 - Curve window 999
 - End online mode 999
- Scroll-mode 475
- Search CAN channel, message or node 411
- Search for devices 151
 - No new devices found 151
- Search for devices by IP/DNS 52, 152
- Security alert 30
- Security software 20, 35
- Seed/Key 446
- Seed-and-Key 443
- Seed-Value 446
- Segment (curve window) 1000
- Segmented data set 731
- SELCONTROL 1229
- Select device with explorer 338, 342
- Select mode (curve window) 1026
- Selected 171, 186
- Selected x-unit (curve window) 852
- Semicolon 546
- Send channels with imc FAMOS 399
- Send channels with process vector variables 399
- Send Curve Segment to FAMOS!:context menu 1016
- Sendable message
 - CAN-Bus 412
- sended channel
 - not sended by log channel 385, 394
- SendMessage 663
- Sensitivity 250
- Sensor 242, 465
- Sensor delay time (CAN message) 402
- Sensor information
 - TEDS 466
 - writing to channel 466
- Sensor information - reset 467
- Sensor initialization 394, 420
- Sensor-PROM 465
- Sensors
 - Reset channel's sensor information 467
- Sensor-TEDS 465
- Sequence table 1141
- Sequencer 1140
 - Start 1144
 - Stop 1144
- Sequencer stop
 - about commands 1165
- Sequencer: Context menu 1147
- Service
 - Tech support 7
- service and maintenance 8
- service check 8
- Set as default
 - Curve Window 1008
- Set markers (curve window) 1030
- Set measurement number 1166
- Set value
 - Zone 1115
- Set variables
 - Delete 1198
 - Reset 1198
- SetPoint 533
- Settings (curve window) 1081
- Setup
 - Printer (curve window) 1069
- Setup data in Panel 1230
- Setup data in Sequencer 1230
- SETUP.SQL 1230
- Shell extension 342
- Short date 1107
- Short time 1107
- Show Grid 1127
 - Property - Pages 1124
- Show message box
 - Cancel 1188
 - Headline 1188
 - Interval (Timeout) 1188
 - Name 1188
 - OK 1188
 - Size 1188
 - Text 1188
 - Timeout-action 1188
 - Voice 1188
- Show Panel page as dialog 1168
- Show Tabs 1129
- Show x=0 (trigger) 852
- Shunt calibration 280
- Signal 254
- Signal < Level 290
- Signal = 0 291
- Signal = 1 291
- Signal > Level 290
- Signal change: 0 -> 1 291
- Signal change: 1 -> 0 291
- Signal shape 255
- Silent installation 37
- Simultaneous start of all devices 235
- Sin 664
- single value (curve window)
 - window 872
- Single value variables 513

- Single values 730
- Single-device operation (Firmware group A)
 - Mapping 775
- Single-phase power measurement 530
- single-track encoder 364
- SingleValueChannel 664
- Size
 - Show message box 1188
- Size automatic (curve window) 868
- Size of measurement point (curve window) 1085
- skb files 446
- Skin
 - Property - Pages 1123
- SkipFirstValues 667
- Slave Pointer
 - Reset 879
- Slider 1114
- SlopeClip 667
- Small Ticks (curve window) 936
- Smo3 667
- Smo5 668
- smoothing (curve window) 1022
- SN 186
- Snap to Grid 1127
 - Property - Pages 1124
- so files 446
- Software
 - Installation 16
 - Uninstall 16
 - Update 16
- Solid
 - Background color 1123
- Sorting
 - Setup table 165
- sound output (curve window) 1081, 1092
 - cut sound 1095
 - directly 1096
 - Toolbar 1093
 - volume 1093
- Sound playback 84
- SoundPressureLevel 668
- Source code
 - Import 1165
- space curve 990
- Special characters
 - imc Online FAMOS 517
- Specification 186
- SpecThirds 668
- Spectrum Analyzer 94
- Spectrum-analyser
 - tutorial 117
- Speed 368
- Spikes 474
- Spinner 1114
- Spline (curve window) 1022
- Split up pretrigger data into intervals 1241
- Spread angle
 - Scale angle 836
- Sqrt 670
- SSD Storage medium 337, 341
- Standard display (curve window) 852
- Standard Format
 - CAN 383
- Standard time 192
- Start 150
 - Measurement 150
 - Simultaneous 235
- Start angle
 - Scale angle 836
- Start button 285
- Start by Trigger 295
- start delay (sound) 1096
- Start delay [ms] (curve window) 1081
- Start edge 255
- start edge (incremental counter) 366
- Start Measurement
 - Automatically upon starting imc STUDIO 31
- Start option 234
 - At defined time 234
 - Automatic timed start 234
 - Immediately 234
- Start page 29
- Start Sequencer
 - Automatically upon starting imc STUDIO 31
- Start with shortcut 31
- Start/Stop behavior 482
- Start: Sequencer 1144
- Starting
 - imc Online FAMOS 504
 - imc WAVE 28
- StartTimerPeriodic 710
- StartTimerSingle 711
- Startup parameters 31
- State of the external power 741
- Static text 243
- Status 242, 1141
 - Property - Widget 831
- STD 192
- StDev 670
- Stop 150
 - Measurement 150

