ZES ZIMMER

Instrument Family LMG600

1 to 7 phase precision power analyzer

User Manual

Konformitätserklärung

für die

Mehrkanal-Leistungsmessgeräte-Familie

Declaration of Conformity

for the

Multi-Channel Power Analyzer Family

LMG600

Hiermit wird bestätigt, dass das oben aufgeführte Gerät den Anforderungen der Richtlinien 2014/30/EU, 2014/35/EU, 2012/19/EU und 2011/65/EU + 2015/863/EU der Europäischen Union entspricht.

Diese Erklärung gilt für alle Geräte, die nach anhängenden Fertigungsunterlagen - die Bestandteil dieser Erklärung sind - hergestellt werden.

Zur Beurteilung wurden folgende Normen herangezogen:

We certify that the above device accomplishes with all requirements which are defined in the directives 2014/30/EU, 2014/35/EU, 2012/19/EU and 2011/65/EU + 2015/863/EU of the European Union.

This certificate is valid for all devices that are produced according to the appending production instructions (which are a part of this certificate).

For the judgment of compatibility of the product the following standards were used:

EN IEC61000-3-2:2019+A1:2021

EN61000-3-3:2013+A1:2019

EN55011:2016+A1:2017+A11:2020+A2:2021

EN IEC61326-1:2021

EN61010-1:2010+A1:2019, EN IEC61010-2-030:2021+A11:2021

EN50581:2012

Diese Erklärung wird vom Hersteller
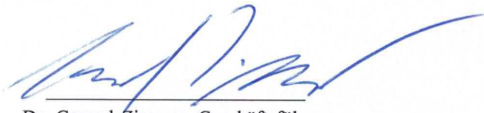
This certificate of the manufacturer

ZES ZIMMER Electronic Systems GmbH
Pfeiffstraße 12
D-61440 Oberursel

abgegeben durch

is given by

Dr. Conrad Zimmer, Geschäftsführer

Oberursel, 16. August 2021

_____
Dr. Conrad Zimmer, Geschäftsführer

Manufacturer Declaration

ZES ZIMMER Electronic Systems GmbH hereby assures that the device to which this declaration belongs complies with all specifications listed in the associated manual and left the factory in mechanically and electrically safe condition.

ZES ZIMMER

Pfeiffstraße 12
61440 Oberursel
Germany
www.zes.com

Contents

List of Figures

List of Tables

1 Legal Notes

```
LEGAL NOTES

This product incorporates the following software components:
(1) Software developed independently by ZES ZIMMER Electronic Systems
    GmbH (ZES ZIMMER)
(2) Software owned by third parties and licensed to ZES ZIMMER
(3) Software licensed as open source software (OSS)

The respective OSS licenses are attached as described below. Anyone can
acquire a legal right of use to the extent regulated therein directly
from the rightholders. The open source software licenses take
precedence over any other license information with respect to the OSS
components contained in the product.

The applicable licenses, copyright information and source codes are
available in a comprehensive ''legal'' folder provided as follows:

 * On an LMG instrument, you can copy the legal folder onto an attached
   USB storage device. Go to Misc menu and select the ''Legal Notes''
   tab, then press/touch the ''Copy Sources to USB'' softkey and follow
   the instructions.

 * Within the authoritative firmware image (the large file with .img
   extension) contained in the ZIP file downloaded from www.zes.com.
   This image can be accessed on Linux using built-in facilities or
   cross-platform using the third-party 7-Zip utility available from
   www.7-zip.org.

The ''legal'' folder contains a copy of these Legal Notes, another folder
with common OSS licenses and a folder with licenses, copyrights and
source codes for the OSS components used. The information for each OSS
component is contained in a corresponding tar archive file named after
the component. The bulk of the OSS components is provided as Debian
source packages. The licenses and copyright information for these
packages is found in the contained source archive with the
.debian.tar.xz extension, in the debian/copyright file. (For references
to the /usr/share/common-licenses folder please use the
legal/common-licenses folder.)

You can also get the source codes of these components from ZES ZIMMER on
a physical medium, if, within 3 years from delivery or at least as long
as we support the product, you make a request to our support department:

ZES ZIMMER Electronic Systems GmbH
Pfeiffstrasse 12, 61440 Oberursel, Germany

Order number: L6-SRC [Product name]

and pay EUR 20,- to cover the expenses for copying and shipping of the
physical medium. The complete documentation of the OSS, the licenses
and the copyrights can be found in the source code of the OSS.

For software components originating from ZES ZIMMER, we allow the
modification for your own use as well as reverse engineering to fix
errors of such modifications, provided that these software components
are linked with libraries licensed under the GNU Lesser General Public
License (LGPL). Proliferation of the information obtained through
reverse engineering or of the modified software is however not
permitted.

On request of the authors and rightholders of the OSS components used we
inform you about the following:

''THE OPEN SOURCE SOFTWARE IN THIS PRODUCT IS DISTRIBUTED IN THE HOPE
THAT IT WILL BE USEFUL, BUT WITHOUT ANY WARRANTY, WITHOUT EVEN THE
```

IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
See the applicable licenses for more details.''

Disclaimer of warranty for software under GPL v1 and GPL v2:

NO WARRANTY

BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM ''AS IS'' WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

Disclaimer of warranty for software under LGPL v2.1:

NO WARRANTY

BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE LIBRARY ''AS IS'' WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE
LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER
SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

Disclaimer of warranty for software under GPL v3 and LGPL v3:

Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE
LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE PROGRAM ''AS IS'' WITHOUT WARRANTY OF ANY KIND,
EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE
ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU.
SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY
SERVICING, REPAIR OR CORRECTION.

Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS
THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING
ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT
OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO
LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU
OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER

PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

To fulfill the requirements of certain licenses (and the GNU Lesser
General Public License or LGPL in particular), the Linux system
partitions and file systems are readily accessible (e.g. for replacing
libraries) by booting from a suitable USB device (e.g. a live Linux system,
pressing the 'F11' key during start-up on a USB keyboard enters the boot
menu).

Linux system components can be found in the usual locations; additional
components are located on /dev/sda1 under the /zes directory tree.

In addition to the usual build tools like compiler, linker, tex, etc.,
supplementary Debian system packages may be required for building (see
the description files (.dsc) contained in each source package).

The build system was based on Debian 9.8. It is recommended to use this
version for builds and modifications.

The following sequence of commands allows to edit the bootable Linux image:

```
mount /tmp/lmg600.original.img /mnt/animg/
mkdir /tmp/newiso
cd /mnt/animg/
tar cf - . | (cd /tmp/newiso ; tar xfp -)
rm /tmp/newiso/live/filesystem.squashfs
unsquashfs  -d /tmp/nonexistent_unsquash_dir /mnt/animg/live/filesystem.squashfs
# edit as needed in /tmp/nonexistent_unsquash_dir

mksquashfs /tmp/nonexistent_unsquash_dir /tmp/newiso/live/filesystem.squashfs -b
1024k -comp xz -Xbcj x86 -e boot

xorriso -as mkisofs -R -r -J -joliet-long -l -cache-inodes -iso-level 3
-V ZESSTICK -o /tmp/changed.img -isohybrid-mbr /usr/lib/ISOLINUX/isohdpfx.bin
-partition_offset 16 -c isolinux/isolinux.cat -b isolinux/isolinux.bin
-no-emul-boot -boot-load-size 4 -boot-info-table -eltorito-alt-boot /tmp/newiso/

# use /tmp/changed.img as described in manual to make a bootable stick
```

2 Introduction

2.1 Typical usage

The requirements of modern power measurement have changed a great deal in recent years. While it was previously sufficient to measure a single three-phase supply system with a three-phase power meter, this has become a rare task by now. When using a variable-frequency drive, the output power to the motor must be measured as well as the DC link circuit and the mechanical power of the motor itself. Modern lighting equipment as well as solar inverters, often requires the efficiency measurement of the electrical ballast. The signals to be measured simultaneously are very different: DC, mixed signals, high frequency AC signals, single-phase, three-phase, either with as much bandwidth as possible or with a reduced bandwidth by filtering. Modern power measuring equipment is required to perform tasks such as star-delta conversion, calculation of harmonic values, measurement of total losses and of mechanical power simultaneously.

Also, the requirements of modern IT systems have changed. Nowadays, mobile USB storage devices are used almost exclusively. Common RS232 and IEEE488.2 interfaces are still in use, but LAN interfaces have become more attractive due to lower costs and higher transfer speed.

In the past, all these tasks were solved by using a variety of instruments. The user had then the painful task of controlling and synchronizing the instruments in order to get consistent and meaningful data.

ZES ZIMMER has addressed all these challenges during the developing the LMG600. While this instrument is pushing the limits of what is technically possible, the complex subject is presented to the user in a novel attractive and easily accessible way.

2.2 New features

The measurement channels are the base of all measurements. Several of these channels can be combined together into groups. The channels of a group have a common setup such as the same synchronization and filter settings. In consequence, a group needs to be be made up of equal channel types with matching properties. For channel types and their properties see TECHNICAL DATA [4→39].

A group represents a logical measurement point (e.g. it can represent a three-phase system with voltages in delta circuit and currents in star circuit, an Aron circuit, a single phase measurement, etc.).

Novel group properties of the LMG600:

DualPath Processing: Parallel measurement of narrow and wide bandwidth values

In previous generations of instruments, a voltage or current channel contained an analog signal conditioner, whose output was digitized by an analog-digital converter (ADC) and then passed on to further digital signal processing. Afterwards, the signal could either be measured with full bandwidth, or after filtering with an anti-aliasing filter (to meet the Nyquist–Shannon sampling theorem) to perform analyses like fast Fourier transform (FFT) or digital filtering. The common practice to measure the signal with full bandwidth and to analyze it without using any anti-aliasing filter bears a very big risk of producing incorrect values. An error of 50 % caused by aliasing could be recognized quickly, but a 0.5 % error would be much harder to detect. While this is considered a practicable way, it is not satisfying at all in many cases.

In the LMG600, the complete signal conditioning has been improved. The analog signal conditioning stays unchanged, but the following processing has been revolutionized: Each single voltage and current channel of the LMG600 contains *two* analog-digital converters (ADC). One for the wide bandwidth signal without filtering and one for the narrow bandwidth signal with proper anti-aliasing filter. The sampling values of both ADCs are digitally processed in parallel. With this novel DualPath approach, the user has access to narrow and wide bandwidth values simultaneously without the risk of aliasing.

Access of the measurement values

Well proven techniques like star-delta conversion have already been known to produce multiples of measured values. In the past, it was relatively simple to assign each of these calculated values to a calculation channel. Today, this is no longer possible because there are too many combinations (narrow/wide bandwidth, physical/star/delta/sum values). The result would be a large confusing number of channels and the handling of the user interface would become very inconvenient.

With the LMG600 comes a complete solution. On the display, the values of a group are show and one can comfortably select any measurand by means of the user interface.
When the instrument is being controlled remotely (via the interfaces), the improved SCPI concept of using a suffix comes into effect. In addition to the physical channel number, there is a 4-digit suffix for each group. Each digit is used to encode a property of the desired value like bandwidth or wiring (e.g. star, delta, etc.). For more details about this system see LOGICAL SUFFIXES [9.2.4→218].

Measuring different types of values simultaneously

In the past, when values like RMS, harmonics, or flicker were measured, the instruments were usually limited in the variety of what type of values could be simultaneously measured and recorded depending on the measuring mode. One reason for this limitation was the limited digital signal processing power. Another reason is the fact, that measurement of both, wide bandwidth values and harmonics, with just one ADC is not possible (as discussed above).
The LMG600 has the capability to measure wide bandwidth values and harmonics simultaneously using the novel DualPath technology. The hardware and software of the instrument is designed for this task. However, this leads to new challenges when recording values. For example if one wants to measure the harmonics according to the standard over an interval of 200 ms, but the RMS values over an interval of 50 ms, the interesting question is when to record which value in order to get a meaningful time relation. With the LMG600, the user can decide whether the recording is triggered by the RMS values or by the harmonics. If the recording is triggered by the slower signal, only the last set of faster values will be stored. If the recording is triggered by the faster signal, the slower values will be stored repeatedly if needed. This process provides the highest possible consistency of recorded data.

2.3 Organization of this manual

This manual was up-to-date at the time of publication. The manuals corresponding to the current software version can be accessed in the "Files" Tab in [**STORAGE**] (see FILES TAB [7.3.20→193]).

This manual is organized in the following way:

- In SAFETY [3→31], very important safety information is provided. Please read it carefully to prevent hazards.

- In TECHNICAL DATA [4→39], information about the performance of the instrument, starting with environmental conditions and ending in the exact specifications, is presented.

- In INSTALLATION [5→85], an overview of the control elements and connectors as well as hints concerning maintenance are provided.

- In BASIC CONCEPTS [6→101], background information about basic working and organization principles is described. Additionally, information concerning the connection of the instrument to different measuring circuits is provided

- In GRAPHICAL USER INTERFACE (GUI) [7→145], the menus and how to use them are described. For those just wanting to remote control the instrument, it is worthwhile to read this section as here the different settings (and so their commands) are arranged in such a way, that you find commands which belong together.

- In REMOTE CONTROL [9→215], one can find a detailed description on how to remotely control the instrument. A list of all available commands with their description (e.g. allowed values, details to pay attention to, etc.) is also provided.

- Finally, an index is provided which can be used to find any topics, commands, etc.

3 Safety

3.1 Used symbols

This user manual describes and explains symbols which can be found throughout this manual and are displayed on the instrument. Strict observation of these warning signs is required for safe operation.

Electric shock
This symbol indicates danger of injury or death from electric shock due to dangerous voltages. *Do not touch. Use extreme caution.*
AC voltages over 30 V RMS, 42.4 V peak and DC voltages over 60 V are deemed to be hazardous live according to IEC 61010 resp. EN 61010. There is a danger of electric shock. This can cause death or injury to body or health. Furthermore, there is a risk of material damage.

High temperature
This symbol indicates a high temperature. There is a burn and fire hazard. There is a danger of fire or injury to body or health due to hot surfaces or material. Furthermore, there can be material damage to other objects due to contact or close proximity. If a burn or fire does occur, there can be further damage which can cause death or injury to body or health.

Caution
This symbol indicates the risk of damage to persons or material. Also if material damage occur, there can be further damage which can cause death or injury to body or health.
This symbol on the instrument indicates that this user manual has to be consulted for instruction or further information to ensure save operation.

Information
This symbol indicates facts or information regarding the power analyzer which should be observed for easy and accurate operation.

Male connector
This symbol indicates a male connector.

Female connector
This symbol indicates a female connector.

Narrow bandwidth
This symbol indicates a narrow bandwidth in DualPath processing mode, see Bandwidth [6.6→110].

Wide bandwidth
This symbol indicates a wide bandwidth in DualPath processing mode, see Bandwidth [6.6→110].

Star circuit
This symbol is used to indicate a circuit or wiring in star circuit, see Star-delta conversion, option L6-OPT-SDC [6.3→106].

Delta circuit
This symbol is used to indicate a circuit or wiring in delta circuit, see Star-delta conversion, option L6-OPT-SDC [6.3→106].

Protective conductor terminal
This symbol indicates the terminal for the protective conductor. See also Connection to Protective Earth and Power Supply [3.3.1→34].

Soft keys

<Symbol>

Soft keys do not have a fixed function. The contextual assigned function is indicated in the GUI next to the soft keys. See also FRONT PANEL [5.2→88].

[Key]

Fixed function keys
Fixed function keys (further on just called keys) have the function designated above the key.

Item

Touch element
A displayed item in the GUI which can be selected, activated or changed by direct touch.

[→31]

References/links
References to tables, figures, listings, etc. consist of their identifier followed by an arrow and the page number.

USED SYMBOLS
[3.1→31]

References to chapters, sections, and subsections consist of the heading of the section and the sectional number followed by an arrow and the page number.

UTRMS
[9.10.217→319]

References to interface commands (always given from the short command set because they are unique) are written in typewriter font.

Clicking on any of these references in the PDF version of this document navigates directly to the respective section.

3.2 Inteded Use

This device is an electrical measuring instrument designed for measuring, displaying, and storing various electrical quantities, whose type and exact rated values can be found in the Technical Data (siehe TECHNICAL DATA [4→39]). With the help of suitable sensors, non-electrical quantities can also be measured. The following are defined as proper use:

- The measuring circuits may be connected to external circuits for the purpose of measurement and testing, as well as for supply purposes when using appropriate sensors.

- These measurements can be carried out in the laboratory, but also in production and during the testing of finished devices and systems.

- Operation is carried out exclusively under the environmental conditions specified in ENVIRONMENTAL CONDITIONS [4.1→39].

- The intended operating positions are:
    - The device is standing firmly on all four feet on a stable, level surface.
    - The device is securely installed in a rack.

- The device may only be operated from a socket with a protective earth contact. Voltage, frequency, and fusing must match the technical data of the device.

- The device does not contain dangerous voltages in its delivered state. These are applied by the user for supply and measurement purposes. Therefore, the device is intended for use only by persons who have relevant training and education. They are familiar with the signals they apply and can assess and minimize the resulting dangers.

- The device must be checked at least annually and after every repair for the following:
    - The proper functioning of the protective earth conductor.
    - The reliability of the insulation.
    - The amount of leakage current.

- The device must be transported in its original packaging.

The following are excluded from proper use:

- The device is not intended for detecting hazards or similar conditions. An incorrect reading (e.g., due to an incorrect filter or measurement range) could give a false impression of a safe state. Devices specifically designed for the safe detection of hazardous conditions, such as voltage testers, must be used for this purpose.

- The use in wet or explosive environments or near flammable gases and liquids is not permitted.

- The device must not be used in medical or any other areas where it could potentially impact the health or life of individuals.

3.3 Safety Instructions and Risks

Occupational safety on site depends not only on a device suitable for the task but also on its proper use. Especially at the interface between a device and the external environment, factors can occur that are beyond the influence of the device design. Therefore, the use of this device is restricted to appropriately trained and educated individuals who are familiar with the relevant dangers and can adequately mitigate them.

Below are notes on some typical risks that the user should consider. It is the user's responsibility to conduct their own risk analysis to identify and manage any additional risks.

Regarding the safety of the device, the following standards were considered during development, the edition can be found in the above declaration of conformity:

- IEC 61010-2-030 and EN 61010-2-030

- IEC 60664-1 and EN 60664-1

The device left the factory in a technically flawless condition. To maintain this condition and ensure safe operation, the user must observe the notes and warnings contained in this manual. The device may only be used for the tasks described in this manual.

If it is assumed that safe operation is no longer possible, the device must be immediately disconnected from all measurement and supply lines, taken out of operation, and secured against unintentional operation. The necessary fault-finding should only be carried out by a trained and educated person who is familiar with the associated dangers.

It is particularly assumed that safe operation is no longer possible if:

- the device shows visible damage or deformation,

- the device no longer operates correctly,

- the device has been severely overloaded by excessive currents (short circuit, etc.),

- the device has been severely overloaded by excessive voltages,

- the device has been connected to excessively high supply voltages,

- insulation breakdown or flashover is suspected or noticed (e.g., crackling noise, loud bang, etc.),

- there are loose parts inside the device,

- there is a noticeable smell of burning or scorching,

- a fuse blows again after replacement,

- after prolonged storage under unfavorable conditions,

- after condensation (formation of condensation) due to strong temperature fluctuations,

- after heavy transport stress.

In principle, when dealing with electricity, the relevant, generally accepted safety regulations must be observed and followed. Additionally, the following points must be particularly noted for safe operation:

- Opening the housing can expose parts that may be under high voltage. Therefore, this is only permitted for instructed personnel. Opening the housing risks possible injuries and voids the manufacturer's warranty. All voltage sources must be disconnected from the device before the housing can be opened. Only sufficiently qualified personnel are permitted to do this for the purpose of adjustment, service, repair, or component replacement.

- Fuses may only be replaced with the specified types with the corresponding ratings as indicated in this manual. The use of repaired, short-circuited, or unsuitable fuses is not permitted. The ratings must not be read off the fuse being replaced.

- Devices and accessories (such as cables or clamps) must be inspected before each use. Defective parts must be replaced immediately.

- The air intake and outlet must be kept clear according to the technical data (see ENVIRONMENTAL CONDITIONS [4.1→39]) to allow the necessary airflow and prevent the device from overheating. Similarly, the air filters must be kept clean. The device must not be operated without air filters and their holders, as injuries may result. When the device is mounted in a rack, it must be ensured that the supporting rails do not block the ventilation openings.

- Shocks and rough handling can damage the device. No heavy objects should be placed on the device.

- If the weight (see WEIGHT [4.2.1→46]) of the device is too great for one person to carry, it must be carried by 2 persons or with the help of suitable tools. In all cases, any available handles should be used to lift and carry the device safely.

- No liquids must be placed on the device.

- When connecting the device, the correct sequence must be observed:

  - First, connect the protective earth and power supply (see CONNECTION TO PROTECTIVE EARTH AND POWER SUPPLY [3.3.1→34]),

  - Then connect the measuring circuits (see CONNECTION TO THE EUT CIRCUIT [3.3.2→35]).

  - Finally, turn on the device and, after rechecking the wiring, switch on the supply voltage of the measuring circuit.

- This device has been designed according to safety standards, which are safety standards for measuring devices. In a specific application or environment, compliance with additional safety standards may be required, which must then be applied as well.

3.3.1 Connection to Protective Earth and Power Supply

- Before connecting the power cable to the power supply, it must be checked whether the supply is compatible with the information on the device or nameplate. Any possible voltage selector switch must be set accordingly. Only a suitable power source may be used.

- An interruption by plugs, cables, extensions, etc., without a protective earth connection is not permitted.

  The device must be connected to a socket with a protective earth connection before any other electrical connections are made. Any interruption of the protective earth conductor, inside or outside the device, can lead to a dangerous condition and is therefore not allowed. The use of plugs, cables, extensions, etc., with only two poles is not permitted.

  The *additional* protective earth connection on the back of the device must be used if the measurement circuit could deliver earth currents of more than 10 A in the event of a fault.

  > *Such currents are too large to be safely conducted by the power cable in the event of a single fault. If this were interrupted, the housing would no longer be safe to touch, and one could receive an electric shock.*

In this case, the additional protective earth connection must be connected with a cable of suitable cross-section to an appropriate protective earth. The additional protective earth connection is designed for currents up to 32 A.

If a sufficient protective earth is not available, the measuring leads to the device can also be adequately fused to safely interrupt excessive earth currents.

The additional protective earth connection must not be used as the only protective earth connection, nor should any external devices or circuits be connected to the device's protective earth through it. The additional protective earth connection is marked with the following symbol:

3.3.2 Connection to the EUT Circuit

- Before the measurement circuit is connected to the device, it must be de-energized. Cables must not be plugged in or unplugged when they have a voltage relative to earth.

- Only measurement cables with safety connectors and adequate cross-section should be used. The cables must have a rated voltage suitable for the measurement, especially with regard to the overvoltage and measurement category.

    Cables with uninsulated 4mm connectors may have inadequate air and creepage distances, making them dangerous and potentially causing electric shock even when plugged in.

Only cables whose color matches the color of the sockets should be used to effectively prevent incorrect connections. When connecting the measurement circuit, special care must be taken to avoid plugging voltage measurement leads into the current measurement sockets, as this will result in a short circuit, which can damage the device and injure the user.

    Such short circuits can be very dangerous, as currents of several thousand amperes can flow.

To prevent this, *only* cables with integrated fuses should be used for applying measurement voltages to the device.

    The fuse in the voltage measurement leads will interrupt the current flow if this lead is accidentally plugged into the low-resistance current measurement sockets. This prevents the short-circuiting of a source with high short-circuit power and/or measurement category (e.g., a distribution transformer) from causing a hazard.

The yellow and black voltage measurement leads each have an integrated fuse. This should be checked before and after each measurement. To replace the fuse, the cable must be disconnected from all circuits on both sides to eliminate dangerous voltages relative to earth. Unscrew the fuse holder and insert a fuse of the following type: 6.3x32 mm, FF 500 mA, 1000 V, AC+DC, 30 kA breaking capacity. Finally, screw the fuse holder back together.

- Mixing up current and voltage connections can also occur on the application side, far away from the device. Therefore, this connection should be checked multiple times.

- When connecting to sources with high short-circuit power and/or measurement category (e.g., a distribution transformer), severe damage can occur from mixing up cables, short-circuiting the measurement circuit, using current measurement sockets instead of voltage measurement sockets, or similar errors. Therefore, it is recommended to include appropriate fuses and residual current circuit breakers in all measurement leads (including current measurement leads). When selecting fuses, at least the following points must be considered:

    – The measurement current must flow safely (rated current of the fuse)

    – The short-circuit current must be safely interrupted (breaking capacity of the fuse)

- The maximum voltage must be safely interrupted (rated voltage of the fuse)

- The type of fuse must be appropriate for the current type: AC, DC, or both (breaking capacity of the fuse)

- The fuse must trip quickly enough to protect cables and the device

- The maximum voltage between the sockets ⊙**U** and ⊙**U*** as well as ⊙**U** and ⊙**U$_{Sensor}$** must not exceed the specified values (see POWER MEASUREMENT CHANNEL L60-CH-A1 AND L60-CH-A2 AND L60-CH-A2.1 [4.7→55], POWER MEASUREMENT CHANNEL L60-CH-B1 AND L60-CH-B2 AND L60-CH-B2.1 [4.8→62], etc.).

- The maximum currents in the sockets ⊙**I** and ⊙**I*** and the maximum voltage between the sockets ⊙**I** and ⊙**I$_{Sensor}$** must not exceed the specified values (see POWER MEASUREMENT CHANNEL L60-CH-A1 AND L60-CH-A2 AND L60-CH-A2.1 [4.7→55], POWER MEASUREMENT CHANNEL L60-CH-B1 AND L60-CH-B2 AND L60-CH-B2.1 [4.8→62], etc.).

- The maximum voltage of the sockets ⊙**U**, ⊙**U***, ⊙**U$_{Sensor}$**, ⊙**I**, ⊙**I*** and ⊙**I$_{Sensor}$** against earth must not exceed the specified values (see POWER MEASUREMENT CHANNEL L60-CH-A1 AND L60-CH-A2 AND L60-CH-A2.1 [4.7→55], POWER MEASUREMENT CHANNEL L60-CH-B1 AND L60-CH-B2 AND L60-CH-B2.1 [4.8→62], etc.).

- The maximum voltage of the sockets ⊙**U** ⊙**U*** and ⊙**U$_{Sensor}$** against ⊙**I** ⊙**I*** and ⊙**I$_{Sensor}$** must not exceed the specified values (see POWER MEASUREMENT CHANNEL L60-CH-A1 AND L60-CH-A2 AND L60-CH-A2.1 [4.7→55], POWER MEASUREMENT CHANNEL L60-CH-B1 AND L60-CH-B2 AND L60-CH-B2.1 [4.8→62], etc.).

- External current transformers or current sensors with 1 A or 5 A secondary current may only be connected to the ⊙**I** and ⊙**I*** sockets. External sensors with higher currents are not permitted, as these could drive too high currents in the event of an overload, which could overload the measurement channel and interrupt the current flow! For the same reason, no fuses should be present in the current measurement leads. Before using the sockets, it must be tested that they are low-resistance connected to each other to avoid high voltages at the output of the external sensors.

  It is dangerous to interrupt the secondary current of a transformer, as this can result in very high voltages and the risk of electric shock. For transformers with 1 A or 5 A secondary current, the ⊙I and ⊙I* sockets offer sufficient overload resistance, so they will not interrupt the circuit during usual overloads.

- Cables from/to external sensors are often only designed for the low voltages (<10 V) conducted through them. If such sensors are used in an environment with hazardous voltages, additional insulation may be necessary. For operation, the insulation is sufficient, but if such a cable comes into contact with a bare busbar or similar with hazardous voltage, an unsafe condition could arise, necessitating further insulation. A typical case is that the secondary current cables of an isolating current clamp touch an adjacent conductor and the cable insulation is not rated for this voltage.

- Measurement circuits must be completely protected against contact to safely avoid hazards from dangerously active parts and potential displacement. These can occur, for example (but not exclusively), when the current is measured over ⊙**I*** and ⊙**I** in one phase and at the same time a cable is plugged into ⊙**I$_{Sensor}$**, which then has the potential of the phase and possibly transfers it to places where it is not expected. Therefore, this contact protection must be designed for the highest voltage present in the setup.

- When making external connections, static discharges should be avoided.

- Different sensors can be connected differently to the device. Therefore, when changing a sensor, it must be ensured that the correct connection cable is used. Typically, a cable is permanently assigned to a sensor.

- Distance must be kept from active circuits to avoid electric shock. When measuring in installations or circuits, all relevant safety measures and regulations must be observed. Only suitable measuring accessories should be used. Only sufficiently trained personnel are allowed to measure on active circuits.

- When the measuring device is switched off, all external cables must be unplugged. Particularly with current sensors and current transformers, it must be ensured that the secondary current is not interrupted or the primary current is switched off. After disconnecting the secondary side, it must be short-circuited again to avoid dangerous voltages.

3.4 Equipment ratings

The instrument is rated for following values:

- Supply, see Basic instrument [4.2→39]

- Connections, see Basic instrument [4.2→39], Power measurement channel L60-CH-A1 and L60-CH-A2 and L60-CH-A2.1 [4.7→55], Power measurement channel L60-CH-B1 and L60-CH-B2 and L60-CH-B2.1 [4.8→62], Power measurement channel L60-CH-C1 and L60-CH-C2 and L60-CH-C2.1 [4.9→67], Power measurement channel L60-CH-S2 [4.6→48]

- Environmental conditions, see Environmental conditions [4.1→39]

- Degree of ingress protection (IP), see Basic instrument [4.2→39]

3.5 Installation

For the installation of the instrument, the following items have to be regarded:

- Apply no signals while the instrument is not operating, see Safety Instructions and Risks [3.3→33]

- Installation/Mounting Desktop/Rack, see Installation [3.5→37]

- The instrument must be connected to a suitable protective earthing, see Connection to Protective Earth and Power Supply [3.3.1→34]

- Before connecting to the power supply, ensure compatibility, see Connection to Protective Earth and Power Supply [3.3.1→34]

- Ventilation requirements, see Environmental conditions [4.1→39]

- The safety of any system incorporating the equipment is the responsibility of the assembler of the system!

3.6 Operation

Before and while operating the instrument, the following items have to be considered:

- Description of the controls, see Front panel [5.2→88].

- The instrument shall not be positioned in such a way that operation of the disconnecting device is hampered.

- Connection of accessories or other equipment, see Connection [6.2→102].

- For explanations of the used symbols on the instrument, see Used symbols [3.1→31].

- Consumable materials (battery), see Battery [5.5.4→95].

- Cleaning of filter, housing, see Cleaning [5.5.1→94] and Air filter [5.5.5→95].

- If the instrument is used in a manner not specified by the manufacturer, the protection provided by the instrument may be impaired!

3.7 Technical assistance

For technical assistance you can contact the supplier of the instrument or the manufacturer:

ZES ZIMMER Electronic Systems GmbH
Pfeiffstraße 12
61440 Oberursel
Germany
Phone: +49 (0)6171/88832-0
Fax: +49 (0)6171/88832-28
Email: info@zes.com
URL: http://www.zes.com

4 Technical data

The terms and definitions used in this section are amongst others defined in IEC 61010-1/EN 61010-1 (version according to the above declaration of conformity).

4.1 Environmental conditions

This instrument is intended for use under the following environmental conditions:

- Indoor use only

- Altitude up to 2000 m

- Temperature 5 ℃ to 40 ℃

- Maximum relative humidity 80 % for temperatures up to 31 ℃ decreasing linearly to 50 % relative humidity at 40 ℃, non-condensing

- Mains supply voltage fluctuations up to ±10 % of the nominal voltage

- Transient overvoltages up to the levels of overvoltage category II, i.e. to be supplied from a power outlet of the building wiring

- Temporary overvoltages occurring on the mains supply

- Pollution degree 2, i.e. only non-conductive pollution occurs except that occasionally a temporary conductivity caused by condensation is expected

- Below the instrument a clearance of >13 mm and on each side of the instrument a clearance of >20 mm must be kept clear of anything which could reduce the air flow for cooling the instrument.

4.2 Basic instrument

4.2.1 The LMG671 ATE (rack device)

The LMG671 ATE (automated test equipment) shares the specifications of the LMG671. Generally, everything in the LMG671's context also applies to the LMG671 ATE. Any important differences are pointed out in the respective sections.

Display

LMG671 ATE: no display, resolution 1280x800 pixels (with external monitor)
LMG671: Color display, resolution 1280x800 pixels
LMG641: Color display, resolution 1024x600 pixels
LMG611: Color display, resolution 1024x600 pixels
LMG670: Color display, resolution 1024x600 pixels
LMG640: Color display, resolution 800x480 pixels
LMG610: Color display, resolution 800x480 pixels

Mains supply LMG670 / LMG671

100 … 240 V, 47 … 63 Hz, max. 300 W (depending on installed options and connected devices), 2 fuses 5x20 mm, F 10 A H 250 V according to IEC 60127-2 sheet 1. Please regard that the breaking capacity has to be "H"!
IEC connector C14 according to IEC 60320-1/EN 60320-1.
Do only use supply cables which have a protective conductor and a cross section of at least 3*0.75 mm^2.

Mains supply LMG640 / LMG641

100 … 240 V, 47 … 63 Hz, max. 200 W (depending on installed options and connected devices), 2 fuses 5x20 mm, F 10 A H 250 V according to IEC 60127-2 sheet 1. Please regard that the breaking capacity has to be "H"!
IEC connector C14 according to IEC 60320-1/EN 60320-1.
Do only use supply cables which have a protective conductor and a cross section of at least 3*0.75 mm^2.

Mains supply LMG610 / LMG611

100 … 240 V, 47 … 63 Hz, max. 200 W (depending on installed options and connected devices), 2 fuses 5x20 mm, F 10 A H 250 V according to IEC 60127-2 sheet 1. Please regard that the breaking capacity has to be "H"!
IEC connector C14 according to IEC 60320-1/EN 60320-1.
Do only use supply cables which have a protective conductor and a cross section of at least 3*0.75 mm^2.

Degree of ingress protection
IP20 according to IEC 60529

Storage temperature
-20 °C … +50 °C

Safety
IEC 61010-1/EN 61010-1, version according to the above declaration of conformity

EMC
EN 61326-1, version according to the above declaration of conformity
EN 61000-3-2, version according to the above declaration of conformity
EN 61000-3-3, version according to the above declaration of conformity

Case dimensions
LMG671 Desktop instr.: (BxHxT) 433 mm x 177 mm x 590 mm
LMG671 19" instrument: (BxHxT) 84 HP x 4 U x 590 mm
LMG641 Desktop instr.: (BxHxT) 284 mm x 177 mm x 590 mm
LMG641 19" instrument: (BxHxT) 84 HP x 4 U x 590 mm
LMG611 Desktop instr.: (BxHxT) 433 mm x 177 mm x 200 mm

LMG670 Desktop instr.: (BxHxT) 432.6 mm x 176.5 mm x 570.7 mm
LMG670 19" instrument: (BxHxT) 84 HP x 4 U x 570.7 mm
LMG640 Desktop instr.: (BxHxT) 283.5 mm x 176.5 mm x 570.7 mm
LMG640 19" instrument: (BxHxT) 84 HP x 4 U x 570.7 mm
LMG610 Desktop instr.: (BxHxT) 432.6 mm x 176.5 mm x 200.0 mm
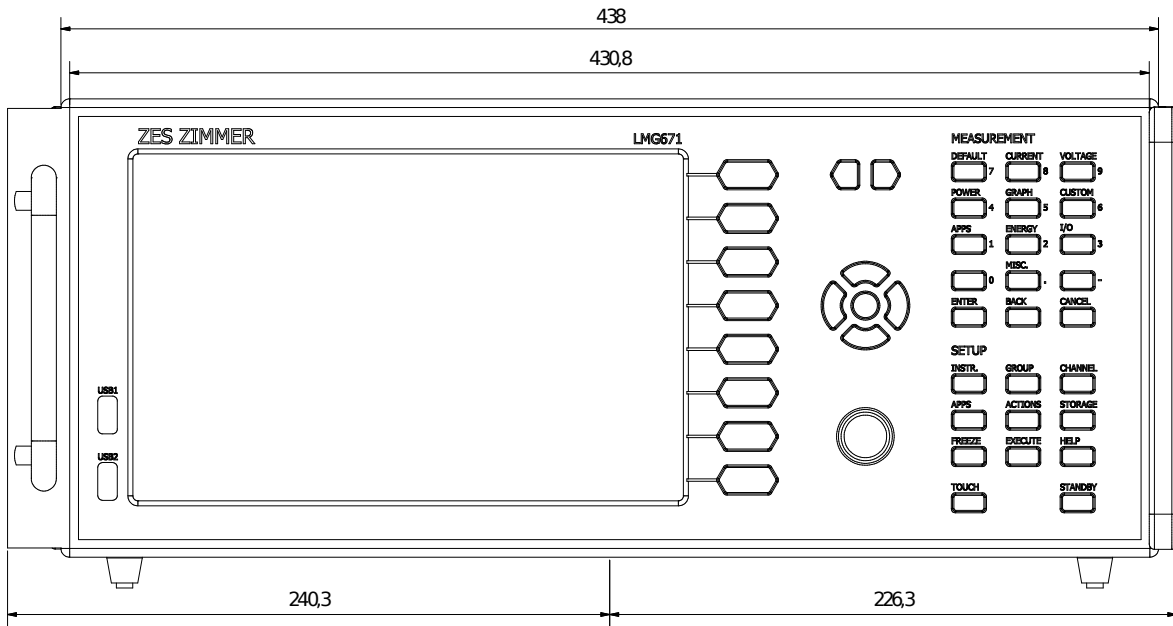
Figure 4.1: LMG671 dimensions front view . On the left with 19" rack mount kit (L671-X-01), on the right with hand grips for the desktop instrument (L6X1-OPT-HDL2 resp. L6X1-OPT-HDL4)
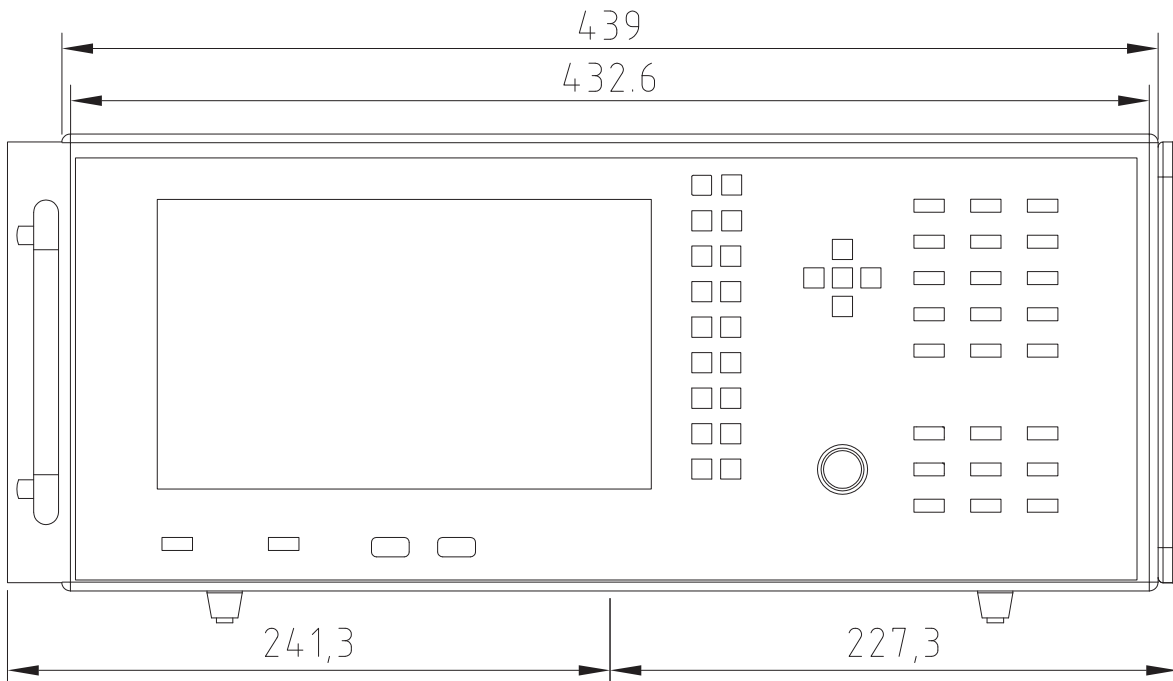
Figure 4.2: LMG670 dimensions front view . On the left with 19" rack mount kit (L67-X-01), on the right with hand grips for the desktop instrument (L6-OPT-HDL2 resp. L6-OPT-HDL4)
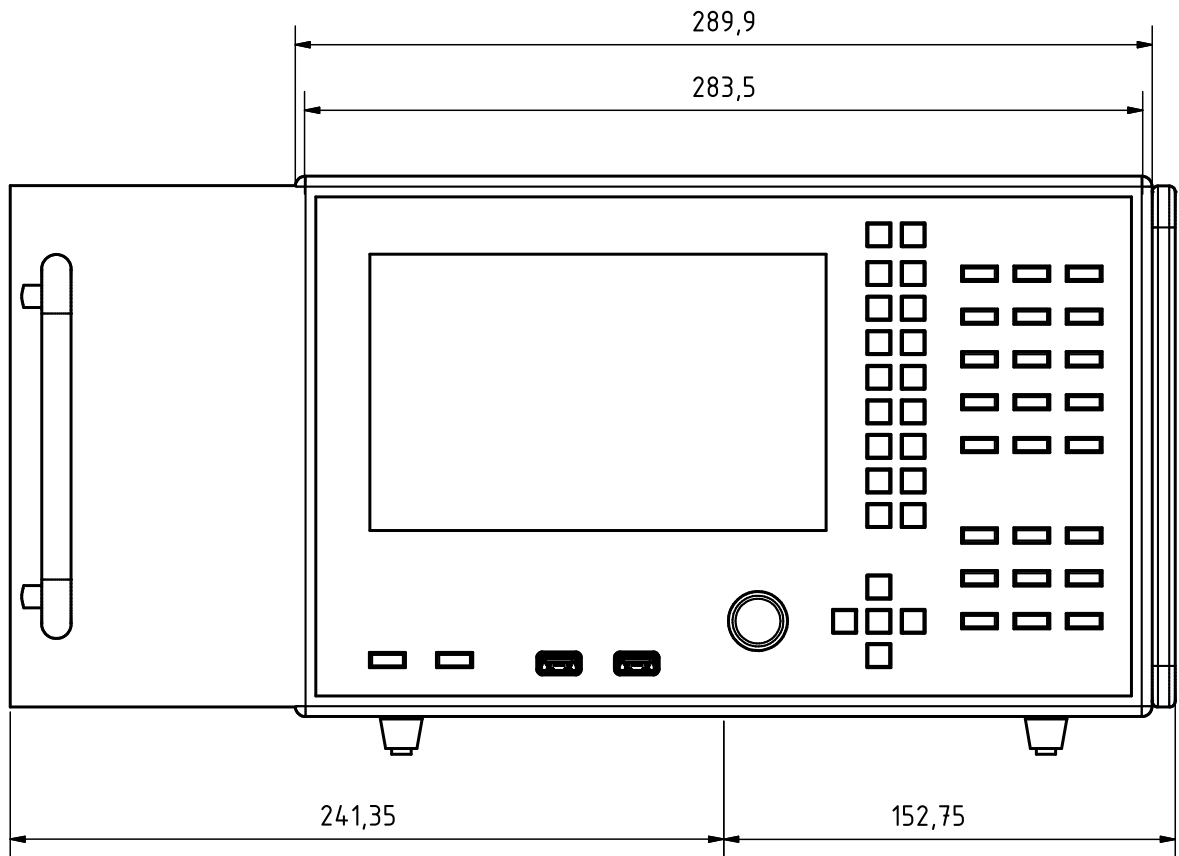
Figure 4.3: LMG640 dimensions front view . On the left with 19" rack mount kit (L64-X-01), on the right with hand grips for the desktop instrument (L6-OPT-HDL2 resp. L6-OPT-HDL4)
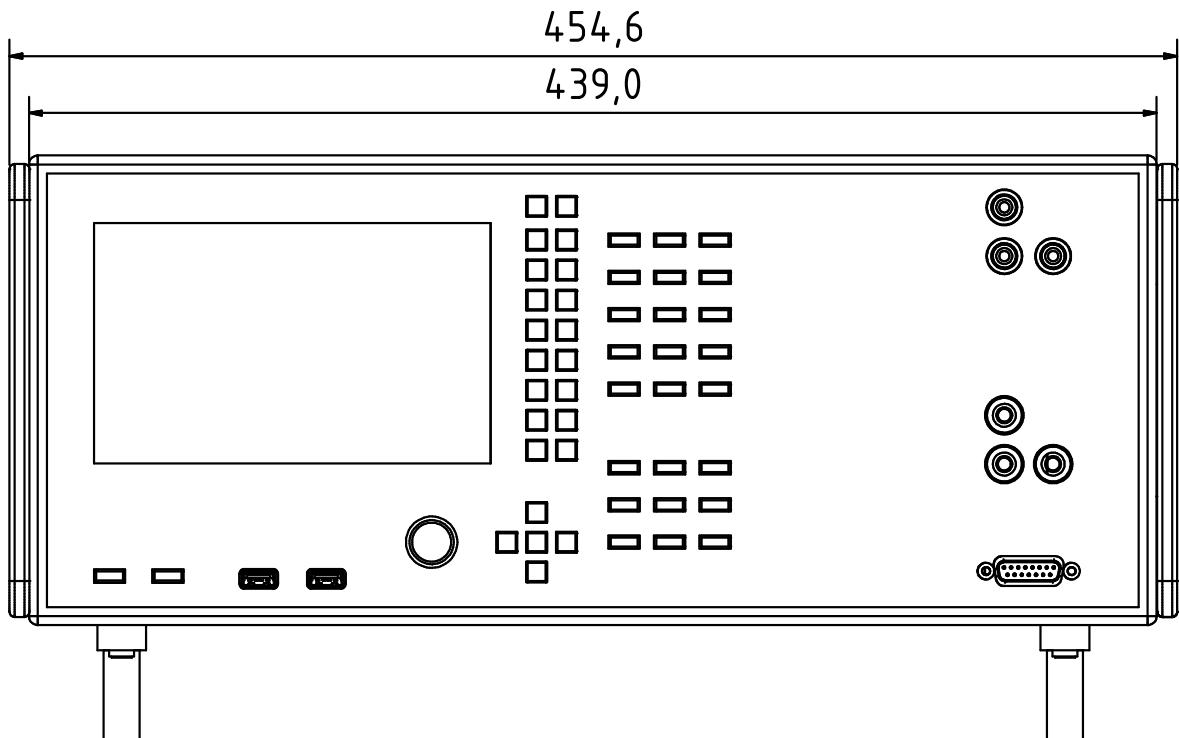
Figure 4.4: LMG610 dimensions front view . On both sides with hand grips for the desktop instrument (L6-OPT-HDL2)

568,7

571,2

614,7

657,7

176,4

191,9
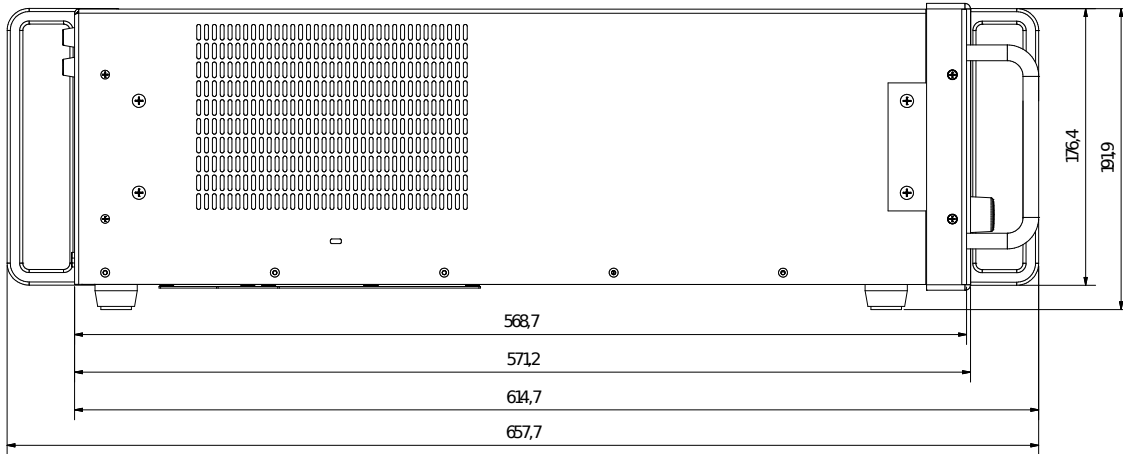
Figure 4.5: LMG671 dimensions side view . On the left (rear) with hand grips for the desktop instrument, on the right (front) with 19" rack mount kit (L671-X-01) in the foreground and with hand grip in the background (L6X1-OPT-HDL2 resp. L6X1-OPT-HDL4)

192,5

176,5

182,9

570,7

573,1

616,7

660,7
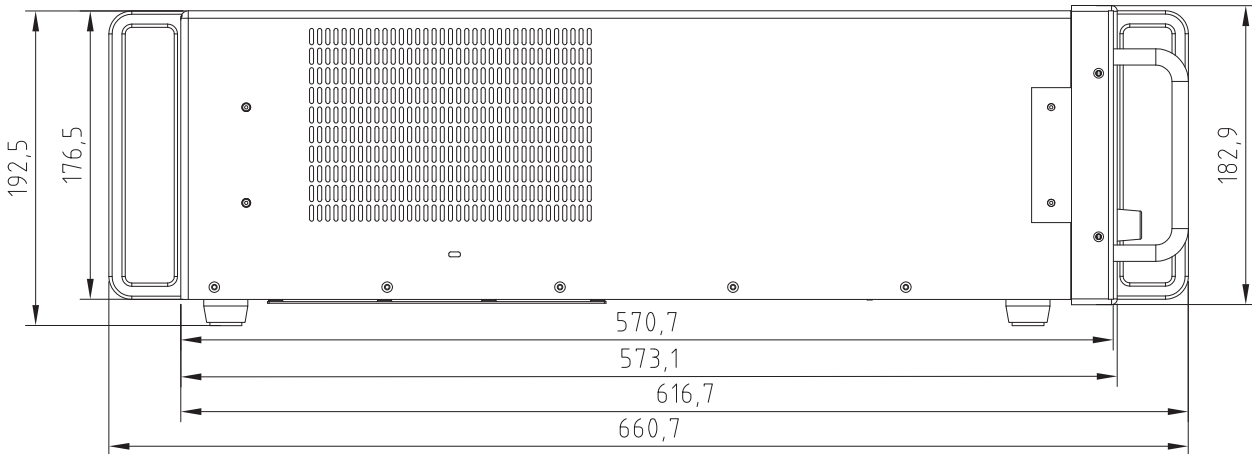
Figure 4.6: LMG670/640 dimensions side view . On the left (rear) with hand grips for the desktop instrument, on the right (front) with 19" rack mount kit (L67-X-01) in the foreground and with hand grip in the background (L6-OPT-HDL2 resp. L6-OPT-HDL4)

243,9

200,3

176,5

192,0

224,1

215,2
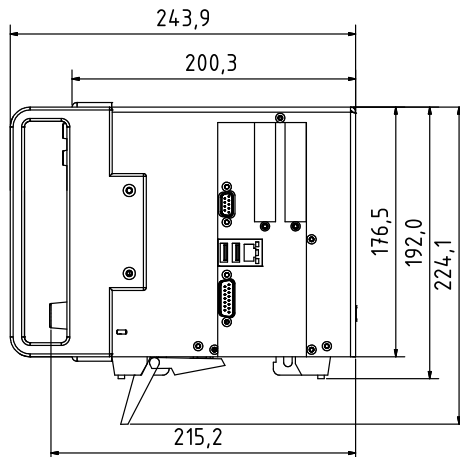
Figure 4.7: LMG610 dimensions side view . On the left (front) with hand grips for the desktop instrument (L6-OPT-HDL2)
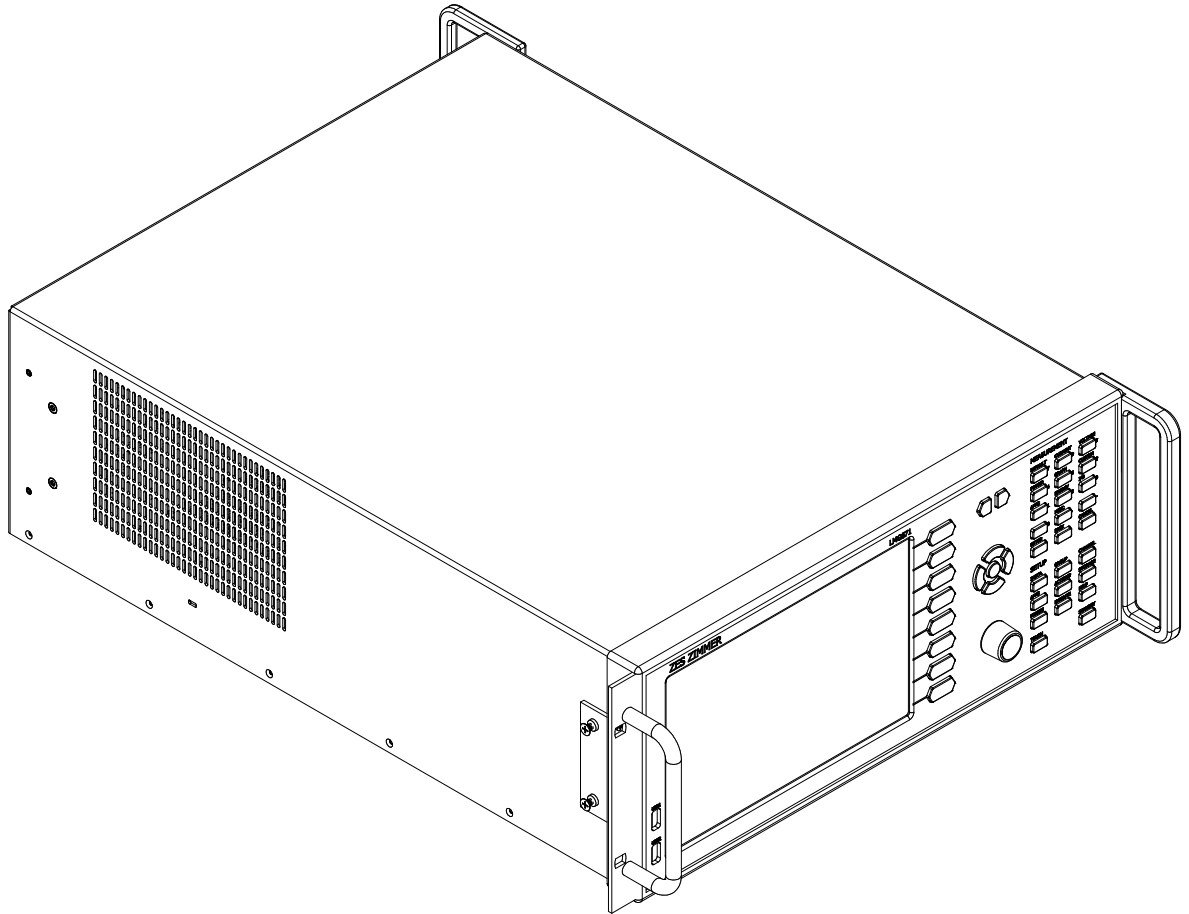
Figure 4.8: LMG671 3D view . with the parts described in Figure 4.1 [→41] and Figure 4.5 [→43].
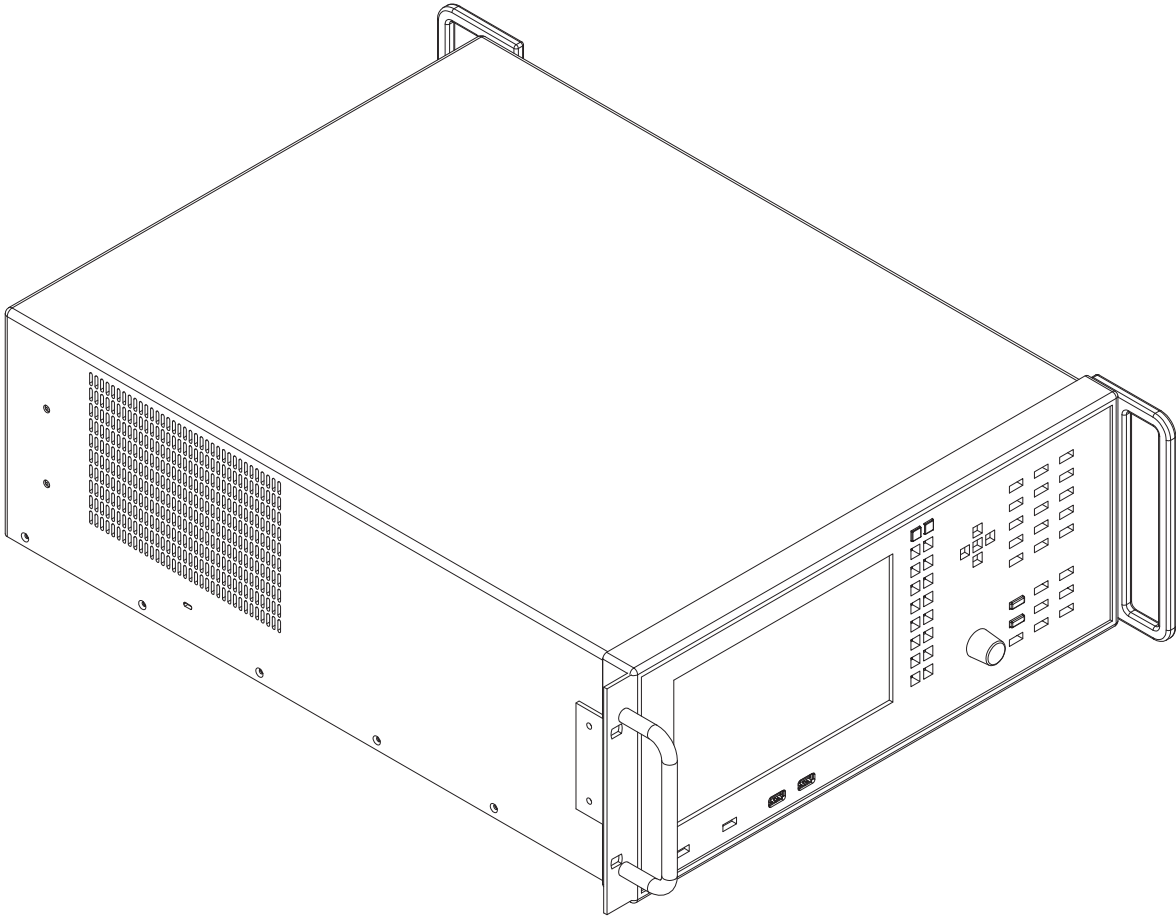
Figure 4.9: LMG670 3D view . with the parts described in Figure 4.2 [→41] and Figure 4.6 [→43].

Weight

Depending on option:
max. 18.5 kg for LMG670 with 7 populated slots
max. 15.5 kg for LMG640 with 4 populated slots
max. 8.0 kg for LMG610

4.3 Synchronization Connector

This jack (see Figure 5.6 [→91]) is used for signals concerning synchronization and measurement control. All of this signals can be configured as input or output and can be used to control the instrument by external devices or to signal the state of the instrument to control external devices, receptively.

Using the cable with the order number L6-ACC-SYNC-2 resp. L6-ACC-SYNC-3 it is possible to connect two resp. three LMG600 instruments to synchronize cyles, frequencies, energy measurement, transient trigger and the clock.The direction of synchronisation can be individually set up by choosing if the individual signals are outputs (master) or inputs (slaves). When having several instruments, of course just one should output, the other should listen.
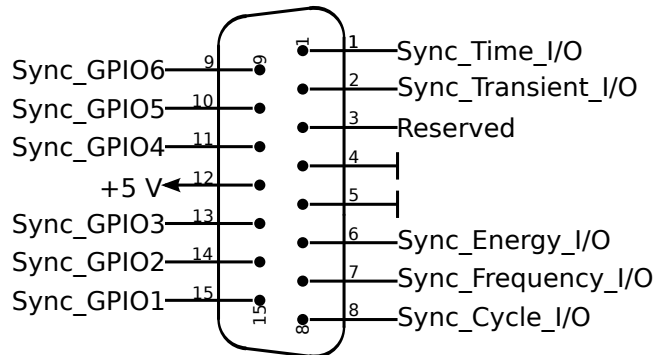
Figure 4.10: Synchronization connector , 15 pin D-SUB male face view

1. SYNC_Time_IO
   This pin is used to synchronize the time of two or more devices.The direction is set up in the [**INSTR.**] menu under the General Tab in the option "Data Source". To send the current time from one device, in "Data Source" the option "Internal" must be activated, for receiving "Other LMG".
   Input: Receives the time send by the sender and sets the internal clock according.
   Output: Sends the current time.
   The time between the devices is synchronized multiple times per second to keep time deviation smaller than 100 µs.

   For reliable time synchronization using LMG6X1 devices, the installed firmware version must be >= 3.012 on all devices.

2. SYNC_Transient_IO
   Input: A rising edge triggers a transient recording in the instrument.
   Output: A rising edge indicates, that a transient was triggered in the instrument. After 0.1 ms the signal returns to 0.

3. SYNC_PPS_IO
   Pulse-Per-Second (PPS) for time synchronization applications. Use <shortRef>CLKDISPPSDIR</shortRef> to configure the I/O direction.

4. GND

SYSDATESOURCE
[9.10.402→406]

5. GND

   GND for all input and output signals and for the auxiliary power supply. Please keep in mind, that this ground is *not directly* connected to the case and the protective earth of the instrument but via suppression components!

6. SYNC_Energy_IO

   During the 1 state the integration is running, during the 0 state there is no integration.

7. SYNC_Frequency_IO

   Input: The synchronisation frequency is determined by the rising edge of the signal. Select external synchronisation (see SYNC SETTINGS [→185].
   Output: The sync of group 1 is output.
   The configuration of the pin as an input or output happens automatically depending on the synchronisation settings of all groups (see SYNC SETTINGS [7.3.12→185]).

8. SYNC_Cycle_IO

   The direction of the pin is controlled by the Cycle setting in MEASUREMENT TAB [7.3.2→176] (CYCLMOD [9.10.355→385]). If an external cycle is selected, the pin is configured as input and each rising edge of the input signal terminates the current measurement cycle and starts a new one. The duration of each cycle (that is the time between two rising edges) must be in the range specified for the CYCL [9.10.356→386] setting. Any other Cycle setting configures the pin as output. If a fixed interval is selected, a short pulse is then output to indicate the end of the current cycle or the start of a new one, respectively. If the cycle is determined by harmonics, no cycle information is available at the pin (static voltage level, no edges are generated).

9. SYNC_GPIO6

   This pin can be configured and used as general purpose input or output.

10. SYNC_GPIO5

    This pin can be configured and used as general purpose input or output.

11. SYNC_GPIO4

    This pin can be configured and used as general purpose input or output.

12. +5 V

    Auxiliary supply with $\pm 10\%$ tolerance, internal serial protective resistor $10\,\Omega$.

13. SYNC_GPIO3

    This pin can be configured and used as general purpose input or output.

14. SYNC_GPIO2

    This pin can be configured and used as general purpose input or output.

15. SYNC_GPIO1

    This pin can be configured and used as general purpose input or output.

Input Signal

Low is detected for voltage $\leq 0.8\,$V (@ -0.15 mA)
High is detected for voltage $\geq 2.3\,$V (@ -0.1 mA)
There is an internal pull up resistor to +5 V. The signal must be in the range from -0.3 V to +5.3 V.

Output Signal

Low voltage $\leq 0.5\,$V, +1.5 mA...-1.5 mA
High voltage $\geq 4.3\,$V, +1 mA...-1 mA

4.4 VGA/DVI interface (option L6-OPT-DVI)

Port to connect an external display or video projector. Supports digital as well as analog signals.
Devices with DVI input can be connected directly. Devices with analog VGA input require an off-the-shelf passive adaptor that converts the DVI pins to the common 15 pin VGA connector (type HD-15).

It is possible to use an external touchscreen with the LMG. In addition to the video connection, such devices have a USB connection for the build-in touch device. This can be plugged into any free front or rear USB port. See also Touchscreen tab [7.3.11→182] for necessary configuration settings, if any.

4.5 DVI interface (LMG6x1 only, option L6X1-OPT-DVI)

This options is by default available in the LMG671 ATE.

Port to connect an external display or video projector. Supports digital signals only.
Devices with DVI input can be connected directly.

It is possible to use an external touchscreen with the LMG. In addition to the video connection, such devices have a USB connection for the build-in touch device. This can be plugged into any free front or rear USB port. See also Touchscreen tab [7.3.11→182] for necessary configuration settings, if any.

4.6 Power measurement channel L60-CH-S2

This channel is rated according to EN 61010-1 and EN 61010-2-030 for measuring voltages from ⊚**U***
to ⊚**U** up to:

- $U_{AC} = U_{DC} = 600\,V$, measurement category CAT IV

- $U_{AC} = U_{DC} = 1000\,V$, measurement category CAT III

- $U_{DC} = 1500\,V$, measurement category CAT II

- other circuits:
  working voltage rms 1000 V,
  working voltage peak 3200 V and
  transient overvoltage 3000 V

This channel is rated according to EN 61010-1 and EN 61010-2-030 for insulation voltages from ⊚**U***, ⊚**U**, ⊚**U**$_{\textbf{Sensor}}$, ⊚**I***, ⊚**I**, ⊚**I**$_{\textbf{Sensor}}$ to protective earth PE and from ⊚**U** to ⊚**I** up to:

- $U_{AC} = U_{DC} = 600\,V$, measurement category CAT IV

- $U_{AC} = U_{DC} = 1000\,V$, measurement category CAT III

- other circuits:
  working voltage rms 1000 V,
  working voltage peak 3200 V and
  transient overvoltage 3000 V

With a measured voltage $U < 1000\,V$ the standard accessory safety laboratory leads with integrated fuses have to be used!

With a measured voltage $1000\,V < U_{DC} < 1500\,V$ safety laboratory leads with appropriate specified voltage rating and suitable fuses has to be used or the measuring circuit has to be protected adequate!

The RMS values specified for the ranges are the maximum RMS values which could be measured

theoretically. This values are not allowed to apply if safe operation or any other safety values are not guaranteed.

Support of *DualPath* (dual bandwidth) mode

This channel supports the *DualPath* mode, see BANDWIDTH [6.6→110].

Sample rate and bandwidth of the AD converter

Table 4.1: L60-CH-S: Sample rate and bandwidth of the AD converter

Wide bandwidth converter	
Sample rate	$1.\overline{21}$ MS/s
Resolution	18 bit
Bandwidth, analog anti-aliasing filter disabled	10 MHz
Bandwidth, analog anti-aliasing filter activated	about 150 kHz
Narrow bandwidth converter	
Sample rate	$151.\overline{51}$ kS/s
Resolution	18 bit
Bandwidth, analog anti-aliasing filter always activated	about 15 kHz

Signal filter

This filters are analog filters in the signal path with a bandwidth of about 15 kHz and 150 kHz. To ensure the anti-aliasing function for the corresponding analog-digital converters (ADC) with an attenuation of >50 dB at higher frequencies than half of sampling rate the specified design parameters were chosen.

Example: The 15 kHz filter has an attenuation of $-14.7440\,\%$ at a signal frequency $f = 10$ kHz. Those $-14.7440\,\%$ are afflicted with an uncertainty of $\pm 0.012 = \pm 1.2\,\%$. Consequently the measuring signal is damped by $-(14.7 \pm 1.2)\,\%$.

Table 4.2: L60-CH-S: Characteristics of the 15 kHz filter

Analog anti-aliasing filter 15 kHz (narrowband)			
Design parameter: Bessel filter, 5th order, cutoff frequency 14.5 kHz			
Frequency / Hz	Filter attenuation in % (nominal)	Worst Case uncertainty of the filter attenuation	Filter attenuation in dB (nominal)
0	0.0004	$\pm 50 \cdot 10^{-6}$	0.0000
10	0.0004	$\pm 50 \cdot 10^{-6}$	0.0000
20	0.0003	$\pm 50 \cdot 10^{-6}$	0.0000
53	0.0000	$\pm 50 \cdot 10^{-6}$	0.0000
100	-0.0012	$\pm 50 \cdot 10^{-6}$	-0.0001
200	-0.0058	$\pm 50 \cdot 10^{-6}$	-0.0005
500	-0.0385	$\pm 50 \cdot 10^{-6}$	-0.0033
1000	-0.1552	$\pm 160 \cdot 10^{-6}$	-0.0135
2000	-0.6210	$\pm 660 \cdot 10^{-6}$	-0.0541
5000	-3.8385	$\pm 3.9 \cdot 10^{-3}$	-0.3400
10000	-14.7440	$\pm 12 \cdot 10^{-3}$	-1.3855
15000	-31.1128	$\pm 18 \cdot 10^{-3}$	-3.2372
75000	-99.7125	—	-50.8267

Table 4.3: L60-CH-S: Characteristics of the 150 kHz filter

Analog anti-aliasing filter 150 kHz (wideband)			
Design parameter: Bessel filter, 7th order, cutoff frequency 145 kHz			
Frequency / Hz	Filter attenuation in % (nominal)	Worst Case uncertainty of the filter attenuation	Filter attenuation in dB (nominal)
0	0.0000	$\pm 50 \cdot 10^{-6}$	0.0000
10	0.0000	$\pm 50 \cdot 10^{-6}$	0.0000
20	0.0000	$\pm 50 \cdot 10^{-6}$	0.0000
53	0.0000	$\pm 50 \cdot 10^{-6}$	0.0000
100	-0.0000	$\pm 50 \cdot 10^{-6}$	-0.0000
200	-0.0001	$\pm 50 \cdot 10^{-6}$	-0.0000
500	-0.0004	$\pm 50 \cdot 10^{-6}$	-0.0000
1000	-0.0016	$\pm 100 \cdot 10^{-6}$	-0.0001
2000	-0.0064	$\pm 100 \cdot 10^{-6}$	-0.0006
5000	-0.0398	$\pm 100 \cdot 10^{-6}$	-0.0035
10000	-0.1593	$\pm 350 \cdot 10^{-6}$	-0.0138
20000	-0.6359	$\pm 1.5 \cdot 10^{-3}$	-0.0554
50000	-3.9203	$\pm 8.7 \cdot 10^{-3}$	-0.3474
100000	-14.9399	$\pm 27.3 \cdot 10^{-3}$	-1.4055
150000	-31.0563	$\pm 42 \cdot 10^{-3}$	-3.2301
600000	-99.7028	—	-50.5399

Harmonics

The special capabilities of this channel for the harmonics measurement are described in INTERHARMONICS = 0 [6.8.1→115].

General Accuracy Specifications For the amplitude, the measurement uncertainties according to tables 4.5 to 4.8 apply to each individual harmonic or intermediate harmonic at the respective frequency as if it had been measured on its own. In addition, the uncertainties (not the damping) of the filter used must be taken into account (table 4.2 and 4.3). The following accuracy specification (4.1) applies to the phase angle if the amplitude of the respective frequency component is $> 0.15\,\%$ of the nominal value of the measuring range and the frequency of the fundamental frequency is in the range from 1 Hz to 10 kHz. h is the order of the harmonics and f_h the according signal frequency.

$$\pm \left(0.05° + (0.005° \cdot h) + \left(0.05° \cdot \frac{f_h}{1\,\text{kHz}} \right) \right) \tag{4.1}$$

A calculated example can be found in 6.16.3.

CE Harmonics

INTERHARM
[9.10.288→353]

CE Harmonische are the case when INTERHARM = 9 and fundamental frequency between 45 Hz and 65 Hz. For those is valid:

1. Relative deviation between f1 and frequency fsyn, to which the sampling rate is synchronized is $<0.03\,\%$ of f1 under steady-state conditions.

2. The attenuation of the anti-aliasing filter is >50 dB at higher frequencies than half of sampling rate.

3. The instrument complies with class I according IEC 61000-4-7 Ed. 2.0 and EN 61000-4-7 Ed. 2.0.

4. Accuracy specification according IEC 61000-4-7 Ed. 2.0 and EN 61000-4-7 Ed. 2.0 with m = measuring value and nom = nominal value of the range:

Table 4.4: L60-CH-S: Accuracy harmonics

voltage measurement	Um ≥ 1 % Unom:	± 5 % Um
	Um < 1 % Unom:	± 0.05 % Unom
current measurement	Im ≥ 3 % Inom:	± 5 % Im
	Im < 3 % Inom:	± 0.15 % Inom

The smaller value of the uncertainty in table 4.4 and in the general accuracy specification applies to the specification of the uncertainty of the amplitude.

CE Flicker

According to EN 61000-4-15:2011 resp. IEC 61000-4-15:2010 this is a class F1 flicker meter.

Flickermeter: ±5 % according to EN 61000-4-15, 0.3<k<5.4

d-meter: ±5 % according to EN 61000-4-15

Accuracy specifications

The accuracy specifications are valid under the following conditions:

1. Sinusoidal voltage and current

2. Ambient temperature (23±3) ℃, no additional heating or cooling (e.g. by sunlight or current of air)

3. The temperature effect is $\pm(0.01\,\%\,\text{of reading})/\,\text{K}$ in the range of $5\,℃ \ldots 20\,℃$ and $26\,℃ \ldots 40\,℃$

4. Warm up time 1 h

5. The upper range value is defined by the maximum peak value, see Voltage channel L60-CH-S2 [4.6.1→54] and Current channel L60-CH-S2 [4.6.2→55]. All other values are calculated from voltage, current and active power. The tolerances results from the functional relationship, see Measurement uncertainty of measured and calculated values [6.16→135].

6. Specified are the direct measured and unfiltered measuring values. For the specification of the filtered measuring values, see Signal filter [4.6→49].

7. Voltage and current are higher than 10 % of nominal range and not higher than the maximum TRMS value of the range. Voltage peak and current peak are not higher than range peak.

8. All specifications above 1 MHz are design values.

9. The instrument should be calibrated every 12 month, last adjustment not older than 12 months and adjustment was done at 23 ℃.

10. Active power specification at alternating current AC: The upper power range value is the product of upper voltage range value and upper current range value. The specification is valid for power factor 0 … 1.

11. To calculate the active power specification ΔP_{DC} use the specification of the voltage ΔU_{DC}, the specification of the current ΔI_{DC} and the measured values of the voltage U_{DC} and of the current I_{DC} with the equation in Table 4.5. The active power specification ΔP_{DC} is based on the total derivative of the equation $P_{\text{DC}} = U_{\text{DC}} \cdot I_{\text{DC}}$.

12. All values of the following tables are to be interpreted as ±(% of reading + % of upper range value), except the active power specification ΔP_{DC}.

Table 4.5: L60-CH-S2: Accuracy DC

Frequency	DC [e]
Voltage ⊙**U***	$0.02 + 0.04$
Voltage ⊙**U$_{\text{Sensor}}$**	$0.02 + 0.04$ [d]
Current ⊙**I*** 5 mA … 5 A range AC, 10 mA … 8 A range DC	$0.02 + 0.04$
Current ⊙**I*** 10 A … 32 A range AC, 15 A … 32 A range DC	$0.02 + 0.1$ [f]
Current ⊙**I$_{\text{Sensor}}$**	$0.02 + 0.04$ [d]
Active power	$\Delta P_{\text{DC}} = \pm\left(\left\lvert\Delta U_{\text{DC}} \cdot I_{\text{DC}}\right\rvert + \left\lvert\Delta I_{\text{DC}} \cdot U_{\text{DC}}\right\rvert\right)$ Description of the used formula symbols, see ACCURACY SPECIFICATIONS [4.6→51], number 11.

[f] Additional accuracy specification in the 10 A … 32 A range AC or 15 A … 32 A range DC: $\pm\frac{80\,\mu A}{A^2} \cdot I_{\text{trms}}{}^2$

[d] Accuracy specification is valid with activated signal filter 15 kHz or 150 kHz

[e] Accuracy specification is valid with activated automatic zero adjustment, max. 24 h after last change of the measuring range in the current measurement channel at jack ⊙**I***, temperature change after change of the measuring range max. ±1 ℃, max. 30 days after persistent zero adjustment in the voltage measurement channel at the jacks ⊙**U*** and ⊙**U$_{\text{Sensor}}$** and in the current measurement channel at jack ⊙**I$_{\text{Sensor}}$** (see ZERO ADJUSTMENT [6.18→139]).

Table 4.6: L60-CH-S2: Accuracy 0.05 Hz … 10 kHz

Frequency	0.05 Hz … 45 Hz 65 Hz … 3 kHz	45 Hz … 65 Hz	3 kHz … 10 kHz
Voltage ⊙**U***	$0.015 + 0.03$	$0.01 + 0.02$	$0.03 + 0.06$
Voltage ⊙**U$_{\text{Sensor}}$**	$0.015 + 0.03$	$0.01 + 0.02$	$0.03 + 0.06$
Current ⊙**I*** 5 mA … 5 A range AC, 10 mA … 8 A range DC	$0.015 + 0.03$	$0.01 + 0.02$	$0.03 + 0.06$
Current ⊙**I*** 10 A … 32 A range AC, 15 A … 32 A range DC	$0.015 + 0.03$ [f]	$0.01 + 0.02$ [f]	$0.1 + 0.2$ [f]
Current ⊙**I$_{\text{Sensor}}$**	$0.015 + 0.03$	$0.01 + 0.02$	$0.03 + 0.06$
Active power ⊙**U***/⊙**I*** 5 mA … 5 A range AC, 10 mA … 8 A range DC	$0.024 + 0.03$	$0.015 + 0.01$	$0.048 + 0.06$
Active power ⊙**U***/⊙**I*** 10 A … 32 A range AC, 15 A … 32 A range DC	$0.024 + 0.03$ [g]	$0.015 + 0.01$ [g]	$0.104 + 0.13$ [g]
Active power ⊙**U***/⊙**I$_{\text{Sensor}}$**	$0.024 + 0.03$	$0.015 + 0.01$	$0.048 + 0.06$
Active power ⊙**U$_{\text{Sensor}}$**/⊙**I*** 5 mA … 5 A range AC, 10 mA … 8 A range DC	$0.024 + 0.03$	$0.015 + 0.01$	$0.048 + 0.06$
Active power ⊙**U$_{\text{Sensor}}$**/⊙**I*** 10 A … 32 A range AC, 15 A … 32 A range DC	$0.024 + 0.03$ [g]	$0.015 + 0.01$ [g]	$0.104 + 0.13$ [g]
Active power ⊙**U$_{\text{Sensor}}$**/⊙**I$_{\text{Sensor}}$**	$0.024 + 0.03$	$0.015 + 0.01$	$0.048 + 0.06$

[f] Additional accuracy specification in the 10 A … 32 A range AC or 15 A … 32 A range DC: $\pm\frac{80\,\mu A}{A^2} \cdot I_{\text{trms}}{}^2$

[g] Additional accuracy specification in the 10 A … 32 A range AC or 15 A … 32 A range DC: $\pm\frac{80\,\mu A}{A^2} \cdot I_{\text{trms}}{}^2 \cdot U_{\text{trms}}$

Table 4.7: L60-CH-S2: Accuracy 10 kHz … 500 kHz

Frequency	10 kHz … 50 kHz	50 kHz … 100 kHz	100 kHz … 500 kHz
Voltage ⊙**U***	0.2 + 0.4		0.5 + 1.0
Voltage ⊙**U**$_{\text{Sensor}}$	0.2 + 0.4		0.4 + 0.8
Current ⊙**I*** 5 mA … 5 A range AC, 10 mA … 8 A range DC	0.2 + 0.4		0.5 + 1.0
Current ⊙**I*** 10 A … 32 A range AC, 15 A … 32 A range DC	0.3 + 0.6 [f]	f/100 kHz · 0.8 + f/100 kHz · 1.2 [f]	
Current ⊙**I**$_{\text{Sensor}}$	0.2 + 0.4		0.4 + 0.8
Active power ⊙**U***/⊙**I*** 5 mA … 5 A range AC, 10 mA … 8 A range DC	0.32 + 0.4		0.8 + 1.0
Active power ⊙**U***/⊙**I*** 10 A … 32 A range AC, 15 A … 32 A range DC	0.4 + 0.5 [g]	f/100 kHz · 0.8 + f/100 kHz · 0.8 [g]	f/100 kHz · 1.0 + f/100 kHz · 1.1 [g]
Active power ⊙**U***/⊙**I**$_{\text{Sensor}}$	0.32 + 0.4		0.72 + 0.9
Active power ⊙**U**$_{\text{Sensor}}$/⊙**I*** 5 mA … 5 A range AC, 10 mA … 8 A range DC	0.32 + 0.4		0.72 + 0.9
Active power ⊙**U**$_{\text{Sensor}}$/⊙**I*** 10 A … 32 A range AC, 15 A … 32 A range DC	0.4 + 0.5 [g]	f/100 kHz · 0.8 + f/100 kHz · 0.8 [g]	f/100 kHz · 1.0 + f/100 kHz · 1.0 [g]
Active power ⊙**U**$_{\text{Sensor}}$/⊙**I**$_{\text{Sensor}}$	0.32 + 0.4		0.64 + 0.8

[f] Additional accuracy specification in the 10 A … 32 A range AC or 15 A … 32 A range DC: $\pm \frac{80\,\mu\text{A}}{\text{A}^2} \cdot I_{\text{trms}}^2$

[g] Additional accuracy specification in the 10 A … 32 A range AC or 15 A … 32 A range DC: $\pm \frac{80\,\mu\text{A}}{\text{A}^2} \cdot I_{\text{trms}}^2 \cdot U_{\text{trms}}$

Table 4.8: L60-CH-S2: Accuracy 500 kHz … 10 MHz

Frequency	500 kHz … 1 MHz	1 MHz … 2 MHz	2 MHz … 10 MHz
Voltage ⊙**U***	0.5 + 1.0	f/1 MHz · 1.5 + f/1 MHz · 1.5	
Voltage ⊙**U**$_{\text{Sensor}}$	0.4 + 0.8	f/1 MHz · 0.7 + f/1 MHz · 1.5	
Current ⊙**I*** 5 mA … 5 A range AC, 10 mA … 8 A range DC	0.5 + 1.0	f/1 MHz · 1.0 + f/1 MHz · 2.0	-
Current ⊙**I*** 10 A … 32 A range AC, 15 A … 32 A range DC	-		
Current ⊙**I**$_{\text{Sensor}}$	0.4 + 0.8	f/1 MHz · 0.7 + f/1 MHz · 1.5	
Active power ⊙**U***/⊙**I*** 5 mA … 5 A range AC, 10 mA … 8 A range DC	0.8 + 1.0	f/1 MHz · 2.0 + f/1 MHz · 1.8	-
Active power ⊙**U***/⊙**I*** 10 A … 32 A range AC, 15 A … 32 A range DC	-		
Active power ⊙**U***/⊙**I**$_{\text{Sensor}}$	0.72 + 0.9	f/1 MHz · 1.8 + f/1 MHz · 1.5	
Active power ⊙**U**$_{\text{Sensor}}$/⊙**I*** 5 mA … 5 A range AC, 10 mA … 8 A range DC	0.72 + 0.9	f/1 MHz · 1.4 + f/1 MHz · 1.8	-
Active power ⊙**U**$_{\text{Sensor}}$/⊙**I*** 10 A … 32 A range AC, 15 A … 32 A range DC	-		
Active power ⊙**U**$_{\text{Sensor}}$/⊙**I**$_{\text{Sensor}}$	0.64 + 0.8	f/1 MHz · 1.1 + f/1 MHz · 1.5	

Common-mode rejection ratio (CMRR)

Common-mode rejection ratio is measured by using a sinusoidal voltage of 100 V against earth. For the voltage ranges the signal is applied to the short circuit jacks ⊙U and ⊙U* resp. ⊙U and ⊙U$_\text{Sensor}$. For the current ranges the signal is applied to the short circuit jacks ⊙I and ⊙I$_\text{Sensor}$ resp. to the ⊙I jack.

Common-mode frequency	range AC range DC Signal-Filter	U*, 3 V 5 V	U*, 1000 V 1500 V	Usensor, 4 V 10 V	I*, 5 mA 10 mA	I*, 32 A 32 A	Isensor, 4 A 10 A
53 Hz	15 kHz	>120 dB	>120 dB	>120 dB	>160 dB	>150 dB	>120 dB
53 Hz	off	>120 dB	>120 dB	>120 dB	>160 dB	>150 dB	>120 dB
100 kHz	15 kHz	>120 dB	>120 dB	>115 dB	>160 dB	>150 dB	>115 dB
100 kHz	off	>75 dB	>75 dB	>110 dB	>100 dB	>150 dB	>110 dB

4.6.1 Voltage channel L60-CH-S2

Measurement ranges for the jacks ⊙U*/⊙U

Nominal range AC / V	3	6	12.5	25	60	130	250	400	600	1000
Nominal range DC / V	5	10	20	45	90	180	360	720	1000	1500
Max. TRMS value / V	5.5	11	22	47	95	190	370	730	1010 [x]	1510 [x]
Max. peak value / V	6	12	25	50	100	200	400	800	1600	3200
Input impedance	2.69 MΩ ±1 % ‖ about 4 pF									
Overload capability	$U_\text{AC} = 1000\,\text{V} + 10\,\%$ continuously$\quad U_\text{AC} = 1500\,\text{V}$ for 1 s$\quad U_\text{DC} = 1500\,\text{V} + 10\,\%$ continuously$\quad U = 2500\,\text{V}$ for 20 ms, transient									
Capacity against earth	about 90 pF									

[x] See specification of overload capability, max. measureable RMS values, max. Isolation voltage and the warnings at the beginning of this section.

Measurement ranges for the jacks ⊙U$_\text{Sensor}$/⊙U

Nominal range AC / V	0.03	0.06	0.12	0.25	0.5	1	2	4
Nominal range DC / V	0.08	0.15	0.3	0.6	1.2	2.5	5	10
Max. TRMS value / V	0.085	0.16	0.32	0.65	1.3	2.75	5.5	11
Max. peak value / V	0.0977	0.1953	0.3906	0.7813	1.563	3.125	6.25	12.5
Input impedance	99.8 kΩ ±1 % ‖ about 34 pF							
Overload capability	100 V continuously, 250 V for 1 s							
Capacity against earth	about 90 pF							

4.6.2 Current channel L60-CH-S2

Measurement ranges for the jacks ⊙I*/⊙I

Nominal range AC / A	0.005	0.01	0.02	0.04	0.08	0.15	0.3	0.6
Nominal range DC / A	0.01	0.02	0.04	0.08	0.15	0.3	0.6	1.2
Max. TRMS value / A	0.011	0.021	0.042	0.084	0.16	0.32	0.64	1.25
Max. peak value / A	0.014	0.028	0.056	0.112	0.224	0.469	0.938	1.875
Input impedance	about $2.2\,\Omega + 200\,$nH		about $600\,$m$\Omega + 200\,$nH			about $80\,$m$\Omega + 200\,$nH		
Overload capability	LMG operating, 10 A continuously, 150 A for 10 ms							
Capacity against earth	about 90 pF							

Nominal range AC / A	1.2	2.5	5	10	20	32
Nominal range DC / A	2.5	5	8	15	22	32
Max. TRMS value / A	2.6	5.2	8.4	15.5	22.5	32.5 [x]
Max. peak value / A	3.75	7.5	15	30	60	120
Input impedance	about $20\,$m$\Omega + 200\,$nH			about $10\,$m$\Omega + 200\,$nH		
Overload capability	LMG operating, 32 A continuously, 150 A for 10 ms					
Capacity against earth	about 90 pF					

[x] See specification of overload capability, max. measureable RMS values, max. Isolation voltage and the warnings at the beginning of this section.

Measurement ranges for the jacks ⊙I$_{Sensor}$/⊙I

Nominal range AC / V	0.03	0.06	0.12	0.25	0.5	1	2	4
Nominal range DC / V	0.08	0.15	0.3	0.6	1.2	2.5	5	10
Max. TRMS value / V	0.085	0.16	0.32	0.65	1.3	2.75	5.5	11
Max. peak value / V	0.0977	0.1953	0.3906	0.7813	1.563	3.125	6.25	12.5
Input impedance	$99.8\,$k$\Omega \pm 1\,\%$ || about $34\,$pF							
Overload capability	100 V continuously, 250 V for 1 s							
Capacity against earth	about 90 pF							

4.7 Power measurement channel L60-CH-A1 and L60-CH-A2 and L60-CH-A2.1

This channel is rated according to EN 61010-1 and EN 61010-2-030 for measuring voltages from ⊙U* to ⊙U and for insulation voltages from ⊙U*, ⊙U, ⊙U$_{Sensor}$, ⊙I*, ⊙I, ⊙I$_{Sensor}$ to protective earth PE and from ⊙U to ⊙I up to:

L60-CH-A1 and L60-CH-A2

- $U_{AC} = U_{DC} = 600\,$V, measurement category CAT IV

- $U_{AC} = U_{DC} = 1000\,$V, measurement category CAT III

- other circuits:
  working voltage rms 1000 V,
  working voltage peak 3200 V and
  transient overvoltage 3000 V

L60-CH-A2.1

- $U_{AC} = U_{DC} = 300\,$V, measurement category CAT IV

- $U_{AC} = U_{DC} = 600\,$V, measurement category CAT III

- $U_{AC} = U_{DC} = 1000\,$V, measurement category CAT II

- other circuits:
  working voltage rms 1000 V,
  working voltage peak 3200 V and
  transient overvoltage 1600 V

The standard accessory safety laboratory leads with integrated fuses have to be used!

The RMS values specified for the ranges are the maximum RMS values which could be measured theoretically. This values are not allowed to apply if safe operation or any other safety values are not guaranteed.

Support of *DualPath* (dual bandwidth) mode

This channel supports the *DualPath* mode, see Bandwidth [6.6→110].

Sample rate and bandwidth of the AD converter

Table 4.9: L60-CH-A: Sample rate and bandwidth of the AD converter

Wide bandwidth converter	
Sample rate	$1.\overline{21}$ MS/s
Resolution	18 bit
Bandwidth, analog anti-aliasing filter disabled	10 MHz
Bandwidth, analog anti-aliasing filter activated	about 150 kHz
Narrow bandwidth converter	
Sample rate	$151.\overline{51}$ kS/s
Resolution	18 bit
Bandwidth, analog anti-aliasing filter always activated	about 15 kHz

Signal filter

This filters are analog filters in the signal path with a bandwidth of about 15 kHz and 150 kHz. To ensure the anti-aliasing function for the corresponding analog-digital converters (ADC) with an attenuation of >50 dB at higher frequencies than half of sampling rate the specified design parameters were chosen.

Example: The 15 kHz filter has an attenuation of -14.7440 % at a signal frequency $f = 10$ kHz. Those -14.7440 % are afflicted with an uncertainty of $\pm0.012 = \pm1.2$ %. Consequently the measuring signal is damped by $-(14.7 \pm 1.2)$ %.

Table 4.10: L60-CH-A: Characteristics of 15 kHz filter

Analog anti-aliasing filter 15 kHz (narrowband)			
Design parameter: Bessel filter, 5th order, cutoff frequency 14.5 kHz			
Frequency / Hz	Filter attenuation in % (nominal)	Worst Case uncertainty of the filter attenuation	Filter attenuation in dB (nominal)
0	0.0004	$\pm50 \cdot 10^{-6}$	0.0000
10	0.0004	$\pm50 \cdot 10^{-6}$	0.0000
20	0.0003	$\pm50 \cdot 10^{-6}$	0.0000
53	0.0000	$\pm50 \cdot 10^{-6}$	0.0000
100	-0.0012	$\pm50 \cdot 10^{-6}$	-0.0001
200	-0.0058	$\pm50 \cdot 10^{-6}$	-0.0005
500	-0.0385	$\pm50 \cdot 10^{-6}$	-0.0033
1000	-0.1552	$\pm160 \cdot 10^{-6}$	-0.0135
2000	-0.6210	$\pm660 \cdot 10^{-6}$	-0.0541
5000	-3.8385	$\pm3.9 \cdot 10^{-3}$	-0.3400
10000	-14.7440	$\pm12 \cdot 10^{-3}$	-1.3855
15000	-31.1128	$\pm18 \cdot 10^{-3}$	-3.2372
75000	-99.7125	—	-50.8267

Table 4.11: L60-CH-A: Characteristics of 150 kHz filter

Analog anti-aliasing filter 150 kHz (wideband)			
Design parameter: Bessel filter, 7th order, cutoff frequency 145 kHz			
Frequency / Hz	Filter attenuation in % (nominal)	Worst Case uncertainty of the filter attenuation	Filter attenuation in dB (nominal)
0	0.0000	$\pm 50 \cdot 10^{-6}$	0.0000
10	0.0000	$\pm 50 \cdot 10^{-6}$	0.0000
20	0.0000	$\pm 50 \cdot 10^{-6}$	0.0000
53	0.0000	$\pm 50 \cdot 10^{-6}$	0.0000
100	−0.0000	$\pm 50 \cdot 10^{-6}$	−0.0000
200	−0.0001	$\pm 50 \cdot 10^{-6}$	−0.0000
500	−0.0004	$\pm 50 \cdot 10^{-6}$	−0.0000
1000	−0.0016	$\pm 100 \cdot 10^{-6}$	−0.0001
2000	−0.0064	$\pm 100 \cdot 10^{-6}$	−0.0006
5000	−0.0398	$\pm 100 \cdot 10^{-6}$	−0.0035
10000	−0.1593	$\pm 350 \cdot 10^{-6}$	−0.0138
20000	−0.6359	$\pm 1.5 \cdot 10^{-3}$	−0.0554
50000	−3.9203	$\pm 8.7 \cdot 10^{-3}$	−0.3474
100000	−14.9399	$\pm 27.3 \cdot 10^{-3}$	−1.4055
150000	−31.0563	$\pm 42 \cdot 10^{-3}$	−3.2301
600000	−99.7028	—	−50.5399

Harmonics

The special capabilities of this channel for the harmonics measurement are described in INTERHARMONICS = 0 [6.8.1→115].

General Accuracy Specifications   For the amplitude, the measurement uncertainties according to tables 4.13 to 4.16 apply to each individual harmonic or intermediate harmonic at the respective frequency as if it had been measured on its own. In addition, the uncertainties (not the damping) of the filter used must be taken into account (table 4.10 and 4.11). The following accuracy specification (4.2) applies to the phase angle if the amplitude of the respective frequency component is $> 0.15\,\%$ of the nominal value of the measuring range and the frequency of the fundamental frequency is in the range from 1 Hz to 10 kHz. h is the order of the harmonics and f_h the according signal frequency.

$$\pm \left(0.05° + (0.005° \cdot h) + \left(0.05° \cdot \frac{f_\mathrm{h}}{1\,\mathrm{kHz}} \right) \right) \tag{4.2}$$

A calculated example can be found in 6.16.3.

CE Harmonics

CE Harmonische are the case when INTERHARM = 9 and fundamental frequency between 45 Hz and 65 Hz. For those is valid:

INTERHARM
[9.10.288→353]

1. Relative deviation between f1 and frequency fsyn, to which the sampling rate is synchronized is $<0.03\,\%$ of f1 under steady-state conditions.

2. The attenuation of the anti-aliasing filter is >50 dB at higher frequencies than half of sampling rate.

3. The instrument complies with class I according IEC 61000-4-7 Ed. 2.0 and EN 61000-4-7 Ed. 2.0.

4. Accuracy specification according IEC 61000-4-7 Ed. 2.0 and EN 61000-4-7 Ed. 2.0 with m = measuring value and nom = nominal value of the range:

Table 4.12: L60-CH-A: Accuracy harmonics

voltage measurement	Um ≥ 1 % Unom:	± 5 % Um
	Um < 1 % Unom:	± 0.05 % Unom
current measurement	Im ≥ 3 % Inom:	± 5 % Im
	Im < 3 % Inom:	± 0.15 % Inom

The smaller value of the uncertainty in table 4.12 and in the general accuracy specification applies to the specification of the uncertainty of the amplitude.

CE Flicker

According to EN 61000-4-15:2011 resp. IEC 61000-4-15:2010 this is a class F1 flicker meter.

Flickermeter: ±5 % according to EN 61000-4-15, 0.3<k<5.4

d-meter: ±5 % according to EN 61000-4-15

Accuracy specifications

The accuracy specifications are valid under the following conditions:

1. Sinusoidal voltage and current

2. Ambient temperature (23±3) °C, no additional heating or cooling (e.g. by sunlight or current of air)

3. The temperature effect is $\pm(0.01\,\%\,\text{of reading})/\,\text{K}$ in the range of 5 °C … 20 °C and 26 °C … 40 °C

4. warm up time 1 h

5. The upper range value is defined by the maximum peak value, see VOLTAGE CHANNEL L60-CH-A1 AND L60-CH-A2 AND L60-CH-A2.1 [4.7.1→61] and CURRENT CHANNEL L60-CH-A1 AND L60-CH-A2 AND L60-CH-A2.1 [4.7.2→61]. All other values are calculated from voltage, current and active power. The tolerances results from the functional relationship, see MEASUREMENT UNCERTAINTY OF MEASURED AND CALCULATED VALUES [6.16→135].

6. Specified are the direct measured and unfiltered measuring values. For the specification of the filtered measuring values, see SIGNAL FILTER [4.7→56].

7. The upper power range value is the product of upper voltage range value and upper current range value. The specification is valid for power factor 0 … 1.

8. Voltage and current are higher than 10 % of nominal range and not higher than the maximum TRMS value of the range. Voltage peak and current peak are not higher than range peak.

9. All specifications above 1 MHz are design values.

10. The instrument should be calibrated every 12 month, last adjustment not older than 12 months and adjustment was done at 23 °C.