- Stop by Trigger 295
- Stop edge 255
- stop edge (incremental counter) 366
- Stop on error 1141, 1144
- Stop Sequencer
 - Command 1165
- Stop: Sequencer 1144
- StopTimer 711
- Storage
 - Memory requirements 755
 - Setup page 187
- storage device
 - formatting 339, 345
 - partition 339, 345
- Storage interval 318
 - Histogram 270
- Storage interval count 318
- Storage location (PC) 316
- Storage location measurement data 330, 1238
- Storage media 337
- Storage medium
 - Age 477
 - CF card 341
 - CFast 341
 - Cluster size 338
 - Compact Flash 341
 - Data transfer 338, 342
 - ExpressCard 341
 - FAT16/FAT32 345
 - FAT32 338
 - File system 338, 345
 - Filesize (maximum) 338, 345
 - Hot-Plug 340
 - Issues 340, 348
 - Limitations 340, 348
 - Manufacturer 477
 - microSD 337
 - Number of folders (max.) 477
 - Problems 340, 348
 - SSD 337, 341
 - USB 341
- Storage on a network drive 314
- Storage options 308
- strain gauge 351
- Strain Gauge (Setup page) 252
- strain gauge: scaling 357
- strain gauges 352
- Stretch
 - Background image 1123
- Stretch mode
 - Curve window 999
- STri 671
- Structural Analyzer 132
- Structure 114
- Subnet 45
- Subnet mask 43
- Subtraction 543
- Sum 672
- Sum sampling rate 267
- Sum2 673
- Summation
 - Calculation uncertainty 540
- Supplemental file
 - Delete 1185
 - Export 1183
 - Import 1184
- Supply 248
- Surface
 - 3D 990
- Swapping the storage medium 337, 340
- Switch 707, 1114
 - Switching behavior 833
- Switching behavior
 - Property - Widget 833
- Switching level 255
- Switching off the device 182
- Switching on the device 182
- Symbols
 - Fixed count at line 962
 - for lines in curve window 955
- SYNC 200
- synchron (sound) 1096
- synchronity re-establish - automatical 231
- Synchronization 188, 190, 192, 474
 - CAN-1 Protocol 385
 - CAN-Bus 385
 - Constraints 233
 - Control 229
 - external timer 200
 - Fieldbus channels 368
 - Frequency error 220
 - GPS 204
 - GPS + DCF Master/Slave 205
 - imc CANSAS 385
 - IRIG-B 200
 - NTP 206
 - Phase offset correction 220
 - Precision 232
 - PTP 210, 217
 - Synchronization varieties 199
- Synchronization signal PTP (Master-Only) 217
- Synchronization status

- Synchronization status
 - Variables 739
 - synchronize devices FAQ 225
 - Synchronized recording 402
 - Synchronous event
 - Python 1149
 - Synchronous start 192
 - Simultaneous 235
 - Synchronous Tasks 503
 - SyncOverload 675
 - SyncRTC 192
 - Internal clocks 194
 - Syntax-Check 507
 - Syslog 312
 - SYSTEM 1231
 - System information
 - Variables 738
 - System requirements
 - Hard drive 14
 - imc Online FAMOS 497
 - imc Online FAMOS Professional 497
 - Operating System 14
 - Storage 14
 - Windows 14
 - System time 1107
 - SYSTEM.DATE
 - Formatting 1234
 - SYSTEM.TIME
 - Formatting 1234
 - SystemClock 1107
- T**
- Table
 - Direction - Prevent a change 1113
 - Widget 1112
 - Table (curve window) 852
 - Table display (curve window) 873
 - Tables
 - Title 1114
 - Tables (curve window) 873
 - Tan 676
 - Target measurement 814, 1197
 - Targeted saving 334
 - Targets
 - Trigger 295
 - Taring 275
 - TCP/IP 49
 - TC-TEDS 465
 - Tech support 7
 - TEDS 465
 - Read sensor information 466
 - Reset channel's sensor information 467
 - supported sensors 470
 - Telephone numbers
 - Tech support 7
 - temperatur characteristic curve
 - How to select? 349
 - temperature measurement 349
 - temperature table 349
 - Temperature unit °F instead of °C 236
 - Terminator
 - CAN 372
 - Terminator in device
 - CAN 384
 - Terminator in device (ARGUS)
 - CAN 420
 - Text 1114
 - Axes 947
 - Show message box 1188
 - Zone 1115
 - Text array 730
 - Text as set value
 - Zone 847, 1115
 - Text display (curve window) 873
 - Text format
 - Property - Widget 833
 - Text input for report channel 1106
 - Text Off/On
 - Property - Widget 832
 - TextAdd 676
 - TextFormatE 677
 - TextFormatF 678
 - TextFormatH 679
 - TextFormatI 680
 - Textual Zone
 - Zone 847
 - Textual zones 1115
 - Text-variables 730
 - The device has been shut down inaccurate in the past 149
 - The measurement process
 - Trigger 301
 - thermo plug 350
 - schematic 351
 - thermocouples 349
 - thermocouples color-coding 349
 - Third/ octave labeling (curve window) 943
 - third-octave axes
 - display options 906
 - Three-phase power measurement 531
 - Tick distance
 - Property - Widget 836