The values are to be interpreted as ±(% of reading + % of upper range value).

Table 4.13: L60-CH-A1 and L60-CH-A2 and L60-CH-A2.1: Accuracy DC

Frequency	DC [c]	DC
Voltage $\odot$**U**$*$	$0.02 + 0.06$	$0.02 + 0.08$
Voltage $\odot$**U**$_{Sensor}$	$0.02 + 0.06$	$0.02 + 0.08$
Current $\odot$**I**$*$ 5 mA ... 5 A range	$0.02 + 0.06$	$0.02 + 0.1$
Current $\odot$**I**$*$ 10 A ... 32 A range	-	$0.02 + 0.1$ [a]
Current $\odot$**I**$_{Sensor}$	$0.02 + 0.06$	$0.02 + 0.08$
Active power $\odot$**U**$*$/$\odot$**I**$*$ 5 mA ... 5 A range	$0.032 + 0.06$	$0.032 + 0.09$
Active power $\odot$**U**$*$/$\odot$**I**$*$ 10 A ... 32 A range	-	$0.032 + 0.09$ [b]
Active power $\odot$**U**$*$/$\odot$**I**$_{Sensor}$	$0.032 + 0.06$	$0.032 + 0.08$
Active power $\odot$**U**$_{Sensor}$/$\odot$**I**$*$ 5 mA ... 5 A range	$0.032 + 0.06$	$0.032 + 0.09$
Active power $\odot$**U**$_{Sensor}$/$\odot$**I**$*$ 10 A ... 32 A range	-	$0.032 + 0.09$ [b]
Active power $\odot$**U**$_{Sensor}$/$\odot$**I**$_{Sensor}$	$0.032 + 0.06$	$0.032 + 0.08$

[a] Additional accuracy specification in the 10 A ... 32 A range: $\pm \frac{80\,\mu A}{A^2} \cdot I_{trms}{}^2$

[b] Additional accuracy specification in the 10 A ... 32 A range: $\pm \frac{80\,\mu A}{A^2} \cdot I_{trms}{}^2 \cdot U_{trms}$

[c] Accuracy specification after non persistent zero adjustment (see ZERO ADJUSTMENT [6.18→139]), temperature change after zero adjustment max. $\pm 1\,°C$

Table 4.14: L60-CH-A1 and L60-CH-A2 and L60-CH-A2.1: Accuracy 0.05 Hz ... 10 kHz

Frequency	0.05 Hz ... 45 Hz 65 Hz ... 3 kHz	45 Hz ... 65 Hz	3 kHz ... 10 kHz
Voltage $\odot$**U**$*$	$0.015 + 0.03$	$0.01 + 0.02$	$0.03 + 0.06$
Voltage $\odot$**U**$_{Sensor}$	$0.015 + 0.03$	$0.01 + 0.02$	$0.03 + 0.06$
Current $\odot$**I**$*$ 5 mA ... 5 A range	$0.015 + 0.03$	$0.01 + 0.02$	$0.03 + 0.06$
Current $\odot$**I**$*$ 10 A ... 32 A range	$0.015 + 0.03$ [a]	$0.01 + 0.02$ [a]	$0.1 + 0.2$ [a]
Current $\odot$**I**$_{Sensor}$	$0.015 + 0.03$	$0.01 + 0.02$	$0.03 + 0.06$
Active power $\odot$**U**$*$/$\odot$**I**$*$ 5 mA ... 5 A range	$0.024 + 0.03$	$0.015 + 0.01$	$0.048 + 0.06$
Active power $\odot$**U**$*$/$\odot$**I**$*$ 10 A ... 32 A range	$0.024 + 0.03$ [b]	$0.015 + 0.01$ [b]	$0.104 + 0.13$ [b]
Active power $\odot$**U**$*$/$\odot$**I**$_{Sensor}$	$0.024 + 0.03$	$0.015 + 0.01$	$0.048 + 0.06$
Active power $\odot$**U**$_{Sensor}$/$\odot$**I**$*$ 5 mA ... 5 A range	$0.024 + 0.03$	$0.015 + 0.01$	$0.048 + 0.06$
Active power $\odot$**U**$_{Sensor}$/$\odot$**I**$*$ 10 A ... 32 A range	$0.024 + 0.03$ [b]	$0.015 + 0.01$ [b]	$0.104 + 0.13$ [b]
Active power $\odot$**U**$_{Sensor}$/$\odot$**I**$_{Sensor}$	$0.024 + 0.03$	$0.015 + 0.01$	$0.048 + 0.06$

[a] Additional accuracy specification in the 10 A ... 32 A range: $\pm \frac{80\,\mu A}{A^2} \cdot I_{trms}{}^2$

[b] Additional accuracy specification in the 10 A ... 32 A range: $\pm \frac{80\,\mu A}{A^2} \cdot I_{trms}{}^2 \cdot U_{trms}$

Table 4.15: L60-CH-A1 and L60-CH-A2 and L60-CH-A2.1: Accuracy 10 kHz … 500 kHz

Frequency	10 kHz … 50 kHz	50 kHz … 100 kHz	100 kHz … 500 kHz
Voltage ◎U*	0.2 + 0.4		0.5 + 1.0
Voltage ◎U$_{Sensor}$	0.2 + 0.4		0.4 + 0.8
Current ◎I* 5 mA … 5 A range	0.2 + 0.4		0.5 + 1.0
Current ◎I* 10 A … 32 A range	0.3 + 0.6 [a]	f/100 kHz · 0.8 + f/100 kHz · 1.2 [a]	
Current ◎I$_{Sensor}$	0.2 + 0.4		0.4 + 0.8
Active power ◎U*/◎I* 5 mA … 5 A range	0.32 + 0.4		0.8 + 1.0
Active power ◎U*/◎I* 10 A … 32 A range	0.4 + 0.5 [b]	f/100 kHz · 0.8 + f/100 kHz · 0.8 [b]	f/100 kHz · 1.0 + f/100 kHz · 1.1 [c]
Active power ◎U*/◎I$_{Sensor}$	0.32 + 0.4		0.72 + 0.9
Active power ◎U$_{Sensor}$/◎I* 5 mA … 5 A range	0.32 + 0.4		0.72 + 0.9
Active power ◎U$_{Sensor}$/◎I* 10 A … 32 A range	0.4 + 0.5 [b]	f/100 kHz · 0.8 + f/100 kHz · 0.8 [b]	f/100 kHz · 1.0 + f/100 kHz · 1.0 [c]
Active power ◎U$_{Sensor}$/◎I$_{Sensor}$	0.32 + 0.4		0.64 + 0.8

[a] Additional accuracy specification in the 10 A … 32 A range: $\pm \frac{80\,\mu A}{A^2} \cdot I_{trms}^2$

[b] Additional accuracy specification in the 10 A … 32 A range: $\pm \frac{80\,\mu A}{A^2} \cdot I_{trms}^2 \cdot U_{trms}$

Table 4.16: L60-CH-A1 and L60-CH-A2 and L60-CH-A2.1: Accuracy 500 kHz … 10 MHz

Frequency	500 kHz … 1 MHz	1 MHz … 2 MHz	2 MHz … 10 MHz
Voltage ◎U*	0.5 + 1.0	f/1 MHz · 1.5 + f/1 MHz · 1.5	
Voltage ◎U$_{Sensor}$	0.4 + 0.8	f/1 MHz · 0.7 + f/1 MHz · 1.5	
Current ◎I* 5 mA … 5 A range	0.5 + 1.0	f/1 MHz · 1.0 + f/1 MHz · 2.0	-
Current ◎I* 10 A … 32 A range	-		
Current ◎I$_{Sensor}$	0.4 + 0.8	f/1 MHz · 0.7 + f/1 MHz · 1.5	
Active power ◎U*/◎I* 5 mA … 5 A range	0.8 + 1.0	f/1 MHz · 2.0 + f/1 MHz · 1.8	-
Active power ◎U*/◎I* 10 A … 32 A range	-		
Active power ◎U*/◎I$_{Sensor}$	0.72 + 0.9	f/1 MHz · 1.8 + f/1 MHz · 1.5	
Active power ◎U$_{Sensor}$/◎I* 5 mA … 5 A range	0.72 + 0.9	f/1 MHz · 1.4 + f/1 MHz · 1.8	-
Active power ◎U$_{Sensor}$/◎I* 10 A … 32 A range	-		
Active power ◎U$_{Sensor}$/◎I$_{Sensor}$	0.64 + 0.8	f/1 MHz · 1.1 + f/1 MHz · 1.5	

Common-mode rejection ratio (CMRR)

Common-mode rejection ratio is measured by using a sinusoidal voltage of 100 V against earth. For the voltage ranges the signal is applied to the short circuit jacks ◎U and ◎U* resp. ◎U and ◎U$_{Sensor}$. For the current ranges the signal is applied to the short circuit jacks ◎I and ◎I$_{Sensor}$ resp. to the ◎I jack.

Common-mode frequency	Signal-Filter	U*, 3 V range	U*, 1000 V range	Usensor, 4 V range	I*, 5 mA range	I*, 32 A range	Isensor, 4 A range
53 Hz	15 kHz	>120 dB	>120 dB	>120 dB	>160 dB	>150 dB	>120 dB
53 Hz	off	>120 dB	>120 dB	>120 dB	>160 dB	>150 dB	>120 dB
100 kHz	15 kHz	>120 dB	>120 dB	>115 dB	>160 dB	>150 dB	>115 dB
100 kHz	off	>75 dB	>75 dB	>110 dB	>100 dB	>150 dB	>110 dB

4.7.1 Voltage channel L60-CH-A1 and L60-CH-A2 and L60-CH-A2.1

Measurement ranges for the jacks ⊙U*/⊙U

Nominal range / V	3	6	12.5	25	60	130	250	400	600	1000
Max. TRMS value / V	3.3	6.6	13.8	27.5	66	136	270	440	660	1000
Max. peak value / V	6	12	25	50	100	200	400	800	1600	3200
Input impedance L60-CH-A1	about 4.59 MΩ, 3 pF									
Input impedance L60-CH-A2 and L60-CH-A2.1	2.69 MΩ ±1 %, 4 pF									
Overload capability	1000 V + 10 % continuously 1500 V for 1 s 2500 V for 20 ms, transient									
Capacity against earth	about 90 pF									

Measurement ranges for the jacks ⊙U_Sensor/⊙U

Nominal range / V	0.030	0.060	0.120	0.250	0.500	1	2	4
Max. TRMS value / V	0.033	0.066	0.132	0.275	0.550	1.1	2.2	4.4
Max. peak value / V	0.0977	0.1953	0.3906	0.7813	1.563	3.125	6.25	12.5
Input impedance	99.8 kΩ ±1 %, 34 pF							
Overload capability	100 V continuously, 250 V for 1 s							
Capacity against earth	about 90 pF							

4.7.2 Current channel L60-CH-A1 and L60-CH-A2 and L60-CH-A2.1

Measurement ranges for the jacks ⊙I*/⊙I

Nominal value / A	0.005	0.010	0.020	0.040	0.080	0.150	0.300	0.600
Max. TRMS value / A	0.0055	0.011	0.022	0.044	0.088	0.165	0.330	0.660
Max. peak value / A	0.014	0.028	0.056	0.112	0.224	0.469	0.938	1.875
Input impedance	about 2.2 Ω		about 600 mΩ			about 80 mΩ		
Overload capability	LMG operating, 10 A continuously, 150 A for 10 ms							
Capacity against earth	about 90 pF							

Nominal value / A	1.2	2.5	5	10	20	32
Max. TRMS value / A	1.32	2.75	5.5	11	22	32
Max. peak value / A	3.75	7.5	15	30	60	120
Input impedance	about 20 mΩ			about 10 mΩ		
Overload capability	LMG operating, 32 A continuously, 150 A for 10 ms					
Capacity against earth	90 pF					

Measurement ranges for the jacks ⊙I_Sensor/⊙I

Nominal range / V	0.030	0.060	0.120	0.250	0.500	1	2	4
Max. TRMS value / V	0.033	0.066	0.132	0.275	0.550	1.1	2.2	4.4
Max. peak value / V	0.0977	0.1953	0.3906	0.7813	1.563	3.125	6.25	12.5
Input impedance	99.8 kΩ ±1 %, 34 pF							
Overload capability	100 V continuously, 250 V for 1 s							
Capacity against earth	about 90 pF							

4.8 Power measurement channel L60-CH-B1 and L60-CH-B2 and L60-CH-B2.1

This channel is rated according to EN 61010-1 and EN 61010-2-030 for measuring voltages from ⊙**U***
to ⊙**U** and for insulation voltages from ⊙**U***, ⊙**U**, ⊙**U_{Sensor}**, ⊙**I***, ⊙**I**, ⊙**I_{Sensor}** to protective earth PE
and from ⊙**U** to ⊙**I** up to:

L60-CH-B1 and L60-CH-B2

- $U_{AC} = U_{DC} = 600\,V$, measurement category CAT IV

- $U_{AC} = U_{DC} = 1000\,V$, measurement category CAT III

- other circuits:
  working voltage rms 1000 V,
  working voltage peak 3200 V and
  transient overvoltage 3000 V

L60-CH-B2.1

- $U_{AC} = U_{DC} = 300\,V$, measurement category CAT IV

- $U_{AC} = U_{DC} = 600\,V$, measurement category CAT III

- $U_{AC} = U_{DC} = 1000\,V$, measurement category CAT II

- other circuits:
  working voltage rms 1000 V,
  working voltage peak 3200 V and
  transient overvoltage 1600 V

The standard accessory safety laboratory leads with integrated fuses have to be
used!

The RMS values specified for the ranges are the maximum RMS values which could be measured
theoretically. This values are not allowed to apply if safe operation or any other safety values are not
guaranteed.

Support of *DualPath* (dual bandwidth) mode

This channel supports the *DualPath* mode, see BANDWIDTH
[6.6→110].

Sample rate and bandwidth of the AD converter

Table 4.17: L60-CH-B: Sample rate and bandwidth of the AD converter

Wide bandwidth converter	
Sample rate	$151.\overline{51}\,kS/s$
Resolution	16 bit
Bandwidth	500 kHz
Narrow bandwidth converter	
Sample rate	$151.\overline{51}\,kS/s$
Resolution	16 bit
Bandwidth, analog anti-aliasing filter always activated	about 15 kHz

Signal filter

This filter is an analog filter in the signal path with a bandwidth of about 15 kHz. To
ensure the anti-aliasing function for the analog-digital converter (ADC) with an attenuation of
$>50\,dB$ at higher frequencies than half of sampling rate the specified design parameters were
chosen.

Example: The 15 kHz filter has an attenuation of $-14.7440\,\%$ at a signal frequency $f = 10\,kHz$. Those
$-14.7440\,\%$ are afflicted with an uncertainty of $\pm 0.012 = \pm 1.2\,\%$. Consequently the measuring signal
is damped by $-(14.7 \pm 1.2)\,\%$.

Table 4.18: L60-CH-B: Characteristics of 15 kHz filter

Analog anti-aliasing filter 15 kHz (narrowband)			
Design parameter: Bessel filter, 5th order, cutoff frequency 14.5 kHz			
Frequency / Hz	Filter attenuation in % (nominal)	Worst Case uncertainty of the filter attenuation	Filter attenuation in dB (nominal)
0	0.0004	$\pm 50 \cdot 10^{-6}$	0.0000
10	0.0004	$\pm 50 \cdot 10^{-6}$	0.0000
20	0.0003	$\pm 50 \cdot 10^{-6}$	0.0000
53	0.0000	$\pm 50 \cdot 10^{-6}$	0.0000
100	-0.0012	$\pm 50 \cdot 10^{-6}$	-0.0001
200	-0.0058	$\pm 50 \cdot 10^{-6}$	-0.0005
500	-0.0385	$\pm 50 \cdot 10^{-6}$	-0.0033
1000	-0.1552	$\pm 160 \cdot 10^{-6}$	-0.0135
2000	-0.6210	$\pm 660 \cdot 10^{-6}$	-0.0541
5000	-3.8385	$\pm 3.9 \cdot 10^{-3}$	-0.3400
10000	-14.7440	$\pm 12 \cdot 10^{-3}$	-1.3855
15000	-31.1128	$\pm 18 \cdot 10^{-3}$	-3.2372
75000	-99.7125	—	-50.8267

Harmonics

The special capabilities of this channel for the harmonics measurement are in principle described in INTERHARMONICS = 0 [6.8.1→115] but with following tables:

Processing single path, auto filter off

#	Filter setting	Harmonics setting	Fundamental frequency	Gaps	Max. Order or frequency
1	<15 kHz	no effect	2 - 7.5 Hz	no	2000th or filter frequency
2	<15 kHz	no effect	7.5 - 33 Hz	no	filter frequency
3	<15 kHz	no effect	33 Hz - 7.5 kHz	yes	filter frequency
4	15 kHz	no effect	2 - 7.5 Hz	no	2000th
5	15 kHz	no effect	7.5 - 33 Hz	no	15 kHz
6	15 kHz	no effect	33 Hz - 7.5 kHz	yes	15 kHz
9	no filter	Custom*	20 - 37.5 Hz	yes	2000th
10	no filter	Custom*	37.5 - 7.5 kHz	yes	75 kHz
11	no filter	Auto	—	—	—

*Aliasing is not prevented by the instrument, see ANTI-ALIASING [6.6.2→111]

The combination #11 (no filter, harmonics set to auto) is not valid, because without filter there can be no guarantee for no aliasing. So there no harmonics will be calculated.

Processing single path, auto filter on, auto harmonics

#	Filter setting	Harmonics setting	Fundamental frequency	Gaps	Max. Order or frequency
12	Auto on	Auto	2 - 7.5 Hz	no	2000th
13	Auto on	Auto	7.5 - 33 Hz	no	15 kHz
14	Auto on	Auto	33 - 7.5 kHz	yes	15 kHz

Processing single path, auto filter on, custom harmonics

#	Filter setting	Harmonics setting	Fundamental frequency	Gaps	Max. Order or frequency
16	Auto on	Custom	2 - 7.5 Hz	no	2000th
17	Auto on	Custom	7.5 - 33 Hz	no	15 kHz
18	Auto on	Custom[*]	33 - 7.5 kHz	yes	75 kHz

[*]Aliasing is not prevented by the instrument, see ANTI-ALIASING [6.6.2→111]

Processing dual path

#	Filter setting	Harmonics setting	Fundamental frequency	Gaps	Max. Order or frequency
20	<15 kHz	no effect	2 - 7.5 Hz	no	2000th or filter frequency
21	<15 kHz	no effect	7.5 - 33 Hz	no	filter frequency
22	<15 kHz	no effect	33 Hz - 7.5 kHz	yes	filter frequency
23	15 kHz or auto filter off	no effect	2 - 7.5 Hz	no	2000th
24	15 kHz or auto filter off	no effect	7.5 - 33 Hz	no	15 kHz
25	15 kHz or auto filter off	no effect	33 - 7.5 kHz	yes	15 kHz

With dual path processing the harmonics are just computed for the small band values. There is no calculation for wide band values.

General Accuracy Specifications   For the amplitude, the measurement uncertainties according to tables 4.20 and 4.21 apply to each individual harmonic or intermediate harmonic at the respective frequency as if it had been measured on its own. In addition, the uncertainties (not the damping) of the filter must be taken into account (table 4.18). The following accuracy specification (4.3) applies to the phase angle if the amplitude of the respective frequency component is $> 0.15\%$ of the nominal value of the measuring range and the frequency of the fundamental frequency is in the range from 1 Hz to 10 kHz. h is the order of the harmonics and f_h the according signal frequency.

$$\pm \left(0.05° + (0.005° \cdot h) + \left(0.05° \cdot \frac{f_\mathrm{h}}{1\,\mathrm{kHz}} \right) \right) \tag{4.3}$$

A calculated example can be found in 6.16.3.

CE Harmonics

CE Harmonische are the case when INTERHARM = 9 and fundamental frequency between 45 Hz and 65 Hz. For those is valid:

1. Relative deviation between f1 and frequency fsyn , to which the sampling rate is synchronised is <0.03 % of f1 under steady-state conditions.

2. The attenuation of the anti-aliasing filter is >50 dB at higher frequencies than half of sampling rate.

3. The instrument complies with class I according IEC 61000-4-7 Ed. 2.0 and EN 61000-4-7 Ed. 2.0.

4. Accuracy specifications according IEC 61000-4-7 Ed. 2.0 and EN 61000-4-7 Ed. 2.0 with m = measuring value and nom = nominal value of the range:

Table 4.19: L60-CH-B: Accuracy harmonics

voltage measurement	Um ≥ 1 % Unom:	± 5 % Um
	Um < 1 % Unom:	± 0.05 % Unom
current measurement	Im ≥ 3 % Inom:	± 5 % Im
	Im < 3 % Inom:	± 0.15 % Inom

The smaller value of the uncertainty in table 4.19 and in the general accuracy specification applies to the specification of the uncertainty of the amplitude.

CE Flicker

According to EN 61000-4-15:2011 resp. IEC 61000-4-15:2010 this is a class F1 flicker meter.

Flickermeter: $\pm 5\,\%$ according to EN 61000-4-15, $0.3 < k < 5.4$

d-meter: $\pm 5\,\%$ according to EN 61000-4-15

Accuracy specifications

The accuracy specifications are valid under the following conditions:

1. Sinusoidal voltage and current

2. Ambient temperature $(23\pm3)\,^{\circ}\mathrm{C}$, no additional heating or cooling (e.g. by sunlight or current of air)

3. The temperature effect is $\pm(0.01\,\%\,\text{of reading})/\,\mathrm{K}$ in the range of $5\,^{\circ}\mathrm{C}\ldots 20\,^{\circ}\mathrm{C}$ and $26\,^{\circ}\mathrm{C}\ldots 40\,^{\circ}\mathrm{C}$

4. warm up time $1\,\mathrm{h}$

5. The upper range value is defined by the maximum peak value, see VOLTAGE CHANNEL L60-CH-B1 AND L60-CH-B2 AND L60-CH-B2.1 [4.8.1→66] and CURRENT CHANNEL L60-CH-B1 AND L60-CH-B2 AND L60-CH-B2.1 [4.8.2→66]. All other values are calculated from voltage, current and active power. The tolerances results from the functional relationship, see MEASUREMENT UNCERTAINTY OF MEASURED AND CALCULATED VALUES [6.16→135].

6. Specified are the direct measured and unfiltered measuring values. For the specification of the filtered measuring values, see SIGNAL FILTER [4.8→62].

7. The upper power range value is the product of upper voltage range value and upper current range value. The specification is valid for power factor $0\ldots 1$.

8. Voltage and current are higher than $10\,\%$ of nominal range and not higher than the maximum TRMS value of the range. Voltage peak and current peak are not higher than range peak.

9. The instrument should be calibrated every $12\,\text{month}$, last adjustment not older than $12\,\text{months}$ and adjustment was done at $23\,^{\circ}\mathrm{C}$.

The values are to be interpreted as $\pm(\%$ of reading $+\ \%$ of upper range value).

Table 4.20: L60-CH-B1 and L60-CH-B2 and L60-CH-B2.1: Accuracy DC ... 5 kHz

Frequency	DC	0.05 Hz ... 45 Hz 65 Hz ... 1 kHz	45 Hz ... 65 Hz	1 kHz ... 5 kHz
Voltage ⊚U*	0.1 + 0.1	0.1 + 0.1	0.03 + 0.03	0.2 + 0.2
Current ⊚I* 5 mA ... 5 A range Current ⊚I$_{\textbf{Sensor}}$	0.1 + 0.1	0.1 + 0.1	0.03 + 0.03	0.2 + 0.2
Current ⊚I* 10 A ... 32 A range	0.1 + 0.1[a]	0.1 + 0.1[a]	0.03 + 0.03[a]	0.2 + 0.2[a]
Active power ⊚U*/⊚I* 5 mA ... 5 A range Active power ⊚U*/⊚I$_{\textbf{Sensor}}$	0.16 + 0.1	0.16 + 0.1	0.05 + 0.02	0.32 + 0.2
Active power ⊚U*/⊚I* 10 A ... 32 A range	0.16 + 0.1[b]	0.16 + 0.1[b]	0.05 + 0.02[b]	0.32 + 0.2[b]

[a] Additional accuracy specification in the 10 A ... 32 A range: $\pm\frac{80\,\mu\mathrm{A}}{\mathrm{A}^2}\cdot I_{\mathrm{trms}}{}^2$

[b] Additional accuracy specification in the 10 A ... 32 A range: $\pm\frac{80\,\mu\mathrm{A}}{\mathrm{A}^2}\cdot I_{\mathrm{trms}}{}^2\cdot U_{\mathrm{trms}}$

Table 4.21: L60-CH-B1 and L60-CH-B2 and L60-CH-B2.1: Accuracy 5 kHz ... 500 kHz

Frequency	5 kHz ... 20 kHz	20 kHz ... 100 kHz	100 kHz ... 500 kHz
Voltage ⊙U*	$0.3 + 0.4$	$0.4 + 0.8$	f/100 kHz · 0.8 + f/100 kHz · 1.2
Current ⊙I* 5 mA ... 5 A range Current ⊙I$_{Sensor}$	$0.3 + 0.4$	$0.4 + 0.8$	f/100 kHz · 0.8 + f/100 kHz · 1.2
Current ⊙I* 10 A ... 32 A range	$0.6 + 1.2^a$	$1.5 + 1.5^a$	f/100 kHz · 2.0 + f/100 kHz · 2.0[a]
Active power ⊙U*/⊙I* 5 mA ... 5 A range Active power ⊙U*/⊙I$_{Sensor}$	$0.48 + 0.4$	$0.64 + 0.8$	f/100 kHz · 1.28 + f/100 kHz · 1.2
Active power ⊙U*/⊙I* 10 A ... 32 A range	$0.72 + 0.8^b$	$1.52 + 1.15^b$	f/100 kHz · 2.24 + f/100 kHz · 1.6[b]

[a] Additional accuracy specification in the 10 A ... 32 A range: $\pm \frac{80\,\mu A}{A^2} \cdot I_{trms}{}^2$

[b] Additional accuracy specification in the 10 A ... 32 A range: $\pm \frac{80\,\mu A}{A^2} \cdot I_{trms}{}^2 \cdot U_{trms}$

Common-mode rejection ratio (CMRR)

Common-mode rejection ratio is measured by using a sinusoidal voltage of 100 V against earth. For the voltage ranges the signal is applied to the short circuit jacks ⊙U and ⊙U* . For the current ranges the signal is applied to the short circuit jacks ⊙I and ⊙I$_{Sensor}$ resp. to the ⊙I jack.

Common-mode frequency	Signal-Filter	U*, 3 V range	U*, 1000 V range	I*, 5 mA range	I*, 32 A range	Isensor, 4 A range
53 Hz	15 kHz	>120 dB	>120 dB	>160 dB	>150 dB	>120 dB
53 Hz	off	>120 dB	>120 dB	>160 dB	>150 dB	>120 dB
100 kHz	15 kHz	>120 dB	>120 dB	>160 dB	>150 dB	>115 dB
100 kHz	off	>75 dB	>75 dB	>100 dB	>150 dB	>110 dB

4.8.1 Voltage channel L60-CH-B1 and L60-CH-B2 and L60-CH-B2.1

Measurement ranges for the jacks ⊙U*/⊙U

Nominal range / V	3	6	12.5	25	60	130	250	400	600	1000
Max. TRMS value / V	3.3	6.6	13.8	27.5	66	136	270	440	660	1000
Max. peak value / V	6	12	25	50	100	200	400	800	1600	3200
Input impedance L60-CH-B1	about 4.59 MΩ, 3 pF									
Input impedance L60-CH-B2 and L60-CH-B2.1	2.69 MΩ ±1 %, 4 pF									
Overload capability	1000 V + 10 % continuously 1500 V for 1 s 2500 V for 20 ms, transient									
Capacity against earth	about 90 pF									

4.8.2 Current channel L60-CH-B1 and L60-CH-B2 and L60-CH-B2.1

Measurement ranges for the jacks ⊙I*/⊙I

Nominal value / A	0.005	0.010	0.020	0.040	0.080	0.150	0.300	0.600
Max. TRMS value / A	0.0055	0.011	0.022	0.044	0.088	0.165	0.330	0.660
Max. peak value / A	0.014	0.028	0.056	0.112	0.224	0.469	0.938	1.875
Input impedance	about 2.2 Ω		about 600 mΩ			about 80 mΩ		
Overload capability	LMG operating, 10 A continuously, 150 A for 10 ms							
Capacity against earth	about 90 pF							

Nominal value / A	1.2	2.5	5	10	20	32
Max. TRMS value / A	1.32	2.75	5.5	11	22	32
Max. peak value / A	3.75	7.5	15	30	60	120
Input impedance	about $20\,\text{m}\Omega$			about $10\,\text{m}\Omega$		
Overload capability	LMG operating, 32 A continuously, 150 A for 10 ms					
Capacity against earth	90 pF					

Measurement ranges for the jacks ⊙I$_{\text{Sensor}}$/⊙I

Nominal range / V	0.030	0.060	0.120	0.250	0.500	1	2	4
Max. TRMS value / V	0.033	0.066	0.132	0.275	0.550	1.1	2.2	4.4
Max. peak value / V	0.0977	0.1953	0.3906	0.7813	1.563	3.125	6.25	12.5
Input impedance	$99.8\,\text{k}\Omega \pm 1\,\%$, 34 pF							
Overload capability	100 V continuously, 250 V for 1 s							
Capacity against earth	about 90 pF							

4.9 Power measurement channel L60-CH-C1 and L60-CH-C2 and L60-CH-C2.1

This channel is rated according to EN 61010-1 and EN 61010-2-030 for measuring voltages from ⊙**U***
to ⊙**U** and for insulation voltages from ⊙**U***, ⊙**U**, ⊙**U$_{\text{Sensor}}$**, ⊙**I***, ⊙**I**, ⊙**I$_{\text{Sensor}}$** to protective earth PE
and from ⊙**U** to ⊙**I** up to:

L60-CH-C1 and L60-CH-C2

- $U_{\text{AC}} = U_{\text{DC}} = 600\,\text{V}$, measurement category CAT IV

- $U_{\text{AC}} = U_{\text{DC}} = 1000\,\text{V}$, measurement category CAT III

- other circuits:
  working voltage rms 1000 V,
  working voltage peak 3200 V and
  transient overvoltage 3000 V

L60-CH-C2.1

- $U_{\text{AC}} = U_{\text{DC}} = 300\,\text{V}$, measurement category CAT IV

- $U_{\text{AC}} = U_{\text{DC}} = 600\,\text{V}$, measurement category CAT III

- $U_{\text{AC}} = U_{\text{DC}} = 1000\,\text{V}$, measurement category CAT II

- other circuits:
  working voltage rms 1000 V,
  working voltage peak 3200 V and
  transient overvoltage 1600 V

The standard accessory safety laboratory leads with integrated fuses have to be used!

The RMS values specified for the ranges are the maximum RMS values which could be measured theoretically. This values are not allowed to apply if safe operation or any other safety values are not guaranteed.

Support of *DualPath* (dual bandwidth) mode

This channel does not measure wide bandwidth values and does therefore not support the *DualPath* mode (see BANDWIDTH [6.6→110]).

Sample rate and bandwidth of the AD converter

Table 4.22: L60-CH-C: Sample rate and bandwidth of the AD converter

Sample rate	151.51 kS/s
Resolution	16 bit
Bandwidth, analog anti-aliasing filter always activated	about 15 kHz

Signal filter

This filter is an analog filter in the signal path with a bandwidth of about 15 kHz. To ensure the anti-aliasing function for the analog-digital converter (ADC) with an attenuation of >50 dB at higher frequencies than half of sampling rate the specified design parameters were chosen.

Example: The 15 kHz filter has an attenuation of $-14.7440\,\%$ at a signal frequency $f = 10$ kHz. Those $-14.7440\,\%$ are afflicted with an uncertainty of $\pm 0.012 = \pm 1.2\,\%$. Consequently the measuring signal is damped by $-(14.7 \pm 1.2)\,\%$.

Table 4.23: L60-CH-C: Characteristics of 15 kHz filter

Analog anti-aliasing filter 15 kHz (narrowband)			
Design parameter: Bessel filter, 5th order, cutoff frequency 14.5 kHz			
Frequency / Hz	Filter attenuation in % (nominal)	Worst Case uncertainty of the filter attenuation	Filter attenuation in dB (nominal)
0	0.0004	$\pm 50 \cdot 10^{-6}$	0.0000
10	0.0004	$\pm 50 \cdot 10^{-6}$	0.0000
20	0.0003	$\pm 50 \cdot 10^{-6}$	0.0000
53	0.0000	$\pm 50 \cdot 10^{-6}$	0.0000
100	−0.0012	$\pm 50 \cdot 10^{-6}$	−0.0001
200	−0.0058	$\pm 50 \cdot 10^{-6}$	−0.0005
500	−0.0385	$\pm 50 \cdot 10^{-6}$	−0.0033
1000	−0.1552	$\pm 160 \cdot 10^{-6}$	−0.0135
2000	−0.6210	$\pm 660 \cdot 10^{-6}$	−0.0541
5000	−3.8385	$\pm 3.9 \cdot 10^{-3}$	−0.3400
10000	−14.7440	$\pm 12 \cdot 10^{-3}$	−1.3855
15000	−31.1128	$\pm 18 \cdot 10^{-3}$	−3.2372
75000	−99.7125	—	−50.8267

Harmonics

The special capabilities of this channel for the harmonics measurement are in principle described in INTERHARMONICS = 0 [6.8.1→115] but with following tables:

Processing single path, auto filter off

#	Filter setting	Harmonics setting	Fundamental frequency	Gaps	Max. Order or frequency
1	<15 kHz	no effect	2 - 7.5 Hz	no	2000th or filter frequency
2	<15 kHz	no effect	7.5 - 33 Hz	no	filter frequency
3	<15 kHz	no effect	33 Hz - 7.5 kHz	yes	filter frequency
4	15 kHz	no effect	2 - 7.5 Hz	no	2000th
5	15 kHz	no effect	7.5 - 33 Hz	no	15 kHz
6	15 kHz	no effect	33 Hz - 7.5 kHz	yes	15 kHz

Processing single path, auto filter on, auto harmonics

#	Filter setting	Harmonics setting	Fundamental frequency	Gaps	Max. Order or frequency
12	Auto on	Auto	2 - 7.5 Hz	no	2000th
13	Auto on	Auto	7.5 - 33 Hz	no	15 kHz
14	Auto on	Auto	33 - 7.5 kHz	yes	15 kHz

Processing single path, auto filter on, custom harmonics

#	Filter setting	Harmonics setting	Fundamental frequency	Gaps	Max. Order or frequency
16	Auto on	Custom	2 - 7.5 Hz	no	2000th
17	Auto on	Custom	7.5 - 33 Hz	no	15 kHz
18	Auto on	Custom	33 - 7.5 kHz	yes	15 kHz

General Accuracy Specifications For the amplitude, the measurement uncertainties according to tables 4.25 and 4.26 apply to each individual harmonic or intermediate harmonic at the respective frequency as if it had been measured on its own. In addition, the uncertainties (not the damping) of the filter must be taken into account (table 4.23). The following accuracy specification (4.4) applies to the phase angle if the amplitude of the respective frequency component is $> 0.15\,\%$ of the nominal value of the measuring range and the frequency of the fundamental frequency is in the range from $1\,\text{Hz}$ to $10\,\text{kHz}$. h is the order of the harmonics and f_h the according signal frequency.

$$\pm \left(0.05° + (0.005° \cdot h) + \left(0.05° \cdot \frac{f_\text{h}}{1\,\text{kHz}} \right) \right) \tag{4.4}$$

A calculated example can be found in 6.16.3.

CE Harmonics

CE Harmonische are the case when INTERHARM = 9 and fundamental frequency between $45\,\text{Hz}$ and $65\,\text{Hz}$. For those is valid:

<div style="float:right">INTERHARM
[9.10.288→353]</div>

1. Relative deviation between f1 and frequency fsyn , to which the sampling rate is synchronised is $<0.03\,\%$ of f1 under steady-state conditions.

2. The attenuation of the anti-aliasing filter is $>50\,\text{dB}$ at higher frequencies than half of sampling rate.

3. The instrument complies with class I according IEC 61000-4-7 Ed. 2.0 and EN 61000-4-7 Ed. 2.0.

4. Accuracy specifications according IEC 61000-4-7 Ed. 2.0 and EN 61000-4-7 Ed. 2.0 with m = measuring value and nom = nominal value of the range:

Table 4.24: L60-CH-C: Accuracy harmonics

voltage measurement	Um ≥ 1 % Unom:	± 5 % Um
	Um < 1 % Unom:	± 0.05 % Unom
current measurement	Im ≥ 3 % Inom:	± 5 % Im
	Im < 3 % Inom:	± 0.15 % Inom

The smaller value of the uncertainty in table 4.24 and in the general accuracy specification applies to the specification of the uncertainty of the amplitude.

CE Flicker

According to EN 61000-4-15:2011 resp. IEC 61000-4-15:2010 this is a class F1 flicker meter.

Flickermeter: ±5 % according to EN 61000-4-15, 0.3<k<5.4

d-meter: ±5 % according to EN 61000-4-15

Accuracy specifications

The accuracy specifications are valid under the following conditions:

1. Sinusoidal voltage and current

2. Ambient temperature $(23\pm3)\,°C$, no additional heating or cooling (e.g. by sunlight or current of air)

3. The temperature effect is $\pm(0.01\,\%\,\text{of reading})/\,K$ in the range of $5\,°C\ldots20\,°C$ and $26\,°C\ldots40\,°C$

4. warm up time $1\,h$

5. The upper range value is defined by the maximum peak value, see VOLTAGE CHANNEL L60-CH-C1 AND L60-CH-C2 AND L60-CH-C2.1 [4.9.1→71] and CURRENT CHANNEL L60-CH-C1 AND L60-CH-C2 AND L60-CH-C2.1 [4.9.2→71]. All other values are calculated from voltage, current and active power. The tolerances results from the functional relationship, see MEASUREMENT UNCERTAINTY OF MEASURED AND CALCULATED VALUES [6.16→135].

6. Specified are the direct measured measuring values. The specification is including the $15\,kHz$ signal filter. For information the exact filter attenuation is specified in SIGNAL FILTER [4.9→68].

7. The upper power range value is the product of upper voltage range value and upper current range value. The specification is valid for power factor $0\ldots1$.

8. Voltage and current are higher than $10\,\%$ of nominal range and not higher than the maximum TRMS value of the range. Voltage peak and current peak are not higher than range peak.

9. The instrument should be calibrated every $12\,month$, last adjustment not older than $12\,months$ and adjustment was done at $23\,°C$.

The values are to be interpreted as $\pm(\%$ of reading $+$ $\%$ of upper range value).

Table 4.25: L60-CH-C1 and L60-CH-C2 and L60-CH-C2.1: Accuracy DC …500 Hz

Frequency	DC	0.05 Hz … 45 Hz 65 Hz … 200 Hz	45 Hz … 65 Hz	200 Hz … 500 Hz
Voltage ⊙**U***	0.1 + 0.1	0.02 + 0.05	0.02 + 0.02	0.05 + 0.05
Current ⊙**I***	0.1 + 0.1[b]	0.02 + 0.05[a]	0.02 + 0.02[a]	0.05 + 0.05[a]
Current ⊙**I**$_{\textbf{Sensor}}$	0.1 + 0.1	0.02 + 0.05	0.02 + 0.02	0.05 + 0.05
Active power	0.16 + 0.1[b]	0.032 + 0.05[b]	0.03 + 0.01[b]	0.08 + 0.05[b]

[a] Additional accuracy specification in the 10 A … 32 A range: $\pm\frac{80\,\mu A}{A^2}\cdot I_{\text{trms}}{}^2$

[b] Additional accuracy specification in the 10 A … 32 A range: $\pm\frac{80\,\mu A}{A^2}\cdot I_{\text{trms}}{}^2\cdot U_{\text{trms}}$

Table 4.26: L60-CH-C1 and L60-CH-C2 and L60-CH-C2.1: Accuracy 500 Hz …10 kHz

Frequency	500 Hz … 1 kHz	1 kHz … 2 kHz	2 kHz … 10 kHz
Voltage ⊙**U***	0.2 + 0.1	1.0 + 0.5	f/1 kHz · 1.0 + f/1 kHz · 1.0
Current ⊙**I***	0.2 + 0.1[a]	1.0 + 0.5[a]	f/1 kHz · 1.0 + f/1 kHz · 1.0[a]
Current ⊙**I**$_{\textbf{Sensor}}$	0.2 + 0.1	1.0 + 0.5	f/1 kHz · 1.0 + f/1 kHz · 1.0
Active power	0.32 + 0.1[b]	1.6 + 0.5[b]	f/1 kHz · 1.6 + f/1 kHz · 1.0[b]

[a] Additional accuracy specification in the 10 A … 32 A range: $\pm\frac{80\,\mu A}{A^2}\cdot I_{\text{trms}}{}^2$

[b] Additional accuracy specification in the 10 A … 32 A range: $\pm\frac{80\,\mu A}{A^2}\cdot I_{\text{trms}}{}^2\cdot U_{\text{trms}}$

Common-mode rejection ratio (CMRR)

Common-mode rejection ratio is measured by using a sinusoidal voltage of 100 V against earth. For the voltage ranges the signal is applied to the short circuit jacks ⊙U and ⊙U* . For the current ranges the signal is applied to the short circuit jacks ⊙I and ⊙I$_{Sensor}$ resp. to the ⊙I jack.

Common-mode frequency	Signal-Filter	U*, 3 V range	U*, 1000 V range	I*, 5 mA range	I*, 32 A range	Isensor, 4 A range
53 Hz	15 kHz	>120 dB	>120 dB	>160 dB	>150 dB	>120 dB
53 Hz	off	>120 dB	>120 dB	>160 dB	>150 dB	>120 dB
100 kHz	15 kHz	>120 dB	>120 dB	>160 dB	>150 dB	>115 dB
100 kHz	off	>75 dB	>75 dB	>100 dB	>150 dB	>110 dB

4.9.1 Voltage channel L60-CH-C1 and L60-CH-C2 and L60-CH-C2.1

Measurement ranges for the jacks ⊙U*/⊙U

Nominal range / V	3	6	12.5	25	60	130	250	400	600	1000
Max. TRMS value / V	3.3	6.6	13.8	27.5	66	136	270	440	660	1000
Max. peak value / V	6	12	25	50	100	200	400	800	1600	3200
Input impedance L60-CH-C1	about 4.59 MΩ, 3 pF									
Input impedance L60-CH-C2 and L60-CH-C2.1	2.69 MΩ ±1 %, 4 pF									
Overload capability	1000 V + 10 % continuously 1500 V for 1 s 2500 V for 20 ms, transient									
Capacity against earth	about 90 pF									

4.9.2 Current channel L60-CH-C1 and L60-CH-C2 and L60-CH-C2.1

Measurement ranges for the jacks ⊙I*/⊙I

Nominal value / A	0.005	0.010	0.020	0.040	0.080	0.150	0.300	0.600
Max. TRMS value / A	0.0055	0.011	0.022	0.044	0.088	0.165	0.330	0.660
Max. peak value / A	0.014	0.028	0.056	0.112	0.224	0.469	0.938	1.875
Input impedance	about 2.2 Ω		about 600 mΩ			about 80 mΩ		
Overload capability	LMG operating, 10 A continuously, 150 A for 10 ms							
Capacity against earth	about 90 pF							

Nominal value / A	1.2	2.5	5	10	20	32
Max. TRMS value / A	1.32	2.75	5.5	11	22	32
Max. peak value / A	3.75	7.5	15	30	60	120
Input impedance	about 20 mΩ			about 10 mΩ		
Overload capability	LMG operating, 32 A continuously, 150 A for 10 ms					
Capacity against earth	90 pF					

Measurement ranges for the jacks ⊙I$_{Sensor}$/⊙I

Nominal range / V	0.030	0.060	0.120	0.250	0.500	1	2	4
Max. TRMS value / V	0.033	0.066	0.132	0.275	0.550	1.1	2.2	4.4
Max. peak value / V	0.0977	0.1953	0.3906	0.7813	1.563	3.125	6.25	12.5
Input impedance	99.8 kΩ ±1 %, 34 pF							
Overload capability	100 V continuously, 250 V for 1 s							
Capacity against earth	about 90 pF							

4.10 Timebases and summation

4.10.1 Timebase of frequency measurement

The timebase which controls the frequency measuring has an accuracy specification of

- $\pm 50\,\text{ppm}$ of measured value for the PSI signal A frequency inputs.

SYNCBW
[9.10.467→432]

- $\pm 50\,\text{ppm}$ of measured value for channel synchronisation if synchronisation is set to full band.

- $\pm 80\,\text{ppm}$ of measured value for channel synchronisation if synchronisation is set to wide band.

CYCL [9.10.356→386]

- $\pm(50\,ppm + 6.6\,\mu s\,/\,cycletime)$ of measured value for channel synchronisation if synchronisation is set to small band.

4.10.2 Timebase of energy measurement

The timebase which controls the energy measuring has an accuracy specification of $\pm 50\,\text{ppm}$ of measured time interval.

4.10.3 Summation of energy

With 30ms cycle time more than 1,000,000,000 single energy values are summed during one year. This can lead to calculation uncertainties. For energy intervals <10 years, this uncertainties are smaller than $\pm 50\,\text{ppm}$ of the measured value. This uncertainty can be significantly reduced by choosing shorter intervals and/or longer cycle times.

4.11 Flicker meter

The flicker meter is build according IEC 61000-4-15:2010 resp. EN 61000-4-15:2011. It is implemented as one or three channel instrument in the first group. The performance tests according to this standards are splitted into two sections: As the implementation of the flicker meter is digital and the hardware has a very small influence, most of the tests are done with simulated signals, using the same routines which the instruments uses for the real calculations. The results of this type tests are printed below. Beside this and to be sure, that the measurement hardware does not has any significant influence, table 5 can be checked with real signals by an individual flicker calibration of a specific instrument.

For all performance tests the reference values are (if not other specified):
$P_{inst,max} = 1.000000.08$, $P_{st} = 1.000000.05$
Utilisationindicates, how much of the specification is used.

4.11.1 Performance for 230 V, 50 Hz systems

f / Hz	table 1 (sinusoidal)			table 2 (rectangular)		
	$(\Delta U/U)$ / %	$P_{inst,max}$	Utilisation / %	$(\Delta U/U)$ / %	$P_{inst,max}$	Utilisation / %
0.5	2.325	0.97277	-34	0.509	0.99772	-3
1.5	1.067	0.99676	-4	n.a.	n.a.	n.a.
3.5	n.a.	n.a.	n.a.	0.342	0.99805	-2
8.8	0.250	1.00032	0	0.196	1.00093	1
18.0	n.a.	n.a.	n.a.	0.446	0.99989	-0
20.0	0.704	1.00090	1	n.a.	n.a.	n.a.
21.5	n.a.	n.a.	n.a.	0.592	1.00087	1
25.0	1.037	0.99939	-1	0.764	1.00055	1
28.0	n.a.	n.a.	n.a.	0.915	1.00254	3
30.5	n.a.	n.a.	n.a.	0.847	1.00585	7
33.3	2.128	0.99958	-1	1.671	1.00359	4

Table 4.27: 230 V / 50 Hz system, Normalized response test according tables 1 and 2

r / min^{-1}	$(\Delta U/U)$ / %	P_{st}	Utilisation / %
1.00	2.715	0.99344	-13
2.00	2.191	0.99237	-15
7.00	1.450	0.99178	-16
39.00	0.894	0.99506	-10
110.00	0.722	0.99941	-1
1620.00	0.407	0.99959	-1
4000.00	2.343	1.00070	1

Table 4.28: 230 V / 50 Hz system, Rectangular voltage change test according table 5

0.10000	$\leq P_{st} \leq$	5.00000

Table 4.29: 230 V / 50 Hz system, P_{st} performance test

$P_{inst,max}$	Utilisation / %
0.98909	-14

Table 4.30: 230 V / 50 Hz system, frequency change test according table 6

$P_{inst,max}$	Utilisation / %
1.00055	1

Table 4.31: 230 V / 50 Hz system, distortion voltage test according table 8

$f_{\nu,max}$ / Hz	1500

Table 4.32: 230 V / 50 Hz system, input bandwidth test according table 9

$\Delta\beta\,/\,°$	$P_{st,ref}$	P_{st}	Utilisation / %
+30	0.86300 to 0.96300	0.91463	3
−30	0.86300 to 0.96300	0.91370	1
+45	1.00700 to 1.11300	1.06886	17
−45	1.00700 to 1.11300	1.06979	18

Table 4.33: 230 V / 50 Hz system, phase jump test according table 10

P_{st}	Utilisation / %
0.99894	-2

Table 4.34: 230 V / 50 Hz system, duty cycle test according table 11

	$d_c\,/\,\%$	$d_{max}\,/\,\%$	$t_{d(t)>3.3\%}\,/\,\mathrm{s}$
Test result	2.00000	4.00000	0.50000
Utilisation / %	0	0	0

Table 4.35: 230 V / 50 Hz system, d-Meter test according table 12

	$d_c\,/\,\%$	$d_{max}\,/\,\%$	$t_{d(t)>3.3\%}\,/\,\mathrm{s}$
Test result	1.00000	5.00000	0.60000
Utilisation / %	0	0	0

Table 4.36: 230 V / 50 Hz system, d-Meter test according table 13

4.11.2 Performance for 230 V, 60 Hz systems

	table 1 (sinusoidal)			table 2 (rectangular)		
f / Hz	$(\Delta U/U)\,/\,\%$	$P_{inst,max}$	Utilisation / %	$(\Delta U/U)\,/\,\%$	$P_{inst,max}$	Utilisation / %
0.5	2.325	0.97260	-34	0.510	0.99843	-2
1.5	1.067	0.99651	-4	n.a.	n.a.	n.a.
3.5	n.a.	n.a.	n.a.	0.342	0.99753	-3
8.8	0.250	1.00015	0	0.196	1.00137	2
18.0	n.a.	n.a.	n.a.	0.457	1.00302	4
20.0	0.703	0.99901	-1	n.a.	n.a.	n.a.
22.0	n.a.	n.a.	n.a.	0.611	1.00474	6
25.5	n.a.	n.a.	n.a.	0.768	1.00264	3
33.3	1.758	0.99997	-0	1.258	1.00316	4
37.0	n.a.	n.a.	n.a.	0.975	1.00956	12
40.0	2.963	0.99924	-1	2.327	1.00488	6

Table 4.37: 230 V / 60 Hz system, Normalized response test according tables 1 and 2

r / min^{-1}	$(\Delta U/U)\,/\,\%$	P_{st}	Utilisation / %
1.00	2.719	0.99353	-13
2.00	2.194	0.99246	-15
7.00	1.450	0.99161	-17
39.00	0.895	0.99574	-9
110.00	0.723	1.00038	1
1620.00	0.409	1.00123	2
4800.00	3.263	1.00290	6

Table 4.38: 230 V / 60 Hz system, Rectangular voltage change test according table 5

0.10000	$\leq P_{\mathrm{st}} \leq$	5.00000

Table 4.39: 230 V / 60 Hz system, P_{st} performance test

$P_{\mathrm{inst,max}}$	Utilisation / %
0.98891	-14

Table 4.40: 230 V / 60 Hz system, frequency change test according table 6

$P_{inst,max}$	Utilisation / %
1.00027	0

Table 4.41: 230 V / 60 Hz system, distortion voltage test according table 8

$f_{\nu,max}$ / Hz	1500

Table 4.42: 230 V / 60 Hz system, input bandwidth test according table 9

$\Delta\beta$ / °	$P_{\mathrm{st},ref}$	P_{st}	Utilisation / %
+30	0.71000 to 0.81000	0.76064	1
−30	0.71000 to 0.81000	0.75976	0
+45	0.83200 to 0.93200	0.88108	2
−45	0.83200 to 0.93200	0.88092	2

Table 4.43: 230 V / 60 Hz system, phase jump test according table 10

P_{st}	Utilisation / %
0.99954	-1

Table 4.44: 230 V / 60 Hz system, duty cycle test according table 11

	d_c / %	d_{max} / %	$t_{d(t)>3.3\%}$ / s
Test result	2.00000	4.00000	0.50000
Utilisation / %	0	0	0

Table 4.45: 230 V / 60 Hz system, d-Meter test according table 12

	d_c / %	d_{max} / %	$t_{d(t)>3.3\%}$ / s
Test result	1.00000	5.00000	0.60000
Utilisation / %	0	0	0

Table 4.46: 230 V / 60 Hz system, d-Meter test according table 13

4.11.3 Performance for 120 V, 50 Hz systems

f / Hz	table 1 (sinusoidal)			table 2 (rectangular)		
	$(\Delta U/U)$ / %	$P_{inst,max}$	Utilisation / %	$(\Delta U/U)$ / %	$P_{inst,max}$	Utilisation / %
0.5	2.453	0.97817	-27	0.597	1.00313	4
1.5	1.126	1.00321	4	n.a.	n.a.	n.a.
3.5	n.a.	n.a.	n.a.	0.408	1.00628	8
8.8	0.321	1.00648	8	0.252	1.00866	11
18.0	n.a.	n.a.	n.a.	0.611	1.00604	8
20.0	0.978	1.00632	8	n.a.	n.a.	n.a.
21.5	n.a.	n.a.	n.a.	0.820	1.00829	10
25.0	1.476	1.00679	8	1.087	1.00716	9
28.0	n.a.	n.a.	n.a.	1.303	1.00987	12
30.5	n.a.	n.a.	n.a.	1.144	1.01259	16
33.3	3.111	1.00571	7	2.443	1.01036	13

Table 4.47: 120 V / 50 Hz system, Normalized response test according tables 1 and 2

r / min^{-1}	$(\Delta U/U)$ / %	P_{st}	Utilisation / %
1.00	3.178	0.99535	-9
2.00	2.561	0.99413	-12
7.00	1.694	0.99305	-14
39.00	1.045	0.99796	-4
110.00	0.844	1.00327	7
1620.00	0.545	1.00313	6
4000.00	3.426	1.00431	9

Table 4.48: 120 V / 50 Hz system, Rectangular voltage change test according table 5

0.10000	$\leq P_{st} \leq$	5.00000

Table 4.49: 120 V / 50 Hz system, P_{st} performance test

$P_{inst,max}$	Utilisation / %
0.99351	-8

Table 4.50: 120 V / 50 Hz system, frequency change test according table 6

$P_{inst,max}$	Utilisation / %
1.00660	8

Table 4.51: 120 V / 50 Hz system, distortion voltage test according table 8

$f_{\nu,max}$ / Hz	1500

Table 4.52: 120 V / 50 Hz system, input bandwidth test according table 9

$\Delta\beta\,/\,°$	$P_{st,ref}$	P_{st}	Utilisation / %
+30	0.65600 to 0.75600	0.70914	6
−30	0.65600 to 0.75600	0.70847	5
+45	0.76900 to 0.86900	0.82994	22
−45	0.76900 to 0.86900	0.83040	23

Table 4.53: 120 V / 50 Hz system, phase jump test according table 10

P_{st}	Utilisation / %
1.00136	3

Table 4.54: 120 V / 50 Hz system, duty cycle test according table 11

	$d_c\,/\,\%$	$d_{max}\,/\,\%$	$t_{d(t)>3.3\%}\,/\,\mathrm{s}$
Test result	1.99999	4.00001	0.50000
Utilisation / %	0	0	0

Table 4.55: 120 V / 50 Hz system, d-Meter test according table 12

	$d_c\,/\,\%$	$d_{max}\,/\,\%$	$t_{d(t)>3.3\%}\,/\,\mathrm{s}$
Test result	1.00000	5.00000	0.60000
Utilisation / %	0	0	0

Table 4.56: 120 V / 50 Hz system, d-Meter test according table 13

4.11.4 Performance for 120 V, 60 Hz systems

	table 1 (sinusoidal)			table 2 (rectangular)		
f / Hz	$(\Delta U/U)\,/\,\%$	$P_{inst,max}$	Utilisation / %	$(\Delta U/U)\,/\,\%$	$P_{inst,max}$	Utilisation / %
0.5	2.453	0.97816	-27	0.598	1.00360	5
1.5	1.126	1.00307	4	n.a.	n.a.	n.a.
3.5	n.a.	n.a.	n.a.	0.408	1.00557	7
8.8	0.321	1.00639	8	0.252	1.00952	12
18.0	n.a.	n.a.	n.a.	0.626	1.00715	9
20.0	0.977	1.00539	7	n.a.	n.a.	n.a.
22.0	n.a.	n.a.	n.a.	0.851	1.01115	14
25.5	n.a.	n.a.	n.a.	1.072	1.00988	12
33.3	2.570	1.00613	8	1.823	1.00830	10
37.0	n.a.	n.a.	n.a.	1.304	1.01538	19
40.0	4.393	1.00502	6	3.451	1.01173	15

Table 4.57: 120 V / 60 Hz system, Normalized response test according tables 1 and 2

$\mathrm{r}\,/\,\mathrm{min}^{-1}$	$(\Delta U/U)\,/\,\%$	P_{st}	Utilisation / %
1.00	3.181	0.99519	-10
2.00	2.564	0.99417	-12
7.00	1.694	0.99275	-15
39.00	1.040	0.99286	-14
110.00	0.844	1.00303	6
1620.00	0.548	1.00442	9
4800.00	4.837	1.00459	9

Table 4.58: 120 V / 60 Hz system, Rectangular voltage change test according table 5

0.10000	$\leq P_{st} \leq$	5.00000

Table 4.59: 120 V / 60 Hz system, P_{st} performance test

$P_{inst,max}$	Utilisation / %
0.99533	-6

Table 4.60: 120 V / 60 Hz system, frequency change test according table 6

$P_{inst,max}$	Utilisation / %
1.00645	8

Table 4.61: 120 V / 60 Hz system, distortion voltage test according table 8

$f_{\nu,max}$ / Hz	1500

Table 4.62: 120 V / 60 Hz system, input bandwidth test according table 9

$\Delta\beta$ / °	$P_{st,ref}$	P_{st}	Utilisation / %
+30	0.53700 to 0.63700	0.58991	6
−30	0.53700 to 0.63700	0.58971	5
+45	0.63100 to 0.73100	0.68413	6
−45	0.63100 to 0.73100	0.68467	7

Table 4.63: 120 V / 60 Hz system, phase jump test according table 10

P_{st}	Utilisation / %
1.00230	5

Table 4.64: 120 V / 60 Hz system, duty cycle test according table 11

	d_c / %	d_{max} / %	$t_{d(t)>3.3\%}$ / s
Test result	1.99999	4.00001	0.50000
Utilisation / %	0	0	0

Table 4.65: 120 V / 60 Hz system, d-Meter test according table 12

	d_c / %	d_{max} / %	$t_{d(t)>3.3\%}$ / s
Test result	1.00000	5.00000	0.60000
Utilisation / %	0	0	0

Table 4.66: 120 V / 60 Hz system, d-Meter test according table 13

4.12 Processing signal interface L6-OPT-PSI

This interface has several analog and digital inputs and outputs. They are arranged in groups and these groups are isolated against each other and against earth for signals with a maximum amplitude of 33 V. This isolation between the groups prevents ground loops.

4.12.1 Fast analog inputs

These two analog inputs have their own ground system (isolated against any other connector) and are sampled with about $150\,kS/s$. They are accessible via the two BNC connectors.

Pins
AFIn_1 and AFIn_2 with AFIn_GND12

Resolution
16 Bit

Accuracy specifications
$\pm(0.05\,\%$ of reading $+\ 0.05\,\%$ of full scale value)

-3 dB bandwidth
14.5 kHz

Full scale value
$\pm20\,V$

Overload capability
$\pm33\,V$

Input resistance
$100\,k\Omega \pm 0.1\,\%$

4.12.2 Slow analog inputs

These eight analog inputs have their own ground system (isolated against any other connector) and are sampled with about $100\,S/s$. They are accessible via the 9 pin female D-SUB connector, see Figure 4.11 [→79]. They are designed to measure DC or slow changing signals like temperature, etc.
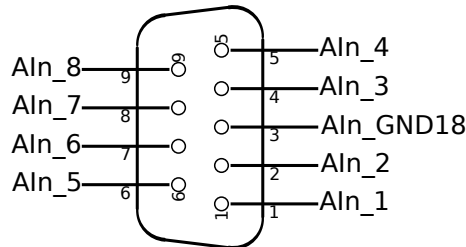
Figure 4.11: Processing signal interface, analog input ,
9 pin D-SUB female face view

Pins
AIn_1 ... AIn_8 with AIn_GND18

Resolution
16 Bit

Accuracy specifications
$\pm(0.05\,\%$ of reading $+\ 0.05\,\%$ of full scale value)

Full scale value
$\pm13\,V$

Overload capability
$\pm33\,V$

Input impedance

$100\,\mathrm{k}\Omega \pm 0.1\,\%$

4.12.3 Analog outputs

These 32 analog outputs have their own ground system (isolated against any other connector) and are set after each measurement cycle. They are accessible via the 15 pin and 25 pin female D-SUB connectors, see Figure 4.12 [→80].
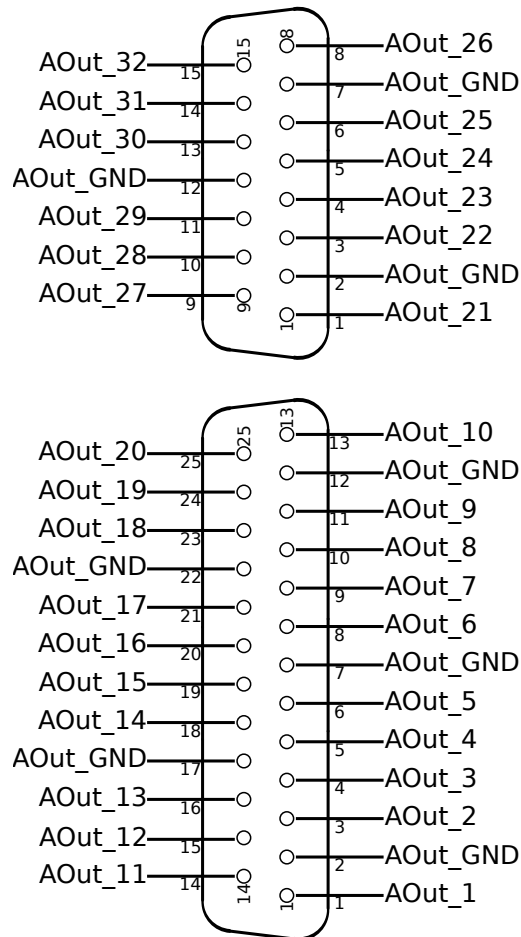
Figure 4.12: Processing signal interface, analog output ,
25 pin and 15 pin D-SUB female face view

Pins

AOut_1 ... AOut_32 with AOut_GND

Resolution

14 Bit

Accuracy specifications

$\pm(0.05\,\%$ of output $+\ 0.05\,\%$ of full scale value$)$

Full scale value

$\pm 11\,\mathrm{V}$
Short circuit proof against AOut_GND

Output load

$<10\,\mathrm{mA}$

Overload

It is not allowed to apply any voltage to the output pins.

Output impedance

$<1\,\Omega$

4.12.4 Switching outputs

These 8 outputs are arranged as 6 switches with two connectors each and 2 switches with a common negative connector. Thus the outputs are isolated against each other and the other ground systems. They are accessible at one female 25 pin D-SUB connector (see Figure 4.13 [→81]) and set after each measurement cycle. This switches are realized as NPN transistors. That is the reason, why one pin is marked with minus (-, emitter) and the other pin is marked with plus (+, collector).
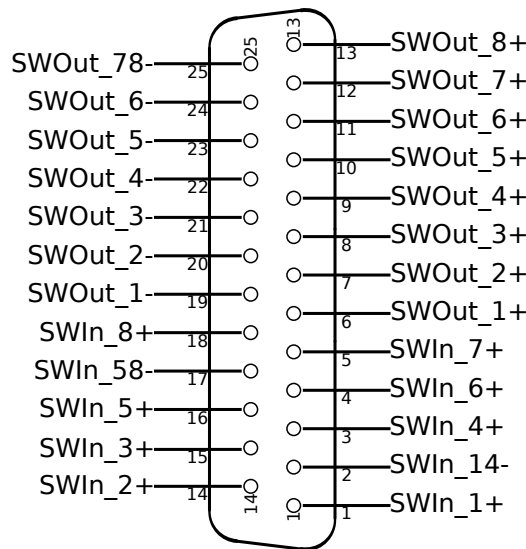
Figure 4.13: Processing signal interface, switching inputs and outputs ,
25 pin D-SUB female face view

Pins

SWOut_1+ ... SWOut_8+   with   SWOut_1- ... SWOut_6-   and   SWOut_78-

Opened switch (State = 0)

Leakage current max. 10 µA

Closed switch (State = 1)

max. 0.8 V @ 1.6 mA (for logic inputs)
max. 0.9 V @ max. 200 mA (for relays)

Reverse polarity

Current through reverse diode max. -100 mA

Overload

-1 V ... +33V

4.12.5 Switching inputs

These 8 inputs are arranged in two groups, each with 4 inputs and a common ground. Thereby the input groups are isolated against each other and the other ground systems. They are accessible at one 25 pin D-SUB connector (see Figure 4.13 [→81]) and are sampled with about 150 kS/s. These inputs have to drive the LED of an optocoupler and therefore require some current to work! That is the reason, why they one pin is marked with minus (-, cathode) and the other pin is marked with plus (+, anode).

Pins
SWIn_1+ ...SWIn_8+ with SWIn_14- and SWIn_58-

Input signal
State = 0 (Low) is detected for voltage ≤1.4 V @ max. 10 µA
State = 1 (High) is detected for voltage ≥2.3 V @ max. 0.5 mA.

Overload
±33 V

4.12.6 Speed/Torque/Frequency inputs

These 2 inputs have one common ground. Each input has 3 signals available:

- A is used for frequency/speed.

- B is used for detection of the direction. At positive direction this signal is lagging 90° against signal A.

- Z is used for the Zero position of the sensor.
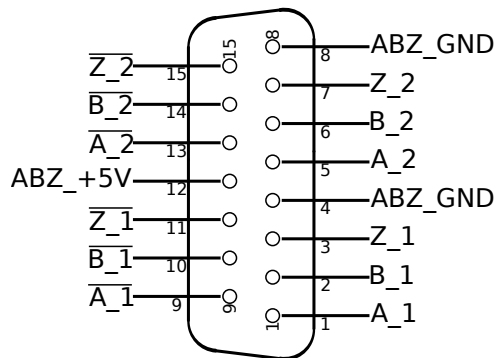
Figure 4.14: Processing signal interface, speed torque frequency input ,
15 pin D-SUB female face view

The type of accepted input signals as well as filtering can be selected:

- TTL: The signal must be applied against ground (single ended)

- HTL: The signal must be applied against ground (single ended)

- RS422 : The signal must be applied against the $\overline{inverse\,input}$ (differential)

- Filter: Low, Mid, High or Off.
  The filters are used to reject possible distortions which are coupled into the signal cable. The cut off frequencies of this filters largely depends on the signal type and the real signal levels. The following table lists maximum frequencies for the logic detection at defined levels. The bandwidth should not be chosen larger than required. At higher frequencies the direction, frequency or zero position could be wrong.
  The minimum frequency depends on the cycle time (it is about the reciprocal of the cycle time)

Signal typ	TTL	HTL	RS422
@level "0"	0 V	0 V	-0.8 V
@level "1"	5 V	10 V	+0.8 V
Filter Low	10 kHz	10 kHz	45 kHz
Filter Mid	60 kHz	60 kHz	250 kHz
Filter High	200 kHz	200 kHz	800 kHz
Filter Off	5 MHz	5 MHz	10 MHz

Table 4.67: Maximum values for frequency inputs at different filters and signals

These inputs can be used as follows:

- Speed input sensor
  The signal A is used to determine the frequency of the sensor. The signals B and Z (if available) are used to determine the rotation direction and the zero position, respectively.

- Fast torque input sensors with digital output
  These sensors have usually only signal A. Please note, that this input can only be used for sensors with digital output. Sensors with analog output can be connected (depending on their speed) to the fast or slow analog inputs.

- General frequency measurement
  This frequency must be applied to the A input, B and Z can be connected to ground to counter interfering signals.

From the A inputs the instrument always calculates the frequency once per cycle time, the B and Z inputs are sampled with about 150 kS/s.

The signals and an auxiliary supply for feeding sensors are accessible via a 15 pin D-SUB connector (see Figure 4.14 [→82]).

Pins
A_1, $\overline{A\_1}$, A_2, $\overline{A\_2}$, B_1, $\overline{B\_1}$, B_2, $\overline{B\_2}$, Z_1, $\overline{Z\_1}$, Z_2, $\overline{Z\_2}$, ABZ_GND, ABZ_+5V

TTL signals
Low is detected for voltage ≤ 1.0 V
High is detected for voltage ≥ 1.8 V
These voltages must be applied between X_n and ABZ_GND.

HTL signals
Low is detected for voltage ≤ 4.0 V
High is detected for voltage ≥ 7.0 V
These voltages must be applied between X_n and ABZ_GND.

RS422 signals (differential)
The differential voltage must be $\geq \pm 0.2$ V and $\leq \pm 10$ V and be applied between X_n and $\overline{X\_n}$.

The allowed common mode voltage must be $\geq$-7.0 V and $\leq$+12.0 V (applied between X_n and ABZ_GND).

Overload
-10 V ... +33 V between any input pins inside this group

Input resistance
Single ended: ≥ 4 kΩ (between X_n and ABZ_GND)
Differential: ≥ 4 kΩ (between X_n and ABZ_GND as well as $\overline{X\_n}$ and ABZ_GND)

Accuracy specifications
See Timebase of frequency measurement [4.10.1→72]

Auxiliary supply

$(+5\pm0.5)\,\text{V}$ @ max. $200\,\text{mA}$ (short circuit proof against ABZ_GND)

5 Installation

5.1 Unpacking and installing the instrument

After unpacking the equipment, it should be checked for damage. Damage due to transportation should be reported to the transporter and to the equipment supplier at the earliest opportunity. If it is uncertain if one can safely use the damaged equipment, then the equipment should not be used.

Next, check if all items listed on the delivery note are included. The package should be stored for future transports, e.g. for the annual calibration.

The instrument shall only be used in an environment as described in this manual. To ascertain sufficient air circulation, the instrument should be operated in a horizontal position. Please regard the technical data given in TECHNICAL DATA [4→39].

5.1.1 Mounting the hand grips L6-OPT-HDL2 resp. L6-OPT-HDL4 (LMG670, 640 und 610)

For the mounting of the hand grips the front frame doesn't need to be removed. Simply mount each grip with the 2 shipped screws M4x16. See also Figure 4.2 [→41], Figure 4.6 [→43] and Figure 4.9 [→45].

Caution! When removing the grips, do *not* put the screws back to the instrument, but store them together with the grip. Without grip the screws are too long and could reduce clearances inside the instrument.

5.1.2 Mounting the hand grips L6X1-OPT-HDL2 resp. L6X1-OPT-HDL4 (LMG671)

For mounting the front grips L6X1-OPT-HDL2 the frame doesn't need to be removed. Only the screws on the left and right hand side of the frame must be removed. In place of the screws a distance ring must be put into the holes left by the screws. The grip can now be hold in position and the 2 shipped M3x20 screws can be put throught the distance rings and tightened. In addition, 2 M4x16 screws attach the grip to the side of the device (see figure Figure 5.1 [→86]).

The set L6X1-OPT-HDL4 also contains rear grips. These are attached at the rear using 2 M3x10 and 2 M4x10 screws for each grip (see figure Figure 5.2 [→86]).

Attention! In case the grips are being removed, the removed screws must not be inserted without the grips and should be stored together with the gips. If used without grips, the screws are too long and may reduce needed clearances.

5.1.3 Mounting the 19" rack mount kit L67-X-01

This kit can be used for LMG670 and LMG610. For the LMG610 please note, that this instrument was not designed for rack mounting. Using this kit the mechanical mounting will work without any problems, but the electrical connectors at the right side of the instrument might be hard to access, depending on the dimensions of the 19" rack. For the mounting of the 19" rack mount kit the front frame has to be removed:

- Remove the 6 existing screws M3x8 from the frame (two rounded head screws left/right each and two countersunk screws on bottom) and remove the frame.

- The two screws on the bottom are replaced by the attached countersunk screws M3x6.

- Mount each hand grip with the two attached screws M5x10 on a bracket.
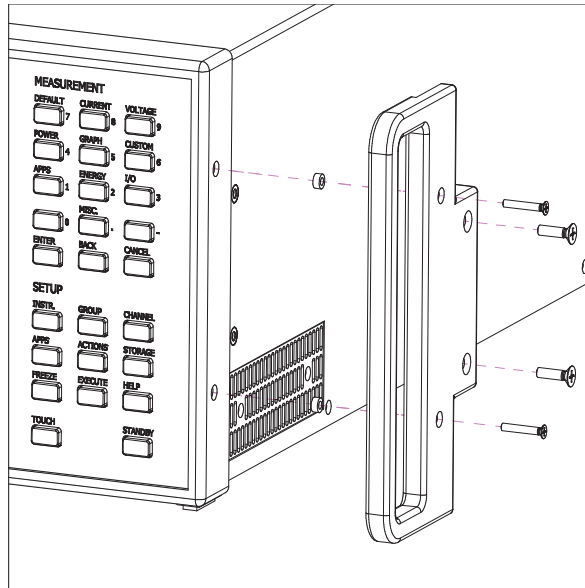
Figure 5.1: Mounting the front grips (**LMG671**)
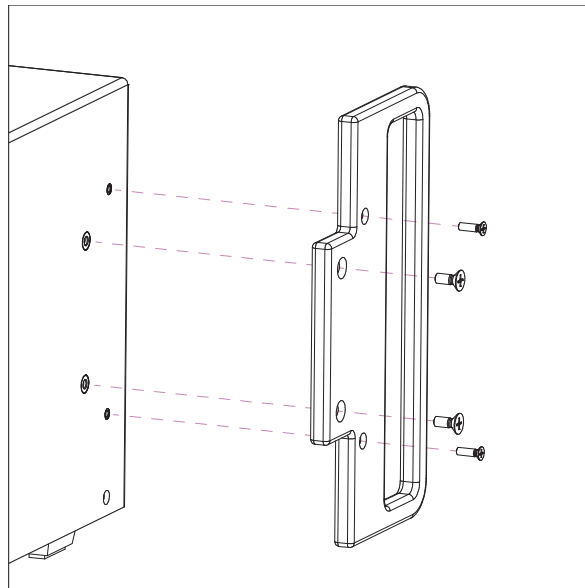
Figure 5.2: Mounting the rear grips (**LMG671**)

- Mount the bracket with the 2 attached screws M4x8, each with one of the attached spring washers, and the 2 removed rounded head screws M3x8.

See also Figure 4.2 [→41], Figure 4.6 [→43] and Figure 4.9 [→45].

Caution! When removing the 19" rack mount kit, do *not* put the screws back to the instrument, but store them together with the 19" rack mount kit. Without 19" rack mount kit the screws are too long and could reduce clearances inside the instrument.

The brackets will not carry the whole weight of the instrument! Please use sufficient rails in the rack.
Depending on the hardware revision, the stands can't be removed. As there is some ventilation space necessary below the instrument this is no problem, see ENVIRONMENTAL CONDITIONS *[4.1→39]*

5.1.4 Mounting the 19" rack mount kit L64-X-01

This kit can be used for LMG640. For the mounting of the 19" rack mount kit the front frame has to be removed:

- Depending on the hardware revision remove the 6 resp. 7 existing screws M3x8 from the frame (two rounded head screws left/right each and two/three countersunk screws on bottom) and remove the frame.

- The two/three screws on the bottom are replaced by the attached countersunk screws M3x6.

- Mount each of the two hand grips with the two attached screws M5x10 on the designated bracket.

- Mount the brackets with the hand grips with the 2 attached screws M4x8, each with one of the attached spring washers, and the 2 removed rounded head screws M3x8 near the front side..

- Mount the brackets without the hand grips with the 2 attached screws M4x8, each with one of the attached spring washers, near the rear side. This brackets are used to hold the rear of the instrument on the rack's rails.

See also Figure 4.2 [→41], Figure 4.6 [→43] and Figure 4.9 [→45].

Caution! When removing the 19" rack mount kit, do *not* put the screws back to the instrument, but store them together with the 19" rack mount kit. Without 19" rack mount kit the screws are too long and could reduce clearances inside the instrument.

The brackets will not carry the whole weight of the instrument! Please use sufficient rails in the rack.
Depending on the hardware revision, the stands can't be removed. As there is some ventilation space necessary below the instrument this is no problem, see ENVIRONMENTAL CONDITIONS *[4.1→39]*

5.1.5 Mounting the 19" rack mount kit L671-X-01 (LMG671)

This kit can only be used with the LMG671 device. The frame on the front needs to be removed before the kit can be mounted:

- The 6 M3x8 screws of the frame needs to be removed (2 fillister head screws on the left and right hand side and 2 countersunk head screws at the bottom). Now the frame itself can be removed.

- The 2 screws at the bottom must be replaced with the shipped M3x6 screws.

- The grips are mounted to the bracket with the 2 shipped M5x10 screws.

- The brackets are mounted usind the shippped M4x8 screws with spring washers and 2 of the removed M3x8 fillister head screws (see Figure 5.3 [→88]).

Caution! If the rack mount kit is being removed, the screws must not stay on the device. They should be kept along with the rack mount kit. If used without the kit, the screws reach too far into the device and may reduce the needed clearances.

The brackets can not carry the full weight of the device on their own. Thus, suitable rails need to be put into the rack.
Below the device some ventilation space is needed (see ENVIRONMENTAL CONDITIONS *[4.1→39]).*
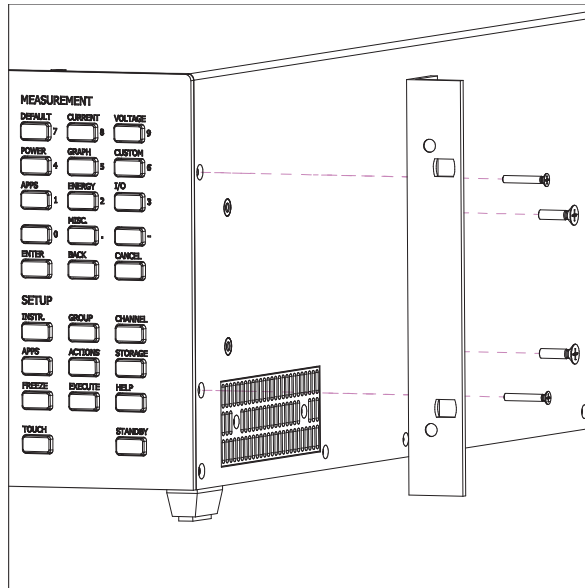
Figure 5.3: Montage des 19"-Montagesatzes (**LMG671**)
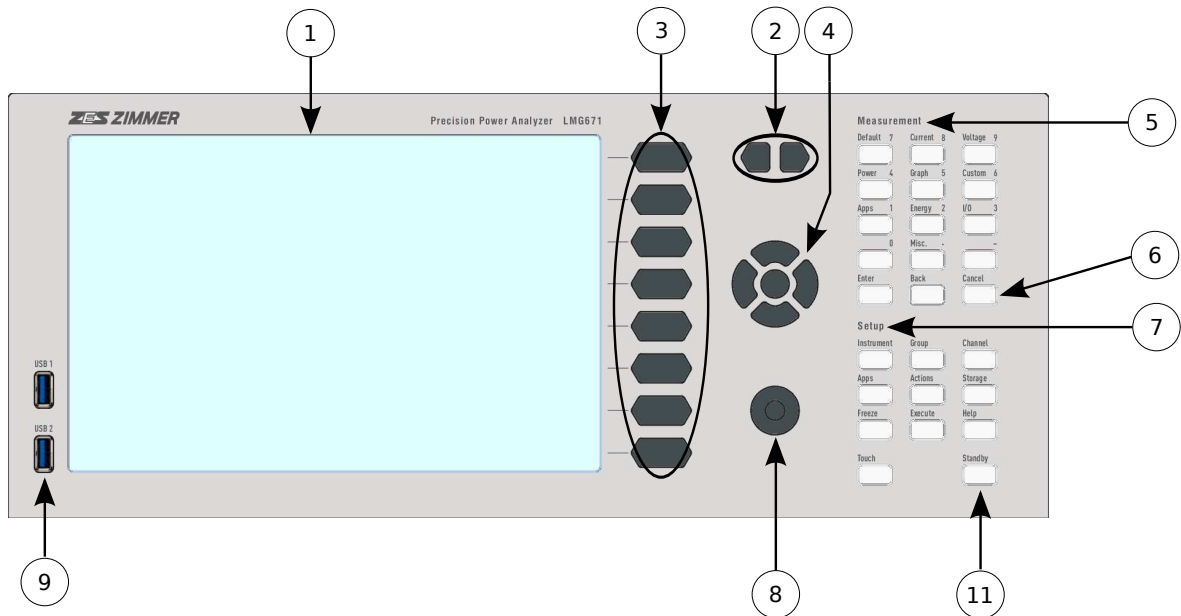
Figure 5.4: Elemente der LMG671-Frontplatte

5.2 Front panel

The LMG671 ATE is equipped with a special front panel. It features a power button and an operational state indicator LED. The screen output can be displayed on an external monitor. USB keyboard and mouse devices can be used for direct operation of the device.

The front panel items are almost identical for all LMG600 series instruments. The numbers in Figure 5.4 [→88] and Figure 5.5 [→89] correspond to the following elements:

1. Touchscreen
   Here all values and menus are displayed. Through the touchscreen, users can operate the instrument in a very simple and direct way.
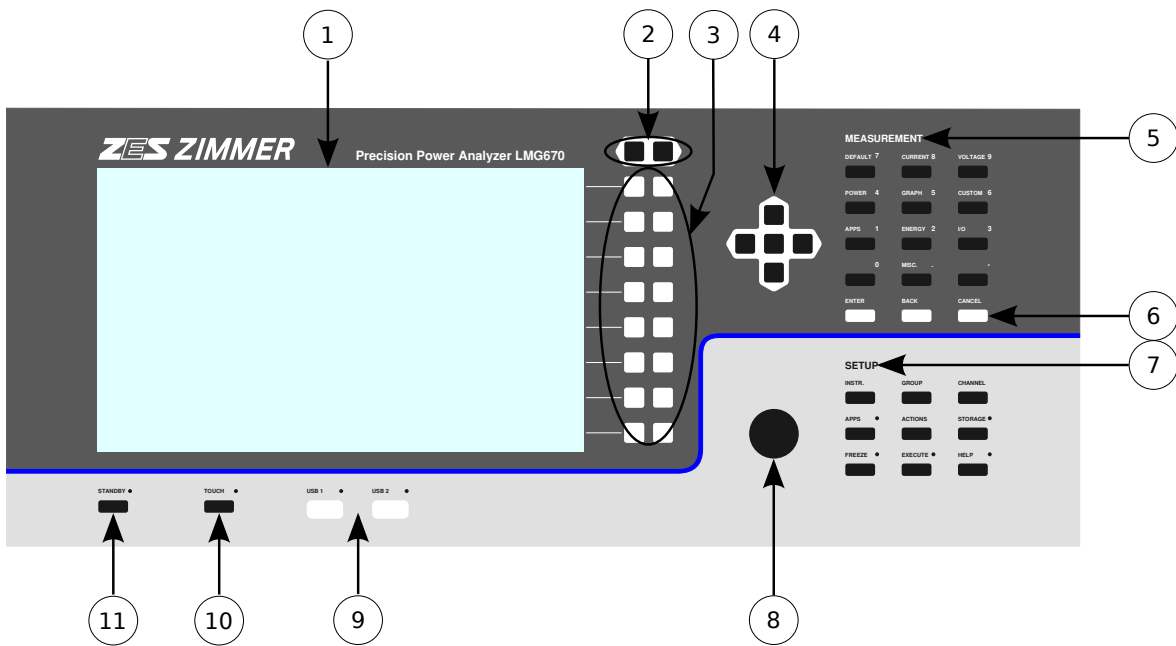
Figure 5.5: Elements of the front panel

2. Tab Keys

   Most menus consist of several tabs displaying different values. With the tab keys, the selected tab can be changed.

   Depending on the context, this function sometimes *can* also be performed by using the arrow keys or the scroll wheel.

3. 8 double soft keys

   The instrument has 8 rows of left and right soft keys. Soft keys do not have a fixed function, but their function depends upon the content which is displayed next to them in the GUI. Depending on the displayed content, the right/left soft keys might have a different function. If there is just one function, both the right and left soft keys invoke this function. In this manual, the usage of soft keys is marked with <**Soft key**>.

4. Arrows keys with Select

   With the arrow keys, you usually move/navigate in blocks and tables. The center key (Select) is used to activate/select/use the currently focused item.

5. MEASUREMENT

   The primary function of these keys is to select a menu displaying measured values. The different menus are described in detail in MEASUREMENT MENUS [7.2→153].

   These keys are also used for entering numbers. In this case, the numerical value of each key is illuminated.

6. [**ENTER**], [**BACK**] and [**CANCEL**]

   [**CANCEL**] aborts the current selection, [**ENTER**] confirms. [**BACK**] is used to backspace while editing or switch to the previous menu from settings or sub-settings menus.

7. SETUP

   In the menus accessed with these keys, one can configure the measuring settings. Global settings are defined in [**INSTR.**]. Settings in group context are done in [**GROUP**] and individual channel parameters can be set up in [**CHANNEL**]These menus, as well as the [**APPS**](Setup) , [**ACTIONS**] and [**STORAGE**] menus, are described in detail in SETUP MENUS [7.3→175].

   [**HELP**]

   When the help function is activated, the keys LED will blink. Touching any displayed item on the screen will show up additional information about this item. Touching any other area will display an overview map. Pressing [**HELP**] again will deactivate the help mode and return to

the regular operation mode.

[**EXECUTE**]
If this key is illuminated, the action assigned in [**ACTIONS**] is executed. The key blinks during execution. If this key is not illuminated, no action will be performed.

[**FREEZE**]
This key allows one to freeze all *displayed* values. When this function is activated, the keys LED will be blinking. One can then switch between all the measurement menus to view all the values which had been simultaneously measured.

The setup menus, however, continue to display the current configuration state of the instrument. If for example auto range is activated and you press [**FREEZE**], the current automatically chosen range is displayed as the signal changes. In the same way, the status bar displays the current configuration.

Note that the interfaces are not affected by this function. By using this function, it is still possible to collect data using the interface and at the same time freeze something interesting on the screen.

8. Scroll wheel
   The scroll wheel can, depending on context, either scroll through a list, move graphical elements, or select different input elements. In the last case, pressing this wheel activates or changes the current setting.

9. USB ports
   The USB ports can be used to connect memory devices or keyboard and mouse. Their function is identical to the USB ports on the back panel. The USB symbol in the display status bar STATUS BAR [7.1→148] indicates on which ports memory devices have been detected. The associated port indicators flash regularly when devices are accessed. On LMG6x0 instruments this is also indicated by a LED directly at the USB port. In order to prevent data loss, memory devices should *not* be removed while these indicators (status bar or LED) are active.

10. [**TOUCH**]
    By using this key, the touch function of the touchscreen can be deactivated. This is useful to prevent unwanted actions when discussing displayed data on the screen. While the touch panel is deactivated, the LED beside the key is off to inform about this state.

11. [**STANDBY**]
    By pressing this key, a running instrument is safely shut down. See also SWITCHING ON/OFF [5.4→93]. When the standby state is active, the LED beside the key is highlighted. Pressing [**STANDBY**] will start the instrument.

5.3 Rear panel

The rear panels of the LMG671/LMG670/LMG640 instruments are almost identical. They consist of a base plug-in module, several channel plug-in modules and an optional PSI card. The LMG610 has a similar base plug-in at the right side while the measuring channel is accessible from the front panel.

5.3.1 Base plug-in module

The numbers in the base plug-in illustration in Figure 5.6 [→91] correspond to the following elements:

1. Additional protective conductor terminal
   The terminal for the protective conductor. See also CONNECTION TO PROTECTIVE EARTH AND POWER SUPPLY [3.3.1→34]. The area of connected solid or stranded wires must not exceed $10\,\mathrm{mm^2}$ or $6\,\mathrm{mm^2}$, respectively.

2. Plate with the serial number of the instrument.
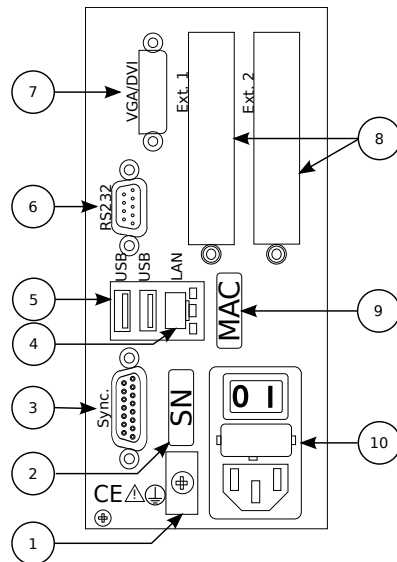
Figure 5.6: Elements of the base plug-in module

3. Sync.
   Connector for external synchronization. For details, see SYNCHRONIZATION CONNECTOR [4.3→46].

4. LAN
   LAN connector for remote controlling the instrument, see also REMOTE CONTROL [9→215].

5. USB
   USB connectors for connection to devices such as mouse, keyboard or storage devices.

6. RS232
   Connector for remote controlling of the instrument via the RS232 interface. See also REMOTE CONTROL [9→215].

7. VGA/DVI
   Connector for an external display or video projector. DVI can be used directly, VGA (LMG670/LMG640 only) requires an external off-the-shelf passive DVI/VGA adaptor.

8. Ext. 1, Ext. 2
   These slots are reserved for future extension boards.

9. Plate with the MAC address of the LAN interface

10. Power connector with power switch and integrated fuses
    Via this connector, the instrument is supplied with power. For further details, see SWITCHING ON/OFF [5.4→93], ENVIRONMENTAL CONDITIONS [4.1→39] and BASIC INSTRUMENT [4.2→39]. LMG610 has this power connector at the rear.

5.3.2 Channel plug-in modules

The numbers shown in the illustration of the channel plug-in module in Figure 5.7 [→92] correspond to the following elements:

1. ⊙**U**
   Voltage input, common low for ⊙**U*** and ⊙**U$_{\text{Sensor}}$**. 4 mm safety laboratory jack, black, surface mounted.

2. ⊙**U***
   Voltage input (high), 4 mm safety laboratory jack, yellow, surface mounted.
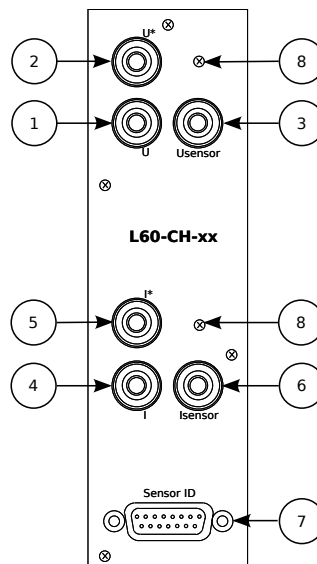
Figure 5.7: Elements of a single channel plug-in module

3. ⊙**U**_{**Sensor**}
   Voltage input for voltage sensor signals (high), 4 mm safety laboratory jack, red, surface mounted. This jack is only available with certain types of channels.

4. ⊙**I**
   Current input, common low for ⊙**I***and ⊙**I**_{**Sensor**}. 4 mm safety laboratory jack, grey, panel mounted.

5. ⊙**I***
   Current input (high), 4 mm safety laboratory jack, violet, panel mounted.

6. ⊙**I**_{**Sensor**}
   Voltage input for current sensor signals (high), 4 mm safety laboratory jack, red, panel mounted.

7. Sensor ID
   D-SUB 15 pin female connector for connecting current or voltage sensors. The stored information is transferred and power is supplied via this jack.

8. Nut which can be used to fix the cables which are plugged into the instrument. This is useful to prevent the signal from being accidentally interrupted. The diameter of the screw to be used must conform to M4 and the maximum length of the part to be inserted into the instrument must not exceed 7 mm.

5.3.3 Processing signal interface plug-in module

The illustration of the processing signal interface plug-in module in Figure 5.8 [→93] shows the following elements:

1. Analog Outputs
   32 voltage outputs, see ANALOG OUTPUTS [4.12.3→80]

2. Slow Analog Inputs
   8 slow voltage inputs, see SLOW ANALOG INPUTS [4.12.2→79]

3. Fast Analog Inputs
   2 fast voltage inputs, see FAST ANALOG INPUTS [4.12.1→79]

4. Speed/Torque/Frequency Inputs
   2 digital inputs , see SPEED/TORQUE/FREQUENCY INPUTS [4.12.6→82]
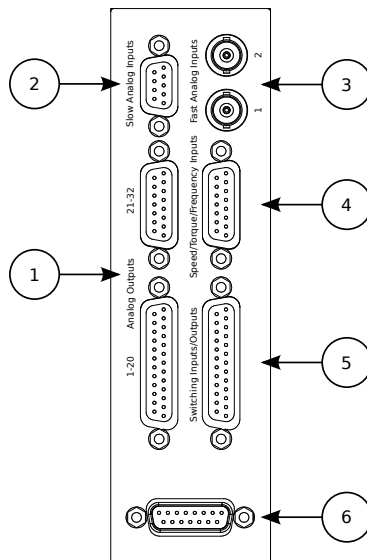
Figure 5.8: Elements of a processing signal interface plug-in module

5. Switching Inputs/Outputs
   8 digital inputs/outputs each, see Switching inputs [4.12.5→82] and Switching outputs [4.12.4→81]

6. Unused, do not connect anything.

5.4 Switching on/off

The instrument has two elements to control the operation mode. On the rear of the instrument, there is a real power switch. If this is set to off, the instrument is disconnected from the power line and does not consume any power. On the front, there is a standby key which is only active if the rear side power switch is in the on position.

The LM671 ATE can be switched on with the "Power" labeled button on the front. During boot-up, the LED blinks with short pulses and during shutdown, it blinks with long pulses. Normal operation is indicated by a steady light and an error condition by fast blinking. In the latter case, the cause of the problem can be investigated with the aid of an external monitor.

When switching the instrument off using the rear side power switch, the current operation state of the instrument is stored and restored when the instrument is switched on again. If the instrument is in standby mode and switched off, the instrument will remain in standby mode when it is switched on again. If the instrument is running and switched off (not recommended, see below), the instrument starts up after being switched on.

For correct standby-state handling, it is necessary to wait, after the rear switch has been used to switch off, until the standby LED turns off (at least 10s). Otherwise, it may be necessary to switch on the device with the standby switch.

The [**STANDBY**] key shuts down the instrument in a safe manner. That means that all open files are closed without losing data. Write caches are stored and will not be lost. When a running instrument is switched off with the back side power switch (or by switching off the power supply), then it is possible that some data (e.g. from open files) can not be saved and will therefore be lost.
If the instrument is in standby mode, it can be switched off using the power switch on the rear panel without losing any data.

The start-up of the instrument can take several seconds.

5.5 Maintenance

To maintain proper and safe operation, the following sections should be taken into consideration.

If the instrument has been opened, a high voltage test and a test of the protective conductor are necessary according to EN 61010 following the closing of the instrument for safety purposes prior to use. The instrument is only allowed to be opened in a pollution degree 2 environment or better!

5.5.1 Cleaning

Before cleaning, disconnect the device from all cables and ensure that there is no danger from open flames or other sources of ignition in the surrounding area.

For cleaning the device, especially the front panel, a soft cotton cloth with a small amount of isopropyl alcohol is suitable. Never apply the cleaning agent directly to the device; instead, apply it sparingly to the cloth. Do not place the container with the cleaning fluid on the device!

Allow the device to dry for 15 minutes afterward.

5.5.2 Calibration

One can only take advantage of the performance and precision of this high-end measuring instrument if the instrument is well adjusted and calibrated. We recommend that the instrument be adjusted and calibrated regularly by ZES ZIMMER. Calibration by third-party laboratories frequently causes problems that can easily be avoided.
If you allow the calibration to be performed by a third-party calibration lab, please pay special attention to the following issues:

- The reference instruments might not have the required accuracy, especially for active AC power. A very common error is that the reference sources, which some companies call 'calibrator', do not have the necessary uncertainty to calibrate this instrument. The calibrators are very useful for many hand held multimeters, but they are often worthless when calibrating active power. The Fluke 5500A calibrator is a typical, incorrect reference source for this instrument. Please keep in mind, that the reference instrument should be at least 3 times more accurate than the device under test. If not, the precision power meter calibrates the so called calibrator and not vice versa! The deviations are, in this case, the deviation of the calibrator and not the deviations of the instrument.

- The reference instrument may be traceable for voltage and current, but it is very rare that it is traceable for active AC power with the required uncertainty. Nevertheless it is common, that calibration labs calibrate active power even though only voltage and current are traceable. This happens very often, even in the context of accredited calibration services. In such cases, it regularly happens that the voltage and current are accredited but the active power is not accredited. Such protocols are worthless for active power!

If a calibration is performed by any third-party lab, the following points should be calibrated to ensure proper function of the instrument:

- Voltage and current of all ranges at a frequency near to 50 Hz.

- A representative selection of voltage/current range combinations to ensure a proper power reading

The ZES ZIMMER calibration service offers a traceable calibration of all relevant parameters and meets the requirements of ISO 17025/IEC 17025. Our active power calibration is directly traceable to the Physikalisch Technische Bundesanstalt (the German National Laboratory) in Braunschweig, Germany.

A further advantage of our calibration is that, in case a defect is identified and service becomes necessary, there are no additional shipping costs and delay times.

Requirements for reference instruments

As generally known, the references (calibration sources and/or reference power meters) have to be in an uncertainty class which is at least 3 times better than the instrument under calibration. Ideally, the uncertainty class of the reference should be between 5 to 10 times better. For the specified accuracies of this instrument, please refer to the TECHNICAL DATA [4→39] section.

A calibration never requires the instrument to be opened.

5.5.3 Adjustment

The adjustment has to be done at $(23\pm1)\,°C$.

ZES ZIMMER offers the adjustment of instruments. Please contact `sales@zes.com` for further information.

5.5.4 Battery

A CR2032-type battery is used in the instrument for preserving data such as the system time while the instrument is switched off.

The battery should be replaced after 3 years or when any problems occur. This battery is soldered, so it is recommended that the replacement of the battery be performed at a ZES ZIMMER service department, for example during calibration of the instrument.

If this replacement is performed outside of a ZES ZIMMER service department, please adhere to the safety check as described at the beginning of this section, because the replacement does require the instrument to be opened.

5.5.5 Air filter

Be sure that the instrument is disconnected from *all* cables before removing the filter holders and that the filter holders are mounted properly before connecting the instrument again.
Only the already mounted screws should be used, longer srews are not alowed!

The air filters are located at the bottom and right hand side (front view) of the instrument. They should be cleaned regularly, depending on the degree of pollution of the environment. To do this, remove the filter holders, clean the filters and remount everything.

Cleaning of the air filters does not require the instrument to be opened.

5.5.6 Fuses

The mains supply fuse is the only fuse in the instrument which is for the protection of the instrument and can be replaced by the user. *All other* fuses , especially those on the measuring channels are *not* for the protection of the instrument but for the protection of the user in case of misuse! If such a fuse has blown it is very likely that the instrument was damaged by the misuse. In this case, the replacement of such a fuse is strictly forbidden! The instrument must not be operated and must be checked for proper and - most importantly - safe operation by a competent service station.

Mains supply fuse

The mains supply fuse is the only fuse in the instrument which is for the protection of the instrument and which can be replaced by the user in case it is defective. This fuse is located in the IEC inlet and can be replaced by a new fuse of the same type. The correct type can be found in MAINS SUPPLY LMG670 / LMG671 [4.2.1→40]. If you are having difficulties acquiring the required fuse, please contact the ZES ZIMMER sales department. The instrument does not have to be opened during replacement of the fuse.

5.5.7 Touchscreen adjustment

The touchscreen is adjusted properly when the instrument is delivered. However, the optimal position of the touch depends slightly on the viewing position of the user (i.e. parallax error). A taller user might require the touchscreen to be adjusted slightly differently than a smaller user. Also, if the instrument is mounted on a higher or lower position in a rack, another adjustment might be necessary.

To adjust the touch, see TOUCHSCREEN TAB [7.3.11→182].

5.5.8 Software update

For the update, the LMG671 ATE needs to be connected to an external monitor and to a USB input device (a keyboard for Boot Stick Method or a mouse for File Copy Method).

The firmware update can be done in two ways as explained in the following sections. In each case the stored data (settings, custom menus, measurement data,...) should be saved prior to the update (see CONFIGURATION TAB [7.3.19→193]).

Firmware file copy method (recommended)

Please be aware that firmware files older then 3.000 are not compatible with this method.

Please note: This procedure can only be followed on the devices GUI.

- Copy the downloaded firmware image (from http://www.zes.com/firmware) file on a USB stick.

- Plug the USB stick in the LMG and open the "File" tab in the "STORAGE" menu of the internal GUI.

- Navigate to the firmware file. If selected and successfully tested for being a valid firmware image an "Install Firmware" softkey shows up.

- Hit the softkey and follow the instructions.

- The installation can take some minutes and is finished after reappearance of the graphical user interface.

Not enough disk space available

If there is not enough space available on the device for the latter two installation methods, it's possible to free space by deleting unused files (e.g. recordings) from the internal hard disc. Alternatively it is possible to use the "Boot stick method" which has less internal disk space requirements. The space requirements must only be met during the installation process.

In case the device doesn't reboot after a while or in case of any malfunction during installation, the device can always be reinstalled using the "Boot stick method"

Boot stick method

- The usb storage device must have a size of 4 GB at least. It is *not* necessary to format it.

- Excract the ZIP file with the LMG600 firmware image, which you have received from ZES ZIMMER.

- Start "balenaEtcher". At first choose the downloaded image file by clicking on the button "Flash from file".

  The image file must not be stored on a network drive but on a local drive to ensure an uninterrupted image writing

  Next step is to choose the USB storage device which should be used for the update by clicking "Select target". If the correct device is not displayed please prepare it as described below. Start the procedure with "Flash!", this can take few minutes. Finally exit the program.

- If your currently installed LMG firmware has a version number less than or equal to 1.016, an external keyboard is required to install a software update. With later firmware versions (that is >= 1.017), the built-in keys on the front panel can be used instead. Connect the USB storage device and, as appropriate, a USB keyboard to the LMG. If an external keyboard is used, start the instrument while pressing and holding the F11 key. When using the front panel keys, start the instrument while pressing and holding [**CANCEL**] and repeatedly pressing [**APPS**](Measure) . In both cases, a dialog should appear with a list of possible boot devices. Use the arrow keys to select the appropriate USB storage device prepared earlier and press ENTER or [**ENTER**], respectively, to start the software update, then follow the on-screen instructions.

  The USB storage device is usually the second entry in the list. It's text is build from manufacturer informations. The items starting with "P0: ..." or "P4: ..." are the internal drive. If this is chosen accidently, the instrument boots in the usual way instead of updating. Entries similar to "UEFI ..." as well as the last entry "Enter Setup" shall not be used.

  When the update procedure has finished the LMG restarts and is then ready for use.

Preparation of the USB storage

The content of this storage will be deleted. Please back it up before continuing
After writing an update image onto the stick, Windows will not detect the USB storage any more. To keep using it, one has to follow the following procedure. In case there is any doubt, consult your system administrator.

If the USB storage is not recognized by Windows or "balenaEtcher", one can try the following with Windows 7:

In the start menu choose "Control Panel". In the top right corner is a selector named "View by:" where a few different possibilities are listed:

- Big icons *or* Small icons
  One can directly click on "Administrative Tools ".

- Category
  Here, one needs to first select "System and Security" and then "Administrative Tools".

Then click on "Computer Management" and then "Disk Management".

All connected storage are listed in the lower pane of the middle area. First, delete all partitions of the USB storage. Ensure that you have selected the correct storage (as the hard drives of the computer are also displayed!), because all data will be deleted!

Click with the right mouse button on all existing partitions of the USB storage and choose "Delete Volume" if this point is available. Then click with the right mouse button on the remaining partition and select "New simple volume". Using the assistant wizard, select the full size and assign a drive letter. If the storage is going to be used with "balenaEtcher", one can select "Do not format this volume" to save time. If the storage should be used under Windows, it must be formated.

After the wizard is finished, the storage is now ready to be recognized by Windows and to be used by "balenaEtcher".

5.5.9 In case of trouble

There can be two kinds of troubles: Either measured values are not plausible or the software crashed.

Values not plausible

This case can have many different reasons, starting from a wrong setup and ending in an hardware defect in the instrument. In this case it is possible to generate a status report which includes all relevant settings and measurements. The detailed description how to do this can be found at the description of the STATUS REPORT TAB [7.3.21→193].

> *At a glance*
> *Press [**STORAGE**] and go to the status report tab to create the report.*
>
> *Please always add as many information as possible: Which values were expected? What is the uncertainty for this values? How was the instrument connected? ...*

The created PDF file can be copied to an USB stick (see FILES TAB [7.3.20→193]) and can be sent via email to your local service point.

Software crashed

If the software crashed the instrument displays a maintenance menu after some seconds. If it doesn't and/or restarting the device doesn't work either, you can do the following:

1. Turn off the instrument.

2. Turn it back on.

3. Press and hold down the MISC key during the startup phase as soon as the progress bar appears.

4. The instrument will go directly into maintenance mode.

There some actions can be chosen:

1. Restart system
   Starts the measuring system again.

2. Shut down system
   Switches off the instrument.

3. Copy event and fault records on USB stick
   Reads the fault memory and copies it to an USB flash drive.

4. Copy full system status on USB stick
   Reads all the important data of the system and copies them to an USB flash drive.

5. System reset
   All settings are reset to factory defaults. All custom changes are lost.

6. Config reset
   All settings are reset to default values.

7. Front unit test
   A test procedure for the front unit. This should only be used by instructed persons or on the instructions of ZES.
   ATTENTION: The "USB/S-ATA Test" will delete all data on any connected USB mass storage device (e.g. usb flash drives) connected to Front-USB1 or -USB2.

The maintenance mode can also be accessed from the running system by pressing <**Maintenance Mode**> in the [**MISC.**] menu.

Usually it is sufficient to use action 3.

The generated file has a name of Type "zes_*". Please send this file via email to your local service point and add a description under which circumstances the software crashed. If there are several effects, please note the time when this effects happened, because the entries in the fault memory have timestamps and can be correlated to the observed effects.

To execute, press 3, connect the USB stick and press [**ENTER**]. After the transfer a message appears. Press [**ENTER**]again and switch off the instrument by pressing 2 and [**ENTER**].

5.6 Security advises for network usage

The LMG600 provides high-speed data transfer over a Gigabit Ethernet connection and allows general network usage as well as direct connections.

In order to asses the compliance with your local network security requirements, we describe the network behaviour of the LMG600 as follows:

LMG600 devices communicate over the network for the following purposes only:

1. The device requests the local network settings and an IP address via DHCP. This behavior can be deactivated.

2. Network services are advertised via UDP port 5353.
   (According to IETF RFC 6762 Multicast DNS & 6763 DNS-Based Service Discovery)
   In particular there are these two services "_workstation._tcp." and "_zeslmg._tcp." (Interface communication with the LMG).

3. The device answers ICMP pings in order to be detectable via its IP address.

4. Interface communication via TCP ports 5025 and 5026.
   Text-based SCPI and control communication can be initiated externally from this ports. Port 5026 enables you to reset the connection on port 5025 (see RESETTING THE INTERFACE [9.9.1→229]).

These services are being maintained and updated by our firmware updates. For security reasons we deliberately keep a low communication profile in order to minimize the ways of possible attacks. If the LMG600 does not comply with your network security standards, you can still consider using it for 1:1 network connections, excluding your network completely.

We can not, however, make any guarantees for the behavior after improper usage of the device.

6 Basic concepts

This chapter describes the basic concepts used by the instrument and should be known when working with the instrument.

6.1 Groups

In order to reflect the physical relation between channels or the physical circuits being measured the power channels (P channels) are organized into *groups*. The criterion for the grouping of P channels is the electrical system to be measured. This in turn translates into two physical criteria: the number of phases and the number of wires.

Groups

Most common electrical systems have either 1 phase (1ϕ) or 3 phases (3ϕ). 2 phase systems (2ϕ) are less common and systems with 4 (4ϕ) or more phases are very rare. Nevertheless, the instrument supports all of these configurations. The wires constitute the physical medium for the phases. Note that protective conductors usually (and also in this manual) do *not* count as wires! One-phase systems (1ϕ) always use 2 wires (2W), while two-phase systems (2ϕ) may have two (2W) or three (3W) wires and three-phase systems (3ϕ) three (3W) or four (4W) wires. Table 6.1 [→101] lists the most common groupings. A description of how to configure the instrument accordingly can be found in Setup menus [7.3→175].

A crucial requirement for all P channel members of a group is that their respective signals have the same fundamental frequency. This is generally the case for the electrical systems described in this manual. If this requirement is not met, a lot of the statements and explanations given in this manual *do not apply!* Therefore, any deviation from this rule should be well justified by expertise.

As an additional constraint, all channels in a group must be of the same type. This serves to prevent subtle issues arising from different behavior due to technical differences between channel types.

Depending on the task at hand and the number of available P channels, one or more groups have to be configured. Each P channel can be a member of exactly one group at all times; ungrouped P channels are not permitted.

While the grouping specifies which channels to combine, the aptly named wiring setting tells the instrument how the conductors of a group are wired up. The wiring therefore determines how the measured signals are to be interpreted by the instrument. So given the assumption of a common fundamental frequency and a correct wiring setting, a group has the following noteworthy properties:

WIRE [9.10.353→383]

- A single synchronization source is used for all P channels in the group. Its frequency is the group's (sync) frequency.

- Various sum values, e.g. total active power, are calculated over all P channels in the group.

1 channel	1ϕ 2W for 1 phase system with two wires
2 channels	1ϕ 3W for 1 phase systems with three wires (split phase) *or* 3ϕ 3W for 3 phase systems with 3 wires (Aron circuit)
3 channels	3ϕ 3W for 3 phase systems with 3 wires *or* 3ϕ 4W for 3 phase systems with 4 wires

Table 6.1: Number of channels in a group and typical systems to be measured

- The instrument applies the various group settings (e.g. filters and coupling) uniformly to all P channels of the group. This eliminates a large number of possible errors due to differing configurations within the same group, which might otherwise lead to dubious results when combining values from such channels.

In the following sections, typical ways to wire *one* group are described.

If you have a more complex system, each group is wired on its own, independent from the other groups. For example, having a frequency converter with a single-phase input and a three-phase output, you can connect the single-phase system as shown in the appropriate section, and independently from that, the three-phase system as shown in its appropriate section.

The principle wirings are shown with direct measurement of current and voltage. But it is also possible to use voltage and/or current sensors/transformers. How to install them in principle is shown in WIRING USING EXTERNAL CURRENT SENSORS [6.2.4→104] and WIRING USING EXTERNAL VOLTAGE SENSORS [6.2.5→106].

6.2 Connection

Before connecting the instrument ensure that all safety requirements are fulfilled. Please regard especially SAFETY INSTRUCTIONS AND RISKS [3.3→33], CONNECTION TO PROTECTIVE EARTH AND POWER SUPPLY [3.3.1→34] and CONNECTION TO THE EUT CIRCUIT [3.3.2→35].

6.2.1 Wiring of single-channel groups

1ϕ 2W

For single-phase measurement, it is usually recommended to measure the current in the wire which is closest to earth, see Figure 6.1 [→102]. This optimizes the common mode rejection. This circuit measures the voltage and current correctly relative to the load and source, respectively.
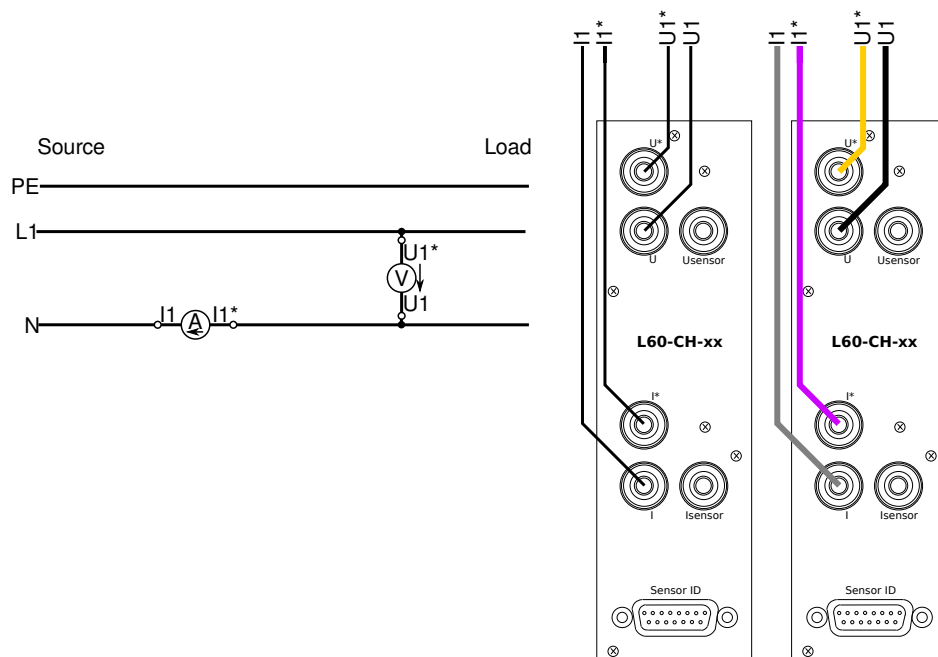
Figure 6.1: Single channel with wiring 1ϕ 2W shown as circuit, logical connection and practical connection with coloured wires

6.2.2 Wiring of dual channel groups

Split phase system, 1ϕ 3W

This system (Figure 6.2 [→103]) is often used in northern America. This is sometimes incorrectly called a two-phase system (2ϕ 3W) , but this is technically not correct. This circuit measures the voltage and current correctly relative to the load and source, respectively.
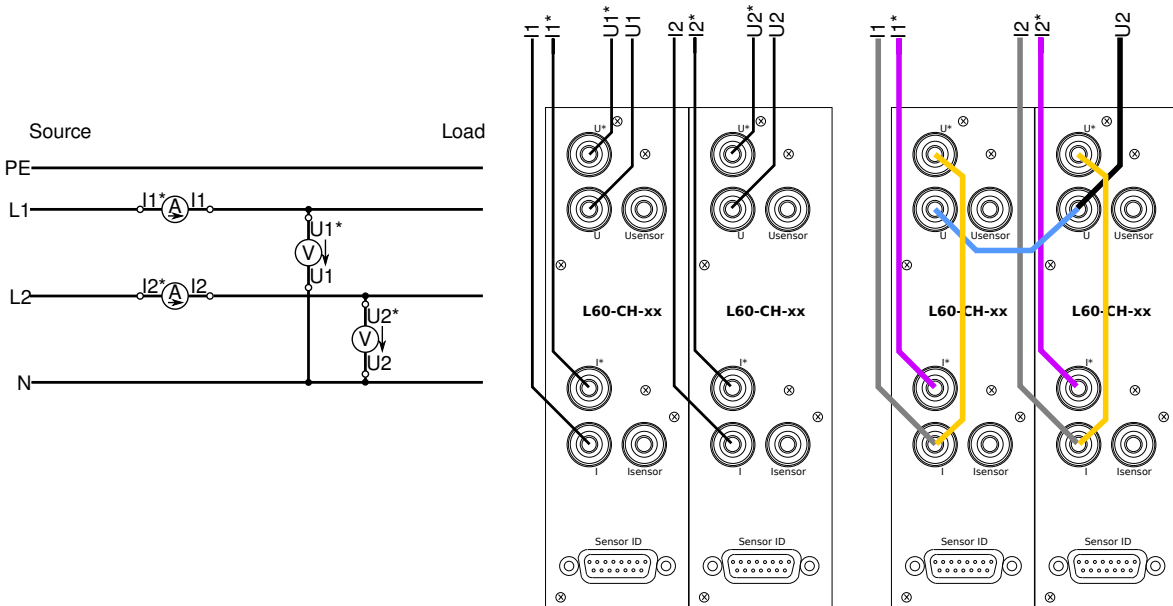
Figure 6.2: 2 channels with wiring 1ϕ 3W, split phase shown as circuit, logical connection and practical connection with coloured wires

Three-phase system, Aron circuit, 3ϕ 3W

Aron wiring (Figure 6.3 [→103]) works only if the system is a real three-wire system. If you have for example a frequency converter and a significant current is flowing from the motor by capacitive coupling to earth and then back to the converter, earth would be a fourth wire. In this case, the requirements for an Aron circuit are not met and the results might be wrong.
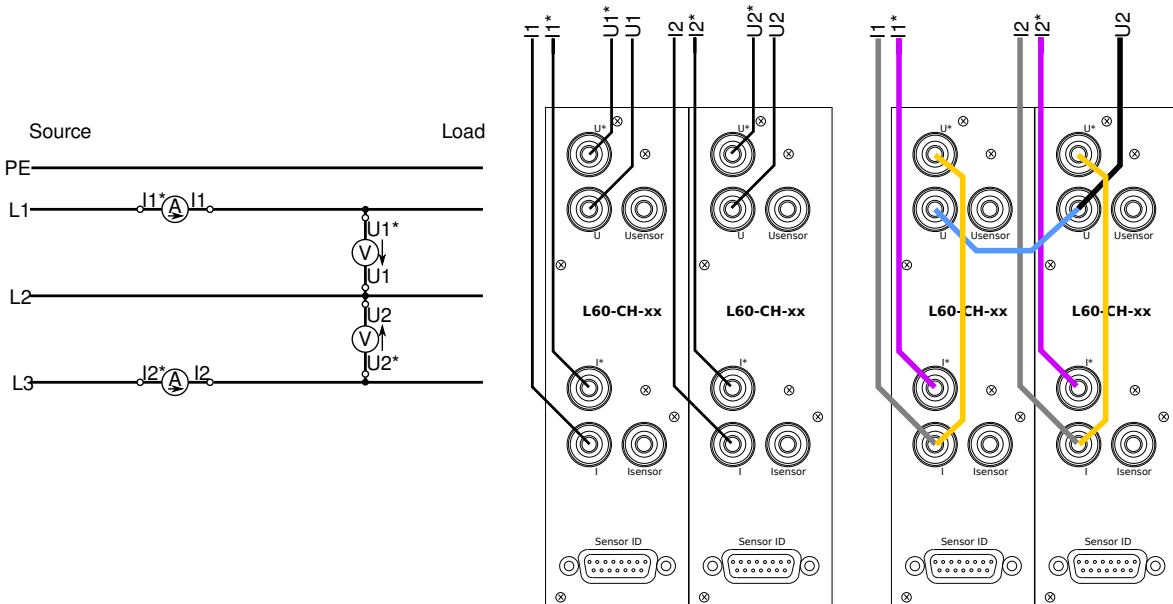
Figure 6.3: 2 channels with wiring 3ϕ 3W, Aron circuit shown as circuit, logical connection and practical connection with coloured wires

When selecting Aron wiring, the instrument will calculate the third missing voltage and current. As default the total active power of the system is the only valid sum value of the group.

When using the installed option star-delta conversion, the complete system is measured and it is possible to recalculate the whole circuit to star or delta. This feature adds to the measurement of the total active power of the system (which is the usual limitation of the Aron circuit) the measurement of all values such as power factor, reactive power, etc.

This conversion is available for LMG670/LMG671 only, but not for LMG640/LMG641.

To set up this calculation, please see GROUP MENU [7.3.12→182].

6.2.3 Wiring of triple channel groups

System with neutral wire N, 3ϕ 4W

When the neutral wire N is accessible, the wiring according to Figure 6.4 [→104] is recommended, because you would not need the star-delta conversion to get valid values. When values are measured in a star circuit, it is possible to calculate the delta values by star-delta conversion.
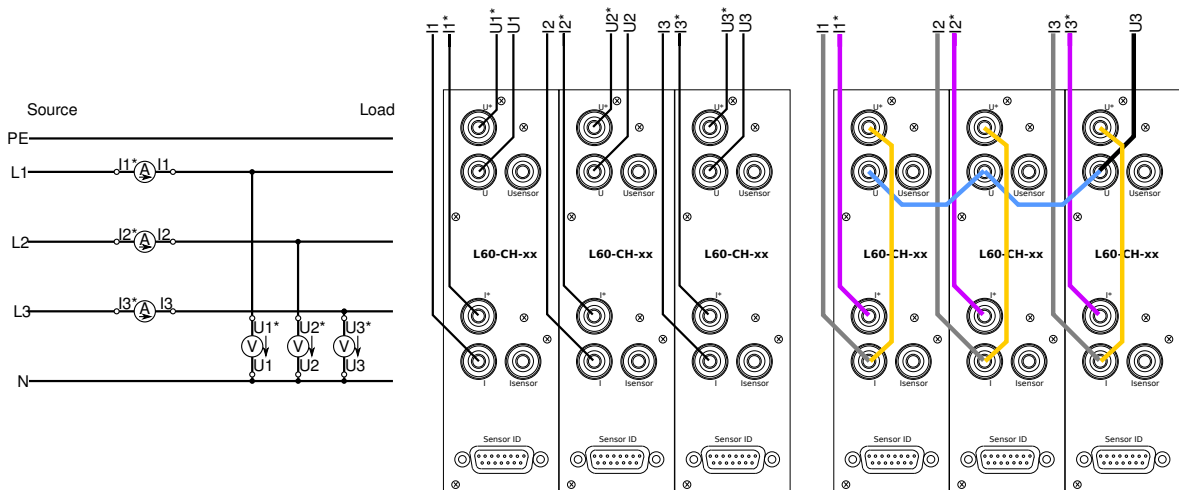
Figure 6.4: 3 channels with wiring 3ϕ 4W, Uλ, Iλ shown as circuit, logical connection and practical connection with coloured wires

System without a neutral wire N, 3ϕ 3W

When the neutral wire N is *not* accessible, a wiring according to Figure 6.5 [→105] is recommended. In this circuit, the problem is that the voltage and current which are applied to a power measurement channel will never apply to any component at the same time. So the product of voltage and current is no longer a real power but a useless phantom power. The solution, in this case, is that the instrument can either recalculate the star voltage from the delta voltage or calculate the delta current from the star current. By this approach, voltage and current apply to a real existing component and become valid. By using brute force calculation, it is possible to measure such systems.

6.2.4 Wiring using external current sensors

Instead of direct current measurement, it is also possible to use external sensors like transformers, transducers, current clamps or shunts. Figure 6.6 [→105] illustrates how they are connected. All shown measurement methods are identical concerning the polarity.

In this example, each of the four channels measure the same current. This current-measurement principle can be transferred to the previously discussed wirings for one to three channels.
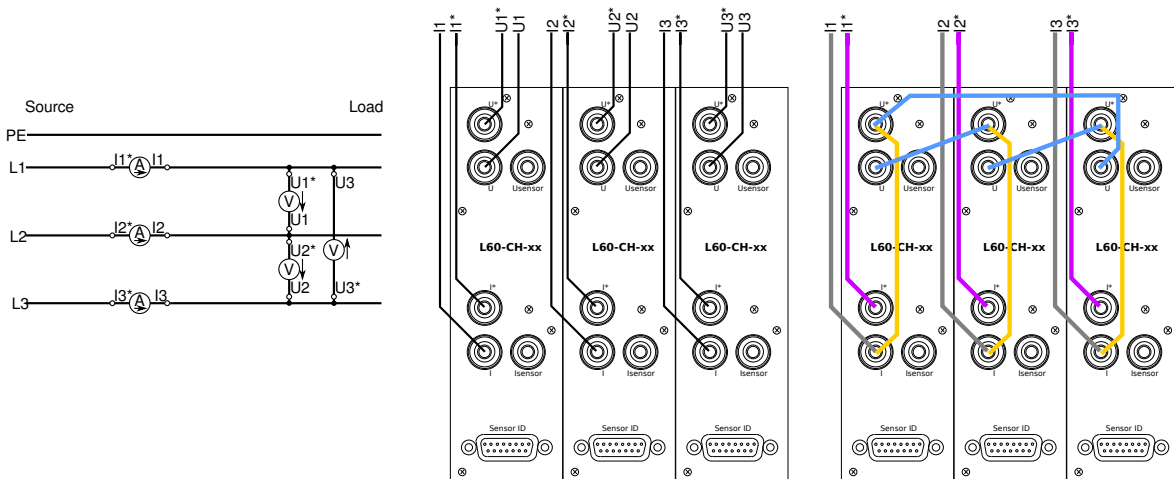
Figure 6.5: 3 channels with wiring 3φ 3W, UΔ, Iλ shown as circuit, logical connection and practical connection with coloured wires
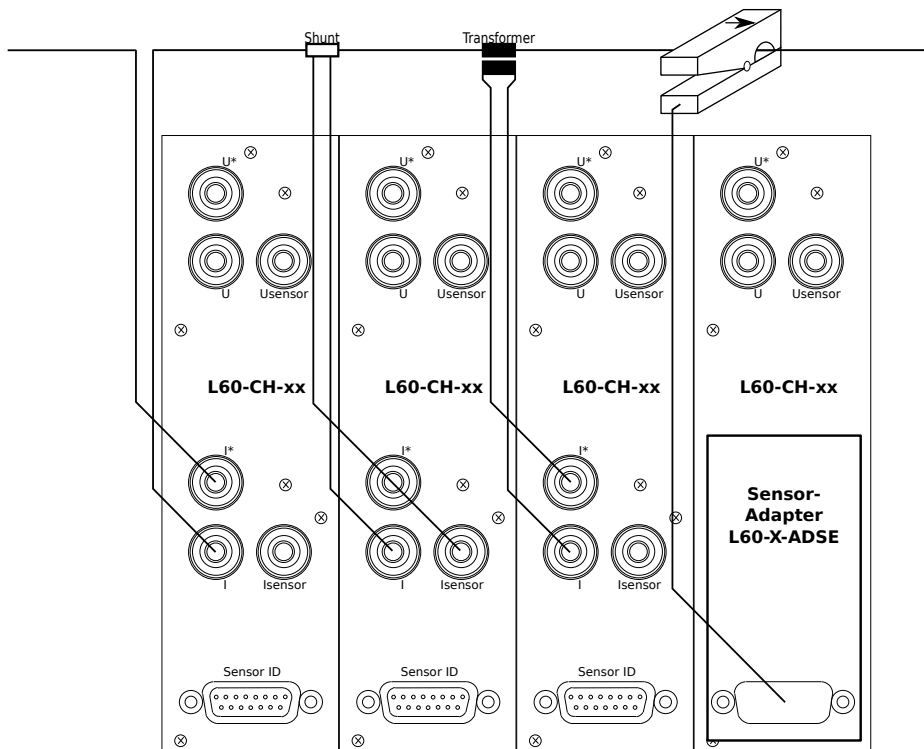
Figure 6.6: Comparing different current measurement techniques

Depending on the output signal they are connected to the current input or to the sensor input of the current channel. Configuration of the sensor scaling is described in SENSOR MENU [7.3.14→187].

Which kind of sensors to use depends on the environment. The best uncertainty is achieved with direct measurement, but this requires that the circuit can be interrupted. Clamps have a worse uncertainty, but they can be used without interrupting the circuit.

For special Plug'n Play sensors the adaptor L60-X-ADSE is available.

> *If you use shielded wires and use clamps or similar, please make sure to exclude the shield current from the measuring.*

Plugging in a Plug'n Play sensor will change the SCPL [9.10.222→321] setting of the channel's group to the value specified by the sensor. The setting can be changed at will afterwards and will not be changed automatically again until another Plug'n Play sensor is attached.

6.2.5 Wiring using external voltage sensors

Instead of direct voltage measurement, it is also possible to use external transformers or voltage dividers. How to connect them is shown in Figure 6.7 [→106]. All shown measurement methods are identical concerning the polarity.
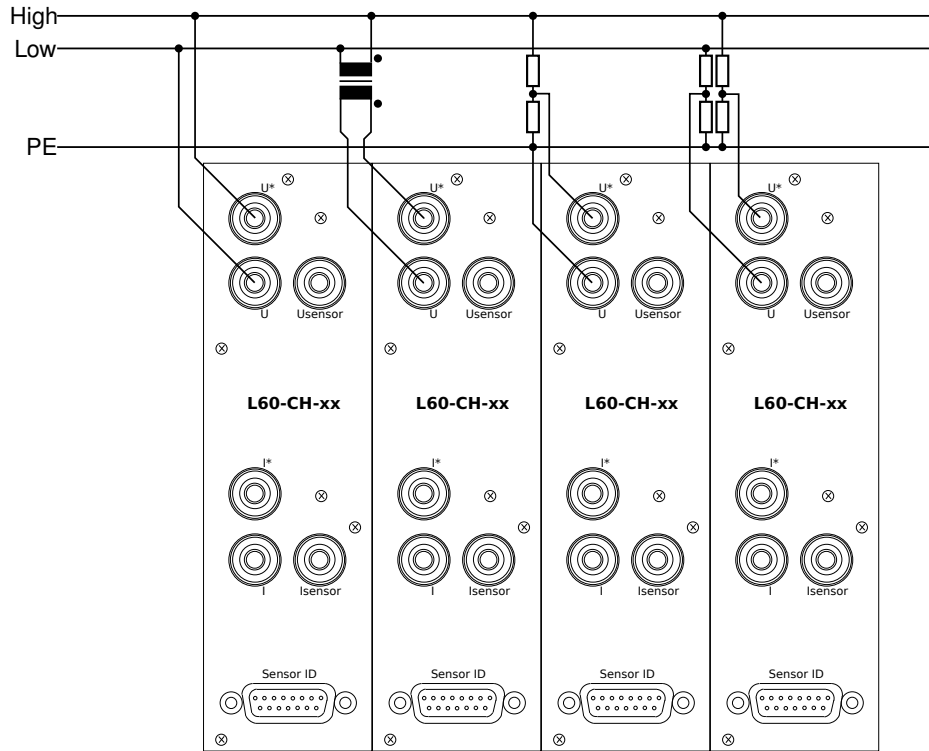
Figure 6.7: Comparing different voltage measurement techniques

In this example, each of the four channels measures the same voltage. This voltage-measurement principle can be transferred to the previously discussed wirings for one to three channels.

Depending on the voltage level, use of the jack ⊙**U_{Sensor}**(not available with all channel types, see POWER MEASUREMENT CHANNEL L60-CH-A1 AND L60-CH-A2 AND L60-CH-A2.1 [4.7→55]) instead of the jack ⊙**U***could result in a better usage of range and thus in a lower uncertainty. Configuration of the sensor scaling is described in SENSOR MENU [7.3.14→187].

The usage of one single high-voltage divider (as shown in the third channel in Figure 6.7 [→106]) is only possible when "Low" and "PE" are the same potential. If they differ, you have to use two high-voltage dividers to measure the difference between them. This is shown at the last channel in Figure 6.7 [→106].

6.3 Star-delta conversion, option L6-OPT-SDC

There are two general ways to measure a three-phase system. One can either measure in the delta circuit (Δ) or in the star circuit (λ).

For measuring voltages, the star circuit requires the presence of a fourth wire, usually the neutral wire N. Each phase is measured against this common wire. If this neutral wire is missing (which is typically

for motors or frequency converters), there is no other choice than to measure in the delta circuit (i.e. between the phases).

For measuring currents, it is usually only possible to measure the current in each phase by measuring the star circuit. A measurement of the delta circuit is theoretically possible, but this would require to open the measurement object which is usually neither desired nor possible.

Usually the measurement takes place on the wires between the power source and the equipment under test (EUT). Both are usually black boxes. So there is just one possible wiring for the current measurement and two for the voltage measurement.

Figure 6.4 [→104] shows the wiring UλIλ, Figure 6.5 [→105] shows the wiring UΔIλ. With both wirings it is possible to calculate and display UλIλ and UΔIΔ.

Please keep the following properties of the star-delta conversion in mind:

- While the directly measured voltages and currents in wiring UΔIλ are, of course, correct, the power calculated from this voltage and current does not exist at any component in the circuit. Either you have the correct voltage of a component but the wrong current or you have the correct current but the wrong voltage.
  It is therefore a phantom value which is rejected to prevent confusion.

- When having a 4-wire system and a current is flowing in the neutral conductor N, it does not make sense to convert this circuit into a delta system because a delta system has 3 wires only.

- When converting a voltage which is measured in delta circuit to star circuit, the common midpoint is assumed to be in the middle of the triangle.
  If the real existing midpoint is shifted due to unsymmetrical load, this cannot be recognized from outside.
  The effect with such an unsymmetrical load would be, that the single-phase power would not meet the real values, but the total power of the system is measured correctly.

The formulas for this calculations are based on sample values and are thus independent from any asymmetries, in amplitude, as well as in phase angle or harmonic distortion. The indices in the following formulas relate to the number of the phase:

Star-delta conversion, 3 channels, Uλ to UΔ and Iλ to IΔ

$$u_{12}(t) = u_1(t) - u_2(t) \tag{6.1}$$
$$u_{23}(t) = u_2(t) - u_3(t) \tag{6.2}$$
$$u_{31}(t) = u_3(t) - u_1(t) \tag{6.3}$$

$$i_{12}(t) = \frac{i_1(t) - i_2(t)}{3} \tag{6.4}$$
$$i_{23}(t) = \frac{i_2(t) - i_3(t)}{3} \tag{6.5}$$
$$i_{31}(t) = \frac{i_3(t) - i_1(t)}{3} \tag{6.6}$$

Delta-star conversion, 3 channels, UΔ to Uλ

$$u_1(t) = \frac{u_{12}(t) - u_{31}(t)}{3} \tag{6.7}$$
$$u_2(t) = \frac{u_{23}(t) - u_{12}(t)}{3} \tag{6.8}$$
$$u_3(t) = \frac{u_{31}(t) - u_{23}(t)}{3} \tag{6.9}$$

The conversion from current in delta circuit to star circuit is not supported, as it is usually not possible to measure such currents.

A special feature is the conversion to star or delta, when the measurement was done in Aron circuit:

Conversion to delta, Aron circuit, I⅄ to IΔ

$$u_{12}(t) = u_{12}(t) \tag{6.10}$$
$$u_{23}(t) = -u_{32}(t) \tag{6.11}$$
$$u_{31}(t) = u_{32}(t) - u_{12}(t) \tag{6.12}$$

$$i_{12}(t) = \frac{2 * i_1(t) + i_3(t)}{3} \tag{6.13}$$
$$i_{23}(t) = \frac{-i_1(t) - 2 * i_3(t)}{3} \tag{6.14}$$
$$i_{31}(t) = \frac{-i_1(t) + i_3(t)}{3} \tag{6.15}$$

Conversion to star, Aron circuit, UΔ to U⅄

$$u_1(t) = \frac{2 * u_{12}(t) - u_{32}(t)}{3} \tag{6.16}$$
$$u_2(t) = \frac{-u_{12}(t) - u_{32}(t)}{3} \tag{6.17}$$
$$u_3(t) = \frac{-u_{12}(t) + 2 * u_{32}(t)}{3} \tag{6.18}$$

$$i_1(t) = i_1(t) \tag{6.19}$$
$$i_2(t) = -i_1(t) - i_3(t) \tag{6.20}$$
$$i_3(t) = i_3 \tag{6.21}$$

These conversions of the Aron wiring are available for LMG670/LMG671 only, but not for LMG640/LMG641.

6.4 Sum channels

In the sum channel, the values U_{trms} , I_{trms} , P, Q, S, and PF are available. The first three are computed in a different way (i indicates the channel number)

$$U_{trms} = \sqrt{\sum_i \left(U_{trms}^{(i)}\right)^2}, \tag{6.22}$$

$$I_{trms} = \sqrt{\sum_i \left(I_{trms}^{(i)}\right)^2}, \tag{6.23}$$

$$P = \sum_i P^{(i)}, \tag{6.24}$$

and the others as usually from the first three

$$S = U_{trms} \cdot I_{trms}, \tag{6.25}$$

$$Q = \sqrt{S^2 - P^2}, \tag{6.26}$$

$$PF = \frac{|P|}{S}. \tag{6.27}$$

If the wiring is direct, the sum runs over the hardware channels of the group. If a wiring is available, the sum runs over the star-star values.

The sum values are accessible like the normal values by a particlar suffix, see LOGICAL SUFFIXES [9.2.4→218] for it. According to DIN 40110, the sum values are only defined for U_{trms}, I_{trms}, P, Q, S, and PF. The remainung sum values are set to NaN. In the GUI were are displayed as bars.

6.5 Functional block diagrams

The functional block diagram in Figure 6.8 [→109] shows an overview of a power measurement channel which consists of a voltage and current measurement section. This overview can be used as an aid to understand the basic concepts described in the following sections.
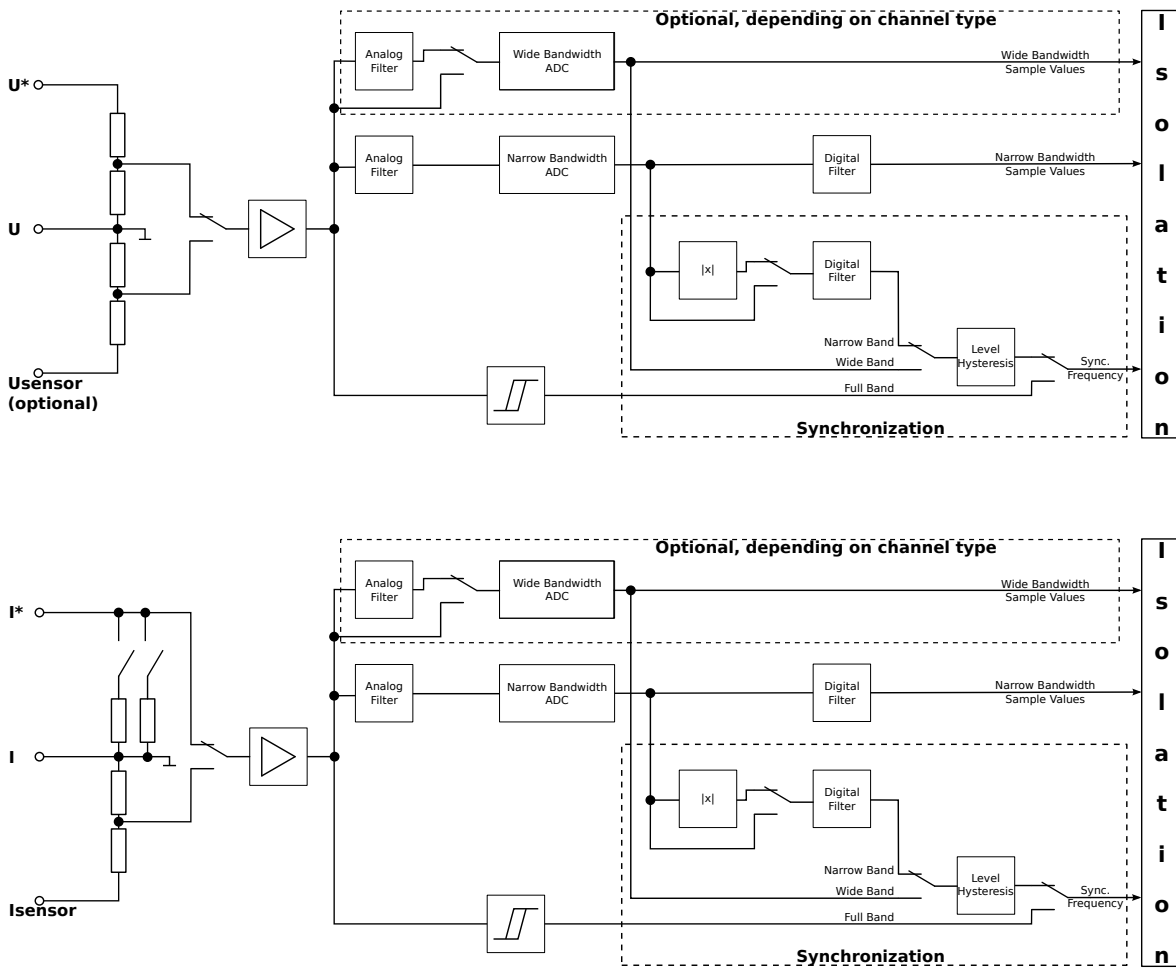
Figure 6.8: Functional block diagram of a power measurement channel

6.6 Bandwidth

Especially when measuring drives and frequency converters, it is a frequent task to measure values of different bandwidths at the same time. For example, one would measure wide bandwidth values caused by the converter and narrow bandwidth values which are relevant for the mechanical power.

This instrument is the first high end power analyzer on the market which offers *DualPath* processing solution. Previously, one was faced with the drawback of having just one analog-digital converter (ADC) and had to decide between one of four unfavorable possibilities:

DualPath
processing
`PROC [9.10.275→345]`

- One could measure with filtering switched *on* to prevent aliasing when performing an fast Fourier transform (FFT). However, wide bandwidth values would not be acquired.

- One could measure with filtering switched *off* to get wide bandwidth values, but one would not acquire the FFT results.

- One could measure with filtering switched *off* to get wide bandwidth values, and ignore the risk of aliasing. Here, the quality of FFT values would be questionable.

- One could switch the filter on and off and measure the currently valid value. This is unfortunately only possible when one has steady state conditions and takes a lot of time.

In the *DualPath* solution (as shown in Figure 6.8 [→109]), exists a common signal path from the input jacks. The input signal is split into two and fed into the narrow bandwidth converter and the wide bandwidth converter. The signal processing unit calculates the values of both converters in parallel: Wide bandwidth values (marked with symbol ⎍⎍⎍) and narrow bandwidth values (marked with symbol ⊥). So this technically elaborate solution eliminates the problems listed above which were encountered in previous generations of instruments. Furthermore, it offers additional advantages over previous solutions which, while offering wide and narrow bandwidth values in "parallel", measured usually just the fundamental of the motor. This new approach allows the user to decide whether only the pure fundamental of the motor (via FFT) should be calculated or if the motor harmonics by acquiring narrow bandwidth values should be included.

This innovative feature was mainly developed for drive applications. Many other applications have to be measured by using just a single bandwidth. Due to this, one can configure the instrument to work either in single-mode or *DualPath* processing mode. The capability to work in *DualPath* processing mode is a property of the measurement channel, so this feature is not available in groups which consist of channels not supporting *DualPath*.

As most of the applications (except frequency converters) are single-bandwidth applications, single-bandwidth processing mode is the default setup. When requesting measurement values in this mode, it does not matter if one requests the wide or narrow bandwidth values, as the instrument delivers the same correct values for both requests.

For further configuration details, please see SIGNAL SETTINGS [7.3.12→183].

6.6.1 Filter

An important difference between C-VALUES [→113] (e.g. U_{trms}, I_{trms} or P) and H-VALUES [→113] is that the C-values *can* be filtered by the user. It is useful to configure filters in a way which is reasonable for the respective application. On the other hand, there are some requirements for the filter that must be met due to the sampling theorem (i.e. Nyquist-Shannon theorem). Depending on the sampling rate, a special filter can be required. There can be a contradiction between the users intention and the required FFT setup of the instrument.

The solution to this dilemma is to let the user decide how to configure the filters:

- Manual
  The main goal is the manual selection of a filter which is suitable for an application. So the user can define filters for this purpose. The FFT, running under this circumstances, has the following characteristics:
  (i) The filter can not be set automatically, so the instrument decides, which harmonic evaluations are possible with the given filter.
  (ii) If its frequency is too high, fewer harmonic values or possibly no harmonic values are calculated.
  (iii) If its frequency is too low, a calculation might be impossible due to aliasing.

- Automatic
  Here, the main goal is to calculate as many orders of harmonics as possible. For this purpose, the bandwidth of the measurement is determined by the automatically configured filter of the FFT.

Configuration of the filter is related to BANDWIDTH [6.6→110] and ANTI-ALIASING [6.6.2→111] and can be configured as shown in SIGNAL SETTINGS [7.3.12→183].

6.6.2 Anti-aliasing

As previously stated, it is necessary to fulfill the criteria of the sampling theorem (i.e., Nyquist-Shannon theorem). This can be done in two ways:

- Automatic
  If the instruments reduces the bandwidth of a signal by using an appropriate filter, it can guarantee that no aliasing occurs. The drawback is that the harmonics can only be analyzed up to the cut off frequency of the filter and not up to half the sampling frequency.
  This approach is called automatic anti-aliasing, because the instrument guarantees this.

- Custom
  If an environment without signal components above the half-sampling frequency in the spectrum of the signal can be guaranteed, then the signal can be analyzed up to about half the sampling frequency. The drawback is, that any spectral component greater than half the sampling frequency *will* lead to aliasing and such aliasing effects are very difficult to detect!
  This behavior is called the custom anti-aliasing, because the customer has to guarantee this.

Configuration of the anti-aliasing is related to the BANDWIDTH [6.6→110] and FILTER [6.6.1→110] and can be configured as shown in SIGNAL SETTINGS [7.3.12→183]. HAAL [9.10.287→352]

6.6.3 Filter characteristics

The manual filter setting allows you to choose between three different filter characteristics. The different filter characteristics have different optimization criteria and therefore different properties:

Bessel filter

- monotonic amplitude response over frequency in passband

- slow roll-off

- moderate attenuation in stopband

- minimal overshoot

- maximally linear phase response, therefore constant group delay in passband, only minimal distortions in time domain

Butterworth filter
- monotonic amplitude response over frequency both in pass- and stopband

- quick roll-off

- good attenuation in stopband

- considerable overshoot and ringing in step response

- slightly non-linear phase response, group delay largely frequency-dependent, therefore moderate distortion in time domain

Chebyshev-II filter
- some residual ripple in passband, considerable ripple in stopband

- very quick roll-off

- excellent attenuation in stopband

- considerable overshoot

- group delay not constant even in passband, therefore strong distortions in time domain

Filter type selection

Depending on application, different filter types can be required:

If desired and undesired signal content are close in frequency, quick roll-off may be required, even at the cost of overshoot and ringing in the time domain. A Chebyshev-II filter might be adviseable.

If signal distortion in the time domain should be minimized and RMS power measurements require maximum precision, the Bessel filter's constant group delay makes it the preferred choice.

Should amplitude error be kept minimal in the passband and average selectivity be sufficient, the Butterworth filter suggests itself.

Digital Filter

For the LMG, it is possible to add a digital filter to the signal path. It is possible to add a digital low pass filter of order 8. You can select between a Bessel filter, Butterworth filter, and a Chebyshev-II filter. Additionally you can add a digital highpass filter of order 1.

In the sync path, you can add a digital low pass filter of order 1, and a digital highpass filter of order 1. If both filters are added to the sync path, the resulting filter is a bandpass of order 2.

The selected frequency, is the frequency such that attenuation has reached the value of -3dB. This means, the amplitude of current and voltage is reduced by a factor $1/\sqrt{2}$ for this frequency.

6.6.4 Fundamental bandwidth

The fundamental bandwidth is a hypothetic sinus signal for voltage and current with the sync frequency of the harmonics analysis. It is also called the third bandwidth. Its amplitude and phase are determined by the harmonics analysis of the applied signal.

The measurement values U_{trms} , U_{ac} , U_{Rec} , U_{dc} , U_{Max} , U_{Min} , U_{PP} , U_{CF} , U_{ff} , I_{trms} , I_{ac} , I_{Rec} , I_{dc} , I_{Max} , I_{Min} , I_{PP} , I_{CF} , I_{ff} , P, Q, S, PF, Z, R_{ser} und X_{ser} are computet for the fundamental bandwidth.

The effect of the analog filter and of the digital filter on the fundamental bandwidth are corrected similarly to the harmonics analysis.

The measurement duration of the fundamental bandwidth are a multiple of periods. This number of periods is the number of inter harmonics +1.

6.7 Parallel calculations

By eliminating some restrictions concerning the bandwidth (see Bandwidth [6.6→110]) there is no longer a need for the user to decide between measuring C-values [→113] (e.g. active power) *or* H-values [→113]. With this instrument, it is possible to do both in parallel. However, there are some physical aspects which should be kept in mind:

For both C-values and H-values, it is necessary to measure over an *integer* number of periods (see Synchronization [6.9→118]). The FFT is usually performed over a fixed number of periods (e.g. 10 according to IEC 61000-3-2/EN 61000-3-2) while RMS values and power are usually measured over a fixed time, the cycle time. This time has to be modified slightly from cycle to cycle to match exactly an integer number of periods. These two different concepts can raise different results when, for example, applying a DC signal. While the measurement over a cycle time works without any issues, the measurement over 10 periods for the FFT will fail, because the 10 periods will never be finished with DC!

Another point to keep in mind is the measurement principle of an FFT. The frequency resolution of the calculated bins (in principle the frequency components) is the inverse of the width of the time window. So when measuring for 200 ms, the frequency resolution of the bins is 5 Hz. When measuring for 10 s, the frequency resolution is 0.1 Hz. The advantage of a finer resolution is compensated by a disadvantage: If the instrument can measure 400 harmonics, a measurement window of 200 ms would allow analysis up to 2 kHz. With a 10 s window, the limit would be 40 Hz. So a 50 Hz fundamental could not be measured with a 10 s cycle time.

To solve this dilemma the main focus of the analysis can be chosen:

- C-values
  Values which are measured over a cycle time
  The focal point is the calculation of the power, rms, and related values. A cycle time can be defined for this purpose. The FFT is running under these circumstances with following characteristics: The number of periods for the measurement window is calculated as usual (see Measurement without gaps [6.10→119]).
  The harmonic values might be calculated faster or slower than the configured cycle time. Because the focus is on the calculation of cycle-based values, one gets a set of harmonics with each set of cycle-based values. If the harmonic values are calculated faster, you get the result of the last harmonics calculation. If the harmonics values are calculated slower, you might get the same set of harmonic values with multiple consecutive sets of cycle values.
  The cycle values are updated whenever a cycle has finished and are provided with the most recently calculated harmonic values, as described above.

- H-values
  Values which are measured over the time of a harmonic window
  The focal point is the calculation of the harmonics of one group (the so called master group). This can be setup with CYCLMOD [9.10.355→385] to e.g. "Harm 1". The number of periods of all groups are set up automatically (see Calculation of harmonics, option L6-OPT-HRM [6.8→114]).
  The master group is used to define a virtual cycle time. The calculation of cycle-based values of all groups is done using this virtual cycle time. The harmonic values are calculated independently, in the same way as before with cycle-based values.
  The consequence for the master group is that the calculation has the following characteristics: The measurement time is exactly the same as the time window of the FFT; so there is exactly one set of cycle-based values per set of harmonic values.

  If several groups are being measured, there are typically two approaches used for choosing the master group:
  (i) Choose the group with the highest signal frequency. This choice yields the best time resolution, but also unfortunately there is a lot of repeated data acquired from the slower groups.
  (ii) Choose the group with the lowest signal frequency. Here, the time resolution is worst, but each set of values consists of non-redundant data.

The configuration of one of these operating modes is combined with the cycle time setting and is described in MEASUREMENT TAB [7.3.2→176].

6.8 Calculation of harmonics, option L6-OPT-HRM

For the calculation of the harmonics, the measurement window has a width of an integer number of signal periods. The sample values of this window are transformed using an FFT algorithm into the desired lines or *bins*. The harmonics are calculated from these bins.

> *Bins, harmonics and interharmonics*
> *For example when measuring a 50 Hz signal over 10 periods (so you use a 200 ms window), the FFT will deliver bins in steps of 5 Hz. Each 10th bin is a harmonic related to 50 Hz and you get 9 interharmonic bins between the harmonics.*

FHARM
[9.10.103→274]
INTERHARM
[9.10.288→353]

The number of calculated harmonics can be fetched by HNUM [9.10.115→279]. It depends on the FILTER [6.6.1→110] configuration, the signal frequency, the ANTI-ALIASING [6.6.2→111] setting, and the number of interharmonics.

This also determines if the harmonics are measured with or without gaps (see MEASUREMENT WITHOUT GAPS [6.10→119]).

BURE [9.10.125→283]
BUIM [9.10.124→283]
BUAM [9.10.123→283]
BUPH [9.10.126→284]
INTERHARM
[9.10.288→353]

In general the instrument can measure up to 2000 *bins* of a signal as real and imaginary components. For the first 1000 bins amplitude and phase are calculated and these 1000 values can be displayed in the GUI. All of these values can be transferred via interface.

The number of interharmonics can be set up from 0 to 19. So with 0 (no) interharmonic, analysis up to harmonics order 2000 is possible. With 9 interharmonics according IEC 61000-4-7/EN 61000-4-7, analysis up to harmonics order 200 is possible and with 19 interharmonics, analysis up to harmonics order 100 is possible.

These bins can be used for evaluations concerning power quality and EMC.

Currently, the LMG supports, partly in conjuntion with the LMG Test Suite, following harmonic standards.

- 61000-4-7 for calculation of the harmonics

- 61000-3-2/-12 for testing of the harmonics

When the set number of periods is not valid under concrete circumstances (e.g. invalid signal frequency or filter), the harmonics are marked as NaN and displayed as dashes in the GUI.

Automatic filters

If the signal filter is set to auto, the filter setting is determined by the fundamental frequency FHARM [9.10.103→274] for the harmonics measurement. A high number of harmonics and measurement without gaps is preferred.

The exact calculation of harmonics depends on different settings (see GROUP MENU [7.3.12→182]) like

- single or dual processing (see BANDWIDTH [6.6→110])

- harmonics auto or custom (see ANTI-ALIASING [6.6.2→111])

- auto filter on/off, filter settings (see AUTOMATIC FILTERS [6.8→114])

- number of interharmonics (see CALCULATION OF HARMONICS, OPTION L6-OPT-HRM [6.8→114])

as well as on the applied signal frequency. The following tables give an overview, up to which order or frequency harmonics can be calculated. Furthermore, these tables provide if the calculation is without gaps and if it is free of aliasing.

In the Signal and Harmonics table in the GROUP menu, starting from version 3.14, the current integration durations of the respective values are additionally included. These help in interpreting the update rates of the normal and harmonic measurement values.

6.8.1 Interharmonics = 0

This section handles the case, if no interharmonics are wanted.

The following tables show the situation for the POWER MEASUREMENT CHANNEL L60-CH-A1 AND L60-CH-A2 AND L60-CH-A2.1 [4.7→55]. For the channel types POWER MEASUREMENT CHANNEL L60-CH-B1 AND L60-CH-B2 AND L60-CH-B2.1 [4.8→62] and POWER MEASUREMENT CHANNEL L60-CH-C1 AND L60-CH-C2 AND L60-CH-C2.1 [4.9→67] the principle is similar, but due to limited bandwidth and dual path technology the tables are different and shown in the respective section. The examples below can easly be translated for those channels.

Processing single path, auto filter off

#	Filter setting	Harmonics setting	Fundamental frequency	Gaps	Max. Order or frequency
1	<15 kHz	no effect	0.9 - 7.5 Hz	no	2000th or filter frequency
2	<15 kHz	no effect	7.5 - 99 Hz	no	filter frequency
3	<15 kHz	no effect	99 Hz - 7.5 kHz	yes	filter frequency
4	15 kHz	no effect	0.9 - 7.5 Hz	no	2000th
5	15 kHz	no effect	7.5 - 99 Hz	no	15 kHz
6	15 kHz	no effect	99 Hz - 7.5 kHz	yes	15 kHz
7	150 kHz	no effect	20 - 75 Hz	yes	2000th
8	150 kHz	no effect	75 Hz - 7.5 kHz	yes	150 kHz
9	no filter	Custom*	20 - 300 Hz	yes	2000th
10	no filter	Custom*	300 - 7.5 kHz	yes	600 kHz
11	no filter	Auto	—	—	—

*Aliasing is not prevented by the instrument, see ANTI-ALIASING [6.6.2→111]

This table shows for a given setting uniquely which fundamental frequency is processed in which way. For a particular fundamental frequency you see the different possible settings and can choose that which is the optimum concerning number of harmonics, gaps and aliasing for the conrete application.

The combination #11 (no filter, harmonics set to auto) is not valid, because without filter there can be no guarantee for no aliasing. So there no harmonics will be calculated.

Processing single path, auto filter on, auto harmonics

#	Filter setting	Harmonics setting	Fundamental frequency	Gaps	Max. Order or frequency
12	Auto on	Auto	0.9 - 7.5 Hz	no	2000th
13	Auto on	Auto	7.5 - 99 Hz	no	15 kHz
14	Auto on	Auto	20 - 75 Hz	yes	2000th
15	Auto on	Auto	75 Hz - 7.5 kHz	yes	150 kHz

For this table, the filter is set to auto, so the instrument has the decision, if a 25 Hz signal is measured with #13 or #14.

Processing single path, auto filter on, custom harmonics

#	Filter setting	Harmonics setting	Fundamental frequency	Gaps	Max. Order or frequency
16	Auto on	Custom[*]	0.9 - 7.5 Hz	no	2000th
17	Auto on	Custom[*]	7.5 - 99 Hz	no	15 kHz
18	Auto on	Custom[*]	20 - 75 Hz	yes	2000th
19	Auto on	Custom[*]	75 Hz - 7.5 kHz	yes	600 kHz

[*]Aliasing is not prevented by the instrument, see ANTI-ALIASING [6.6.2→111]

For this table the filter is set to auto, so the instrument has the decision, if a 25 Hz signal is measured with #17 or #18.

Processing dual path

#	Filter setting	Harmonics setting	Fundamental frequency	Gaps	Max. Order or frequency
20	<15 kHz	no effect	0.9 - 7.5 Hz	no	2000th or filter frequency
21	<15 kHz	no effect	7.5 - 99 Hz	no	filter frequency
22	<15 kHz	no effect	99 Hz - 7.5 kHz	yes	filter frequency
23	15 kHz or auto filter off	no effect	0.9 - 7.5 Hz	no	2000th
24	15 kHz or auto filter off	no effect	7.5 - 99 Hz	no	15 kHz
25	15 kHz or auto filter off	no effect	99 - 7.5 kHz	yes	15 kHz

This table shows for a given setting uniquely which fundamental frequency is processed in which way. For one single fundamental frequency you see the different possible settings and can choose that which is the optimum concerning number of harmonics, gaps and aliasing for the conrete application.

With dual path processing the harmonics are just computed for the small band values. There is no calculation for wide band values.

Examples

1. Consider single path processing, auto filter and auto harmonics. How many harmonics are calculated for a 32 Hz signal, and how many for a 34 Hz signal?
   For the 34 Hz signal #14 is the correct line and 2000 harmonics will be calculated. For the 32 Hz signal lines #13 or #14 could apply, and thus also 2000 or 468 harmonics are calculated. ($15\,kHz/32\,Hz = 468.75$.)

2. For an avionic application the harmonics from 400 Hz fundamental up to 150 kHz should be measured. What to set up?
   The probably best solutions are #8 and #15, because they have no potential disadvantages. #10 and #19 are possible, but have a potential risk of aliasing.

3. If you want the small bandwidth values for a frequency converter application with 25 Hz fundamental frequency and 1 kHz bandwidth, which setting has to be used? Are there any gaps in the measurement? *Probably, the best solution is #21, because there you get $1\,kHz/25\,Hz = 40$ harmonics beside the dual path advantages without any gaps.*
   #2 would also be possible but without the dual path advantages.
   #5, #7 and #9 could be used, but violate the 1 kHz bandwidth.
   #13, #14, #17 and #18 could be used, but the filter is not defined and some have a risk of aliasing.

Low Fundamental Frequencies

If the lowpass filter cut-off frequency is set to 5 kHz or lower, a measurement of harmonics with a fundamental frequency down to 300 mHz is possible. For 3 KHz or lower, down to 180 mHz is possible and for 1 kHz or lower down to 60 mHz is possible.

6.8.2 Interharmonics = 9

The following tables show the situation for the POWER MEASUREMENT CHANNEL L60-CH-A1 AND L60-CH-A2 AND L60-CH-A2.1 [4.7→55]. For the channel types POWER MEASUREMENT CHANNEL L60-CH-B1 AND L60-CH-B2 AND L60-CH-B2.1 [4.8→62] and POWER MEASUREMENT CHANNEL L60-CH-C1 AND L60-CH-C2 AND L60-CH-C2.1 [4.9→67] the principle is the same also if they are not shown.

This section handles the case, if 9 interharmonics are wanted. This is typical for measurements according to IEC 61000-3-2, IEC 61000-3-12, IEC 61000-4-7, EN 61000-3-2, EN 61000-3-12 and EN 61000-4-7 where a 200 ms window is required for a 50 Hz fundamental frequency. Due to this 10 period window, the fundamental frequencies are multiplied with 10, the number of bins stays constant and the order or harmonics is divided by 10. Following just the tables for this setting.

Processing single path, auto filter off

#	Filter setting	Harmonics setting	Fundamental frequency	Gaps	Max. Order Harm/Bin or frequency
1	<15 kHz	no effect	9 - 75 Hz	no	$200^{th}/2000^{th}$ or filter frequency
2	<15 kHz	no effect	75 - 990 Hz	no	filter frequency
3	<15 kHz	no effect	990 Hz - 7.5 kHz	yes	filter frequency
4	15 kHz	no effect	9 - 75 Hz	no	$200^{th}/2000^{th}$
5	15 kHz	no effect	75 - 990 Hz	no	15 kHz
6	15 kHz	no effect	990 Hz - 7.5 kHz	yes	15 kHz
7	150 kHz	no effect	200 - 750 Hz	yes	2000th
8	150 kHz	no effect	750 Hz - 7.5 kHz	yes	150 kHz
9	no filter	Custom*	200 - 3 kHz	yes	2000th
10	no filter	Custom*	3 - 7.5 kHz	yes	600 kHz
11	no filter	Auto	—	—	—

*Aliasing is not prevented by the instrument, see ANTI-ALIASING [6.6.2→111]

Processing single path, auto filter on, auto harmonics

#	Filter setting	Harmonics setting	Fundamental frequency	Gaps	Max. Order Harm/Bin or frequency
12	Auto on	Auto	9 - 75 Hz	no	$200^{th}/2000^{th}$
13	Auto on	Auto	75 - 990 Hz	no	15 kHz
14	Auto on	Auto	200 - 750 Hz	yes	$200^{th}/2000^{th}$
15	Auto on	Auto	750 Hz - 7.5 kHz	yes	150 kHz

Processing single path, auto filter on, custom harmonics

#	Filter setting	Harmonics setting	Fundamental frequency	Gaps	Max. Order Harm/Bin or frequency
16	Auto on	Custom*	9 - 75 Hz	no	$200^{th}/2000^{th}$
17	Auto on	Custom*	75 - 990 Hz	no	15 kHz
18	Auto on	Custom*	200 - 750 Hz	yes	$200^{th}/2000^{th}$
19	Auto on	Custom*	750 Hz - 7.5 kHz	yes	600 kHz

*Aliasing is not prevented by the instrument, see ANTI-ALIASING [6.6.2→111]

Processing dual path

#	Filter setting	Harmonics setting	Fundamental frequency	Gaps	Max. Order Harm/Bin or frequency
20	<15 kHz	no effect	9 - 75 Hz	no	200th/2000th or filter frequency
21	<15 kHz	no effect	75 - 990 Hz	no	filter frequency
22	<15 kHz	no effect	990 Hz - 7.5 kHz	yes	filter frequency
23	15 kHz or auto filter off	no effect	9 - 75 Hz	no	200th/2000th
24	15 kHz or auto filter off	no effect	75 - 990 Hz	no	15 kHz
25	15 kHz or auto filter off	no effect	990 - 7.5 kHz	yes	15 kHz

6.8.3 Amplitude correction

As the frequency of each bin is known and also the used filters are known it is possible to correct the filter response for each bin. For usual values like the rms value calculated from sample values, this is not possible, as the spectrum of the signal is unknown with this method. This can lead to an effect, that the fundamental of a voltage is displayed bigger than the rms value.

For example a 50 Hz signal is filtered with a 50 Hz filter. In this case the rms value calculated from sample values will be about 30 % smaller due to the -3dB attenuation of the filter. But the fundamental of the harmonic analysis will display the real value, as the -3dB attenuation is corrected.

6.9 Synchronization

Synchronization

The usual definitions for C-values [→113] (such as active power and RMS values) use always a measurement over an integer number of periods. For this reason, it is necessary that the instrument *synchronizes* to the applied signal. The synchronization is a common property of all channels within a group as these will always have the same fundamental frequency. For the synchronization sources, the following options exist:

- A voltage or current channel of the group
  The user can select one of the group's channels and configure how the frequency is determined in detail.
  If in a three phase system the synchronization is done on the zero crossing of one phase (so from 0° of one period to 0° of a later period), the two other phases are measured from 120° (resp. -120°) of one period to 120° (resp. -120°) of the same later period. So the measuring interval of all three phases is exactly identical!

- Another group
  A group can also be synchronized to another group.

- External
  The frequency which is applied to the external synchronization jack is used.

If using a voltage or current channel as a synchronization source, the instrument evaluates by default the zero crossings of the signal to determine the period duration and thus the frequency. For distorted or specially modulated signals there are additional settings. One can modify the following parameters:

- Level and hysteresis of the comparator
  By default the level is zero and the hysteresis is 2 %. For noisy signals, one can increase the hysteresis. For a signal that does not have any zero crossings, the level can be changed.

- Bandwidth
  For signals with high-frequency distortions (e.g. at the output of frequency converters), it is possible to apply a low-pass filter in order to get the fundamental signal.
  For example a typical pulse-width modulation inverter with fundamental frequency up to 300 Hz and a switching frequency of 16 kHz is measured by rejecting the 16 kHz component to get the 300 Hz component. So a filter with a cutoff frequency between these two frequencies would be used.

  > *This filter is just in the synchronization path of the instrument and does* not *influence the sampled measurement values!*

  In addition to the low-pass filter, it is also possible to configure a high-pass filter (e.g. to eliminate a DC component).

Filter position

- Demodulation
  A device using package control has in fact an amplitude modulated current. With the built-in demodulator, it is possible to demodulate this current and to synchronize to the fundamental of the modulation instead of the frequency of the supply system.

Except when measuring DC signals (where the exact measurement time does not matter), it is always important to have a correct synchronization in order to measure an integer number of periods. The best way to check the correct synchronization is to watch the frequency which is displayed for each group. If this frequency meets the fundamental frequency of the system, the synchronization is configured correctly.

> *For a correct synchronization a sufficient measurement duration is required. If for example the measurement duration is 1 s, it is not possible to measure a signal with 0.5 Hz (2 s periodic time) because there is no integer number of periods that fit into this duration.*

6.10 Measurement without gaps

For many applications, it is important to measure continuously (i.e. without gaps). For steady state signals, there will not be any difference if gaps in the measurement occur. However, with fluctuating signals, there is a high risk that the interesting part of the signal falls into a gap of the measurement where it would be disregarded. This can lead to significant errors in the reading of a signal.

As usual, ZES ZIMMER has implemented a way to perform gap-less measurement. The correctness of the results obtained depends on both, sufficient signal processing power as well as the right choice of algorithm:

The signal is measured and integrated over full periods, i.e. from zero crossing to zero crossing (determined by SYNCHRONIZATION [6.9→118]). When a measurement cycle ends, the completed periods are used to calculate the measurement values such as U_{trms}, I_{trms}, and P.

The uncompleted period is used as the starting value for the next coming samples. In other words, the measurements cover from the last positive zero crossing *before* the cycle until the last positive zero crossing within the cycle. With this behavior, it can happen that the real measurement duration is up to one period shorter or longer than the configured cycle time. In average it is equal to the cycle time.

> *Example: If one measures a signal with a periodicity of 20 ms with 50 ms cycle time, the real measurement duration alternates between 40 ms and 60 ms.*

If there is no synchronization signal available (e.g. when measuring DC signals), the measurement duration always equals the configured cycle time.

6.11 The Questionable Status Register

The SCPI standard (Version 1999.0, Vol. 1, Abschnitt 9) requires to provide a Questionable Status Register, in order to give an indication of the quality of the measurement values. Most commonly, a value is tagged questionable if the measurment range or a filter is currently being switched. Please note that a signal greater than the current range is not considered questionable.

The default behavior of this device is that questionable measurement data is not accessible from an interface and therefore cannot be gathered by calls to INIM [9.10.221→321] or CONT [9.10.219→320]. However, external programs might rely on receiving a steady output of values (e.g. for plots), no matter the quality. For tasks like these one can switch off this behaviour with the QUESTMASK [9.10.394→403] command. In order to determine whether the current measurement buffer is valid/questionable we provide a query to the Questionable Status Register (SQC [9.10.395→403]). Please note that we do not distinguish between the various reasons of data being questionable. Either all bits of the register are set (meaning that the data is questionable) or cleared. Alternatively, you can query QUEST [9.10.204→314], which returns a simpler value.

6.12 Energy Measurement

The time of the beginning of the energy measurement can be fetched by TSEN [9.10.194→310] and its duration by DUREN [9.10.191→309].

During the energy measurement the values EP [9.10.74→262], EQ [9.10.82→265], ES [9.10.75→262], EI [9.10.76→263], PM [9.10.130→286], QM [9.10.134→288] and SM [9.10.131→286] were integrated.

The precise duration of the energy measurement depends on the sync of the group, because the measuring duration of the normal values is adapted to the synchronization. It can be fetched by DUREN [9.10.191→309] followed by the coresponding suffix of the measurement value.

If you start the energy measurement by ENERGYSTART [9.10.461→430], the running measurement cycle is added to the energy. If you stop the energy measurement by ENERGYSTOP [9.10.462→431] the running measurement cycle will be discarded.

The energy measurement can be reset by ENERGYRESET [9.10.460→430]. In this case all integrated values are set to zero.

Changing the grouping by GROUP [9.10.286→351] or the wiring by WIRE [9.10.353→383] causes an automatic reset of the energy measurement.

Mathematical Formulas for Computation of the Energy

The formulas for the active energy EP, the apparent energy ES, the reactive energy EQ and the charge EI are:

$$EP \quad = \quad \sum_i T_i \cdot P_i, \tag{6.28}$$

$$ES \quad = \quad \left(\sum_i T_i \cdot U_{\mathrm{T}RMS,i} \right) \cdot \left(\sum_i T_i \cdot I_{\mathrm{T}RMS,i} \right) \tag{6.29}$$

$$EQ \quad = \quad \sqrt{ES^2 - EP^2}, \quad EI = \sum_i T_i \cdot I_{\mathrm{D}C,i}, \tag{6.30}$$

where the sum runs over the measurement cycles i. The real measurement time DURCYCL [9.10.189→309] of the measurement cycle i is denoted by T_i. Furthermore, P_i denotes the power value, $U_{\mathrm{T}RMS,i}$ the effective voltage, $I_{\mathrm{T}RMS,i}$ the effective current, and $I_{\mathrm{D}C,i}$ the direct current of the measurement cycle i. The net time of the energy measurement is computed by

$$T_{\mathrm{C}YC} \quad := \quad \sum_i T_i. \tag{6.31}$$

The average values of the active power PM, apparent power SM and reactive power QM are

$$PM := \frac{1}{T_{CYC}}EP, \quad SM := \frac{1}{T_{CYC}}ES, \quad QM := \frac{1}{T_{CYC}}EQ. \tag{6.32}$$

In contrast to previous LMGs, within the LMG6xx series the energy values are independent of the cycle time that is configured.

This means, that an unexpected but correct effect with volatile loads might occur. For example, if a non-reactive appliance is measured, the active energy increases after the energy measurement is being switched on. The apparent and the reactive energy remain low (at zero). If ONLY the current is being switched of, the active energy stops increasing, but the reactive and apparent energy start to increase. This unexpected behavior has a good reason. The switching of the current is a current modulation. This results in additional spectral components next to the 50Hz component. All the different components of U and I result in a distorted reactive power and this is being correctly taken into account during the whole energy measurement. Even if the normal cycle values (e.g. Default-Menu) can not reflect this due to their short time horizon.

Groups and Masks

The energy is integrated cycle by cycle. Moreover, the energy measurement of different groups can be activated separately. With the command ENERGYSTATE [9.10.83→266] you can fetch a bitmask that indicates for which group the cycle values were added. The value of ENERGYSTATE [9.10.83→266] is updated for each cycle.

By the command ENERGYMASK [9.10.459→430], you can fetch the bitmask that shows the configuration for which group the cycle values will be added to the energy. By the commmand ENERGYSTART [9.10.461→430] you can add bits to this mask and with ENERGYSTOP [9.10.462→431] you remove bits from this mask.

If ENERGYMASK [9.10.459→430] returns the value zero, then there is no energy measurement running for any group.

If the energy measurement is running depends on whether the energy I/O pin is set to high.

The policy for the energy I/O pin being set to high can be configured via the command ENERGYMOD [9.10.254→334].

If ENERGYMOD is set to manual, then the energy I/O pin is set to high if and only if ENERGYMASK is not equal to zero.

If ENERGYMOD is set to external control, the energy I/O pin is configured to be an input pin, such that its value is determined by the applied voltage.

If ENERGYMOD is set to schedule control, the energy I/O pin is configured to be an output pin. The start of the interval is configured by ENERGYSCHEDSTART [9.10.256→336] and its end by ENERGYSCHEDSTOP [9.10.257→336]

In ENERGYMOD Schedule, for an energy measurement to run and the Energy I/O pin to be asserted at all the ENERGYMASK must not be equal to zero, which is configurable via ENERGYSTART and ENERGYSTOP.

If the energy I/O pin is set to high at the end of the measurement cycle, the value of ENERGYSTATE takes on the value of ENERGYMASK. If the energy I/O pin is set to low at the end of the measurement cycle, the value of ENERGYSTATE becomes zero.

6.12.1 Energy interval measurement

In parallel to the energy measurement there is an energy interval measurement. When an energy measurement is running, energy is also integrated over successive intervals of fixed configurable duration.

The corresponding values for this interval can be fetched by the short commands EPINT, EQINT, ESINT, EIINT, PMINT, QMINT, and SMINT.

The interval length can be configured by ENERGYINTLEN [9.10.255→335]. The starting time of an interval can be fetched by TSENINT and the real duration of an interval by DURENINT.

6.13 Measurement Cycle

The measurement cycle is the main time frame for the LMG to provide measurement values. There are several modi for the measurement cycle, that can be set by CYCLMOD [9.10.355→385].

6.13.1 Cycle Modi

This section describes, what can be the set by CYCLMOD [9.10.355→385].

Fixed interval
If CYCLMOD is set on 0 or CYCLetime, the user can specify a fix time interval by CYCL [9.10.356→386] between 30 ms and 60 s. Moreover, the cycle pin of the synchronization connector is set to output.

6.13.2 External signal

If CYCLMOD is set on 1 or EXTern, the measurement cycle is determined by an external signal (rising edges) on the cycle pin of the synchronization connector (SYNCHRONIZATION CONNECTOR [8→47]). The duration between two of such impulses must be between 30 ms and 60 s. Any faster will be ignored. After 60 s, if no impulse was detected, a measurement cycle will be triggered automaticcaly by the LMG.

Harmonics
If CYCLMOD is set on HARMx, where $x = 1, 2, \cdots$ is a valid group, the measurement cycle is determined by the synchronization signal of the group x. In this case, the cycle length is INTERHARM [9.10.288→353] $+1$ times the perode length. Nevertheless, the minimum cycle time is about 30 ms and the maximum cycle time is about 1.5 s. Note that, for the gapless measurement of the harmonics of the group x, this setup is necessary, but not sufficient.

Scope
If CYCLMOD is set on 9 or SCOPE, the cycle length is determined by the sampling rate of the gapless scope. The cycle time is the quotient of GLPSR [9.10.170→302] and GLPTLEN [9.10.171→302]. For more about the gapless scope see .

Flicker
If CYCLMOD is set on 10 or FLICKER, the cycle length are 10 periods, that are determined by the first voltage channel for the flicker which is used for synchronization.

6.13.3 Cycle time

The start time of a measurement cycle can be fetched by TSCYCL [9.10.190→309], and the duration by DURCYCL [9.10.189→309].

6.14 Sample Values

All measured values are based on sample values of voltage and current. There are three ways to access the sample values directly; 4 oscilloscopes, the transient scope, and the gapless scoping.

6.14.1 Oscilloscopes

There are 4 oscilloscopes (scopes) implemented in the instrument. Each scope has 8 tracks with 2048 values The scopes with suffixes 21 and 22 are reserved for the internal GUI, the others (suffix 1 and 2) for the external GUI and for the remote control.

For each track of each scope, a signal can be selected by SCTRAC [9.10.352→383]. Possible signals are e.g. the sample values (u, i, or p) of each channel (directly measured) or linked values. Each scope has to be setup with a sample rate by SCSR [9.10.351→382]. The true sample rate can be read by SPSR [9.10.175→303]. The pretrigger can be set up by SCPTRT [9.10.349→381]. Each oscilloscopes can be attached by SCSYNC [9.10.466→432] to the sync of a group. Finally, the sample values itself can be read out by SPVAL [9.10.180→305].

If the scope with suffix 1 is finish, the Cont-On-Event scope.1 is sended, respectively 2, 21 or 22. In this way, it is possible to trigger a Cont-On with one of the 4 oscilloscopes.

6.14.2 Transient scope

There is a special transient scope to store a snapshot of a large number of samples triggered by a particular event (LMG option L6-OPT-EVT).

This scope has up to 16 tracks (LMG610 and LMG611 4 tracks) which can store 16 million (LMG6X0 4 million) samples each. It is shared by the GUI and remote control and thus usage on one interface also affects the other. For instance a recording that is triggered remotely is displayed in the GUI and can be logged there. The other way round, the event trigger can be comfortably configured using the GUI and after the trigger has fired, the recorded data can be retrieved via remote control.

The sampling rate can be set up via TRCSR [9.10.367→391], the signal for each track via TRCTRAC [9.10.369→392] and the recording time via TRRECLEN [9.10.365→390]. The transient trigger system is activated by TRANSIENTRESTART [9.10.465→432]. A trigger event can be generated manually by TRANSIENTNOW [9.10.464→431].

Furthermore, an automatic conditional event trigger can be configured. For this purpose the commands TRSOURCE [9.10.366→391] (the signal to monitor), TRLIMITA [9.10.361→388] and TRLIMITB [9.10.362→389] (two threshold values), TRCOND [9.10.357→386] (condition like "greater than" or "less than" for comparison against the threshold values) and TRCHLINK [9.10.363→389] (AND/OR combination of the trigger channels) are provided.

The transient can also be triggered externally, if the "Transient I/O" pin of the synchronization interface is configured as input (see TREXTERN [9.10.360→388] and SYNCHRONIZATION CONNECTOR [4.3→46]).

The recording may take some time. It is possible to request the current status via TRPSTAT [9.10.185→307]. The actual values are read via TRPVAL [9.10.188→308].

An example how to use by interface, can be found in READING SAMPLE VALUES OF AN EVENT [9.12.1→440].

6.14.3 Gapless scope

To activate the gapless sample scoping, switch the CYCLMOD [9.10.355→385] to SCOPE. Select the signals by GLCTRAC [9.10.285→351] for a particular track. To set up the sampling rate, use GLCSR [9.10.284→350] for your preferred value. The true sampling rate can be fetched by GLPSR [9.10.170→302]. GLPNTR [9.10.169→301] reads the number of tracks that provide the configured signals for this sampling rate. GLPTLEN [9.10.171→302] reads the number of samples per track, that were provided within one meassurement cycle. Use GLPVAL [9.10.172→302] to fetch the actual sample values. The real cycle time depends on the true sampling rate. An example how to use can be found in CONTINUOUS OUTPUT OF SCOPE VALUES [9.12.2→441].

As in this example illustrated, you have to use a cont-on, to fetch the gapless scope values.

6.14.4 Sample Sources

To set up the sample source use SCTRAC [9.10.352→383] for the oscilloscopes, TRCTRAC [9.10.369→392] for the transient scope and GLCTRAC [9.10.285→351] for the gapless scope. A sample source consists of a sequence of letters called sample identifier and a number called sample suffix. The range of the sample suffix depends on the sample identifier. Possible sample identifier are:

- **n**
  to deactivate the track

- **u**
  for the voltage. For possible suffixes see LOGICAL SUFFIXES [9.2.4→218].

- **i**
  for the current.

- **p**
  for the power.

There are additional sample identifiers available if a PSI card is installed. In case more than one card is present, see SUFFIXES OF PSI CARDS [9.2.4→220] for detailed information.

- **psiFain**
  For the fast analog in, the suffixes 1 and 2 are valid.

- **psiAin**
  For the analog in, the suffixes 1 to 8 are valid.

- **psiSwitchIn**
  For the digital switch in, the suffixes 1 to 8 are valid.

- **psiDirection**
  For the direction, the suffixes 1 and 2 are valid.

- **psiA**
  For the psi A, the suffixes 1 and 2 are valid.

- **psiB**
  For the psi B, the suffixes 1 and 2 are valid.

- **psiZ**
  For the psi Z, the suffixes 1 and 2 are valid.

- **psiTorque**

- **psiSpeed**

- **psiPower**

- **psiId**

- **psiIq**

- **psiINull**

- **psiTheta** The value of this sample stream is

$$\text{psiTheta} = \frac{360°}{\text{PSIPULPERREV}} \cdot N(t),$$

where $N(t)$ is the number of rising edges detected by PsiA. The value of $N(t)$ is set to zero, if a rising edges is detected by PsiZ. The suffixes 1 and 2 are valid. The value of PSIPULPERREV [9.10.333→374] is configurable.

Examples

Set the sample source to "" (empty) or "n1" to deactivate the trace. Use "u1" for the voltage of the first hardware channel or "p2311" for the power of the third physical channel with narrow bandwidth of the second group.

6.15 Scripteditor

6.15.1 General

The scripteditor allows the creation of own measuring values by use of a built-in scripting language. The presently installed script will be executed once per measuring cycle and will result in one or more values to be displayed by names defined by the script.

The language works similar to an actual programming language and consists of statements that are separated or terminated by semicolon (;) or newlines, whichever is more convenient for the user.

There can be comments/remarks in the script, introduced by a double slash (//). Text that follows it, until the start of the next line of text will be ignored. Comments that should span multiple lines can be written by enclosing them by /* at the begin and */ at the end.

6.15.2 Types of values

The scripteditor supports a couple of different kinds of values (also called "types").

Truth Values, also called `bool`, are used in conditions of selection and loop statements (see SELECTION STATEMENTS [6.15.4→128]).

Floatingpoint Number, also called `float`, stores fractional numbers, e.g. π. Only these values can be assigned as results to public variables (see PUBLICVARIABLES [→126]). Values of this kind range from approx. 10^{-308} up to approx. 10^{+308} (and the same range in the negative direction), according to the IEEE754 standard for double-precision floating point numbers, but in this range, only the upper approx. 15 decimal digits are significant. Normal numeric constants are of this type. When these values are assigned to public variables, the value that is published on the interface (INTERFACE LANGUAGES [9.2→215]) will be converted to IEEE754 single-precision floating point numbers, which have a range of approx. 10^{-38} up to approx. 10^{+38} (including in the negative direction), with only the non-zero upper-most approx. 6 digits being significant. The higher precision and range for script intermediate variables exists so that a large number of operations are possible without leading to high inaccuracies.

Dates, also called `date`, store points in time. These values can only be created by special functions (see CONSTANTS [6.15.5→129]).

Times, also called `time`, store time durations. These values also can only be created by special functions (see CONSTANTS [6.15.5→129]).

Integer, also called `int`, is an integral number that ranges from -2^{63} to $+2^{63} - 1$. It's often used in loops (Loops [6.15.4→128]) as a counter variable.

Each of these values can also appear in list values. Lists are created by reading device values that happen to be lists, such as BUAM [9.10.123→283]. Such lists are read-only, elements of them cannot be modified by the script. A second kind of lists can be created by the script itself using a special syntax explained at Lists [6.15.5→134].

6.15.3 Variables

Variables can be used to store values within the script or publish values to the interface or the GUI. They have a name that can consist of a sequence of digits and letters (`a-zA-Z0-9`), and start with a letter. To access measuring values generated by the LMG itself, the SHORT interface notation is used, see Interface Languages [9.2→215]. This is different from accessing variables. The following kind of variables exist:

Script internal variables These variables will not be displayed in the device and are used to save intermediate values, for example to break up long expressions into multiple smaller ones. These variables start with a dollar sign (`$`), e.g. `$i`.

Public variables These variables will be displayed as result of the script in the LMG. These variables do not start with a dollar sign. Variables of this type must contain a floatingpoint number at the end of the script execution.

Initial values

You can only read from variables if you have previously written to the variables in the current cycle. If you wish to read the value written in a previous cycle, you need to initialize that variable once, in the first run of the script. This way, typing mistakes and logical errors will be minimized and detected early instead of only when running the script and actually using the unassigned variable.

```
MaximalValue = max((utrms1?), MaximalValue)
```

This kind of error is detected directly while writing this script within the GUI (General GUI handling [7.1→145]) without actually running the script first and would result in

```
(l 1, c 32..44): error: variable 'MaximalValue' wasn't initialized by
                        all possible paths through the script till here
```

The error message points to the use of the uninitialized variable in column 32 till 44 of line 1. The solution is to initialize the variable in a `def`-block. Def blocks contain definitions about variables, such as their initial value.

```
def { MaximalValue = utrms1?; }
MaximalValue = max((utrms1?), MaximalValue)
```

Type conflicts

When a variable is read, the last assignments to it must all store the same type of value (Types of values [6.15.2→125]), to prevent possible mistakes and increase the performance of the script. The following script is not valid because the variable `MyCounter` on the left side of the addition has the value `0d` during the first run of the script, which is an integer (see Constants [6.15.5→129]) that could possibly represent values up to $+2^{63}-1$, but during the second run of the script, the variable will store the value `1`, which is a floating point number. These values can store numbers of only up to around 15 decimal places, and will loose accuracy in favor of a greater range of values.

```
def { MyCounter = 0d; }
MyCounter = MyCounter + 1
```

This kind of error is detected directly while writing this script within the GUI (General GUI handling [7.1→145]) and would result in

```
(l 2, c 13..22): error: ambiguous type for variable 'MyCounter'
(l 1, c 17..18): note: has type 'int' by assignment here
(l 2, c 11..12): note: has type 'float' by assignment here
```

Fix this error by making the first assignment to use a Floatingpoint number 0 instead of 0d (Types of values [6.15.2→125]).

Def Blocks

A def-block can be used to provide attributes of variables that are independent of individual repeated execution-cycles, such as their initial value, their unit and their formula-id. As an example, consider a first example that returns the first value of BUAM [9.10.123→283] without the use of a def block

```
buam0 = buam?[0]
```

This script reads the BUAM [9.10.123→283] list, truncated to its first value, and then reads out that first value at index 0 from this truncated list using the list index operator list[index]. The GUI would display this using the variable name buam0, because while it knows how the value is computed and how the variable is named, it has no idea about the meaning/use of this value to the user. Ideally, the value would be displayed using the correct unit V and with a meaningful formula-id. This can be achieved using a def block as in the following

```
def { buam0 "buam_0" = "V"; }
buam0 = buam?[0]
```

In this example, the formula id will be displayed as $buam_0$. The _ character indicates subscript, while a ^ character would indicate superscript (such as expressing A^2). As was demonstrated at Initial values [6.15.3→126], an initial value can also be provided. To accumulate $buam_0$ across cycles, you can initialize $buam_0$ by zero such as in the following

```
def { buam0 "buam_0" = 0 "V"; }
buam0 = buam0 + buam?[0]
```

Environment Variables

Environment variables can be set using the function **SetEnv(n, value)** and read using the function **Env(n)**. One special feature of environment variables is that they can be visualized and modified by certain custom menu content (see Types of content and background image [7.2.11→167]).

6.15.4 Statements

Statements consist of sequential statements, selection and loop statements. Sequential statements will be executed once, but can be skipped by selection statements or executed multiple times by loop statements. Examples of these sequential statements are block statements and assignments. The following list gives an overview of the available statements.

```
def { $MyCycleCounter = 1d; }
// sequential statements
MyVariable = 3.14
$MyCycleCounter = $MyCycleCounter + 1d

// selection statements
if(utrms? > 230) {
  // statements executed if utrms? > 230
 MySecondVariable = 1
} else {
  // statements executed otherwise
  MySecondVariable = 0
}

// loop statements
$buam = buam? (0:400)
$buamLength = len($buam)
for($i = 0; $i < $buamLength; $i = $i + 1) {
  // statement that is executed 401 times
}
```

Multiple statements can be grouped together into a single statement (also called "block statement") by using {…}. Statements can also contain expressions that can be linked together by operators. The right side of an assignment to a variable (`a = b`) is an expression too, for example.

Assignments

An assignment consists of the assigned variable, an equal sign and the assigned value. The variable will then store the assigned value, until after the variable is assigned again with a different value. In the following example, the amount of pulses above `3A` are counted (ACCESS TO MEASURING VALUES [6.15.5→134]):

```
def { n = 0; $r = 0; }
$ibat = abs(idc1?)
if($ibat => 3) {
  if($r == 0) {
    n = n + 1
    $r = 1
  }
} else {
  $r = 0
}
```

As is described at INITIAL VALUES [6.15.3→126], a def-Block can be used to define that a variable has an initial value. Here, the variables n and $r are given initial values (VARIABLES [6.15.3→126]). If the assigned variable is a list element, the referred list must not be a measuring value list, and must have been created using a list-creation operator (see LISTS [6.15.5→134]).

Selection statements

To execute statements only when certain conditions are true, `if` statements can be used. They have two forms. The first form executes its statement when the accompanioned condition is `true`, and skips it when the condition is `false`. For the second form, another statement can be given that is executed when the condition is `false`.

```
if(condition) statement
```

```
if(condition) statement else statement
```

For an example of using `if`, refer to ASSIGNMENTS [6.15.4→128].

Loops

Loops are used to execute certain statements multiple times during a single script cycle. They consist of an initialization part which is executed once at the start of the loop, a condition part which is tested before the body of the loop is executed, and an increment part that is executed after each time the body is excecuted. As soon as the condition turns out to be `false`, the body won't be executed once more and the loop is terminated instantly. The execution continues after the loop.

The following example shows a loop which sets a variable to 0 in its initialiation part. In the condition part it checks that the variable is smaller than 100. Furthermore, in the increment part, the loop increments the value of the variable by one. The body of the loop uses the variable as a position (index) into the voltage harmonics list (BUAM [9.10.123→283]) of the first channel. The square of the referenced harmonic (or interharmonic) is added to another variable with every loop body execution. After termination of the loop, this variable will contain the square-sum of the first 100 harmonics (and interharmonics).

```
sum = 0
for($i = 0; $i < 100; $i = $i + 1) {
    $v = buam1? [$i]
    if($v == $v)
        sum = sum + $v * $v
}
```

Because of running signal filters, not all of the harmonics may be available. Therefore, a self-comparison will check whether the value is NaN (Not-a-Number). Only if it's not NaN, it will be used.

6.15.5 Expressions

Expressions are used to compute values and are used in several kind of statements, for example in the right side of an assignment or in the condition of an `if` or `for` statement.

Constants

Constants are the most simple form of expressions. There are integer-, floating point, and the truth value constants `true` and `false`. To create a constant `date` or `time` value, special functions can be used (`TimeHMS`, `TimeS`, `DateYMD`). In the following example a variable will be incremented each time the measuring duration of the first channel is above the constant time duration *50ms*.

```
def { longerThan50Ms = 0; }
if(durnorm1? > TimeS(0, 50))
   longerThan50Ms = longerThan50Ms + 1
```

An integer constant can be indicated by a trailing "d" character. Then, values of up to $2^{63} - 1$ can be saved, as in the following example.

```
def { $longerThan50Ms = 0d; }
if(durnorm1? > TimeS(0, 50))
$longerThan50Ms = $longerThan50Ms + 1d
```

Operators

The script language supports the operators shown in the following table.

`<N>(a)`	List creation of a list of `N` elements. The expression `e` is used as the initial value of all elements.
`funktion(a1, a2, ..., aN)`	Function calls. The expressions within the parentheses are used as input arguments.
`a1[a2]`	Indexes the element at position "a2" of the list "a1".
`-a`	Negation of a numerical value - integer or floating point number: ($-value == -1 * value$)
`!a`	Negation of a truth value. ($!$(`a1 < a2`)$==$ (`a1 >= a2`))
`a1 * a2, /`	Numeric multiplication and division.
`a1 + a2, -`	Numeric addition or subtraction of two integer or floating point numbers.
`a1 && a2`	Used in conditions to express that both `a1` *and* `a2` shall be true in order for the condition to be `true`.
`a1 || a2`	used in conditions to express that it suffixes that one of `a1` *or* `a2` is true for the condition to be `true`.
`a1 < a2` `a1 <= a2` `a1 == a2` `a1 >= a2` `a1 > a2` `a1 != a2`	Used in conditions to perform comparisons.

Table 6.2: Operators of the scripting language (from highest to lowest priority)

Functions

The scripting language supports various functions that can be called by `functionname(arguments)`. Tables Table 6.3 [→130], Table 6.4 [→131], Table 6.5 [→132] and Table 6.6 [→133] list all available functions. Functions that are indicated to accept `int` will also accept `float`-Values, as long as there are no fractional components in the `float`-Value.

Energy measurement can be controlled by the functions `StartEnergy`, `StopEnergy` and `ResetEnergy` that each take a single parameter denoting the group that should be affected. If no group number is passed, all gropups's energy measurements are started, stopped or have their values resetted respectively.

```
if(P2111? > 100) {
   // start measurement of group 1 and 2, starting with this cycle.
   startEnergy(1)
   startEnergy(2)
} else {
   // stops all groups (including 1 and 2)
   stopEnergy()
}
```

The side-effects of any function like `StartEnergy` take effect only at the end of the current script iteration. Therefore, calling `LogOnce` multiple times, for example, is allowed and has the same effect as calling it only once.

> *Avoid calling the functions to control the energy measurement (Start-, Stop-, and ResetEnergy) in each iteration, especially with short cycle times. This increases the system load and the responsiveness of the device may suffer. The correct approach would be to use a state machine design for the code and to control the energy measurement only on state transitions.*

Name	Arguments	
`Int(a1)`	`float`, `int`	Converts a numeric value into an integer value.
`Float(a1)`	`float`, `int`	Converts a numeric value into a floating point value.
`DateYMD(year, month, day,` `[hour, min, sec, milli, micro, nano])`	`int` for all	Creates a `date`-value with the passed field values. Optional arguments are assumed to be zero/one as appropriate.
`TimeHMS(hour, min, sec,` `[milli, micro, nano])`	`int` for all	Creates a `time`-value with the passed field values. Optional arguments are assumed to be zero if not passed.
`TimeS(sec, [milli, micro, nano])`	`int` for all	Creates a `time`-value with the passed field values. Optional arguments are assumed to be zero if not passed.

Table 6.3: Provided functions for creating or converting values and their arguments. Optional arguments are indicated by brackets.