- Ticks (curve window)
 - small 936
 - TI-float
 - pv-variable 741
 - Tile
 - Background image 1123
 - Time Base
 - imc Online FAMOS 517
 - Time Base for Channels in a Formula 517
 - Time counter
 - GPS 745
 - Time measurement 361, 366
 - Time offset 234
 - Time resolution between fieldbus and analog channels 368
 - time stamp
 - data recording 472
 - Time Stamp ASCII (curve window) 873, 875
 - Time to suspend 390
 - Time zone 30, 192, 1107
 - Time zones 220
 - Time-correct display 225
 - Timed start 192
 - Setup page 233
 - Start option 234
 - Synchronous start 235
 - Timeout-action
 - Show message box 1188
 - Timer functions 710
 - Time-Shift 963, 1061
 - in the ccv 1081
 - TimeStamp-ASCII (TSA) 730
 - time-stamped numeric channels 732
 - time-stamped text channels 732
 - Title
 - Property - Pages 1123
 - Property - Widget 831
 - Title column 1114
 - Title in tables 1114
 - Tolerance 680
 - Toolbar
 - Sound output (curve window) 1092, 1093
 - Toolbar (curve window) 1089
 - touchscreen operation
 - in curve window 953
 - Traceability of data 173, 1239
 - Traceability of measurements 176
 - Transfer experiment to other devices 179
 - Transfer Options
 - Transfer to imc FAMOS 1087
 - Transfer rate
 - CAN 373
 - Transfer to imc FAMOS 1085
 - from curve window 1067
 - Transfer Options 1087
 - Transfer to PC 308
 - Transferring curves
 - using clipboard 1052
 - TransRec 680
 - Transverse strain coeff. v 253
 - Treat error as warning
 - Load variable 814
 - Refill variable 814
 - Trigger 282, 286
 - Event-controlled digital outputs 304
 - Hysteresis 293
 - Multiple triggering 298
 - Multiple-triggered signals 1137
 - Noise suppression 293
 - Variables 303, 738
 - Virtual channels from imc Online FAMOS 302
 - Trigger events 287
 - in individual files 329
 - Trigger machine 285
 - Trigger time 1107
 - Triggering behavior 482
 - Troubleshooting 182
 - TSA 730, 732
 - extended ID 396
 - TSAGate 685
 - TSA-Kit 875
 - tutorial
 - Spectrum-analyser 117
 - Twin Window (curve window) 1071
 - Two point calibration 281
 - Two-phase power measurement (ARON) 530
 - Two-Point controller 716
 - Two-position controller 536
 - Type (filter) 258
 - Types of Variables 525
 - Typical functionality incl. imc STUDIO Professional 22
- U**
- UAC 19
 - UDS
 - Dynamic lists 458
 - Unable to find any new devices 151
 - Unable to find device 44
 - Unattended installation 37
 - uncertainty (display) 962
 - Unchanged
 - Background image 1123