Name	Arguments	
SetEnv(n, value)	int, float	Sets the nth environment variable to *value*. This function can work together with the "Env ..."-Elements in the Custom Menu (TYPES OF CONTENT AND BACKGROUND IMAGE [7.2.11→166]).
Env(n)	int	Reads the nth environment variable. This function can work together with the "Env ..."-Elements in the Custom Menu (TYPES OF CONTENT AND BACKGROUND IMAGE [7.2.11→166]).
SetEnvInt(n, value)	int, float	Sets the nth integer environment variable to *value*. This function can work together with the "Env ..."-Elements in the Custom Menu (TYPES OF CONTENT AND BACKGROUND IMAGE [7.2.11→166]).
EnvInt(n)	int	Reads the nth integer environment variable. This function can work together with the "Env ..."-Elements in the Custom Menu (TYPES OF CONTENT AND BACKGROUND IMAGE [7.2.11→166]).
Switch(n, btn)	int, int	Tests, whether the button *btn* of the custom switch that uses the environment variable n, is active (TYPES OF CONTENT AND BACKGROUND IMAGE [7.2.11→166]).
SetSwitch(n, btn, onOFF)	int, int, bool	Sets oder un-sets the state of the Button *btn* of the switch that uses the environment variable n (TYPES OF CONTENT AND BACKGROUND IMAGE [7.2.11→166]).
Beep()		Makes a short signal sound.

Table 6.4: Provided general functions and their arguments.

Name	Arguments	
StartEnergy([n])	int	Starts the energy measurement of group **n**, or all groups if missing. The effect is the same as short command ENERGYSTART [9.10.461→430] used over the interface.
StopEnergy([n])	int	Stops the energy measurement of group **n**, or all groups if missing. The effect is the same as short command ENERGYSTOP [9.10.462→431] used over the interface.
ResetEnergy([n])	int	Resets the energy measurement of group **n**, or all groups if missing. The effect is the same as the short command ENERGYRESET [9.10.460→430] used over the interface.
SetPSIDigitalOut(n, v)	int and bool or int	Sets the PSI digital-out pin at position **n** (starting from 1) to the value **v** which must be boolean value or an integer 0 or 1 (TYPES OF VALUES [6.15.2→125]).
SetGPIOOut(n, v)	int and bool or int	Sets the digital-out pin of the GPIO-card at position **n** (starting from 1) to the value **v** which must be boolean value or an integer 0 or 1.
logOnce()		Causes an already created log to log the values of the current cycle.
exec()		Causes the device GUI to simulate an "Execute" button press (see ACTIONS MENU [7.3.15→187]).
execRemote()		Causes the remote GUI (REMOTE CONTROL OF THE DEVICE WITH LMG REMOTE [8.2→196]) to simulate an "Execute" button press.
Len(list)	A list	Yields the length of the passed list.

Table 6.5: More provided general functions and their arguments.

Name	Arguments	
`Pow(base, exp)`	`int, float`	$base^{exp}$.
`Pow10(exp)`	`int, float`	10^{exp}.
`Log(a1)`	`int, float`	Natural logarithm of `a1`.
`Log10(a1)`	`int, float`	Logarithm for base 10 of `a1`.
`Log2(a1)`	`int, float`	Logarithm for base 2 of `a1`.
`Sqrt(a1)`	`int, float`	Squareroot of `a1`.
`Cbrt(a1)`	`int, float`	Cubicroot von `a1`.
`Sin(a1)`	`int, float`	Sine of `a1`, radiant.
`Cos(a1)`	`int, float`	Cosine of `a1`, radiant.
`Tan(a1)`	`int, float`	Tangent of `a1`, radiant.
`Asin(a1)`	`int, float`	Arc Sine of `a1`, radiant.
`Acos(a1)`	`int, float`	Arc Cosine of `a1`, radiant.
`Atan(a1)`	`int, float`	Arc Tangent of `a1`, radiant.
`Atan2(y, x)`	`int, float`	Arc Tangent of y/x, radiant. Uses their sign to determine the result quadrant. $\alpha = atan2(sin(\alpha), cos(\alpha)),\ -\pi < \alpha \le \pi$.
`Sinh(a1)`	`int, float`	Sine Hyperbolic of `a1`.
`Cosh(a1)`	`int, float`	Cosine Hyperbolic of `a1`.
`Tanh(a1)`	`int, float`	Tangent Hyperbolic of `a1`.
`Asinh(a1)`	`int, float`	Arc Sine Hyperbolic of `a1`.
`Acosh(a1)`	`int, float`	Arc Cosine Hyperbolic of `a1`.
`Atanh(a1)`	`int, float`	Arc Tangent Hyperbolic of `a1`.
`Min(a1, a2, ..., aN)`	`int, float`	Returns the smallest value among the passed arguments as a floating point number.
`Max(a1, a2, ..., aN)`	`int, float`	Returns the largest value among the passed values as a floating point number.
`Pi()`		Yields the constant π
`E()`		Yields the constant e.

Table 6.6: Provided functions and their arguments for mathematical calculations.

Lists

The script can also create lists of a fixed size and use it internally or publish it (see VARIABLES [3→126]). A fixed-size modifiable list is created by specifying its size within angle brackets, followed by the initial value of the elements in parentheses. The following shows part of the computation of a sliding average using a sliding window of size 32. The momentary window content is defined as a public list of 32 `float`-Values. If it's not desired to publish the window values, the name can be prefixed by a dollar sign, as with other non-list variables. This will allow using a much larger list size of a few thousand floats (the exact bound varies depending on how large the script is), as the total amount of published values is limited.

```
def { U_win = <32>(utrms?) ''V''; }
// shift all values by one
for($i=len(U_win);$i>0;$i=$i-1)
   U_win[$i] = U_win[$i-1]
// Insert the current value of utrms?
U_win[0] = utrms?
```

As opposed to other values, list values cannot be discarded. Once a list is stored into a variable, it must be kept inside of that variable throughout the script run. The reason for this is the amount of space these lists consume, which would otherwise gobble up the memory of the LMG and prevent deterministic running times of the script. The following example shows the implications.

```
// Invalid: Overwrites itself in the next cycle!
example1 = <32>(utrms?)

// Invalid: Discards the already stored list value!
example1 = 0

// Invalid: The list cannot be copied to another variable
example2 = example1
```

These constraints do not apply to measurement value lists, as these cannot be modified and do not consume additional memory in the LMG script.

```
// Retrieve a read-only measurement list
sum = buam? (1:10)
$tmp_sum = 0
for($i=0;$i<len(sum);$i=$i+1)
   $tmp_sum = $tmp_sum + sum[$i]
// Valid: sum now contains the average value
sum = $tmp_sum / len(sum)
```

Access to measuring values

To access measuring values (e.g. UTRMS [9.10.217→319]), the SHORT notation is used (INTERFACE LANGUAGES [9.2→215]). The presence of the question mark in this notation indicates that a measuring value is being accessed. After the question mark, additional parameters for the measuring value can be specified, such as list parameters. This special SHORT command string is terminated by a following newline, semicolon or other script operator. Some SHORT commands also accept multiple parameters by separating them with a comma (as described in INTERFACE LANGUAGES [9.2→215]). If a comma needs to be specified as part of the script, instead of the SHORT command, the command can be parenthesized.

```
MyTwiceUtrms = utrms1? * 2
// SHORT command is ''buam?'', index by 0
dc = buam?[0]
// First sample of scope 2, first trace
s0 = spval2? 0 [0]
// First ten samples: len(s010) == 10
s010 = spval2? 0,(0:9)
// The max of buam?[1] and buam?[3]
m = max((buam?[1]), (buam?[3]))
```

Optionally and for compatibility with earlier versions of the scripting support, the SHORT command may be enclosed in square brackets. That allows the SHORT command to contain operators that normally would terminate the SHORT command.

6.16 Measurement uncertainty of measured and calculated values

The specified accuracy of the basic values U_{trms}, I_{trms} and P can be found in TECHNICAL DATA [4→39]. The following calculations illustrate how to use these tables to calculate the uncertainty of measured and calculated values.

We will use the following measurement as example:

$$
\begin{aligned}
U_{trms} &= 230.000\,V,\text{ nominal range } 250\,V,\text{ peak value of range } 400\,V & (6.33)\\
I_{trms} &= 0.95000\,A,\text{ nominal range } 1.2\,A,\text{ peak value of range } 3.75\,A & (6.34)\\
PF &= 0.25000 & (6.35)\\
f &= 50.0000\,Hz & (6.36)\\
P &= 54.625\,W,\text{ nominal range } = 250\,V * 1.2\,A = 300\,W, & (6.37)\\
&\quad\text{ peak value of range } = 400\,V * 3.75\,A = 1500\,W
\end{aligned}
$$

AC coupling mode for the signal.

From the tables with the specifications, the following uncertainties can be calculated directly:

$$
\begin{aligned}
\Delta U_{trms} &= \pm(0.01\,\%\text{ of reading} + 0.02\,\%\text{ of upper range value}) & (6.38)\\
\Delta U_{trms} &= \pm(0.023\,V + 0.08\,V) = \pm0.103\,V & (6.39)\\
\Delta I_{trms} &= \pm(0.01\,\%\text{ of reading} + 0.02\,\%\text{ of upper range value}) & (6.40)\\
\Delta I_{trms} &= \pm(0.095\,mA + 0.75\,mA) = \pm0.845\,mA & (6.41)\\
\Delta P &= \pm(0.015\,\%\text{ of reading} + 0.01\,\%\text{ of upper range value}) & (6.42)\\
\Delta P &= \pm(8.194\,mW + 150\,mW) = \pm0.158\,W & (6.43)
\end{aligned}
$$

Why use the upper range value (peak value) instead of the nominal value for error calculation?

Analog instruments use the DC or RMS component of a signal to display the true value. Due to some analog-related characteristics of the working principle (e.g. saturation, non-linearities of components) of these instruments, it could possibly happen that the reading of a signal with a small RMS component but a large peak value gets impaired. Therefore, these instruments had to specify a maximum allowed crest factor (the ratio of peak value to RMS value) for which they could guarantee a reading inside their specification. However, there was really no concrete limit for the maximum measurable peak value so the range used for error calculations was the RMS range for sinusoidal signals.

The world has changed and the signals have become more and more distorted and modern instruments, like the LMG, use ADC. The range is now defined as the largest value an ADC can sample. The RMS can be as large as this largest value (e.g. in the case of DC) or also significantly smaller (e.g. in the case of inrush current). For this reason, the only range value for an error calculation which could be legitimated physically is the peak value (i.e. the range of the ADC).

The definition of the RMS value (as well as the no longer necessary crest factor) is rather arbitrary: A 100 Vpk range can be specified as 70 Arms range with crest factor 1.43 or also as 5 Vrms range with crest factor 20. In other words, it makes sense to specify the crest factor for a signal (to select the correct peak range) but it is pointless to specify a crest factor for a modern digital instrument, since it does not carry meaningful information. The only important point is that the peak value does not exceed the ADC range!

*It is absolutely meaningless to specify am RMS range for error calculations for modern instruments with ADC sampling. This is the reason why **ZES ZIMMER**and other companies*

use the technically correct peak value of a range, as it represents the true upper limit of the ADC.

The power factor is computed as follows:

$$PF \quad = \quad \frac{P}{S} = \frac{P}{U * I} \tag{6.44}$$

The uncertainty of the power factor is calculated using the total differential:

$$\Delta PF \quad = \quad \frac{\partial PF}{\partial P}\Delta P + \frac{\partial PF}{\partial U}\Delta U + \frac{\partial PF}{\partial I}\Delta I \tag{6.45}$$

$$\Delta PF \quad = \quad \frac{\Delta P}{U * I} - \frac{P * \Delta U}{I * U^2} - \frac{P * \Delta I}{I^2 * U} \tag{6.46}$$

$$\Delta PF \quad = \quad \frac{0.158\,W}{230\,V * 0.95\,A} - \frac{54.625\,W * 0.103\,V}{0.95\,A * (230\,V)^2} - \frac{54.625\,W * 0.845\,mA}{(0.95\,A)^2 * 230\,V} \tag{6.47}$$

$$\Delta PF \quad = \quad 0.00039 \tag{6.48}$$

This is the maximum uncertainty of the instrument. The typical uncertainty is two to five times smaller!

These are the uncertainties related to the instrument. For the total uncertainty of the measurement there are further things to take into account such as:

- The error occurring when reading a digital number is ± 0.5 of the last digit.

- Environmental influences like coupling into the measuring wires

- The impact if voltage or current were measured correctly

- etc.

If you use external sensors, please see the **ZES ZIMMER** Sensors and Accessories Manual for hints on how to calculate the total uncertainty.

6.16.1 Uncertainty of derived values like energy

The uncertainty of derived values is generally calculated using the total differential. As an example this is demonstrated for the energy. The formula for the calculation itself is

$$E \quad = \quad P * t \tag{6.49}$$

Now this equation is derived by each independent variable and multiplied with the uncertainty of if:

$$\Delta E \quad = \quad \frac{\partial E}{\partial P}\Delta P + \frac{\partial E}{\partial t}\Delta t \tag{6.50}$$

$$\Delta E \quad = \quad t * \Delta P + P * \Delta t \tag{6.51}$$

By this the uncerainty of the derived value can be easily calculated.

6.16.2 Interpolation of the uncertainty specifications over the frequency

For an easy handling the specifications of the instrument are given for several frequency ranges. By this for example between the 3 kHz ... 10 kHz range and the 10 kHz ... 50 kHz range the specification changes in a step. This does of course not reflect the physical behaviour of the instrument but is a simplification. To estimate the real behaviour, a linear interpolation in logarithmic frequency diagram can be assumed. So if the specification changes from 0.015 % @ 3 kHz to 0.03 % @ 10 kHz a value of

$$0.015\,\% + (0.03\,\% - 0.015\,\%) \cdot \frac{\log_{10}\left(\frac{5\,\text{kHz}}{3\,\text{kHz}}\right)}{\log_{10}\left(\frac{10\,\text{kHz}}{3\,\text{kHz}}\right)} = 0.021\,\% \tag{6.52}$$

at 5 kHz can be expected typically instead of the guaranteed 0.03 %.

6.16.3 Uncertainty of harmonics

The specifications of the harmonics can be found in the sections 4.7, 4.8 and 4.9. The following calculations show for the A channel how to use this specification.

A voltage signal consists of the harmonics
Uh1 = 230 V, 50 Hz, 0° and
Uh5 = 5 V, 250 Hz, −130°

The 250 V range is used (400 V peak) with active 15 kHz filter.

Fundamental amplitude uncertainty Uh1:
$\pm (0.01\,\% \cdot 230\,\text{V} + 0.02\,\% \cdot 400\,\text{V}) = \pm 0.10\,\text{V}$

Fundamental amplitude uncertainty Uh5:
$\pm (0.015\,\% \cdot 5\,\text{V} + 0.03\,\% \cdot 400\,\text{V}) = \pm 0.12\,\text{V}$

The additional uncertainty by the filter is for both harmonics $50 \cdot 10^{-6}$ of the measured value, so:

Filter uncertainty Uh1: $\pm (0.005\,\% \cdot 230\,\text{V}) = \pm 0.011\,\text{V}$
Filter uncertainty Uh5: $\pm (0.005\,\% \cdot 5\,\text{V}) = \pm 0.25\,\text{mV}$

In sum you get following uncertainties for the amplitude:

Amplitude uncertainty Uh1: $\pm (0.10\,\text{V} + 0.01\,\text{V}) = \pm 0.11\,\text{V}$
Amplitude unceratinty Uh5: $\pm (0.12\,\text{V} + 0.00\,\text{V}) = \pm 0.12\,\text{V}$

For the phase you get

Phase uncertainty Uh1:
$\pm \left(0.05° + 0.005° \cdot 1 + 0.05° \cdot \frac{50\,\text{Hz}}{1\,\text{kHz}}\right) = \pm 0.058°$

Phase uncertainty Uh5:
$\pm \left(0.05° + 0.005° \cdot 5 + 0.05° \cdot \frac{250\,\text{Hz}}{1\,\text{kHz}}\right) = \pm 0.088°$

6.17 Measuring ranges

This instrument offers various measuring ranges for voltage and current to allow measuring signals with the least uncertainty possible. The choice of measuring range should always be guided by the signal to be measured. In the past, analog instruments used the DC or RMS value to display the true value of a signal. Due to some peculiarities of analog equipment (saturation, non-linear component behavior) it may be possible for signals with an extremely large peak value to be significantly distorted. Therefore, for these instruments, a maximum permitted crest factor had to be specified (ratio of peak to RMS value) up to which the specs regarding precision would be complied with. There was no explicit hard limit for the maximum peak value however. Thus for these instruments the RMS value of the measuring range had to be used for error calculation.

But the world has changed: Real-life signals become more and more distorted and modern instruments like this LMG use A/D converters. The measuring range is now explicitly defined by the largest value that these A/D converters can process. The RMS value may be as big as the peak value (in the case of DC) or even a lot smaller (e.g. for inrush currents). For this reason there is now only one measuring range to take into account for error calculation that can be justified physically in this context: The peak value, which is equal to the operating range of the A/D converter.

The nominal value as well as its possible permitted crest factor can be set to any preferred value: A measuring range with a peak value of 100 V can be labeled as an RMS of 70 V with a corresponding crest factor of 1.43 or as a 5 V RMS with a corresponding crest factor of 20. Or to put it another way: It is reasonable to define a crest factor for a signal, but for a modern digital instrument it is rather unhelpful since it doesn't provide any useful information. What is important is that the peak value of the signal is inside the operating range of the A/D converter.

Consequently, from a technical point of view it doesn't make sense to draw on the RMS value of a measuring range for error calculation in a modern sampling instrument. Therefore ZES ZIMMER specifies their precision using the only technically reasonable figure: The maximum operating range of the A/D converter or for short: the peak value! Traditionally power analyzers were used for AC measuments, but in recent years the number of applications where DC power plays an important role has increased:

- Photovoltaics

- Automotive

- EV charging stations

- Storage

- HVDC (High-Voltage, Direct Current power transmission)

There are even considerations to lay out certain parts of the grid with DC.

> *In principle it is possible to measure DC with simple DC measurement instruments, but there are good reasons to use a typical power analyzer here, too:*
>
> - *There is a certain comfort in not having to first multiply U and I to determine the power.*
>
> - *Often, to determine efficiency, the DC power in proportion to an AC power is required, but the latter is only available with an LMG which can also readily calculate the efficiency.*
>
> - *DC signals are often overlaid by ripple caused by switch frequencies. This is often high-frequency and, depending on phase shifts, yields an active power that cannot be determined with two independent instruments because the phase information is missing. Even though this active power is small compared to the DC portion it still matters when it comes to large efficiencies >95 percent.*

How do AC and DC measurements differ? In modern sampling instruments the signal samples must be in the dynamic range of the A/D converter to be captured correctly. In general one does not think in terms of peak values but rather in terms of RMS values. The ratio of peak to RMS value is the crest factor. Given a mains power of 230 V, in an ideal case this would be $\sqrt{2}$, yielding a peak value of 325 V. In practice it can deviate significantly due to distortions. Thinking primarily in RMS values, it is of course substantially more convenient to directly select a measuring range with nominal value 250 V than to bother with peak values, in which case you have to look up that a range with 400 V peak would be most suitable with a corresponding nominal value of 250 V.

Very often the crest factor of voltage signals is in the region of $\sqrt{2}$, for current signals even higher values up to 4 are not that uncommon. The situation changes a bit with DC: Here, peak and RMS values of the signals are close together, the crest factor is near 1. To continue the example from above, it is very well possible to measure a DC voltage of 350 V in a range with 400 V peak and still have

headroom for ripple. It may feel wrong to select a 250 V nominal range for a peak 350 V signal, but intuitively going up sacrifices precision.

ZES addresses this situation with the new S channels by letting the user choose whether a physical measuring range (characterized by its peak value) should be assigned an AC nominal value (with a large crest factor) or a DC nominal value (with small crest factor), thereby simplifying the correct choice of measuring range, with DC signals in particular.

6.18 Zero Adjustment

The LMG600 series machines are built to have very stable long term DC characteristics, which usually are specified for one year. Nevertheless, over shorter periods of time, small shifts in DC level are normal. To allow for the best DC performance in a measurement, and to avoid frequent adjustments by the manufacturer, the device is equipped with a built-in zero adjustment tool (see ZERO ADJUST [7.3.15→188]). There are external and internal zero adjustments. For A, B, and C channels only external adjustments are possible, S types, however, can adjust the current channel internally. Additionally, these adjustments can be scheduled to run automatically or remind the user to do them.

6.18.1 External adjustment

It is possible to either adjust the entire device (all jacks and all ranges), or to adjust only the currently used groups, jacks, and ranges.

Complete adjustments are stored permanently and will still be available after a restart of the device.

Partial adjustments are intended for fast adjustments of the currently used part of the device. This adjustment is saved to RAM and not to storage, i.e. after a restart, the adjustment is wiped and the last permanent adjustment will be loaded again.

In both cases a very exact zero voltage and zero current must be provided. The zero voltage is achieved best by short circuiting all voltage jacks (U* and USensor with U, and ISensor with I) and to disconnect the current jack (I*). The short-circuit loop of the voltage jacks should be as short as possible, because large loops can easily introduce interferences. Any DC signals connected during the adjustment would be mis-interpreted as internal deviation to be corrected. This would lead to a distortion of the zero adjustment, and could only be fixed by a new zero adjustment under corrected conditions.

Because the DC zero levels are relatively sensitive to temperature changes, it is recommended to give the device enough time for warm up prior to the adjustment.

6.18.2 Internal adjustment of current channels

Unlike external calibration, there's no need to turn off or disconnect the measurement signal during internal calibration, making it possible to calibrate while the device is in operation. Typically, internal calibration is automatically performed whenever there's a change in the measurement range or filter. Additionally, it's refreshed if there's a significant temperature change or if an extended period of time is spend within a single measurement range. This can be configured in the [**INSTR.**] menu.

6.19 Calculation of flicker, option L6-OPT-FLK

The flicker option offers a one or three phase flicker meter in group 1 of the instrument. This group has to be setup with one or three channels respectively. The flicker measurement takes about 8 s after starting, until valid values occur.

The instrument itself can measure P_{st}, P_{lt} and P_{inst} according to IEC/EN 61000-4-15. For measuring the d-meter values, the instrument delivers the raw half wave rms values. An external software (see CE CONFORMITY TESTING WITH LMG TEST SUITE [8.4→205]) then calculates the d-values ($d(t)$, d_c and d_{max}) and compares them to the limits.

Due to the fast generation of the halfwave wise calculated flicker values, the transfer has been optimized by sending the data in blocks of 20 halfwaves. I.e. requesting any flicker measurand, timestamp or state will return a list of 20 values. With a frequency of 50Hz a new block of 20 flicker values would generated approx. every 200ms. Concatenating these blocks results in gapless series of flicker measurands

Any reconfiguration stops a running flicker measurement. Thus, the autorange should be deactivated and any other configurations need to be done before the starting a flicker measurement. The reason for this is, that the flicker measurement is defined as a gapless measurement and a change in configuration could interrupt the measurement.

6.20 Current overload protection

In the current path of the measuring channel different shunts are placed for different ranges. To protect these shunts in case of a massive overload, there is a protection circuit which works as follows: If the sample values in the cureent range are bigger than the allowed range (due to an overload), a special time window of 1 s starts. If during this time window more than 50 % of the sample values are bigger than the allowed range, the measuring channel enters a safety state. In this safety state, the impedance is set to a value, where as few power as possible is converted in the channel. This helps, that the shunt resistors do not change their values due to thermal stress. After about 5 s this state is left and normal operation continuous.

As there is no defined shunt active, it is not possible to determine values and as a consequence the values in the GUI are not updated. Just the overload indicator at the bottom of the instrument blinks.

6.21 PSI, the processing signal interface, option L6-OPT-PSI

Many applications need auxiliary values, that have to be provided. Examples are values from temperature sensors, speed or torque sensors, auxiliary voltages, or alarm signals. Such values can be handled by the processing signal interface (PSI). This interface has several inputs and outputs. In this section, the working principle is shown. For the technical data, see PROCESSING SIGNAL INTERFACE L6-OPT-PSI [4.12→78].

In general any LMG600 (with the exception of the single-channel LMG61x) can be equipped with a PSI card (option L6-OPT-PSI) in addition to the power channels. The LMG671 and LMG641 further offer the possibility to install a second PSI card (requires firmware version 3.041 or later).

The processing signal interface provides the following features:

- 8 switching outputs,
- 32 analog outputs,
- 8 switching inputs,
- 8 analog inputs,

- 2 fast analog inputs (not in motor mode),

- 2 frequency inputs (not in motor mode), and

- motor mode with speed, torque and power measurement.

- Id-Iq-transformation

The outputs are updated after a measurement cycle. The inputs are sample-based with a particular sampling rate, and an averaging over the measurement cycle is provided.

The processing signal interface is assigned to a reference group specified by PSIGROUP [9.10.327→371]. The measuring time of the PSI values is determined by the synchronization of the reference group in the same way as for the normal values.

The motor mode can be activated/deactivated by PSIMODE [9.10.331→373].

6.21.1 Switching outputs

There are 8 Switching outputs. A measurement value can be chosen by PSISWSOURCE [9.10.339→377]. This measurement value will be compared with the value defined by PSISWTHRESH [9.10.340→377]. The compare operation itself can be chosen by PSISWCOND [9.10.337→376] and PSISWINV [9.10.338→376].

This defines a logical statement. If this statement is fulfilled the value of the switching output PSISWITCH [9.10.162→299] is set to 1, and the switch is closed SWITCHING OUTPUTS [4.12.4→81]. Otherwise the value is set to 0 and the switch is open.

The script has the possibility to overrule the result of the logical statement and therefore the state of the switch itself. The can be done by a built-in function FUNCTIONS [6.15.5→130].

6.21.2 Analog outputs

The PSI has 32 analog outputs. It is possible to generate an output voltage that depends linearly on an arbitrary chosen measured value x. This measured value can be chosen by PSIAOSOURCE [9.10.299→358]. The output voltage is computed by

$$y = y_a + \frac{y_b - y_a}{x_b - x_a}(x - x_a)$$

where x_a, x_b, y_a, and y_b can be set by PSIAOXA [9.10.300→358], PSIAOXB [9.10.301→358], PSIAOYA [9.10.302→359], and PSIAOYB [9.10.303→359]. The output voltage itself can be fetched by the short command PSIAOUT [9.10.149→294]. The output voltage is cut off at the full scale value. For the full scale value of the output voltage see ANALOG OUTPUTS [4.12.3→80].

6.21.3 Switching inputs

The 8 switching inputs are used to read in the state of an external logic signal. The signal is internally sampled with about $150\,\text{kS/s}$ and can be viewed in any scope. using the sample source PsiSwitchIn. The value PSIDIN [9.10.152→295] is determined by majority decision based on all sample values within the measuring time of the cycle. If PSIDIINV [9.10.305→360] is set to 1, the logic is inverted.

6.21.4 Analog inputs

There are 8 analog input signals with an internal sampling rate of about $250 \,\mathrm{S/s}$ available. They can be viewed in any scope by the sample identifier PsiAin. The average over the measuring cycle can be fetched by PSIAIN [9.10.147→294]. Using PSIAIXA [9.10.295→356], PSIAIXB [9.10.296→356], PSIAIYA [9.10.297→357], and PSIAIYB [9.10.298→357], the analog input values can be modified according to

$$y = y_a + \frac{y_b - y_a}{x_b - x_a}(x - x_a).$$

6.21.5 Fast analog inputs

There are 2 fast analog inputs with a sampling rate of $150 \,\mathrm{kS/s}$ available. One can set up the filter as follows: Choose via PSIFILT [9.10.311→362] whether the filter settings are taken from the reference group specified by PSIGROUP [9.10.327→371] or configured by PSI card-specific settings. With PSIHPCOF [9.10.312→363], PSIHPFILT [9.10.313→363], and PSIHPTYP [9.10.314→364] you can set up the highpass filter. With PSILPCOF [9.10.315→364], PSILPFILT [9.10.316→365], and PSILPTYP [9.10.317→365] you can set up the lowpass filter. To scope these samples use the sample identifier psiFain (see SAMPLE SOURCES [6.14.4→124]). The averaged value over a cycle can be fetched by PSIFAIN [9.10.148→294]. Using PSIFAIXA [9.10.307→360], PSIFAIXB [9.10.308→361], PSIFAIYA [9.10.309→361], and PSIFAIYB [9.10.310→362] the result values can be modified according to

$$y = y_a + \frac{y_b - y_a}{x_b - x_a}(x - x_a).$$

6.21.6 Frequency inputs

There are 2 fast frequency inputs available. Any associated interface command can be suffixed with 1 or 2. For the pins of the A, B, and Z signals see SPEED/TORQUE/FREQUENCY INPUTS [4.12.6→82]. The signal types can be configured by PSIFRQTYPE [9.10.322→368] and the filter by PSIFRQFILT [9.10.320→367]. For the bandwidth of these signals see MAXIMUM VALUES FOR FREQUENCY INPUTS AT DIFFERENT FILTERS AND SIGNALS [4.67→83]. These signals can be scoped with a reduced maximum sampling rate of $150 \,\mathrm{kS/s}$ by using the sample source identifiers PsiA, PsiB, and PsiZ. The frequency PSIFREQ [9.10.155→296] is determined from the A signal. The direction PSIDIR [9.10.153→296] is determined from the A and B signals. The frequency and the direction can be scoped with the sample source identifier PsiFreq and PsiDirection respectively.

The position angle can be scoped with the sample source identifier PsiTheta. It counts the tick of the A signal. The Z signal resets this counter to zero. Using PSIFRQXA [9.10.323→369], PSIFRQXB [9.10.324→369], PSIFRQYA [9.10.325→370], and PSIFRQYB [9.10.326→370], the frequency values PSIFREQ [9.10.155→296] can be modified according to

$$y = y_a + \frac{y_b - y_a}{x_b - x_a}(x - x_a).$$

6.21.7 Motor mode

To activate/deactivate the motor mode use PSIMODE [9.10.331→373]. The filter settings and the measurement time are the same as for the direct mode.

Torque

The torque value M can be fetched by PSITORQUE [9.10.163→299]. PSITRQTYPE [9.10.343→378] determines whether the torque is measured by the 1. frequency input or by the 1. fast analog input.

For the 1. frequency input the torque value is computed according to

$$M = M_{\mathrm{nom}} * \frac{f - f_0}{f_{\mathrm{span}}},$$

where f_0 is torque zero frequency PSITRQFRQZERO [9.10.344→379], f_{span} is the torque frequency span PSITRQFRQSPAN [9.10.342→378], M_{nom} is the nominal torque of sensor PSITRQNOM [9.10.341→378], and f is the applied frequency.

For the 1. fast analog input the torque value is computed according to

$$M = M_0 + (M_{10} - M_0) * \frac{U}{10\,\text{V}},$$

where U is the applied voltage, M_0 is the torque for zero voltage PSITRQVOLTAGEZERO [9.10.348→381], and M_{10} is the torque for 10 V PSITRQVOLTAGESPAN [9.10.347→380].

Speed

The speed value can be fetched by PSISPEED [9.10.161→298] and the direction by PSIDIR [9.10.153→296]. PSISPTYPE [9.10.334→374] determines whether the speed is measured by the 2. fast analog input or the 2. frequency input.

If the speed is measured by the 2. frequency input PSIPULPERREV [9.10.333→374] specifies the number of pulses per revolution.

For the 2. fast analog input the speed value n is computed according to

$$n = n_0 + (n_{10} - n_0) * \frac{U}{10\,\text{V}},$$

where U is the applied voltage, n_0 is the speed for zero voltage PSISPEEDVOLTAGEZERO [9.10.336→375], and n_{10} is the speed for 10 V PSISPEEDVOLTAGESPAN [9.10.335→375].

Power

The mechanical power value P_{mech} can be fetched by PSIPOWER [9.10.160→298]. It is computed by

$$\text{POWER} = 2\pi \frac{\text{SPEED}}{60} * \text{TORQUE}.$$

This power is included in the efficiency calculation.

6.21.8 Id-Iq-Transformation

For synchronous motors, the Id-Iq-transformation is a composition of a Clarke transformation and a Park transformation. They are sample based.

We start with the samples i_a, i_b and i_c of the three currents of the Id-Iq group, that can be configured by IDIQGROUP [9.10.328→371]. The Clarke transformation reads

$$
\begin{aligned}
i_\alpha &= \frac{1}{3}(2\,i_a - i_b - i_c), & (6.53)\\
i_\beta &= \frac{1}{\sqrt{3}}\,(i_b - i_c),\\
i_0 &= \frac{1}{3}\,(i_a + i_b + i_c)
\end{aligned}
$$

and the Park transformation

$$
\begin{aligned}
i_d &= i_\alpha \cos n\,[\vartheta_0 + \vartheta(t)] + i_\beta \sin n\,[\vartheta_0 + \vartheta(t)] & (6.54)\\
i_q &= -i_\alpha \sin n\,[\vartheta_0 + \vartheta(t)] + i_\beta \cos n\,[\vartheta_0 + \vartheta(t)],
\end{aligned}
$$

where n is the number of pole pairs specified by PSIPOLPAIR [9.10.329→372]. The position angle of the motor is

$$\vartheta(t) = \frac{2\pi}{PSIPULPERREV}\,N(t),$$

with the config variables PSIPULPERREV [9.10.333→374], where $N(t)$ is the number of counts, that is detected by the 2. frequency input of the PSI card. Moreover,

$$\vartheta_0 = \frac{2\pi}{360°} PSITHETANULL$$

with the config variable PSITHETANULL [9.10.330→372]. The scope provides the samples of i_d by the sample source psiId, i_q by psiIq, i_0 by psiINull, and $\vartheta(t)$ by psiTheta2 (see SAMPLE SOURCES [6.14.4→124]). The corresponding average values over a cycle can be fetched by PSIID [9.10.156→297], PSIIQ [9.10.158→297], and PSIINULL [9.10.157→297].

Remarks

The filter delays of the currents are not automatically adjusted and the value ϑ will not be adapted.

The value of ϑ_0 is time independant.

With the setting of ϑ_0, one can compensate an offset, that is independent of the angular speed. For example, if the counting disc is assembled with a deviation.

By setting the value of ϑ_0, filter delays can be compensated only if the frequency is known and fix.

7 Graphical user interface (GUI)

The graphical user interface (GUI) allows users to control the instrument's measurement core via the front panel. By using touchscreen and keys, one can configure the instrument and then view the measured values on the instrument's display.

The GUI is internally connected to the measurement core and communicates with it by using the interface commands as described in REMOTE CONTROL [9→215]. In this manner, the GUI behaves like any other user application which communicates with the instrument via an interface such as RS232, LAN, or IEEE488.1.

The commands internally used by the GUI are also of interest to users who are using another interface to remotely control the instrument. Therefore, the most important commands are marked in the margin like in the example here. This chapter describes just the general handling but by following the link in the margin, one can find the complete documentation for each command.

IRNG [9.10.252→333]

7.1 General GUI handling
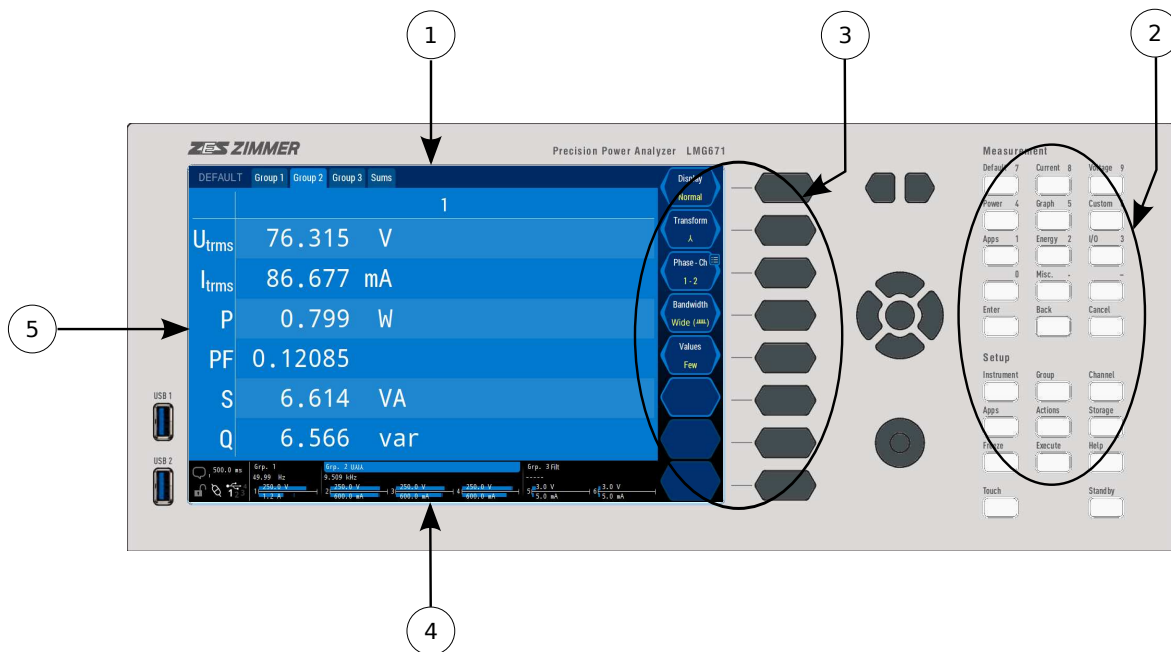
Figure 7.1: Elements of the user interface (LMG671)

The user interface consists of the following elements (as shown in Figure 7.1 [→145]/Figure 7.2 [→146]):

1. Touchscreen

2. Fixed function keys

3. Soft keys
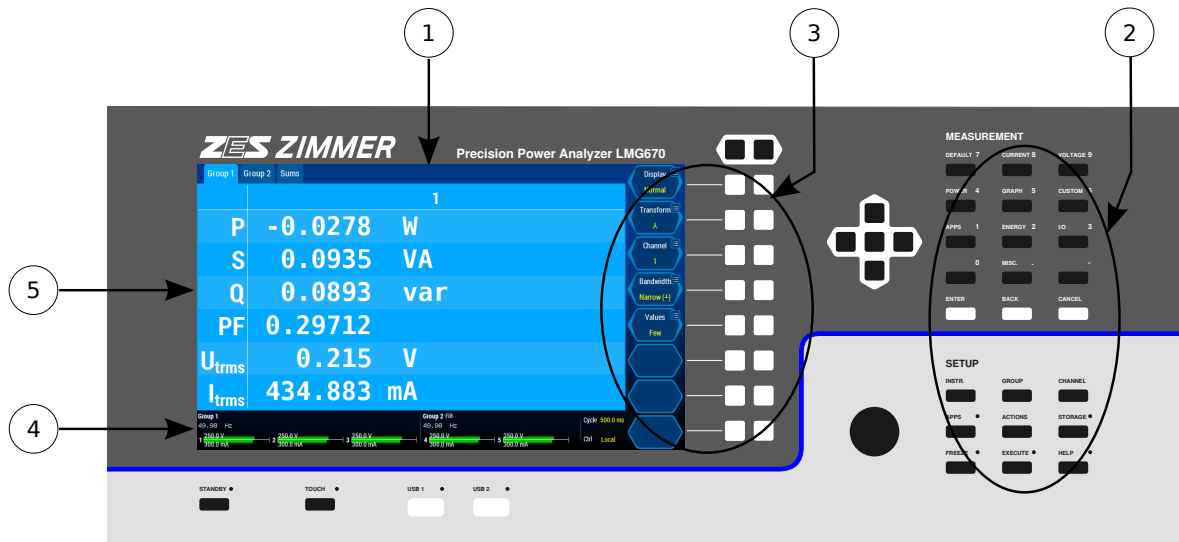
Figure 7.2: Elements of the user interface (LMG670)

4. Status bar

5. Menus and dialogs

Touchscreen

The touchscreen offers the most direct and intuitive handling of the instrument. If for any reason the touch functionality needs to be deactivated (e.g. a user wants to point at something directly on the screen without the risk of accidentally activating any function), the [**TOUCH**] key can be used to deactivate the touchscreen completely. One can *always* operate the instrument without using the touchscreen by using fixed function keys and/or soft keys only.

External Touchscreen

It is also possible to use an external Touchscreen with the LMG. See TOUCHSCREEN TAB [7.3.11→182] for setting up an external touch device.

External Mouse

If needed, the GUI can be controlled using an external USB mouse. Thereby, the mouse cursor works like a finger and the usage is the same as with direct touch control.

Fixed function keys

The fixed function keys in the Measurement and Setup blocks have specific functions which are printed above each key. The brightly illuminated text indicates what action will occur when the key is pressed. If the text is dark, then this key is deactivated and nothing will happen when it is pressed.

Soft keys

At the right side of the screen, the displayed functions of the soft keys are displayed. Each soft key has two adjoined physical keys. Pressing either the right or left physical key correlates to pressing the right/left side of the touchscreen's soft key. If there are just two choices (like an on/off selection), it does not matter if one presses the left or right key. If more choices are available, one can scroll through the list of available settings by pressing either the right or left key.

A symbol in the upper right or left corner of the soft key indicates that an action (illustrated by the symbol) will be triggered when a long press (i.e. more than 600 ms) occurs on the respective side of the soft key. There are just a few long-press symbols available:

- Symbol for entering text or a value. A long press on this side of the soft key opens a dialog to enter/edit a text or numerical value.

- Symbol for selecting an item from a list. A long press on this side of the soft key opens a dialog where one can select from all available choices.

- Symbol for configuration. A long press on this side of the soft key opens a configuration dialog.

- Symbol for action. A long press on this side of the soft key will perform the action displayed in the soft key.

The exact meaning and behavior of the soft keys depends on the context. In general, *white* text can provide information about the meaning of the soft key or also about a value which can not be changed. *Yellow* text always indicates a setting which can be changed by pressing the soft key. There are just a few general types of soft keys as illustrated in Figure 7.3 [→147], Figure 7.4 [→147], and Figure 7.5 [→147]. A disabled soft key is shaded to indicate that this soft key cannot be pressed (see Figure 7.6 [→147]). Nevertheless, this disabled soft key could display the actual state of a value as described above (see Figure 7.7 [→147]).

Figure 7.3: Soft key displaying a state. The current scaling is 1.0. By pressing the soft key left or right, one can change this scaling. A long press will open a dialog to edit the scaling value.

Figure 7.4: Soft key to change an entry from a list. The selected value of the list is displayed. By pressing either the left or right side of the soft key, the previous or next value of the list is chosen, respectively. In most cases, the selection will be wrapped-around to beginning once the end of the list is reached. A long press will open a selection dialog.
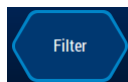
Figure 7.5: Soft key to trigger an action. In this example, the triggered action is to edit filter settings. When this soft key is pressed, a sub-menu is opened which allows users to configure the filter settings.

Figure 7.6: Soft key which is currently disabled as the key is not applicable in the current context. Pressing has no effect. Once the context is changed in such a way that this soft key has a purpose, the soft key will be activated again.

Figure 7.7: A disabled soft key which still displays a state. Although similar to the soft key in Figure 7.6 [→147], this soft key displays the current setting. This happens usually in conjunction with automatic functions. In this example, the auto-ranging mechanism is active, so it is not possible to manually change the range but the soft key provides information on which range was selected.

Status bar

Located at the bottom of the screen, the status bar provides summarized information about the most important states of the instrument. One can see information such as which channels belong to which group and if any channels are over-ranged and the selected ranges. If the signal filter is active, the synchronization frequency for successful synchronization is displayed.

If the frequency of a group is displayed as dashes, this group is not synchronized to the measured signal.
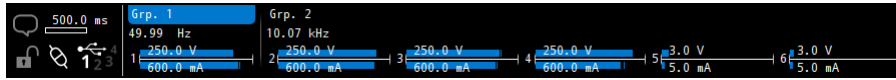
Figure 7.8: Status bar which is located at the bottom of the screen

By touching an over range indicator, one opens directly the [**CHANNEL**] *menu to change the range. If one touches a filter setting, one access the* [**GROUP**] *menu where the filter setting can be configured.*

The meaning of the symbols in the lower left corner is as follows.

First row:

- **Speech bubble** Blinks if errors or warnings are present. Tapping the symbol will show the Information in a dialog. Any time later, the occured messages can be reviewed, as long as the device has not been restarted.

- **Cycle time / Cycle mode and Cycle bar**. Here can be seen the current cylce mode, if indicated the current cylcle time, and the progress of the current cycle. This settings can be configured after tapping this area. In the mode "Fixed Interval" the progress of the current cycle can be seen for cycle times of 500ms and longer. In most other cases, an animated bar indicates the measuring activity of the device.

Second row:

- **Lock** The lock indicates wheter or not the front panel is locked. If the lock is grey and open, the configuration of the device can be changed by using the front panel. If the lock is closed and white, the front panel is locked an can only be used for reading of measurands but the configuration of the device is not possible.

- **Interface connection** If currently an interface connection is active (via LAN or RS232) the connectors of this symbol are connected and white. If the connectors are not connected and grey, no connection is active. The active (externally controlled / remote) will show up as soon a s communication occurs. If the communication is not being triggered contiuously, a tap on this symbol will switch the device from "Remote" to "Local".

- **USB ports** This symbol indicated which USB ports are in use with a USB storage device. If the device is being accessed, the respective port number flashes. While being in this state, the USB storage device must not be removed. Connection of a keyboard and of a computer mouse ist possible but will not be indicated by this symbol.

Menus and dialogs

The remaining screen is reserved for the display of measurement menus, setup menus and setup dialogs.

By pressing a key from the *MEASUREMENT* area, one can view any menu displaying measured values. What is currently displayed in these menus can be modified by the soft keys. The measurement menus all have a blue color scheme.

To change any instrument setting, one can press any key from the *SETUP* area. Then the associated setup menu is displayed on top of the underlying measurement menu. All the relevant settings are displayed in the setup menus. Here, one can reach further sub-menus, setup dialogs or directly change some parameter values. The soft keys will change accordingly.

In the setup *menus*, one has has the following possibilities:

- Opening setup dialogs by pressing the relevant <**soft key**> or by ¦touching¦ the relevant element on the screen.

- The wheel can be used to navigate the dashed selection frame to the setting to be changed. If a setting has only two possible states (e.g. On/Off) a short ¦wheel¦ press will toggle between these. If more than two states are available, pressing the ¦wheel¦ shortly will change the dashed selection to a solid selection. This indicates, that rotating the wheel will change the setting directly. The solid frame will turn dashed after approx. two seconds after last action. Long press of the ¦wheel¦ (> 600 ms) while the frame is dashed opens a dialog to display all possible states as would do the corresponding <**soft key**>.

- Navigating to a sub-menu by pressing the relevant <**soft key**> or by ¦touching¦ the relevant element on the screen.

- Leaving the currently opened sub-menu and returning to parent menu by pressing [**BACK**].

In the setup *dialogs*, one has the following possibilities:

- Entering a setting. By pressing ¦the desired element¦, it is directly selected and the dialog is immediately closed. For some settings, one can select an element with the wheel and then press [**ENTER**] which sets the selection and closes the dialog.

- Canceling the dialog. By pressing the [**CANCEL**], the dialog's settings are reverted to the previous settings.

- Previewing different settings by pressing <**Preview**>.

- Entering text or numbers. See below for additional information.

The setup menus and dialogs are displayed in a gray color scheme so that they are clearly differentiated from the *MEASUREMENT* menus.

7.1.1 On-screen menu
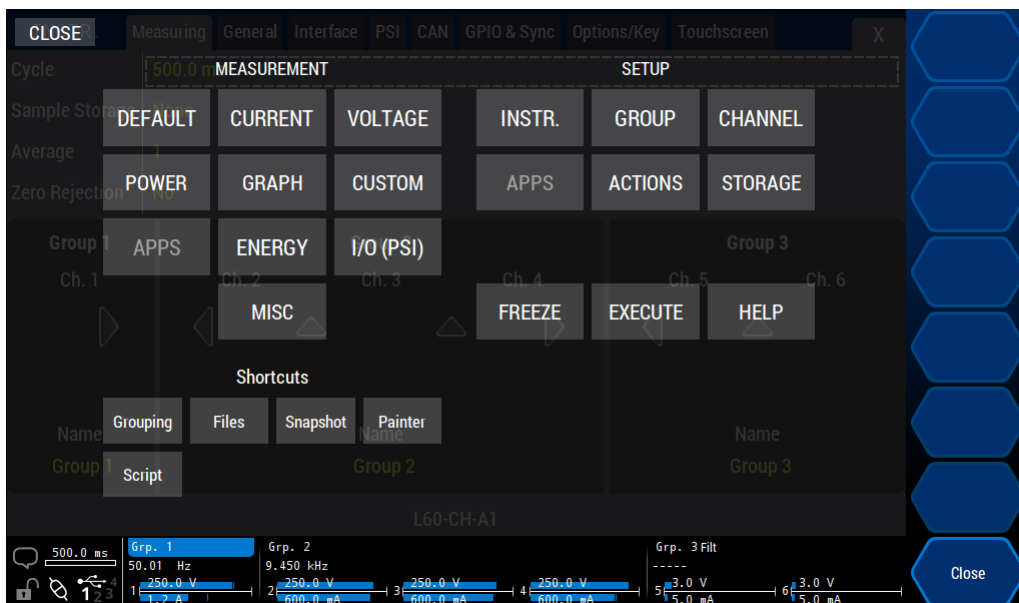
Figure 7.9: The on-screen menu. It can be opened by tapping the upper left corner of the screen

The on-screen menu allows to navigate between measurement and setup menus without using the LMGs front panel. This is especially useful for operating an external touchscreen because

for most usecases, direct access to the device is no longer necessary (see Touchscreen tab [7.3.11→182]).

In addition to navigating the regular menus, the on-screen menu allows to trigger freeze, execute the set action or enter the help screen. On the lower left-hand side some shortcuts and access to useful locations and tools are provided.

The on-screen menu can be opened by tapping the upper left corner in measurement or setup menus. It is not possible to open it while any dialog is being displayed (e.g. a message dialog or number entering dialog).

7.1.2 On-screen painter

Figure 7.10: The on-screen painter. It can be opened in the on-screen menu (see On-screen menu [7.1.1→149]).

The on-screen painter is part of the option L6-OPT-SMV. It can be opened using the "Painter" shortcut in the on-screen menu (see On-screen menu [7.1.1→149]). This tool enables to freely draw sketches on top of any measurement or setup menu. This can be very useful for highlighting interesting features of a curve or values. Within the paint tool <**Take Snapshot**> allows to take a screenshot anytime regardless whether the snapshot action has been assigned to the "EXECUTE" key before or not.

The drawing is saved automatically and will reappear after closing the painter or rebooting the device. The painting is not part of a saved configuration.

Following painter functions are available.

- <**Clear**> Clears everything (can not be undone).

- <**Undo | Redo**> Allows to undo and redo the drawing steps. Using the rubber will allow to redo no farther than up to the rubber usage.

- <**Tool**> Chooses the active tool from the tools like freehand, line, rectangle, and rubber. Using the rubber will show up dots on the different parts of the drawing. Tapping close to such a dot will remove the respective part.

- <**Line Color**> Opens a color choosing dialog for picking a new drawing color.

- <**Line Width**> Allows to choose the width of the line (max. 24 pixel).

- <**Take Snapshot**> Triggers the snapshot action.

- <**Close**> Closes the painter. The drawing will be saved automatically and will reappear when it's opened again.

Entering values

Pop-up dialogs are provided to change or enter values such as text or numbers. The currently configured value is provided in the edit field of the dialog. As the value in the edit field is preselected, indicated by the blue background, the actual value will be replaced completely if one enters any value. For example, if the old value is 2.0 and the user wants to have a new value of 0.5, then it is faster to enter the three characters ("0.5") directly than to edit and modify the former value.
However, it is always possible to edit any existing entry by moving the arrow keys. In this case the blue selection vanishes and one can change the entry character-by-character. The [**BACK**] can be used to backspace and delete the character left of the cursor position.

The scroll wheel is used to navigate through the keys of the virtual on-screen keyboard. The selection key (located in the middle of the arrow keys) is used to press the currently selected key of the virtual keyboard. For entering numbers, one can also use the numerical keyboard which uses the keys located in the Measurement section ([**DEFAULT**], [**CURRENT**], etc.).

Unit prefixes are available when entering numerical values. So instead of typing 0.005, one could just type 5 and then <**m**>.

Preview mode

When settings are changed, the instrument will process these changes as soon as possible. Many dialogs offer a preview mode (accessed by <**Preview**>) that allows one to immediately see the effect that the changed settings have on the measuring environment. For example, the preview mode can be used to view the changed waveform directly after the filter frequency has been changed.
The preview mode is not available for all settings as the instrument requires additional time when setting some parameters. Such parameters are not suitable for swift previewing in the GUI. The group setting (located in the Measurement tab [7.3.2→176]) is an example of one such instrument setting where the preview mode is not available.

Tabs and sticky properties

The contents of most menus are organized into tabs which can either be selected through clicking the desired tab or by using the 'Tab Keys' which are located above the soft keys on the front panel, see Figure 5.4 [→88]/Figure 5.5 [→89]. Most tabs are organized so there there is a separate tab for each group. Additionally, there is often a "Sums" tab which contains overview of the sum values of every group.

In most menus, the active tab can be switched using the 'Arrow Keys' $\lfloor$left/right$\rfloor$ in addition to the 'Tab Keys'.

The selected group (i.e. active tab) is a sticky property which applies to the GUI. For example, if one selects "Group 2" tab in the [**VOLTAGE**] menu and then changes to the [**CURRENT**] menu, the "Group 2" tab will also be selected in this menu. Similarly, if one then changes to the [**CHANNEL**] menu, one can immediately configure the channels of the second group directly as the "Group 2" tab is also selected here.

A similar sticky behavior also occurs if one has selected a specific channel, phase, link or transformation inside a group. If one changes the menu, this selected information will continue to be selected for that specific group.

For most applications, this sticky behavior is desirable. For example if a user configures a special voltage view (i.e. a certain channel, phase or transformation of a specific group), it is most likely that the user will want to see a corresponding view of current values if the user navigates to the current menu.

Display of values

Measured values are displayed, whenever possible, with 6 digits. If the values directly depend on a range (i.e. voltage, current, power) the decimal separator is fixed so that is suitable for the selected range.

> *Example:*
> *In a 250 mV range, the values of this channel will be displayed with three digits before and 3 digits after the decimal separator:* `123.456 mV`. *If there are smaller values, they are also displayed with 3 digits after the decimal separator and have the same prefix. This results in a display where the decimal separators line up correctly which help one to quickly read and compare the numbers:*
> `Utrms  123.456 mV`
> `Udc      1.234 mV`

There are some special displays for particular values:

- - - - - - -

  This is used to represent invalid values, i.e. values which could not be measured or calculated for any reason. For example, the frequency is displayed this way if it could not be measured. This can happen if one, for example, applies a DC signal. This kind of value is also known as not a number or NaN.

- ^^^^^^

  This is used to indicate that a value is outside of the displayable range (i.e. the amount is bigger than $99.9999*10^{12}$)

- 0.0

  If a value is exactly zero, it is displayed this way.

- 0.00000

  If a value is too small to be displayed (i.e. smaller than $1.00000*10^{-9}$), it is displayed this way.

- +Inf./-Inf.

  Indicates, that the result of a calculation was too big or small to be represented by the internal data type. This can happen, if a division by zero takes place.

In the STANDARD MENUS [7.2.1→153] (such as [**DEFAULT**] and [**VOLTAGE**]), the instrument displays only values measured from a unique circuit such as bandwidth star circuit (λ) or delta circuit (Δ). In other words, measurements from a star circuit (λ) or delta circuit (Δ) are never displayed at the same time. Similarly, when dual processing is enabled, narrow bandwidth (⊥) or wide bandwidth (⊔⊔⊔⊔) measurements are never displayed at the same time. Although every value does not have its own identifier in the standard menus, the exact meaning of a value can be determined through the context of the table and soft keys.

However, there are also menus (such as the plot, scope and custom menus) where these values can be mixed (e.g. narrow and wide bandwidth values are simultaneously displayed). In these cases, there exists a general nomenclature which uniquely identifies each displayed value. This is illustrated in the following examples:

The example $U_{trms}3\,G_2^{⊥}$ defines the following:

- The value is a Utrms value.

- It is the 3rd value of the 2nd group.

  > *If this second group has 3 channels and the wiring is set to UλIλ or UΔIλ, this is the voltage of phase 3 against the star point. So this is a star value (λ).*
  > *If this second group would, for example, have 4 channels and the first group contains two channels, then this phase would be the fifth physical channel.*

- This value is a narrow bandwidth value.

The example $U_{trms}31\,G_2^{⊔⊔⊔⊔}$ defines the following:

- The value is a Utrms value.

- It is a linked value between phases 3 and 1 of the second group.

  For this group, a star-delta conversion is activated and the measured value is the delta value (Δ) between the phases 3 and 1.

- This value is a wide bandwidth value.

In the STANDARD MENUS [7.2.1→153], a subset of the complete identifiers is displayed in table column headers as other information is already displayed in the soft keys.

Throughout the user interface, the term channel *is used to refer to physical channels. The term* phase *is used to refer to the corresponding phase within a group and the term* link *is used for linked values. E.g.* `channel four` *and* `five` *are assigned to group 2. Within this group channel four is referred to as* `phase one` *and channel five is referred to as* `phase two`.

7.2 Measurement menus

In the measurement menus, all measured values and other useful information can be accessed.

The different STANDARD MENUS [7.2.1→153] are organized according to the kind of values. For example, current values can be found in the [**CURRENT**] menu, voltages in the [**VOLTAGE**] menu, and power in the [**POWER**] menu. In the [**DEFAULT**] menu, an overview of the values (i.e. selected mix of voltage, current and power values) of a P phase is provided.

Different kinds of graphical representations are available in the [**GRAPH**] menu: Scopes with sample values, plot (i.e. trend graphs) with values over a measurement cycle, spectra with harmonic values and vector diagrams.

In the [**CUSTOM**] menu, one can define user scripts with formulas to calculate custom values. Furthermore, one can define a customized measure screen by compiling and arranging specific values, graphs and images.

In the [**APPS**](Measure) menu, measurements which were calculated by the activated apps are provided. The values presented in this menu are selected and arranged in a way to get an optimized view of the most important values provided by each app.

In the [**I/O**] menu, access to the values of the processing signal interface is provided.

Finally, the [**MISC.**] menu offers further information such as serial number, adjustment dates and software version.

7.2.1 Standard menus

The standard menus are [**DEFAULT**], [**CURRENT**], [**VOLTAGE**], and [**POWER**]. As the control of each these menus is almost identical, their handling is explained here just once:

The first soft key <**Display**> allows to toggle between "Normal" (C-VALUES [→113]) and "Harmonics" (H-VALUES [→113]).

<**Transform**> allows one to control which transformation of the values is displayed. This is only possible to change, if the option star-delta transformation is activated and the wiring of the group is *not* set to "Direct".

- λ
  The values provided are using the star circuit transformation (λ).

- Δ
  The values provided are using the delta circuit (Δ) transformation.
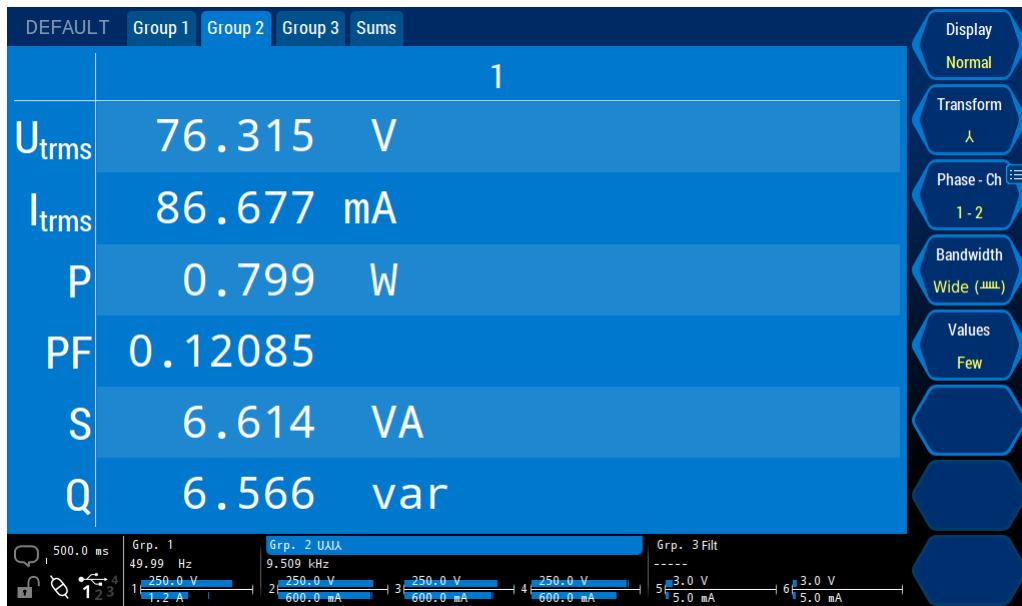
Figure 7.11: A standard menu displaying a few values from the first group. This menu shows normal values in the default menu. The values displayed here are from the first phase of the first group in star circuit with narrow bandwidth.

If no transformation is possible, the directly measured values are always displayed (i.e. the values which apply directly to the phase are displayed).

<**Phase - Ch**> allows one to select which phase/linked value of the group is displayed. This setting works in cooperation with <**Transform**>:

- If no transformation is possible, one can select each *phase* value by choosing the phase number.

- If the transformation is set to λ(star circuit), one can select every *phase* of the star circuit.

- If the transformation is set to Δ(delta circuit), one can select every *link* of the delta circuit.

If possible with the current transformation or if no transformations are available the number of the corresponding physical channel is displayed on the right hand side of the dash.

With this soft key, one can also select "All" or "Σ" to get an overview of all phases of the group (including their sum) or just the sums, respectively.

C-values display

C-VALUES [→113] are, for example, such values as U_{trms}, I_{trms}, and P. It is not possible to list here all available values, but the links located in the margin provide a good starting point. Additionally, all available values can be found in the ":READ" and ":FETCh" section of SCPI syntax tree.

<**Values**> allows one choose the number (and often by this the size) of the values to be displayed. The exact display depends on the number of phases and links being simultaneously displayed (see <**Phase - Ch**>).

<**Bandwidth**> (see BANDWIDTH [6.6→110]) allows one to choose if the narrow or wide band values are shown. This is only possible when the group is in dual processing mode; in single processing mode, this soft key has no effect (see SIGNAL SETTINGS [7.3.12→183]).

If the number of values cannot be displayed in the available space, one can either use the rotary knob, the up/down arrow keys or direcly drag the table content up and down to scroll through the list of values (see Figure 7.12 [→155]).

UTRMS
[9.10.217→319]
ITRMS [9.10.40→250]
P [9.10.132→287]
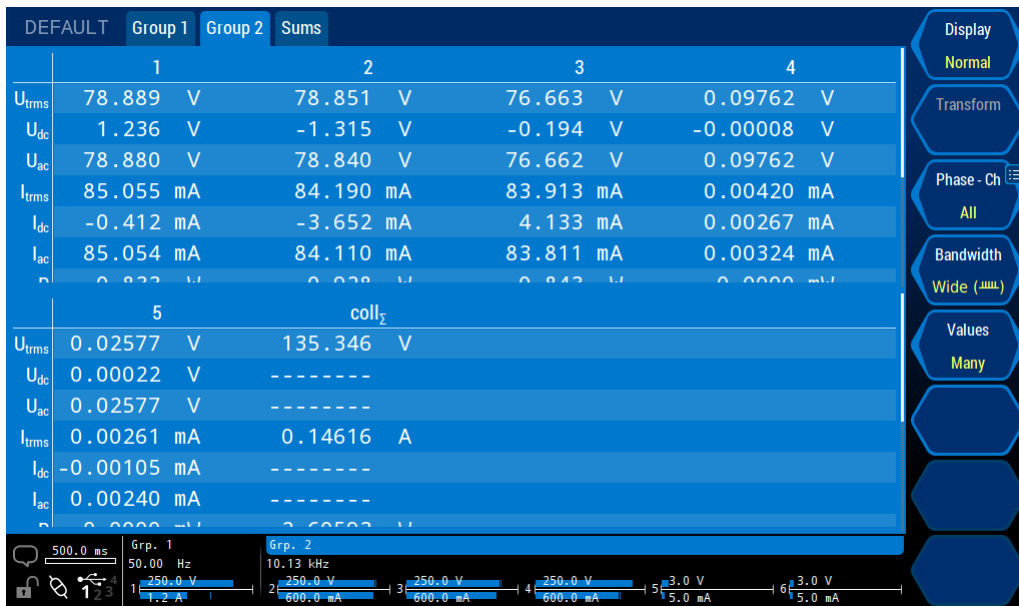
Figure 7.12: Standard menu showing a scrollable list of sum values from each group. This [**DEFAULT**] menu shows an overview of normal values. The menu currently displays the wide bandwidth sum values of all groups. The white scrollbar on the right side of the table (adjacent to the soft keys) indicates, that this table can be scrolled and which area of the entire scrollable range is currently being viewed.
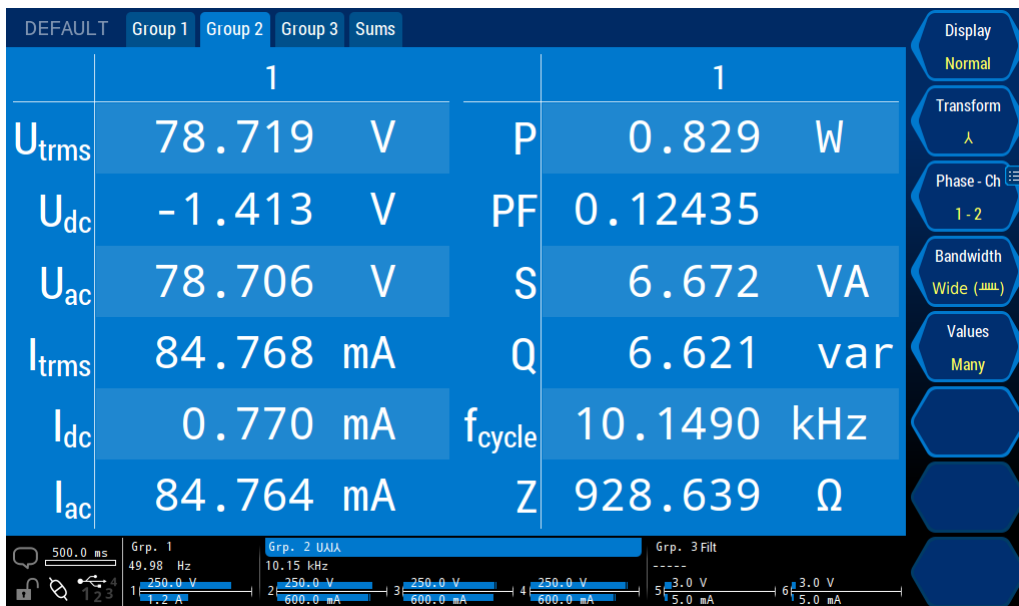
Figure 7.13: A standard menu displaying many values. This [**VOLTAGE**] menu displays the normal values of all three links in delta circuit with wide bandwidth plus the sum values of group 1 are currently displayed.

H-values display

One can scroll through the list of harmonic values by usingt he rotary knob, the up/down arrow keys or dragging the table content up and down.

<**Harmonics**> allows one to toggle between showing either the *odd & even* harmonics or *odd only*. This is often useful, as the even harmonics are usually less interesting as they typically have a very low magnitude. If interharmonics are present, this setting is not available.
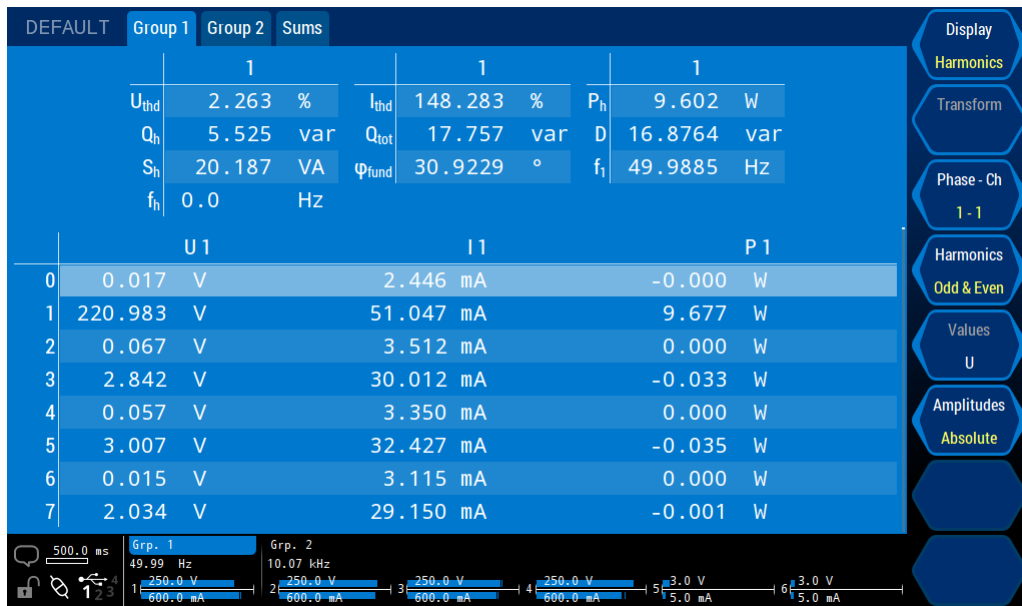
Figure 7.14: A standard menu displaying harmonic values. The [**DEFAULT**] menu displays in this configuration the harmonics voltages of all phases of group 1.

7.2.2 Efficiency view

[**POWER**] followed by choosing the 'Efficiency' tab navigates to this menu. All available efficiencies and power losses are arranged in a grid to allow easy access to the values of interest. Both values are calculated for any pairwise combination of at least two groups including the PSI-card, if available. Within each cell, the upper value shows the efficiency and the lower value the power loss.

By tapping on the grid cells, values of interest can be copied into the list on the right hand side for more clarity. The selection can also be moved around by using the wheel and the cell can be highlighted by pressing the wheel.

The selected cell or the complete list of highlighted cells can be copied into the custom menu.

- <**Copy List**> copies the complete list of highlighted cells as a measurand list into the custom menu.

- <**Copy Eta**> copies the Efficiency of the last selected cell (dashed border) as a single measurand into the custom menu.

- <**Copy Ploss**> copies the power loss of the last selected cell (dashed border) as a single measurand into the custom menu.

In the layout editor of the custom menu (long tap on cell) <**Paste**> allows to insert the copied content (see CUSTOM MENU [7.2.11→163]).

The copied values can also be pasted for logging in the measurand choose dialog of the logger.

The non-clickable diagonal of the efficiency view shows the sum power of each group.

7.2.3 Flicker view

[**VOLTAGE**] followed by 'Flicker' shows a display for flicker measurements, if the respective option is available. Here, basic flicker measurements can be configured and carried out, independent of any external Software. For measurements conforming to standards, the use of additional software

is necessary (see CE Conformity testing with LMG Test Suite [8.4→205]). This display shows independent of the mode of flicker use the current state of the flicker system and the progress of a running measurement.

Flicker measurements are only possible with a group having either one or three channels. It is not possible to start a flicker measurement with other group configurations.

Following parameters can be set up:

- <**Flicker Group**> sets the group to use for the flicker measurement.

FLGROUP
[9.10.278→347]

- <**Interval Length**> sets the interval duration.

FLSTINT
[9.10.282→349]

- <**Num. of Intervals**> sets the number of intervals.

FLNUMINT
[9.10.280→348]

- <**Lamp Model**> sets the lamp model.

FLLAMP
[9.10.279→348]

- <**Frequency**> sets the system frequency.

FLSYSF
[9.10.281→349]

- <**Algorithm**> experimental. "Default" is the setting to use.

'Warm Up Lenght' shows the lead time before the actual flicker measurement begins. This value can not be modified.

Unless a flicker measurement runs, 'Total Duration' indicates the total time of the flicker measurement regarding the current setup. During running flicker measurements the remaining time will be counted down.

The flicker measurement can be controlled as follows:
Please note: <**Start**> changes depending on the state of the flicker measurement to <**Stop**>.

FLICKERSTART
[9.10.97→271]

- If <**Start**> is active, a flicker measurement can be started using the currents settings. If <**Start**> is inactive the configuration of the device, especially the grouping (see above), needs to be checked.

- <**Stop**> stops a running flicker measurement. The up to now measured values stay visible.

FLICKERSTOP
[9.10.99→272]

- <**Reset**> resets the flicker system and the value display (State 'Inactive'). A restart of the flicker measurement after a finished or aborted measurement requires a preceding reset.

FLICKERRESET
[9.10.96→271]

Any reconfiguration stops a running flicker measurement. Thus, the autorange should be deactivated and any other configurations need to be done before the starting a flicker measurement.

The flicker values are displayed in one or three columns (depending on the group setting, see above). The instantaneous flicker value is displayed as P_{inst}, the interval value as P_{st} and the long term value as P_{lt}. While P_{inst} is being updated continuously, P_{st} and P_{lt} are being updated at the end of an interval and the end of the measurement, respectively.

FLPINST
[9.10.93→270]
FLPST [9.10.95→271]
FLPLT [9.10.94→270]

7.2.4 ENERGY Menu

This menu can be accessed using [**ENERGY**] . It displays and controls the integration of various types of power for any given duration.

The navigation works in the same way as with the standard menus (see Standard menus [7.2.1→153]). In addition, this menu allows to control and configure the energy measurement. Independent of the currently visible group, the controls like "Start", "Stop" or "Reset" are applied to all groups. This simplifies the synchronous energy measurement of all groups or channels.

The duration of the energy measurement (t_{dur}) is formatted according to ISO 8601. To provide the most precise value in the given space, the decimal fraction is added to the lowest order time element being second, minute, or hour. The current highest time order is indicated in the unit field.

The measure screens offer following softkeys:

- <**Value Select**> toggles the view between continuous and interval values (see next paragraph).

- <**Mode**> switches between the three different modes of energy measurement start/stop control. "Manual" starts and stops the measurement immediately when <**Start**> or <**Stop**> is pressed. "External" requires in addition to "Start" an external signal on the synchronization interface to start the energy measurement (see SYNCHRONIZATION CONNECTOR [4.3→46] and Figure 4.10 [→46]). "Schedule" starts and stops at the time points set in the "Settings" tab, if the energy system has been started with <**Start**>.

- <**Start**> and <**Stop**> start and stop the energy measurement. If <**Mode**> is not set to "Manual" additional conditions may need to be met (e.g. schedule) or may interfere (e.g. external signal).

- <**Reset**> resets all energy measurands and all integration times of all groups. But the state of the energy system is not affected. If the integration was active before the reset, it will also be active after the reset.

The scheduler and the interval time can be configured in the "Settings" tab. If "Mode" is set to "Scheduler" the start on stop time points can be set. If the energy system is being started the energy will only be integrated between the start and stop time points. If an intervall time different from 0.0 s is set, interval values are being provided in addition to the continuous energy values (see <**Value Select**> above). If an interval time of 5.0 s has been set, the interval values will every 5.0 seconds be updated to show the change of energy of the last 5.0 seconds interval.

7.2.5 GRAPH menus

[**GRAPH**] allows one to navigate to this menu. A variety of graphical representations are provided in different tabs:

- Scope
  Two scopes are available for the GUI. They can be used to display the *sampled* values of voltage, current or power of the measuring channels and, if available, of the PSI.

- Transient
  This scope shows the values recorded by the transient system (L6-OPT-EVT). In addition, it is possible the set up the record and event trigger settings.

- Plot
  Two plots (sometimes called trend graphs) are available. They are used to display values (e.g. U_{trms}, I_{ac} and active power) which are computed over several sampled values (which is typically a cycle time).

- Harmonics
  The two spectra can be used to visualize the results of an harmonic analysis. The information displayed here is the same as what is presented in the [**VOLTAGE**] or [**CURRENT**] menu, but this graphical representation is useful to get an overview.

- Vector
  The vector diagram (sometimes called a Fresnel diagram) is used to get an overview of the amplitudes and phases of a group. This is useful to see if there are any effects such as non-symmetry.

7.2.6 Scope

Two independent oscilloscopes are provided. Each scope can display up to eight signals (i.e. one signal for each track). The scopes can be used to display pure sample values and will synchronize to a selectable group.

If the same signal is chosen one different tracks, one time with small bandwidth and one time with wide bandwidth, there will be a time delay between these two signals. This delay is physically caused by the different bandwidths.

Navigation

All tracks have a common time base which can be changed by <**t/div**> or with the rotary knob. For the currently selected track, the up/down arrow keys are used to move the track in the Y-direction (i.e. changing the Y-offset) and the left/right arrow keys to move the track in the X-direction. By pressing the center of the arrow keys, the default trigger point is set to the center of the screen.

Setup

One can select the desired track by <**Track**>. Long pressing this key opens a dialog to configure the signal for the selected track: <**Signal**> chooses from one of the available signal types. <**Phase/Link**> chooses the desired phase or link of a group. For touchscreen users, these selections can be be configured by using the wheel lists. It is also possibel to to select one of the wheels using the rotary knob. After pressing the knob the selected value can be changed by turning the knob. If the dual-processing mode is selected, <**Bandwidth**> allows one to select between narrow or wide bandwidth for this signal.

SCTRAC
[9.10.352→383]
[_]

<**Arrange:**> provides a way to fit or split the tracks. Pressing the left or right soft key selects an action and a long press then executes the selected action.

The 'Fit' action causes the selected track (or all tracks) to be displayed as large as possible in the Y-direction. Additionally, the rotary knob can be pressed and then the track is fitted in the X-direction so that at least one period of the synchronization signal is displayed.

The 'Split' action first divides the Y-axis by the number of active tracks and then scales and shifts each track so that they do not overlap. This is done by changing y/div and/or the Y-offset.

If a track with a valid signal is selected, <**y/div**> and <**y-Offset**> show the actual scaling and offset of the signal. The values can be decreased or increased by pressing the left or right soft keys, respectively. A long press opens a dialog where one can enter a specific value.

Data cursors can be used to mark parts or measure differences in the scope. Pressing <**Cursors**> turns the display of all cursors on/off. A long press opens a sub-menu where one can configure the cursors. When the display of cursors is switched on, information on the cursors is displayed below the graph. In addition to the time and amplitude position of both cursors, the time and amplitude differences are also presented. These differences are always displayed as *positive* values. The reciprocal value of the time difference (i.e. frequency) is also provided.

With <**Cursor A**> or <**Cursor B**>, one can select which track each the data cursor is assigned to (i.e. the values of which track are displayed for that cursor's position). If the track is not set, the cursor will be inactive and colored gray.

An active Cursor can be directly moved using the touchscreen. It is be done by tap and hold in the vertical area of a cursor and moving it around.

<**Scroll**> allows one select the action of the horizontal arrow keys:

- Time allows one to move the view of the tracks in the X-direction. Cursors are unaffected.

- Cursor x allows one to scroll the respective cursor in the X-direction. If the cursor was outside the visible range before moving, the immediately jumps to the appropriate border.

- Both both cursors are moved in the X-direction.

<**A | Center | B**> moves cursor A or B to the plot center.

<**Snap to value**> allows one to control how the cursors are scrolled. When disabled (i.e. off), the cursors are scrolled pixel-by-pixel through a scope. If a cursor is located at a pixel position which lies between two real sample values, the displayed value is interpolated. When enabled, the cursor can only be placed on real sample value positions. Especially when the time resolution is very low, this can cause the cursor to jump several pixels at once when scrolled.

These actions are only valid while the cursor sub-menu is active.

<**Sync Group**> sets syncronisation group for the scope. The synchronization settings of a group can be accessed via [**GROUP**](see GROUP MENU [7.3.12→182]).

<**Display**> allows to maximize the diagram area.

7.2.7 Transient

This tab contains the event or manually triggered transient scope. All signals available in the scope can be recorded on up to 16 tracks simultaneously. Differently from the scope, the values are not synchronized to a group and transferred continuously, but are triggerd manually or by the occurence of a previously defined event (by signal magnitude and/or trigger pin of the sync. interface).

The navigation and display settings are identical to the scope handling (see SCOPE [7.2.6→158]). The only exception is that <**Sync Group**> is not available. Instead, <**Transient Control**> provides access to transient recording configuration and event configuration.

The upper right-hand corner of the plot area shows the current state of the trigger system. The lower right-hand corner of the plot area shows the number of recorded samples. This number can be used in the log configuration to log the right number of samples.

Transient configuration (Recording)

After tapping <**Transient Control**> additional softkeys for transient record settings show up.

- <**Record Length**> the record length.

- <**Sample Rate**> the sample rate (will be automatically corrected according to the next higher available rate).

- <**Pretrigger**> the pre-trigger length.

- <**Manual Trigger**> triggers the transient immediately. This is only possible, if the state of the transient system is "Searching".

- <**Tr. System Start | Stop**> Left side/key: starts the transient trigger system. The state changes via "Loading" (if pre-trigger length > 0) to "Searching". The pre-trigger buffer is now filled and the trigger system waits for the configured event to occur or can be manually triggered by tapping <**Manual Trigger**>. Right side/key: stops the transient trigger system. The state changes to "Aborted".

- <**Configure Trigger**> opens the event setup dialog (see next section).

- <**Back**> returns to the scope settings.

The "Event Trigger" action allows manual triggering anywhere and anytime in the gui (see EVENT TRIGGER ACTION [7.3.15→189] for details).

If the "Auto-Rearm" feature is in use together with active trigger conditions, contiuous triggering may occour if the conditions are always true. If this is not the desired behaviour, the conditions or the measured signals should be adapted to the need. Deactivation of the "Auto-Rearm" feature or deactivation of the trigger conditions will prevent further triggers. The possibly still searching trigger system can by stopped with <**Tr-System**> -> "Stop".

Transient configuration (Trigger)

After tapping <**Configure Trigger**> in the <**Transient Control**> settings (see previous section), a dialog for the configuration of the event trigger shows up.

Three independently configurable triggers are provided. In addition, the "Transient I/O" pin of the synchronization interface can be configured as input or output.

Each trigger has two freely configurable limits, "Limit A" and "Limit B", against which the signal will be tested. The plot shows the limits together with the signal and the chosen condition.

The first two softkeys fit the graph or move the limits to the specified location of all ("All") or of the currently active ("Cur") trigger. The last softkey allows to access additional trigger settings being equally valid for all three triggers.

The plot is horizontally divided in three evenly sized areas, one for each trigger, from left to right. The area of the currently active trigger is highligted by larger fonts and brighter colors. The trigger to be configured can either be chosen by <**Trigger**> or by tapping the respective colored button at the top of the plot.

The first three softkeys below the softkey labeled "Trigger" configure the signal, the two limits and the condition of the currently chosen trigger. The fourth softkey applies the signal, limit and condition settings of the currently chosen trigger to the two remaining triggers.

> *The trigger and the limits to by configured can be controlled directly by touch in the plot area. The "Transient I/O" pin, the logical linkage ("Link"), the trigger mode ("Trigger On"), and the duration can be konfigured directly by touch on the right hand side next to the plot.*

The function of the individual softkeys is as follows:

- <**Arrange**> arranges all graphs ("All") or the graph of the currently chosen trigger ("Cur") as indicated.

- <**Move**> moves all limits ("All") or the limits of the currently chosen trigger ("Cur") as indicated.

- <**Trigger**> chooses the trigger to be configured. Configuration of limits, signal and condition is applied to this trigger if not indicated differently.

- <**Signal**> brings up a dialog for choosing the signal to be monitored.

- <**Set Limits**> brings up a dialog for manually entering the limits.

- <**Condition**> sets the condition against which the signal will be tested.

- <**Apply To All Channels**> applies the signal, limit, and condition settings of the currently chosen trigger to the two remaining triggers.

- <**More Trigger Settings**> ⇒ <**External Trigger**> sets whether the "Transient I/O" pin of the synchronization interface works as "Input" or as "Output".

- <**More Trigger Settings**> ⇒ <**Logical Linkage**> sets the logical linkage of the three triggers and the "Transient I/O" pin.

- <**More Trigger Settings**> ⇒ <**Trigger On**> sets the desired behaviour when the state changes to "Searching". "Event Edge" will trigger on the first reoccurence of the event (e.g. signal rises above the limit). "Event Level" will trigger immediately if the event already occured (e.g. signal rose in the past and is still above the limit).

- <**More Trigger Settings**> ⇒ <**Event Duration**> sets how long the conditions must be met without interruption to trigger the transient recording.

7.2.8 Plot

Two independent plotters are provided. Each plotter can display up to eight values (i.e. one value per track). The plotters are used to display values (e.g. Utrms, Iac and active power) which are computed over several sampled values (which is typically a cycle time).

The interface of the plots is very similar to the interface of the scopes (see SCOPE [7.2.6→158]). The main difference lies in how the values to be plotted are selected. By long pressing <**Track**>, a dialog is opened where one can configure the value to be plotted:

<**Enabled**> allows one to disable or enable the display of the plotted value.

With <**Type**>, one can control the values which are displayed under "Symbol". If "All" is selected, all values which can be plotted are displayed under "Symbol". This list can be very long. However, for example, by selecting "Voltage", the list of symbols becomes more manageable as it now contains only voltage-related symbols. "Search" followed by tapping the black text field or the always available <**Search**> allow to directly search for measurands.

<**Name**> defines the symbol of the value to be plotted. This symbol is identical to what is displayed in the measurement menus.

<**Phase/Link**> defines from which phase or calculation the symbol should be used. For example, the trms value of the voltage can be configured to be from the second phase in group 3 (i.e. $2\,G_3$), from the linked value between phases 1 and 2 of group 1 (i.e. $12\,G_1$), the sum of group 2 (i.e. $\Sigma\,G_2$), etc. When the number *before* the "G" is a single digit, it defines the phase of the group. When the number has two digits, it defines the linkage between phases *inside* the group. The Σ-symbol stands for the sum values of a group.
The number *after* the "G" defines the number of the group.

<**Pin, Connector, ..**> If a measurand is group independent the, like PSI values, the respective pin or connector number can be selected.

When dual-processing mode is active, the <**Bandwidth**> is used to select the narrow or wide bandwidth value.

For some symbols which represent lists, it is necessary to specify an <**Index**>. For harmonics (such as BUAM and BIAM), the index is used to select which bin should be plotted.

BUAM [9.10.123→283]
BIAM [9.10.105→274]

7.2.9 Harmonics

The two spectra can be used to visualize the results of a harmonic analysis. While the source information is identical to what is presented in the tables located in the [**VOLTAGE**] or [**CURRENT**] menu, the graphical representation provides users a useful overview of the harmonic analysis.

By long pressing <**Track**>, a dialog is opened where one can configure the value to be plotted. The usage is similar to the corresponding dialog of the Scopes. See SCOPE [7.2.6→158]).

<**Zoom**> changes the number of displayed bars.

When <**Odd Only**> is activated (only possible if no plot shows interharmonics), only the odd-numbered harmonics are displayed.

<**Split**> enables/disables the split view of multiple tracks.

<**Arrange**> Fits either the selected track or all tracks.

<**Range**> shows and sets the displayed range of the y-axis of selected track.

<**Cursor**> shows/hides the cursor. The cursor can be used to examine the amplitude of a specific bin. It can be moved using the left/right arrow keys and directly by tap, hold and drag.

<**Display**> allows to maximize the plot area.

7.2.10 Vector

The vector diagram (sometimes called a Fresnel diagram) is used to get an overview of the amplitudes and phases of a group. This is useful to analyze for effects like non-symmetry.

Navigation

The rotary knob and the arrow keys have no effect in this menu.

Setup

<**Auto Fit**> activates or deactivates the automatic fitting function. If this function is active, all displayed vectors are continuously fitted. If inactive, the vectors for voltage and current can be scaled for each track individually.
 <**Arrange**> allows to manually fit all vectors, or the ones of the active track, if the setting "Auto Fit" is inactive.
 <**Track**> selects one of the three tracks. For each track followings settings are available:
 <**Group**> defines the group which should be displayed in this track. All phases of of the group are displayed. <**Bin**> selects which bin to display. <**Scale U**> and <**Scale I**> change the length of the displayed vectors, if the setting "Auto Fit" is inactive.

7.2.11 CUSTOM menu

The custom menu allows to build up user defined menus with a very high degree of freedom and enables the user to create custom measurement variables (see Scripts [7.2.11→169] for the latter). Any kind of measuring values, accompanied by images, graphs and texts, can be freely arranged in a flexible layout. This allows the creation of complex measuring views to perfectly serve the tasks at hand.

The following sections provide a quick introduction of the features.

Basic Concept

The custom menu uses a grid oriented approach to allow the free arrangement of content like text, graphs and measurands. The number of grid rows and columns determine the number of available cells and can be set to fit the needs. Content can be placed in any cell of the grid.

In addition, any rectangular group of cells can be merged to bigger cells to provide more space for graphs, important measurands or measurand lists. In the following chapters the term 'cell' refers to both, *single grid cells* and *merged cells*.

Cells can be freely resized and relocated within the grids resolution. In cases where more complexity is needed a higher number of rows and columns provides a higher degree of freedom and allows to create complex menus with many differently sized cells.

A background image can be added and any content will be shown on top of it. This allows for example to arrange contents like measurands or graphs directly on circuit diagrams.

Navigation and basic usage

[**CUSTOM**] enters the custom menu.

The custom menu has different edit level which are specialized for specific tasks such as modifying the layout, adding new content, and viewing. Navigation between these level is done as shown in Figure 7.15 [→165]. Following lines assume that the top softkey <**Mode**> is set to "Edit". If <**Mode**> is set to "View" the menu can be used without the risk of accidental modifications.

In this view, all measurands can be copied as a list using <**Copy Measurands**>. This list can be inserted in the measurand choose dialog of the logger. This is very convenient if a complete custom menu shall be logged.

It is not possible to use the Custom menus to the full extent with keys only. If the touch sensor is not applicable, for example if it was intentionally deactivated, it is always possible to use a mouse. As an alternative, it is also posible to create the menus on a PC using the Remote software and copy it to the device afterwards, if needed.

Within the "Edit" mode, three edit level are available.

- The **Menu editor** (See top of Figure 7.15 [→165]). Here the grid is hidden. The softkeys provide functions affecting the complete menu like setting a background image, loading, saving, and reset functions.

  From this level two other level for layout and content editing can be reached by short or long tap on the grid, in the center of the cursor keys or on the wheel.

- A short tap on the grid activates the **Content Editor** (See center of Figure 7.15 [→165]). Here the grid with all cells along with the content is shown. Returning to the menu editor (top of Figure 7.15 [→165]) is possible by short tapping on the currently selected cell (white dashed border), by <**Exit Content Editor**> or automatically following a timeout of approx. 30 seconds after the last action.

  Tapping a previously unselected (no white dashed border) *empty* cell, will select it and the softkeys on the right allow to add various types of content (See next section for a list of content types).

  If the tapped cell already contains any content the softkeys will display content specific options like altering the displayed text or changing colors.

- A long tap on the grid (in menu editor or content editor) activates the **Layout Editor** which allows to modify the layout or content (See bottom of Figure 7.15 [→165]). On this level the selected cell is highlighted and the remaining menu is shaded. By tapping in the shaded area, any cell can be highlighted to work with. In this level the softkeys provide the *content specific* functions cut, copy, remove, and paste and *layout specific* functions like splitting large cells into single cells. A timeout after the last action returns to the menu editor (top of Figure 7.15 [→165]). Using <**Exit Layout Editor**> will show the content editor (center of Figure 7.15 [→165]).

  The copied content can, if one or more measurands are contained, be pasted for logging in the measurand choose dialog of the logger.

  The currently highlighted cell can be resized or moved by dragging its corners or its body, respectively. This is always possible, whether the cell contains content or not. Any content would be resized to fit the new size of the cell. During the re-size or move action, the red border snaps to the grid and indicates the final position and size of the currently modified cell. Please note, that the final geometry of the cell must not intersect with existing cells or contents after any move or re-size action. In such case, the cell will snap back into the previous position and size.

On this level it is also possible to work with frames. <**Modify**> switches between container/content editing and frame editing. See next section for more instructions about frames.

Figure 7.15: The different edit level of the custom menu.

Setting and unsetting of a script

The Mode "Edit" allows to set or remove (unset) a script to this custom menü.

A set script becomes a part of this menu and will be loaded and saved together with it. If this menu is **in use and visible**, regular checks ensure that the currently excuted script matches the set script. By use of this feature it is easy to make sure that a custom menu is not accidentally used in conjuction with a mismatching script intended to be used with another script. This ceck can be disabled until a new script is being set or until the device has been switched off and on. This is useful in case the script needs changes. Otherwise the check would alert the user about a mismatching script due to the changes. The appearing dialog allows to disable the checks.

In case <**Un- / Set Script**> is being tapped and no script has been set, yet, the currently running

script or a script from a file can be associated with the menu. This script will be copied into the menu and becomes a part of it. In case the script was read from a file one can choose to set this script as executed script.

In case <**Un-** / **Set Script**> is being tapped with an already associated script, it can be displayed, removed, or, if it is not the currently executing script, set to be executed.

If the executing script is being set from within a custom menu, it may happen that the currently visible and editable script in the script editor differs from this script. The skript editor notifies about this case and the script in the editor can still be saved. Hereby, accidental loss of script code can be avoided.

Working with frames
Frames are a great tool to enhance the visibility of important content and for grouping together related content for clarity and structure. They can be moved around freely, changed in size (just like the containers), and even be stacked. Further classification can be achieved by colorizing the frames individually.

Frames can be added, edited, and removed inside the **Layout Editor**. To switch from editing containers to editing frames, toggle <**Modify**>. In the "Frames" mode, the existing frames can be selected, moved, and deleted. Containers can be selected to add a frame of that size at the containers position.

To create a frame around a group of containers, just click on any container (not necessarily belonging to the target area). Add a frame by pressing <**Add**>. A white frame appears on the container. Now, the frame can be freely moved and resized to surround the group of containers. The color of the frame can be set using <**Color**>.

By default, new frames will be drawn as full rectangle on the outer line of the perimetric cells. <**Type**> allows to draw only a single line of the rectangle or a central horizontal or vertical line. <**Location**> sets were the bounds of the rectagle or the line are located. This can be the outer border (the default), the center, or the inner border of the perimetric cells.

These settings allow very flexible layouts with boxes and lines in any location.

<**Remove**> removes the currently selected frame and <**Clear**> removes all frames after confirmation.

Types of content and background image
Contents like measurands, images or graphs can be placed in empty cells.

The following list explains all available types of content:

- **Measurand**
  Displays a single measurand. If placed and selected following settings are available:

  - <**Text**>: Allows to edit the displayed text and format. The tags [symbol], [value], [request] and [channel] will be replaced by the specific information of the currently chosen measurand. Any additional plain text will be displayed as is and simple HTML tags will be interpreted. E.g. '<br>' results in a line break.

  - <**Color**>: Allows to choose the color of the displayed text.

  - <**Direction**>: Horizontal or Vertical (from the bottom up).

  - <**Measurand**> (symbol with channel): Allows to choose the measurand to be displayed.

- **Measurand List**
  Displays a scrollable list of measurands. If placed and selected following settings are available:

  - <**Choose Measurands**>: Lets the user choose multiple measurands to be displayed.

  - <**Font Size**>: Allows to optimize the font size of the measurand list.

- – <**Color**>: Allows to choose the color of the displayed text. The color will be applied to the values and the headers.

- **Graph**
  Displays one of the graphs available in the graph menu. Please note that this graph reflects the graph settings configured in the respective Graph menu. If placed and selected following settings are available:

  - – <**Graph Type**>: Chooses from the graphs available in the Graph menu.

  - – <**Setup**>: Redirects to the Graph menu page of the currently displayed graph type. Any settings made here will directly affect the graphs in the custom menus. To return to the custom menus press the [**CUSTOM**] key.

- **Text**
  Display free text. If placed and selected following settings are available:

  - – <**Text**>: Allows to edit the displayed text. Some HTML tags like <br> will be interpreted.

  - – <**Color**>: Allows to choose the color of the displayed text.

  - – <**Direction**>: Horizontal or Vertical (from the bottom up).

- **Image**
  Displays an image. If placed and selected following settings are available:

  - – <**Choose Image**>: Allows to place an image loaded from any device. The format of the image should be PNG and the individual file size may not exceed 1024 kB (1 MB).

  - – <**Keep Ratio**>: If disabled the image will be re-sized to fill the cell.

- <**Add More ...**> ⇒ **Env Var Watcher**
  This content is part of the option L6-OPT-SMV. Shows the red/green encoded value of any environment variable. Values between the chosen red/green value pair will be faded between red and green. If placed and selected following settings are available:

  - – <**Env Var Index**>: The environment variable index (0 to 9).

  - – <**Red Value**>: The value assigned to red.

  - – <**Green Value**>: The value assigned to green.

  - – <**Frame Color**>: Sets the outer frame color.

As environment variables can be set by a script ENVIRONMENT VARIABLES [6.15.3→127], it is possible to create a measurand surveillance system.

- <**Add More ...**> ⇒ **Env Var Setter**
  This content is part of the option L6-OPT-SMV. Shows the value of an evironment variable and allows, if not set to "Read Only", to modify it. It can be displayed using a vertical or horizontal layout. The Layout changes accordingly if resized. If placed and selected following settings are available:

  - – <**Env Var Index**>: The environment variable index (0 to 9).

  - – <**Number Format**>: Setup of the number format. It can be chosen between integer "Int" and floating point "Float". The respective script functions to access the values are "env(*envvarno*)" and "setEnv(*envvarno, floatvalue*)", or "envInt(*envvarno*)" and "setEnvInt(*envvarno, intvalue*)".

  - – <**Value**>: Sets the value of the environment variable. The value changes by +/- 1.0 if tapped short and can be entered manually after long tap on the softkey.

  - – <**Read Only / Read & Write**> Switches between the view mode (Value can not be changed, arrow buttons not visible) and and the settable mode (Value can be changed, arrow buttons visible).

  - – <**Color**>: Sets the color.

Environment variables can be used in scripts. Hence, it is possible to control a script by the custom menu (e.g. number of turns in a core loss application). The script results of the current cycle will be based on the changed value of the environment variable.

- **<Add More ...> ⇒ Env Var Switch**
  This content is part of the option L6-OPT-SMV. It provides one or more buttons designed for script control. The switching behavior can be configured to facilitate the given control task. The script function "switch(*envvarno*, *buttonno*)" allows to conveniently query the state of the buttons using the associated environment variable and button number. The script function "setSwitch(*envvarno*, *buttonno*)" allows to programmatically manipulate the switch state.

  Following settings are available:

  - **<Env Var Index>**: Sets the environment variable (stores the switch state, 0 bis 128). This variable will be used by this switch and should not be used in another context by the script or in another custom menu.

  - **<Button Count>**: Number of buttons in this switch (max. 7).

  - **<Orientaton>**: Orientation of the buttons (vertical or horizontal).

  - **<Switching Mode>**: Selects the desired switching mode. Following modes are available:

    * **Switch**: Every button can be independently activated or deactivated. Multiple active switches are possible.
      Usecase: Setting a resistor combination and control of respective GPIO outputs by the script.

    * **Tap**: Every button can be independently pressed (button-like). The associated environment variable will encode this button as active for a short time. The script function "switch(envvarno, buttonno)" will return "true" for a short time (at least for one cycle). If it has to be ensured that the key is active for no longer than one cycle, the switch can be deactivated with the script function "setswitch(envvarno, 0)".
      Use case: Start and stop of energy measurement by the script.

    * **Selector**: Only one button can be active at a time in this mode. The other buttons will be deactivated (released) if one button is activated (pressed).
      Use case: Selecting a calculation paths in the script.

  - **<Edit Button>**: Selects the button the next softkey works with. This selection will also be updates if a button is being pressed.

  - **<Change Text>**: Sets the text for the button selected in the previous softkey. The subscripted button number will be appended to this text.

  - **<Change Color>**: Sets the color of the button.

In addition to this grid oriented content it is possible to set a background image to be displayed format filling behind the content. This allows for example to display measuring values or graphs on top of circuit diagrams. A background image can be set or removed using **<Un- / Set Background>** in the view mode. Information about the best format filling pixel size is given in the title bar of the appearing file dialog. Other sizes will be stretched or compressed to fill the background. The image format should be PNG and the file size may not exceed 1024 kB (1 MB).

Clear, Load and Save

In the view mode **<Clear>** allows to remove all content including the background image, reset the layout, and set the default grid resolution of 8 rows by 16 columns. This action can not be undone and the current menu should be saved if necessary.

<Save Menu> allows to store the currently visible custom menu in any location. The file extension will be '.zmenu'. **<Load Menu>** loads a custom menu file with the extension '.zmenu' from any location. A loaded menu will replace the currently displayed menu. If the currently displayed menu shall be used in the future it must be saved before loading any other menu using **<Save Menu>**.

Changes to any loaded or just created menu will be saved automatically in a working copy when major alterations like layout changes or content placing happen or on shut down of the device. The stored working copy with the latest changes will be loaded automatically on start up of the device. These operations will *not* alter the original '.zmenu' file. To update the '.zmenu' file as well, the menu needs to be saved explicitly as described above. This way accidental changes to '.zmenu' files are not possible and latest changes to the menu are saved as well. To undo changes to a loaded menu just reload the menu from the '.zmenu' file.

Grid size

The softkey <**Settings**> in the view mode shows a dialog which allows to set the grid dimensions. This also determines the minimal cell size. The number of rows and columns can be set by using the wheels on the left hand side or the respective softkeys. On the right hand side the resulting new grid and the effect of the new settings on the current layout is shown. Cells not fitting into the new dimensions (colored orange) will be removed including any content after confirming the dialog.

In addition, checking <**Reset Cells**> and/or <**Remove Background**> triggers the respective actions on confirmation of the dialog. Setting 8 rows, 16 columns and checking both softkeys is equivalent to use <**Clear**> (see previous section).

Scripts

In the tab "Script" of the custom menu, one can find the script editor. This editor allows to create a set of variables that can be plot, logged and displayed in the custom menu.

To input a new script, <**Mode**> must be set to "Edit". A text field allows you to enter the script code. The rules by which the script is built up are explained in chapter SCRIPTEDITOR [6.15→125]. After entering a script, one can press <**Install Script**> to install your script on the device and continuously run it with each measuring cycle. It will also persist during power-off periods of the device. To view the computed measuring values, change to the "View" mode using <**Mode**>. The results are displayed below each other, with the code of the installed script below them.

The softkey <**Script Enabled/Disabled**> allows the deactivation of the running script ("Disabled"). Setting it to "Enabled" reactivates the script. The Script will start over. This is useful in case a script controls the GPIO outputs. For temporary measurments at the GPIO outputs or at a connected circuit, interferences by the script can be eliminated by setting this softkey to "Disabled". This is different from a break, because activation of the script by "Enabled" is equivalent to a new installation of the skript.

During the execution of the script, errors can happpen, for example if the script takes too long to compute its values or if an integer-division by zero happens. The execution of the script will be canceled and an error message will be displayed together with the location of the error within the script. You can switch to the "Edit" mode to change the script and try to run it again. The script then executes again from scratch (for example, assignments using '?=' will be carried out again, see ASSIGNMENTS [6.15.4→128]).

To access device variables such as UTRMS [9.10.217→319], <**Insert Measurand**> can be used. This opens a dialog to choose the desired measurand. Alternatively you can input the measurand name manually as explained at ACCESS TO MEASURING VALUES [6.15.5→134].

If you have connected an external keyboard by using the USB-port of the LMG, the internal touch keyboard can be hidden to maximize the area of the text editor using <**Keyboard**>.

The Softkey <**Code Snippets**> allows to insert and manage code snippets (see CODE SNIPPETS [7.2.11→170]).

Code Snippets

Code snippets are short code sequences ranging from a simple function call to multiple lines of code.

The softkey <**Code Snippets**> in the script editor opens the "Code Snippet"-dialog. This dialog shows the names and information about all available functions and flow control structures (e.g. for loop, if-else conditionals). The currently selected code snippet will be inserted at the current cursor position if the dialog is left by the "Insert" button.

The various categories can be accessed using the tabs.

A special role plays the category "Custom Snippets". Here, individual code snippets of interest can be stored for reuse. It is possible to enter or insert code, name the snippets, and add additional information. All this input can be edited, later. In addition to adding and editing, the code snippets can also be deleted.

Some code snippets contain fields for needed parameters. After inserting such a code snippet, these fields will be highlighted and can be filled-in one after another.

Displaying, Logging and Plotting of script variables

The variables of the currently running script can be accessed just as regular measurands in the respective dialogs for choosing measurands. If the left wheel ("Type") is set to "Custom". The right wheel ("Name") will show the currently available script variables.

> *A modification of the running script will likely affect the mapping of the variable names. Especially, if variables are added or removed. In this case the variables have to be chosen again.*

7.2.12 PSI (I/O) Menu

The [**I/O**] switches to the measure menu of the Process Signal Interface (PSI). Here, measurands and settings from all parts of the PSI are displayed within sub-menus. The technical specifications of the PSI can be found at PROCESSING SIGNAL INTERFACE L6-OPT-PSI [4.12→78].

> *Two PSI tabs will be visible in case of two installed PSI cards. Both tabs contain the same sub-tabs and each tab controls exclusively the respective card.*

> *In case no PSI is installed 'PSI Not Installed - Preview Only' will overlay the PSI menus.*

Following softkeys are available in the measuring menu:

- <**Tab Switch**> switches through the different parts of the PSI.

- <**Go To PSI Settings**> switches to the respective setup menu of the PSI. Here, all PSI configurations can be done. [**I/O**] returns to the PSI measuring menu.

> *The grayish PSI setup menu is almost identical to the blueish measuring menu. While the measuring menu allows a safe view of the PSI values without the possibility to change settings, the setup menu allows access to all parameters of the PSI. The setup menu can also be entered using [**INSTR.**]. The softkeys mentioned below in the settings context can only be found in the setup menu. The measured values are displayed in both menus.*

The following sections explain the different parts of the PSI.

Analog In

Displays the measurands and settings of the analog input.

The analog input provides up to 8 linearly scaled input voltages ranging from -10 V to +10 V. The linear scaling is defined by 0 V and +10 V reference points.

The right side displays the matching connector on the PSI back plane (see also Figure 4.11 [→79]).

Available information, measurands and settings:

- **Input** The pin descriptor.

- **0 V Ref.** 0 V reference point voltage (An input voltage of 0 V will result in this value). Settable using <**Set 0 V Ref.**>.

- **+10 V Ref.** +10 V reference point voltage (An input voltage of +10 V will result in this value). Settable using <**Set +10 V Ref.**>.

- **Scaled U** Value of the measured input voltage after linear scaling using the reference points above.

Analog Out

Displays the measurands and settings of the analog output.

The analog output provides up to 32 from measurands derived and linearly scaled output voltages ranging from -10 V to +10 V. The linear scaling is defined by 0 V and +10 V reference points.

The right side displays the relevant pins on both connectors (see also Figure 4.12 [→80]).

Available information, measurands and settings:

- **Output** The pin descriptor

- **Source** The measurand to derive the output voltage from. Settable using <**Choose Source**>.

- **0 V Ref.** 0 V reference value (This measurand outputs 0 V). Settable using <**Set 0 V Ref.**>

- **+10 V Ref.** +10 V reference value (This measurand outputs +10 V). Settable using <**Set +10 V Ref.**>

- **Scaled U** The current output voltage

Switch In

This sub menu shows the measurands and settings of the digital switching input of the PSI.

The switching input detects the current state of up to 8 digital signals. The right side of the screen displays the relevant pins on the connector (see also Figure 4.13 [→81]).

Available information, measurands and settings:

- **Input** The pin descriptor

- **Pulse Count** counts continuously the pulses received on the respective pin. This counters can be reset individually using <**Reset Counter**> or simultaneously for all pins using <**Reset All Counters**>.

- **Invert** Inverts the state. Settable using <**Invert**>.

- **State** The current state. "1", if a voltage is being applied to the corresponding input pin, else "0".

Switch Out

Displays measurands and settings of the PSI switching output.

The switching output controls up to 8 switches directly or in dependence of defined conditions. The right side of the screen shows the relevant pins of the connector (see also Figure 4.13 [→81]).

In case a switching output is in control of the script, the "Source" display indicates "Script Ctrl.". The configure buttons for a script controlled pin are disabled.

Available information, measurands, and settings:

- **Output** The pin descriptor

- **Source** The measurand on which the condition set in **Condition** and the threshold set in **threshold** will be applied to evaluate the state of the switch. Settable using <**Choose Source**>

- **Condition** The condition applied together with the threshold on the measurand to evaluate the state of the switch. The settings 'On' and 'Off' work independently of the measurand and threshold. Settable using <**Set Condition**>.

- **Threshold** The threshold applied together with the condition on the measurand. Settable using <**Set Threshold**>.

- **Invert** Inverts the state. Settable using <**Invert**>.

- **State** The current state. A closed switch is indicated by "1".

Fast In (Single Motor and Dual Motor)

This sub menu contains the measurands, settings, and operation modes of the fast analog and frequency input connectors.

This inputs can be either used in 'Direct' mode or in one of two motor modes ('Single Motor' or 'Dual Motor'). In 'Direct' mode, all four inputs (2 analog, 2 frequency) can be used independently from one another. In 'Single Motor' mode the type of a single torque and a single speed sensor can be configured individually in order to measure mechanical power. In 'Dual Motor' mode two analog torque sensors and two digital speed sensors can be used to simultaneously measure two mechanical powers.

The mode can be toggled using <**Mode**>.

The valid pins/connectors are highlighted on the right hand side of the screen (see also Figure 4.14 [→82] and FAST ANALOG INPUTS [4.12.1→79]).

Following information, measurands, and settings are available in 'Direct' mode:

(The displayed softkeys refer to the highlighted row in the upper or lower table. The highlight can be moved by the wheel and the up and down arrow keys.)

Upper table (<**Settings**> set to 'Analog')

- **Analog** Number of the fast analog input.

- **0 V Ref.** 0 V reference point voltage (An input voltage of 0 V will result in this value). Settable using <**Set 0 V Ref.**>.

- **+10 V Ref.** +10 V reference point voltage (An input voltage of +10 V will result in this value). Settable using <**Set +10 V Ref.**>.

- **Scaled U** Value of the measured input voltage after linear scaling using the reference points above.

Lower table (<**Settings**> set to 'Frequency')

- **Freq.** The number of the speed/torque/frequency input connector.

- **0 Hz Ref.** 0 Hz reference point frequency (An input frequency of 0 Hz will result in this value). Settable using <**0 kHz Ref.**>.

- **10 kHz Ref.** 10 kHz reference point frequency (An input frequency of 10 kHz will result in this value). Settable using <**10 kHz Ref.**>.

- **Filter** Setting of the low-pass filter. Settable using <**Filter**>.

- **Invert** Inverts the direction. Settable using <**Invert**>.

- **Signal** Signal type (TTL, HTL, RS422). Settable using <**Signal Type**>.

- **f** The measured and scaled frequency.

- **Dir** The Direction. Displays "+1" if A before B and "-1" if B before A.

Following information, measurands, and settings are available in 'Motor' mode:

In the PSI settings menu: <**Sensor Trq - Speed**> configures the type of torque and speed sensor and thereby the used PSI inputs. 'Ana' means the sensor provides an analog signal and 'Dig' the sensor provides a digital (frequency) signal.

Digital torque settings (<**Sensor Trq - Speed**> is set to '**Dig** - Ana' or '**Dig** - Dig' and <**Settings**> is set to '**Torque**')

- **Nominal Trq.** The nominal torque of the sensor. Settable using <**Nominal Trq**>.

- **0 Nm Freq.** The frequency for 0 torque. Settable using <**0 Nm Freq.**>.

- **Freq. Span** The frequency at nominal torque. Settable using <**Freq. Span**>.

- **Signal Type** Signal type (TTL, HTL, RS422). Settable using <**More : Signal Type**>.

- **Filter** Setting of the low-pass filter. Settable using <**More : Filter**>.

- **Torque** The resulting torque.

- **Mechanical Power (page bottom)** The resulting mechanical power.

Digital speed settings (<**Sensor Trq - Speed**> is set to 'Ana - **Dig**' or 'Dig - **Dig**' and <**Settings**> is set to '**Speed**')

- **Pulses p. Rev.** Number of Pulses per Revolution.

- **Filter** Setting of the low-pass filter. Can be set using <**Filter**>.

- **Signal Type** Signal type (TTL, HTL, RS422). Settable using <**Signal Type**>.

- **Speed** The resulting speed.

- **Direction** The Direction. Displays "+1" if A before B and "-1" if B before A.

- I_{null} The I_{null} value of the $I_d - I_q$ transformation.

- I_d The I_d value of the $I_d - I_q$ transformation.

- I_q The I_q value of the $I_d - I_q$ transformation.

- **Id/Iq Settings** This softkey opens a dialog for the configuration of the number of pol pairs, theta null, and the Id/Iq group.

Analog torque settings (<**Sensor Trq - Speed**> is set to '**Ana** - Ana' or '**Ana** - Dig' and <**Settings**> is set to '**Torque**')

- **0 V Ref.** 0 V at the fast analog input 1 will result in this torque value. Settable using <**0 V Ref.**>.

- **+10 V Ref.** +10 V at the fast analog input 1 will result in this torque value. Settable using <**+10 V Ref.**>.

- **Torque** The resulting torque.

- **Mechanical Power** The resulting mechanical power taking the measured speed (lower table) into account.

Analog speed settings (<**Sensor Trq - Speed**> is set to 'Ana - **Ana**' or 'Dig - **Ana**' and <**Settings**> is set to '**Speed**')

- **0 V Ref.** 0 V at the fast analog input 2 will result in this speed value. Settable using <**0 V Ref.**>.

- **+10 V Ref.** +10 V at the fast analog input 2 will result in this speed value. Settable using <**+10 V Ref.**>.

- **Speed** The resulting speed.

The 'Dual Motor' mode provides following information, measurands and settings:

<**Select**> chooses the motor whose configuration can be set using the softkeys below. <**Settings**> toggles between the softkeys for the speed sensor and the torque sensor.

Following settings are available for the analog torque sensors:

- **0 V Ref.** 0 V at the fast analog input will result in this torque value. Settable using <**0 V Ref.**>.

- **+10 V Ref.** +10 V at the fast analog input will result in this torque value. Settable using <**+10 V Ref.**>.

- **Torque** The resulting torque.

Following settings are available for the digital speed sensors:

- **Pulses p. Rev.** Number of Pulses per Revolution.

- **Filter** Setting of the low-pass filter. Can be set using <**Filter**>.

- **Signal Type** Signal type (TTL, HTL, RS422). Settable using <**Signal Type**>.

- **Direction** The Direction. Displays "+1" if A before B and "-1" if B before A.

- **Speed** The resulting speed.

- **Z-Pos.** '1' if in zero position.

Reference point calculation examples for torque and speed sensors

Digital Torque settings for torque measuring shaft with frequency output and symmetrical range of $\pm M_n$

Use of frequency input A1 (and $\overline{A1}$ for RS422).
Set signal type (TTL 5 V, HTL 12...30 V, RS422).
Set filter if necessary.

Needed information:
Nominal Torque M_n (in Nm)
Center frequency f_0 bei $M = 0\ Nm$ (in Hz)
Nominal frequency deviation df at $M = M_n$ (in Hz)

This values can be entered directly.

Digital Speed settings for incremental encoder having two outputs with a 90° shift.

Use of frequency inputs A2 und B2 (and $\overline{A2}$ and $\overline{B2}$ for RS422).
Set signal type (TTL 5 V, HTL 12...30 V, RS422).
Set filter if necessary.

Needed information:
Number of pulses per revolution I_{mp} (no unit)

This value can be entered directly.

The displayed speed has the unit min^{-1}.

Analog torque settings

Example: Torque sensor with $\pm 1\ kNm$ range and voltage output $\pm 10\ V$

A voltage of $0\ V$ corresponds to $0\ Nm$, hence "0 V Ref." needs to be set to $0\ Nm$

A voltage of $+10\ V$ corresponds to $+1000\ Nm$, hence "+10 V Ref." needs to be set to $1\ kNm$

Analog speed settings

Example: Tachogenerator with 5mV per min^{-1}

A voltage of $0\ V$ corresponds to $0\ min^{-1}$, hence "0 V Ref." needs to be set to $0\ min^{-1}$

A voltage of $+10\ V$ corresponds to $2000\ min^{-1}$, hence "+10V Ref." needs to be set to $2k\ min^{-1}$

Filter And Sync Group

The group selected by <**Sync Group**> is the reference group for all PSI inputs. That is, the measurands of the PSI inputs are being determined for the same interval as the measurands of that group.

If the channel grouping changes, group 1 will be automatically set as reference group.

The PSI outputs are updated synchronously to the currently active cycle (see CYCLMOD [9.10.355→385]).

<**Filter**> configures how the signals of the fast analog inputs (BNC connectors) are being filtered.

- **Group** aims to match the filter settings of the selected group. In general, the upper limit of the PSI low-pass filter is 15.0 kHz. The group filter settings can be set using <**Group Settings**> (see GROUP MENU [7.3.12→182]). [**INSTR.**] navigates back to the PSI menu.

- **PSI** allows a group independent setting of the PSI filters. The PSI filters are configured analogously to the group narrowband filters as described in GROUP MENU [7.3.12→182].

7.2.13 Misc. menu (measurement)

[**MISC.**] allows one to navigate to this menu where information about the installed software is provided. <**Licenses**> opens the license information. <**Recent Messages**> shows system messages received since system start up. <**GUI Lock**> Activates ("Active") or deactivates ("InActive") the lock of the graphical user interface. If active, the most important device settings can no longer be changed by the GUI. If the GUI lock is active, a closed lock is being displayed in the status bar.

7.3 Setup menus

In the setup menus, one has access to all settings which define the measurement environment. The most important things to configure are grouping, the synchronization parameters and the ranges. Because the most important menus are used so frequently, they each have their own button ([**INSTR.**], [**GROUP**] and [**CHANNEL**]).

The menus are described in INSTR. MENU [7.3.1→176], GROUP MENU [7.3.12→182] and CHANNEL MENU [7.3.14→186]. Background information concerning these settings can be found in BASIC CONCEPTS [6→101].

7.3.1 INSTR. menu

The instrument menu (accessed by clicking [**INSTR.**]) contains the global settings which apply for the whole instrument.

7.3.2 Measurement tab

CYCLMOD
[9.10.355→385]
CYCL [9.10.356→386]

With <**Cycle**>, one can configure the main time interval for updating the data and thus for the average measurement duration. Long pressing <**Cycle**> opens a dialog where one can select the cycle mode. If the cycle mode is set to "Fixed Interval", it is possible to configure a cycle time . In addition to the two command references, further information is located in PARALLEL CALCULATIONS [6.7→113]. Left/right pressing of <**Cycle**> allows one to change the cycle time if the cycle mode is set to fixed interval.
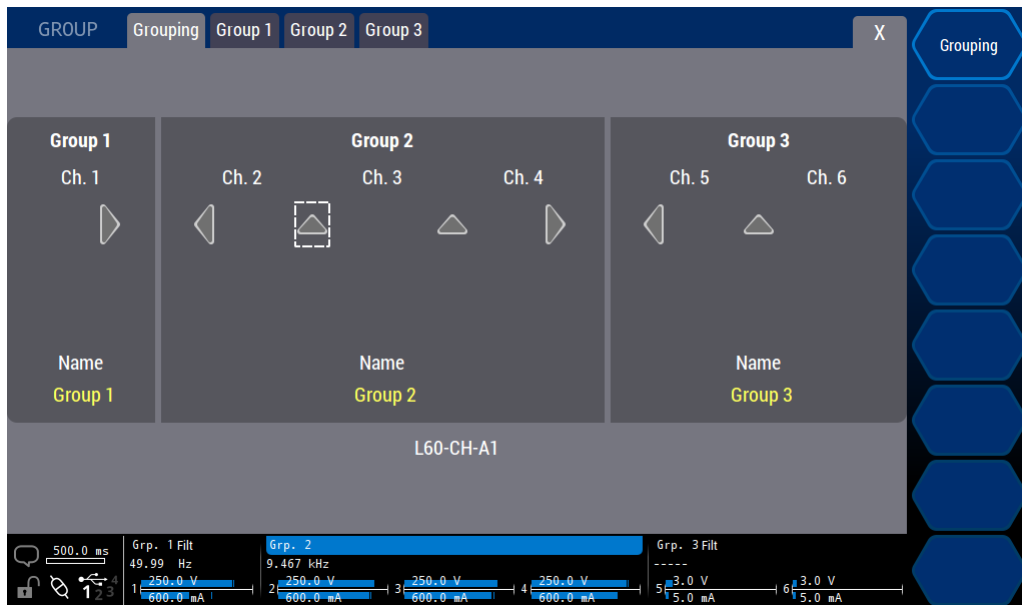

Figure 7.16: Menu to configure global measurement settings.

AVER [9.10.236→326]

<**Average**> allows one to configure the average which is the number of measuring cycles over which the sliding average is calculated. This applies to (C-VALUES [→113]) only and does not affect other values such as harmonics, flicker, and samples. Harmonics and flicker values have unique averaging algorithms, which are implemented in the respective analysis software.

ZSUP [9.10.388→400]

The instrument can be configured to display small values, which are close to zero, as zero in order to remove unwanted noise (e.g if no signal is applied). This function can be enabled or disabled by using <**Zero Rejection**>.

GROUP
[9.10.286→351]

<**Grouping**> allows one to configure which channel(s) belongs to which group. The configuration is done in a dialog which can also be reached by touching the arrow shaped buttons.
In this dialog, one can use the arrow keys or the rotary knob to select one of the split (arrow up) or grow (arrow left/right) symbols. To trigger the symbol, one can press the selection key, press the rotary knob or simply touch the arrow buttons directly. These changes only take place after committing by [**ENTER**].

7.3.3 Int. Zero Adjust

This tab allows controlling the internal current zero adjust feature of the S and ABC channels from version 3.

Please note that the upper two tables show the dates and states of individual channels while the settings in the table at the bottom refer to groups.

The top table indicates the time point of the last internal zero adjustment within the last 24 hours of the two bandwidths of the channels.

This time is not displayed in case the last adjustment dates back longer than 24h or any other reason for invalidity is given.

The second table shows whether or not the state of the internal adjustment is OK. In case an adjustment is due, the reason for this is shown. If "Auto Adjust" were active, and the respective triggers have not been deactivated, this state would always be OK.

The third table allows configuring the internal zero adjustment behavior for each group that contains supported channels.

"Auto Adjust" indicates whether or not the automatic zero adjustment is allowed. In some cases, when short measure gaps are not tolerable, automatic zero adjustments can be disabled.

The checkboxes allow further configuring which triggers are allowed in case "Auto Adjust" is active.

> *In case "Auto Adjust" is enabled, changes of current measure range and of some configurations, will always trigger an adjustment.*

> *In case the current state is not OK, enabling "Auto Adjust" will immediately perform a zero adjustment.*

Independent from the settings, <**Adjust Now**> triggers an adjustment of the active current ranges anytime.

7.3.4 General tab

<**Date**> is used to set the system date and time. This time is used for example for the time stamp of a measurement or for the date of a stored file.

SYSDATE
[9.10.401→406]

<**Date Source**> sets the source for the internal clock. The setting "Internal" allows to set time and date manually and the current time is put out at the "Time_I/O" pin of the synchronization interface. The setting "External" allows to synchronize the Clock by using the "Time_I/O" pin of the synchronization interface.

SYSDATESOURCE
[9.10.402→406]

<**Time Zone**> opens a dialog which allows the setup of the appropriate time zone.

TZONE
[9.10.456→429]

When connecting an external keyboard, one can use <**Keyboard**> to select the keyboard layout.

<**Help Lang.**> sets the language of the manual being displayed after using [**HELP**].

<**Format USB**> allows to format any USB device to be usable with a PC. For example, this can be useful with a boot stick which can not be easily formatted with the full capacity using a PC.

Tapping left or right chooses the USB device in case more than one is connected. The device name and its capacity is shown. In case the indicated nName and capacity could match more than one connected device formatting will not be possible unless all ambiguous devices were removed.

A long press, followed by confirmation of the informational dialog, will format the USB device. The progress dialog may be hidden and a brief message will show up when the formatting has finished. Only then may the USB device be removed.

Softkey "Change Display and LED Brightness Settings"

This entry opens a dialog for setting up LED and display brightness.

Reducing the regular display brightness may be more comfortable for working in low ambient light conditions. In addition, the timed dimming and turning off (if supported by the device) preserves the display light and improves the longevity.

DISB [9.10.28→244] The display brightness can be set with <**Display Brightness**>.

<**Dimmed Brightness**> sets the reduced brightness if dimmed.

<**Display Dim Timeout**> is the time duration without using the Front until the display will be dimmed.

<**Display Off Timeout**> is the time duration without using the Front until the display will be turned off (only available if supported by the device). The display can be turned off manually with the DISOFF [9.10.29→245] command.

<**LED Brightness**> allows to set the regular brightness of the key LEDs.

<**Dimmed Brightness**> set the dimmed brightness of the key LEDs. The key LEDs will be dimmed to this level after approx. 2 minutes.

Best results can be expected with approx. 80 and 10 % for "LED Brightness" and "Dimmed Brightness", respectively. Depending on the ambient light, other settings may be more useful. Especially in dark environments lower settings can be more comfortable.

7.3.5 Interface tab

This tab allows to set the active interface and to setup the interfaces.

Most settings can only be changed by the internal GUI.

The currently active interface can be used to connect to the LMG and can be set using <**Active Interface**>.

To configure an interface its settings can be shown using <**Show Settings for**>.

Following interfaces are available for remote controlling the instrument:

- LAN
IPDHCP The first decision to make here (by using <**DHCP**>) is whether the instrument acquires its IP
[9.10.439→422] address and other information from a DHCP server or if these settings must be manually set. To find a DHCP server can take up to 1 min. If no DHCP server is available (e.g. in a point-to-point connection to the PC), the LMG will use an IP address of the address block 169.254.0.0/16 (zeroconf).
IPADDR If DHCP is disabled, one has to manually specify the <**IP Address**> and the <**Netmask**>
[9.10.437→421] which is in most cases 255.255.255.0 in a class C network. Only if one needs access to a wider
IPMASK network or the internet via a gateway would one need to specify the address of the <**Gateway**>.
[9.10.444→423]
IPGATE By using <**Advanced**> additional LAN settings like Broadcast and DNS server can be accessed
[9.10.442→423] and edited.

- RS232

  Beside the <**Baud**> rate, there are two additional parameters to be configured: The hardware <**Flow Control (RTS/CTS))**> can be used to temporarily interrupt the data transfer if, for example, the PC cannot receive the data in real time. This is recommended in order to prevent the loss of data.

  For testing purposes, one can test commands by using a terminal program or by testing both data directions through activating/deactivating character echoing (<**Echo**>). For remote controlling this parameter is usually deactivated.

COMBD
[9.10.433→419]
COMHFC
[9.10.435→420]
COMECHO
[9.10.434→419]

By default, an active connection puts the GUI into "Remote" mode and it is no longer possible to change settings using the front panel without further ado. In case a connected program ceases to work, it is possible to set the LMG back into "Local" operation mode by using <**Go To Local**>.

GTL [9.10.19→240]

7.3.6 NTP tab

In this tab, the NTP time synchronization can be configured.

The NTP time synchronization is available for the devices LMG611, LMG641, LMG671, and LMG671-ATE.

The NTP server, Max Poll, and Min Poll can be set. The default setting of 10 for Max Poll and 6 for Min Poll should typically provide good results while ensuring a non-critical request frequency to a public NTP server.

In addition to the general synchronization status (1/0), the values for NTP Discipline Estimated Error and NTP Discipline Max Error are available for assessing accuracy. Further information can be found in the whitepaper on NTP (see download section at https://www.zes.com).

If NTP is selected as the time source (see General tab) and a valid and reachable NTP server is set, the NTP synchronization will start. It may take a few minutes for the synchronization status to switch to 1 and for accuracy values to be provided.

7.3.7 PSI tab

In this tab the PSI can be configured (requires option L6-OPT-PSI). The meanings of all sub tabs and settings are covered in PSI (I/O) Menu [7.2.12→170].

7.3.8 CAN tab

This tab allows to configure and control the CAN slots that are used for broadcasting measurands over and receiving data from a connected CAN bus (requires option L6-OPT-CAN).

Transmit tab

In this tab, the 128 slots for broadcasting measurands over the CAN bus can be configured.

Up the 64bit can be send in a single CAN message. This equates 2 floating point numbers (most regular measurands) or 1 integer number (many configuration values and few measurands). The values can be set in the first or the second field. The offset in the message will always be 0, in case that only a single value is defined. The user will be informed of any bit length conflict. The bit offset of the second float number will always be 8 byte.

The following settings are available:

- <**CAN Id**> Sets the CAN id of the slot.

CANCONTID
[9.10.416→412]

- <**Toggle EFF**> Whether this slot makes use of the extended frame format. If this bit is not set ("Off"), the upper limit for the CAN id is 2047. If this bit is set ("On") the upper limit for the CAN id is 536870911.

CANCONTIDEFF
[9.10.415→411]
CANACTN
[9.10.414→411]

- <**Choose Measurand 1**> Allows to choose the first measurand for this CAN slot.

- <**Choose Measurand 2**> Allows to choose the second measurand for this CAN slot.

The following controls are available:
(The current selection can be moved using the wheel or the up / down arrow keys)

CANCONT
[9.10.413→411]

- <**Start**/**Stop Selected**> Starts or stops the currently selected slot.

- <**Clear Selected**> Clears the currently selected slot.

- <**More ...**> ⇒ <**Start**/**Stop All**> Starts or stops all slots having a measurand set.

- <**More ...**> ⇒ <**Clear All**> Clears all slots.

- <**More ...**> ⇒ <**Enumerate Ids**> Allows easy enumeration of CAN ids (see last section in this chapter).

Trigger tab

Here, the 128 slots for receiving data over the CAN bus can be configured.
The following settings are available

(<**Control**> = "CAN Trigger Settings 1"):

CANTRIGID
[9.10.428→416]

- <**CAN Id**> Sets the CAN id of the slot.

- <**Toggle EFF**> Whether this slot makes use of the extended frame format. If this bit is not set ("Off") the upper limit for the CAN id is 2047. If this bit is set ("On") the upper limit for the CAN id is 536870911.

CANTRIGIDEFF
[9.10.426→415]
CANTRIGOFFSET
[9.10.430→417]

- <**Offset**> Allows to set the bit offset of the value to be read.

- <**Bit Lenght**> The number of bits of the value to be read. Note: The floats may consist of eighter 32 or 64 bits. Integer can have any number of bits from 1 to 64.

CANTRIGBITLEN
[9.10.423→414]
CANTRIGOPTYPE
[9.10.432→418]

- <**Operand Type**> Sets whether the data shall be interpreted as "integer" or "float".

- <**Enumerate Ids**> Allows easy enumeration of CAN ids (see last section in this chapter).

(<**Control**> = "CAN Trigger Settings 2"):

- <**Condition**> Sets the condition for the comparison of the received with the reference value. Note: The condition "True" triggers always.

CANTRIGCOMP
[9.10.425→415]

- <**Action String**> The short commands, separated by semicolon, to be executed if the condition was met. The received value can be inserted using "%v".

CANTRIGACTNSTR
[9.10.422→414]
CANTRIGOPFLT
[9.10.427→416]
CANTRIGOPINT
[9.10.429→417]
CANTRIGSWITCH
[9.10.431→418]

- <**Reference Value**> The value the received value shall be compared with.

(<**Control**> = "CAN Trigger Control"):

- <**Start**/**Stop Selected**> Starts or stops the currently selected trigger slot.

- <**Clear Selected**> Clears the currently selected trigger slot.

- <**Start**/**Stop All**> Starts or stops all trigger slots.

- <**Clear All**> Clears all trigger slots.

- <**Check Selected**> Does a plausibility check on the selected trigger. Any issues will be communicated using the error queue.

Global Settings / Misc tab

Here are global CAN bus settings and CAN related tools available.

- <**Bit Rate**> Needs to be set according to the bit rate of the connected CAN bus.

- <**One Shot Mode**> Sends messages without waiting for an acknowledgement.

- <**DBC Export**> Export of transmit settings (Transmit tab) in DBC format. It is possible to export active slots only (state "On") or all configured slots.

CANBIT
[9.10.412→410]
CANONESHOT
[9.10.421→413]
CANDBC
[9.10.418→412]

CANDBCALL
[9.10.417→412]

Assisted enumeration of ids

This feature allows to enumerate any number of ids starting at the currently selected slot with an automatically incremented id.

For an example with 17 ids starting with id 42 it is assumed that the currently selected slot (can be moved using the wheel, the arrow keys, or by tapping a setting) is number 10. After tapping <**Enumerate Ids**>, a dialog will ask for the starting id ("Start Id") and the number of ids ("Id Count"). After entering the abovementioned numbers and committing the dialog, the slots from 10 to 26 will be set to consectuive ids from 42 to 58. Assistance is given for the EFF setting.

7.3.9 GPIO and Sync tab

This tab allows to configure the GPIO pins (GPIO, General Purpose Input Output) of the synchronization interface (see SYNCHRONIZATION CONNECTOR [4.3→46]) and shows the state of the various synchronization and control pins.

GPIO pins

The upper part lists the configuration and current state of the GPIO pins (GPIO, General Purpose Input Output). They can be configured either as "Input" or "Output" using <**Direction**>. In case the direction is set to "Output" the state of the respective pin can be set using <**Set State**>. Regardless of the selected direction, the column "State" indicates the current state of the pin.

In case a GPIO pin is in control of the script, the "Set State" display indicates "Script Ctrl.". If the pin can not be controlled by the script, because it is configured as input, the "Set State" display indicates "Script: Check Dir!". If it is ensured that the GPIO pins are connected correctly, the pin control by script can be enabled by setting "Direction" to "Output".

Synchronization and control pins

The lower part shows the states of the various synchronization and control pins.

Usually, the states of the synchronization and control pins are derived from settings in other menus or dialogs. Information about the states and assistance on changing the states is given after tapping the state or the associated softkey.

- **Transient I/O** Indicates if this pin is asserted when a trigger event occurs ("Output") or if the transient can be triggered using this pin ("Input").

- **Energy I/O** Indicates if this pin is asserted when the energy measurement is running ("Output") or if the energy measurement can be controlled using this pin ("Input").

- **Frequency In** This pin works always as input. Groups with synchronization setting "Extern" synchronize on this pin.

- **Cycle I/O** Indicates if this pin is asserted synchronously to the current cycle ("Output") or if the cycle will by synchronized to this pin ("Input").

7.3.10 Options/Key tab

This menu shows the installed options. The options list consists of both, hardware-based and software-based options. It is possible to enable and/or disable the latter by importing a key provided by ZES ZIMMER.

Key export

<**Export Key**> saves a file containing the current option key (extension: .zco, ZES current option key). On request, this file can be sent to ZES ZIMMER via email.

Key import

<**Import Key**> allows to import a provided activation key (extension: .zak, ZES activation key). If the imported key is valid, the software-based options will be set as needed and the device will restart.

7.3.11 Touchscreen tab

The upper section shows the date of the last adjustment of the internal touchscreen. If necessary, the touchscreen can be adjusted with <**Adjust Internal Touch**>.

This adjustment must not be carried out on an external display. External capacitive touch devices usually do not need to be adjusted.

If the result of the adjustment is noch satisfying after several attempts, <**Load Internal Defaults**> followed by another adjustment should solve the problem.

Any connected external usb touch devices show up in the section "External Screen and Touch Device". It may be neccessary to alter the "Touch Area" setting in case the format of the touchscreen does not match the format of the signal. E.g. with a 4:3 signal on a 16:9 widescreen touch display, <**Touch Area**> should be set to "4:3 on 16:9".

In the lower section the needed display and signal format combination can be set. The recommended setup would be a 16:9 or 16:10 screen format. Only widescreens can display the graphical user interface full-blown (left configuration). The only requirement to achieve this, is the capabilty if the screen to stretch a 4:3 signal horizontally. This feature is usually given with state of the art displays. If a display does not support 4:3 stretching, the middle configuration should be used and the "Touch Area" setting needs to be set as explained above.

7.3.12 GROUP menu

[**GROUP**] allows one to navigate to this menu. Alternatively, one can also press the channel number or frequency in the status bar to edit the group settings directly. Each group has its own tab.

There are three tables displayed: one for the signal settings, one for the harmonics settings, and one for the synchronization settings. The displayed softkeys correspond to the table selected by <**Selected Table**>.

In the Signal and Harmonics table, starting from version 3.14, the current integration durations of the respective values are additionally included. These help in interpreting the update rates of the normal and harmonic measurement values.
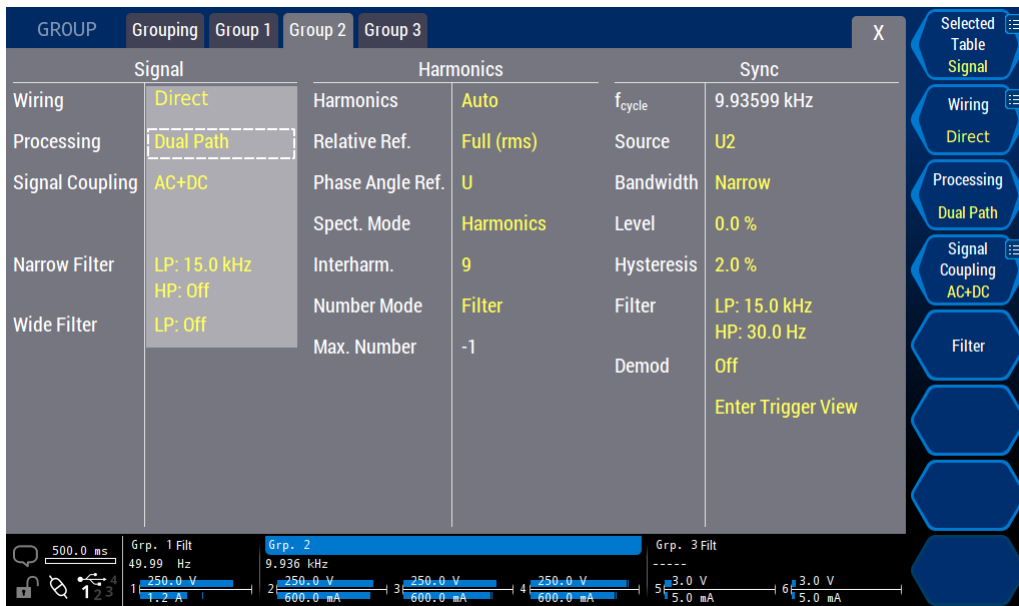
Figure 7.17: Menu to configure a specific group.

Signal settings

The settings in this table are valid for *all* channels in this group. They influence only the measured values and *not* the synchronization.

<**Wiring**> is used to provide the instrument with information on how the EUT (equipment under test) is connected. This is important because a star-delta conversion has to know this in order to use the correct algorithm. For further details concerning possible wirings, please see CONNECTION [6.2→102]. If no conversion is required, one should choose direct wiring. The list of possible wiring depends on the setting. For example an aron wiring can only be set up if the group has exactly 2 channels.

WIRE [9.10.353→383]

GROUP
[9.10.286→351]

<**Processing**> allows one select the appropriate processing mode. For details, please see BANDWIDTH [6.6→110] and the description of the interface command.

PROC [9.10.275→345]

In single-processing mode, one can just configure the desired filter (i.e. the bandwidth). The instrument decides then automatically how to get the desired data. In *DualPath* mode, one has to configure the filter for narrow and wide bandwidth independently.
In both cases the currect filter configuration is displayed in the lower part.

<**Signal Coupling**> defines whether the signal should be measured with or without the DC component.

SCPL [9.10.222→321]

> *Note: This feature can not be used to measure in a lower range. If, for example, one has a 10 A DC signal with an 0.1 A AC signal superimposed, one can select 'AC' coupling to reject the DC component in the display. However, one can not switch to a 0.1 A range, because the TRMS current of about 10 A is still flowing and would overload the 0.1 A range. The intention of this feature is to reject DC noise if there is no DC component in the signal (e.g. when measuring with current transformers).*

<**Filter**> opens the sub-menu for filter configuration.

Signal filter setting Single Processing

<**Auto Filter**> allows one decide if the filter are set manually by the user or automatically by the instrument. In the second case, the filter settings are optimized for the harmonic analysis, see FILTER [6.6.1→110]. The automatically configured filter settings are displayed.

FAUTO
[9.10.258→336]

<**LP State**> allows one to choose between full bandwidth (when switched off), a fixed bandwidth or

LPFILT
[9.10.265→340]

a user-defined bandwidth. In the last case, one can select a filter type with <**LP Type**> and cutoff frequency with <**LP Freq**>.

LPTYP
[9.10.266→340]
LPCOF
[9.10.266→340]
HPFILT
[9.10.261→338]
HPCOF
[9.10.260→337]

Similar settings are valid for the high-pass filter: It is activated with <**HP State**>, and the cutoff frequency with <**HP Freq**>. The high-pass filter can only be activated, when the low-pass state is set to "Custom" or "15 kHz".

Any filter settings are applied immediately. One can leave this sub-menu using [**BACK**].

Signal filter setting dual processing

<**Filter**> changes between the narrowband and wideband filter table.

FAUTO
[9.10.258→336]
WFAUTO
[9.10.276→346]

<**Auto Filter**> allows one decide if the filter is set manually by the user or automatically by the instrument. In the last case, the filter settings are optimized for the harmonic analysis, see CALCULATION OF HARMONICS, OPTION L6-OPT-HRM [6.8→114]. While in automatic mode, the settings are still displayed.

LPFILT
[9.10.265→340]
WLPFILT
[9.10.277→346]
LPTYP
[9.10.266→340]
LPCOF
[9.10.264→339]
HPFILT
[9.10.261→338]
HPCOF
[9.10.260→337]

<**LP State**> allows one to choose between full bandwidth (when switched off), a fixed bandwidth or a user-defined bandwidth. In the last case, one can select a filter type with <**LP Type**> and cutoff frequency with <**LP Freq**>.

Similar settings are valid for the high-pass filter. It is activated with <**HP State**> and the cutoff frequency is configured with <**HP Freq**>.

Any filter settings are applied immediately. One can leave this sub-menu using [**BACK**].

Harmonics Settings

<**Harmonics**> allows one to select between two harmonic settings:

HAAL [9.10.287→352]

- Auto
  Here the instrument ensures that *no* aliasing occurs. To fulfill the sampling theorem, either another filter is selected or the number of harmonics is reduced.

- Custom
  Here, it is possible to measure up to half of the sampling rate, but with the *risk* of aliasing. It is the user's responsibility to guarantee that the environment is free from unwanted signal components which could be present in the analyzed range.

For further details please see ANTI-ALIASING [6.6.2→111].

HRELREF
[9.10.122→282]

<**Relative Ref.**> sets the reference for the calculation of the relative amplitudes:

- Fundamental
  Uses the fundamental as reference.

- Spectrum
  Uses the sum of the available harmonics as reference.

HREF [9.10.291→354]

<**Phase Angle Ref.**> sets the reference for the phase angle calculation to voltage ("U"), current ("I"), or timepoint of synchronization.

SPECTMODE
[9.10.292→355]

<**Spect. Mode**> sets the interval for the harmonics calculation.

- Harmonics
  The group's frequency and the "Interharm." setting (see below) determine the total interval length.

- Frequency
  The interval lenght matches the currently set cycle time. The "Interharm." setting is not regarded. This setting takes only effect if the cycle mode is set to "Fixed Interval".

CYCLMOD
[9.10.355→385]

<**Interharm.**> sets the number of interharmonics.

<**Number Mode**> Sets whether the number of harmonics (including interharmonics) is set manually or is being determined by the filter.

<**Max Number**> The number of harmonics (including interharmonics) to calculate if "Number Mode" is set to "Custom".

INTERHARM
[9.10.288→353]
HNUMMODE
[9.10.290→354]
HMAXNUM
[9.10.289→353]

Sync settings

The settings located in this table are valid for *all* channels in this group and influence only the synchronization. These settings do *not* influence the measured signal directly. The measured values could only be affected *indirectly* by changing a different measurement time.

At the top of the synchronization table, the currently measured frequency of this group is displayed. This is helpful when optimizing the synchronization settings, because a correct frequency indicates correct settings. For background information concerning synchronization, see SYNCHRONIZATION [6.9→118].

<**Source**> allows one to configure the synchronization source. One can choose any voltage or current channel from the current group, extern or also any other group. When the synchronization source is set to a channel of the current group, there are additional configuration settings:

SYNC [9.10.477→437]

<**Bandwidth**> defines the exact path of the synchronization of the selected channel, see SYNCHRONIZATION [6.9→118]. When the setting is set to "Narrow" or "Wide", the following settings are also possible:
<**Level**> allows one to set the trigger level, <**Hysteresis**> is for a range around this. With bandwidth set to "Narrow" one can additionally configure the <**Low-pass**> filter, the <**High-pass**> filter and the <**Demodulation**> for the signal.

SYNCBW
[9.10.467→432]

TLEV [9.10.473→436]
THYS [9.10.471→435]
TLPCOF
[9.10.474→436]
THPCOF
[9.10.469→434]
TDEM [9.10.468→433]

If bandwidth setting "Narrow" is active changing synchronization settings or "Enter Trigger View" (displayed only if bandwidth "Narrow" is active) open the Trigger View Dialog (see below).

When selecting the external synchronization source the Frequency I/O pin of the synchronization connector is configured as an input. If none of the groups uses external synchronization the pin is configured as an output and outputs the sync of group 1. Further details can be found in SYNCHRONIZATION CONNECTOR [4.3→46].

7.3.13 Trigger View

This dialog will be displayed automatically on sync setting changes in bandwidth mode "Narrow" or after tapping "Enter Trigger View". The trigger view visualizes the effects of the sync settings like sync filters and level or hysteresis. The resulting synchronization frequency is, in addition to the footer at the screen bottom, shown between the dialog buttons.

Below the diagram, the y-offset and the x- and y-resolutions are shown. The graph can be navigated like the scopes using touch or the arrow keys. The time resolution can be set using <**t/div**> or the rotary knob.

> *The level and hysteresis can be directly moved by touch.*

<**Fit**> allows to to set the y-resolution to either "Full Peak Range", "Half Peak Range" or "Signal". "Follow Hysteresis" follows continuously the level and hysteresis settings. "Hysteresis To Graph" sets the level and hysteresis to the current view.

To activate a fit setting the softkey needs to be pressed long after the fit setting has been chosen.

7.3.14 CHANNEL menu

Use [**CHANNEL**] to enter this menu (see Figure 7.18 [→186]). Here one can configure parameters related to correct range selection. Alternatively, one can press one of the channel indicators in the status bar to navigate directly to a specific channel in this menu.
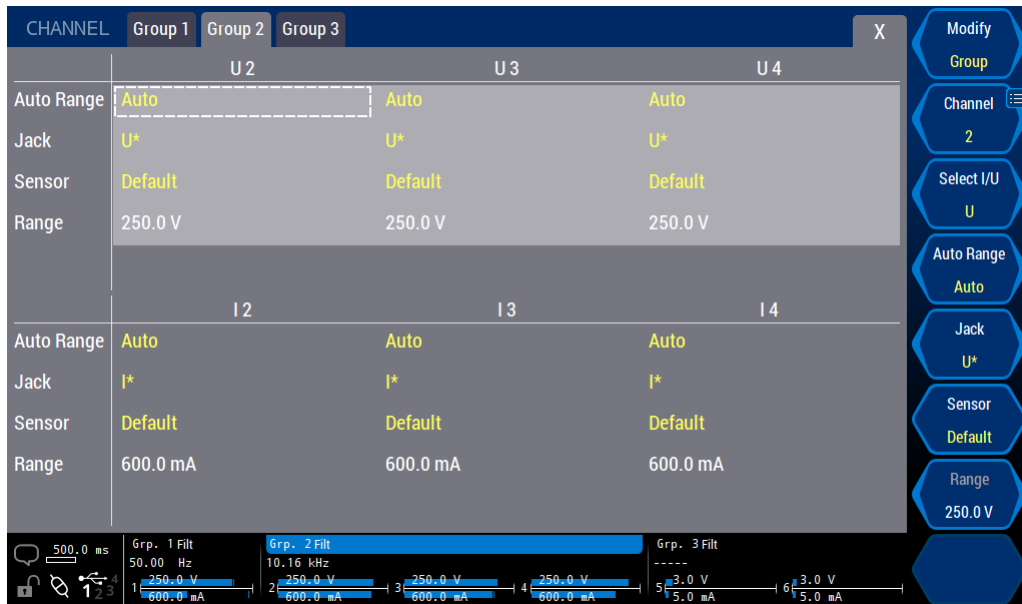

Figure 7.18: Menu to configure range-related parameters for each channel.

Navigation

With <**Modify**>, it is possible to select whether the modifications will effect only one channel or the whole group (i.e. all channels of a group). This is very useful as one usually has similar signals on all channels of a group. One can select the desired channel with <**Channel**>.

Rangesettings can only be applied to all channels of the group, if the range lists are identical. This is usually the case. But the lists may differ if a sensor is in use on some channels of the group. In this case, information is giving on this issue and the setting must be made separately.

To switch between the voltage or current channel settings, one has to use <**Select U/I**>. In addition it is also possible to use the rotary knob for navigating to and changing of settings.

Setup

UJACK
[9.10.373→394]
IJACK
[9.10.239→328]
URNG [9.10.386→399]
IRNG [9.10.252→333]

The first thing to configure is the <**Jack**> which is used to connect the signal. The current channels and several voltage channels have different input jacks. They are used to connect sensors with small signals to the instrument. Depending on this jack, one can select a relevant range from the list of available ranges <**Range**> . If a sensor is connected, each range entry in the list consists of two values: The first value specifies the range which physically exists in the instrument. The optional second value (which is presented in parenthesis) is the range which is constructed by the sensor. This second value is constructed from the information which are obtained from ZES ZIMMERsensors with the Sensor ID connector (see CHANNEL PLUG-IN MODULES [5.3.2→91]) and from the scaling factor of the channel. Scaling is usually used in conjunction with the external sensors.

UAUTO
[9.10.375→395]
IAUTO
[9.10.241→328]

Using <**Auto Range**>, one can switch between automatic range selection and manual range selection .

With automatic range selection, there are some possible drawbacks due to how an automatic range selection works:

Automatic
range selection

When the sample values are larger than the maximum allowed peak value of a range, the sample values are clipped which causes measurement errors. This situation is detected and a change into the next higher range is activated. Such a change requires some time as the analog hardware has to settle as well as the digital filters. Due to the time required for such changes, a gap with invalid values occurs. For switching down into a lower range, the situation is even more complex: Let us assume that the signal's amplitude decreases in the middle of a measurement cycle. The instrument cannot know at this point if this is a periodic behavior of the signal or a transient one. Therefore, the instrument cannot switch down in the middle of the measurement cycle. Instead, the instrument must wait until the end of the next measurement cycle to be sure if the signal was small enough to suffice a range change. Of course, a gap of invalid values occurs again due to settling times.
This measurement gap (consisting of invalid values) are usually not displayed by the instrument's GUI and should also be discarded by user applications.

In particular, the following should be regarded when using automatic range selection:

- *If one wants to measure a single peak value, do not use the automatic range selection as there is such a strong chance that this peak will be clipped or will fall into a range-change gap.*

- *If one requires that the measurement contains no gaps containing invalid values (for example during harmonics, flicker, and energy measurements), one needs to use the manual range selection. This is the only way to avoid the situation of having gaps containing invalid values in the measurement.*

- *When performing any uncertainty calculations, automatic-range selection requires one to not only know the measurement but also the actual range which was used during the measurement.*

Sensor menu

Finally, it is possible to configure external sensors with <**Sensor**>. In this sub-menu, two parameters can be configured.

The first column shows information about the connected sensor, if this sensor has a connection to the Sensor ID jack at the back of the instrument.

The scaling factor <**Scale U**>/ <**Scale I**> for this sensor is useful if one, for example, has a current or voltage transformer with a fixed ratio between the primary and secondary signal (e.g. 100 A:1 A or 3000 V:1.5 V). In such a case, one can enter the appropriate ratio (e.g. 100 or 2000) and the instrument displays all voltages, currents, power and derived values correctly. The currently measured voltage and current are shown in the next column. This scaling factor could also be used to correct the uncertainty of a connected sensor.

USCA [9.10.387→400]
ISCA [9.10.253→334]

Note: It is possible to enter negative scaling factors. This is useful if, for any optimization purposes, a channel was inversely connected. With a negative scale factor, each sample value gets the opposite sign and the measurement is corrected.

The (group) delay (<**Delay U**> and <**Delay I**>) of this sensor is useful if the group delay is know or can be determined (i.e. through measurement). For this reason, the last column shows currently measured active power and power factor as one of these values is often used to determine the group delay time.

UDLY [9.10.371→393]
IDLY [9.10.237→327]

7.3.15 ACTIONS menu

Use [**ACTIONS**] navigates to this menu where one can execute and configure various actions.

A List list of available actions is provided. One can either use the <**Action**>, the up/down arrow keys, or the wheel to navigate the list of actions. Actions can be performed immediately by pressing <**Execute Now**> or pressing the wheel. By pressing <**Map To Execute Key**>, the action will be mapped to [**EXECUTE**] and will be executed in any menu whenever [**EXECUTE**] is pressed. The action which is currently mapped to the [**EXECUTE**] is displayed on top of the menu. The mapping

of the execute key can be removed using <**Remove Mapping**>. Whenever an action is mapped, the [**EXECUTE**] the will be illuminated.

Some actions have additional setting parameters. For such actions, <**Settings**> opens sub-menu where the settings of a specific action can be configured.

The script editor can request an execution of the mapped action by using the `exec()` and `execRemote()` functions (see Table 6.5 [→132]). The only action that can be executed using this is the "Snapshot"-Action. The purpose is to automatically, non-interactively execute a screenshot or log the visible values. Execution using the script editor will therefore not cause any dialogs to be displayed (even in the case of an error). Using `execRemote()` will request the execution on the Remote GUI (REMOTE CONTROL OF THE DEVICE WITH LMG REMOTE [8.2→196]).

> *Note: To log the current cycle using the script editor, use the `logOnce()` function instead of trying to execute the "Log" action. Action execution with the script is not tied to any cycle being visible on the GUI or to the logger, and the GUI may accumulate multiple `exec()` calls into a single action trigger. The `logOnce()` call does not have this behavior, and ensures the exact cycle being seen by the script is logged.*

Log action
The log action allows you to log the last received data from the device into your preconfigured log file (see LOG TAB [7.3.17→189]). Alternatively you can enable logging continuously until you execute the action again. Whether the last received data is logged, or logging continues until you execute the action again, is configured in the settings of this log action.

Snapshot action
The snapshot action allows to create a screenshot of the currently displayed content into a previously configured folder and image file name. For some menus, the currently displayed values can be exported into a Log-file along with the screenshot. The file names of the image and Log-file can be configured independently ahead of the action execution. In the configured file names, the placeholders [name] and [date] will be replaced by the menu name and the current date and time, respectively.

If you log a scope trace by the action, then the created Log-file will contain the signal sources, the sample rate, the pretrigger-duration and an accurate timestamp of the first sample aswell as the trace data itself. Logging a plot tab (PLOT [7.2.8→162]) will record the timestamp of every logged cycle aswell as the displayed values. The created Log-files can be exported into an CSV- or Octave-File by using the (see FILES TAB [7.3.20→193]) as usual.

Zero Adjust
The Zero Adjust - Action offers a comfortable interface for the configuration and execution of the zero adjustment.

Within the manual "Zero Adjust" mode, two modes are available which differ in terms of customization and persistence.

The mode "Instrument" adjusts always the complete instrument. I.e. the voltage and current paths of all channels of all groups. This adjustment will be persistent and would still be present after a restart of the device.

Because this adjustment presumes a bare decive removing any attached sensors is required, and in case of Plug 'n' Measure - sensors enforced. This ensures that distorting effects from sensors won't affect the zero adjustment.

This mode adjusts all available star and sensor jacks.

The mode "Custom" offers a wide range of customizations. This allows for a faster task specific adjustment. The scope of adjustment can be optionally limited to a single group, the voltage or the current path, and to the current range. This adjustments won't be persistent and may involve attached sensors. Attached sensors are not regarded in any way. They will be included as if they are part of the device. Hence, it is obligatory to readjust the device after any changes of the sensor hardware configuration, before subsequent measurements.

Following settings are available:

- <**Group**> The adjustment will be done for the selected or for all groups.

- <**Path**> The adjustment will be done for the selected path (U, I, or both).

- <**Range**> the adjustment will be done for the current or for all available ranges.

This mode always adjusts the currently active jack of the adjusted channels (see CHANNEL MENU [7.3.14→186]).

After triggering the adjustment by tapping <**Excute Now**> or by pressing [**EXECUTE**] a dialog shows up for the duration of the adjustment.

Event Trigger Action

The Event Trigger - Action allows to trigger the transient recording anywhere and anytime.

For this purpose, the traces to be recorded, the recording duration, the sample rate, and the pre-trigger must be set as needed, beforehand.

This action is intended for manual triggering of a transient recording. Hence, any set trigger conditions would be disabled on first execution of the action. If a rearming behaviour with set conditions is needed, the setting "Auto-Rearm" of the transient configuraton (TRANSIENT CONFIGURATION (RECORDING) [7.2.7→160]) must be used.

In case the recordings shall be logged, the logger must be in a configured and activated state (see LOG TAB [7.3.17→189]);

After configuration, it is recommended to make sure everything works as intended by executing the action at least once.

The setting "Rearm after execution" sets whether the transient system will be automatically rearmed. If active ("yes"), the transient system will be immediately rearmed after the action finished, the data is being displayed, and, if wanted, has been completely logged. The next recording starts immediately when the action is being triggered. If inactive ("No") and the transient system is not already in the armed state, it will automatically be armed prior to the recording and NOT rearmed after the action finished.

When the transient system enters one the modes "Loading" or "Searching" previously recorded data can no longer be accessed by the interface, because the internal transient storage is being overwritten. If "Rearm After Execution" is active, the graph will show the last recording until the next recording starts.

7.3.16 STORAGE menu

[**STORAGE**] allows one enter this menu. Here, one can log measured values to internal or external memory, manage files and generate a status report.

7.3.17 Log tab

In this tab, one can define which values should be logged to a mass storage device. <**File**> is used to define the location and name of the file where the values should be stored: If there are multiple drives available, they can be selected by using the up/down cursor keys. The wheel can be used to navigate in the right-hand side file and folder list. <**Up**> changes back into the parent directory. On the instrument, the internal memory and external USB memory sicks are presented as a separate drives. <**Copy**>, <**Paste**>, <**Rename**>, <**Remove**> and <**Add folder**> are used to manipulate files and directories.

With <**Comment**>, one can enter a comment concerning this measurement which is also stored along with the logged data in the file. Finally <**Configure Values**> is used to define which values should be recorded:

Copied measurands from the Custom or the Efficiency menu can be inserted using <**More ...**> followed by <**Paste**>.

With <**Type**>, one can control the values which are displayed under "Name". If "All" is selected, all values are displayed. This list can be very long. However, for example, by selecting "Voltage", the list of symbols becomes more manageable as it now contains only voltage-related entries.

<**Symbol**> defines the name/symbol of the value to be logged. It is identical to what is displayed in the measurement menus.

<**Phase**/**Link**> defines, from which phase or calculation the symbol should be used. For example, the trms value of the voltage can be from the second phase in group 3 (i.e. $2\,G_4$), from the linked value between phases 1 and 2 of group 1 (i.e. $12\,G_1$), the sum of group 2 (i.e. $\Sigma\,G_2$), etc.
When the number *before* the "G" is a single digit, it defines the phase of the group. When the number has two digits, it defines the linkage between phases *inside* the group. The Σ-symbol stands for the sum values of a group.
The number *after* the "G" defines the number of the group.

Depending on the currently chosen measurand (<**Name**>), other suffixes like <**Flicker Ch.**> or <**Group**> can be selected instead of <**Phase**/**Link**>.

When dual-processing mode is active, the <**Bandwidth**> is used to select the narrow or wide bandwidth value.

A currently selected value can be added to the list of selected values using <**Add**>. To remove an item from the list, use <**Remove**>.

After one has chosen values to log, <**Create & Start**> can be pressed to create the log file and immediately start the recording. Alternatively, one can take the individual steps of creating the file and starting the logging by pressing <**Create**>.

When having created the log, one can then switch between a continuous mode and a single-step mode of the log. When single-stepping, <**Once**> can be used to log a single cycle of values. When in single-stepping mode, one can export the log (see below). By pressing <**Stop**>, the log is terminated and the setup screen is displayed again, allowing to setup a new logging.

> *Logging can also be controlled by a running script (see* FUNCTIONS *[6.15.5→130]). In this case, the log should only be created but not started. <**Script logOnce**> allows the logger to ignore the script controlled logging. This is of use, when the already logged values shall be exported and the still running script must not add any more values.*

<**Export**> provides a way to export the current logged file by opening the export dialog. See LOG EXPORT [7.3.18→190] for more details on this.

7.3.18 Log export

The log file has a special binary format which is useful due to its smaller file size. Exporting a log means to take a log file, and convert it to another file, that can be read by third-party programs of other companies.

> *Exporting from a recorded binary file instead of recording directly into the export format has big advantages: By this it is possible to export* many times *just parts of a file or to export to multiple formats. By this it is possible to test different formats and subsets without loosing any data.*

There are two ways to export a log file. Export can be done in the GUI using the export dialog, and using a command line tool on windows for advanced users. With both ways, multiple log files can be exported at once.

A log file can be exported either by using <**Export**> on the LOG TAB [7.3.17→189] when being in single-step mode of a created log, or when being in the FILES TAB [7.3.20→193] and having a single log file selected and pressing <**View**/**Export Record**>. Alternatively, the command line tool can be used, which is described below.

Multiple log files can be exported at once when selecting multiple log files in the FILES TAB [7.3.20→193] and pressing <**Export selected Logs**>.

Export dialog

The export dialog provides a list of the logged values and a initially empty list containing the values which should be exported. The values are added and removed from the export list by using <**Add**> and <**Remove**>, respectively. Switching between the lists can be done by touch or by the rotary knob. After pressing the knob it moves the selection within the list and the respective action (<**Add**> or <**Remove**>) can be triggered.

The data is exported as either comma or character separated values (CSV) or the binary Matlab format. Some export settings are only valid for certain export formats, while other settings are common to all export formats. The following lists the supported formats and their options. Following that, the common settings are described. Some of the more common softkey options are also displayed in the dialog directly for more convenient operation. Refer to Figure 7.19 [→191] for an example scenario picture of the export dialog. In the example, a single file was selected for export. When selecting multiple files, some options are not available, because they do not apply to a multiple-files export.

Figure 7.19: The export dialog for a cycle values log.

Comma or character separated values (**CSV**) The values from one point of time are printed in one row. <**Split on Rows**> allows to split the exported file based on the number of rows (which can be configured by long pressing this key). This is useful for older spreadsheet programs which can typically only handle up to 32767 rows.

> <**Split on Size**> allows to set a maximal file size. A larger export will be split into files of this size. The size can be set after a long press on the key. The minimal size limit is 1Mb.

> <**Field Separator**> and <**Decimal Separator**> set the respective separating character. This Softkeys can be found in the sub-softkey menu reachable by <**More ...**>.

Matlab/Octave writes the values in a binary format (**MAT**) readable by the Matlab (v6+) and GNU Octave programs. Samples and sample timestamps from the scope and transient scope are written as a NxC matrix (Rows x Columns), where C is the number of shots recorded and N is the number of samples recorded in each shot. All other values (including the sample rate of a scope shot, and normal cycle values) are written as a vector of Cx1 entries, where C is the number of cycles or shots, respectively.

<**Export From/To Time**> allows to export values between two timepoints only. <**Step**> (The soft key is located in the sub-menu reachable by <**More ...**>) allows to export every nth row or value

sets separated by a defined duration. Left/right sets the mode and long press edits the respective value.

If the currently selected value on the right side is a list, <**List Interval**> and <**Odds Only**> specify the parts of the list to be exported.

After selecting options for the export, a filename must be chosen for the destination export file. When exporting multiple files at once, a pattern must be provided, as shown in Figure 7.20 [→192].
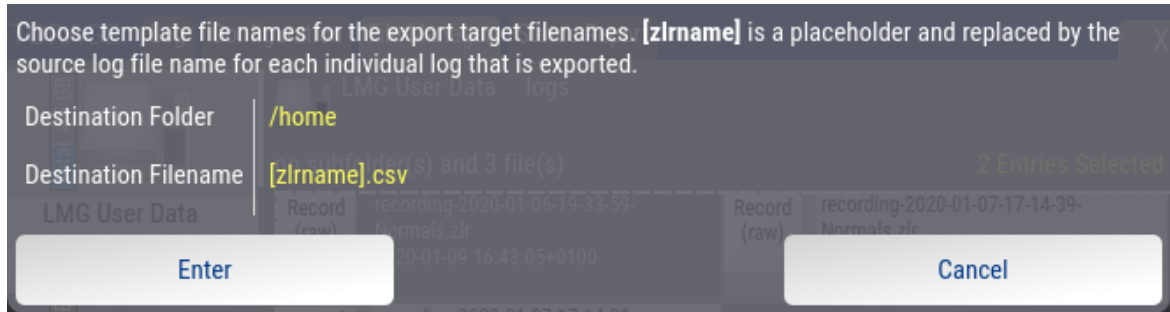

Figure 7.20: Choosing export file names for multiple files at once.

.

Export command line tool

Using the command line utility, log files can be exported automatically, with options passed to the utility programmatically. The utility comes bundled with PC SOFTWARE [8→195] and is automatically added to the system path of command line utilities. Calling it from a command line shell such as the Windows PowerShell, `cmd.exe` or `bash` allows to flexibly export one or multiple log files automatically.

Starting `zlrexport.exe` without any arguments or with `-help` gives a summary screen of accepted options.

```
../bin/zlrexport.exe [arguments] <input-file>
Allowed options:
  --help                        produce help message, then exit
  --format arg (=csv)           export format. Possibilities:
                                csv: Export to the .csv format
                                mat, matlab, octave: Export to the matlab .mat
                                format
  --export-times [=arg(=1)] (=1) Include the timestamp and
                                measurement-durations of each column in the
                                export. This increases the size of the export
                                considerably, so it may be disabled.
  --output arg                  export-file name, instead of the default. The
                                default is constructed from the input export
                                file name, with the appropriate extension.
  --file arg                    file to export, alternative to passing
                                <input-file>

CSV format options:
  --csv-fieldsep arg (=comma)   separator between columns. Possibilities:
                                comma: ','
                                semi: ';'
                                tab: tab character
                                space: ' '
  --csv-decsep arg (=dot)       decimal point. Possibilities:
                                dot: '.'
                                comma: ','

Matlab/Octave format options:
  --mat-timeformat arg (=1970)  encoding for time values. Possibilities:
                                1970: Seconds since 1.1.1970 CE, UTC.
                                0000: Days since January 0, 0000
```

In the following listing, an example is given in the traditional command line shell `cmd.exe` that comes bundled with Windows. The example iterates over log files within the current working directory, and exports files found therein. For further information and examples, please consult the respective manuals of the shell that you use.

```
for %f in (*.zlr) do (
   zlrexport --output "example-%~nf.csv" "%f"
)
```

Listing 7.1: Example for cmd.exe

7.3.19 Configuration Tab

This menu allows to save and load the complete configuration including the custom menus.

This does not include files created by the user like recordings or screenshots. This data stored in "User Data" must be saved separately.

<**Directory**> allows to choose a directory. The list beneath will be populated with the configuration files located in the chosen directory.

Using <**Load**> allows to load a selected configuration file (grey highlight). This action will also replace the current custom menus by the ones loaded from the file.

<**Save Config**> allows to save the current configuration and custom menus. In addition to the file location and name, it is possible to enter a comment which will be displayed in the configuration file list.

A configuration file can be deleted using <**Remove Config**>.

7.3.20 Files tab

This menu works like a file manager where one can browse, copy, paste, rename and remove files.

Any available drives, including USB-Storage devices, are listed on the left hand side. The Right hand side shows the files and folders contained in the current Folder. The current folder is displayed above the files.

The rotary knob switches between the drives and files lists. After a long press on the knob, a drive or a file/folder can be selected. A short press on the knob checks the currently selected file/folder (gray highlight) or switches to the selected drive. Soft keys with functions like "Copy" or "Paste" can be applied on checked items. Cecking of multiple files can be done by long press of the middle arrow key after selecting the file/folder using the rotary knob. Actions like "Copy" or "Paste" will be applied on all cecked items.

<**Up**> changes back into the parent directory.

<**Copy**>, <**Paste**>, <**Rename**>, <**Remove**> and <**Add folder**> are used to manipulate files and directories as indicated.

7.3.21 Status report tab

The status report consists of a PDF document and a ZLOG file (packed in a single ZIP file). The status report contains all settings and most measured values. It is ensured that the printed measured values were aquired using the printed settings.
With <**Edit**>, users can add additional information such as contact person, address and an error description. <**Directory**> allows one to choose the directory where the status report is written to.
To create a report, the instrument must be frozen. There are two ways to do this: If suspicious values are present, press [**FREEZE**] and change to this menu to create a report. Alternatively, one can press <**Create Report**> and following an info dialog the current values will be frozen. Before

creating a report, the description field is shown for editing (or confirming).

8 PC software

The PC software "LMG600 Control" and the manuals in PDF format are part of the devices firmware.

The integrated software and documents can be accessed in the menu [**STORAGE**] in the file manager on the drive "Library".

The manual is located in the folder "documents" and the PC software is located in the folder "tools". The files can be tranferred onto an USB storage device and used with a Windows PC.

This integrative concept ensures that the PC software and the manual versions always match the version of the installed firmware. To ensure flawless operation, it is needed to update the Windows software whenever the firmware of the device is updated to a new version.

Optional software modules like LMG Test Suite and LMG Sample Vision can always be started in a demo mode. Without purchase of the respective options, this software can usually not retrieve data from a connected device but can be tested to a certain extend with the provided demonstration files.
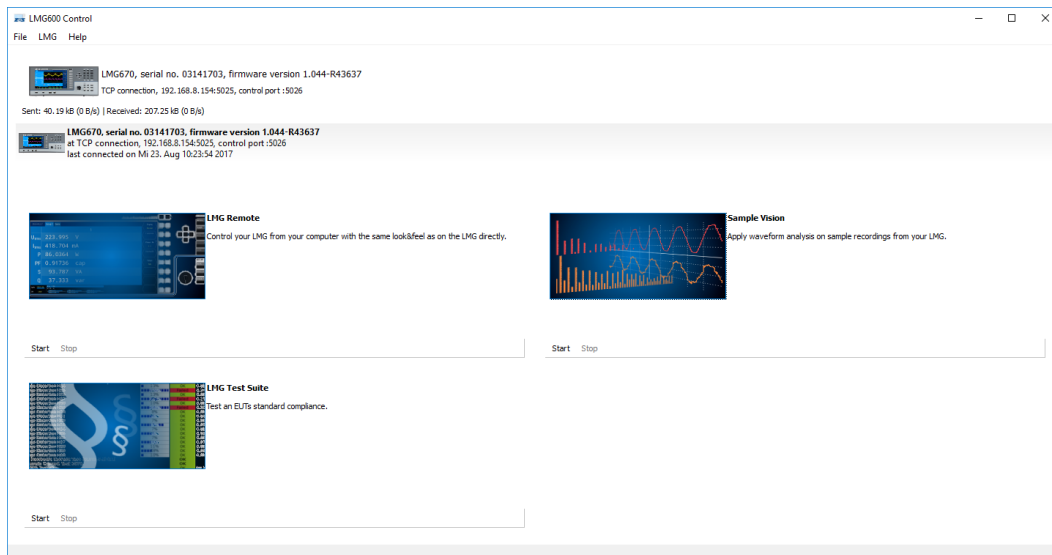
Figure 8.1: Main window of LMG600 Control

8.1 Connecting the LMG Device to LMG600 Control

The devices list in the main window shows all LMG devices that could automatically be found in your LAN network. Doubleclicking on one of the entries creates a connection to the respective device. The entry "Connect to..." in the menu "LMG" allows to connect via LAN to an LMG with network connection if the desired device does not appear in the list. If LMG600 Control was connected to the same device with the same network address in the past, the connection can be established by double click on the respective entry in the device list in the main window.

The IP address of the device can be found in the "Interface" tab of the [**INSTR.**] menu of the LMG (see INTERFACE TAB [7.3.5→178]).

Following the modules provided by LMG600 Control are explained.

8.2 Remote Control of the Device with LMG Remote

LMG Remote is a free module of the LMG600 Control software an can be launched from within LMG600 Control.

It offers a fully functional remote userinterface on a PC and allows access to all features of the instrument.
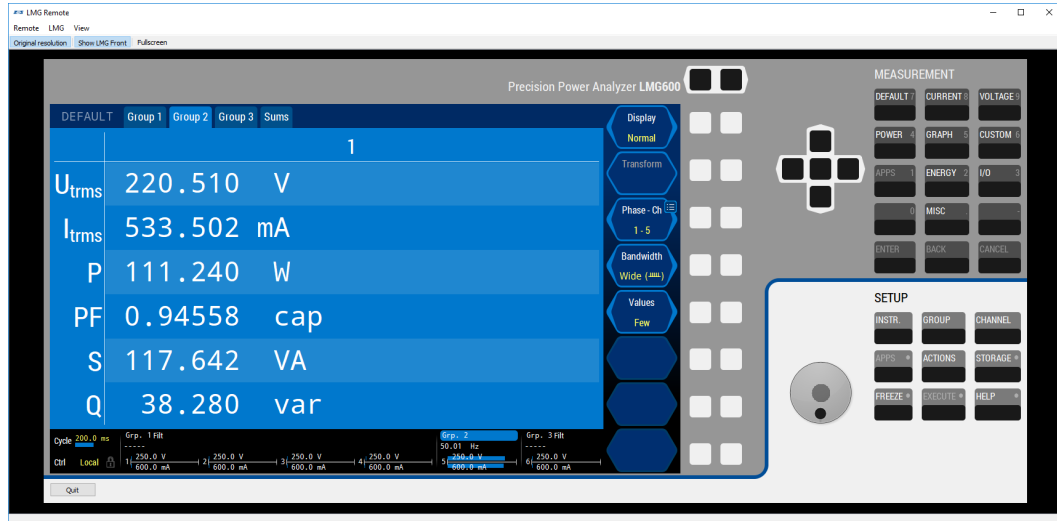
Figure 8.2: Hauptfenster von LMG Remote

8.2.1 Connecting LMG Remote to a Device

LMG Remote uses the device connection established by the LMG600 Control software. Whichever device is currently connected (see CONNECTING THE LMG DEVICE TO LMG600 CONTROL [8.1→195]), the LMG Remote module will control that device. If no connection has been established or if the connection has been closed, the LMG Remote module will switch to an offline mode which still allows to work with records.

8.2.2 Remote Control

If a connection to an LMG is active, LMG Remote shows the complete front including the graphical display of the device (depending on current view setting, see below); (see Fig. 8.2).

All knobs and buttons, and also the touchscreen, can be operated using a computer mouse or a touchscreen analogue to operationg the hardware front of the device.

A detailed description of all features of the graphical user interface can be found in Section 7.

8.2.3 Specifics when using LMG Remote

Because LMG Remotes runs on the PC, the "File Manager" in the [**STORAGE**] menu shows the local storage locations of the PC. Consequently, recording done with the LMG Remote module will always be stored to a local device of the PC. Access to the remote storage of the LMG is possible by the use of special drives in the [**STORAGE**] menu. There you can also do most of the standard operations like renaming, deletion and copying of files of the LMG.

Also, custom menus created with LMG Remote are stored on a local device of the PC. If a custom menu shall be used internally it can be easily stored to an USB device and loaded into the custom screens on the device.

The scripteditor behaves differently. A script entered in LMG Remote will always be installed and executed on the device (and also show up in the devices script editor) and vice versa. Despite this, a script should also be saved locally on the PC.

By default, LMG Remote displays the touch screen in the native resolution of the devices screen. This can be changed by deactivation of "Original Resolution" in the toolbar of the LMG Remote module. Now the display and the front, if visible, scale according to the size of the window.

Figure 8.3: Mainwidow of LMG Remote with hidden Front and inactive "Original Resolution"

Disabling "Show LMG Front" allows a large view of the screen without the front panel taking up any space.

A fullscreen view of the display can be achieved by deactivation of "Original Resolution" and "Show LMG Front" and activation of "Fullscreen".

8.2.4 System requirements

To conveniently use all features of the software we recommend to use a contemporary PC system that meets the following minimum requirements:

- Operating system: Windows 7/8/10 (32/64 bit)

- Hard drive: Software: min. 200 MB. With event trigger option (L6-OPT-EVT) min 1.2 GB. Data: approx. 20 MB per minute of measurement/phase

- RAM memory: min. 2 GB

- Processor: min. 2 GHz, dual-core

- Interfaces supported: Gbit-Ethernet

8.3 Waveform Analysis with LMG Sample Vision

8.3.1 Introduction

LMG Sample Vision by ZES ZIMMER is your solution for analysis of signals in the time and frequency domain, aswell as the gapless recording of sample values.

- For gapless recording of samples with a sample rate of up to 10 kHz, you can record up to 16 tracks at once

- Automatic gapless recording for harmonic analysis, with specification of desired amount of harmonics and interharmonics

- Display of previously recorded values from a log file in the ZLR format

- Display the frequency-spectrum of a measuring series, with calculation of the effective values of current, voltage and power

- Export the calculated frequency spectrum and effective values to the CSV or Octave (MAT) format

LMG Sample Vision is optimized for simple usage and the fast creation of meaningful analysis results. The software automatically detects the best kind of graph for different measuring series, for example of tracks from the scope, gapless-scope or transient scope, but also of cycle-based values like UTRMS [9.10.217→319], and knows the correct and specially designed axis-labeling. Different tracks are positioned correctly relative to one another, in consideration of their respective pre-trigger and sample recording rate. Cycle-based values can be shown together with scope tracks in the same plot, with the correct absolute time position.

If you have any question about LMG Sample Vision, don't hesitate to contact the ZES ZIMMER support at support@zes.com or by using the contact-details at the begin of this document.

8.3.2 Installation

The LMG Sample Vision is included in the LMG600 Control package and doesn't need to be installed separately. You can find LMG Sample Vision next to LMG Remote in the main application window of LMG600 Control. From there, you can load LMG Sample Vision.

8.3.3 Overview

LMG Sample Vision reads the content of Log-files (ZLR) which were created by LMG Remote, or by the LMG Device (see LOG TAB [7.3.17→189]) or by LMG Sample Vision itself by the recording feature. LMG Sample Vision divides the main window into an area which shows these opened ZLR-files. They can be opened by the "Open file" menu action, or automatically when recording using the "Record gapless"-functionality (see 8.3.8).

The center area of the main window shows plots of the values in these ZLR files. The values to be plot can be selected by clicking the checkmark next to the value names. The plots are grouped within tabs. Every tab has its own plot and displays it in a so-called *analysis window*. In case multiple ZLR-files are opened, it's recommended that each ZLR-file is added to a separate plot for reasons of clarity. Yet, it's still possible to show values of multiple ZLR-files in a single plot.
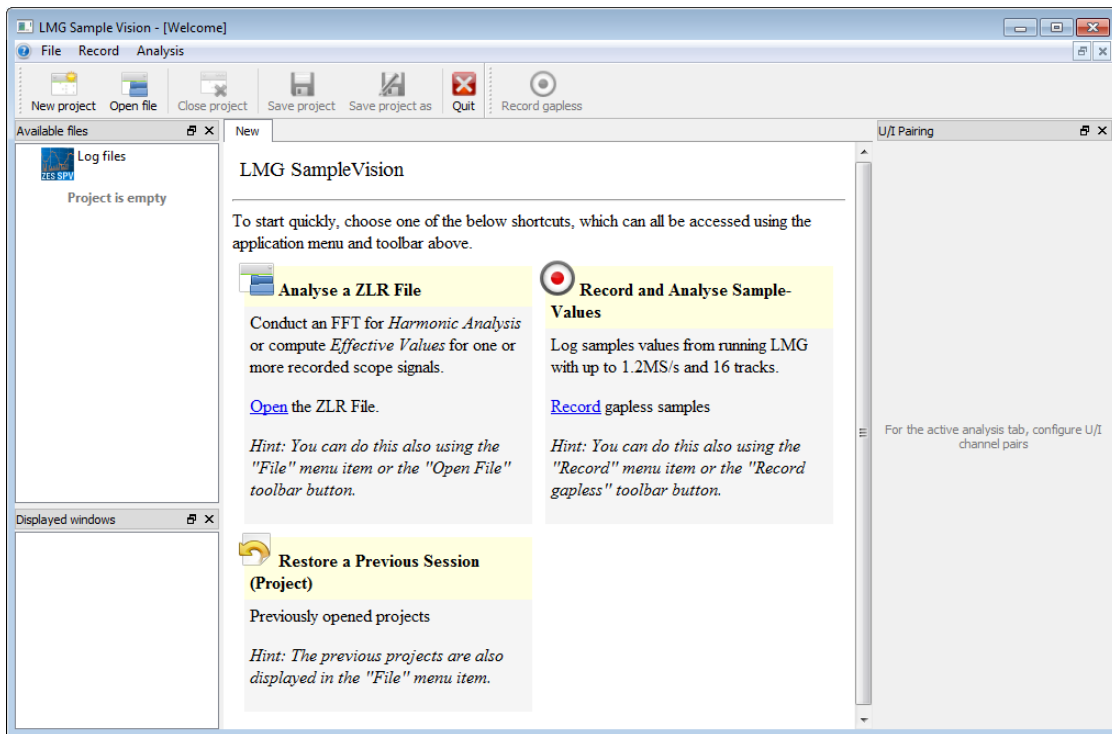
Figure 8.4: The initial view of LMG Sample Vision
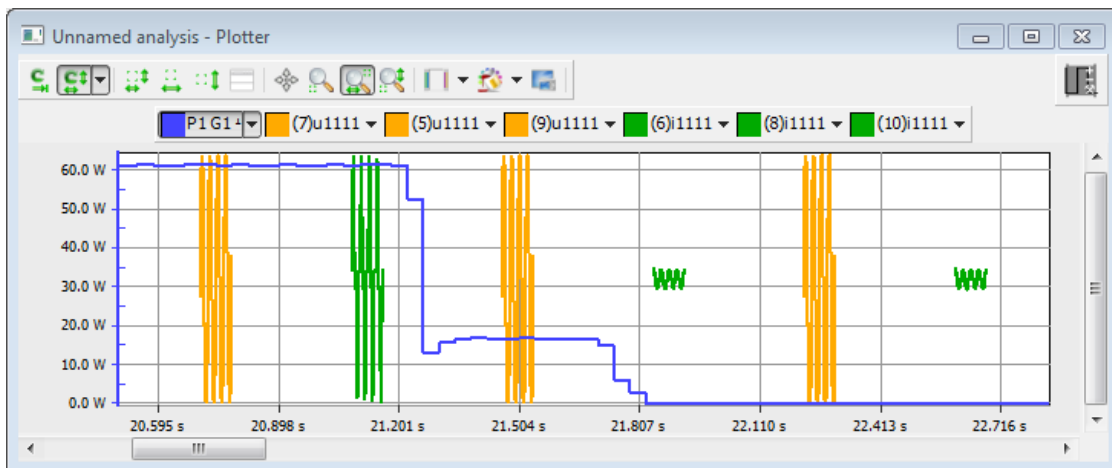
Figure 8.5: The active power (blue) plotted across the momentan values of voltage (orange) and current (green) in a time plot

8.3.4 Frequency-/Harmonic analysis and Effective values

For each tab, there is also a frequency plot, besides the usual time-based plot. You can show this plot by checking the mark in the lower left part of the main window (see 8.3.5). At first, this plot doesn't display anything, until you enable a frequency analysis in the right part of the main window. This can be done using the switch "FFT (All)". This right part of the main window also groups the signals according to their channel numbers, each group containing the respective voltage and current signal of their channel. For example, if the voltage and current of the first channel is being displayed in the currently selected tab, these signals are being grouped in the right part of the main window. The switch "FFT (All)" enables the frequency analysis for all these grouped entries. For individual current/voltage pairs, you can click the small downwards-arrow next to the entry, to enable or disable the frequency analysis separately from the other entries.

Besides the frequency analysis, you can also enable the UIP-Analysis (also called cyclebased-Analysis).
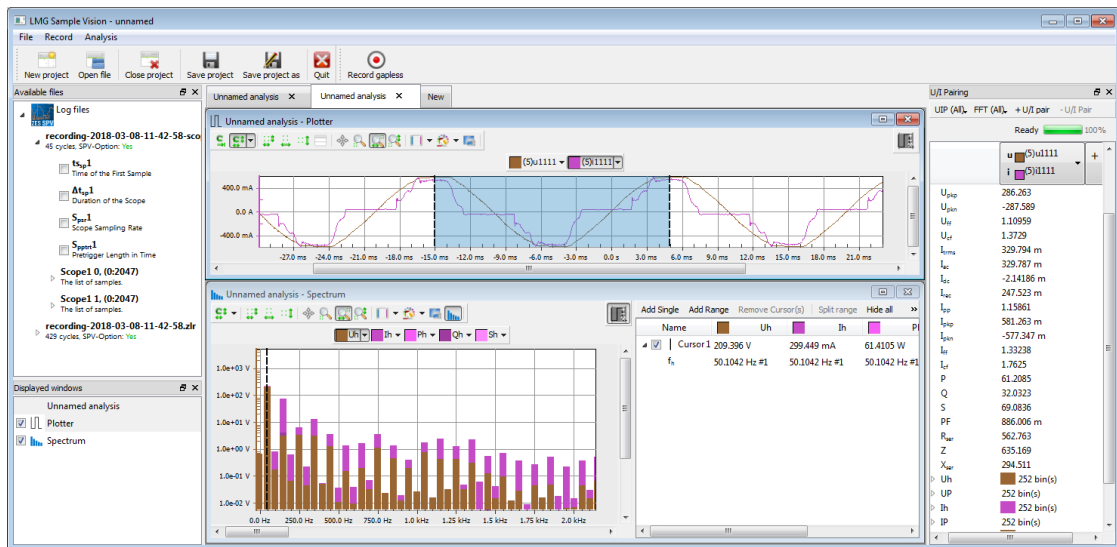
Figure 8.6: The scope tracks of the fifth cycle in the ZLR-file are used in a frequency analysis. A "Single"-cursor was added and moved to the bin of the fundamental frequency of the voltage signal (50Hz)

This calculates cycle-based values such as UTRMS [9.10.217→319]. For this, there also exists a global switch, "UIP (All)", including the ability to enable or disable the analysis for individual current/voltage pairs.

The frequency plot and the UIP-analysis determines the analyses area of the values based on a range-cursor. As soon as the first value is being added to the time plot, this range cursor is displayed as a weak blue overlay across the values. If the range cursor is moved by the mouse, the frequency and enabled UIP-analyses are updated to reflect the new selected area. If the first value that is added to a tab is from a ZLR file that originates from a recording in which you selected a harmonic analysis (see 8.3.8), the range cursor will be automatically adjusted such that it encloses the correct number of periods, according to the configured harmonics and interharmonics.

Export of Analysis results

All shown cycle-based values and the results of the frequency analysis of a tab can be exported by using the menu item "Tab", at the top of the main window. In picture 8.7, the follow-up dialog is displayed. For this, all ZLR-files of a tab need to be created with an LMG that has the L6-OPT-SPV option. If this option is not present, the "Export"-Button in the dialog is not available.

The "CSV"-format which was used in the example picture saves all values as a comma-separated list of values (other seperators than comma are available). Values formatted this way can be read by many office suites.

8.3.5 Analysiswindows (Plots)

The analysis windows available in the currently visible tab are enlisted in the lower left area of the main window. Each tab has a single time plot and a single frequency plot. Both of these analysis windows can be hidden by closing them as usual by their upper-right close button. They can be shown again by checking them using the checkmark in the lower left area of the main window. By using the context menu in this area, the analysis window can also be *detached*. This way, the analysis windows are decoupled from the main window and become normal application windows as part of your desktop. If you close the analysis window while it's detached, it will become a part of the main window again.
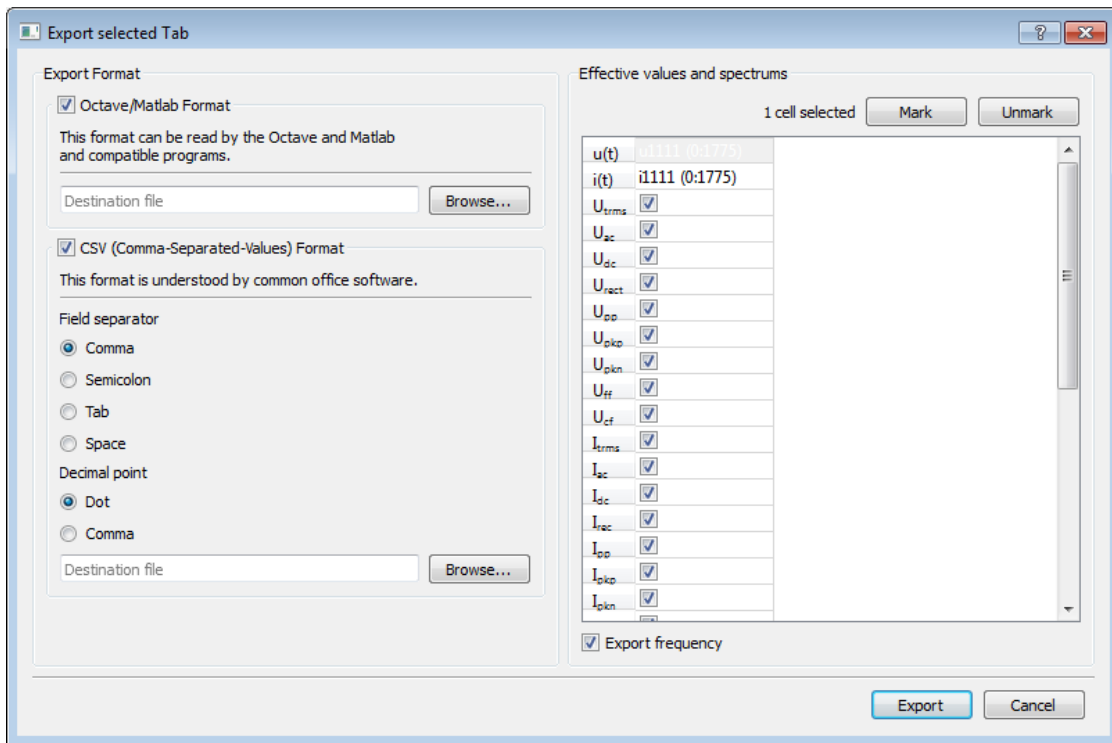
Figure 8.7: Export der berechneten Werte
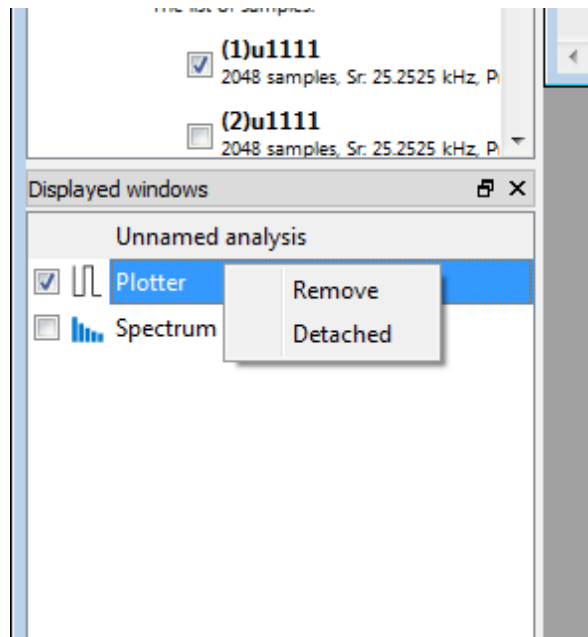
Figure 8.8: By using the context menu, an analysis window can be detached from the main window

8.3.6 Time and frequency plot

The plot has a toolbar in its upper area by which various tools can be activated. These tools determine the behavior of the mouse within the plotter. This includes the tool "Move", by which the display can be moved around. Furthermore, there are tools "Zoom", "Zoom X" and "Zoom Y" available that provide a magnifier for zooming into interesting areas of curves. If you want to move a cursor, it suffices to press on either ends of it and drag the mouse around.

The frequency plot has an additional switch to activate or deactivate the bar plot display. If this switch

is deactivated, the amplitudes are displayed as lines instead of bars. This can increase the clarity when much spectrums are displayed at once.

Some switches in the toolbar provide a small downwards arrow besides their icon. The arrow provides access to further options if clicked.
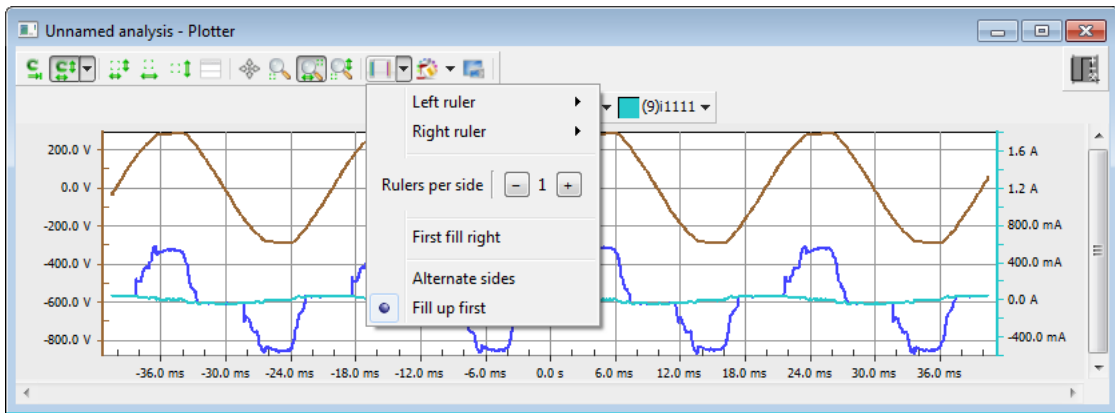
Figure 8.9: The left and right ruler space can both be assigned used to display axis values. The rulers are assigned starting from the selected signal and continuing with the last selected signal and so on. Subsequent rulers are put next to other rulers on the same side if "Fill up first" is selected. Otherwise ("Alternate sides" is selected), the rulers are distributed across both sides equally.
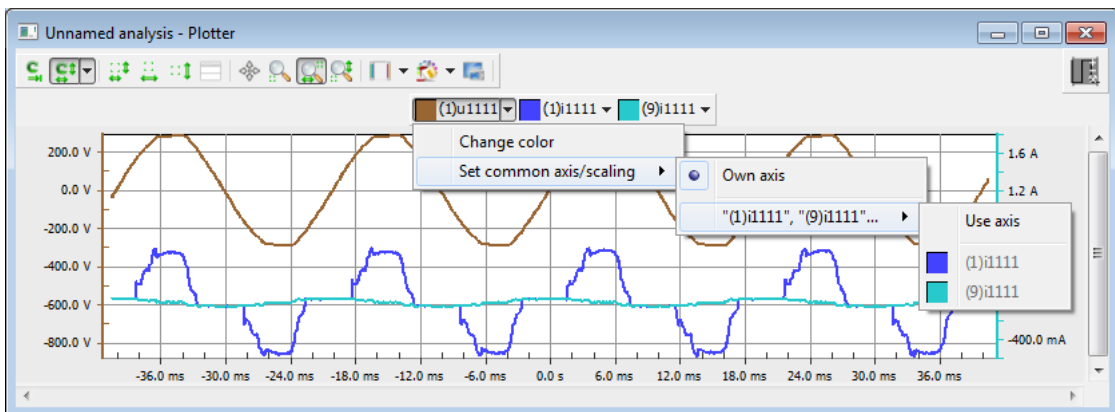
Figure 8.10: Clicking the arrow next to a signal's name allows to change its color and assign it to the same axis as another signal's axis. In this example, all current signals were given the same axis, and the voltage signal has an own axis.