- Undo (curve window) 1020, 1081
 - UnEqual 686
 - Uninstall 16
 - Unit 250, 253, 255
 - Property - Widget 831
 - Unlock page 1127
 - Untriggered Measurements 285
 - Unzoom (curve window) 1011
 - Update
 - Database 18
 - Update rate
 - pv-variable 744
 - Updating
 - Curve window 1137
 - Upper 686
 - UPS
 - Acquisition time 154
 - USB 337
 - supply of storage medium 341
 - USB storage medium 341
 - Use DHCP 46
 - User account control 19
 - User Ticks 949
 - User-defined
 - Installation type 22, 24
 - Title - Default Source 831
 - User-defined characteristic curve
 - creating 467
 - User-defined device 52
 - userdefined pages 93
 - User-defined variables 749
 - UTC 192
 - UTC (IRIG Format) 1107
- V**
- VAL_BLK 458
 - VALUE
 - Formatting 1235
 - Value representation
 - Property - Widget 832
 - Variable
 - loading 808
 - Property - Widget 831
 - unloading 808
 - Variable linkage 1133
 - Variables
 - Types 525
 - VARs 1232
 - VectorChannel 688
 - VectorChannelSet 689
 - VectorFromFile 689
 - VectorizeAndSkip 689
 - VectorizeOverlapped 690
 - Vectors 730
 - VectorStatic 690
 - Version information 33
 - Version of the firmware 57
 - Vibration 108
 - VibrationFilter 692
 - Virtual bits
 - Trigger target 295
 - Virtual channels 238, 513
 - Properties 527
 - Trigger 537
 - trigger machine 302
 - Virtual channels and the trigger machine 537
 - virtual clock 195
 - Virtual device clock 195
 - Minimum accuracy [ms] 1240
 - Virus scanner 20, 35, 1137
 - VIsAnyGreater 694
 - Visible
 - Property - Widget 832
 - Visible in printout
 - Property - Widget 831
 - VMax 695
 - VMaxV 695
 - VMean 695
 - VMeanV 695
 - VMin 696
 - VMinV 696
 - Voice
 - Show message box 1188
 - volume sound output 1093
 - VRedV 696
 - VRMS 696
 - VRTC 195
 - VSum 696
 - VValueAtXValue 697
 - VXValueOfMax 697
 - VXValueOfMin 697
 - VXValueWithYValue 697
- W**
- Wait
 - Command 1165
 - Wake on CAN 390
 - Warning (Logbook) 1214
 - Warranty 8
 - Waterfall diagram (3D) 852
 - display 862

- Waterfalls 852
 - WAVE_OverloadVariable 116
 - Waveform Preselection
 - More Channels in Curve Window 930
 - Waveforms in axes list
 - More Channels in Curve Window 931
 - While 709
 - Widget
 - DIO 1117
 - Table 1112
 - Widget linkage 1133
 - Widgets 1106
 - Aligning 827
 - Arrangement 826, 827
 - Cascade 828
 - Dock 829
 - Grouping 830
 - insert 821
 - new 821
 - Realign to Grid 1127
 - Snap to Grid 1127
 - Width
 - Property - Pages 1123
 - Window size
 - Panel 1125
 - Windows
 - Firewall 30
 - Security Alert 30
 - Shell extension 342
 - User account control 19
 - Windows-Explorer
 - Preview 1105
 - WindRoseCorr 698
 - Wiring 248
 - WLAN
 - Access Point 54
 - Accesspoint (Server) 53
 - Ad-Hoc 53
 - Integrated WLAN 54
 - Managed (Client) 53
 - WPA2-PSK 53
 - WORD 1054
 - Write protected
 - Property - Widget 831
- X**
- X-axis 263, 417, 432
 - automatic 936
 - changes 1085
 - fixed range 936
 - lin/log 936
 - markings 936
 - rounding 936
 - scaling 936
 - Scaling the Axes 935
 - XCP 457
 - X-link with curve windows 1058
 - x-offset 527
 - x-unit 527
 - XY data
 - complex data 913
 - different sampling rates 913
 - Lissajous figures 913
 - measurement cursors 913
 - polar plots 913
 - XY-data sets 730, 732
 - XY-plot 91
- Y**
- Y-adaptor
 - CAN-Bus 371
 - Y-axes stacked 852
 - Y-axis
 - color 949
 - line thickness 949
 - small ticks 949
 - y-axis max 271
 - y-axis min 271
 - y-axis range 271
 - Y-connection 531
 - y-unit 527
- Z**
- Z-axis 862, 900
 - zero pulse 364
 - Zip - Logbook 1216
 - ZIP format
 - Limitations 753
 - z-offset 527
 - Zone
 - Advanced display 844
 - Zone dialog 837
 - Zone representation
 - Property - Widget 833
 - Zone ring
 - Zone 848
 - Zones
 - Property - Widget 833
 - Zoom
 - Background image 1123
 - Zoom (curve window) 1010, 1085
 - Zooming Panel pages 1128

z-unit 527



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/>

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://twitter.com/imc_de

<https://www.linkedin.com/company/imc-test-&-measurement-gmbh>