Cursor

The time values and duration, as well as further values of a range cursor can be displayed by pressing the small icon in the upper-right area of a plot. Then, the plot is split and for each signal in the plot, the cursor values are displayed. The range cursor is displayed as two separate entries that represent the left and right endpoint of the range. If you click the small "+"-symbol next to a cursor in this view, more values are displayed for the cursors, such as the duration of the range cursor.

Further cursors can be added by using the "Add Single" and "Add Range" actions in the top area of the cursors view. A "Single"-cursor consists only of a single line that marks a specific time point, instead of a time range. The frequency and UIP-analyses always use the *first* range cursor to determine the range of values to be analysed. With the "Remove cursor" action, the selected cursor (click on the cursor's row to select it) can be removed. The first range cursor cannot be removed, as it's reserved to be used for the frequency and UIP-analyses.
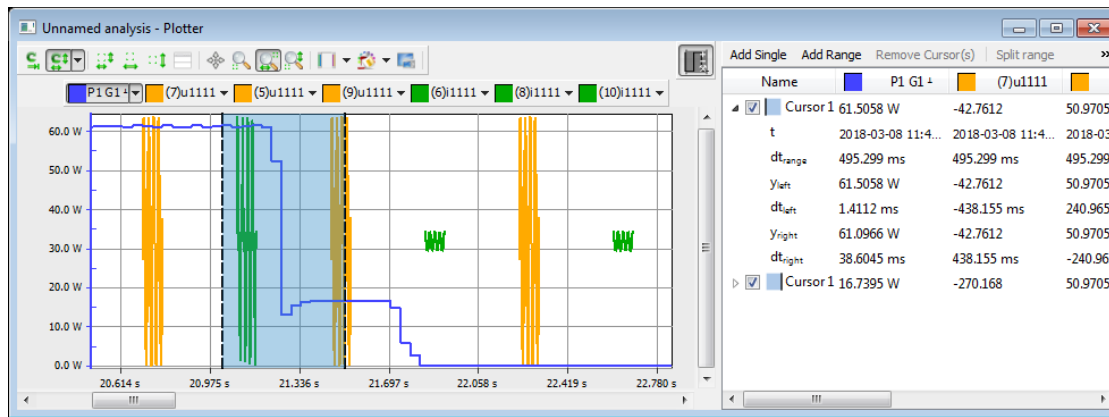
Figure 8.11: Range cursor in the time plot. The left and right values next to the left end of the range $((y, dt)_{(left, right)})$ are displayed alongside the length of the range cursor. The icon of the cursor shows whether the values show the left or right end of the range

8.3.7 Save and Load

If you have a number of ZLR-files and tabs opened, you can save this state by using the menu item "File" and then "Save". This creates a file on your computer with the ending *SVP* that can later be reopened in LMG Sample Vision. Reopening provides all previously opened files and opened tabs again. These SVP-files can even include complete Log-files. If a ZLR-file was opened for instance using "Open file", then this file by default won't be embedded. This can be changed by moving the mouse over the entry of this ZLR-file in the left area of the main window. Then switches appear above the ZLR-file name. The switch with the arrow on it can be clicked to embed the file. After saving, the file data will then be a part of the SPV-file.

8.3.8 Recording of sample values

If the software is connected with an LMG600, one ore more tracks of the gapless scope (see GAPLESS SCOPE [6.14.3→124]) can be recorded and later be analyzed like any other Log file. A device connection can be established in the LMG600 Control main application window. Afterwards, the record function is accessible by the "Record → Record Samples" menu item. The connected LMG needs to have the L6-OPT-SPV option for this gapless scope record feature to be supported. If the option is missing, a warning will be displayed in the record configuration dialog. Different options are available in the configuration dialog and can be selected depending on the use of the recorded samples. There are different main parts in this dialog

Log information The main log parameters are set here, which includes a suitable name and the file location on your local computer disk.

LMG Channel Groups Here you can see the current group sync frequencies of the LMG which are updated regularly and displayed in the first row. On the first update, these frequencies are taken and written into the second row. The frequencies of the second row are used to determine the sample rate and record duration for the Log (see below). If you wish to take them from the LMG once again, you can press "Use LMG's" to update them a single time. If you want to provide own frequencies, for example if you know that the ideal frequency would be 50 Hz, you can check "Custom" and enter these frequencies manually into the second row.

Signals tab Here you can enter up to 16 signals that are to be recorded.

Harmonics/Samplerate tab This tab configures the sample rate with which the signal is recorded and the duration for which the samples are recorded. To determine these parameters, you can either specify these directly. This mode is known as the *Frequency Analysis*. Alternatively, the sample rate and record duration can be inferred by providing the number of harmonics and the number of interharmonics that you want to later analyze by doing an FFT on the sampled signal. This mode is known as the *Harmonic Analysis* and requires you to specify a "Sync group". This channel

group provides the frequency of the fundamental wave f_1, which together with the number of harmonics N_h determines the needed sample rate according to $S/s = 2f_1(N_h + 1)$. The number of interharmonics determines the fraction of the fundamental frequency of which you later also want analyze the amplitudes. For a fundamental frequency of $60\,\mathrm{Hz}$, an interharmonic of 1 means that a later FFT will initially be done over two complete cycles, to discover the amplitudes in bin-steps of $\frac{60}{2}\,\mathrm{Hz}$, instead of $60\,\mathrm{Hz}$. Therefore, this interharmonic setting influences the amount of cycles that needs to be recorded, by which LMG Sample Vision infers the record duration.

The recording from the gapless scopes is specified with a maximal sample rate of at least $10\,\mathrm{kS/s}$. The use of higher sample rates may be possible, but can result in skipped cycles in the plot, log, and other parts of the LMG. These higher sample rates are called "highspeed". If such a sample rate is selected or infered, LMG Sample Vision displays a disclaimer that informs you of the possible consequences of doing this and that ZES ZIMMER holds no liability for malfunctions in your LMG or associated software that results from the use.
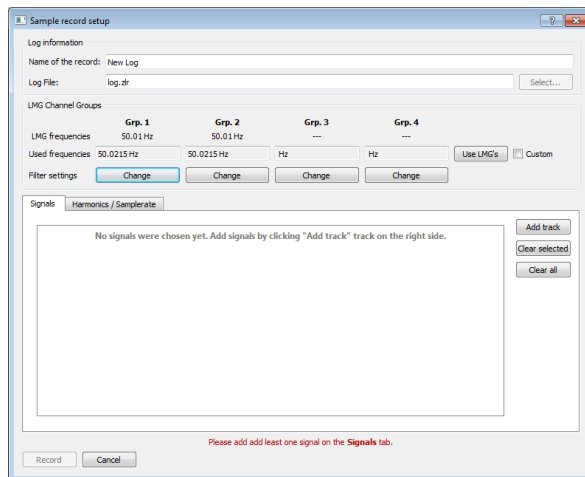
Figure 8.12: Preparation of a gapless-recording
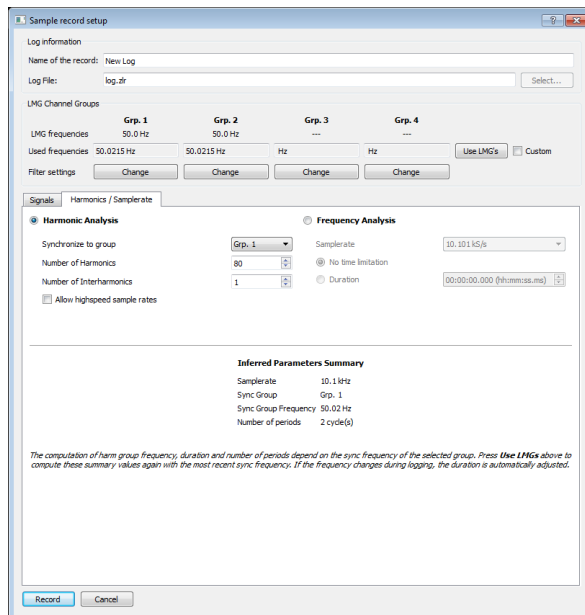
Figure 8.13: Setup of an harmonic analysis

If you start a recording by pressing "Record", you can follow the progress in the follow-up dialog and can stop the recording at any time, if you did not configure a record duration limit. Afterwards, you can add the log-file to the currently opened project by clicking on "Add to project" or discard the recorded file, if you wish to do so. The file is then automatically opened, and the cursors

(see Chapter 8.3.3) are automatically set to start at a zero crossing and extend for the number of previously configured cycles or record duration. As the zero crossing information depends on several LMG device options, the corresponding information is delivered by the LMG during the record. It may happen that the LMG sometimes cannot determine a zero crossing for low sync frequencies. In this case, and if you selected to do a frequency analysis in the record config dialog, the cursors are positioned starting at the begin of the recording. This has no effect on the FFT for periodic signals, however.
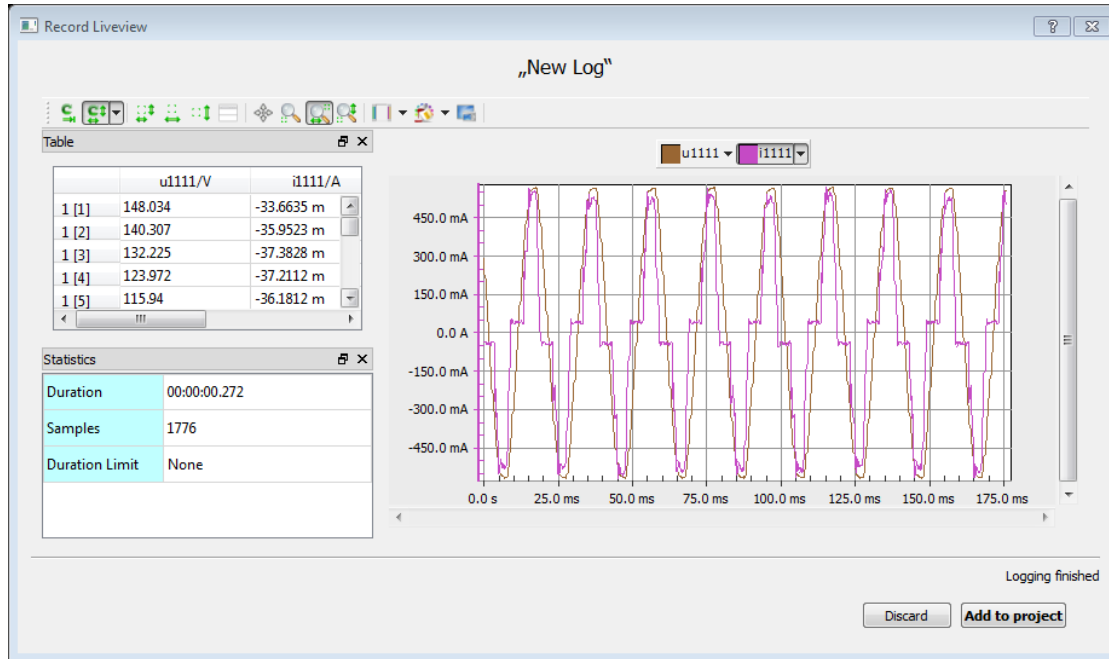
Figure 8.14: Live-View of a recording

8.3.9 Frequently Asked Questions

If your questions are not answered here feel free to contact ZES ZIMMER support via support@zes.com or use the contact information provided on page 2 of this document.

Scope-Tracks in the live-view dialog of a recording flicker

If many scope tracks with many samples are displayed at the same time, you may experience flickering of the displayed value graphs. This happens because the LMG Sample Vision tries to keep the user interface responsible. Drawing all the graphs at once would block the user interface for an unacceptable amount of time. If you zoom into the graphs (using the "Zoom"-tool), the flickering will go away.

8.4 CE Conformity testing with LMG Test Suite

The LMG Test Suite by ZES ZIMMER is the solution for conformity tests regarding

- Harmonics from 0 to 16 A according to IEC/EN 61000-3-2

- Harmonics from 16 to 32 A according to IEC/EN 61000-3-12

- Flicker from 0 to 16 A according to EN 61000-3-3

- Flicker from 0 to 75 A according to EN 61000-3-11

- Standby according to EN 50564:2011/IEC 62301:2011

LMG Test Suite is optimized to be easy usable and produce test results very quickly. Nevertheless it is possible to perform very detailed analysis of all test criteria if desired. Thus LMG Test Suite can be used in both conformity testing and system development to not only perform tests of existing products but also support development engineers in creating new products that meet legal requirements.

If any questions arise when setting up or using LMG Test Suite, the ZES ZIMMER support should be contacted via support@zes.com or via the contact information provided at the beginning of this document. Some issues that may arise are covered at chapter 8.4.5 with a collection of answers to the most frequently asked questions regarding LMG Test Suite.

8.4.1 Installation

LMG Test Suite is a module of the LMG600 Control Software and can be launched from within it. Nothing more than LMG600 Control needs to be installed.

LMG600 Control should be installed on a local drive on the PC for optimal performance. The same is true for working copies of recorded data.

8.4.2 Overview
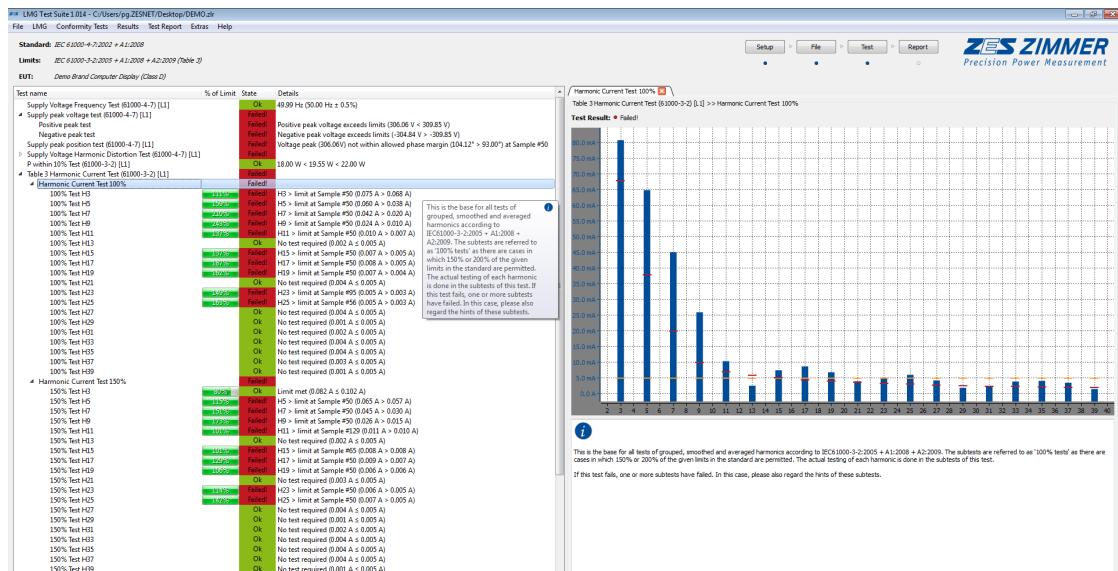
Figure 8.15: LMG Test Suite main window

The main window of LMG Test Suite is designed to give an ideal overview of the current test state and results. In the upper right corner you can find the Quick Access Bar which allows to directly open the most important dialogs. The arrangement of the buttons corresponds to the order in the testing procedure. Blue dots visualize the overall progress on your way to the final test report with all test information.

The Quick Access Bar contains the fastest way to start a conformity test: By simply clicking on the "Test" button. LMG Test Suite will ask you for all required information before starting the actual measurement.

The main part of the window is divided into two parts: On the left side all subtests of the selected conformity test are shown together with their individual results (see section 8.4.2). All tests can be analysed in a very detailed way on the right side of the window. See section 8.4.2 for details.

The vertical divider of the view can be moved with your mouse cursor. If a smaller display is used with LMG Test Suite one may even completely hide one side by dragging the divider completely to the border of the window to use less display space.

Setting the test parameters
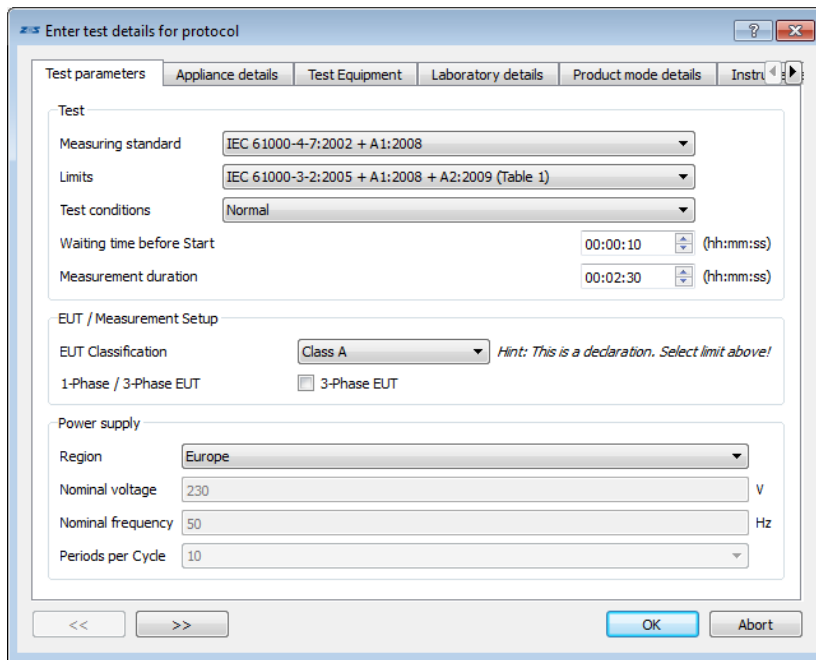
Figure 8.16: Test Details dialog

Before starting a measurement, information about the equipment under test (EUT) must be entered and the standard for the test must be set. The most important settings can be found on the first tab of the test details dialog (fig. 8.16). This includes settings about the conformity test itself, like the limits or the measurement duration.

All information that is necessary to perform the measurement data evaluation is displayed on the "Test Parameters" tab. Depending on the selected standard, corresponding data in the EUT section of the test parameters tab must be filled in. Most of these settings can be changed afterwards.

The fastest way to open this dialog is to click on the "Setup" button in the Quick Access Bar. If information is missing, LMG Test Suite will open this dialog before starting a measurement.

Additional information may be entered about your EUT, your laboratory, the measurement setup, and much more in the other tabs of this dialog. All information given here will be included in the test report after the conformity testing procedure. Depending on the kind of measurement some of this information may be directly required by the standard. The relevant copy of the standard text should be followed closely to be sure to not miss some formal requirements for the test report!

The "Ok" button confirms the entered information.

This dialog can be opened any time later to change the settings. Settings that cannot be changed afterwards will be greyed out. Depending on the changed values a new test run might be performed. This is needed if settings have been changed, that may directly affect the test result.

LMG Configuration

The LMG Test Suite uses the connection established by the LMG600 Control software. The LMG configuration dialog (see fig. 8.17) shows information about the existing connection and about the automatically configured settings which match the chosen standard. In addition it is possible to set the group and the ranges for the measurement.

The fastest way to open this dialog is to click on the "LMG" button in the Quick Access Bar. If information is missing, LMG Test Suite will open this dialog before starting a measurement.
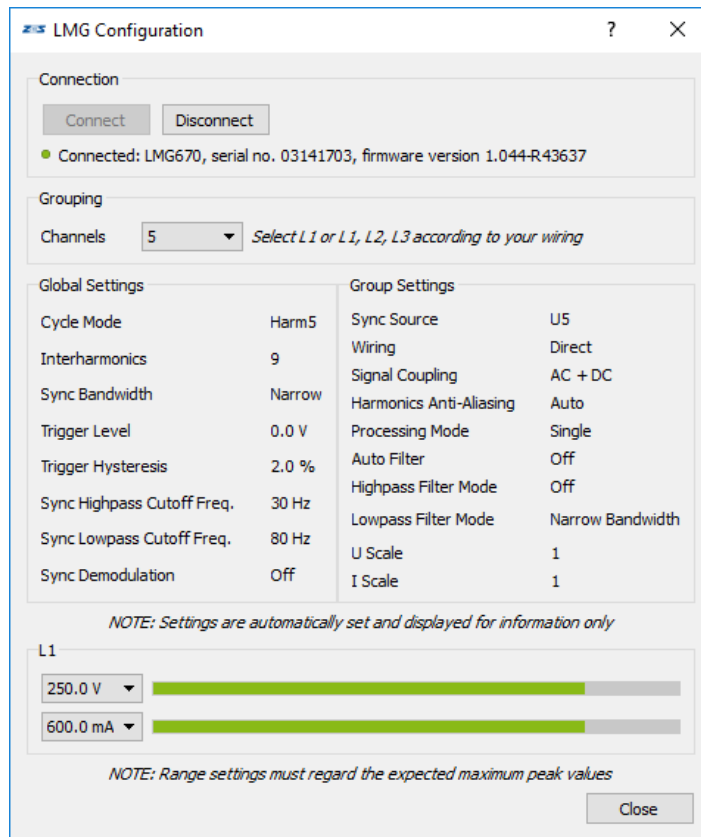
Figure 8.17: LMG Configuration dialog

If no connection has been established by now, a press on the button "Connect" will switch to LMG600 Control to allow to connect to an LMG.

After establishing a connection, LMG Test Suite will automatically configure most settings like synchronization and filtering of signals. As different settings will directly lead to different measurement values and therefore a possibly different test result, these settings are not changeable by the user. However all forced settings are displayed for information.

The LMG has to be used in manual range mode to comply to the supported standards. LMG Test Suite automatically switches to this setting after the connection is established. One may change the chosen voltage and current ranges at the bottom of the dialog.

It is important to regard the highest expected range usage throughout the whole test procedure. An insufficient range setting might spoil the complete test.

Pressing "Ok" after finishing the LMG configuration will proceed with the measurement.

LiveView

While LMG Test Suite is recording measurement data from your LMG, the LiveView dialog fig. 8.18 is displayed. Depending on the standard, the LiveView may show the root mean square values of the current and the corresponding harmonic currents during harmonics measurements. The remaining test time is displayed at the lower right of the dialog. For flicker measurements, the flicker interval progress is visualized by one pie chart per interval, also showing the progress of the current interval.

The recording can be stopped by pressing the "Abort" button. In this case, the data is evaluated only over the recording time. This can be useful if different settings are being tested and waiting for the whole testing period would be too long.

After the recording of data is finished, the LiveView can be closed to proceed with the test analysis and the generation of the test report.
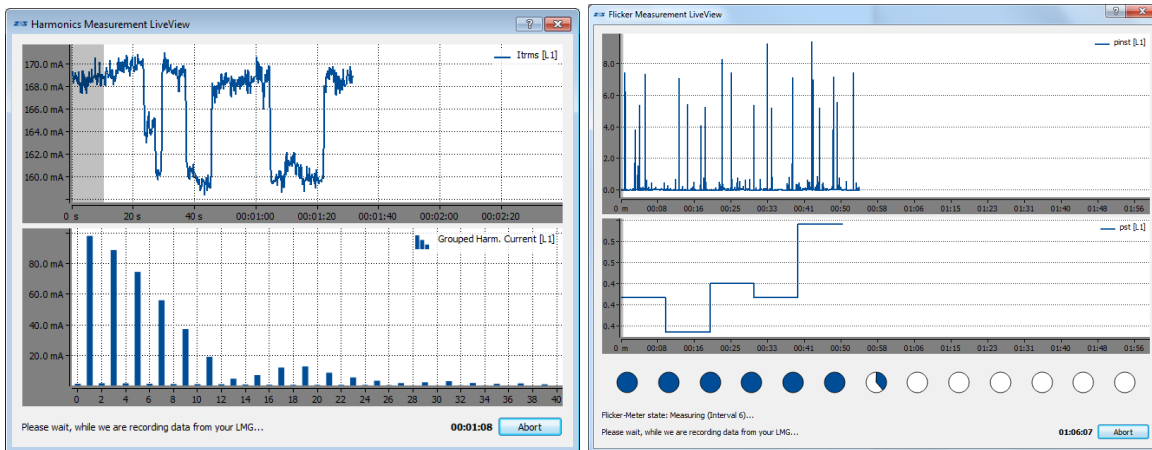
(a) Harmonics

(b) Flicker

Figure 8.18: LiveView dialogs

Test Result Tree

After finishing the measurement, the test result is diplayed on the left side of the main window (see fig. 8.15). The whole conformity test is split up into a number of subtests which represent the different criteria required by the selected standard.

Depending on the kind of conformity test, there are different ways for an EUT to pass. Hence, it is possible for the whole conformity test to pass, even if not all sub-tests did pass.

Each item of the Test Result Tree matches one requirement of the standard. Besides the individual name, the reached maximum percentage of a given limit is displayed. Also the result of the subtest is displayed along with a short description about the source of a fail, if possible.

A double-click on a test gives further information in a Test Analysis Tab on the right side of the main window (see section 8.4.2).

Test Analysis

On the right side of the main window several Test Analysis Tabs can be displayed (see fig. 8.15 and fig. 8.15). Each of these tabs contains detailed information about one of the tests shown in the Test Result Tree described in section 8.4.2. Depending on the type of test, different items may be shown:

- Test result: The top of each Test Analysis Tab states the outcome of this test together with a short description.

- Plot: If a measurement value was tested, a plot of this value will be shown together with the applicable limit(s). The plots can be navigated using the "zoom" and "pan" tools. All plots in the Analysis Tabs are included in the docx and pdf report – exactly as shown here.

- Table: If applicable, a table with the measured values is displayed. These values correspond to the plots shown above the table. The values in the table can be selected and copied by pressing "Ctrl + C" or by right-clicking on the table. The copied values can then be pasted into a spreadsheet application or text editor. If further analysis shall be done on the data, the export function (see section 8.4.2) allows to work on the data using a spreadsheet application.

- Info Text: This will contain a short information about the test like references to the correspondig sections of the underlying standard. Also LMG Test Suite will give hints on the meaning of the test result and whether this result affects the overall pass / fail decision.

In the upper left corner of a plot, a toolbar provides some tools to zoom and move the plot. This makes it possible to examine the data and to generate a view of an area of interest for the test report. This view will show up in the report afterwards.
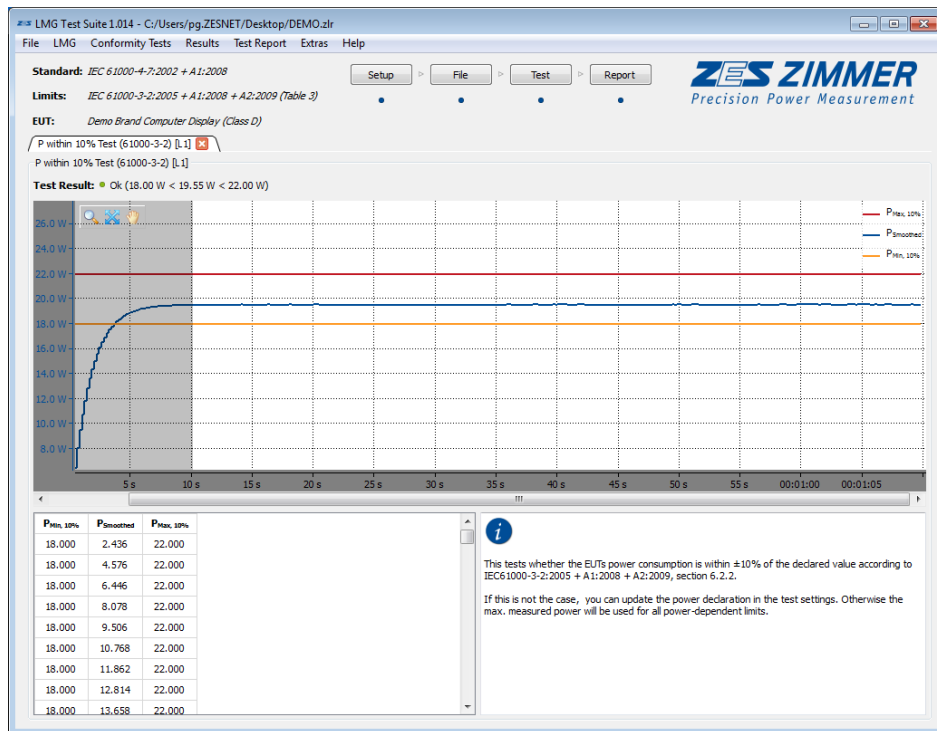
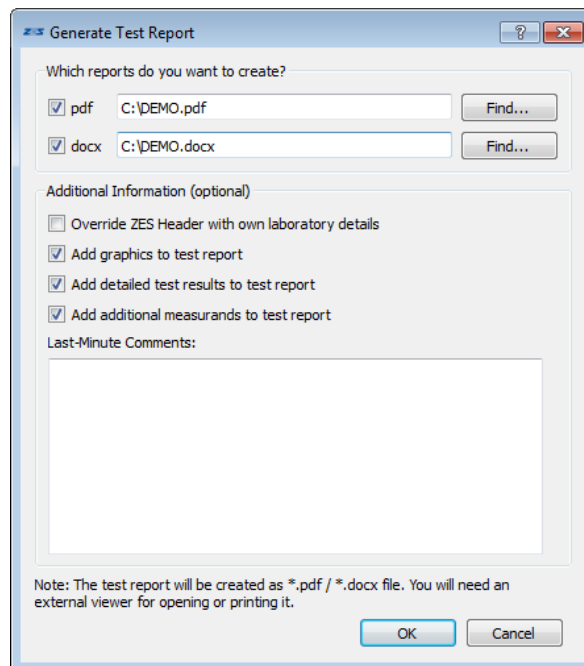Figure 8.19: Screenshot of Test Analysis

Test Report

Figure 8.20: Report generation dialog

When a test is finished, a test report can be created by pressing the "Report" button in the Quick Access Bar. The ZES document header may be overridden with the earlier entered information if the respective box is checked in the "Generate Test Report" dialog (see fig. 8.20). The report will then include the laboratory information entered in the test parameters menu as described in section 8.4.2. Other optional contents for the test report are the analysis graphics and the detailed test result table. Furthermore a last-minute comment might be added to the first page of the test report. This can be useful to add instructions or comments about the test result.

Data Export

All recorded data can be exported as a comma-separated values (CSV) file for custom analysis. "Export Data" in the "File" menu allows to choose the target file name. After selecting a location for the target file LMG Test Suite will export all recorded data to a text file that can be directly opened or imported by any common spreadsheet application or text editor.

8.4.3 Conformity Testing

After starting LMG Test Suite the fastest way to start a conformity test is to press on the "Test" button in the Quick Access Bar at the top of the main screen. LMG Test Suite will ask for all required information before starting the actual measurement. Most of the test settings may be changed afterwards, so there is no need to worry about detailed descriptions of the measurement setup at this stage of the process.

Wiring the test setup

To avoid any interference with other measurements you should disconnect any signal cables from your LMG before starting to wire your test setup.

More information on how to wire the EUT can be found in chapter 6.2. Also, possible normative requirements concerning the wiring must be regarded.

It is possible to connect a ZES ZIMMER reference impedance of type NI2415 , using a cable L6-K-NI2415, with the sync jack of the LMG. This enables the software to control the reference impedance depending on the used standard.

Choosing the required tests

In the test parameters menu, the needed tests for your EUT must be selected. Usually, it is sufficient to set the type of conformity test (like Harmonics or Flicker). The specific limits may be changed after the measurement. For details, see section 8.4.2.

Performing the measurement

After LMG Test Suite has collected all needed information to perform the conformity test, the LiveView will be displayed, showing some informative signals during the measurement.

Due to the nature of some tests defined in the supported standards, it is impossible to show a reliable forecast of the test result before the measurement is completed. It is necessary to wait for the measurement and possible second test runs to finish.

Analyzing the test results

On the left side of the main page LMG Test Suite shows detailed information about the test results. The tests are shown in a tree view including tests and subtests. More information on this feature can be found in section 8.4.2.

A double-click on an entry in this view opens a detailed analysis of this test on the right side of the main window. Depending on the selected test, information about the test result, possible error sources, and more, is presented. Measured signals will be shown graphically together with the corresponding limits, if available.

Generating a test report

This is the last step in the testing procedure. In most applications a written test report is needed to document the results of the performed conformity test. Pressing the "Report" button in the Quick Access Bar opens a dialog which allows to generate a detailed test report in the formats PDF, DOCX, or CSV. This file will include all information provided about the EUT and the measurement setup as well as the test parameters and the test outcome.

If plots shall be included in the test report, the corresponding test analysis tabs must be opened in the main window. All plots shown there will be included in the report. This feature will be especially useful to document failed conformity tests.

8.4.4 System Requirements

To conveniently use all features of the software a contemporary PC system that meets the following minimum requirements is recommended:

- Operating system: Windows 7/8/10 (32/64 bit)

- Hard drive: software: min. 200 MB, data: approx. 20 MB per minute of measurement/phase

- RAM memory: min. 2 GB

- Processor: min. 2 GHz, dual-core

- Interfaces supported: Gbit-Ethernet

To control the reference impendance network NI2415, the cable L6-K-NI2415 is needed.

LMG Test Suite is designed to be used with a Precision Power Analyzer of the LMG600 series from ZES ZIMMER Electronic Systems GmbH. One power channel will be needed for each phase of the EUT.

Following activated LMG software options are needed to run the tests:

CE Harmonics:

- L6-OPT-HRM

- LMG-TEST-CE-HARM

CE Flicker:

- L6-OPT-FLK

- LMG-TEST-CE-FLK

CE Standby:

- L6-OPT-HRM (If the THD test of the power supply shall be included)

- LMG-TEST-CE-STBY

If assistance is needed to configure the LMG, or information about the different types of power channels available, ZES ZIMMER support should be contacted.

For communication with the LMG an ethernet connection (LAN) will be needed. Although an RS232 connection can generally be used to control an LMG from a PC, it will not suit the demands of LMG Test Suite.

We recommend to use this software with a widescreen display for best use of the provided test analysis features.

8.4.5 Frequently Asked Questions

If your questions are not answered here feel free to contact ZES ZIMMER support via support@zes.com or use the contact information provided on page 2 of this document.

File evaluation is slow

If the evaluation of a data file takes too long, please check if you are using a local drive on your computer. Usage of remote folders might lead to a significantly decreased performance.

Also please check if your system meets the requirements listed in section 8.4.4.

Problems during measurement

If you are experiencing problems during the measurement, please check if the LMG setup is correct and test the interface connection as described in your LMG documentation.

Also check if you are using a local drive on your computer.

Problems during LMG connection

If you have problems to connect to your LMG via ethernet, please first check your wiring.

Go to the ''Interface'' tab in the ''Instrument'' menu on your LMG and ensure the displayed IP address matches the one you entered in the LMG Configuration dialog in LMG Test Suite.

If you still cannot connect to your LMG, please consult your system administrator and refer to the descriptions in section 7.3.5.

Test always fails on 3-channel measurements

Please check if your wiring is as follows: L1 is connected to power channel 1, L2 to power channel 2 and L3 to power channel 3, accordingly.

Ensure that the selected wiring in the LMG settings (star / delta and possible transformations) matches your wiring. For details on wiring and the correspondig LMG settings please refer to section 6.2.

9 Remote control

9.1 General

The instrument is built around a measurement and calculation core which provides several communication interfaces. The built-in GRAPHICAL USER INTERFACE (GUI) [7→145] is connected to one of these interfaces. The other interfaces (e.g. LAN, RS232) are available for user applications. Via these interfaces, the core can be configured and all the measurements can be requested.

Due to this system design, the GUI is in principle only a special application software which runs inside the instrument and communicates with the core. Each of the commands the GUI uses to communicate with the core can also be similarly used by customers to program their own application.

9.2 Interface Languages

The LMG600 series understands different languages with which the device can be remotely controlled. Remote control works by connecting to the device using one of the given interfaces (LAN, RS232).

- SCPI (*Standard Commands for Programmable Instruments*) is a language defined by the IEEE and a standard language for measurement devices.

- Short is a ZES ZIMMER specific language for LMG instruments, which uses the same syntax as SCPI, but shorter mnemonic command names.

After a system startup, as well as after a reset of the *interface*, (INTERFACES [9.9→228]) the device is in the SCPI language. One can change the language - from any language - using the command

`*ZLANG language`

*ZLANG
[9.10.17→239]

where *language* can be either scpi or short. For example

`*ZLANG short`

switches to the short language.

The `*ZLANG?` queries the active language. The advantage of *ZLANG versus the official SCPI command is that the query works for all languages. When using LANG, one has to know the current language in order to query it.

LANG [9.10.449→426]

9.2.1 Asterisk Commands (IEEE488.1)

The IEEE488.1 standard defines a series of commands that begin with an asterisk (*). These commands are called star commands. The star commands are available in every language

The syntax of a star command is the same as that of a short command except that it starts with an asterisk, which is considered to be a part of the command name. Besides the commands defined in IEEE488.1, additional star commands have been introduced by ZES ZIMMER . These commands all start with a *Z.

9.2.2 Syntax overview with short commands

There are query and no-query commands. Query commands are used to query measurement values, configuration settings or device states. No-query commands are used to change configuration settings or to trigger actions.

Query commands have the general syntax `<command name><optional suffix>?`:

```
UTRMS?
UTRMS4?
```

The suffix is optional. If it is missing, a suffix of `1` (one) is assumed. The question mark is mandatory and marks the command as a query. Please note that there are *no* spaces allowed *before* the suffix and before the question mark.

Some commands expect an additional list parameter (see also LIST SYNTAX [9.3.1→221]):

```
BUAM? (1:3)
```

A non-query command does not have a question mark. The general syntax is `<command name><optional suffix> <parameter list>`:

```
COMBD 9600
```

The previous command sets the baud rate of the serial interface to 9600. Note that there is *no* space allowed *before* the suffix. However, a space in front of the parameter(s) is *mandatory*.

Some commands do not have a parameter.

```
*CLS
```

While other commands have more than one parameter (which are separated by commas).

```
CONT ON,"foo"
```

Commands are case insensitive: `UTRMS`, `utrms` and `UtRmS` both refer to the same command.

A line may contain several commands separated by a semicolon.

```
UTRMS? ; UTRMS2?;UTRMS3?
```

Spaces before or after a semicolon (like before `UTRMS2?` in the previous example) are ignored.

9.2.3 SCPI-Command Names

SCPI uses the same syntax as short. However, the command names have a different structure. The name is composed of several parts separated by a colon:

```
:FETCH:SCALAR:VOLTAGE:TRMS?
```

This command can be abbreviated:

```
:FETC:SCAL:VOLT:TRMS?
```

In the command reference (see UTRMS [9.10.217→319]) in this manual, the mandatory parts of the name are written in capital letters while the optional parts are written in small letters (e.g. `:FETCh:SCALar:VOLTage:TRMS`). For some commands, whole parts of the name can be skipped. Thus the following commands

```
:FETCH:SCALAR:VOLT?
:FETCH:SCAL:TRMS?
:FETCH:VOLT:TRMS?
:FETCH:VOLT?
:FETCH?
```

all refer to the same command. In the command reference, the optional name parts are written in brackets: `:FETCh[:SCALar][:VOLTage][:TRMS]`.

If there are more than one command in a line, the beginning ("path") of a second command can be left out, if it is identical to the previous command (in the same line!):

```
:FETCH:VOLT:TRMS?;RECT?;PPEAK?
```

is identical to

```
:FETCH:VOLT:TRMS?;:FETCH:VOLT:RECT?;:FETCH:VOLT:PPEAK?
```

A leading colon before a command indicates that the command includes the complete path. Therefore, *no* leading colon allowed before `RECT?` and `PPEAK?`.

However, the following command

```
:FETCH:VOLT:TRMS?; :FETCH:CURRENT:TRMS?
```

cannot be abbreviated by

```
:FETCH:VOLT:TRMS?; CURRENT:TRMS?
```

The latter one would be interpreted as `:FETCH:VOLT:CURRENT:TRMS?`, which does not exist.
The leading colon of the first command of a line is optional as the complete path must always be specified there.

SCPI commands might have optional suffixes as well. These follow immediately the command name.

```
:FETCH:VOLT:TRMS4?
```

9.2.4 Suffixes

For some commands, a suffix is expected. If the suffix is omitted, a suffix of *1 (one)* is assumed. The meaning of the suffix depends on the command. It can for example be the channel number (physical or logical), the group number, the scope number or the PSI channel.

n/a No suffix must be specified

  p Specified the number of the power measurement channel

  l A channel number according to the GCBT system, see Logical Suffixes [9.2.4→218]

  g Specifies a group number

psi Specifies a PSI card

  ai Specifies the number of the analogue input of PSI option

afi Specifies the number of the fast analogue input of PSI option

ao Specifies the number of the analogue output of PSI option

di Specifies the number of the digital input of PSI option

do Specifies the number of the digital output of PSI option

fi Specifies the number of the frequency input of PSI option

motor Specifies the number of a motor channel of PSI option

fl Specifies the number of the flicker channel

gpio Specifies the number of the general purpose IO

can Specifies the number of the CAN message slot

Physical channel numbers (1,2...7) indicate the channels in the order they are inserted into the instrument.

Logical channel numbers (used only for measurement values not for settings) are composed of 4 digits. They are explained in detail below.

Group suffixes are enumerated by 1,2,...7.

For measurement and configuration commands concerning scopes, the suffix describes the scope number, see SPVAL [9.10.180→305]. Possible scope suffixes are

   1 First scope with 8 traces and 2048 samples per trace.

   2 The second scope with 8 traces and 2048 samples per trace.

The scopes 21 and 22 are reserved for the GUI, and should *not* be used directly.

Logical Suffixes

As described in BASIC CONCEPTS [6→101] in this manual, you have several possibilities to calculate values in parallel such as narrow and wide bandwidth values or also star and delta values. Additionally, there are several channels organized in different groups which also have sum values.

The SCPI standard was *not* written for instruments containing such complexity. It introduced the suffix concept which briefly states that the number of the desired channel can be directly appended to the command. FOO2 would apply command FOO to the second channel. For additional details, please see the exact syntax description in SYNTAX OVERVIEW WITH SHORT COMMANDS [9.2.2→216].

This basic suffix concept was extended by ZES ZIMMER to get access to all the various calculated values. In addition to the simple numbers for each measurement channel (which is a physical channel), we have introduced a 4-digit suffix which specifies exactly which value you want to access.

GCBT

   G The first digit represents the number of the group, starting with '1'.

   C The second digit stands for the phase or channel *inside the group*.

   B The third digit represents the bandwidth:
      '1' for narrow bandwidth,
      '2' for wide bandwidth,
      '3' for fundamental bandwidth.

   T The fourth digit stands for the transformation of values:
      '0' No transformation, only allowed with sum values, not with directly measured values
      '1' No transformation, only allowed with directly measured values, not with sum values
      '2' The values in star circuit (if available and option STAR-DELTA CONVERSION, OPTION L6-OPT-SDC [6.3→106] is installed)
      '3' The values in delta circuit (if available and option STAR-DELTA CONVERSION, OPTION L6-OPT-SDC [6.3→106] is installed))

In addition to the above, the following conventions have been used:

- Channel 0 stands for the sum values of a group.

- For suffixes specified with just 1-digit, you get always the wide bandwidth values of the channel.

- If a requested values is not available, a NaN (Not a Number) instead of a value is provided.

- For values which do not have a narrow/wide bandwidth difference (like harmonics or when working with single processing), you can use any of this values and you will get the same results in both cases.

Suffixes of groups with 3 channels

Let us assume, we configured the instrument to have the measuring channels 4, 5 and 6 inside the second group. Then, the following suffixes are valid (for this group):

- 4, 5, 6
  The *wide bandwidth* values of the measuring channels 4, 5 and 6

- 2010
  The narrow bandwidth sum values of group 2

- 2111 , 2211, 2311
  The untransformed narrow bandwidth values of the phases 1, 2 and 3 of group 2 (which are the narrow bandwidth values of measuring channel 4, 5 and 6)

- 2112, 2212, 2312
  The star transformation narrow bandwidth values of the phases 1, 2 and 3 of group 2

- 2113, 2213, 2313
  The delta transformation narrow bandwidth values of the phases 1, 2 and 3 of group 2

- 2020
  The wide bandwidth sum values of group 2

- 2121, 2221, 2321
  The untransformed wide bandwidth values of the phases 1, 2 and 3 of group 2 (which are the wide bandwidth values of measuring channel 4, 5 and 6)

- 2122, 2222, 2322
  The star transformation wide bandwidth values of the phases 1, 2 and 3 of group 2

- 2123, 2223, 2323
  The delta transformation wide bandwidth values of the phases 1, 2 and 3 of group 2

This is the general scheme for any 3-phase system. Thus, it is a very simple and most importantly a logical way to describe the desired measurement values. By using phase numbers instead of channel numbers, one can easily address the channels within the group.

Suffixes of groups with 2 channels

Groups with 2 channels in an Aron circuit have similar suffixes: Let us assume that we configured the instrument to have the measuring channels 4 and 5 inside the third group. Then following suffixes are valid (for this group):

- 4, 5
  The *wide bandwidth* values of the measuring channels 4 and 5

- 3010
  The narrow bandwidth sum values of group 3

- 3111 , 3211, 3311
  The untransformed narrow bandwidth values of the phases 1, 2 and 3 of group 3 (which are the narrow bandwidth values of measuring channel 4 and 5 as well as the calculated 3rd channel)

- 3112, 3212, 3312
  The star transformation narrow bandwidth values of the phases 1, 2 and 3 of group 3

- 3113, 3213, 3313
  The delta transformation narrow bandwidth values of the phases 1, 2 and 3 of group 3

- 3020
  The wide bandwidth sum values of group 3

- 3121, 3221, 3321
  The untransformed wide bandwidth values of the phases 1, 2 and 3 of group 3 (which are the wide bandwidth values of measuring channel 4 and 5 as well as the calculated 3rd channel)

- 3122, 3222, 3322
  The star transformation wide bandwidth values of the phases 1, 2 and 3 of group 3

- 3123, 3223, 3323
  The delta transformation wide bandwidth values of the phases 1, 2 and 3 of group 3

Suffixes of groups with 1, 4, 5, 6 or 7 channels

A star-to-delta transformation does not exist for groups containing 1, 4, 5, 6 or 7 channels. For such groups, there are just a few valid suffixes. For example, the following suffixes are valid for group 1 (where x represents the channel number(s) and y the phase number(s)).

- x
  The *wide bandwidth* values of the measuring channel x

- 1010
  The narrow bandwidth sum values of group 1

- 1y11
  The untransformed narrow bandwidth values of the phase y (which are the narrow bandwidth values of a measuring channel x)

- 1020
  The wide bandwidth sum values of group 1

- 1y21
  The untransformed wide bandwidth values of the phases y

Suffixes of PSI cards

The PSI cards are installed beginning at the highest channel slot (7/4), the numbering is done likewise: The first PSI card in slot 7/4 is assigned the number 1, a second card in slot 6/3, if any, is assigned the number 2. As is usual the suffix determines which card is addressed; the settings and measured values otherwise behave identical for all PSI cards.

The settings and measured values of a PSI card are numbered consecutively starting at 1, e.g. analog inputs 1–8. The associated suffixes are given by the number of the PSI card and the number of the desired setting or measured value, respectively, as follows:

$$PSI\ suffix = 100 * card\ number + setting/measured\ value\ number$$

The suffix of the fourth analog input of the second PSI card for instance is 204. For reasons of backwards compatibility PSI card number 1 can also be addressed using card number 0. This makes it possible to continue to use the settings/measured values' numbers directly as a suffix for addressing the first PSI card. So suffix 4 and 104 for instance both denote the fourth analog input of the first PSI card.

9.3 Input format

The commands can have parameters of different type. The format of these parameters is described below.

The default input format is ASCII. To be precise, the input is in UTF-8 encoded, also if the usage of non-ASCII characters is rare.

- Boolean (`<Boolean>`) are expected as decimal numbers 0 resp. 1.

- Integers (`<NRi>`) are expected as decimal numbers. Negative numbers are prefixed by a minus sign.

- Float values (`<NRf>`) are expexted as usual float numbers, e.g. (-)xxx.xxx or in scientific style: (-)x.xxxxxxE±yy (e.g. `3.14159E+00`).

- Strings (`<string program data>`) are quoted with double quotes (e.g. `"Hello World"`). Double quotes inside a string are quoted by an extra double quote (e.g. `'My name is '"Frank"'`).

- Dates (`<Date>`) are expected in the Form
  <year>:<month>:<day>D<hour>:<minute>:<second>.<fraction>, e.g.
  `2013:11:04D06:26:17.123456789`.

- Times (`<Time>`) are written as float values (`<NRf>`) in seconds or as `[[<hour>:]<minute>:]<second>.<fraction>`, where each statement in `[]` is optional. Example: `1:20.123456789` stands for 1 minute and 20.123456789 seconds.

9.3.1 List Syntax

Some commands return a list of values. For example, the command `BUAM` [9.10.123→283] without parameter returns 401 values (the DC part and the first 400 harmonics) of the voltage amplitude. It is possible to request only a part of these values using a list parameter (in the command description marked with `<list>`):

```
BUAM? (3)
```

returns the value of the third harmonics.

To specify ranges:

```
BUAM? (1:4)
```

returns the first to the fourth harmonics.

Some configuration settings are lists, too. (e.g.`SCTRAC` [9.10.352→383]). One can query them in the same manner that ones queries the measurement values. Setting these values requires a list parameter.

```
SCTRAC2 (0),"u1111"
```

sets the trace zero of the second scope to "u1111" (the direct measured narrow band voltage of the first channel of the first group).

Setting a range of indices with one value is also possible. E.g:

```
SCTRAC2 (0:7),""
```

This deactivates all scope tracks.

9.4 Output Format

The instrument only provides output when requested. The format of these answers is described below.

9.4.1 ASCII

The default output format is ASCII. To be precise, the output is in UTF-8 encoded. The output of non-ASCII characters is rare.

- Boolean (`<Boolean>`) are printed as decimal numbers 0 resp. 1.

- Integers (`<NRi>`) are printed as decimal numbers. Negative numbers are prefixed by a minus sign.

- Float values (`<NRf>`) are written in scientific style with a precision of 6 digits: (-)x.xxxxxxE±yy (e.g. `3.14159E+00`).

- Infinity is printed as `9.9E+37` and negative infinity as `-9.9E+37`.

- NaNs are represented as `9.91E+37`.

- Strings (`<string program data>`) are quoted with double quotes (e.g. `"Hello World"`). Double quotes inside a string are quoted by an extra double quote (e.g. `'My name is '"Frank"'`).

- Dates (`<Date>`) are printed in the Form
  <year>:<month>:<day>D<hour>:<minute>:<second>.<fraction>,
  e.g. `2013:11:04D06:26:17.123456789`. The fraction part has a resolution of nanoseconds.

- Times (`<Time>`) are written as <second>.<fraction>, e.g. `120.123456789`.

If several values are printed as a result of one query (e.g. the output of lists) the values are separated by commas.

```
BUAM? (1:3)
```

results in something like that:

```
1.21732E+01,1.12873E-01,8.85671E+00
```

If the output is a result of several queries separated by a semicolon, the output is separated by semicolons as well:

```
UTRMS?; ITRMS?   =>  2.30139E+02;1.09134E+01
```

A line is terminated by a line feed character (\n, ASCII 0x0a).

9.4.2 Binary Output

The LMG600 series supports a little endian binary output format.

The 32 bit refers only to float values. Integer values are encoded always as 64-bit signed integers (this is also applies to dates, times and duration information).

- Integer values are encoded as 64-bit signed integer.

- Float values are encoded according to IEEE-754 with single precision (32 bit). NaN and infinity are real NaN and infinity values according to the IEEE standard.

  Most measurement values are floats. By using 32-bit instead of 64-bit precision, significant bandwidth and/or memory capacity is saved. 32-bit precision is completely sufficient to represent all measurement values.

- Dates are encoded as 64-bit signed integers, representing the nanoseconds since 1970 (January, 1st 1970 0:0 GMT). Dividing these values by 10^9 results in the usual unix time.

- Times are encoded as 64-bit signed integers, representing nanoseconds.

- Strings are prefixed with the string length in byte (64-bit signed integer). The string itself is UTF-8 encoded, not quoted and 0-terminated. The null terminator is included in the string length.

The output of lists (e.g. `BUAM? 1:200`) is prefixed with the list length (i.e. the number of values in the list). The list prefix is a 64 Bit signed integer that is followed by the values in the format as described above. List of tuples have an extra 64 bit signed integer indicating the number of elements within the tuple (e.g.: `ERRALL [9.10.405→407]`).

Finally the binary output is prefixed with a ASCII-Header. This always starts with a number sign (#) followed by one digit (ASCII encoded) which is the length of the following header bytes, which then represents the number (ASCII encoded) of binary data following the header. For example `#3124` means, that 124 bytes of binary data will follow. `#6000124` has the same meaning, the header is followed by 124 bytes of binary data.

The output of one line might be split in several chunks. Each chunk has its own ASCII header. The line then is terminated by a single line feed character (\n, ASCII 0x0a) without the ASCII header. Thus it is necessary to read all chunks up to the line feed character, strip the ASCII headers and concatenate the chunks. That results in the complete answer.

Examples

The following examplary listings describe the binary format.

```
UTRMS? =>
23 36 30 30 30 30 30 34  33 8a aa 43 0a
```

- 23 36 30 30 30 30 30 34 is the ASCII header: "#6000004", which translates into:
  - 0x23 → ASCII pound sign. Introduces the binary header.
  - 0x36 → ASCII character 6 (six). Therefore, the next six characters tell the length of the binary output.
  - 0x30 30 30 30 30 34 → ASCII string 000004. This means, four bytes binary data belong to this header.
- 0x33 8a aa 43 → float value of UTRMS: 341.08
- 0a → Line feed.

For the sake of readability, the description of the ASCII header is omitted in the following examples.

```
UTRMS?;ITRMS? =>
23 36 30 30 30 30 30 38  33 8a aa 43 90 8f 19 3f
0a
```

- 23 36 30 30 30 30 30 38 → ASCII header: "#6000008". 8 bytes of binary data incoming.
- 33 8a aa 43 → float value of UTRMS: 341.08
- 90 8f 19 3f → float value of ITRMS: 0.599847
- 0a → Line feed.

```
BUAM? (0:4) =>
23 36 30 30 30 30 32 38  05 00 00 00 00 00 00 00
33 a4 36 3e 3c 33 50 43  e1 d7 1c 3e b6 d0 2e 40
5d a2 2d 3d 0a
```

- 23 36 30 30 30 30 32 38 → ASCII header: "#6000028". 28 bytes of binary data incoming.
- 05 00 00 00 00 00 00 00 → List header: This list contains 5 elements.
- 33 a4 36 3e → Result of BUAM? (0): 0.178361
- 3c 33 50 43 → Result of BUAM? (1): 208.2
- e1 d7 1c 3e → Result of BUAM? (2): 0.153167
- b6 d0 2e 40 → Result of BUAM? (3): 2.73149
- 5d a2 2d 3d → Result of BUAM? (4): 0.0423912
- 0a: Line feed

```
errall? =>
23 36 30 30 30 30 34 31   01 00 00 00 00 00 00 00
02 00 00 00 00 00 00 00   00 00 00 00 00 00 00 00
09 00 00 00 00 00 00 00   4e 6f 20 65 72 72 6f 72
00 0a
```

- 23 36 30 30 30 30 34 31 → ASCII header: #6000041. 41 bytes of binary data incoming.

- 01 00 00 00 00 00 00 00 → List header: This list contains one element.

- 02 00 00 00 00 00 00 00 → Tuple header: This tuple contains two elements.

- 00 00 00 00 00 00 00 00 → <NrI>-portion from the error message. Error code: 0.

- 09 00 00 00 00 00 00 00 → <String>-portion from the error message contains 9 bytes.

- 4e 6f 20 65 72 72 6f 72 00 → <String>:"No error" (incl. 0x00 byte at the end).

- 0a: Line feed

9.5 Timing Behavior

9.5.1 Parallel vs. sequential configuration

If several values are queried, the output occurs *always* in the same order as they are requested:

```
UTMRS?;ITRMS?
```

first returns UTRMS and then ITRMS.

The behavior is different when values are set. They are set in parallel and thus in arbitrary order:

```
USCA 1;ISCA 2;USCA 2
```

This sets ISCA to 2. USCA is set to 1 and to 2 in the same command line. However the order of configuration is not defined, and thus the result is undefined, too. USCA might have the value of 1 or 2.

The order can be influenced by inserting the command *OPC [9.10.7→236]: This can only be executed, if all commands preceding it have been executed completely.

```
USCA 1;ISCA 2;*OPC;USCA 2
```

This guarantees that USCA is set to 2 *after* the first two commands are finished.

At the end of each line an implicit *OPC is executed. The above result can thus also be achieved by sending the following two lines:

```
USCA 1;ISCA 2
USCA 2
```

If the same value is set and queried in the same line, the query always returns the *old* value. This, too can be serialized using *OPC.

```
USCA 1;*OPC;USCA 2;USCA?
```

Also if this line sets USCA to 2, it returns a 1! To read the value of 2 correctly, the query has to be done in a second line.

> *The advantage of parallel configuration is the speed. For example setting ranges can take some time. If ranges for several channels are set sequentially, the time is summed up. If they are sent in a single line, the configuration is in parallel and therefore much faster.*

*OPC?

`*OPC [9.10.7→236]` (operation complete) waits for all commands to be finished before continuing with other commands. `*OPC?` does the same but, after all commands have been finished a 1 (a single ASCII character 0x31) is printed. This way one gets a feedback, when the execution has been finished.

ESR Register

A `*OPC` after being finished sets the bit 0 in the ESR register (`*ESR [9.10.4→235]`). A `*OPC?` as well as the implicit OPC at the end of a line does not set this bit.

*ACK?

`*ACK?` prints (similarly to `*OPC?`) a 1 but it does not wait. `*ACK [9.10.1→234]`

9.6 INIM, :READ and :FETCh

`INIM [9.10.221→321]` waits for the actual measurement cycle to finish before updating the local buffer from which the measurement values are read.

```
INIM;UTRMS?;INIM;ITRMS?
```

reads UTRMS and ITRMS values from two different but consecutive cycles. Note that these values were not measured over the same interval! This is identical to

```
INIM;UTRMS?
INIM;ITRMS?
```

```
INIM;UTRMS?;ITRMS?
```

reads UTRMS and ITRMS values from the same cycle. Those values were measured exactly over the same interval! This is identical to

```
INIM;UTRMS?
ITRMS?
```

In SCPI there are two possible ways to get values: Either via the `:READ` or the `:FETCh` path of the tree. The short commands are equal to the `:FETCh` commands (which means there is no INIM performed!). The `:READ` path always has an implicit INIM included. So if you want to perform the last example with SCPI commands you have to enter:

```
:READ:VOLTAGE:TRMS?
:FETC:CURRENT:TRMS?
```

The example

```
:READ:VOLTAGE:TRMS?;:READ:CURRENT:TRMS?
```

is equal to

```
INIM;UTRMS?;INIM;ITRMS?
```

The INIM command takes an optional parameter which is used to specify specify on which event the INIM shall wait for before updating the buffer. Possible events to wait for are:

"now" Does not wait, but updates the buffer immediately with the most recent available values.

"mc" Waits for the next measurement cycle. That is the default behavior, if no parameter is given.

"scope.x" Waits that scope x is triggered and thus can be accessed.

The INIM command blocks the used interface until the measurement buffer is updated. Given certain circumstances, this can take a long time (e.g. the measurement cycle is set to a minute). Given a trigger on an invalid group, the inim command blocks ad infinitum. In these cases, the interface can be reset by sending a break.

9.7 Cont On

The instrument allows values to be outputted continuously. Therefore one must give the action command to specify what shall be the output. For example

```
ACTN;UTRMS?;ITRMS?;UTRMS2?;ITRMS2?
```

defines to output U_{trms} and I_{trms} of channel 1 and 2 after each measurement cycle. The next step is to activate the Cont On. This is done using the command

```
CONT ON
```

The continuous output can be switched off using the command `CONT OFF`.

It is possible to define more than one Cont-On at a time. Therefore the ACTN command takes an optional parameter, the name of the Cont On.

```
ACTN ''chan1'';UTRMS?;ITRMS?
ACTN ''chan2'';UTRMS2?;UTRMS2?
```

The command

```
CONT ON,''chan1''
```

then switches on the desired Cont On. If the name is omitted, the implicit name "cont" is used.

The CONT OFF command switches off *all* running Cont Ons independent of their names. Due to buffer effects there can be several regular answers after this command. After the last buffer a "1" is returned to indicate this last buffer.

The Cont On command has a third, optional parameter. This specifies on which event the Cont On shall trigger. For example

```
CONT ON,''cont'',''scope.1''
```

triggers every time scope 1 is finished. Apart of "now" and "wai" any event can be used that is accepted by the INIM [9.10.221→321] command. See INIM, :READ AND :FETCH [9.6→225].

9.7.1 Timing Behavior

One should note that depending on the configuration settings and the signal frequency, the harmonics are calculated faster or slower than the measurement cycle, and thus the normal measurement values.

Therefore, configuring a Cont On to trigger whenever the harmonic analysis is completed can result in normal measurement values being skipped or doubled (depending on which is faster). The same also applies when one triggers on measurement cycle (mc) event, a scope events, etc.

One should always trigger on the event which corresponds to the measurement values which are most important for the desired measurement; see PARALLEL CALCULATIONS [6.7→113].

9.8 Example dialogue

The following example dialogue shows what one can send via the interface, the expected answer and some comments about it. Some results may vary when reproduced on other machines, like measurement or configuration values. These examples work on a freshly powered-on device which had no commands received earlier. We recommend using the RS-232 interface for the first steps in talking to the device, but the commands work all the same on the LAN interface.

Two comments about the „Sent"-column: A `` denotes a space character, the '↩' marks a <cr> (the carriage return character), which usually is the enter key in most terminal programs.

No.	Sent	Received	Comments
1	`*rst↩`		Resets the instrument to it's default values. After this first command the instrument switches to the remote state.
2	`*idn?↩`	"ZES ZIMMER Electronic Systems GmbH, LMG [...]"	The instrument provides it's identification.
3.a	`:sense:grouping 1,1,1↩`		Given an instrument with three channels, this will put each channel in a separate group (GROUP [9.10.286→351]).
3.b	`:sense:grouping 3,1,1,1,1↩`		Given an instrument with seven channels, this will put the first three channels in one group and the rest in separate groups.
4	`:read:volt:trms[*n*] ↩`	220.345	Reads the channel's voltage (UTRMS [9.10.217→319]) according to the suffix code (n=1111, 2111, etc. siehe LOGICAL SUFFIXES [9.2.4→218]). Because of `read`, this will wait until the current measurement cycle is complete, refreshes the interface's measurement buffer and provides the fresh measurement value afterwards. (see INIM, :READ AND :FETCH [9.6→225]).
5	`:fetc:curr:trms[*n*] ↩`	0.5354	Reads the channel's current (ITRMS [9.10.40→250]) according to the suffix code (n=1111, 2111, etc.). Due to `:fetc`, the measurement buffer will not be refreshed.
6	`:fetc:curr:pow[*n*] ↩`	115.610	Reads the channel's power (P [9.10.132→287]) according to the suffix code (n=1111, 2111, etc.). Due to `:fetc`, the measurement buffer will not be refreshed.
7	`:fetc:volt:dc[*n*] ↩`	0.144	Reads the DC of U (UDC [9.10.210→316]) according to the suffix code (n=1111, 2111, etc.). Due to `:fetc`, the measurement buffer will not be refreshed.
8	`:fetc:volt:ppe[*n*] ↩`	629.335	Reads the peak-to-peak value of U (UPP [9.10.214→318]) according to the suffix code (n=1111, 2111, etc.). Due to `:fetc`, the measurement buffer will not be refreshed.

9	`:fetc:volt:pp ↩`		Intentional errorneous input.
10	`:syst:err:all? ↩`	"-100,"Command error;Error in column 12 at: [...] fetc:volt:pp: Unknown SCPI path;"	Provides an error description of **all** errors occured since the last call.
11	`gtl ↩`		Switches back from remote to local control, so the instrument can be used from the GUI.

You can have the same dialogue using the SHORT language, having shorter (hence the name) commands.

9.9 Interfaces

As the LMG671 ATE is not equipped with an internal display, the configuration of the interfaces needs to be done with an external monitor and USB input devices.

The LMG has several interfaces to access the device. To select the active interface, use INTSELECT [9.10.448→426].

Every time the instrument starts, each interface has the following initial properties:

- The language is always set to SCPI.

- The output format is always ASCII.

- The continuous output is always deactivated.

- All in and out buffers are empty and there are no commands for processing.

These properties can be changed during use of the instrument. Therefore, it is recommended that interface be reseted to these default properties. How to reset a specific interface is provided in the interface descriptions below.

9.9.1 LAN Access

This section describes the LAN interface (a gigabit Ethernet interface).

Activating the LAN

In order to use this interface, the device requires an IP address (IPv4). This can be obtained by DHCP, which is the default behavior of the LMG. If there is no DHCP server (e.g. using a point-to-point connection to the PC), the LMG will use an IP address of the address block 169.254.0.0/16 (zeroconf). Otherwise the IP address, the netmask and (if a connection to the internet or beyond the edge of routers is desired) the gateway address can be manually configured. This can be done using the keypad of the device or using another interface, see INTERFACE TAB [7.3.5→178]. To find a DHCP server can take up to 1 min. zeroconf is just performed if no DHCP server could be found.

> *The LMG behaves very passive being connected to the net. That is, it does* not *connect to any other server or automatically receives updates over the internet without being asked to do. See also* SECURITY ADVISES FOR NETWORK USAGE *[5.6→99].*

Once the LAN interface has been configured, one can connect to the device using the TCP port *5025*. Only one connection at a time is allowed. If one tries to establish a second connection, the LMG accepts and immediately closes the connection.

If a connection is terminated and reopened, the interface is in the state as it was before. Cont Ons are still running (unless they were stopped due to buffer overruns). The connection retains the old previous state of properties (e.g. interface language, output formats etc.) as though the connection was never interrupted. Here the behavior of a RS232 interface is emulated, where it is not possible to terminate a connection.

Resetting the Interface

To bring the interface into a defined start state, one must reset it. Using RS232, one would send a Break. A similar approach is also available for the LAN interface. If a connection to the TCP control port *5026* has been established, following commands (terminated by a newline character \n, 0x0a) are available:

- **break** : resets the LAN interface, the connection remains untouched

- **cbreak** : resets the LAN interface, the connection will be closed

- **status** : shows various interface status, only for diagnostic purposes

- **busy** : returns "true" if the device is connected (on port 5025), otherwise returns "false"

- **endpoint** : reports the IP address of the endpoint currently connected to the device

  This this commands apply to the LAN interface only. Other interfaces and the instrument itself are not affected.

break and cbreak return 0 ok after finishing with success. In case an error occurs, the commands return an error code like -8 System Error. After the return code 0 ok has been received the interface has been reset and is ready to use.

9.9.2 RS232 Access

This section describes the RS232 interface.

Activating the RS232

To make use of this interface, a baud rate, a flow control and the echo must be configured in the instrument. This can be done by using either another interface or using the keypad of the device, see INTERFACE TAB [7.3.5→178].

Once the RS232 interface has been configured, one can connect to the device. If a connection is reused without rebooting the device, the interface is in the state as it was before. Cont Ons are still running (unless they were stopped due to buffer overruns). The interface properties such as language and output formats remain in the old state

Resetting the Interface

To bring the interface into a defined state, one must reset it. For the RS232 interface, one has to send a *break* to do that.

  A break *is not a string with these letters but a special signal timing which can usually reached by an API function*
  The reset just applies to the RS232 interface. Other interfaces and the instrument itself are not affected.

9.9.3 CAN Bus

The LMG optionally provides a CAN bus interface conforming to ISO 11898-2 (Highspeed CAN). It supports the CAN standards 2.0A and 2.0B and is therefore able to send and receive CAN messages with 11 or 29 bit CAN-IDs. Bus speeds between 5 kbit/s and 1 Mbit/s are available for the user to set.

The CAN interface is physically provided via a 9-Pin D-SUB connector, as described by the specifications of CiA® 102. There is no internal termination, therefore, the device must be used on a terminated bus.
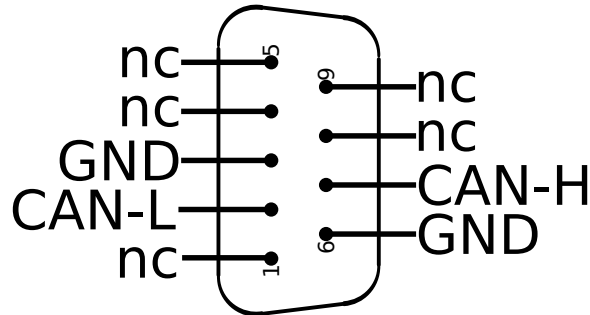
Figure 9.1: CAN Bus interface, 9 Pin D-SUB male face view

The LMG acts as a sensor on the CAN bus, able to provide multiple measurement values on different CAN IDs. Additionally, it is possible to trigger user-defined actions on the LMG via CAN.

Defining the sensor slots
For every measurement value which shall be announced via CAN, one must define a sensor slot. A sensor slot consists of:

- a CAN-ID

- an EFF-bit marker which determines if this message shall be sent in the extened frame format (EFF).

- up to two measurands

As soon as a sensor slot is defined and activated, the LMG will continously send the measurement value according to the given measurement cycle. Additionaly, incoming remote transmission requests (RTR) inquiring about this CAN ID will be answered with the latest measurement value available for this slot.

Please note when setting up a sensor slot:

- The allowed range of CAN IDs depends on whether the EFF bit is set or cleared. The valid range with EFF cleared is 0-2047, with EFF set it's 0-536870911.

- While it is possible to define multiple sensor slots with the same CAN-ID / EFF bit combination, for the sake of discriminability we highly discourage from doing so.

- The data is being transmitted binary, much similar to the binary format used in the SCPI output format (see BINARY OUTPUT [9.4.2→222]), but without a SCPI binary header.

- Setting the measurand is done via canactn

The following example sets up the first sensor slot to publish the current voltage and peak-to-peak value on CAN ID 42, EFF bit cleared:

```
syst:lang short            # Switch to short language
cancontid1 42                  # Set the CAN ID of sensor slot 1 to 42
cancontideff1 0                # Clear the EFF bit of sensor slot 1
canactn1; utrms?; upp?     # Sets up the measurands of sensor slot 1
cancont1 on                    # Activate sensor slot 1
```

Setting the bitrate

The adapter's bitrate must match the defined speed of the connected bus. A different bitrate can result in other bus-participants considering the LMG's messages as errors on the bus, which leads to a general decline of the bus' performance.

To set the bitrate, for example to 500 kBit/s, one can use the following commands:

```
syst:lang short      # Switch to short language
canbit 500000            # Bitrate to 500.000 bit/s
```

Error cases

The LMG sends every measurement values (for the set-up measurand and cycle time) continuously via CAN. But, as on any transport media, errors can happen and our continuous output can be interrupted. The most common reason is overload: The bus-participants try to send more messages than physically possible. This problem can be approached by either reducing the message count (e.g. by adjusting the cycle-time) or upgrading to a higher bus-speed.

If the LMG detects that it can't deliver it's CAN messages, the continuous output will be turned off automatically and an error will be placed in the errorqueue,

Setting the trigger-table

The instrument is able to react on previously defined CAN messages with given actions. The whole concept on what to do on what kind message is defined in a so-called trigger-table. In this table, we define the conditions and resulting actions, column by column. A condition consists of the following elements:

- What messages shall be inspected?

    - Which CAN ID has the incoming message? (CANTRIGID [9.10.428→416])

    - Is the message in extended or simple frame format? (CANTRIGIDEFF [9.10.426→415])

- What is the relevant data of this message?

    - At which bit position starts the relevant data (CANTRIGOFFSET [9.10.430→417])?

    - How many bits long is the relevant data (in bits, CANTRIGBITLEN [9.10.423→414])?

- What condition must be met?

    - Interpret the data as integer or as floating point number? (CANTRIGOPTYPE [9.10.432→418])

    - What comparison operator shall be used? (CANTRIGCOMP [9.10.425→415])

    - What is the incoming number being compared to? (CANTRIGOPINT [9.10.429→417], bzw. CANTRIGOPFLT [9.10.427→416])

Each column of the trigger-table contains a single condition. To make changes on different columns, use the appropiate suffix on each command. For example

```
CANTRIGID1 42
```

changes the first column of the table, whereas

```
CANTRIGID20 62
```

changes the 20th entry. There are 32 entries available.

If an incoming message matches a condition in the table, a command in SHORT syntax (INTERFACE LANGUAGES [9.2→215]) shall be executed. This so called action string must have been defined by CANTRIGACTNSTR [9.10.422→414] beforehand. To make things more interesting, one can use the incoming CAN data as values within that action string. The second example makes use of this.

Example: Setting the cycle time via CAN trigger The following example will make an entry in the trigger-table in the first column. This trigger shall change the cycle time, if an incoming CAN message matches the condition. The following examples uses the SHORT syntax, so make sure the interface language has been switched accordingly before proceeding:

```
*zlang short
```

First, we define that we want to trigger on simple frame format CAN messages with the ID 42.

```
CANTRIGID1 42
CANTRIGIDEFF1 0
```

For this example, we arbitrarily choose that we want to investigate the first 4 bytes of the incoming message. That means, we set the offset to 0 and the bitlength to 32. (Another example: To investigate the first byte, use offset 0, bitlength 8. To investigate the first 4 bit of the second byte, use offset 8, bitlength 4).

```
CANTRIGOFFSET1 0
CANTRIGBITLEN1 32
```

These four bytes shall be interpreted as (unsigned) integer.

```
CANTRIGOPTYPE1 INTEGER
```

Alternative: It is possible to interpret 4 or 8 bytes of data as a floating point number:

```
CANTRIGOPTYPE1 FLOAT
```

If the bitlength is set to 64, the value will be interpreted as double-precision floating point type (*double* for short). We now set that the trigger-condition is met, if the numeric value is greater than 1337. Therefore, we choose the comparison operator:

```
CANTRIGCOMP1 GT
```

Other valid comparison operators are greater-than (GT), greater-or-equal (GE), less-than (LT), less-or-equal (LE), equal (EQ), unequal (NE) and true (TRUE). The last operator means that the condition will be always met. This is useful when we actually don't care about the incoming data. For any other operand, we must choose an comparison operand:

```
CANTRIGOPINT1 1337
```

Alternative: As mentioned above, we can choose to interpret the data as floating point number. You need to set a floating point comparator with a similar command:

```
CANTRIGOPFLT1 1337.0
```

Now, the condition is complete. At last, the action must be defined:

```
CANTRIGACTNSTR1 ''CYCL 0.1''
```

Optional: The command CANTRIGCHECK [9.10.424→414] performs a santity check upon the given entry. If there has been made a mistake, it will show in the error queue.

```
CANTRIGCHECK1
ERRALL?
```

At last, this entry can be activated:

```
CANTRIGSWITCH1 ON
```

Example: Setting the cycle-time via CAN   It is possible to let incoming CAN message decide the instrument's cycle time. In this example, we will make use of the formatting symbol within action strings. At fist, we set up a trigger, that will react on incoming messages on CAN ID 21, which have a float value greater than 0.03.

```
*zlang short
CANTRIGID1 21
CANTRIGIDEFF1 0
CANTRIGOFFSET1 0
CANTRIGBITLEN1 32
CANTRIGOPTYPE1 FLOAT
CANTRIGCOMP1 GT
CANTRIGOPFLT1 0.03
```

Within the action string, instead of a fixed cycle time, we place the formatting symbol:

```
CANTRIGACTNSTR1 ''CYCL %v''
```

Any occurence of the formatting symbol `%v` will be replaced with the currently incoming float value, as defined by CANTRIGOFFSET [9.10.430→417] and CANTRIGBITLEN [9.10.423→414]. Depending upon whether CANTRIGOPTYPE [9.10.432→418] defined the incoming data as float or integer, the formatting symbol will place it as such.

```
CANTRIGSWITCH1 ON
```

Example: Combine/correlate any can data with internal measurement data   The full capacity of CAN-data based command formatting unfolds in using it for correlation and combination of CAN and instrument data. The example above for switching the cycle time can be modified as follows:

```
CANTRIGACTNSTR1 ''SETENV 1,%v''
```

This sets the first float enviroment variable of the measurement subsystem. Integer values can be placed as follows:

```
CANTRIGACTNSTR1 ''SETENVINT 1,%v''
```

These values are accessible with the next complete measurement cycle and can be accessed just like any other internal measurement value. For example:

- They can be queried within cont-ons via ENV1?, resp. ENVINT1?, together with regular measurement values.

- They can be used in the instrument's scripting-system for internal calulations, as trigger for certian actions, etc. (SCRIPTEDITOR [6.15→125])

- They can be logged via the GUI (LOG TAB [7.3.17→189])

- They can be monitored in a custom-built menu in the GUI (CUSTOM MENU [7.2.11→163])

Warning: Incoming external data cannot be guaranteed to be from the same timeframe than our own measurement data. Expect a lag between these sources.

9.10 Remote commands

In the following sections, one can find a description of all available commands. The commands are sorted according to the SCPI tree structure. The layout of the description follows a unique scheme:

- Most commands start with a brief description.

- The different kinds of formal command description follow:

LANG [9.10.449→426]

    – Short command
      The short-form of the command. These short commands are only allowed if one has switched to the short command language!

    – SCPI command
      As the SCPI commands are built in a hierarchical tree structure, the last part of the branch is displayed here in order to provide a a concise representation.

    – Full SCPI command
      Here, the full SCPI branch is provided. For all measurement commands, it is possible to use the FETCh or READ structure. Therefore, both branches are described. They are separated by a logical OR sign (|).

- Display ID
  The ID of the value which is displayed in the GUI.

- Unit
  The physical unit of the value.

- Return value
  The kind of value that is returned (if the command returns a value at all).

- Suffix
  A suffix specifies which value is meant, if there is a choice of several available, see SUFFIXES [9.2.4→217].

- [*RST Default value]
  If this command sets any value, this is the value which is set when the command *RST [9.10.9→237] is given to reset the instrument.

A detailed description of the commands and parameters is given below

9.10.1 *ACK
Acknowledge

Returns a 1. Otherwise does nothing

Query form
Short form: *ACK?
SCPI form: *ACK?

Response
Default return value: Always "1"

Type: <NRi>

9.10.2 *CLS
Clear status

Clears

- the event registers of all state data structures

- the error/event queue.

Command form
Short form: *CLS
SCPI form: *CLS

9.10.3 *ESE

Event status enable register

Used to set up or read out the Event Status Enable register.

Command form
Short form: *ESE␣<NRi>registerValue
SCPI form: *ESE␣<NRi>registerValue

| <NRi>registerValue | The register's value

  Possible range: 0 ... 255

Query form
Short form: *ESE?
SCPI form: *ESE?

Response
The register's value

Type: <NRi>
Possible range: 0 ... 255

9.10.4 *ESR

Event status register

Reads out and clears the Event Status register.

Query form
Short form: *ESR?
SCPI form: *ESR?

Response
The register's value.

Type: <NRi>

Possible values:

bit 0: Operation Complete
bit 1: Request Control
bit 2: Querry Error
bit 3: Device Dependend Error
bit 4: Execution Error
bit 5: Command Error
bit 6: User Request
bit 7: Power On

9.10.5 *IDN

Human-readable instrument indentification

Identifies the instrument by a string of 4 comma-separated fields:

- Manufacturer

- Model

- Serial Number

- Software Version

The output differs from common strings, as it has no quotation marks (ASCII format) or length information (binary format).

Query form
Short form: *IDN?
SCPI form: *IDN?

Response
IDN return string, as described above

Type: <string program data>

9.10.6 *IST
Individual status query

This returns the status of the 'ist' local message in the device.

Query form
Short form: *IST?
SCPI form: *IST?

Response
'ist' local message

Type: <Boolean>

9.10.7 *OPC
Operation complete

Waits for all operations to be completed and causes the device to set the operation complete bit in the Standard Event Status Register, when all pending selected device operations have been finished. Returns a '1' when finished. The output differs from common strings, as it has no quotation marks (ASCII format) or length information (binary format).

Command form
Short form: *OPC
SCPI form: *OPC

Query form
Short form: *OPC?
SCPI form: *OPC?

Response
returns a '1' when done

Type: <string program data>

9.10.8 *PRE
Parallel poll register enable

Set or read the Parallel Poll Enable Register.

Command form

Short form: *PRE␣<NRi>registerValue
SCPI form: *PRE␣<NRi>registerValue

$\boxed{\text{<NRi>registerValue}}$ The register's value

Possible range: 0 ... 65535

Query form

Short form: *PRE?
SCPI form: *PRE?

Response

The register's value

Type: <NRi>
Possible range: 0 ... 65535

9.10.9 *RST
Reset

This performs a device reset. The internal settings of the instrument itself (like ranges, synchronization, ...) are reset to default values, but *not* the interface! These default values can be found under [*RST default value] within the command description. All time dependant measurements like energy are stopped. The interfaces and their parameters are not reset! If you want to reset them, please use a BREAK for RS232 or a 'device clear' for IEEE488 interface.

Command form

Short form: *RST
SCPI form: *RST

9.10.10 *SRE
Service request enable register

Sets or queries the Service Request Enable register.

Command form

Short form: *SRE␣<NRi>registerValue
SCPI form: *SRE␣<NRi>registerValue

$\boxed{\text{<NRi>registerValue}}$ The register's value

Possible range: 0 ... 255

Query form

Short form: *SRE?
SCPI form: *SRE?

Response

The register's value

Type: <NRi>
Possible range: 0 ... 255

9.10.11 *STB
Status byte register

Sets or queries the Status Byte Register. This command is identical to :STATus:REGister
(STB [9.10.398→404])

Query form
Short form: *STB?
SCPI form: *STB?

Response
The register's value

Type: <NRi>
Possible range: 0 ... 255

9.10.12 *TRG
Trigger

Triggers the same action that happens when a programmer sends DT1 via IEEE 488.1 interface or TRG
via RS232 interface. As for now, no action will be actually triggered.

Command form
Short form: *TRG
SCPI form: *TRG

9.10.13 *TST
Self test

returns 1, otherwise does nothing

Query form
Short form: *TST?
SCPI form: *TST?

Response
selftest exit status

Type: <NRi>

9.10.14 *WAI
Wait All Intructions

Waits for all pending operations to be completed and until an INIM causes to read measurement
values recorded with the new settings.

Command form
Short form: *WAI
SCPI form: *WAI

Query form
Short form: *WAI?
SCPI form: *WAI?

Response

exit status

Type: <string program data>

9.10.15 *ZCONFLD
load config from a file on the device

This command allows you to load a device configuration from a file.

Command form
Short form: *ZCONFLD␣<string program data>filename
SCPI form: *ZCONFLD␣<string program data>filename

| <string program data>filename | Name of the file to load from |

9.10.16 *ZCONFST
store config to a file on the device

This command allows you to save the current device configuration to a file.

Command form
Short form: *ZCONFST␣<string program data>filename
SCPI form: *ZCONFST␣<string program data>filename

| <string program data>filename | Name of the file to store into |

9.10.17 *ZLANG
Interface language setting

Changes the interface language. Available languages are SCPI and Short. The parameter is an unquoted string, which must be one of scpi or short, e.g. *zlang short. This command can be queried, too.

Command form
Short form: *ZLANG␣<string program data>retVal
SCPI form: *ZLANG␣<string program data>retVal

| <string program data>retVal | The interface language |

Query form
Short form: *ZLANG?
SCPI form: *ZLANG?

Response
The interface language

Type: <string program data>

9.10.18 *ZSRST
Interface reset

Resets the interface, and is almost identical in sending a break to the interface. The interface then switches to SCPI language, the format settings are set back to default. Cont-ons are removed from the system.

Command form
Short form: *ZSRST
SCPI form: *ZSRST

9.10.19 GTL
Go to local handling

If a device is remote controlled it is in a remote mode. In this mode it cannot be controlled by the front panel. This command sets the device back to local mode. Thus it can be controlled by the front panel of the device again. This command should be the last one when finishing remote control.

Command form
Short form: GTL
SCPI form: :GTL

9.10.20 LEN
Local enable

Enters a mode, where local handling of the instrument is possible, even if the instrument is remote controlled. Please use this carefully! It can result in unexpected behavior for the program, if a user changes a settings and the program does not know this!

Command form
Short form: LEN
SCPI form: :LEN

9.10.21 AIZERODATE
I channel zero adjustment date

This is the date of the last I channel zero adjustment. The date is usually set by calibration laboratories (e.g. at ZES) or when the instrument completes a zero adjustment requested by the user.

Suffix: p (1 ... 7)

Command form
Short form: AIZERODATE␣<Date>val
SCPI form: :ADJust:CURRent:DATE:ZERO␣<Date>val

| <Date>val | Config Value |

Query form
Short form: AIZERODATE?
SCPI form: :ADJust:CURRent:DATE:ZERO?

Response
Config Value

Type: <Date>

9.10.22 AUZERODATE
U channel zero adjustment date

This is the date of the last U channel zero adjustment. The date is usually set by calibration laboratories (e.g. at ZES) or when the instrument completes a zero adjustment requested by the user.

Suffix: p (1 ... 7)

Command form
Short form: AUZERODATE␣<Date>val
SCPI form: :ADJust:VOLTage:DATE:ZERO␣<Date>val

$\boxed{\text{<Date>val}}$ Config Value

Query form
Short form: AUZERODATE?
SCPI form: :ADJust:VOLTage:DATE:ZERO?

Response
Config Value

Type: <Date>

9.10.23 ZADJACT
Zero adjust

Performs a zero adjustment. The resulting adjustments are lost when the instrument is turned off or restarted unless saved with the 'persist' parameter.

Command form
Short form: ZADJACT␣<NRi>chn, <string program data>jack, <string program data>range,
        <string program data>sensor, <Boolean>noise, <Boolean>persist
SCPI form: :ADJust:ZADJ:ACT␣<NRi>chn, <string program data>jack,
        <string program data>range, <string program data>sensor, <Boolean>noise,
        <Boolean>persist

$\boxed{\text{<NRi>chn}}$ Defines, which channel(s) shall be affected.

Possible values:

bit 0 or Channel1: Physical channel 1
bit 1 or Channel2: Physical channel 2
bit 2 or Channel3: Physical channel 3
bit 3 or Channel4: Physical channel 4
bit 4 or Channel5: Physical channel 5
bit 5 or Channel6: Physical channel 6
bit 6 or Channel7: Physical channel 7

$\boxed{\text{<string program data>jack}}$ Defines, which jacks should be adjusted.

Possible values:

"activeU" or activeU: activeU
"activeI" or activeI: activeI

"activeUI" or activeUI: activeUI

| <string program data>range | Defines, if only the active range or all ranges of the selected jacks shall be adjusted.

Possible values:

"all" or all: all
"active" or active: active

| <string program data>sensor | Defines, if the sensor shall be adjusted. If it should be adjusted, but no sensor is connected, an error occurs. Vice versa, if no sensor should be adjusted, but an sensor is present an error occurs. Warning: If an sensor is adjusted and the channel itself was not adjusted before, the sensor adjustment will include the errors of the channels. This will lead to problems, if the sensor is connected to another channel or even if the range at the same channel is changed! If set to ignore, the complete chain of sensor and channel is adjusted and the value is stored in the channel. This is implemented to be consistent with the LMG500.

| <Boolean>noise | Defines, if the noise shall also be adjusted. Please note that sensors have no noise adjustment

| <Boolean>persist | Whether to make the results of this zero adjustment persistent. When false, the adjustment values are stored in volatile memory only and are lost when the instrument is turned off or restarted. If true, the instrument will first reset all adjustment settings, then perform the zero adjustment, and finally persist the resulting adjustment values.

9.10.24 ZADJIAUTO
I Auto Zero Adjustment

Enable or disable the internal automatic zero adjustment for a group of current (I) channels. (Automatic) internal zero adjustment is available with all S and A3, B3, C3 power channels (though not for I_{sensor} jacks as these are technically voltage inputs). The adjustment status can be viewed with ZADJISTAT [9.10.25→243]. The automatic adjustment runs, if enabled, when certain (partially configurable) events occur *and* the adjustment becomes/is invalid; see ZADJITRIG [9.10.26→243]. To unconditionally perform an internal zero adjustment, even if the current adjustment is still valid, use ZADJNOW [9.10.27→244].

Display ID: I Auto Zero Adjustment
Suffix: g (1 ... 7)

Command form
Short form: ZADJIAUTO␣<Boolean>val
SCPI form: :ADJust:ZADJ:IAUTo␣<Boolean>val

| <Boolean>val | Config Value

*RST value: 1

Query form
Short form: ZADJIAUTO?
SCPI form: :ADJust:ZADJ:IAUTo?

Response
Config Value

Type: <Boolean>

9.10.25 ZADJISTAT
Zero adjustment I status

The status of the internal zero adjustment for a current (I) channel. This setting is only meaningful for S and A3, B3, C3 power channels. Each status bit denotes a possible event that causes the internal zero adjustment for the currently selected jack/range combination to become invalid. If all bits are cleared, the internal zero adjustment is valid. Otherwise, those bits that are set indicate the reason(s) why the adjustment became/is invalid. An *automatic* internal zero adjustment may be configured to renew the adjustment when it becomes invalid; see ᴢᴀᴅᴊɪᴀᴜᴛᴏ [9.10.24→242]. Note that during the warm-up phase of the instrument, no valid internal zero adjustments may be performed in general due to thermal effects! So for the duration of the warm-up phase, the warm-up bit will always be set and cannot be cleared until warm-up has completed. The internal zero adjustment is not available with sensor jacks active.

Display ID: Zero Adjustment I Status
Suffix: p (1 ... 7)

Query form
Short form: ZADJISTAT?
SCPI form: :ADJust:ZADJ:ISTATus?

Response
Zero adjustment I status

Type: <NRi>

Possible values:

bit 0: The instrument is still warming up and no valid internal zero adjustment may be performed.
bit 1: The ambient temperature has significantly diverged from the one at which the adjustment was made.
bit 2: The (fixed) validity period of the adjustment has expired.
bit 63: Internal zero adjustment is not available (only with S and A3, B3, C3 power channels, though not for I_{sensor} jacks as these are technically voltage inputs).

9.10.26 ZADJITRIG
Zero adjustment I trigger

Specifies the trigger events for the automatic internal zero adjustment for a group of current (I) channels. This setting only takes effect if ᴢᴀᴅᴊɪᴀᴜᴛᴏ [9.10.24→242] is enabled for the group. If this is the case *and* the internal zero adjustment becomes invalid (see ᴢᴀᴅᴊɪsᴛᴀᴛ [9.10.25→243]) for one of the channels in the group, the trigger settings determine when an adjustment is actually performed in response to the invalidation. There are fixed and configurable trigger conditions. The fixed trigger conditions comprise a set of configuration settings. Any changes to these settings will (possibly) trigger an automatic internal zero adjustment. More specifically, these configuration settings are:

- ɪᴊᴀᴄᴋ [9.10.239→328]

- ɪʀɴɢ [9.10.252→333]

- ᴘʀᴏᴄ [9.10.275→345]

- ʟᴘꜰɪʟᴛ [9.10.265→340] or ɴʟᴘꜰɪʟᴛ [9.10.271→343]/ᴡʟᴘꜰɪʟᴛ [9.10.277→346], respectively.

In addition to the fixed trigger conditions above, which are always enabled, the ZADJITRIG setting allows the user to configure further optional ones. Note that the condition bits match the bits in ᴢᴀᴅᴊɪsᴛᴀᴛ [9.10.25→243] that indicate the respective event that caused the adjustment to become invalid.

Display ID: Zero Adjustment I Trigger
Suffix: g (1 ... 7)

Command form
Short form: ZADJITRIG␣<NRi>val
SCPI form: :ADJust:ZADJ:ITRigger␣<NRi>val

$\boxed{\text{<NRi>val}}$ Zero adjustment I status

  *RST value: 1

  Possible values:

   bit 0 or WARMup: Warm-up
   bit 1 or TEMPerature: Temperature
   bit 2 or EXPired: Expired

Query form
Short form: ZADJITRIG?
SCPI form: :ADJust:ZADJ:ITRigger?

Response
Zero adjustment I status

Type: <NRi>

Possible values:

bit 0: Warm-up
bit 1: Temperature
bit 2: Expired

9.10.27 ZADJNOW
Zero Adjust Now

Requests an immediate and unconditional internal zero adjustment for all eligible channels across groups. The adjustment will be executed as soon as possible and even if the internal adjustment is already valid. Currently only S and A3, B3, C3 current (I) channels are eligible for internal zero adjustment.

Command form

Short form: ZADJNOW␣[<NRi>val]
SCPI form: :ADJust:ZADJ:NOW␣[<NRi>val]

$\boxed{\text{[<NRi>val]}}$ Config Value

9.10.28 DISB
Display brightness

Sets the front panel display brightness.

Display ID: Brightness

Command form

Short form: DISB␣<NRf>val
SCPI form: :DISPlay:BRIGhtness␣<NRf>val

| <NRf>val | Config Value

      Possible range: 1.0 ... 100.0
      Unit: %

Query form

Short form: DISB?
SCPI form: :DISPlay:BRIGhtness?

Response

Config Value

Type: <NRf>
Possible range: 1.0 ... 100.0
Unit: %

9.10.29 DISOFF
Display backlight control

In recent LMGs the front panel's LCD backlight can be turned off completely to protect the inverter e.g. during remote-only operation. If this functionality is not available, DISOFF is set to a negative value and the setting cannot be changed by the user; in this case the display is always on (but may be dimmed via the GUI). Otherwise, the device starts with a default value of 0 which turns the display *on*. By setting a value of 1 the display can be turned *off*. While the display is off, most LEDs (including the STANDBY LED!) are turned off, too; the BACK LED blinks to indicate that the device is *not* turned off, while others may indicate that certain operations like e.g. logging are still in progress. *Any* keypress or (if enabled) touch turns the display on again, which also switches all LEDs back into normal operation.

Display ID: Display backlight control

Command form

Short form: DISOFF␣<NRi>val
SCPI form: :DISPlay:OFF␣<NRi>val

| <NRi>val | Config Value

Query form

Short form: DISOFF?
SCPI form: :DISPlay:OFF?

Response

Config Value

Type: <NRi>

9.10.30 IAC
AC Value of I

This command reads I_{AC}, the AC value of the current defined by

$$I_{\mathrm{AC}} := \sqrt{I_{\mathrm{trms}}^2 - I_{\mathrm{DC}}^2},$$

where I_{trms} is the root mean square ITRMS [9.10.40→250] and I_{DC} the DC value IDC [9.10.32→247] of the current.

Display ID: I_{ac}
Suffix: l

Query form

Short form: IAC?
SCPI form: :FETCh[:SCALar]:CURRent:AC?
            :READ[:SCALar]:CURRent:AC?

Response

The measured value

Type: <NRf>
Unit: A

9.10.31 ICF
Crest Factor of I

This command reads I_{CF}, the crest factor of the current defined by

$$I_{\mathrm{CF}} := \frac{I_{\mathrm{peak}}}{I_{\mathrm{trms}}},$$

where I_{trms} is the root mean square value of the voltage ITRMS [9.10.40→250] and I_{peak} is the maximum absolute sample value of the voltage occuring during the time interval given by TSNORM [9.10.200→313] and DURNORM [9.10.197→311]. This can be expressed in an other way as

$$I_{\mathrm{peak}} = \max\{I_{\mathrm{Max}}, |I_{\mathrm{Min}}|\}$$

with I_{Max} the biggest sample IMAX [9.10.35→248] value and $|I_{\mathrm{Min}}|$ the modulus of the smallest sample IMIN [9.10.36→248] value.

Display ID: I_{cf}
Suffix: l

Query form

Short form: ICF?
SCPI form: :FETCh[:SCALar]:CURRent:CFACtor?
            :READ[:SCALar]:CURRent:CFACtor?

Response

The measured value

Type: <NRf>

9.10.32 IDC
DC Value of I

This command reads the DC value of the current I_DC. defined by

$$I_\mathrm{DC} := \frac{1}{T} \int\limits_{t_1}^{t_1+T} i(t) \, dt,$$

where T is the duration DURNORM [9.10.197→311] and t_1 the starting time point TSNORM [9.10.200→313] of the measurement.

Display ID: I_dc
Suffix: l

Query form

Short form: IDC?
SCPI form: :FETCh[:SCALar]:CURRent:DC?
 :READ[:SCALar]:CURRent:DC?

Response

The measured value

Type: <NRf>
Unit: A

9.10.33 IFF
Form Factor of I

This command reads I_ff, the form factor of the current defined by

$$I_\mathrm{ff} := \frac{I_\mathrm{trms}}{I_\mathrm{rect}},$$

where I_trms is the root mean square ITRMS [9.10.40→250] and I_rect the rectified value IREC [9.10.38→249] of the voltage.

Display ID: I_ff
Suffix: l

Query form

Short form: IFF?
SCPI form: :FETCh[:SCALar]:CURRent:FFACtor?
 :READ[:SCALar]:CURRent:FFACtor?

Response

The measured value

Type: <NRf>

9.10.34 IINR
Inrush Current

This command reads I_{inr} , the value of the inrush current defined by

$$I_{\mathrm{inr}} := \max|i(t)|,$$

to be the maximum of the momentary current $i(t)$ since the start of the power meter, respectively the last reset by IINC [9.10.458→429] of the inrush current, until now.

Display ID: $\mathrm{I_{inr}}$
Suffix: 1

Query form
Short form: IINR?
SCPI form: :FETCh[:SCALar]:CURRent:INRush?
        :READ[:SCALar]:CURRent:INRush?

Response
The measured value

Type: <NRf>
Unit: A

9.10.35 IMAX
Max. Current in Interval

This command reads I_{Max}, the biggest sample value of the current occuring during the time interval given by TSNORM [9.10.200→313] and DURNORM [9.10.197→311].

Display ID: $\mathrm{I_{max}}$
Suffix: 1

Query form
Short form: IMAX?
SCPI form: :FETCh[:SCALar]:CURRent:MAXPk?
        :READ[:SCALar]:CURRent:MAXPk?

Response
The measured value

Type: <NRf>
Unit: A

9.10.36 IMIN
Min. Current in Interval

This command reads I_{Min}, the smallest sample value of the current occuring during the time interval given by TSNORM [9.10.200→313] and DURNORM [9.10.197→311].

Display ID: $\mathrm{I_{min}}$
Suffix: 1

Query form
Short form: IMIN?
SCPI form: :FETCh[:SCALar]:CURRent:MINPk?
        :READ[:SCALar]:CURRent:MINPk?

Response

The measured value

Type: $<$NRf$>$
Unit: A

9.10.37 IPP
Peak to Peak Value of I

This command reads I_{PP}, the peak to peak value of the current defined by

$$I_{\mathrm{PP}} := I_{\mathrm{Max}} - I_{\mathrm{Min}},$$

where I_{Max} is the biggest IMAX [9.10.35→248] and I_{Min} the smallest IMIN [9.10.36→248] sample value of the current.

Display ID: I_{pp}
Suffix: 1

Query form

Short form: IPP?
SCPI form: :FETCh[:SCALar]:CURRent:PPEak?
:READ[:SCALar]:CURRent:PPEak?

Response

The measured value

Type: $<$NRf$>$
Unit: A

9.10.38 IREC
Rectified Value of I

This command reads I_{rect}, the rectified value of the current defined by

$$I_{\mathrm{rect}} := \frac{1}{T} \int\limits_{T}^{T+t_1} |i(t)| \, dt,$$

where T is the duration DURNORM [9.10.197→311] and t_1 the starting time point TSNORM [9.10.200→313] of the measurement.

Display ID: I_{rect}
Suffix: 1

Query form

Short form: IREC?
SCPI form: :FETCh[:SCALar]:CURRent:RECTify?
:READ[:SCALar]:CURRent:RECTify?

Response

The measured value

Type: $<$NRf$>$
Unit: A

9.10.39 IRUSAGE
Usage of current range

This command reads the cycle-based range-usage of the current.

Display ID: Irusage
Suffix: p (1 ... 7)

Query form

Short form: IRUSAGE?
SCPI form: :FETCh[:SCALar]:CURRent:RUSage?
          :READ[:SCALar]:CURRent:RUSage?

Response

The measured value

Type: <NRf>
Unit: %

9.10.40 ITRMS
RMS of I

This command reads I_{trms}, the root mean square value of the current defined by

$$I_{\text{trms}} := \sqrt{\frac{1}{T} \int_{t_1}^{t_1+T} i(t)^2 \, dt},$$

where T is the duration DURNORM [9.10.197→311] and t_1 the starting time point TSNORM [9.10.200→313] of the measurement.

Display ID: I_{trms}
Suffix: l

Query form

Short form: ITRMS?
SCPI form: :FETCh[:SCALar]:CURRent[:TRMS]?
          :READ[:SCALar]:CURRent[:TRMS]?

Response

The measured value

Type: <NRf>
Unit: A

9.10.41 CSBIAM
Custom Samples based BIAM

Custom Samples based BIAM [9.10.105→274].

Display ID: csIh

Query form

Short form: CSBIAM?␣[<numeric_list>traces]

SCPI form: :FETCh[:SCALar]:CUSTomerSamples:BIAM?␣[<numeric_list>traces]

:READ[:SCALar]:CUSTomerSamples:BIAM?␣[<numeric_list>traces]

| [<numeric_list>traces] | Specifies the positions of the elements to query.

List elements: 0 ... 1000

Response

The measured value

Type: <NRf>,...

List elements: 0 ... 1000

Unit: A

9.10.42 CSBIPH

Custom Samples based BIPH

Custom Samples based BIPH [9.10.108→276].

Display ID: csIP

Query form

Short form: CSBIPH?␣[<numeric_list>traces]

SCPI form: :FETCh[:SCALar]:CUSTomerSamples:BIPH?␣[<numeric_list>traces]

:READ[:SCALar]:CUSTomerSamples:BIPH?␣[<numeric_list>traces]

| [<numeric_list>traces] | Specifies the positions of the elements to query.

List elements: 0 ... 1000

Response

The measured value

Type: <NRf>,...

List elements: 0 ... 1000

Unit: °

9.10.43 CSBPAM

Custom Samples based BPAM

Custom Samples based BPAM [9.10.112→277].

Display ID: csPh

Query form

Short form: CSBPAM?␣[<numeric_list>traces]

SCPI form: :FETCh[:SCALar]:CUSTomerSamples:BPAM?␣[<numeric_list>traces]

:READ[:SCALar]:CUSTomerSamples:BPAM?␣[<numeric_list>traces]

| [<numeric_list>traces] | Specifies the positions of the elements to query.

List elements: 0 ... 1000

Response

The measured value

Type: <NRf>,...

List elements: 0 ... 1000

Unit: W

9.10.44 CSBQAM
Custom Samples based BQAM
────

Custom Samples based BQAM [9.10.117→280].

Display ID: csQh

Query form

Short form: CSBQAM?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:BQAM?␣[<numeric_list>traces]
          :READ[:SCALar]:CUSTomerSamples:BQAM?␣[<numeric_list>traces]

| [<numeric_list>traces] | Specifies the positions of the elements to query.

List elements: 0 ... 1000

Response

The measured value

Type: <NRf>,...
List elements: 0 ... 1000
Unit: var

9.10.45 CSBSAM
Custom Samples based BSAM
────

Custom Samples based BSAM [9.10.113→278].

Display ID: csSh

Query form

Short form: CSBSAM?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:BSAM?␣[<numeric_list>traces]
          :READ[:SCALar]:CUSTomerSamples:BSAM?␣[<numeric_list>traces]

| [<numeric_list>traces] | Specifies the positions of the elements to query.

List elements: 0 ... 1000

Response

The measured value

Type: <NRf>,...
List elements: 0 ... 1000
Unit: VA

9.10.46 CSBUAM
Custom Samples based BUAM
────

Custom Samples based BUAM [9.10.123→283].

Display ID: csUh

Query form

Short form: CSBUAM?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:BUAM?␣[<numeric_list>traces]
:READ[:SCALar]:CUSTomerSamples:BUAM?␣[<numeric_list>traces]

[<numeric_list>traces] Specifies the positions of the elements to query.

List elements: 0 ... 1000

Response

The measured value

Type: <NRf>,...
List elements: 0 ... 1000
Unit: V

9.10.47 CSBUPH
Custom Samples based BUPH

Custom Samples based BUPH [9.10.126→284].

Display ID: csUP

Query form

Short form: CSBUPH?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:BUPH?␣[<numeric_list>traces]
:READ[:SCALar]:CUSTomerSamples:BUPH?␣[<numeric_list>traces]

[<numeric_list>traces] Specifies the positions of the elements to query.

List elements: 0 ... 1000

Response

The measured value

Type: <NRf>,...
List elements: 0 ... 1000
Unit: °

9.10.48 CSFHARM
Custom Samples based FHARM

Custom Samples based FHARM [9.10.103→274].

Display ID: csf$_1$

Query form

Short form: CSFHARM?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:FHARM?
:READ[:SCALar]:CUSTomerSamples:FHARM?

Response

The measured value

Type: <NRf>
Unit: Hz

9.10.49 CSIAC
Custom Samples based IAC

Custom Samples based IAC [9.10.30→246].

Display ID: csI$_{ac}$

Query form
Short form: CSIAC?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:IAC?
:READ[:SCALar]:CUSTomerSamples:IAC?

Response
The measured value

Type: <NRf>
Unit: A

9.10.50 CSICF
Custom Samples based ICF

Custom Samples based ICF [9.10.31→246].

Display ID: csI$_{cf}$

Query form
Short form: CSICF?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:ICF?
:READ[:SCALar]:CUSTomerSamples:ICF?

Response
The measured value

Type: <NRf>

9.10.51 CSIDC
Custom Samples based IDC

Custom Samples based IDC [9.10.32→247].

Display ID: csI$_{dc}$

Query form
Short form: CSIDC?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:IDC?
:READ[:SCALar]:CUSTomerSamples:IDC?

Response
The measured value

Type: <NRf>
Unit: A

9.10.52 CSIFF
Custom Samples based IFF

Custom Samples based IFF [9.10.33→247].

Display ID: csI_{ff}

Query form
Short form: CSIFF?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:IFF?
:READ[:SCALar]:CUSTomerSamples:IFF?

Response
The measured value

Type: $<NRf>$

9.10.53 CSIMAX
Custom Samples based IMAX

Custom Samples based IMAX [9.10.35→248].

Display ID: csI_{max}

Query form
Short form: CSIMAX?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:IMAX?
:READ[:SCALar]:CUSTomerSamples:IMAX?

Response
The measured value

Type: $<NRf>$
Unit: A

9.10.54 CSIMIN
Custom Samples based IMIN

Custom Samples based IMIN [9.10.36→248].

Display ID: csI_{min}

Query form
Short form: CSIMIN?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:IMIN?
:READ[:SCALar]:CUSTomerSamples:IMIN?

Response
The measured value

Type: $<NRf>$
Unit: A

9.10.55 CSIPP
Custom Samples based IPP

Custom Samples based IPP [9.10.37→249].

Display ID: csI$_{pp}$

Query form
Short form: CSIPP?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:IPP?
              :READ[:SCALar]:CUSTomerSamples:IPP?

Response
The measured value

Type: <NRf>
Unit: A

9.10.56 CSIREC
Custom Samples based IREC

Custom Samples based IREC [9.10.38→249].

Display ID: csI$_{rect}$

Query form
Short form: CSIREC?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:IREC?
              :READ[:SCALar]:CUSTomerSamples:IREC?

Response
The measured value

Type: <NRf>
Unit: A

9.10.57 CSITRMS
Custom Samples based ITRMS

Custom Samples based ITRMS [9.10.40→250].

Display ID: csI$_{trms}$

Query form
Short form: CSITRMS?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:ITRMS?
              :READ[:SCALar]:CUSTomerSamples:ITRMS?

Response
The measured value

Type: <NRf>
Unit: A

9.10.58 CSP
Custom Samples based P

Custom Samples based P [9.10.132→287].

Display ID: csP

Query form

Short form: CSP?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:P?
　　　　　:READ[:SCALar]:CUSTomerSamples:P?

Response

The measured value

Type: <NRf>
Unit: W

9.10.59 CSPF
Custom Samples based PF

Custom Samples based PF [9.10.141→291].

Display ID: csPF

Query form

Short form: CSPF?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:PF?
　　　　　:READ[:SCALar]:CUSTomerSamples:PF?

Response

The measured value

Type: <NRf>

9.10.60 CSQ
Custom Samples based Q

Custom Samples based Q [9.10.143→292].

Display ID: csQ

Query form

Short form: CSQ?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:Q?
　　　　　:READ[:SCALar]:CUSTomerSamples:Q?

Response

The measured value

Type: <NRf>
Unit: var

9.10.61 CSRSER
Custom Samples based RSER

Custom Samples based RSER [9.10.165→300].

Display ID: csR$_{ser}$

Query form
Short form: CSRSER?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:RSER?
          :READ[:SCALar]:CUSTomerSamples:RSER?

Response
The measured value

Type: <NRf>
Unit: Ω

9.10.62 CSS
Custom Samples based S

Custom Samples based S [9.10.133→287].

Display ID: csS

Query form
Short form: CSS?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:S?
          :READ[:SCALar]:CUSTomerSamples:S?

Response
The measured value

Type: <NRf>
Unit: VA

9.10.63 CSUAC
Custom Samples based UAC

Custom Samples based UAC [9.10.207→315].

Display ID: csU$_{ac}$

Query form
Short form: CSUAC?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:UAC?
          :READ[:SCALar]:CUSTomerSamples:UAC?

Response
The measured value

Type: <NRf>
Unit: V

9.10.64 CSUCF
Custom Samples based UCF

Custom Samples based UCF [9.10.208→315].

Display ID: csU$_{cf}$

Query form
Short form: CSUCF?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:UCF?
:READ[:SCALar]:CUSTomerSamples:UCF?

Response
The measured value

Type: <NRf>

9.10.65 CSUDC
Custom Samples based UDC

Custom Samples based UDC [9.10.210→316].

Display ID: csU$_{dc}$

Query form
Short form: CSUDC?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:UDC?
:READ[:SCALar]:CUSTomerSamples:UDC?

Response
The measured value

Type: <NRf>
Unit: V

9.10.66 CSUFF
Custom Samples based UFF

Custom Samples based UFF [9.10.211→317].

Display ID: csU$_{ff}$

Query form
Short form: CSUFF?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:UFF?
:READ[:SCALar]:CUSTomerSamples:UFF?

Response
The measured value

Type: <NRf>

9.10.67 CSUMAX
Custom Samples based UMAX

Custom Samples based UMAX [9.10.212→317].

Display ID: csU$_{max}$

Query form
Short form: CSUMAX?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:UMAX?
:READ[:SCALar]:CUSTomerSamples:UMAX?

Response
The measured value

Type: <NRf>
Unit: V

9.10.68 CSUMIN
Custom Samples based UMIN

Custom Samples based UMIN [9.10.213→317].

Display ID: csU_{min}

Query form
Short form: CSUMIN?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:UMIN?
:READ[:SCALar]:CUSTomerSamples:UMIN?

Response
The measured value

Type: <NRf>
Unit: V

9.10.69 CSUPP
Custom Samples based UPP

Custom Samples based UPP [9.10.214→318].

Display ID: csU_{pp}

Query form
Short form: CSUPP?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:UPP?
:READ[:SCALar]:CUSTomerSamples:UPP?

Response
The measured value

Type: <NRf>
Unit: V

9.10.70 CSUREC
Custom Samples based UREC

Custom Samples based UREC [9.10.215→318].

Display ID: csU_{rect}

Query form
Short form: CSUREC?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:UREC?
:READ[:SCALar]:CUSTomerSamples:UREC?

Response
The measured value

Type: $<$NRf$>$
Unit: V

9.10.71 CSUTRMS
Custom Samples based UTRMS

Custom Samples based UTRMS [9.10.217→319].

Display ID: csU$_{\text{trms}}$

Query form
Short form: CSUTRMS?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples[:UTRMS]?
          :READ[:SCALar]:CUSTomerSamples[:UTRMS]?

Response
The measured value

Type: $<$NRf$>$
Unit: V

9.10.72 CSXSER
Custom Samples based XSER

Custom Samples based XSER [9.10.167→301].

Display ID: csX$_{\text{ser}}$

Query form
Short form: CSXSER?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:XSER?
          :READ[:SCALar]:CUSTomerSamples:XSER?

Response
The measured value

Type: $<$NRf$>$
Unit: Ω

9.10.73 CSZ
Custom Samples based Z

Custom Samples based Z [9.10.166→300].

Display ID: csZ

Query form
Short form: CSZ?
SCPI form: :FETCh[:SCALar]:CUSTomerSamples:Z?
          :READ[:SCALar]:CUSTomerSamples:Z?

Response
The measured value

Type: $<$NRf$>$
Unit: Ω

9.10.74 EP
Active energy

This command reads E_{P}, the active energy defined by

$$E_{\mathrm{P}} := \int\limits_{t_S}^{t_S+T_{\mathrm{EN}}} u(t) \cdot i(t) \, dt,$$

where t_S is the time stamp of the energy start ᴛꜱᴇɴ [9.10.194→310] and T_{EN} the integration time of the energy ᴅᴜʀᴇɴ [9.10.191→309]. This definition is conform to DIN 40110.

Display ID: EP
Suffix: l

Query form

Short form: EP?
SCPI form: :FETCh[:SCALar]:ENERgy[:ACTive]?
:READ[:SCALar]:ENERgy[:ACTive]?

Response

The measured value

Type: <NRf>
Unit: Wh

9.10.75 ES
Apparent energy

This command reads E_{S}, the apparent energy defined by

$$E_{\mathrm{S}} := \sqrt{\int\limits_{t_S}^{t_S+T_{\mathrm{EN}}} u^2(t) \, dt \cdot \int\limits_{t_S}^{t_S+T_{\mathrm{EN}}} i^2(t) \, dt}$$

where t_S is the time stamp of the energy start ᴛꜱᴇɴ [9.10.194→310] and T_{EN} the integration time of the energy ᴅᴜʀᴇɴ [9.10.191→309].This definition is conform to DIN 40110.

Display ID: ES
Suffix: l

Query form

Short form: ES?
SCPI form: :FETCh[:SCALar]:ENERgy:APParent?
:READ[:SCALar]:ENERgy:APParent?

Response

The measured value

Type: <NRf>
Unit: VAh

9.10.76 EI
Charge

This command reads E_I, the charge defined by

$$E_\mathrm{I} := \int\limits_{t_S}^{t_S + T_{\mathrm{EN}}} i(t)\ dt,$$

where t_S is the time stamp of the energy start ᴛꜱᴇɴ [9.10.194→310] and T_{EN} the integration time of the energy ᴅᴜʀᴇɴ [9.10.191→309]. This definition is conform to DIN 40110.

Display ID: q
Suffix: l

Query form

Short form: EI?
SCPI form: :FETCh[:SCALar]:ENERgy:CHARge?
      :READ[:SCALar]:ENERgy:CHARge?

Response

The measured value

Type: <NRf>
Unit: Ah

9.10.77 EPINT
Active energy of last interval

This command reads E_P, the active energy of the last interval defined by

$$E_\mathrm{P} := \int\limits_{t_S}^{t_S + T_{\mathrm{EN}}} u(t) \cdot i(t)\ dt,$$

where t_S is the time stamp of the interval start ᴛꜱᴇɴɪɴᴛ [9.10.193→310] and T_{EN} the integration time of the interval ᴅᴜʀᴇɴɪɴᴛ [9.10.192→310]. This definition is conform to DIN 40110.

Display ID: $\mathrm{EP_{int}}$
Suffix: l

Query form

Short form: EPINT?
SCPI form: :FETCh[:SCALar]:ENERgy:INTerval:ACTiveint?
      :READ[:SCALar]:ENERgy:INTerval:ACTiveint?

Response

The measured value

Type: <NRf>
Unit: Wh

9.10.78 ESINT
Apparent energy of last interval

This command reads E_S, the apparent energy of last interval defined by

$$E_\text{S} := \sqrt{\int\limits_{t_S}^{t_S+T_\text{EN}} u^2(t)\ dt \cdot \int\limits_{t_S}^{t_S+T_\text{EN}} i^2(t)\ dt}$$

where t_S is the time stamp of the energy start ᴛꜱᴇɴɪɴᴛ [9.10.193→310] and T_EN the integration time of the energy ᴅᴜʀᴇɴɪɴᴛ [9.10.192→310].This definition is conform to DIN 40110.

Display ID: ES_int
Suffix: l

Query form

Short form: ESINT?
SCPI form: :FETCh[:SCALar]:ENERgy:INTerval:APParentint?
:READ[:SCALar]:ENERgy:INTerval:APParentint?

Response

The measured value

Type: <NRf>
Unit: VAh

9.10.79 EIINT
Charge of last interval

This command reads E_I, the charge of the last interval defined by

$$E_\text{I} := \int\limits_{t_S}^{t_S+T_\text{EN}} i(t)\ dt,$$

where t_S is the time stamp of the energy start ᴛꜱᴇɴɪɴᴛ [9.10.193→310] and T_EN the integration time of the energy ᴅᴜʀᴇɴɪɴᴛ [9.10.192→310]. This definition is conform to DIN 40110.

Display ID: q_int
Suffix: l

Query form

Short form: EIINT?
SCPI form: :FETCh[:SCALar]:ENERgy:INTerval:CHARgeint?
:READ[:SCALar]:ENERgy:INTerval:CHARgeint?

Response

The measured value

Type: <NRf>
Unit: Ah

9.10.80 ENERGYINTERVAL
Last finished interval

Returns the number of the latest complete energy interval.

Display ID: En_{int}
Suffix: l

Query form

Short form: ENERGYINTERVAL?
SCPI form: :FETCh[:SCALar]:ENERgy:INTerval[:NUMBer]?
:READ[:SCALar]:ENERgy:INTerval[:NUMBer]?

Response

The measured value

Type: <NRi>

9.10.81 EQINT
Reactive energy of last interval

This command reads E_Q, the reactive energy of the last interval defined by

$$E_Q := \sqrt{E_S^2 - E_P^2}$$

where E_S is the apparent energy ESINT [9.10.78→264] and E_P is the active energy EPINT [9.10.77→263]. If the voltage is sinusoidal, this definition is conform to DIN 40110,

Display ID: EQ_{int}
Suffix: l

Query form

Short form: EQINT?
SCPI form: :FETCh[:SCALar]:ENERgy:INTerval:REACtiveint?
:READ[:SCALar]:ENERgy:INTerval:REACtiveint?

Response

The measured value

Type: <NRf>
Unit: varh

9.10.82 EQ
Reactive energy

This command reads E_Q, the reactive energy defined by

$$E_Q := \sqrt{E_S^2 - E_P^2}$$

where E_S is the apparent energy ES [9.10.75→262] and E_P is the active energy EP [9.10.74→262]. If the voltage is sinusoidal, this definition is conform to DIN 40110,

Display ID: EQ
Suffix: l

Query form

Short form: EQ?
SCPI form: :FETCh[:SCALar]:ENERgy:REACtive?
:READ[:SCALar]:ENERgy:REACtive?

Response

The measured value

Type: <NRf>
Unit: varh

9.10.83 ENERGYSTATE
Current state of energy measurement

This commands reads the energy measuring state of all groups as a bitmask.

Display ID: Energystate

Query form

Short form: ENERGYSTATE?
SCPI form: :FETCh[:SCALar]:ENERgy:STATe?
:READ[:SCALar]:ENERgy:STATe?

Response

The measured value

Type: <NRi>

9.10.84 ENV
Read a float environment variable

This command reads a float environment variable. Float evironment variables are settable via SETENV [9.10.87→267] and can be used as external value store, bound to the current measurement cycle.

Display ID: Env

Query form

Short form: ENV?␣<numeric_list>rangevalue
SCPI form: :FETCh[:SCALar]:ENVironment:ENV?␣<numeric_list>rangevalue
:READ[:SCALar]:ENVironment:ENV?␣<numeric_list>rangevalue

| <numeric_list>rangevalue | Specifies the positions of the elements to query. |

List elements: 0 ... 127

Response

This field tells which environment variables shall be returned.

Type: <NRf>,...
List elements: 0 ... 127

9.10.85 ENVINT

Read an integer environment variable

This command reads an integer environment variable. Integer evironment variables are settable via SETENVINT [9.10.86→267] and can be used as external value store, bound to the current measurement cycle.

Display ID: Env$_{\text{Int}}$

Query form

Short form: ENVINT?␣<numeric_list>rangevalue
SCPI form: :FETCh[:SCALar]:ENVironment:INTeger:ENVInt?␣<numeric_list>rangevalue
:READ[:SCALar]:ENVironment:INTeger:ENVInt?␣<numeric_list>rangevalue

| <numeric_list>rangevalue | Specifies the positions of the elements to query. |

> List elements: 0 ... 127

Response

This field tells which environment variables shall be returned.

Type: <NRi>,...
List elements: 0 ... 127

9.10.86 SETENVINT

Set an integer environment variable

This command sets an integer environment variable.

Command form

Short form: SETENVINT␣<numeric_list>rangevalue, <NRi>value
SCPI form: :FETCh[:SCALar]:ENVironment:INTeger:SETint␣<numeric_list>rangevalue, <NRi>value
:READ[:SCALar]:ENVironment:INTeger:SETint␣<numeric_list>rangevalue, <NRi>value

| <numeric_list>rangevalue | Specifies the positions of the elements to modify. |

> List elements: 0 ... 127

| <NRi>value | The set-value |

9.10.87 SETENV

Set a float environment variable

This command sets a float environment variable.

Command form

Short form: SETENV␣<numeric_list>rangevalue, <NRf>value
SCPI form: :FETCh[:SCALar]:ENVironment:SET␣<numeric_list>rangevalue, <NRf>value
:READ[:SCALar]:ENVironment:SET␣<numeric_list>rangevalue, <NRf>value

| <numeric_list>rangevalue | Specifies the positions of the elements to modify. |

> List elements: 0 ... 127

| <NRf>value | The set-value |

9.10.88 FLNUMCH
Currently startable flicker channels

This command reads the number of currently startable flicker channels (Blockwise for 20 halfwaves, see CALCULATION OF FLICKER, OPTION L6-OPT-FLK [6.19→140]).

Display ID: Flicker$_{\mathrm{NumCh}}$
Suffix: fl (1 ... 3)

Query form
Short form: FLNUMCH?␣[<numeric_list>retVal]
SCPI form: :FETCh[:SCALar]:FLICker:CHAnnelnumbers?␣[<numeric_list>retVal]
:READ[:SCALar]:FLICker:CHAnnelnumbers?␣[<numeric_list>retVal]

| [<numeric_list>retVal] | Specifies the positions of the elements to query.

List elements: 0 ... 19

Response
The measured value

Type: <NRi>,...
List elements: 0 ... 19

9.10.89 FLIHP
Half period value of current

This command reads the half period value of the current (Blockwise for 20 halfwaves, see CALCULATION OF FLICKER, OPTION L6-OPT-FLK [6.19→140]).

Display ID: FlI$_{\mathrm{hp}}$
Suffix: fl (1 ... 3)

Query form
Short form: FLIHP?␣[<numeric_list>retVal]
SCPI form: :FETCh[:SCALar]:FLICker:Ihp?␣[<numeric_list>retVal]
:READ[:SCALar]:FLICker:Ihp?␣[<numeric_list>retVal]

| [<numeric_list>retVal] | Specifies the positions of the elements to query.

List elements: 0 ... 19

Response
The measured value

Type: <NRf>,...
List elements: 0 ... 19
Unit: A

9.10.90 FLINTINDEX
Currently processed flicker interval

This command reads the index of the short time interval of the flicker measurement that is currently processed (Blockwise for 20 halfwaves, see CALCULATION OF FLICKER, OPTION L6-OPT-FLK [6.19→140]).

Display ID: Flicker$_{\mathrm{int}}$
Suffix: fl (1 ... 3)

Query form

Short form: FLINTINDEX?␣[<numeric_list>retVal]
SCPI form: :FETCh[:SCALar]:FLICker:IINDex?␣[<numeric_list>retVal]
:READ[:SCALar]:FLICker:IINDex?␣[<numeric_list>retVal]

[<numeric_list>retVal] Specifies the positions of the elements to query.

List elements: 0 ... 19

Response

The measured value

Type: <NRi>,...
List elements: 0 ... 19

9.10.91 FLLTINT
Duration of the long term flicker interval

This command reads the duration of the long term interval of the flicker measurement in seconds (Blockwise for 20 halfwaves, see CALCULATION OF FLICKER, OPTION L6-OPT-FLK [6.19→140]).

Display ID: FlickerLT$_{dur}$
Suffix: fl (1 ... 3)

Query form

Short form: FLLTINT?␣[<numeric_list>retVal]
SCPI form: :FETCh[:SCALar]:FLICker:LTInterval?␣[<numeric_list>retVal]
:READ[:SCALar]:FLICker:LTInterval?␣[<numeric_list>retVal]

[<numeric_list>retVal] Specifies the positions of the elements to query.

List elements: 0 ... 19

Response

The measured value

Type: <Time>,...
List elements: 0 ... 19
Unit: s

9.10.92 FLPHP
Half period value of power

This command reads the half period value of the power (Blockwise for 20 halfwaves, see CALCULATION OF FLICKER, OPTION L6-OPT-FLK [6.19→140]).

Display ID: FlP$_{hp}$
Suffix: fl (1 ... 3)

Query form

Short form: FLPHP?␣[<numeric_list>retVal]
SCPI form: :FETCh[:SCALar]:FLICker:PHP?␣[<numeric_list>retVal]
:READ[:SCALar]:FLICker:PHP?␣[<numeric_list>retVal]

[<numeric_list>retVal] Specifies the positions of the elements to query.

List elements: 0 ... 19

Response

The measured value

Type: $<$NRf$>$,...
List elements: 0 ... 19
Unit: W

9.10.93 FLPINST
Instantaneous flicker value

This command reads the instantaneous flicker value (Blockwise for 20 halfwaves, see CALCULATION OF FLICKER, OPTION L6-OPT-FLK [6.19→140]).

Display ID: P_{inst}
Suffix: fl (1 ... 3)

Query form

Short form: FLPINST?␣[$<$numeric_list$>$retVal]
SCPI form: :FETCh[:SCALar]:FLICker:PINSt?␣[$<$numeric_list$>$retVal]
              :READ[:SCALar]:FLICker:PINSt?␣[$<$numeric_list$>$retVal]

| [$<$numeric_list$>$retVal] | Specifies the positions of the elements to query.

      List elements: 0 ... 19

Response

The measured value

Type: $<$NRf$>$,...
List elements: 0 ... 19

9.10.94 FLPLT
Long term flicker value

This command reads the flicker long term value (Blockwise for 20 halfwaves, see CALCULATION OF FLICKER, OPTION L6-OPT-FLK [6.19→140]).

Display ID: P_{lt}
Suffix: fl (1 ... 3)

Query form

Short form: FLPLT?␣[$<$numeric_list$>$retVal]
SCPI form: :FETCh[:SCALar]:FLICker:PLT?␣[$<$numeric_list$>$retVal]
              :READ[:SCALar]:FLICker:PLT?␣[$<$numeric_list$>$retVal]

| [$<$numeric_list$>$retVal] | Specifies the positions of the elements to query.

      List elements: 0 ... 19

Response

The measured value

Type: $<$NRf$>$,...
List elements: 0 ... 19

9.10.95 FLPST
Flicker short term value

This command reads the flicker short term value (Blockwise for 20 halfwaves, see Calculation of flicker, option L6-OPT-FLK [6.19→140]).

Display ID: P_{st}
Suffix: fl (1 ... 3)

Query form
Short form: FLPST?␣[<numeric_list>retVal]
SCPI form: :FETCh[:SCALar]:FLICker:PST?␣[<numeric_list>retVal]
:READ[:SCALar]:FLICker:PST?␣[<numeric_list>retVal]

[<numeric_list>retVal] Specifies the positions of the elements to query.

List elements: 0 ... 19

Response
The measured value

Type: <NRf>,...
List elements: 0 ... 19

9.10.96 FLICKERRESET
reset of flicker

This Command resets the flicker measurement (not the flicker settings).

Command form
Short form: FLICKERRESET
SCPI form: :FETCh[:SCALar]:FLICker:RESET
:READ[:SCALar]:FLICker:RESET

9.10.97 FLICKERSTART
start of flicker

This Command starts the flicker measurement.

Command form
Short form: FLICKERSTART
SCPI form: :FETCh[:SCALar]:FLICker:START
:READ[:SCALar]:FLICker:START

9.10.98 FLSTATE
Current state of flicker system

This command reads state of the flicker measurement (Blockwise for 20 halfwaves, see Calculation of flicker, option L6-OPT-FLK [6.19→140]).

Display ID: FlState
Suffix: fl (1 ... 3)

Query form
Short form: FLSTATE?␣[<numeric_list>retVal]
SCPI form: :FETCh[:SCALar]:FLICker:State?␣[<numeric_list>retVal]
:READ[:SCALar]:FLICker:State?␣[<numeric_list>retVal]

| [<numeric_list>retVal] | Specifies the positions of the elements to query.

List elements: 0 ... 19

Possible values:

0 or inactive:	The flicker is inactive.
1 or warming:	The flicker is warming up.
2 or meassuring:	The flicker is meassuring.
3 or finish:	The flicker meassuring of the long term value is finished.
4 or aborted:	The flicker measurement aborted.

Response
The measured value

Type: <NRi>,...
List elements: 0 ... 19

Possible values:

0: The flicker is inactive.
1: The flicker is warming up.
2: The flicker is meassuring.
3: The flicker meassuring of the long term value is finished.
4: The flicker measurement aborted.

9.10.99 FLICKERSTOP
stop of flicker

This Command stops the flicker measurement.

Command form
Short form: FLICKERSTOP
SCPI form: :FETCh[:SCALar]:FLICker:STOP
:READ[:SCALar]:FLICker:STOP

9.10.100 FLTIMEININT
Progress in current interval

This command reads the elapsed time of current short term interval (Blockwise for 20 halfwaves, see CALCULATION OF FLICKER, OPTION L6-OPT-FLK [6.19→140]).

Display ID: FlT$_{InInt}$
Suffix: fl (1 ... 3)

Query form
Short form: FLTIMEININT?␣[<numeric_list>retVal]
SCPI form: :FETCh[:SCALar]:FLICker:TimeInInt?␣[<numeric_list>retVal]
:READ[:SCALar]:FLICker:TimeInInt?␣[<numeric_list>retVal]

| [<numeric_list>retVal] | Specifies the positions of the elements to query.

List elements: 0 ... 19

Response

Elapsed time of interval

Type: <Time>,...
List elements: 0 ... 19

9.10.101 FLUHP
Half period value of voltage

This command reads the half period value of the voltage (Blockwise for 20 halfwaves, see CALCULATION OF FLICKER, OPTION L6-OPT-FLK [6.19→140]).

Display ID: FlU$_{hp}$
Suffix: fl (1 ... 3)

Query form

Short form: FLUHP?␣[<numeric_list>retVal]
SCPI form: :FETCh[:SCALar]:FLICker:Uhp?␣[<numeric_list>retVal]
            :READ[:SCALar]:FLICker:Uhp?␣[<numeric_list>retVal]

| [<numeric_list>retVal] | Specifies the positions of the elements to query.

List elements: 0 ... 19

Response

The measured value

Type: <NRf>,...
List elements: 0 ... 19
Unit: V

9.10.102 FCYC
Cycle Frequency

This command reads f_{cyc}, the cycle frequency. It is determined during the time interval given by TSNORM [9.10.200→313] and DURNORM [9.10.197→311].

Display ID: f$_{cycle}$
Suffix: l

Query form

Short form: FCYC?
SCPI form: :FETCh[:SCALar]:FREQuency[:CYCLe]?
            :READ[:SCALar]:FREQuency[:CYCLe]?

Response

The measured value

Type: <NRf>
Unit: Hz

9.10.103 FHARM
Frequency in Harm. Interval

This command reads f_{harm}, the frequency determined during the time intervall of the harmonics analysis with starting point TSHARM [9.10.196→311] and duration DURHARM [9.10.195→311].

Display ID: f_1
Suffix: l

Query form
Short form: FHARM?
SCPI form: :FETCh[:SCALar]:FREQuency:HARMonics?
          :READ[:SCALar]:FREQuency:HARMonics?

Response
The measured value

Type: <NRf>
Unit: Hz

9.10.104 GPIOREAD
reads the current gpio state

The current state of the GPIO pin of the sync connector (see GPIODIR [9.10.409→409] and GPIODATA [9.10.408→408]).

Display ID: GPIO
Suffix: gpio (1 ... 6)

Query form
Short form: GPIOREAD?
SCPI form: :FETCh[:SCALar]:GPIO:READ?
          :READ[:SCALar]:GPIO:READ?

Response
The measured value

Type: <NRi>

9.10.105 BIAM
RMS of I of Harm.

This command reads I_k, the root mean square value of the current of the k-th harmonics.

Display ID: Ih
Suffix: l

Query form
Short form: BIAM?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:HARMonics:CURRent:AMPLitude?␣[<numeric_list>traces]
          :READ[:SCALar]:HARMonics:CURRent:AMPLitude?␣[<numeric_list>traces]

[<numeric_list>traces]	Specifies the positions of the elements to query.

      List elements: 0 ... 1000

Response

The measured value

Type: <NRf>,...
List elements: 0 ... 1000
Unit: A

9.10.106 BIIM
Imaginary part of current harmonics

This command reads the imaginary part of the current of the k-th harmonics bin.

Display ID: BI_{IM}
Suffix: l

Query form

Short form: BIIM?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:HARMonics:CURRent:BIMaginary?␣[<numeric_list>traces]
                :READ[:SCALar]:HARMonics:CURRent:BIMaginary?␣[<numeric_list>traces]

| [<numeric_list>traces] | Specifies the positions of the elements to query.

      List elements: 0 ... 2000

Response

The measured value

Type: <NRf>,...
List elements: 0 ... 2000
Unit: A

9.10.107 BIRE
Real part of current harmonics

This command reads the real part of the current of the k-th harmonics bin.

Display ID: BI_{RE}
Suffix: l

Query form

Short form: BIRE?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:HARMonics:CURRent:BREal?␣[<numeric_list>traces]
                :READ[:SCALar]:HARMonics:CURRent:BREal?␣[<numeric_list>traces]

| [<numeric_list>traces] | Specifies the positions of the elements to query.

      List elements: 0 ... 2000

Response

The measured value

Type: <NRf>,...
List elements: 0 ... 2000
Unit: A

9.10.108 BIPH
Phase of I of Harm.

This command reads φ_{I_k}, the phase angle of the current of the k-th harmonic.

Display ID: IP
Suffix: 1

Query form
Short form: BIPH?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:HARMonics:CURRent:PHASe?␣[<numeric_list>traces]
:READ[:SCALar]:HARMonics:CURRent:PHASe?␣[<numeric_list>traces]

[<numeric_list>traces] Specifies the positions of the elements to query.

List elements: 0 ... 1000

Response
The measured value

Type: <NRf>,...
List elements: 0 ... 1000
Unit: °

9.10.109 BRELI
Relative RMS of I of Harmonics

This command reads the relative root mean square value of the current of the k-th harmonics. The reference is given by HRELREF [9.10.122→282].

Display ID: Irelh
Suffix: 1

Query form
Short form: BRELI?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:HARMonics:CURRent:RelAMPLitude?␣[<numeric_list>traces]
:READ[:SCALar]:HARMonics:CURRent:RelAMPLitude?␣[<numeric_list>traces]

[<numeric_list>traces] Specifies the positions of the elements to query.

List elements: 0 ... 1000

Response
The measured value

Type: <NRf>,...
List elements: 0 ... 1000
Unit: %

9.10.110 IHRMS
RMS of I of Harm.

Summation of the effective current of all valid harmonics.

Display ID: I$_{\mathrm{hrms}}$
Suffix: 1

Query form

Short form: IHRMS?
SCPI form: :FETCh[:SCALar]:HARMonics:CURRent:RMS?
:READ[:SCALar]:HARMonics:CURRent:RMS?

Response

The measured value

Type: <NRf>
Unit: A

9.10.111 HIHD
Total Harm. Distortion of I

This command reads the total harmonics distortion I_{THD} of the current defined by

$$I_{\text{THD}} := \frac{1}{I_1}\sqrt{I_2^2 + \cdots I_N^2},$$

where N is the number of harmonics (without inter harmonics) available and $I_1, I_2 \cdots$ are the harmonic amplitudes of the current BIAM [9.10.105→274].

Display ID: I_{thd}
Suffix: l

Query form

Short form: HIHD?
SCPI form: :FETCh[:SCALar]:HARMonics:CURRent:THDistort?
:READ[:SCALar]:HARMonics:CURRent:THDistort?

Response

The measured value

Type: <NRf>
Unit: %

9.10.112 BPAM
Active Power of Harmonic

This command reads the active power P_k of the k-th harmonics defined by

$$P_k = U_k \cdot I_k \cdot \cos(\varphi_{U_k} - \varphi_{I_k}).$$

In this definition, U_k is the amplitude of the voltage BUAM [9.10.123→283], I_k the amplitude of the current BIAM [9.10.105→274], φ_{U_k} the phase of the voltage BUPH [9.10.126→284], and φ_{I_k} the phase of the current BIPH [9.10.108→276]. All these values are with respect to the k-th harmonic.

Display ID: Ph
Suffix: l

Query form

Short form: BPAM?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:HARMonics:POWer:ACTive?␣[<numeric_list>traces]
:READ[:SCALar]:HARMonics:POWer:ACTive?␣[<numeric_list>traces]

[<numeric_list>traces]	Specifies the positions of the elements to query.

List elements: 0 ... 1000

Response

The measured value

Type: $<$NRf$>$,...
List elements: 0 ... 1000
Unit: W

9.10.113 BSAM
Apparent Power of n-th Harm.

This command reads

$$\sqrt{P_k^2 + Q_k^2},$$

where P_k is the active BPAM [9.10.112→277] and Q_k the reactive power BQAM [9.10.117→280] of the k-th harmonic.

Display ID: Sh
Suffix: l

Query form

Short form: BSAM?␣[$<$numeric_list$>$traces]
SCPI form: :FETCh[:SCALar]:HARMonics:POWer:APParent?␣[$<$numeric_list$>$traces]
               :READ[:SCALar]:HARMonics:POWer:APParent?␣[$<$numeric_list$>$traces]

$\boxed{[<\text{numeric\_list}>\text{traces}]}$ Specifies the positions of the elements to query.

List elements: 0 ... 1000

Response

The measured value

Type: $<$NRf$>$,...
List elements: 0 ... 1000
Unit: VA

9.10.114 D
Distortion Power

This command reads the distortion power D defined by

$$D := \sqrt{S_h^2 - P_h^2 - Q_h^2},$$

where

$$S_{\mathrm{h}} := U_{\mathrm{hrms}} \cdot I_{\mathrm{hrms}}$$

is the apparent power of the harmonics SHARM [9.10.119→281] with $U_{\mathrm{hrms}} := \sqrt{U_0^2 + \cdots + U_N^2}$ and $I_{\mathrm{hrms}} := \sqrt{I_0^2 + \cdots + I_N^2}$. Moreover,

$$P_{\mathrm{h}} := \sum_{k=0}^{N} U_k \cdot I_k \cdot \cos(\varphi_{U_k} - \varphi_{I_k})$$

is the active power of the harmonics PHARM [9.10.118→280],

$$Q_{\mathrm{h}} := \sum_{k=0}^{N} U_k \cdot I_k \cdot \sin(\varphi_{U_k} - \varphi_{I_k})$$

the reactive power of the harmonics QHARM [9.10.120→281], and N the number of harmonics available HNUM [9.10.115→279].

Display ID: D
Suffix: 1

Query form

Short form: D?
SCPI form: :FETCh[:SCALar]:HARMonics:POWer:PERiod:DISTortion?
:READ[:SCALar]:HARMonics:POWer:PERiod:DISTortion?

Response

The measured value

Type: <NRf>
Unit: var

9.10.115 HNUM
Number of available Harmonics

This command reads the number N of harmonics available, including the inter harmonics.

Display ID: N_1
Suffix: 1

Query form

Short form: HNUM?
SCPI form: :FETCh[:SCALar]:HARMonics:POWer:PERiod:NUM?
:READ[:SCALar]:HARMonics:POWer:PERiod:NUM?

Response

The measured value

Type: <NRi>

9.10.116 HPERNUM
Period number of harmonic window

This command reads the harmonics periods number. This is the number of periods within the time intervall of the harmonics analysis. The harmonics analysis is performed over an integer number of periods.

Suffix: 1

Query form

Short form: HPERNUM?
SCPI form: :FETCh[:SCALar]:HARMonics:POWer:PERiod:PNUMber?
:READ[:SCALar]:HARMonics:POWer:PERiod:PNUMber?

Response

The measured value

Type: <NRf>

9.10.117 BQAM
Reactive Power of Harmonics

This command reads the reactive power Q_k of the k-th harmonics defined by

$$Q_k := U_k \cdot I_k \cdot \sin(\varphi_{U_k} - \varphi_{I_k}).$$

In this definition, U_k is the amplitude of the voltage BUAM [9.10.123→283], I_k the amplitude of the current BIAM [9.10.105→274], and $\varphi_{U_k} - \varphi_{I_k}$ the difference between the phase of the voltage BUPH [9.10.126→284] and the phase of the current BIPH [9.10.108→276]. All these values are with respect to the k-th harmonics. Note that Q_k can be negative.

Display ID: Qh
Suffix: l

Query form
Short form: BQAM?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:HARMonics:POWer:REACtive?␣[<numeric_list>traces]
              :READ[:SCALar]:HARMonics:POWer:REACtive?␣[<numeric_list>traces]

| [<numeric_list>traces] | Specifies the positions of the elements to query.

      List elements: 0 ... 1000

Response
The measured value

Type: <NRf>,...
List elements: 0 ... 1000
Unit: var

9.10.118 PHARM
Active Power of Harmonics

This command reads the active power P_{h} of the harmonics defined by

$$P_{\mathrm{h}} := \sum_{k=0}^{N} P_k,$$

where

$$P_k = U_k \cdot I_k \cdot \cos(\varphi_{U_k} - \varphi_{I_k})$$

is the active power of the k-th harmonics BPAM [9.10.112→277] and N the number of harmonics available HNUM [9.10.115→279].

Display ID: P_{h}
Suffix: l

Query form
Short form: PHARM?
SCPI form: :FETCh[:SCALar]:HARMonics:POWer:SUM:ACTive?
              :READ[:SCALar]:HARMonics:POWer:SUM:ACTive?

Response
The measured value

Type: <NRf>
Unit: W

9.10.119 SHARM
Apparent Power of Harmonics

This command reads the apparent power S_h of the harmonics defined by

$$S_\mathrm{h} := U_\mathrm{hrms} \cdot I_\mathrm{hrms},$$

where

$$U_\mathrm{hrms} := \sqrt{U_0^2 + U_1^2 + \cdots + U_N^2}$$

is the root mean square value of the voltage of the harmonics UHRMS [9.10.128→285] and

$$I_\mathrm{hrms} := \sqrt{I_0^2 + I_1^2 + \cdots + I_N^2},$$

the corresponding current IHRMS [9.10.110→276] and N the number of harmonics available HNUM [9.10.115→279].

Display ID: S_h
Suffix: 1

Query form

Short form: SHARM?
SCPI form: :FETCh[:SCALar]:HARMonics:POWer:SUM:APParent?
           :READ[:SCALar]:HARMonics:POWer:SUM:APParent?

Response

The measured value

Type: <NRf>
Unit: VA

9.10.120 QHARM
Reactive Power of Harmonics

This command reads the reactive power Q_h of the harmonics by Budeanu, defined to be

$$Q_\mathrm{h} := \sum_{k=0}^{N} Q_k,$$

where

$$Q_k = U_k \cdot I_k \cdot \sin(\varphi_{U_k} - \varphi_{I_k})$$

is the reactive power of the k-th harmonics BQAM [9.10.117→280] and N the number of harmonics available HNUM [9.10.115→279]. Note that Q_h can be negative.

Display ID: Q_h
Suffix: 1

Query form

Short form: QHARM?
SCPI form: :FETCh[:SCALar]:HARMonics:POWer:SUM:REACtive?
           :READ[:SCALar]:HARMonics:POWer:SUM:REACtive?

Response

The measured value

Type: <NRf>
Unit: var

9.10.121 QTOT
Total Reactive Power of Harm.

This command reads the total reactive power Q_{tot} of the harmonics by Budeanu, defined to be

$$Q_{\text{tot}} := \sqrt{D^2 + Q_{\text{h}}^2},$$

where D is the distortion power D [9.10.114→278] and Q_{h} reactive power of the harmonics by Budeanu QHARM [9.10.120→281].

Display ID: Q_{tot}
Suffix: l

Query form
Short form: QTOT?
SCPI form: :FETCh[:SCALar]:HARMonics:POWer:SUM:TOTReactive?
:READ[:SCALar]:HARMonics:POWer:SUM:TOTReactive?

Response
The measured value

Type: $<$NRf$>$
Unit: var

9.10.122 HRELREF
Harmonics Relative Reference

Sets whether the reference of the relative harmonic values is the fundamental harmonic or the total harmonic content. See BRELU [9.10.127→284], BRELI [9.10.109→276], BRELP [9.10.144→292], BRELS [9.10.145→293], and BRELQ [9.10.146→293]

Display ID: Harmonics Relative Reference
Suffix: g (1 ... 7)

Command form
Short form: HRELREF␣$<$NRi$>$val
SCPI form: :FETCh[:SCALar]:HARMonics:RReference␣$<$NRi$>$val
:READ[:SCALar]:HARMonics:RReference␣$<$NRi$>$val

| $<$NRi$>$val | Config Value |

   *RST value: 0

   Possible values:

   0 or FUNDamental: Harmonics are relative to the fundamental.
   1 or FULL: Harmonics are relative to the rms of all harmonics.

Query form
Short form: HRELREF?
SCPI form: :FETCh[:SCALar]:HARMonics:RReference?
:READ[:SCALar]:HARMonics:RReference?

Response
Config Value

Type: $<$NRi$>$

Possible values:

0: Harmonics are relative to the fundamental.
1: Harmonics are relative to the rms of all harmonics.

9.10.123 BUAM
RMS of U of Harmonics

This command reads U_k, the root mean square value of the voltage of the k-th harmonics.

Display ID: Uh
Suffix: l

Query form
Short form: BUAM?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:HARMonics[:VOLTage]:AMPLitude?␣[<numeric_list>traces]
　　　　　　:READ[:SCALar]:HARMonics[:VOLTage]:AMPLitude?␣[<numeric_list>traces]

[<numeric_list>traces] Specifies the positions of the elements to query.

　　　List elements: 0 ... 1000

Response
The measured value

Type:　　　　<NRf>,...
List elements: 0 ... 1000
Unit:　　　　V

9.10.124 BUIM
Imaginary part of voltage harmonics

This command reads the imaginary part of the voltage of the k-th harmonics bin.

Display ID: BU_{IM}
Suffix: l

Query form
Short form: BUIM?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:HARMonics[:VOLTage]:BIMaginary?␣[<numeric_list>traces]
　　　　　　:READ[:SCALar]:HARMonics[:VOLTage]:BIMaginary?␣[<numeric_list>traces]

[<numeric_list>traces] Specifies the positions of the elements to query.

　　　List elements: 0 ... 2000

Response
The measured value

Type:　　　　<NRf>,...
List elements: 0 ... 2000
Unit:　　　　V

9.10.125 BURE
Real part of voltage harmonics

This command reads the real part of the voltage of the k-th harmonics bin.

Display ID: BU_{RE}
Suffix: l

Query form
Short form: BURE?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:HARMonics[:VOLTage]:BREal?␣[<numeric_list>traces]
:READ[:SCALar]:HARMonics[:VOLTage]:BREal?␣[<numeric_list>traces]

| [<numeric_list>traces] | Specifies the positions of the elements to query.

List elements: 0 ... 2000

Response
The measured value

Type: <NRf>,...
List elements: 0 ... 2000
Unit: V

9.10.126 BUPH
Phase of U of Harmonics

This command reads φ_{U_k}, the phase of the voltage of the k-th harmonic.

Display ID: UP
Suffix: l

Query form
Short form: BUPH?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:HARMonics[:VOLTage]:PHASe?␣[<numeric_list>traces]
:READ[:SCALar]:HARMonics[:VOLTage]:PHASe?␣[<numeric_list>traces]

| [<numeric_list>traces] | Specifies the positions of the elements to query.

List elements: 0 ... 1000

Response
The measured value

Type: <NRf>,...
List elements: 0 ... 1000
Unit: °

9.10.127 BRELU
Relative RMS of U of Harmonics

This command reads the relative root mean square value of the voltage of the k-th harmonics. The reference is given by HRELREF [9.10.122→282].

Display ID: Urelh
Suffix: l

Query form
Short form: BRELU?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:HARMonics[:VOLTage]:RelAMPLitude?␣[<numeric_list>traces]
:READ[:SCALar]:HARMonics[:VOLTage]:RelAMPLitude?␣[<numeric_list>traces]

| [<numeric_list>traces] | Specifies the positions of the elements to query.

List elements: 0 ... 1000

Response

The measured value

Type: <NRf>,...
List elements: 0 ... 1000
Unit: %

9.10.128 UHRMS
RMS of U of Harm.

Summation of Utrms of all valid harmonics.

Display ID: U$_{\mathrm{hrms}}$
Suffix: 1

Query form

Short form: UHRMS?
SCPI form: :FETCh[:SCALar]:HARMonics[:VOLTage]:RMS?
             :READ[:SCALar]:HARMonics[:VOLTage]:RMS?

Response

The measured value

Type: <NRf>
Unit: V

9.10.129 HUHD
Total Harm. Distortion of U

This command reads the total harmonics distortion U_{THD} of the voltage defined by

$$U_{\mathrm{THD}} = \frac{1}{U_1}\sqrt{U_2^2 + \cdots U_N^2},$$

where N is the number of harmonics (without inter harmonics) Ind $U_1, U_2 \cdots$ are the harmonics amplitudes of the voltage ʙᴜᴀᴍ [9.10.123→283]].

Display ID: U$_{\mathrm{thd}}$
Suffix: 1

Query form

Short form: HUHD?
SCPI form: :FETCh[:SCALar]:HARMonics[:VOLTage]:THDistort?
             :READ[:SCALar]:HARMonics[:VOLTage]:THDistort?

Response

The measured value

Type: <NRf>
Unit: %

9.10.130 PM
Averaged Active Power

This command reads P_{M}, the average active power of the energy measuring defined by

$$P_{\mathrm{M}} := \frac{1}{T_{\mathrm{EN}}} \int\limits_{t_S}^{t_S+T_{\mathrm{EN}}} u(t) \cdot i(t)\, dt,$$

where t_S is the time stamp of the energy start TSEN [9.10.194→310] and T_{EN} the integration time of the energy DUREN [9.10.191→309]. This definition is conform to DIN 40110.

Display ID: Pm
Suffix: l

Query form

Short form: PM?
SCPI form: :FETCh[:SCALar]:POWer:AACTive?
   :READ[:SCALar]:POWer:AACTive?

Response

The measured value

Type: <NRf>
Unit: W

9.10.131 SM
Averaged Apparent Power

This command reads S_{M}, the average apparent power of the energy measuring defined by

$$S_{\mathrm{M}} := \frac{1}{T_{\mathrm{En}}} \sqrt{\int\limits_{t_S}^{t_S+T_{\mathrm{EN}}} u^2(t)\, dt \cdot \int\limits_{t_S}^{t_S+T_{\mathrm{EN}}} i^2(t)\, dt},$$

where t_S is the time stamp of the energy start TSEN [9.10.194→310] and T_{EN} the integration time of the energy DUREN [9.10.191→309]. This definition is conform to DIN 40110.

Display ID: Sm
Suffix: l

Query form

Short form: SM?
SCPI form: :FETCh[:SCALar]:POWer:AAPParent?
   :READ[:SCALar]:POWer:AAPParent?

Response

The measured value

Type: <NRf>
Unit: VA

9.10.132 P
Active Power

This command reads P, the active power defined by

$$P := \frac{1}{T} \int\limits_{t_1}^{t_1+T} u(t) \cdot i(t)\, dt$$

where T is the duration DURNORM [9.10.197→311] and t_1 the starting time point TSNORM [9.10.200→313] of the measurement.

Display ID: P
Suffix: 1

Query form

Short form: P?
SCPI form: :FETCh[:SCALar]:POWer[:ACTive]?
          :READ[:SCALar]:POWer[:ACTive]?

Response

The measured value

Type: <NRf>
Unit: W

9.10.133 S
Apparent Power

This command reads S, the apparent power defined by

$$S := U_{\mathrm{trms}} \cdot I_{\mathrm{trms}},$$

where U_{trms} and I_{trms} are the root mean square values of the voltage UTRMS [9.10.217→319] and current ITRMS [9.10.40→250].

Display ID: S
Suffix: 1

Query form

Short form: S?
SCPI form: :FETCh[:SCALar]:POWer:APParent?
          :READ[:SCALar]:POWer:APParent?

Response

The measured value

Type: <NRf>
Unit: VA

9.10.134 QM
Averaged Reactive

This command reads Q_{M}, the average reactive power of the energy measuring defined by

$$Q_{\mathrm{M}} := \sqrt{S_{\mathrm{M}}^2 - P_{\mathrm{M}}^2}$$

where S_{M} is the average apparent power SM [9.10.131→286] and P_{M} the average active power PM [9.10.130→286]. If the voltage is sinusoidal, this definition is conform to DIN 40110,

Display ID: Qm
Suffix: l

Query form

Short form: QM?
SCPI form: :FETCh[:SCALar]:POWer:AREactive?
          :READ[:SCALar]:POWer:AREactive?

Response

The measured value

Type: <NRf>
Unit: var

9.10.135 ETA
Energy Conversion Efficiency

This command reads the energy conversion efficiency. The efficiency is available for any combination of groups (and between groups and PSI card). The suffix denotes the INPUT and the index denotes the OUTPUT group. The efficiency is calculated as P_{OUTPUT}/P_{INPUT}. The suffix and index start both from 1 for groups, and from 7 for the PSI values (mechanical power).

Display ID: η
Suffix: eff (1 ... 10)

Query form

Short form: ETA?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:POWer:ETA?␣[<numeric_list>traces]
          :READ[:SCALar]:POWer:ETA?␣[<numeric_list>traces]

[<numeric_list>traces]	Specifies the positions of the elements to query.

      List elements: 0 ... 10

Response

The measured value

Type: <NRf>,...
List elements: 0 ... 10

9.10.136 INCA
Inductive/Capacitive Flag

This command reads the inductive capacitive flag. Its value is determined by Φ, the phase difference between the current and the voltage of the first harmonics PHI [9.10.142→292]. It returns 1 for the inductive case where Φ is between 1 and 179°, and it returns -1 for the capacitive case where Φ is between -179 and $-1°$. In any other cases the INCA flag is 0.

Display ID: Inca
Suffix: l

Query form
Short form: INCA?
SCPI form: :FETCh[:SCALar]:POWer:ICAPacity?
:READ[:SCALar]:POWer:ICAPacity?

Response
The measured value

Type: $<$NRi$>$

Possible values:

-1: Capacitive behaviour
0: Undefined behaviour
1: Inductive behaviour

9.10.137 PMINT
Averaged Active Power of last interval

This command reads P_{M}, the average active power of the energy measuring of the last interval defined by

$$P_{\mathrm{M}} := \frac{1}{T_{\mathrm{EN}}} \int\limits_{t_S}^{t_S + T_{\mathrm{EN}}} u(t) \cdot i(t)\ dt,$$

where t_S is the time stamp of the interval start TSENINT [9.10.193→310] and T_{EN} the integration time of the INTERVAL DURENINT [9.10.192→310]. This definition is conform to DIN 40110.

Display ID: $\mathrm{Pm}_{\mathrm{int}}$
Suffix: l

Query form

Short form: PMINT?
SCPI form: :FETCh[:SCALar]:POWer:INTerval:AACTiveint?
:READ[:SCALar]:POWer:INTerval:AACTiveint?

Response

The measured value

Type: $<$NRf$>$
Unit: W

9.10.138 SMINT

Averaged Apparent Power of last interval

This command reads S_M, the average apparent power of the energy measuring of the last interval defined by

$$S_M := \frac{1}{T_{En}} \sqrt{\int\limits_{t_S}^{t_S+T_{EN}} u^2(t)\,dt \cdot \int\limits_{t_S}^{t_S+T_{EN}} i^2(t)\,dt,}$$

where t_S is the time stamp of the interval start ᴛsᴇɴɪɴᴛ [9.10.193→310] and T_{EN} the integration time of the interval ᴅᴜʀᴇɴɪɴᴛ [9.10.192→310]. This definition is conform to DIN 40110.

Display ID: Sm$_{int}$
Suffix: 1

Query form

Short form: SMINT?
SCPI form: :FETCh[:SCALar]:POWer:INTerval:AAPParentint?
:READ[:SCALar]:POWer:INTerval:AAPParentint?

Response

The measured value

Type: <NRf>
Unit: VA

9.10.139 QMINT

Averaged Reactive of last interval

This command reads Q_M, the average reactive power of the energy measuring of the last interval defined by

$$Q_M := \sqrt{S_M^2 - P_M^2}$$

where S_M is the average apparent power of the interval sᴍɪɴᴛ [9.10.138→290] and P_M the average active power of the interval ᴘᴍɪɴᴛ [9.10.137→289]. If the voltage is sinusoidal, this definition is conform to DIN 40110,

Display ID: Qm$_{int}$
Suffix: 1

Query form

Short form: QMINT?
SCPI form: :FETCh[:SCALar]:POWer:INTerval:AREactiveint?
:READ[:SCALar]:POWer:INTerval:AREactiveint?

Response

The measured value

Type: <NRf>
Unit: var

9.10.140 PLOSS
Power Loss

This command reads the power loss. The power loss is available for any combination of groups (and between groups and PSI card). The suffix denotes the INPUT and the index denotes the OUTPUT group. The power loss is calculated as $P_{OUTPUT} - P_{INPUT}$. The suffix and index start both from 1 for groups, and from 7 for the PSI values (mechanical power).

Display ID: P_{loss}
Suffix: eff (1 ... 10)

Query form

Short form: PLOSS?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:POWer:LOSS?␣[<numeric_list>traces]
         :READ[:SCALar]:POWer:LOSS?␣[<numeric_list>traces]

| [<numeric_list>traces] | Specifies the positions of the elements to query.

List elements: 0 ... 10

Response

The measured value

Type: <NRf>,...
List elements: 0 ... 10
Unit: W

9.10.141 PF
Power Factor

This command reads λ, the power factor defined by

$$\lambda := \frac{|P|}{S},$$

where P ist the active P [9.10.132→287] and S the apparent S [9.10.133→287] power.

Display ID: PF
Suffix: 1

Query form

Short form: PF?
SCPI form: :FETCh[:SCALar]:POWer:PFACtor?
         :READ[:SCALar]:POWer:PFACtor?

Response

The measured value

Type: <NRf>

9.10.142 PHI
Phase difference of fund. frequency

This command reads Φ, the phase difference between the current φ_{I_1} and the voltage φ_{U_1} of the fundamental frequency. This can be written in a formula by

$$\Phi := \varphi_{U_1} - \varphi_{I_1}.$$

Display ID: φ_{fund}
Suffix: 1

Query form
Short form: PHI?
SCPI form: :FETCh[:SCALar]:POWer:PHASe?
　　　　　 :READ[:SCALar]:POWer:PHASe?

Response
The measured value

Type: <NRf>
Unit: °

9.10.143 Q
Reactive Power

This command reads Q, the reactive power defined by

$$Q := \sqrt{S^2 - P^2},$$

where S ist the apparent ${}_{S\ [9.10.133 \to 287]}$ and P the active ${}_{P\ [9.10.132 \to 287]}$ power.

Display ID: Q
Suffix: 1

Query form
Short form: Q?
SCPI form: :FETCh[:SCALar]:POWer:REACtive?
　　　　　 :READ[:SCALar]:POWer:REACtive?

Response
The measured value

Type: <NRf>
Unit: var

9.10.144 BRELP
Relative RMS of P of Harmonics

This command reads the relative value of the power of the k-th harmonics. The reference is given by HRELREF [9.10.122→282].

Display ID: Prelh
Suffix: 1

Query form

Short form: BRELP?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:POWer:RELative[:ACTive]?␣[<numeric_list>traces]
              :READ[:SCALar]:POWer:RELative[:ACTive]?␣[<numeric_list>traces]

| [<numeric_list>traces] | Specifies the positions of the elements to query.

List elements: 0 ... 1000

Response

The measured value

Type: <NRf>,...
List elements: 0 ... 1000
Unit: %

9.10.145 BRELS
Relative RMS of S of Harmonics

This command reads the relative value of S of the *k*-th harmonics. The reference is given by HRELREF [9.10.122→282].

Display ID: Srelh
Suffix: l

Query form

Short form: BRELS?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:POWer:RELative:APParent?␣[<numeric_list>traces]
              :READ[:SCALar]:POWer:RELative:APParent?␣[<numeric_list>traces]

| [<numeric_list>traces] | Specifies the positions of the elements to query.

List elements: 0 ... 1000

Response

The measured value

Type: <NRf>,...
List elements: 0 ... 1000
Unit: V

9.10.146 BRELQ
Relative RMS of Q of Harmonics

This command reads the relative value of Q of the *k*-th harmonics. The reference is given by HRELREF [9.10.122→282].

Display ID: Qrelh
Suffix: l

Query form

Short form: BRELQ?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:POWer:RELative:REactive?␣[<numeric_list>traces]
              :READ[:SCALar]:POWer:RELative:REactive?␣[<numeric_list>traces]

| [<numeric_list>traces] | Specifies the positions of the elements to query.

List elements: 0 ... 1000

Response

The measured value

Type: <NRf>,...
List elements: 0 ... 1000
Unit: %

9.10.147 PSIAIN

scaled slow analog input

The scaled voltage of the analog input of the PSI. This voltage can by scaled using the four scaling values (see PSIAIXA [9.10.295→356] and following).

Display ID: PSIA$_{IN}$
Suffix: ai (1 ... 8, 101 ... 108 and 201 ... 208)

Query form

Short form: PSIAIN?
SCPI form: :FETCh[:SCALar]:PSI:AIN[:VALue]?
          :READ[:SCALar]:PSI:AIN[:VALue]?

Response

The measured value

Type: <NRf>
Unit: V

9.10.148 PSIFAIN

scaled fast analog input

The scaled fast analog input voltage of the PSI. This voltage can by scaled using the four scaling values (see PSIFAIXA [9.10.307→360] and following).

Display ID: PSIFA$_{IN}$
Suffix: afi (1 ... 2, 101 ... 102 and 201 ... 202)

Query form

Short form: PSIFAIN?
SCPI form: :FETCh[:SCALar]:PSI:AIN[:VALue]?
          :READ[:SCALar]:PSI:AIN[:VALue]?

Response

The measured value

Type: <NRf>
Unit: V

9.10.149 PSIAOUT

scaled analog output

The scaled voltage of the analog output of the PSI. This voltage can by scaled using the four scaling values (see PSIAOXA [9.10.300→358] and following).

Display ID: PSIA$_{OUT}$
Suffix: ao (1 ... 32, 101 ... 132 and 201 ... 232)

Query form
Short form: PSIAOUT?
SCPI form: :FETCh[:SCALar]:PSI:AOUT?
:READ[:SCALar]:PSI:AOUT?

Response
The measured value

Type: <NRf>
Unit: V

9.10.150 PSIACOUT
scaled analog clipped output

The scaled and to the limits of the analog output clipped voltage of the analog output of the PSI (see PSIAOUT [9.10.149→294]).

Display ID: PSIA$_{COUT}$
Suffix: ao (1 ... 32, 101 ... 132 and 201 ... 232)

Query form
Short form: PSIACOUT?
SCPI form: :FETCh[:SCALar]:PSI:CAOUT?
:READ[:SCALar]:PSI:CAOUT?

Response
The measured value

Type: <NRf>

9.10.151 PSICOUNT
pulse count

Counted digital in pulses since the last reset (see PSIRESETCOUNT [9.10.306→360] and PSIRESETALLCOUNT [9.10.304→360]). The Pulses are counted with full PSI digital input sampling rate (sub-cycle resolution).

Display ID: PSID$_{Count}$
Suffix: di (1 ... 8, 101 ... 108 and 201 ... 208)

Query form
Short form: PSICOUNT?
SCPI form: :FETCh[:SCALar]:PSI:COUNT?
:READ[:SCALar]:PSI:COUNT?

Response
The measured value

Type: <NRi>

9.10.152 PSIDIN
digital input

The state of the digital input pin. This value is the median of states in the respective cycle. The state can by inverted using PSIDIINV [9.10.305→360].

Display ID: PSID$_{IN}$
Suffix: di (1 ... 8, 101 ... 108 and 201 ... 208)

Query form
Short form: PSIDIN?
SCPI form: :FETCh[:SCALar]:PSI:DIN?
　　　　　　:READ[:SCALar]:PSI:DIN?

Response
The measured value

Type: <NRi>

9.10.153 PSIDIR

───── **direction**

The direction of the frequency input of the PSI in 'Motor' mode (see PSIMODE [9.10.331→373]).

Display ID: PSIM$_{DIR}$
Suffix: motor (1 ... 2, 101 ... 102 and 201 ... 202)

Query form
Short form: PSIDIR?
SCPI form: :FETCh[:SCALar]:PSI:DIR?
　　　　　　:READ[:SCALar]:PSI:DIR?

Response
The measured value

Type: <NRi>

9.10.154 PSIFDIR

───── **frequency direction**

The direction of the frequency input of the PSI in 'Direct' mode (see PSIMODE [9.10.331→373]).

Display ID: PSIF$_{DIR}$
Suffix: fi (1 ... 2, 101 ... 102 and 201 ... 202)

Query form
Short form: PSIFDIR?
SCPI form: :FETCh[:SCALar]:PSI:FDIR?
　　　　　　:READ[:SCALar]:PSI:FDIR?

Response
The measured value

Type: <NRi>

9.10.155 PSIFREQ

───── **frequency**

The frequency at the frequency input of the PSI card in 'Direct' mode (see PSIMODE [9.10.331→373]). This value can by scaled using the four scaling values (see PSIFRQXA [9.10.323→369] and following).

Display ID: PSIF$_{FREQ}$
Suffix: fi (1 ... 2, 101 ... 102 and 201 ... 202)

Query form
Short form: PSIFREQ?
SCPI form: :FETCh[:SCALar]:PSI:FREQ?
            :READ[:SCALar]:PSI:FREQ?

Response
The measured value

Type: <NRf>
Unit: Hz

9.10.156 PSIID

Id value of the Id/Iq tranformation

This command reads the Id value of the Id-Iq-transformation, averaged over a meassuring cycle. For more information see ID-IQ-TRANSFORMATION [6.21.8→143].

Display ID: IDIQ_{Id}
Suffix: motor (1 ... 2, 101 ... 102 and 201 ... 202)

Query form
Short form: PSIID?
SCPI form: :FETCh[:SCALar]:PSI:IDIQ:ID?
            :READ[:SCALar]:PSI:IDIQ:ID?

Response
The measured value

Type: <NRf>
Unit: A

9.10.157 PSIINULL

Inull value of the Id/Iq tranformation

This command reads the Inull value of the Id-Iq-transformation, averaged over a meassuring cycle. For more information see ID-IQ-TRANSFORMATION [6.21.8→143].

Display ID: $\text{IDIQ}_{\text{Inull}}$
Suffix: motor (1 ... 2, 101 ... 102 and 201 ... 202)

Query form
Short form: PSIINULL?
SCPI form: :FETCh[:SCALar]:PSI:IDIQ:INULL?
            :READ[:SCALar]:PSI:IDIQ:INULL?

Response
The measured value

Type: <NRf>
Unit: A

9.10.158 PSIIQ

Id value of the Id/Iq tranformation

This command reads the Iq value of the Id-Iq-transformation, averaged over a meassuring cycle. For more information see ID-IQ-TRANSFORMATION [6.21.8→143].

Display ID: IDIQ_{Iq}
Suffix: motor (1 ... 2, 101 ... 102 and 201 ... 202)

Query form
Short form: PSIIQ?
SCPI form: :FETCh[:SCALar]:PSI:IDIQ:IQ?
          :READ[:SCALar]:PSI:IDIQ:IQ?

Response
The measured value

Type: <NRf>
Unit: A

9.10.159 PSIPOLPAIRM
Number of pole pairs

This command reads the config value PSIPOLPAIR [9.10.329→372] used by the current cycle.

Display ID: Polpair

Query form
Short form: PSIPOLPAIRM?
SCPI form: :FETCh[:SCALar]:PSI:IDIQ:POLPAIR?
          :READ[:SCALar]:PSI:IDIQ:POLPAIR?

Response
The measured value

Type: <NRi>

9.10.160 PSIPOWER
mechanical power

The mechanical power calculated in 'Motor' mode using the PSI input from the torque and speed sensors (see PSIMODE [9.10.331→373], PSITRQTYPE [9.10.343→378], and PSISPTYPE [9.10.334→374]).

Display ID: PSIM$_{POWER}$
Suffix: motor (1 ... 2, 101 ... 102 and 201 ... 202)

Query form
Short form: PSIPOWER?
SCPI form: :FETCh[:SCALar]:PSI:POWER?
          :READ[:SCALar]:PSI:POWER?

Response
The measured value

Type: <NRf>
Unit: W

9.10.161 PSISPEED
speed

In motor mode, this command reads speed value detected by the connected analog or digital speed sensor. For more information see SPEED [6.21.7→143].

Display ID: PSIM$_{SPEED}$
Suffix: motor (1 ... 2, 101 ... 102 and 201 ... 202)

Query form
Short form: PSISPEED?
SCPI form: :FETCh[:SCALar]:PSI:SPEED?
:READ[:SCALar]:PSI:SPEED?

Response
The measured value

Type: <NRf>
Unit: min⁻¹

9.10.162 PSISWITCH
switch state

The state of the digital switch pin. The state is derived from the chosen source, the condition, the inverted flag, and the threshold (see PSISWSOURCE [9.10.339→377], PSISWCOND [9.10.337→376], PSISWINV [9.10.338→376], and PSISWTHRESH [9.10.340→377]).

Display ID: PSISW
Suffix: do (1 ... 8, 101 ... 108 and 201 ... 208)

Query form
Short form: PSISWITCH?
SCPI form: :FETCh[:SCALar]:PSI:SWITCH?
:READ[:SCALar]:PSI:SWITCH?

Response
The measured value

Type: <NRi>

9.10.163 PSITORQUE
torque

In motor mode, this command reads the value of the torque detected by the connected analog or digital torque sensor. For more information see SPEED [6.21.7→143].

Display ID: PSIM$_{\text{TORQUE}}$
Suffix: motor (1 ... 2, 101 ... 102 and 201 ... 202)

Query form
Short form: PSITORQUE?
SCPI form: :FETCh[:SCALar]:PSI:TORQUE?
:READ[:SCALar]:PSI:TORQUE?

Response
The measured value

Type: <NRf>
Unit: Nm

9.10.164 PSIZERO
zero position

The zero position of the PSI frequency input in 'Direct' mode (see PSIMODE [9.10.331→373]).

Display ID: PSIF$_{\text{ZERO}}$
Suffix: fi (1 ... 2, 101 ... 102 and 201 ... 202)

Query form
Short form: PSIZERO?
SCPI form: :FETCh[:SCALar]:PSI:ZERO?
:READ[:SCALar]:PSI:ZERO?

Response
The measured value

Type: <NRi>

9.10.165 RSER
Active Serial Resistance

This command reads R_{ser}, the active serial resistance defined by

$$R_{\text{ser}} := \frac{P}{I_{\text{trms}}^2},$$

where P is the active power P [9.10.132→287] and I_{trms} the root mean square value of the current ITRMS [9.10.40→250].

Display ID: R_{ser}
Suffix: l

Query form
Short form: RSER?
SCPI form: :FETCh[:SCALar]:RESistance:ASResist?
:READ[:SCALar]:RESistance:ASResist?

Response
The measured value

Type: <NRf>
Unit: Ω

9.10.166 Z
Impendance

This command reads the impedance Z defined by

$$Z := \frac{U_{\text{trms}}}{I_{\text{trms}}},$$

where U_{trms} and I_{trms} are the root mean square values of the voltage UTRMS [9.10.217→319] and current ITRMS [9.10.40→250].

Display ID: Z
Suffix: l

Query form
Short form: Z?
SCPI form: :FETCh[:SCALar]:RESistance:IMPedance?
:READ[:SCALar]:RESistance:IMPedance?

Response
The measured value

Type: <NRf>
Unit: Ω

9.10.167 XSER
Reactive Serial Resistance

This command reads X_{ser}, the reactive serial resistance defined by

$$X_{\mathrm{ser}} := \frac{Q}{I_{\mathrm{trms}}^2},$$

where Q ist the reactive power q [9.10.143→292] and I_{trms} the root mean square value of the current ITRMS [9.10.40→250].

Display ID: X_{ser}
Suffix: l

Query form
Short form: XSER?
SCPI form: :FETCh[:SCALar]:RESistance:RSIMpedance?
            :READ[:SCALar]:RESistance:RSIMpedance?

Response
The measured value

Type: <NRf>
Unit: Ω

9.10.168 SPCLEN
Written Samples of the Scope

This command reads the number of samples, which were written in scope until now.

Suffix: sp (1, 2, 21 and 22)

Query form
Short form: SPCLEN?
SCPI form: :FETCh[:SCALar]:SCOPe:CLENgth?
            :READ[:SCALar]:SCOPe:CLENgth?

Response
The measured value

Type: <NRi>

9.10.169 GLPNTR
Number of Tracks for the gapless Scope

This command reads the number of tracks of the gapless scope, that are available. It is limited by GLCSR [9.10.284→350].

Query form
Short form: GLPNTR?
SCPI form: :FETCh[:SCALar]:SCOPe:GAPLess:NTRacks?
            :READ[:SCALar]:SCOPe:GAPLess:NTRacks?

Response
The measured value

Type: <NRi>

9.10.170 GLPSR
Gapless Scope Sampling Rate

The same as GLCSR [9.10.284→350], but this is the value the scope was recorded with.

Display ID: GL_{psr}

Query form
Short form: GLPSR?
SCPI form: :FETCh[:SCALar]:SCOPe:GAPLess:SRATe?
　　　　　:READ[:SCALar]:SCOPe:GAPLess:SRATe?

Response
The measured value

Type: <NRf>

9.10.171 GLPTLEN
Number of All Samples

This command reads the number of all samples of a gapless scope track within one cycle. It is determined by GLCSR [9.10.284→350], to adapt the cycle time to a suitable value.

Query form
Short form: GLPTLEN?
SCPI form: :FETCh[:SCALar]:SCOPe:GAPLess:TLENgth?
　　　　　:READ[:SCALar]:SCOPe:GAPLess:TLENgth?

Response
The measured value

Type: <NRi>

9.10.172 GLPVAL
The list of samples.

This command reads the samples of the gapless scope. The first parameter specifies the track, which must be less than the value of GLPNTR [9.10.169→301]. The second parameter specifies the range of the samples of the track. This is typically (0 : GLPTLEN [9.10.171→302] - 1) for GLPVAL.

Display ID: Gapless Scope

Query form
Short form: GLPVAL?␣<NRi>trace, [<numeric_list>samples]
SCPI form: :FETCh[:SCALar]:SCOPe:GAPLess:VALues?␣<NRi>trace, [<numeric_list>samples]
　　　　　:READ[:SCALar]:SCOPe:GAPLess:VALues?␣<NRi>trace, [<numeric_list>samples]

| <NRi>trace | The trace to inspect |

| [<numeric_list>samples] | Specifies the positions of the elements to query. |

　　List elements: 0 ... 4194303

Response

The array of samples. 16 traces are allowed with list values 0..4194304

Type: <NRf>,...

List elements: 0 ... 4194303

9.10.173 SPNTR

Number of Tracks for the Scope

This command reads the number of tracks of the scope, that are available.

Suffix: sp (1, 2, 21 and 22)

Query form

Short form: SPNTR?

SCPI form: :FETCh[:SCALar]:SCOPe:NTRacks?
:READ[:SCALar]:SCOPe:NTRacks?

Response

The measured value

Type: <NRi>

9.10.174 SPPTRT

Pretrigger Length in Time

The same as SCPTRT [9.10.349→381], but this is the value the scope was recorded with.

Display ID: S_pptrt
Suffix: sp (1, 2, 21 and 22)

Query form

Short form: SPPTRT?

SCPI form: :FETCh[:SCALar]:SCOPe:PTRigger[:TIMe]?
:READ[:SCALar]:SCOPe:PTRigger[:TIMe]?

Response

The measured value

Type: <Time>

9.10.175 SPSR

Scope Sampling Rate

The same as SCSR [9.10.351→382], but this is the value the scope was recorded with.

Display ID: S_psr
Suffix: sp (1, 2, 21 and 22)

Query form

Short form: SPSR?

SCPI form: :FETCh[:SCALar]:SCOPe:SRATe?
:READ[:SCALar]:SCOPe:SRATe?

Response
The measured value

Type: <NRf>

9.10.176 SPSTAT
Scope state

This command reads the status of the scope.

Suffix: sp (1, 2, 21 and 22)

Query form
Short form: SPSTAT?
SCPI form: :FETCh[:SCALar]:SCOPe:STATus?
        :READ[:SCALar]:SCOPe:STATus?

Response
The measured value

Type: <NRi>

Possible values:

0: The scope is invalid.
1: It is loading the presamples.
2: It is searching for the trigger.
3: The scope is triggered, but still no sample is recorded, because the pretrigger is negative.
4: The scope is triggered and samples are recording.
5: All samples are recorded.
6: The scope is aborted.

9.10.177 SPTLEN
Number of All Samples

This command reads the number of all samples to be written. This is he case when the scope status SPSTAT [9.10.176→304] indicates finish.

Suffix: sp (1, 2, 21 and 22)

Query form
Short form: SPTLEN?
SCPI form: :FETCh[:SCALar]:SCOPe:TLENgth?
        :READ[:SCALar]:SCOPe:TLENgth?

Response
The measured value

Type: <NRi>

9.10.178 SPTPOS
Trigger Position

This command reads the index of the sample, that was recorded when the trigger occured.

Suffix: sp (1, 2, 21 and 22)

Query form
Short form: SPTPOS?
SCPI form: :FETCh[:SCALar]:SCOPe:TPOSition?
　　　　　　 :READ[:SCALar]:SCOPe:TPOSition?

Response
The measured value

Type: <NRi>

9.10.179 SPTRAC

source of the recorded trace

Returns the source of the recorded samples. This may differ from the current SCTRAC setting if SCTRAC has been changed meanwhile.

Display ID: S$_{ptrac}$
Suffix: sp (1, 2, 21 and 22)

Query form
Short form: SPTRAC?⌣[<numeric_list>trace source]
SCPI form: :FETCh[:SCALar]:SCOPe:TRACe?⌣[<numeric_list>trace source]
　　　　　　 :READ[:SCALar]:SCOPe:TRACe?⌣[<numeric_list>trace source]

| [<numeric_list>trace source] | Specifies the positions of the elements to query.

　　　List elements: 0 ... 7

Response
trace source

Type:　　　　　　<string program data>,...
List elements: 0 ... 7

9.10.180 SPVAL

The list of samples.

This command reads the samples of the scope. The size value specifies the number of sample values which can be stored for one scope. The tracks value specifies the number of tracks which are available for one scope. For the transient scope this is the maximum number of tracks. So each track can have a maximum number of sample values which is calculated by size/tracks. The list parameter specifies, which sample values are wanted. The smalles list index is 0, the biggest one is calculated like above minus 1. Each scope can be identified with a number which you can find in the suffixes row. The number of the track is the <NRi>parameter in front of the list.

Display ID: Scope
Suffix: sp (1, 2, 21 and 22)

Query form
Short form: SPVAL?⌣<NRi>trace, [<numeric_list>samples]
SCPI form: :FETCh[:SCALar]:SCOPe:VALues?⌣<NRi>trace, [<numeric_list>samples]
　　　　　　 :READ[:SCALar]:SCOPe:VALues?⌣<NRi>trace, [<numeric_list>samples]

| <NRi>trace | The trace to inspect

| [<numeric_list>samples] | Specifies the positions of the elements to query.

　　　List elements: 0 ... 2047

Response

The array of samples. For the suffixes 9 and 29 just one trace is alloed with list values 0..2047. For the suffixes 1, 2, 21 and 22 eight traces are allowed with list values 0..2047. For suffix 10 just one trace is allowed with list values 0..4194304

Type: <NRf>,...
List elements: 0 ... 2047

9.10.181 SCRRESULT
Result Values of the Script

This command reads the result value of the script. Direct access by varaiable or gui name is possible with SCRVARRESULT [9.10.182→306]

Display ID: Script Results

Query form
Short form: SCRRESULT?␣[<numeric_list>variable]
SCPI form: :FETCh[:SCALar]:SCRIPT:RESult?␣[<numeric_list>variable]
              :READ[:SCALar]:SCRIPT:RESult?␣[<numeric_list>variable]

| [<numeric_list>variable] | Specifies the positions of the elements to query.

     List elements: 0 ... 63

Response
The measured value

Type: <NRf>,...
List elements: 0 ... 63

9.10.182 SCRVARRESULT
Result of a named script variable

Allows to retrieve a script result by the variable name or by the gui representation defined in the def block. E.g. "a = 5" can be retrived with "scrvarresult? a" and 'v "myvoltage" = 5 "V"' can be retrievd with "scrvarresult? myvoltage".

Display ID: Script Results

Query form
Short form: SCRVARRESULT?␣<string program data>variable
SCPI form: :FETCh[:SCALar]:SCRIPT:VARRESult?␣<string program data>variable
              :READ[:SCALar]:SCRIPT:VARRESult?␣<string program data>variable

| <string program data>variable | The name of the variable

Response
The value of the variable

Type: <NRf>

9.10.183 TSCFG
Configuration timestamp

The timestamp of the last configuration change.

Suffix: 1

Query form
Short form: TSCFG?
SCPI form: :FETCh[:SCALar]:SLOTs:CONFig:TIMestamp?
:READ[:SCALar]:SLOTs:CONFig:TIMestamp?

Response
The timestamp of the last configuration change.

Type: <Date>

9.10.184 TRPSR
Transient Sampling Rate

The same as TRCSR [9.10.367→391], but this is the value the transient was recorded with.

Display ID: TR$_{psr}$

Query form
Short form: TRPSR?
SCPI form: :FETCh[:SCALar]:SLOTs:CONFig:TRANSient:SRATe?
:READ[:SCALar]:SLOTs:CONFig:TRANSient:SRATe?

Response
The measured value

Type: <NRf>

9.10.185 TRPSTAT
Scope state

This command reads the status of the transient.

Query form
Short form: TRPSTAT?
SCPI form: :FETCh[:SCALar]:SLOTs:CONFig:TRANSient:STATus?
:READ[:SCALar]:SLOTs:CONFig:TRANSient:STATus?

Response
The measured value

Type: <NRi>

Possible values:

0: The scope is invalid.
1: It is loading the presamples.
2: It is searching for the trigger.
3: The scope is triggered, but still no sample is recorded, because the pretrigger is negative.
4: The scope is triggered and samples are recording.
5: All samples are recorded.
6: The scope is aborted.

9.10.186 TRPTLEN
Number of All Samples

This command reads the number of all samples to be written. This is the case, when the transient status TRPSTAT [9.10.185→307] indicates finish.

Display ID: TR$_{ptlen}$

Query form
Short form: TRPTLEN?
SCPI form: :FETCh[:SCALar]:SLOTs:CONFig:TRANSient:TLENgth?
　　　　　:READ[:SCALar]:SLOTs:CONFig:TRANSient:TLENgth?

Response
The measured value

Type: <NRi>

9.10.187 TRPTRAC
source of the recorded trace

Returns the source of the recorded samples. This may differ from the current TRCTRAC setting if TRCTRAC has been changed meanwhile.

Display ID: TR$_{ptrac}$

Query form
Short form: TRPTRAC?␣[<numeric_list>trace source]
SCPI form: :FETCh[:SCALar]:SLOTs:CONFig:TRANSient:TRACe?␣[<numeric_list>trace source]
　　　　　:READ[:SCALar]:SLOTs:CONFig:TRANSient:TRACe?␣[<numeric_list>trace source]

[<numeric_list>trace source] Specifies the positions of the elements to query.

　　List elements: 0 ... 15

Response
trace source

Type:　　　　<string program data>,...
List elements: 0 ... 15

9.10.188 TRPVAL
Transient sample values

This command reads the samples of the transient. The size value specifies the number of sample values which can be stored for one transient. The tracks value specifies the number of tracks which are available for one scope. For the transient scope this is the maximum number of tracks. The list parameter specifies, which sample values are wanted. The smalles list index is 0, the biggest one is max number values minus 1.

Display ID: Transient

Query form
Short form: TRPVAL?␣<NRi>trace, [<numeric_list>samples]
SCPI form: :FETCh[:SCALar]:SLOTs:CONFig:TRANSient:VALues?␣<NRi>trace,
　　　　　[<numeric_list>samples]
　　　　　:READ[:SCALar]:SLOTs:CONFig:TRANSient:VALues?␣<NRi>trace,
　　　　　[<numeric_list>samples]

<NRi>trace The trace to inspect

[<numeric_list>samples] Specifies the positions of the elements to query.

　　List elements: 0 ... 16777215 for LMG6X1, 0 ... 16777215 for LMG671 ATE and 0 ... 4194303
　　　　　　　for other devices

Response
The array of samples. 16 traces are allowed with sample values 0..4194303 on LMG6x0 or 0..16777215 on LMG6x1.

Type: $<$NRf$>$,...
List elements: 0 ... 16777215 for LMG6X1, 0 ... 16777215 for LMG671 ATE and 0 ... 4194303 for
                other devices

9.10.189 DURCYCL

Cycle Duration

The global duration of the cycle. Note, this duration is usually different from the individual duration of the synchronized groups (e.g. DURNORM [9.10.197→311]).

Display ID: Δ_{cycle}

Query form
Short form: DURCYCL?
SCPI form: :FETCh[:SCALar]:SLOTs:DEC:DURationCYCLe?
             :READ[:SCALar]:SLOTs:DEC:DURationCYCLe?

Response
The measured value

Type: $<$Time$>$
Unit: s

9.10.190 TSCYCL

Cycle Timestamp

The global timestamp of the cycle. Note, this timestamp is usually different from the individual timestamps of the synchronized groups (e.g. TSNORM [9.10.200→313]). TSCYCLE can aid in merging cycles from multiple devices (little deviation of the internal clocks presumed).

Display ID: ts_{cycle}

Query form
Short form: TSCYCL?
SCPI form: :FETCh[:SCALar]:SLOTs:DEC:TIMestampCYCLe?
             :READ[:SCALar]:SLOTs:DEC:TIMestampCYCLe?

Response
The Timestamp of the cycle

Type: $<$Date$>$

9.10.191 DUREN

Duration of energy measurement

This command reads T_{EN}, the integration time of the energy.

Display ID: Δt_{en}
Suffix: l

Query form
Short form: DUREN?
SCPI form: :FETCh[:SCALar]:SLOTs:ENERgy:DURation?
             :READ[:SCALar]:SLOTs:ENERgy:DURation?

Response

The measured value

Type: $<$Time$>$
Unit: s

9.10.192 DURENINT
Duration of energy measurement of last interval

This command reads T_{EN}, the integration time of the energy of the last interval.

Display ID: $\Delta t_{\mathrm{enint}}$
Suffix: l

Query form
Short form: DURENINT?
SCPI form: :FETCh[:SCALar]:SLOTs:ENERgy:INTervals:DURationint?
:READ[:SCALar]:SLOTs:ENERgy:INTervals:DURationint?

Response
The measured value

Type: $<$Time$>$
Unit: s

9.10.193 TSENINT
Timestap of begin of last interval

This command reads t_S, the time stamp of the begin of the last interval.

Display ID: $\mathrm{TS}_{\mathrm{enint}}$
Suffix: l

Query form
Short form: TSENINT?
SCPI form: :FETCh[:SCALar]:SLOTs:ENERgy:INTervals:TIMestampint?
:READ[:SCALar]:SLOTs:ENERgy:INTervals:TIMestampint?

Response
The measured value

Type: $<$Date$>$

9.10.194 TSEN
Timestap of energy start

This command reads t_S, the time stamp of the energy start.

Display ID: $\mathrm{TS}_{\mathrm{en}}$
Suffix: l

Query form
Short form: TSEN?
SCPI form: :FETCh[:SCALar]:SLOTs:ENERgy:TIMestamp?
:READ[:SCALar]:SLOTs:ENERgy:TIMestamp?

Response

The measured value

Type: <Date>

9.10.195 DURHARM
Duration of the Harmonic Analysis

This command reads the duration of the harmonics analysis.

Display ID: Δt_{harm}
Suffix: l

Query form
Short form: DURHARM?
SCPI form: :FETCh[:SCALar]:SLOTs:HARMonics:DURation?
:READ[:SCALar]:SLOTs:HARMonics:DURation?

Response

The measured value

Type: <Time>
Unit: s

9.10.196 TSHARM
Start Time of the Harmonic Analysis

This command reads the start time of the harmonics analysis.

Display ID: ts_{harm}
Suffix: l

Query form
Short form: TSHARM?
SCPI form: :FETCh[:SCALar]:SLOTs:HARMonics:TIMestamp?
:READ[:SCALar]:SLOTs:HARMonics:TIMestamp?

Response

The measured value

Type: <Date>

9.10.197 DURNORM
Duration of Norm. Analysis

This command reads the duration of measurement interval for the normal values.

Display ID: Δt_{norm}
Suffix: l

Query form
Short form: DURNORM?
SCPI form: :FETCh[:SCALar]:SLOTs:NORMal:DURation?
:READ[:SCALar]:SLOTs:NORMal:DURation?

Response

The measured value

Type: <Time>
Unit: s

9.10.198 DURETAPLOSS
Duration of Eta Ploss Analysis

This command reads the duration of measurement interval for the eta and ploss values.

Display ID: $\Delta t_{etaploss}$
Suffix: eff (1 ... 10)

Query form
Short form: DURETAPLOSS?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:SLOTs:NORMal:EPDURation?␣[<numeric_list>traces]
:READ[:SCALar]:SLOTs:NORMal:EPDURation?␣[<numeric_list>traces]

[<numeric_list>traces] Specifies the positions of the elements to query.

List elements: 0 ... 10

Response

The measured value

Type: <Time>,...
List elements: 0 ... 10
Unit: s

9.10.199 TSETAPLOSS
Start Time of Eta Ploss Analysis

This command reads the starting point of measurement interval for the eta and ploss values.

Display ID: $ts_{etaploss}$
Suffix: eff (1 ... 10)

Query form
Short form: TSETAPLOSS?␣[<numeric_list>traces]
SCPI form: :FETCh[:SCALar]:SLOTs:NORMal:EPTIMestamp?␣[<numeric_list>traces]
:READ[:SCALar]:SLOTs:NORMal:EPTIMestamp?␣[<numeric_list>traces]

[<numeric_list>traces] Specifies the positions of the elements to query.

List elements: 0 ... 10

Response

The measured value

Type: <Date>,...
List elements: 0 ... 10

9.10.200 TSNORM
Start Time of Norm. Analysis

This command reads the starting point of measurement interval for the normal values.

Display ID: ts$_{norm}$
Suffix: l

Query form

Short form: TSNORM?
SCPI form: :FETCh[:SCALar]:SLOTs:NORMal:TIMestamp?
:READ[:SCALar]:SLOTs:NORMal:TIMestamp?

Response

The measured value

Type: <Date>

9.10.201 DURPSI
Duration of PSI values

Duration of PSI values.

Suffix: l

Query form

Short form: DURPSI?
SCPI form: :FETCh[:SCALar]:SLOTs:PSI:DURation?
:READ[:SCALar]:SLOTs:PSI:DURation?

Response

The measured value

Type: <Time>
Unit: s

9.10.202 TSPSI
Timestamp of PSI values

Timestamp of PSI values.

Suffix: l

Query form

Short form: TSPSI?
SCPI form: :FETCh[:SCALar]:SLOTs:PSI:TIMestamp?
:READ[:SCALar]:SLOTs:PSI:TIMestamp?

Response

The measured value

Type: <Date>

9.10.203 INVPSI
Invalid state of PSI values

Invalid state of PSI values.

Suffix: l

Query form
Short form: INVPSI?
SCPI form: :FETCh[:SCALar]:SLOTs:PSI:VALidity?
           :READ[:SCALar]:SLOTs:PSI:VALidity?

Response
The measured value

Type: $<$NRi$>$

Possible values:

bit 0: measurement is questionable
bit 1: measurement is invalid

9.10.204 QUEST
Determine if measurement value is questionable

If QUESTMASK [9.10.394→403] is set to 1, this query tells if the current measurement buffer is considered to provide values of questionable quality.

Query form
Short form: QUEST?
SCPI form: :FETCh[:SCALar]:SLOTs:QUEStionable?
           :READ[:SCALar]:SLOTs:QUEStionable?

Response
Returns 0 if the current measurement buffer provides values according to specification. Otherwise, returns a value greater than 0.

Type: $<$NRi$>$

9.10.205 DURSP
Duration of the Scope

This command reads the duration of the recorded samples by the scope.

Display ID: Δt_{sp}
Suffix: sp (1, 2, 21 and 22)

Query form
Short form: DURSP?
SCPI form: :FETCh[:SCALar]:SLOTs:SCOPe:DURation?
           :READ[:SCALar]:SLOTs:SCOPe:DURation?

Response
The measured value

Type: $<$Time$>$
Unit: s

9.10.206 TSSP
Time of the First Sample

This command reads the point in time of the first sample in the scope.

Display ID: ts$_{\text{sp}}$
Suffix: sp (1, 2, 21 and 22)

Query form
Short form: TSSP?
SCPI form: :FETCh[:SCALar]:SLOTs:SCOPe:TIMestamp?
          :READ[:SCALar]:SLOTs:SCOPe:TIMestamp?

Response
The measured value

Type: <Date>

9.10.207 UAC
AC Value of U

This command reads U_{AC}, the AC value of the voltage defined by

$$U_{\text{AC}} := \sqrt{U_{\text{trms}}^2 - U_{\text{DC}}^2}.$$

See UTRMS [9.10.217→319] and UDC [9.10.210→316] for U_{trms} and U_{DC}.

Display ID: U$_{\text{ac}}$
Suffix: l

Query form
Short form: UAC?
SCPI form: :FETCh[:SCALar][:VOLTage]:AC?
          :READ[:SCALar][:VOLTage]:AC?

Response
The measured value

Type: <NRf>
Unit: V

9.10.208 UCF
Crest Factor of U

This command reads U_{CF}, the crest factor of the voltage defined by

$$U_{\text{CF}} := \frac{U_{\text{peak}}}{U_{\text{trms}}},$$

where U_{trms} is the root mean square value of the voltage UTRMS [9.10.217→319] and U_{peak} is the biggest absolute sample value of the voltage occuring during the time interval given by TSNORM [9.10.200→313] and DURNORM [9.10.197→311]. This can be expressed in an other way as

$$U_{\text{peak}} = \max\{U_{\text{Max}}, |U_{\text{Min}}|\}$$

with U_{Max} the biggest sample UMAX [9.10.212→317] value and $|U_{\text{Min}}|$ the modulus of the smallest sample UMIN [9.10.213→317] value.

Display ID: U_{cf}
Suffix: 1

Query form
Short form: UCF?
SCPI form: :FETCh[:SCALar][:VOLTage]:CFACtor?
:READ[:SCALar][:VOLTage]:CFACtor?

Response
The measured value

Type: <NRf>

9.10.209 UCONST
A constant voltage of 10.0V

This commando reads a constant value of 10.0V

Display ID: U_{CONST}
Suffix: 1

Query form
Short form: UCONST?
SCPI form: :FETCh[:SCALar][:VOLTage]:CONST?
:READ[:SCALar][:VOLTage]:CONST?

Response
The generated value

Type: <NRf>
Unit: V

9.10.210 UDC
DC Value of U

This command reads the DC value of the voltage U_{DC} defined by

$$U_{DC} := \frac{1}{T} \int\limits_{t_1}^{t_1+T} u(t)\ dt,$$

where T is the duration DURNORM [9.10.197→311] and t_1 the starting time TSNORM [9.10.200→313] of the measurement.

Display ID: U_{dc}
Suffix: 1

Query form
Short form: UDC?
SCPI form: :FETCh[:SCALar][:VOLTage]:DC?
:READ[:SCALar][:VOLTage]:DC?

Response
The measured value

Type: <NRf>
Unit: V

9.10.211 UFF
Form Factor of U

This command reads U_{ff}, the form factor of the voltage defined by

$$U_{\text{ff}} := \frac{U_{\text{trms}}}{U_{\text{rect}}},$$

where U_{trms} is the root mean square UTRMS [9.10.217→319] and U_{rect} the rectified value UREC [9.10.215→318] of the voltage.

Display ID: U_{ff}
Suffix: l

Query form
Short form: UFF?
SCPI form: :FETCh[:SCALar][:VOLTage]:FFACtor?
          :READ[:SCALar][:VOLTage]:FFACtor?

Response
The measured value

Type: <NRf>

9.10.212 UMAX
Max. Voltage in Interval

This command reads U_{Max}, the biggest sample value of the voltage occuring during the time interval given by TSNORM [9.10.200→313] and DURNORM [9.10.197→311].

Display ID: U_{max}
Suffix: l

Query form
Short form: UMAX?
SCPI form: :FETCh[:SCALar][:VOLTage]:MAXPk?
          :READ[:SCALar][:VOLTage]:MAXPk?

Response
The measured value

Type: <NRf>
Unit: V

9.10.213 UMIN
Min. Voltage in Interval

This command reads U_{Min}, the smallest sample value of the voltage occuring during the time interval given by TSNORM [9.10.200→313] and DURNORM [9.10.197→311].

Display ID: U_{min}
Suffix: l

Query form
Short form: UMIN?
SCPI form: :FETCh[:SCALar][:VOLTage]:MINPk?
          :READ[:SCALar][:VOLTage]:MINPk?

Response

The measured value

Type: <NRf>
Unit: V

9.10.214 UPP
Peak to Peak of U

This command reads U_{PP}, the peak to peak value of the voltage defined by

$$U_{\mathrm{PP}} := U_{\mathrm{Max}} - U_{\mathrm{Min}},$$

where U_{Max} is the biggest UMAX [9.10.212→317] and U_{Min} the smallest UMIN [9.10.213→317] sample value of the voltage.

Display ID: U_{pp}
Suffix: 1

Query form

Short form: UPP?
SCPI form: :FETCh[:SCALar][:VOLTage]:PPEak?
          :READ[:SCALar][:VOLTage]:PPEak?

Response

The measured value

Type: <NRf>
Unit: V

9.10.215 UREC
Rectified Value of U

This command reads U_{rect}, the rectified value of the voltage defined by

$$U_{\mathrm{rect}} := \frac{1}{T} \int\limits_{t_1}^{t_1+T} |u(t)| \, dt,$$

where T is the duration DURNORM [9.10.197→311] and t_1 the starting time point TSNORM [9.10.200→313] of the measurement.

Display ID: U_{rect}
Suffix: 1

Query form

Short form: UREC?
SCPI form: :FETCh[:SCALar][:VOLTage]:RECTify?
          :READ[:SCALar][:VOLTage]:RECTify?

Response

The measured value

Type: <NRf>
Unit: V

9.10.216 URUSAGE
Usage of voltage range

This command reads the cycle-based range-usage of the voltage.

Display ID: Urusage
Suffix: p (1 ... 7)

Query form
Short form: URUSAGE?
SCPI form: :FETCh[:SCALar][:VOLTage]:RUSage?
        :READ[:SCALar][:VOLTage]:RUSage?

Response
The measured value

Type: <NRf>
Unit: %

9.10.217 UTRMS
RMS of U

This command reads U_{trms}, the root mean square value of the voltage defined by

$$U_{\mathrm{trms}} := \sqrt{\frac{1}{T} \int\limits_{t_1}^{t_1+T} u(t)^2\, dt},$$

where T is the duration DURNORM [9.10.197→311] and t_1 the starting time point TSNORM [9.10.200→313] of the measurement.

Display ID: $\mathrm{U}_{\mathrm{trms}}$
Suffix: l

Query form
Short form: UTRMS?
SCPI form: :FETCh[:SCALar][:VOLTage][:TRMS]?
        :READ[:SCALar][:VOLTage][:TRMS]?

Response
The measured value

Type: <NRf>
Unit: V

9.10.218 FRMT
Output format

Query/Set the output format

Command form
Short form: FRMT␣<NRi>val
SCPI form: :FORMat:DATa␣<NRi>val

| <NRi>val | Output format |

   Possible values:

0 or ASCii: Switches to ASCII output format which is default after a reset of the interface.

1 or PACKed: Switches to a packed, binary format (32 bit little endian) for the output.

Query form

Short form: FRMT?

SCPI form: :FORMat:DATa?

Response

Output format

Type: <NRi>

Possible values:

0: Switches to ASCII output format which is default after a reset of the interface.

1: Switches to a packed, binary format (32 bit little endian) for the output.

9.10.219 CONT
Cont-On control

This command switches on/off a cont-on or deletes it from the system. The first parameter can be on (switches it on), off (switches it off), del (deletes it). The second parameter is the name of the cont-on. In case the first parameter is 'on', there is an optional third parameter indicating the event to trigger on. By default 'mc' (measurement cycle: which is set with the command CYCL) will be used. Every switched off continuous output ends with a "1".

Command form

Short form: CONT␣<string program data>ctrl, [<string program data>actnName], [<string program data>event]

SCPI form: :INITiate:CONTinuous␣<string program data>ctrl, [<string program data>actnName], [<string program data>event]

| <string program data>ctrl | The Control parameter can be: on (switches it on) off (switches it off) del (deletes it)

| [<string program data>actnName] | The name of the action to be used. If none given, the default name "cont" is used.

| [<string program data>event] | The event to trigger on. These are described in INIM, :READ AND :FETCH [9.6→225]

9.10.220 COPY
Immediate reload measurement values

Immediately refreshes the read buffer for the measurement values. This command is identical to INIM "NOW"

Command form

Short form: COPY

SCPI form: :INITiate:COPY

9.10.221 INIM

Reload measurement values

Waits for the next measurement cycle or the event given by its first parameter, and then actualize the read buffer for the measurement values. If no parameter is given, 'mc' is assumed. See INIM, :READ and :FETCH [9.6→225] for a detailed command reference.

Command form
Short form: INIM␣[<string program data>event]
SCPI form: :INITiate:IMMediate␣[<string program data>event]

[<string program data>event] The event to trigger inim on. These are described in INIM, :READ and :FETCH [9.6→225]

9.10.222 SCPL

Signal coupling

Specifies a group's signal coupling. Note that attaching a Plug'n Play sensor (see WIRING USING EXTERNAL CURRENT SENSORS [6.2.4→104]) may change the signal coupling.

Display ID: Coupling
Suffix: g (1 ... 7)

Command form
Short form: SCPL␣<NRi>val
SCPI form: :INPut:COUPling␣<NRi>val

<NRi>val Config Value

*RST value: 0

Possible values:

0 or ACDC: AC+DC coupling
1 or AC: AC coupling, that is DC components are rejected

Query form
Short form: SCPL?
SCPI form: :INPut:COUPling?

Response
Config Value

Type: <NRi>

Possible values:

0: AC+DC coupling
1: AC coupling, that is DC components are rejected

9.10.223 MMCAT

Content of the current directory

Reads the contents of the current folder.

Query form

Short form: MMCAT?␣[<string program data>dir], [<NRi>mode]
SCPI form: :MMEMOry:CATalog?␣[<string program data>dir], [<NRi>mode]

[<string program data>dir] The directory for which to query the content. If left out, use the current directory.

[<NRi>mode] Listing mode

Possible values:

0 or COMPact: Only the name and size (for files) is given for each entry in the directory
1 or LONG: The name, size, permissions and modification date is given for each entry in the directory

Response

A list of all files in the active directory

Type: <string program data>,...
List elements: 0 ... (dynamic size)

9.10.224 MMCDATA
Reads/writes additional data from/to a file

This command reads/writes additional data from/to a file

Command form

Short form: MMCDATA␣<string program data>val, <IEEE Block Program Data>val, [<Boolean>keepopen]
SCPI form: :MMEMOry:CDATa␣<string program data>val, <IEEE Block Program Data>val, [<Boolean>keepopen]

<string program data>val The name of the file to read/write.

<IEEE Block Program Data>val The contents of the file.

[<Boolean>keepopen] Tells the instrument that followup data for this file is expected. Followup data is appended to the file. If true, the MMCLOSE command must be used afterwards to release ressources needed to service this feature.

Query form

Short form: MMCDATA?␣<string program data>val, [<NRi>maxlen]
SCPI form: :MMEMOry:CDATa?␣<string program data>val, [<NRi>maxlen]

<string program data>val The name of the file to read/write.

[<NRi>maxlen] Limits the size of the reply, in bytes. If left out, the entire file is printed. If given, the MMCLOSE command must be used afterwards to release ressources needed to service this feature.

Response

The contents of the file.

Type: <IEEE Block Program Data>

9.10.225 MMCDIR
Changes the active directory

Changes the directory for the mass memory file system.

Command form

Short form: MMCDIR␣<string program data>val
SCPI form: :MMEMOry:CDIRectory␣<string program data>val

$\boxed{\text{<string program data>val}}$ The directory to go to. A double dot (..) as parameter requests to traverse to the parent directory. A slash (/) requests to go to the root directory.

*RST value: "/data"

9.10.226 MMCLOSE
Finishes a file operation to release resources

Finishes a file operation to release resources needed for MMCDATA handling.

Command form

Short form: MMCLOSE␣<string program data>val
SCPI form: :MMEMOry:CLOSe␣<string program data>val

$\boxed{\text{<string program data>val}}$ The file to close

9.10.227 MMCOPY
Copies an existing file to a new file

Copies an existing file to a new file. If the new file already exists, this action will be aborted and an appropiate error will be placed in the error queue.

Command form

Short form: MMCOPY␣<string program data>val, <string program data>val
SCPI form: :MMEMOry:COPY␣<string program data>val, <string program data>val

$\boxed{\text{<string program data>val}}$ The source file.

$\boxed{\text{<string program data>val}}$ The destination file.

9.10.228 MMDATA
Reads/writes a file

This command reads/writes a file.

Command form

Short form: MMDATA␣<string program data>val, <IEEE Block Program Data>val,
[<Boolean>keepopen]
SCPI form: :MMEMOry:DATA␣<string program data>val, <IEEE Block Program Data>val,
[<Boolean>keepopen]

| <string program data>val | The name of the file to read/write.

| <IEEE Block Program Data>val | The contents of the file.

| [<Boolean>keepopen] | Tells the instrument that followup data for this file is expected by use of the MMCDATA. Followup data is appended to the file. If true, the MMCLOSE command must be used afterwards to release ressources needed to service this feature.

Query form

Short form: MMDATA?␣<string program data>val, [<NRi>maxlen]
SCPI form: :MMEMOry:DATA?␣<string program data>val, [<NRi>maxlen]

| <string program data>val | The name of the file to read/write.

| [<NRi>maxlen] | Limits the size of the reply, in bytes. Further bytes may be requested by use of MMCDATA. If left out, the entire file is printed. If given, the MMCLOSE command must be used afterwards to release ressources needed to service this feature.

Response

The contents of the file.

Type: <IEEE Block Program Data>

9.10.229 MMDEL
Removes a file or empty directory

Removes a file or an empty directory.

Command form

Short form: MMDEL␣<string program data>val
SCPI form: :MMEMOry:DELete␣<string program data>val

| <string program data>val | The file or directory to remove.

9.10.230 MMMKDIR
Creates a new directory.

Creates a new directory at the passed location.

Command form

Short form: MMMKDIR␣<string program data>val
SCPI form: :MMEMOry:MKDIR␣<string program data>val

| <string program data>val | The new directory path

9.10.231 MMMOVE

Moves an existing file to a new file location.

Moves an existing file to a new file location. If the new file already exists, this action will be aborted and an appropiate error will be placed in the error queue.

Command form

Short form: MMMOVE␣<string program data>val, <string program data>val
SCPI form: :MMEMOry:MOVE␣<string program data>val, <string program data>val

| <string program data>val | The source file. |

| <string program data>val | The destination file. |

9.10.232 MMOPEN

Opens the file specifies as the parameter.

Opens the file specified as the parameter. Afterwards MMCDATA may be used to read or write data. Optionally, an initial byte position within the file may be given. The file should be closed using MMCLOSE after use.

Command form

Short form: MMOPEN␣<string program data>val, [<NRi>pos]
SCPI form: :MMEMOry:OPEN␣<string program data>val, [<NRi>pos]

| <string program data>val | The file to open |

| [<NRi>pos] | Initial position within the file |

9.10.233 MMRENAME

Renames an existing file or directory.

Renames an existing file or directory, given as first argument, to the name given as second argument. The file stays in its current directory, therefore the name must not contain directory separators. The new name must not yet exist.

Command form

Short form: MMRENAME␣<string program data>val, <string program data>name
SCPI form: :MMEMOry:RENAME␣<string program data>val, <string program data>name

| <string program data>val | The source file or directory. |

| <string program data>name | The new name. |

9.10.234 MMSPACE

Queries the available space of a path.

Results in two numbers. The first is the total capacity of a path in bytes. The second is the number of bytes that is still available for new data.

Query form

Short form: MMSPACE?␣[<string program data>name]
SCPI form: :MMEMOry:SPACE?␣[<string program data>name]

| [<string program data>name] | The path of which to query the space. If omitted, defaults to the current directory.

Response

A list of two values, the first of which is the total capacity in bytes. The second of which is the still available space in bytes.

Type: <NRi>,...
List elements: 0 ... 1

9.10.235 MMSYNC
Synchronizes data to storage device

Data written to a path may be buffered. MMSYNC enforces that the data written so far is on storage media immediately.

Command form

Short form: MMSYNC␣[<string program data>val]
SCPI form: :MMEMOry:SYNC␣[<string program data>val]

| [<string program data>val] | The file to flush

9.10.236 AVER
Average Setting

This command configures the number of Cycle to average over. Only the normal values are affected. Particulary, if averaging is active, the values UMAX [9.10.212→317], IMAX [9.10.35→248], UMIN [9.10.213→317], IMIN [9.10.36→248], UPP [9.10.214→318], IPP [9.10.37→249], and UCF [9.10.208→315], ICF [9.10.31→246] were not computed.

Display ID: Average

Command form

Short form: AVER␣<NRi>val
SCPI form: :SENSe:AVERage:COUNt␣<NRi>val

| <NRi>val | Config Value

    Possible range: 1 ... 99
    *RST value: 1

Query form

Short form: AVER?
SCPI form: :SENSe:AVERage:COUNt?

Response

Config Value

Type: <NRi>
Possible range: 1 ... 99

9.10.237 IDLY
I channel DeLaY correction

Configures a (positive or negative!) *correction* time to compensate for external delays in the signal path, e.g. due to a sensor. Note that delay compensation for compatible ZES sensors is done automatically by the device (using data stored in the sensor's nonvolatile memory). Gain correction is configured via ISCA [9.10.253→334].

Display ID: I Delay
Suffix: p (1 ... 7)

Command form
Short form: IDLY␣<Time>val
SCPI form: :SENSe:CURRent:DELay␣<Time>val

| <Time>val | Config Value |

> Possible range: -5.5e-6 ... 5.5e-6
> *RST value: 0
> Unit: s

Query form
Short form: IDLY?
SCPI form: :SENSe:CURRent:DELay?

Response
Config Value

Type: <Time>
Possible range: -5.5e-6 ... 5.5e-6
Unit: s

9.10.238 IDNI
Human-readable I sensor identification

If no sensor is attached to the channel the string is empty. Otherwise its value consists of 4 comma-separated fields identifying the sensor:

- Manufacturer

- Model

- Serial Number

- Firmware Version

Suffix: p (1 ... 7)

Query form
Short form: IDNI?
SCPI form: :SENSe:CURRent:IDENtify?

Response
Config Value

Type: <string program data>

9.10.239 IJACK
I channel jack

Selects an I channel jack. IJLS [9.10.240→328] provides a list of all currently available jacks or more precisely, their numeric IDs.

Display ID: I Jack
Suffix: p (1 ... 7)

Command form
Short form: IJACK␣<NRi>val
SCPI form: :SENSe:CURRent:JACK␣<NRi>val

| <NRi>val | Config Value

    *RST value: 0

    Possible values:

      0 or ISTar: I* jack
      1 or ISENsor: I$_{sensor}$ jack

Query form
Short form: IJACK?
SCPI form: :SENSe:CURRent:JACK?

Response
Config Value

Type: <NRi>

Possible values:

0: I* jack
1: I$_{sensor}$ jack

9.10.240 IJLS
I channel jack list

Specifies the currently available I channel jacks or more precisely, their numeric IDs. See the documentation for IJACK [9.10.239→328] to decode the values; use IJACK with one of the values to select the appropriate jack.

Suffix: p (1 ... 7)

Query form
Short form: IJLS?
SCPI form: :SENSe:CURRent:LJACk?

Response
Config Value

Type: <NRi>,...
List elements: 0 ... (dynamic size)

9.10.241 IAUTO
I channel auto range

In manual mode the configured range, if valid, is taken as given and stays fixed until changed by a configuration command, for instance explicitly by IRNG [9.10.252→333] or implicitly by

IJACK [9.10.239→328]. With auto range enabled, the instrument monitors the range usage and if necessary automatically selects an appropriate smaller or larger range in order to best measure the signal applied to the selected jack, with the same set of ranges available as in manual mode (see IRNLS [9.10.243→329]).

Display ID: I Auto Range
Suffix: p (1 ... 7)

Command form
Short form: IAUTO␣<Boolean>val
SCPI form: :SENSe:CURRent:RANGe:AUTo␣<Boolean>val

<Boolean>val	Config Value

   *RST value: 1

Query form
Short form: IAUTO?
SCPI form: :SENSe:CURRent:RANGe:AUTo?

Response
Config Value

Type: <Boolean>

9.10.242 IRNMAXLS
List of nominal maximum RMS values of I channel ranges

Specifies the maximum RMS values corresponding to the nominal range values listed in IRNLS [9.10.243→329].

Suffix: p (1 ... 7)

Query form
Short form: IRNMAXLS?
SCPI form: :SENSe:CURRent:RANGe:LNMax?

Response
Config Value

Type: <NRf>,...
List elements: 0 ... (dynamic size)

9.10.243 IRNLS
List of I channel nominal ranges

Specifies the currently available I channel measuring ranges or more precisely, their nominal values. In general, this list changes when a different jack is selected (via IJACK [9.10.239→328]) or a sensor is connected to the channel, etc. For S channels the values listed also depend on IRNGMOD [9.10.250→332]. Use IRNG [9.10.252→333] with one of the values to select the appropriate range. The corresponding maximum RMS and peak values are listed in IRNMAXLS [9.10.242→329] and IRNPKLS [9.10.244→330], respectively.

Suffix: p (1 ... 7)

Query form
Short form: IRNLS?
SCPI form: :SENSe:CURRent:RANGe:LNOMinal?

Response

Config Value

Type: <NRf>,...
List elements: 0 ... (dynamic size)

9.10.244 IRNPKLS
List of nominal peak values of I channel ranges

Specifies the peak values corresponding to the nominal range values listed in IRNLS [9.10.243→329].

Suffix: p (1 ... 7)

Query form

Short form: IRNPKLS?
SCPI form: :SENSe:CURRent:RANGe:LNPeak?

Response

Config Value

Type: <NRf>,...
List elements: 0 ... (dynamic size)

9.10.245 IRNULS
List of units of I channel nominal ranges

Specifies the unit of each currently available I channel nominal measuring range (as given by IRNLS [9.10.243→329]):

- ”A” (Ampere)

- ”V” (Volt)

In general, this list changes whenever IRNLS changes.

Suffix: p (1 ... 7)

Query form

Short form: IRNULS?
SCPI form: :SENSe:CURRent:RANGe:LNUNit?

Response

Config Value

Type: <string program data>,...
List elements: 0 ... (dynamic size)

9.10.246 IRSLS
List of I channel scaled ranges

Specifies the currently available I channel measuring ranges including custom (ISCA [9.10.253→334]) and possibly ZES sensor scaling. In general, this list changes when ISCA changes, a different jack is selected (IJACK [9.10.239→328]) or a sensor is connected to the channel, etc. For S channels the listed values also depende on IRNGMOD [9.10.250→332]. Be aware that when using one of these *scaled* values to select a range via IRNG [9.10.252→333], a match with a *nominal* range value takes precedence. To unambiguously select a range, use its *nominal* value (see IRNLS [9.10.243→329]). The corresponding scaled maximum RMS and peak values are listed in IRSMAXLS [9.10.247→331] and IRSPKLS [9.10.248→331], respectively.

Suffix: p (1 ... 7)

Query form
Short form: IRSLS?
SCPI form: :SENSe:CURRent:RANGe:LSCaled?

Response
Config Value

Type: <NRf>,...
List elements: 0 ... (dynamic size)

9.10.247 IRSMAXLS
List of scaled maximum RMS values of I channel ranges

Specifies the maximum RMS values corresponding to the scaled range values listed in IRSLS [9.10.246→331].

Suffix: p (1 ... 7)

Query form
Short form: IRSMAXLS?
SCPI form: :SENSe:CURRent:RANGe:LSMax?

Response
Config Value

Type: <NRf>,...
List elements: 0 ... (dynamic size)

9.10.248 IRSPKLS
List of scaled peak values of I channel ranges

Specifies the peak values corresponding to the scaled range values listed in IRSLS [9.10.246→331].

Suffix: p (1 ... 7)

Query form
Short form: IRSPKLS?
SCPI form: :SENSe:CURRent:RANGe:LSPeak?

Response
Config Value

Type: <NRf>,...
List elements: 0 ... (dynamic size)

9.10.249 IRSULS

List of units of I channel scaled ranges

Specifies the unit of each currently available I channel scaled measuring range (as given by IRSLS [9.10.246→331]):

- "A" (Ampere)
- "Apk" (Ampere peak)

In general, this list changes whenever IRSLS changes.

Suffix: p (1 ... 7)

Query form

Short form: IRSULS?
SCPI form: :SENSe:CURRent:RANGe:LSUNit?

Response

Config Value

Type: $<$string program data$>$,...
List elements: 0 ... (dynamic size)

9.10.250 IRNGMOD

I range mode

This setting only applies to groups of S channels. The I range mode selects whether the display of I range *nominal* values is geared to AC or DC measurements, including

- IRNG [9.10.252→333],
- IRSCALED [9.10.251→333]
- IRNLS [9.10.243→329], and
- IRSLS [9.10.246→331].

Consequently it also affects which values may be used to select a range via IRNG, that is the values listed in IRNLS and IRSLS.

Display ID: I range mode
Suffix: g (1 ... 7)

Command form

Short form: IRNGMOD␣$<$NRi$>$val
SCPI form: :SENSe:CURRent:RANGe:MODe␣$<$NRi$>$val

$\boxed{<\text{NRi}>\text{val}}$ Config Value

    *RST value: 0

    Possible values:

    0 or AC: AC
    1 or DC: DC

Query form

Short form: IRNGMOD?
SCPI form: :SENSe:CURRent:RANGe:MODe?

Response

Config Value

Type: <NRi>

Possible values:

0: AC
1: DC

9.10.251 IRSCALED
Scaled I channel measuring range

Gives the I channel measuring range including any scaling due to e.g. ISCA [9.10.253→334] and/or a plugged-in ZES sensor. Use IRNG [9.10.252→333] to query the nominal range value without scaling or to set the active range.

Display ID: Scaled I Range
Suffix: p (1 ... 7)

Query form
Short form: IRSCALED?
SCPI form: :SENSe:CURRent:RANGe:RSCaled?

Response
Config Value

Type: <NRf>
Unit: A

9.10.252 IRNG
I channel measuring range

Specifies the selected I channel measuring range. A query always returns the *nominal* range value. Note that for S channels these values depend on IRNGMOD [9.10.250→332]. The scaled range (possibly including custom ISCA [9.10.253→334] and ZES sensor scaling, if any) is given by IRSCALED [9.10.251→333]. When setting the range, either the nominal *or* the scaled value may be used. If the given (nominal or scaled) value matches the nominal value of an available range, that one is selected; otherwise, if the given value matches the scaled value of an available range, that one is selected. A matching nominal value thus takes precedence over a matching scaled value. IRNLS [9.10.243→329] provides a list of currently available I channel ranges or more precisely, their nominal values; IRSLS [9.10.246→331] provides the corresponding scaled values. The user manual includes detailed technical data for each range.

Display ID: I Range
Suffix: p (1 ... 7)

Command form
Short form: IRNG␣<NRf>val
SCPI form: :SENSe:CURRent:RANGe[:UPPer]␣<NRf>val

| <NRf>val | Config Value |

  Unit: A

Query form
Short form: IRNG?
SCPI form: :SENSe:CURRent:RANGe[:UPPer]?

Response
Config Value

Type: <NRf>
Unit: A

9.10.253 ISCA
I channel scaling

Specifies a custom I channel scaling factor. A scaling factor of zero is not permitted. Delay correction is configured via IDLY [9.10.237→327].

Display ID: I Scale
Suffix: p (1 ... 7)

Command form
Short form: ISCA␣<NRf>val
SCPI form: :SENSe:CURRent:SCALe␣<NRf>val

$\boxed{\text{<NRf>val}}$ Config Value

    Possible range: -1e6 ... 1e6
    *RST value: 1

Query form
Short form: ISCA?
SCPI form: :SENSe:CURRent:SCALe?

Response
Config Value

Type: <NRf>
Possible range: -1e6 ... 1e6

9.10.254 ENERGYMOD
Energy control mode

Determines how energy measurement is to be controlled. For an energy measurement to actually run, two conditions must be met:

- The Energy I/O pin of the SYNCHRONIZATION CONNECTOR [4.3→46] (regardless of whether it is configured as an input or output) must be asserted. ENERGYMOD determines how this is controlled (see below).

- At least one group must be selected for energy measurement. This is accomplished by ENERGYSTART [9.10.461→430].

In manual mode, the Energy I/O pin is configured as an output and is asserted automatically by the instrument when groups are selected for energy measurement (via ENERGYSTART [9.10.461→430]); it is deasserted otherwise (that is when no groups are selected, see ENERGYSTOP [9.10.462→431]). The energy measurement is thus effectively controlled by ENERGYSTART and ENERGYSTOP and runs immediately. In external mode, the Energy I/O pin is configured as an input, and whether or not energy measurement runs for any selected groups is determined by the signal level applied to the pin. In schedule mode the Energy I/O pin is again configured as an output. The interval during which it is to be asserted is determined by the ENERGYSCHEDSTART [9.10.256→336] and ENERGYSCHEDSTOP [9.10.257→336] settings. Note that in order for a scheduled energy measurement to actually run at least one group has to be selected for energy measurement in advance (via ENERGYSTART [9.10.461→430]), otherwise the pin will not be asserted (and energy measurement won't run)! A scheduled energy measurement can thus be stopped prematurely by deselecting all groups via ENERGYSTOP [9.10.462→431].

Display ID: Energy mode

Command form
Short form: ENERGYMOD␣<NRi>val
SCPI form: :SENSe:ENERgy:ENABle␣<NRi>val

| <NRi>val | Config Value

> *RST value: 0
>
> Possible values:
>
> **0 or MANual:** Manual control
> **1 or EXTernal:** External control
> **2 or SCHedule:** Schedule control

Query form
Short form: ENERGYMOD?
SCPI form: :SENSe:ENERgy:ENABle?

Response
Config Value

Type: <NRi>

Possible values:

0: Manual control
1: External control
2: Schedule control

9.10.255 ENERGYINTLEN
Energy interval length

The energy interval length. The energy integration results are copied into the interval measurands after running for the given time, and the ENERGYINTERVAL [9.10.80→265] counter is incremented.

Display ID: Interval length

Command form
Short form: ENERGYINTLEN␣<Time>val
SCPI form: :SENSe:ENERgy:ILENgth␣<Time>val

| <Time>val | Config Value

> Possible range: 0 ... 9223372036
> *RST value: 0
> Unit: s

Query form
Short form: ENERGYINTLEN?
SCPI form: :SENSe:ENERgy:ILENgth?

Response
Config Value

Type: <Time>
Possible range: 0 ... 9223372036
Unit: s

9.10.256 ENERGYSCHEDSTART
Energy schedule start

Specifies the start date for a scheduled energy measurement. This setting only takes effect in ENERGYMOD [9.10.254→334] Schedule. The stop date is given by ENERGYSCHEDSTOP [9.10.257→336].

Command form
Short form: ENERGYSCHEDSTART␣<Date>val
SCPI form: :SENSe:ENERgy:SCHedule:STARt␣<Date>val

<Date>val Config Value

Query form
Short form: ENERGYSCHEDSTART?
SCPI form: :SENSe:ENERgy:SCHedule:STARt?

Response
Config Value

Type: <Date>

9.10.257 ENERGYSCHEDSTOP
Energy schedule stop

Specifies the stop date for a scheduled energy measurement. This setting only takes effect in ENERGYMOD [9.10.254→334] Schedule. The start date is given by ENERGYSCHEDSTART [9.10.256→336].

Command form
Short form: ENERGYSCHEDSTOP␣<Date>val
SCPI form: :SENSe:ENERgy:SCHedule:STOP␣<Date>val

<Date>val Config Value

Query form
Short form: ENERGYSCHEDSTOP?
SCPI form: :SENSe:ENERgy:SCHedule:STOP?

Response
Config Value

Type: <Date>

9.10.258 FAUTO
Signal filter automatic mode

If auto filter is enabled, the instrument takes care of selecting appropriate signal filter settings to adapt to the measured signal. In manual mode (that is FAUTO disabled), the signal filters must be set up and adapted manually. This setting only takes effect when single path processing is selected (PROC [9.10.275→345]); the corresponding settings for dual path processing are NFAUTO [9.10.267→341] and WFAUTO [9.10.276→346].

Display ID: Auto Filter
Suffix: g (1 ... 7)

Command form
Short form: FAUTO␣<Boolean>val
SCPI form: :SENSe:FILTer:AUTo␣<Boolean>val

<Boolean>val | Config Value

*RST value: 1

Query form
Short form: FAUTO?
SCPI form: :SENSe:FILTer:AUTo?

Response
Config Value

Type: <Boolean>

9.10.259 BWLS
Wide and Narrow Bandwidth List

Yields the available analog lowpass filter values of a group's channel type. The first value determines the filter frequency of the wideband converter. The second value determines the frequency of the narrowband converter. If a converter is missing for a certain channel type, NaN is given for the respective frequency value.

Display ID: BWLS
Suffix: g (1 ... 7)

Query form
Short form: BWLS?␣[<numeric_list>listVal]
SCPI form: :SENSe:FILTer:BAndwidthLs?␣[<numeric_list>listVal]

[<numeric_list>listVal] | Specifies the positions of the elements to query.

List elements: 0 ... 1

Response
Config Value

Type: <NRf>,...
List elements: 0 ... 1

9.10.260 HPCOF
High pass cutoff frequency

This command sets the cutoff frequency of the highpass filter for single path processing PROC [9.10.275→345]. This setting only takes effect if custom HPFILT [9.10.261→338] is selected with single path processing (PROC [9.10.275→345]). The range of permissible values can be queried via HPCOFRNG [9.10.262→338].

Display ID: High Pass Cut Off
Suffix: g (1 ... 7)

Command form
Short form: HPCOF␣<NRf>val
SCPI form: :SENSe:FILTer:HPASs[:COFRequency]␣<NRf>val

<NRf>val | Config Value

Unit: Hz

Query form
Short form: HPCOF?
SCPI form: :SENSe:FILTer:HPASs[:COFRequency]?

Response
Config Value

Type: <NRf>
Unit: Hz

9.10.261 HPFILT

High-pass filter mode

Sets the high-pass filter mode when working with single path processing PROC [9.10.275→345].

Display ID: High Pass Filter Mode
Suffix: g (1 ... 7)

Command form
Short form: HPFILT␣<NRi>val
SCPI form: :SENSe:FILTer:HPASs:FSTate␣<NRi>val

| <NRi>val | Config Value

Possible values:

0 or OFF: Filter off
1 or CUSTom: Custom digital filter settings

Query form
Short form: HPFILT?
SCPI form: :SENSe:FILTer:HPASs:FSTate?

Response
Config Value

Type: <NRi>

Possible values:

0: Filter off
1: Custom digital filter settings

9.10.262 HPCOFRNG

High-pass filter cutoff frequencies range

Specifies the range of permissible HPCOF [9.10.260→337] settings depending on the filter type selected by HPFILT [9.10.261→338]. The first element contains the minimum value, the second one contains the maximum value. Note that this setting is only meaningful if the respective group is operating in single-path processing mode (see PROC [9.10.275→345]).

Display ID: HPCOFRNG
Suffix: g (1 ... 7)

Query form
Short form: HPCOFRNG?␣[<numeric_list>listVal]
SCPI form: :SENSe:FILTer:HPASs:HPCRange?␣[<numeric_list>listVal]

| [<numeric_list>listVal] | Specifies the positions of the elements to query.

List elements: 0 ... 1

Response
Config Value

Type: <NRf>,...
List elements: 0 ... 1

9.10.263 LPCOFRNG

Low-pass filter cutoff frequency range

Specifies the range of permissible LPCOF [9.10.264→339] settings depending on the filter type selected by LPFILT [9.10.265→340]. The first element contains the minimum value, the second one contains the maximum value. This setting is only meaningful if the respective group is operating in single-path processing mode (see PROC [9.10.275→345]).

Display ID: LPCOFRNG
Suffix: g (1 ... 7)

Query form
Short form: LPCOFRNG?␣[<numeric_list>listVal]
SCPI form: :SENSe:FILTer:HPASs:LPCRange?␣[<numeric_list>listVal]

[<numeric_list>listVal]   Specifies the positions of the elements to query.

List elements: 0 ... 1

Response
Config Value

Type: <NRf>,...
List elements: 0 ... 1

9.10.264 LPCOF

Low-pass cutoff frequency

Sets the low-pass filter cutoff frequency when working with single path processing PROC [9.10.275→345]. This setting only takes effect if custom LPFILT [9.10.265→340] is selected with single path processing (PROC [9.10.275→345]). The range of permissible values can be queried via LPCOFRNG [9.10.263→339].

Display ID: Low Pass Cut Off
Suffix: g (1 ... 7)

Command form
Short form: LPCOF␣<NRf>val
SCPI form: :SENSe:FILTer:LPASs[:COFRequency]␣<NRf>val

<NRf>val   Config Value

Unit: Hz

Query form
Short form: LPCOF?
SCPI form: :SENSe:FILTer:LPASs[:COFRequency]?

Response
Config Value

Type: <NRf>
Unit: Hz

9.10.265 LPFILT

Low-pass filter mode

Sets the low-pass filter mode when working with single path processing. PROC [9.10.275→345].

Display ID: Low Pass Filter Mode
Suffix: g (1 ... 7)

Command form
Short form: LPFILT␣<NRi>val
SCPI form: :SENSe:FILTer:LPASs:FSTate␣<NRi>val

| <NRi>val | Config Value

      Possible values:

    0 or OFF: Filter off
    1 or WIDe: Wideband converter
    2 or NARRow: Narrowband converter
    3 or CUSTom: Custom digital filter settings

Query form
Short form: LPFILT?
SCPI form: :SENSe:FILTer:LPASs:FSTate?

Response
Config Value

Type: <NRi>

Possible values:

0: Filter off
1: Wideband converter
2: Narrowband converter
3: Custom digital filter settings

9.10.266 LPTYP

Low Pass Filter Type

Sets the low-pass filter type when working with single path processing PROC [9.10.275→345]. This setting only takes effect if custom LPFILT [9.10.265→340] is selected with single path processing (PROC [9.10.275→345]).

Display ID: Low-pass filter type
Suffix: g (1 ... 7)

Command form
Short form: LPTYP␣<NRi>val
SCPI form: :SENSe:FILTer:LPASs:TYPe␣<NRi>val

| <NRi>val | Config Value

      Possible values:

    1 or BESSel: Bessel filter
    2 or BUTTerworth: Butterworth filter
    3 or CHEByshev: Chebyshev filter

Query form
Short form: LPTYP?
SCPI form: :SENSe:FILTer:LPASs:TYPe?

Response
Config Value

Type: <NRi>

Possible values:

1: Bessel filter
2: Butterworth filter
3: Chebyshev filter

9.10.267 NFAUTO
Narrowband signal filter automatic mode

If narrow bandwidth auto filter is enabled, the instrument takes care of selecting appropriate *narrow bandwidth* signal filter settings to adapt to the measured signal. With auto filter disabled, the narrow bandwidth filters must be set up and adapted manually. Wide bandwidth auto filter is configured via WFAUTO [9.10.276→346]. These settings only take effect when dual path processing is selected PROC [9.10.275→345]; the corresponding setting for single path processing is FAUTO [9.10.258→336].

Display ID: Narrow Auto Filter
Suffix: g (1 ... 7)

Command form
Short form: NFAUTO␣<Boolean>val
SCPI form: :SENSe:FILTer:NARROWband:AUTo␣<Boolean>val

| <Boolean>val | Config Value |

   *RST value: 0

Query form
Short form: NFAUTO?
SCPI form: :SENSe:FILTer:NARROWband:AUTo?

Response
Config Value

Type: <Boolean>

9.10.268 NHPCOF
High-pass cutoff frequency

This command sets the cutoff frequency of the highpass filter for the narrowband in dual path processing (see also PROC [9.10.275→345]). This setting only takes effect if custom NHPFILT [9.10.269→342] is selected with dual path processing (PROC [9.10.275→345]). The range of permissible values can be queried via NHPCOFRNG [9.10.272→344].

Display ID: Narrow HP Cut Off
Suffix: g (1 ... 7)

Command form
Short form: NHPCOF␣<NRf>val
SCPI form: :SENSe:FILTer:NARROWband:HPASs[:COFRequency]␣<NRf>val

| <NRf>val | Config Value

      *RST value: 1
      Unit: Hz

Query form
Short form: NHPCOF?
SCPI form: :SENSe:FILTer:NARROWband:HPASs[:COFRequency]?

Response
Config Value

Type: <NRf>
Unit: Hz

9.10.269 NHPFILT

Narrow HP Filter Mode

This command sets the highpass filter mode for the narrowband in dual path processing (see also PROC [9.10.275→345]).

Display ID: Narrow HP Filter Mode
Suffix: g (1 ... 7)

Command form
Short form: NHPFILT␣<NRi>val
SCPI form: :SENSe:FILTer:NARROWband:HPASs:FSTate␣<NRi>val

| <NRi>val | Config Value

      *RST value: 0

      Possible values:

      0 or OFF: Filter off
      1 or CUSTom: Custom digital filter settings

Query form
Short form: NHPFILT?
SCPI form: :SENSe:FILTer:NARROWband:HPASs:FSTate?

Response
Config Value

Type: <NRi>

Possible values:

0: Filter off
1: Custom digital filter settings

9.10.270 NLPCOF

low pass cut off frequency

This command sets the cutoff frequency of the lowpass filter for the narrow band of dual processing. This setting only takes effect if custom NLPFILT [9.10.271→343] is selected with dual path processing (PROC [9.10.275→345]). The range of permissible values can be queried via NLPCOFRNG [9.10.273→344].

Display ID: Narrow LP Cut Off
Suffix: g (1 ... 7)

Command form
Short form: NLPCOF␣<NRf>val
SCPI form: :SENSe:FILTer:NARROWband:LPASs[:COFRequency]␣<NRf>val

| <NRf>val | Config Value

  *RST value: 2000
  Unit: Hz

Query form
Short form: NLPCOF?
SCPI form: :SENSe:FILTer:NARROWband:LPASs[:COFRequency]?

Response
Config Value

Type: <NRf>
Unit: Hz

9.10.271 NLPFILT

Narrowband low-pass filter mode

This command sets the lowpass filter mode for the narrowband in dual path processing (see also PROC [9.10.275→345]).

Display ID: Narrow LP Filter Mode
Suffix: g (1 ... 7)

Command form
Short form: NLPFILT␣<NRi>val
SCPI form: :SENSe:FILTer:NARROWband:LPASs:FSTate␣<NRi>val

| <NRi>val | Config Value

  *RST value: 2

  Possible values:

  2 or NARRow: Analog filter only
  3 or CUSTom: Custom digital filter settings

Query form
Short form: NLPFILT?
SCPI form: :SENSe:FILTer:NARROWband:LPASs:FSTate?

Response
Config Value

Type: <NRi>

Possible values:

2: Analog filter only
3: Custom digital filter settings

9.10.272 NHPCOFRNG

──── **Narrow high-pass filter cutoff frequency range**

Specifies the range of permissible NHPCOF [9.10.268→341] settings depending on the filter type selected by NHPFILT [9.10.269→342]. The first element contains the minimum value, the second one contains the maximum value. Note that this setting is only meaningful if the respective group is operating in dual-path processing mode (see PROC [9.10.275→345]).

Display ID: NHPCOFRNG
Suffix: g (1 ... 7)

Query form
Short form: NHPCOFRNG?␣[<numeric_list>listVal]
SCPI form: :SENSe:FILTer:NARROWband:LPASs:HPCRange?␣[<numeric_list>listVal]

| [<numeric_list>listVal] | Specifies the positions of the elements to query.

   List elements: 0 ... 1

Response
Config Value

Type: <NRf>,...
List elements: 0 ... 1

9.10.273 NLPCOFRNG
──── **Narrow low-pass filter cutoff frequency range**

Specifies the range of permissible NLPCOF [9.10.270→342] settings depending on the filter type selected by NLPFILT [9.10.271→343]. The first element contains the minimum value, the second one contains the maximum value. Note that this setting is only meaningful if the respective group is operating in dual-path processing mode (see PROC [9.10.275→345]).

Display ID: NLPCOFRNG
Suffix: g (1 ... 7)

Query form
Short form: NLPCOFRNG?␣[<numeric_list>listVal]
SCPI form: :SENSe:FILTer:NARROWband:LPASs:LPCRange?␣[<numeric_list>listVal]

| [<numeric_list>listVal] | Specifies the positions of the elements to query.

   List elements: 0 ... 1

Response
Config Value

Type: <NRf>,...
List elements: 0 ... 1

9.10.274 NLPTYP
──── **Narrow LP Filter Type**

This command sets the lowpass filter type for the narrowband in dual path processing (see also PROC [9.10.275→345]). This setting only takes effect if custom NLPFILT [9.10.271→343] is selected with dual path processing (PROC [9.10.275→345]).

Display ID: Narrow LP Filter Type
Suffix: g (1 ... 7)

Command form
Short form: NLPTYP␣<NRi>val
SCPI form: :SENSe:FILTer:NARROWband:LPASs:TYPe␣<NRi>val

<NRi>val Config Value

*RST value: 1

Possible values:

1 or BESSel: Bessel filter
2 or BUTTerworth: Butterworth filter
3 or CHEByshev: Chebyshev filter

Query form
Short form: NLPTYP?
SCPI form: :SENSe:FILTer:NARROWband:LPASs:TYPe?

Response
Config Value

Type: <NRi>

Possible values:

1: Bessel filter
2: Butterworth filter
3: Chebyshev filter

9.10.275 PROC
Processing mode

Specifies the processing mode for a group, which provides basic principles as to how signals are sampled and measurement values are then computed. The valid settings are given by PROCLS [9.10.293→355]. Single path processing is the default and always available; the availability of dual path processing depends on the channel type and hardware. When single path operation is selected, only one A/D converter (per U/I channel) is used at a time and the instrument automatically selects the one that best matches the given filter settings. As a consequence, there is only a single set of measurement values which (for convencience) is accessible using *both* narrowband *and* wideband suffixes, that is the bandwidth part of the suffix does not matter. With dual path operation two A/D converters are used simultaneously, producing two separate sets of measurement values. Each bandwidth can be configured individually and the measured values may possibly differ depending on the respective settings. Thus the bandwidth part of the suffix *does* matter here. See BANDWIDTH [6.6→110] for an explanation of the underlying basic concepts.

Display ID: Bandwidth Mode
Suffix: g (1 ... 7)

Command form
Short form: PROC␣<NRi>val
SCPI form: :SENSe:FILTer:PROCessing␣<NRi>val

<NRi>val Config Value

*RST value: 0

Possible values:

0 or SINGle: Use a single (narrow or wide bandwidth) converter only
1 or DUAL: Use both narrow and wide bandwidth converters simultaneously

Query form

Short form: PROC?
SCPI form: :SENSe:FILTer:PROCessing?

Response

Config Value

Type: <NRi>

Possible values:

0: Use a single (narrow or wide bandwidth) converter only
1: Use both narrow and wide bandwidth converters simultaneously

9.10.276 WFAUTO
Wideband signal filter automatic mode

If wide bandwidth auto filter is enabled, the instrument takes care of selecting appropriate *wide bandwidth* signal filter settings to adapt to the measured signal. With auto filter disabled, the wide bandwidth filters must be set up and adapted manually. Narrow bandwidth auto filter is configured via NFAUTO [9.10.267→341]. These settings only take effect when dual path processing is selected PROC [9.10.275→345]; the corresponding setting for single path processing is FAUTO [9.10.258→336].

Display ID: Wide Auto Filter
Suffix: g (1 ... 7)

Command form

Short form: WFAUTO␣<Boolean>val
SCPI form: :SENSe:FILTer:WIDeband:AUTo␣<Boolean>val

| <Boolean>val | Config Value |

*RST value: 0

Query form

Short form: WFAUTO?
SCPI form: :SENSe:FILTer:WIDeband:AUTo?

Response

Config Value

Type: <Boolean>

9.10.277 WLPFILT
Wideband filter mode

This command sets the lowpass filter mode for the wideband in dual path processing (see also PROC [9.10.275→345]).

Display ID: Wide Filter Mode
Suffix: g (1 ... 7)

Command form
Short form: WLPFILT␣<NRi>val
SCPI form: :SENSe:FILTer:WIDeband:LPASs:FSTate␣<NRi>val

| <NRi>val | Config Value

   *RST value: 0

   Possible values:

   0 or OFF: Filter off
   1 or WIDe: Wideband converter

Query form
Short form: WLPFILT?
SCPI form: :SENSe:FILTer:WIDeband:LPASs:FSTate?

Response
Config Value

Type: <NRi>

Possible values:

0: Filter off
1: Wideband converter

9.10.278 FLGROUP

Flicker Group

Specifies the group (number) that is used for flicker measurements (see GROUP [9.10.286→351]). Only non-empty groups are permitted.

Command form
Short form: FLGROUP␣<NRi>val
SCPI form: :SENSe:FLICker:GRoup␣<NRi>val

| <NRi>val | Config Value

   Possible values:

   0 or NONE: No Flicker Group Set
   1 or G1:   Flicker Group 1
   2 or G2:   Flicker Group 2
   3 or G3:   Flicker Group 3
   4 or G4:   Flicker Group 4
   5 or G5:   Flicker Group 5
   6 or G6:   Flicker Group 6
   7 or G7:   Flicker Group 7

Query form
Short form: FLGROUP?
SCPI form: :SENSe:FLICker:GRoup?

Response
Config Value

Type: <NRi>

Possible values:

0: No Flicker Group Set
1: Flicker Group 1

2: Flicker Group 2
3: Flicker Group 3
4: Flicker Group 4
5: Flicker Group 5
6: Flicker Group 6
7: Flicker Group 7

9.10.279 FLLAMP
Flicker lamp model

This value specifies, if the flicker test is for 120 or 230 V.

Command form
Short form: FLLAMP␣<NRi>val
SCPI form: :SENSe:FLICker:LAMP␣<NRi>val

| <NRi>val | Config Value

      Possible values:

        120 or V120: 120V
        230 or V230: 230V

Query form
Short form: FLLAMP?
SCPI form: :SENSe:FLICker:LAMP?

Response
Config Value

Type: <NRi>

Possible values:

120: 120V
230: 230V

9.10.280 FLNUMINT
Number of Pst intervals

This command specifies the number of the short term intervals of the flicker measurement.

Display ID: STInt

Command form
Short form: FLNUMINT␣<NRi>val
SCPI form: :SENSe:FLICker:NInterval␣<NRi>val

| <NRi>val | Config Value

      Possible range: 1 ... 1008
      *RST value: 10

Query form
Short form: FLNUMINT?
SCPI form: :SENSe:FLICker:NInterval?

Response
Config Value

Type: <NRi>
Possible range: 1 ... 1008

9.10.281 FLSYSF
Flicker system frequency

Defines   the   systems   frequency   of   the   power   supply   system   for   the flicker.

Command form
Short form: FLSYSF␣<NRi>val
SCPI form: :SENSe:FLICker:SFRequency␣<NRi>val

$\boxed{\text{<NRi>val}}$ Config Value

     *RST value: 50

     Possible values:

        50 or F50: 50Hz system
        60 or F60: 60Hz system

Query form
Short form: FLSYSF?
SCPI form: :SENSe:FLICker:SFRequency?

Response
Config Value

Type: <NRi>

Possible values:

50: 50Hz system
60: 60Hz system

9.10.282 FLSTINT
Pst interval time

This command specifies the duration of the short term interval of the flicker measurement in seconds.

Display ID: T_{st}

Command form
Short form: FLSTINT␣<Time>val
SCPI form: :SENSe:FLICker:STInterval␣<Time>val

$\boxed{\text{<Time>val}}$ Config Value

     Possible range: 1 ... 900
     *RST value: 600
     Unit: s

Query form
Short form: FLSTINT?
SCPI form: :SENSe:FLICker:STInterval?

Response

Config Value

Type: <Time>
Possible range: 1 ... 900
Unit: s

9.10.283 FLWARM

Flicker meter warmup time

This command specifies the warming up time for the flicker measurement in seconds.

Display ID: T_{warm}

Command form

Short form: FLWARM␣<Time>val
SCPI form: :SENSe:FLICker:WUp␣<Time>val

&boxed;<Time>val&boxed; Config Value

     Possible range: 10 ... 100
     *RST value: 10
     Unit: s

Query form

Short form: FLWARM?
SCPI form: :SENSe:FLICker:WUp?

Response

Config Value

Type: <Time>
Possible range: 10 ... 100
Unit: s

9.10.284 GLCSR

Gapless sampling rate

This command sets the preferred sampling rate of the gapless scope. To fetch the true sampling rate, use the command GLPSR [9.10.170→302].

Command form

Short form: GLCSR␣<NRf>val
SCPI form: :SENSe:GAPLess:SRATe␣<NRf>val

&boxed;<NRf>val&boxed; Config Value

     *RST value: 8000
     Unit: Hz

Query form

Short form: GLCSR?
SCPI form: :SENSe:GAPLess:SRATe?

Response

Config Value

Type: <NRf>
Unit: Hz

9.10.285 GLCTRAC
Gapless tracks' signal sources

Each element of the array specifies the signal for the scope track. For a description of possible signals see SAMPLE SOURCES [6.14.4→124].

Command form

Short form: GLCTRAC␣<numeric_list>listVal, <string program data>val
SCPI form: :SENSe:GAPLess:TRACk␣<numeric_list>listVal, <string program data>val

$\boxed{\text{<numeric\_list>listVal}}$ Specifies the positions of the elements to modify.

List elements: 0 ... 15
*RST value: Positions 0 ... 15.

$\boxed{\text{<string program data>val}}$ Track source

*RST value: ""

Query form

Short form: GLCTRAC?␣<numeric_list>listVal
SCPI form: :SENSe:GAPLess:TRACk?␣<numeric_list>listVal

$\boxed{\text{<numeric\_list>listVal}}$ Specifies the positions of the elements to query.

List elements: 0 ... 15

Response

Config Value

Type: <string program data>,...
List elements: 0 ... 15

9.10.286 GROUP
Partitioning of power channels into groups

Specifies how the instrument's power channels are partitioned into groups. Only P channels of the same type may be grouped together in a single group. Two P channels are of the same type if their respective CTYP [9.10.294→356] strings are equal. The grouping is specified as a string of comma-separated positive integers, for instance "3,3" specifies 2 groups of 3 P channels each for an instrument with 6 P channels in total. Each integer specifies, from left to right, the number of P channels in the group whose suffix is given by the position of that integer in the string: The first integer specifies the size of group 1, the second integer specifies the size of group 2, and so forth. P channels are grouped starting at P channel 1. At least one group must be specified. If the specified grouping is incomplete, that is if it leaves some P channels unassigned, then these will automatically be assigned to one or more additional groups by the instrument. PSI channels cannot be grouped. In general, a change in the instrument's grouping will affect many other settings that depend on the number of P channels in a group, their types etc. It is thus advisable to carefully verify all relevant settings after a grouping has been configured and make sure that they still apply to the measurement task at hand.

Display ID: Grouping

Command form

Short form: GROUP␣<NRi>groupone, [<NRi>grouptwo], [<NRi>groupthree], [<NRi>groupfour], [<NRi>groupfive], [<NRi>groupsix], [<NRi>groupseven]

SCPI form: :SENSe:GROuping[:LIST]␣<NRi>groupone, [<NRi>grouptwo], [<NRi>groupthree], [<NRi>groupfour], [<NRi>groupfive], [<NRi>groupsix], [<NRi>groupseven]

| <NRi>groupone | P channels in group 1

| [<NRi>grouptwo] | P channels in group 2

| [<NRi>groupthree] | P channels in group 3

| [<NRi>groupfour] | P channels in group 4

| [<NRi>groupfive] | P channels in group 5

| [<NRi>groupsix] | P channels in group 6

| [<NRi>groupseven] | P channels in group 7

Query form

Short form: GROUP?
SCPI form: :SENSe:GROuping[:LIST]?

Response

Group parameter as described above

Type: <NRi>,...
List elements: 0 ... (dynamic size)

9.10.287 HAAL

Harmonics Anti-Aliasing

Determines wether aliasing is prevented automatically or must be taken care of externally by the instrument's operator.

Display ID: Aliasing
Suffix: g (1 ... 7)

Command form

Short form: HAAL␣<NRi>val
SCPI form: :SENSe:HARMonics:AALiasing␣<NRi>val

| <NRi>val | Config Value

*RST value: 0

Possible values:

0 or AUTo: Aliasing is prevented automatically.
1 or CUSTom: Aliasing must be considered/prevented externally by the instrument's operator.

Query form

Short form: HAAL?
SCPI form: :SENSe:HARMonics:AALiasing?

Response

Config Value

Type: <NRi>

Possible values:

0: Aliasing is prevented automatically.
1: Aliasing must be considered/prevented externally by the instrument's operator.

9.10.288 INTERHARM
Number of proper interharmonics.

This command sets the number of proper interharmonics.

Display ID: Interharm
Suffix: g (1 ... 7)

Command form
Short form: INTERHARM␣<NRi>val
SCPI form: :SENSe:HARMonics:INTerharm␣<NRi>val

| <NRi>val | Config Value

     Possible range: 0 ... 59
     *RST value: 0

Query form
Short form: INTERHARM?
SCPI form: :SENSe:HARMonics:INTerharm?

Response
Config Value

Type: <NRi>
Possible range: 0 ... 59

9.10.289 HMAXNUM
Maximum number of harmonics

If HNUMMODE [9.10.290→354] is set to Custom, this command sets the number of harmonics (including interharmonics) to a fixed value. See also HNUM [9.10.115→279].

Display ID: Maximum Number of Harmonics
Suffix: g (1 ... 7)

Command form
Short form: HMAXNUM␣<NRi>val
SCPI form: :SENSe:HARMonics:NMAX␣<NRi>val

| <NRi>val | Config Value

     Possible range: 0 ... 2000
     *RST value: 15

Query form
Short form: HMAXNUM?
SCPI form: :SENSe:HARMonics:NMAX?

Response
Config Value

Type: <NRi>
Possible range: 0 ... 2000

9.10.290 HNUMMODE

Mode to determine the number of harmonics

This command specifies, if the number of harmonics, that can be fechted by HNUM [9.10.115→279], is determined by the filter or is determined by the customers specified value HMAXNUM [9.10.289→353].

Display ID: Number of Harmonics Mode
Suffix: g (1 ... 7)

Command form
Short form: HNUMMODE␣<NRi>val
SCPI form: :SENSe:HARMonics:NMODe␣<NRi>val

| <NRi>val | Config Value

> *RST value: 0

> Possible values:

> **0 or Filter:** The number of harmonics is determined by the signal filter.
> **1 or Custom:** The number of harmonics is determined by HMAXNUM [9.10.289→353].

Query form
Short form: HNUMMODE?
SCPI form: :SENSe:HARMonics:NMODe?

Response
Config Value

Type: <NRi>

Possible values:

0: The number of harmonics is determined by the signal filter.
1: The number of harmonics is determined by HMAXNUM [9.10.289→353].

9.10.291 HREF

Harmonics reference phase

This command specifies the reference for the phase angles of the harmonics.

Display ID: Harmonics Reference Phase
Suffix: g (1 ... 7)

Command form
Short form: HREF␣<NRi>val
SCPI form: :SENSe:HARMonics:REFerence␣<NRi>val

| <NRi>val | Config Value

> *RST value: 0

> Possible values:

> **0 or U:** The reference for the phase angles of the harmonics is the voltage
> **1 or I:** The reference for the phase angles of the harmonics is the current
> **2 or SYNC:** The reference for the phase angles of the harmonics is real measuring window, as determined by the synchronization signal

Query form
Short form: HREF?
SCPI form: :SENSe:HARMonics:REFerence?

Response
Config Value

Type: <NRi>

Possible values:

0: The reference for the phase angles of the harmonics is the voltage
1: The reference for the phase angles of the harmonics is the current
2: The reference for the phase angles of the harmonics is real measuring window, as determined by the synchronization signal

9.10.292 SPECTMODE
Spectrum mode

Determines if the time interval of the Fourier analysis is based on the group sync or on the cycle length.

Display ID: Spectral Mode
Suffix: g (1 ... 7)

Command form
Short form: SPECTMODE␣<NRi>val
SCPI form: :SENSe:HARMonics:SPECtrummode␣<NRi>val

| <NRi>val | Config Value

  *RST value: 0

  Possible values:

  0 or HARMonics: Based on the group sync.
  1 or FREQuency: Based on the cycle length.

Query form
Short form: SPECTMODE?
SCPI form: :SENSe:HARMonics:SPECtrummode?

Response
Config Value

Type: <NRi>

Possible values:

0: Based on the group sync.
1: Based on the cycle length.

9.10.293 PROCLS
List of available processing modes

Specifies the available processing modes for a group, depending on the type of its P channel members. Each element holds a value equal to one of the processing modes specified by PROC [9.10.275→345], which is also used to set the respective mode for a group. The type of a P channel can be accessed via CTYP [9.10.294→356]. Single path processing is supported by all P channel types. See TECHNICAL DATA [4→39] for whether a certain P channel type also supports *dual* path processing. An explanation of the underlying basic concepts can be found in BANDWIDTH [6.6→110].

Suffix: g (1 ... 7)

Query form
Short form: PROCLS?
SCPI form: :SENSe:LPRocessing?

Response
Config Value

Type: <NRi>,...
List elements: 0 ... (dynamic size)

9.10.294 CTYP
Channel type

Specifies the channel type as a human-readable string. An empty string means that the channel is not available.

Suffix: p (1 ... 7)

Query form
Short form: CTYP?
SCPI form: :SENSe:POWer[:CHANnel]:TYPe?

Response
Config Value

Type: <string program data>

9.10.295 PSIAIXA
scaling point xa of slow analog input

Scaling point xa for linear scaling of slow analog input.

Suffix: ai (1 ... 8, 101 ... 108 and 201 ... 208)

Command form
Short form: PSIAIXA␣<NRf>val
SCPI form: :SENSe:PSI:AIN:XA␣<NRf>val

| <NRf>val | Config Value

　　*RST value: 0

Query form
Short form: PSIAIXA?
SCPI form: :SENSe:PSI:AIN:XA?

Response
Config Value

Type: <NRf>

9.10.296 PSIAIXB
scaling point xb of slow analog input

Scaling point xb for linear scaling of slow analog input.

Suffix: ai (1 ... 8, 101 ... 108 and 201 ... 208)

Command form
Short form: PSIAIXB␣<NRf>val
SCPI form: :SENSe:PSI:AIN:XB␣<NRf>val

| <NRf>val | Config Value

    *RST value: 10

Query form
Short form: PSIAIXB?
SCPI form: :SENSe:PSI:AIN:XB?

Response
Config Value

Type: <NRf>

9.10.297 PSIAIYA
scaling point ya of slow analog input

Scaling point ya for linear scaling of slow analog input.

Suffix: ai (1 ... 8, 101 ... 108 and 201 ... 208)

Command form
Short form: PSIAIYA␣<NRf>val
SCPI form: :SENSe:PSI:AIN:YA␣<NRf>val

| <NRf>val | Config Value

    *RST value: 0

Query form
Short form: PSIAIYA?
SCPI form: :SENSe:PSI:AIN:YA?

Response
Config Value

Type: <NRf>

9.10.298 PSIAIYB
scaling point yb of slow analog input

Scaling point yb for linear scaling of slow analog input.

Suffix: ai (1 ... 8, 101 ... 108 and 201 ... 208)

Command form
Short form: PSIAIYB␣<NRf>val
SCPI form: :SENSe:PSI:AIN:YB␣<NRf>val

| <NRf>val | Config Value

    *RST value: 10

Query form
Short form: PSIAIYB?
SCPI form: :SENSe:PSI:AIN:YB?

Response
Config Value

Type: <NRf>

9.10.299 PSIAOSOURCE
source of analog output

Source of analog input.

Suffix: ao (1 ... 32, 101 ... 132 and 201 ... 232)

Command form
Short form: PSIAOSOURCE␣<string program data>val
SCPI form: :SENSe:PSI:AOUT:SOURce␣<string program data>val

| <string program data>val | Config Value |

　　*RST value: "utrms1111"

Query form
Short form: PSIAOSOURCE?
SCPI form: :SENSe:PSI:AOUT:SOURce?

Response
Config Value

Type: <string program data>

9.10.300 PSIAOXA
scaling point xa of analog output

Scaling point xa for linear scaling of analog output.

Suffix: ao (1 ... 32, 101 ... 132 and 201 ... 232)

Command form
Short form: PSIAOXA␣<NRf>val
SCPI form: :SENSe:PSI:AOUT:XA␣<NRf>val

| <NRf>val | Config Value |

　　*RST value: 0

Query form
Short form: PSIAOXA?
SCPI form: :SENSe:PSI:AOUT:XA?

Response
Config Value

Type: <NRf>

9.10.301 PSIAOXB
scaling point xb of analog output

Scaling point xb for linear scaling of analog output.

Suffix: ao (1 ... 32, 101 ... 132 and 201 ... 232)

Command form
Short form: PSIAOXB␣<NRf>val
SCPI form: :SENSe:PSI:AOUT:XB␣<NRf>val

$\boxed{\text{<NRf>val}}$ Config Value

*RST value: 10

Query form
Short form: PSIAOXB?
SCPI form: :SENSe:PSI:AOUT:XB?

Response
Config Value

Type: <NRf>

9.10.302 PSIAOYA
scaling point ya of analog output

Scaling point ya for linear scaling of analog output.

Suffix: ao (1 ... 32, 101 ... 132 and 201 ... 232)

Command form
Short form: PSIAOYA␣<NRf>val
SCPI form: :SENSe:PSI:AOUT:YA␣<NRf>val

$\boxed{\text{<NRf>val}}$ Config Value

*RST value: 0

Query form
Short form: PSIAOYA?
SCPI form: :SENSe:PSI:AOUT:YA?

Response
Config Value

Type: <NRf>

9.10.303 PSIAOYB
scaling point yb of analog output

Scaling point yb for linear scaling of analog output.

Suffix: ao (1 ... 32, 101 ... 132 and 201 ... 232)

Command form
Short form: PSIAOYB␣<NRf>val
SCPI form: :SENSe:PSI:AOUT:YB␣<NRf>val

$\boxed{\text{<NRf>val}}$ Config Value

*RST value: 10

Query form
Short form: PSIAOYB?
SCPI form: :SENSe:PSI:AOUT:YB?

Response
Config Value

Type: <NRf>

9.10.304 PSIRESETALLCOUNT
Resets all PSI digitial input counter

This Command resets all PSI digital input pulse counters.

Command form
Short form: PSIRESETALLCOUNT
SCPI form: :SENSe:PSI:DIN:ALLCountreset

9.10.305 PSIDIINV
digital input invert

Inverts the state of the digital input

Suffix: di (1 ... 8, 101 ... 108 and 201 ... 208)

Command form
Short form: PSIDIINV␣<Boolean>val
SCPI form: :SENSe:PSI:DIN:INVert␣<Boolean>val

| <Boolean>val | Config Value |

   *RST value: 0

Query form
Short form: PSIDIINV?
SCPI form: :SENSe:PSI:DIN:INVert?

Response
Config Value

Type: <Boolean>

9.10.306 PSIRESETCOUNT
Resets one PSI digitial input counter

This Command resets one psi digital input pulse counter.

Command form
Short form: PSIRESETCOUNT␣<NRi>value
SCPI form: :SENSe:PSI:DIN:RESetcount␣<NRi>value

| <NRi>value | The digital input pin |

9.10.307 PSIFAIXA
scaling point xa of fast analog input

Scaling point xa for linear scaling of fast analog input.

Suffix: afi (1 ... 2, 101 ... 102 and 201 ... 202)

Command form
Short form: PSIFAIXA␣<NRf>val
SCPI form: :SENSe:PSI:FAIN:XA␣<NRf>val

| <NRf>val | Config Value

*RST value: 0

Query form
Short form: PSIFAIXA?
SCPI form: :SENSe:PSI:FAIN:XA?

Response
Config Value

Type: <NRf>

9.10.308 PSIFAIXB
scaling point xb of fast analog input

Scaling point xb for linear scaling of fast analog input.

Suffix: afi (1 ... 2, 101 ... 102 and 201 ... 202)

Command form
Short form: PSIFAIXB␣<NRf>val
SCPI form: :SENSe:PSI:FAIN:XB␣<NRf>val

| <NRf>val | Config Value

*RST value: 10

Query form
Short form: PSIFAIXB?
SCPI form: :SENSe:PSI:FAIN:XB?

Response
Config Value

Type: <NRf>

9.10.309 PSIFAIYA
scaling point ya of fast analog input

Scaling point ya for linear scaling of fast analog input.

Suffix: afi (1 ... 2, 101 ... 102 and 201 ... 202)

Command form
Short form: PSIFAIYA␣<NRf>val
SCPI form: :SENSe:PSI:FAIN:YA␣<NRf>val

| <NRf>val | Config Value

*RST value: 0

Query form
Short form: PSIFAIYA?
SCPI form: :SENSe:PSI:FAIN:YA?

Response
Config Value

Type: <NRf>

9.10.310 PSIFAIYB
scaling point yb of fast analog input

Scaling point yb for linear scaling of fast analog input.

Suffix: afi (1 ... 2, 101 ... 102 and 201 ... 202)

Command form
Short form: PSIFAIYB␣<NRf>val
SCPI form: :SENSe:PSI:FAIN:YB␣<NRf>val

| <NRf>val | Config Value

     *RST value: 10

Query form
Short form: PSIFAIYB?
SCPI form: :SENSe:PSI:FAIN:YB?

Response
Config Value

Type: <NRf>

9.10.311 PSIFILT
PSI fast analog input filter mode

Specifies how the filter parameters for a PSI's fast analog inputs are determined.

Display ID: PSI$_{\text{FAI-mode}}$
Suffix: psi (1, 101 and 201)

Command form
Short form: PSIFILT␣<NRi>val
SCPI form: :SENSe:PSI:FILTER:FSTate␣<NRi>val

| <NRi>val | Config Value

     *RST value: 0

     Possible values:

0 or GRoup: Use filter settings from reference group (PSIGROUP [9.10.327→371])
1 or PSi: Use PSI-specific filter settings (see PSILPFILT [9.10.316→365] and
             PSIHPFILT [9.10.313→363])

Query form

Short form: PSIFILT?
SCPI form: :SENSe:PSI:FILTER:FSTate?

Response

Config Value

Type: <NRi>

Possible values:

0: Use filter settings from reference group (PSIGROUP [9.10.327→371])
1: Use PSI-specific filter settings (see PSILPFILT [9.10.316→365] and PSIHPFILT [9.10.313→363])

9.10.312 PSIHPCOF

PSI fast analog input high-pass cutoff frequency

Sets the high-pass cutoff frequency for a PSI's fast analog inputs. This setting only takes effect if a PSI-specific (via PSIFILT [9.10.311→362]) custom PSIHPFILT [9.10.313→363] is selected.

Display ID: PSI$_{\text{FAI-HP-cutoff}}$
Suffix: psi (1, 101 and 201)

Command form
Short form: PSIHPCOF␣<NRf>val
SCPI form: :SENSe:PSI:FILTER:HPASs[:COFRequency]␣<NRf>val

| <NRf>val | Config Value

    Possible range: 0.1 ... 15000
    *RST value: 1
    Unit: Hz

Query form
Short form: PSIHPCOF?
SCPI form: :SENSe:PSI:FILTER:HPASs[:COFRequency]?

Response
Config Value

Type: <NRf>
Possible range: 0.1 ... 15000
Unit: Hz

9.10.313 PSIHPFILT

PSI fast analog input high-pass filter mode

Sets the high-pass filter mode for a PSI's fast analog inputs. This settings only takes effect if PSI-specific filters are selected via PSIFILT [9.10.311→362].

Display ID: PSI$_{\text{FAI-HP-mode}}$
Suffix: psi (1, 101 and 201)

Command form
Short form: PSIHPFILT␣<NRi>val
SCPI form: :SENSe:PSI:FILTER:HPASs:FSTate␣<NRi>val

| <NRi>val | Config Value

    *RST value: 0

    Possible values:

    0 or OFF: Filter off
    1 or CUSTom: Custom digital filter settings

Query form
Short form: PSIHPFILT?
SCPI form: :SENSe:PSI:FILTER:HPASs:FSTate?

Response
Config Value

Type: <NRi>

Possible values:

0: Filter off
1: Custom digital filter settings

9.10.314 PSIHPTYP

PSI fast analog input high-pass filter type

This sets the high-pass filter type for a PSI's fast analog inputs. This settings only takes effect if a PSI-specific (via PSIFILT [9.10.311→362]) custom PSIHPFILT [9.10.313→363] is selected.

Display ID: PSI$_{\text{FAI-HP-type}}$
Suffix: psi (1, 101 and 201)

Command form
Short form: PSIHPTYP␣<NRi>val
SCPI form: :SENSe:PSI:FILTER:HPASs:TYPe␣<NRi>val

| <NRi>val | Config Value

*RST value: 1

Possible values:

1 or RC: RC filter

Query form
Short form: PSIHPTYP?
SCPI form: :SENSe:PSI:FILTER:HPASs:TYPe?

Response
Config Value

Type: <NRi>

Possible values:

1: RC filter

9.10.315 PSILPCOF

PSI fast analog input low-pass cutoff frequency

Sets the low-pass filter cutoff frequency for a PSI's fast analog inputs. This setting only takes effect if a PSI-specific (via PSIFILT [9.10.311→362]) custom PSILPFILT [9.10.316→365] is selected.

Display ID: PSI$_{\text{FAI-LP-cutoff}}$
Suffix: psi (1, 101 and 201)

Command form
Short form: PSILPCOF␣<NRf>val
SCPI form: :SENSe:PSI:FILTER:LPASs[:COFRequency]␣<NRf>val

| <NRf>val | Config Value

Possible range: 0.1 ... 15000
*RST value: 2000
Unit: Hz

Query form
Short form: PSILPCOF?
SCPI form: :SENSe:PSI:FILTER:LPASs[:COFRequency]?

Response
Config Value

Type: <NRf>
Possible range: 0.1 ... 15000
Unit: Hz

9.10.316 PSILPFILT
PSI fast analog input low-pass filter mode

Sets the low-pass filter mode for a PSI's fast analog inputs. This setting only takes effect if PSI-specific filters are selected via PSIFILT [9.10.311→362].

Display ID: PSI$_{\text{FAI-LP-mode}}$
Suffix: psi (1, 101 and 201)

Command form
Short form: PSILPFILT⌴<NRi>val
SCPI form: :SENSe:PSI:FILTER:LPASs:FSTate⌴<NRi>val

| <NRi>val | Config Value |

*RST value: 0

Possible values:

0 or NARRow: Narrow bandwidth (see FAST ANALOG INPUTS [4.12.1→79])
1 or CUSTom: Custom digital filter settings

Query form
Short form: PSILPFILT?
SCPI form: :SENSe:PSI:FILTER:LPASs:FSTate?

Response
Config Value

Type: <NRi>

Possible values:

0: Narrow bandwidth (see FAST ANALOG INPUTS [4.12.1→79])
1: Custom digital filter settings

9.10.317 PSILPTYP
PSI fast analog input low-pass filter type

This sets the low-pass filter type for a PSI's fast analog inputs. This settings only takes effect if a PSI-specific (via PSIFILT [9.10.311→362]) custom PSILPFILT [9.10.316→365] is selected.

Display ID: PSI$_{\text{FAI-LP-type}}$
Suffix: psi (1, 101 and 201)

Command form
Short form: PSILPTYP␣<NRi>val
SCPI form: :SENSe:PSI:FILTER:LPASs:TYPe␣<NRi>val

| <NRi>val | Config Value

    *RST value: 1

    Possible values:

       1 or BESSel: Bessel filter
       2 or BUTTerworth: Butterworth filter
       3 or CHEByshev: Chebyshev filter

Query form
Short form: PSILPTYP?
SCPI form: :SENSe:PSI:FILTER:LPASs:TYPe?

Response
Config Value

Type: <NRi>

Possible values:

1: Bessel filter
2: Butterworth filter
3: Chebyshev filter

9.10.318 PSIDUALFRQFILT

PSI frequency input filter

Selects the filter setting for frequency input pins in dual motor mode. See also SPEED/TORQUE/FREQUENCY INPUTS [4.12.6→82].

Display ID: PSI$_{\text{FRQ FILT}}$
Suffix: motor (1 ... 2, 101 ... 102 and 201 ... 202)

Command form
Short form: PSIDUALFRQFILT␣<NRi>val
SCPI form: :SENSe:PSI:FREQuency:DUAL:FILTer␣<NRi>val

| <NRi>val | Config Value

    *RST value: 0

    Possible values:

       0 or OFF: Turn filter off
       1 or LOW: Low cutoff frequency
       2 or MEDium: Medium cutoff frequency
       3 or HIGH: High cutoff frequency

Query form
Short form: PSIDUALFRQFILT?
SCPI form: :SENSe:PSI:FREQuency:DUAL:FILTer?

Response
Config Value

Type: <NRi>

Possible values:

0: Turn filter off
1: Low cutoff frequency
2: Medium cutoff frequency
3: High cutoff frequency

9.10.319 PSIDUALFRQTYPE
PSI frequency input signal type

Selects the type of input signal for PSI speed/torque/frequency input pins in dual motor mode. See also SPEED/TORQUE/FREQUENCY INPUTS [4.12.6→82].

Suffix: motor (1 ... 2, 101 ... 102 and 201 ... 202)

Command form
Short form: PSIDUALFRQTYPE␣<NRi>val
SCPI form: :SENSe:PSI:FREQuency:DUAL:TYPE␣<NRi>val

| <NRi>val | Config Value |

    *RST value: 0

Possible values:

0 or TTL: Signal must be applied against ground (single-ended)
1 or HTL: Signal must be applied against ground (single-ended)
2 or RS422: Signal must be applied against the inverted input (differential)

Query form
Short form: PSIDUALFRQTYPE?
SCPI form: :SENSe:PSI:FREQuency:DUAL:TYPE?

Response
Config Value

Type: <NRi>

Possible values:

0: Signal must be applied against ground (single-ended)
1: Signal must be applied against ground (single-ended)
2: Signal must be applied against the inverted input (differential)

9.10.320 PSIFRQFILT
PSI frequency input filter

Selects the filter setting for frequency input pins in single motor mode. See also SPEED/TORQUE/FREQUENCY INPUTS [4.12.6→82].

Display ID: PSI$_{\text{FRQ FILT}}$
Suffix: fi (1 ... 2, 101 ... 102 and 201 ... 202)

Command form
Short form: PSIFRQFILT␣<NRi>val
SCPI form: :SENSe:PSI:FREQuency:FILTer␣<NRi>val

| <NRi>val | Config Value |

    *RST value: 0

Possible values:

0 or OFF:	Turn filter off
1 or LOW:	Low cutoff frequency
2 or MEDium:	Medium cutoff frequency
3 or HIGH:	High cutoff frequency

Query form

Short form: PSIFRQFILT?
SCPI form: :SENSe:PSI:FREQuency:FILTer?

Response

Config Value

Type: <NRi>

Possible values:

0: Turn filter off
1: Low cutoff frequency
2: Medium cutoff frequency
3: High cutoff frequency

9.10.321 PSIFRQDIRINV
Invert direction of PSI digital speed sensor

Invert the direction of the PSI digital speed sensor.

Suffix: fi (1 ... 2, 101 ... 102 and 201 ... 202)

Command form

Short form: PSIFRQDIRINV␣<Boolean>val
SCPI form: :SENSe:PSI:FREQuency:INVert␣<Boolean>val

<Boolean>val	Config Value

*RST value: 0

Query form

Short form: PSIFRQDIRINV?
SCPI form: :SENSe:PSI:FREQuency:INVert?

Response

Config Value

Type: <Boolean>

9.10.322 PSIFRQTYPE
PSI frequency input signal type

Selects the type of input signal for PSI speed/torque/frequency input pins in single motor mode. See also SPEED/TORQUE/FREQUENCY INPUTS [4.12.6→82].

Suffix: fi (1 ... 2, 101 ... 102 and 201 ... 202)

Command form
Short form: PSIFRQTYPE␣<NRi>val
SCPI form: :SENSe:PSI:FREQuency:TYPE␣<NRi>val

| <NRi>val | Config Value

    *RST value: 0

    Possible values:

        0 or TTL: Signal must be applied against ground (single-ended)
        1 or HTL: Signal must be applied against ground (single-ended)
        2 or RS422: Signal must be applied against the inverted input (differential)

Query form
Short form: PSIFRQTYPE?
SCPI form: :SENSe:PSI:FREQuency:TYPE?

Response
Config Value

Type: <NRi>

Possible values:

0: Signal must be applied against ground (single-ended)
1: Signal must be applied against ground (single-ended)
2: Signal must be applied against the inverted input (differential)

9.10.323 PSIFRQXA
scaling point xa of frequency input

Scaling point xa for the linear scaling of the frequency input.

Suffix: fi (1 ... 2, 101 ... 102 and 201 ... 202)

Command form
Short form: PSIFRQXA␣<NRf>val
SCPI form: :SENSe:PSI:FREQuency:XA␣<NRf>val

| <NRf>val | Config Value

    *RST value: 0

Query form
Short form: PSIFRQXA?
SCPI form: :SENSe:PSI:FREQuency:XA?

Response
Config Value

Type: <NRf>

9.10.324 PSIFRQXB
scaling point xb of frequency input

Scaling point xb for the linear scaling of the frequency input.

Suffix: fi (1 ... 2, 101 ... 102 and 201 ... 202)

Command form
Short form: PSIFRQXB␣<NRf>val
SCPI form: :SENSe:PSI:FREQuency:XB␣<NRf>val

| <NRf>val | Config Value

  *RST value: 10000

Query form
Short form: PSIFRQXB?
SCPI form: :SENSe:PSI:FREQuency:XB?

Response
Config Value

Type: <NRf>

9.10.325 PSIFRQYA
scaling point ya of frequency input

Scaling point ya for the linear scaling of the frequency input.

Suffix: fi (1 ... 2, 101 ... 102 and 201 ... 202)

Command form
Short form: PSIFRQYA␣<NRf>val
SCPI form: :SENSe:PSI:FREQuency:YA␣<NRf>val

| <NRf>val | Config Value

  *RST value: 0

Query form
Short form: PSIFRQYA?
SCPI form: :SENSe:PSI:FREQuency:YA?

Response
Config Value

Type: <NRf>

9.10.326 PSIFRQYB
scaling point yb of frequency input

Scaling point yb for the linear scaling of the frequency input.

Suffix: fi (1 ... 2, 101 ... 102 and 201 ... 202)

Command form
Short form: PSIFRQYB␣<NRf>val
SCPI form: :SENSe:PSI:FREQuency:YB␣<NRf>val

| <NRf>val | Config Value

  *RST value: 10000

Query form
Short form: PSIFRQYB?
SCPI form: :SENSe:PSI:FREQuency:YB?

Response

Config Value

Type: <NRf>

9.10.327 PSIGROUP
PSI reference group

Reference group for the synchronization of the PSI.

Suffix: psi (1, 101 and 201)

Command form

Short form: PSIGROUP␣<NRi>val
SCPI form: :SENSe:PSI:GRoup␣<NRi>val

| <NRi>val | Config Value

*RST value: 1

Query form

Short form: PSIGROUP?
SCPI form: :SENSe:PSI:GRoup?

Response

Config Value

Type: <NRi>

9.10.328 IDIQGROUP
id/iq group

This command specifies the group, that provides the Id-Iq-transformation with the samples of the current It is necessary, that the group contains three power channels. For more information see ID-IQ-TRANSFORMATION [6.21.8→143].

Suffix: psi (1, 101 and 201)

Command form

Short form: IDIQGROUP␣<NRi>val
SCPI form: :SENSe:PSI:IDIQ:GRoup␣<NRi>val

| <NRi>val | Config Value

Possible values:

0 or NONE: No group selected
1 or G1: Group 1
2 or G2: Group 2
3 or G3: Group 3
4 or G4: Group 4
5 or G5: Group 5
6 or G6: Group 6
7 or G7: Group 7

Query form

Short form: IDIQGROUP?
SCPI form: :SENSe:PSI:IDIQ:GRoup?

Response

Config Value

Type: <NRi>

Possible values:

0: No group selected
1: Group 1
2: Group 2
3: Group 3
4: Group 4
5: Group 5
6: Group 6
7: Group 7

9.10.329 PSIPOLPAIR
Number of pole pairs

This sets the number of pole pairs for Id-Iq-transformation. For more information see Id-Iq-Transformation [6.21.8→143].

Suffix: psi (1, 101 and 201)

Command form
Short form: PSIPOLPAIR␣<NRi>val
SCPI form: :SENSe:PSI:IDIQ:POLepairs␣<NRi>val

| <NRi>val | Config Value

     Possible range: 1 ... 20
     *RST value: 1

Query form
Short form: PSIPOLPAIR?
SCPI form: :SENSe:PSI:IDIQ:POLepairs?

Response
Config Value

Type: <NRi>
Possible range: 1 ... 20

9.10.330 PSITHETANULL
Offset for theta

This command specifies the offset for theta. For more information see Id-Iq-Transformation [6.21.8→143].

Suffix: psi (1, 101 and 201)

Command form
Short form: PSITHETANULL␣<NRf>val
SCPI form: :SENSe:PSI:IDIQ:TNull␣<NRf>val

| <NRf>val | Config Value

     *RST value: 0
     Unit: °

Query form
Short form: PSITHETANULL?
SCPI form: :SENSe:PSI:IDIQ:TNull?

Response
Config Value

Type: <NRf>
Unit: °

9.10.331 PSIMODE
PSI mode for fast inputs

The mode of the PSIs fast analog and frequency inputs.

Suffix: psi (1, 101 and 201)

Command form
Short form: PSIMODE␣<NRi>val
SCPI form: :SENSe:PSI:MOde␣<NRi>val

| <NRi>val | Config Value

> *RST value: 0

> Possible values:

> **0 or Direct:** Direct mode
> **1 or Motor:** Single Motor mode
> **2 or DualMotor:** Dual motor mode

Query form
Short form: PSIMODE?
SCPI form: :SENSe:PSI:MOde?

Response
Config Value

Type: <NRi>

Possible values:

0: Direct mode
1: Single Motor mode
2: Dual motor mode

9.10.332 PSIDUALPULPERREV
Pulses per revolution

Specifies the number of pulses per revolution for the digital speed sensor in dual motor mode.

Suffix: motor (1 ... 2, 101 ... 102 and 201 ... 202)

Command form
Short form: PSIDUALPULPERREV␣<NRi>val
SCPI form: :SENSe:PSI:SPEed:DUAL:PPRevolution␣<NRi>val

| <NRi>val | Specifies the number of pulses per revolution for the digital speed sensor in dual motor mode.

> Possible range: 0 ... 16384
> *RST value: 1

Query form
Short form: PSIDUALPULPERREV?
SCPI form: :SENSe:PSI:SPEed:DUAL:PPRevolution?

Response
Specifies the number of pulses per revolution for the digital speed sensor in dual motor mode.

Type: <NRi>
Possible range: 0 ... 16384

9.10.333 PSIPULPERREV

Pulses per revolution

Specifies the number of pulses per revolution for the digital speed sensor in single motor mode.

Suffix: psi (1, 101 and 201)

Command form
Short form: PSIPULPERREV␣<NRi>val
SCPI form: :SENSe:PSI:SPEed:PPRevolution␣<NRi>val

<NRi>val Specifies the number of pulses per revolution for the digital speed sensor in single motor mode.

Possible range: 0 ... 16384
*RST value: 1

Query form
Short form: PSIPULPERREV?
SCPI form: :SENSe:PSI:SPEed:PPRevolution?

Response
Specifies the number of pulses per revolution for the digital speed sensor in single motor mode.

Type: <NRi>
Possible range: 0 ... 16384

9.10.334 PSISPTYPE

Speed sensor type

The type of the used speed sensor.

Suffix: psi (1, 101 and 201)

Command form
Short form: PSISPTYPE␣<NRi>val
SCPI form: :SENSe:PSI:SPEed:TYPe␣<NRi>val

<NRi>val Config Value

*RST value: 1

Possible values:

0 or Analog: Analog Speed Sensor
1 or Digital: Digital Speed Sensor

Query form
Short form: PSISPTYPE?
SCPI form: :SENSe:PSI:SPEed:TYPe?

Response
Config Value

Type: <NRi>

Possible values:

0: Analog Speed Sensor
1: Digital Speed Sensor

9.10.335 PSISPEEDVOLTAGESPAN
Speed of 10 V

In motor mode, if PSISPTYPE [9.10.334→374] is set to analog, this command specifies the speed for the voltage of 10 V from the speed sensor. For more information see SPEED [6.21.7→143].

Suffix: psi (1, 101 and 201)

Command form
Short form: PSISPEEDVOLTAGESPAN␣<NRf>val
SCPI form: :SENSe:PSI:SPEed:VOLTage:SPAN␣<NRf>val

| <NRf>val | Config Value

Unit: min^{-1}

Query form
Short form: PSISPEEDVOLTAGESPAN?
SCPI form: :SENSe:PSI:SPEed:VOLTage:SPAN?

Response
Config Value

Type: <NRf>
Unit: min^{-1}

9.10.336 PSISPEEDVOLTAGEZERO
Speed of zero voltage

In motor mode, if PSISPTYPE [9.10.334→374] is set to analog, this command specifies the speed for zero voltage from the speed sensor. For more information see SPEED [6.21.7→143].

Suffix: psi (1, 101 and 201)

Command form
Short form: PSISPEEDVOLTAGEZERO␣<NRf>val
SCPI form: :SENSe:PSI:SPEed:VOLTage:ZERo␣<NRf>val

| <NRf>val | Config Value

Unit: min^{-1}

Query form
Short form: PSISPEEDVOLTAGEZERO?
SCPI form: :SENSe:PSI:SPEed:VOLTage:ZERo?

Response
Config Value

Type: <NRf>
Unit: min⁻¹

9.10.337 PSISWCOND

switch condition

The condition of the switching output.

Suffix: do (1 ... 8, 101 ... 108 and 201 ... 208)

Command form
Short form: PSISWCOND␣<NRi>val
SCPI form: :SENSe:PSI:SWITch:CONDition␣<NRi>val

$\boxed{\text{<NRi>val}}$ Config Value

> *RST value: 0

> Possible values:

> **0 or Off:** Off
> **1 or On:** On
> **2 or LE:** Less Than or Equal
> **3 or GT:** Greater Than

Query form
Short form: PSISWCOND?
SCPI form: :SENSe:PSI:SWITch:CONDition?

Response
Config Value

Type: <NRi>

Possible values:

0: Off
1: On
2: Less Than or Equal
3: Greater Than

9.10.338 PSISWINV

switch invert

Inverter for the switching output.

Display ID: PSI$_{\text{SW INV}}$
Suffix: do (1 ... 8, 101 ... 108 and 201 ... 208)

Command form
Short form: PSISWINV␣<Boolean>val
SCPI form: :SENSe:PSI:SWITch:INVert␣<Boolean>val

$\boxed{\text{<Boolean>val}}$ Config Value

> *RST value: 0

Query form
Short form: PSISWINV?
SCPI form: :SENSe:PSI:SWITch:INVert?

Response
Config Value

Type: <Boolean>

9.10.339 PSISWSOURCE
source of switch

The source of the switching output.

Suffix: do (1 ... 8, 101 ... 108 and 201 ... 208)

Command form
Short form: PSISWSOURCE␣<string program data>val
SCPI form: :SENSe:PSI:SWITch:SOURce␣<string program data>val

| <string program data>val | Config Value |

  *RST value: "utrms1111"

Query form
Short form: PSISWSOURCE?
SCPI form: :SENSe:PSI:SWITch:SOURce?

Response
Config Value

Type: <string program data>

9.10.340 PSISWTHRESH
switch threshold

The threshold of the switching output.

Suffix: do (1 ... 8, 101 ... 108 and 201 ... 208)

Command form
Short form: PSISWTHRESH␣<NRf>val
SCPI form: :SENSe:PSI:SWITch:THReshold␣<NRf>val

| <NRf>val | Config Value |

  *RST value: 5

Query form
Short form: PSISWTHRESH?
SCPI form: :SENSe:PSI:SWITch:THReshold?

Response
Config Value

Type: <NRf>

9.10.341 PSITRQNOM
Nominal torque of sensor

In motor mode, if PSITRQTYPE [9.10.343→378] is set to digital, this command specifies the nominal torque M_{nom} of the torque sensor. For more information see TORQUE [6.21.7→142].

Suffix: psi (1, 101 and 201)

Command form
Short form: PSITRQNOM␣<NRf>val
SCPI form: :SENSe:PSI:TORQue:FREQuency:NOMinal␣<NRf>val

| <NRf>val | Config Value

     Unit: Nm

Query form
Short form: PSITRQNOM?
SCPI form: :SENSe:PSI:TORQue:FREQuency:NOMinal?

Response
Config Value

Type: <NRf>
Unit: Nm

9.10.342 PSITRQFRQSPAN
Torque frequency span

In motor mode, if PSITRQTYPE [9.10.343→378] is set to digital, this command specifies the frequency from the torque sensor, for the nominal torque M_{nom} specified by PSITRQNOM [9.10.341→378]. For more information see TORQUE [6.21.7→142].

Suffix: psi (1, 101 and 201)

Command form
Short form: PSITRQFRQSPAN␣<NRf>val
SCPI form: :SENSe:PSI:TORQue:FREQuency:SPAN␣<NRf>val

| <NRf>val | Config Value

     Unit: Hz

Query form
Short form: PSITRQFRQSPAN?
SCPI form: :SENSe:PSI:TORQue:FREQuency:SPAN?

Response
Config Value

Type: <NRf>
Unit: Hz

9.10.343 PSITRQTYPE
Torque sensor type

The type of the used torque sensor.

Suffix: psi (1, 101 and 201)

Command form
Short form: PSITRQTYPE␣<NRi>val
SCPI form: :SENSe:PSI:TORQue:FREQuency:TYPe␣<NRi>val

$\boxed{\text{<NRi>val}}$ Config Value

> *RST value: 1

> Possible values:

> > **0 or Analog:** Analog Torque Sensor
> > **1 or Digital:** Digital Torque Sensor

Query form
Short form: PSITRQTYPE?
SCPI form: :SENSe:PSI:TORQue:FREQuency:TYPe?

Response
Config Value

Type: <NRi>

Possible values:

0: Analog Torque Sensor
1: Digital Torque Sensor

9.10.344 PSITRQFRQZERO
Torque zero frequency

In motor mode, if PSITRQTYPE [9.10.343→378] is set to digital, this command specifies the frequency from the torque sensor for zero torque. For more information see TORQUE [6.21.7→142].

Suffix: psi (1, 101 and 201)

Command form
Short form: PSITRQFRQZERO␣<NRf>val
SCPI form: :SENSe:PSI:TORQue:FREQuency:ZERo␣<NRf>val

$\boxed{\text{<NRf>val}}$ Config Value

> Unit: Hz

Query form
Short form: PSITRQFRQZERO?
SCPI form: :SENSe:PSI:TORQue:FREQuency:ZERo?

Response
Config Value

Type: <NRf>
Unit: Hz

9.10.345 PSIDUALTRQVOLTAGESPAN
Torque of the 10 V

In dual motor mode, this command specifies the torque value for a voltage of 10 V of the analog torque sensor. For more information see TORQUE [6.21.7→142].

Suffix: motor (1 ... 2, 101 ... 102 and 201 ... 202)

Command form
Short form: PSIDUALTRQVOLTAGESPAN␣<NRf>val
SCPI form: :SENSe:PSI:TORQue:VOLTage:DUAL:SPAN␣<NRf>val

$\boxed{\text{<NRf>val}}$ Config Value

  Unit: Nm

Query form
Short form: PSIDUALTRQVOLTAGESPAN?
SCPI form: :SENSe:PSI:TORQue:VOLTage:DUAL:SPAN?

Response
Config Value

Type: <NRf>
Unit: Nm

9.10.346 PSIDUALTRQVOLTAGEZERO
Torque of zero voltage

In dual motor mode, this command specifies the torque value for zero voltage output of the analog torque sensor. For more information see TORQUE [6.21.7→142].

Suffix: motor (1 ... 2, 101 ... 102 and 201 ... 202)

Command form
Short form: PSIDUALTRQVOLTAGEZERO␣<NRf>val
SCPI form: :SENSe:PSI:TORQue:VOLTage:DUAL:ZERO␣<NRf>val

$\boxed{\text{<NRf>val}}$ Config Value

  Unit: Nm

Query form
Short form: PSIDUALTRQVOLTAGEZERO?
SCPI form: :SENSe:PSI:TORQue:VOLTage:DUAL:ZERO?

Response
Config Value

Type: <NRf>
Unit: Nm

9.10.347 PSITRQVOLTAGESPAN
Torque of the 10 V

In motor mode, if PSITRQTYPE [9.10.343→378] is set to analog, this command specifies the torque value, for the voltage of 10 V from the torque sensor. For more information see TORQUE [6.21.7→142].

Suffix: psi (1, 101 and 201)

Command form
Short form: PSITRQVOLTAGESPAN␣<NRf>val
SCPI form: :SENSe:PSI:TORQue:VOLTage:SPAN␣<NRf>val

$\boxed{\text{<NRf>val}}$ Config Value

  Unit: Nm

Query form
Short form: PSITRQVOLTAGESPAN?
SCPI form: :SENSe:PSI:TORQue:VOLTage:SPAN?

Response
Config Value

Type: <NRf>
Unit: Nm

9.10.348 PSITRQVOLTAGEZERO
Torque of zero voltage

In motor mode, if PSITRQTYPE [9.10.343→378] is set to analog, this command specifies the torque value, for zero voltage from the torque sensor. For more information see Torque [6.21.7→142].

Suffix: psi (1, 101 and 201)

Command form
Short form: PSITRQVOLTAGEZERO␣<NRf>val
SCPI form: :SENSe:PSI:TORQue:VOLTage:ZERo␣<NRf>val

| <NRf>val | Config Value

  Unit: Nm

Query form
Short form: PSITRQVOLTAGEZERO?
SCPI form: :SENSe:PSI:TORQue:VOLTage:ZERo?

Response
Config Value

Type: <NRf>
Unit: Nm

9.10.349 SCPTRT
Pretrigger length in time

This command sets the time interval to record samples before the trigger event.

Display ID: Scope Pretrigger Time
Suffix: sp (1, 2, 21 and 22)

Command form
Short form: SCPTRT␣<Time>val
SCPI form: :SENSe:SCOPe:PTRigger[:TIMe]␣<Time>val

| <Time>val | Config Value

  *RST value: 0.0
  Unit: s

Query form
Short form: SCPTRT?
SCPI form: :SENSe:SCOPe:PTRigger[:TIMe]?

Response
Config Value

Type: <Time>
Unit: s

9.10.350 SAMPLESTORAGEMODE
Sample storage mode

Determines how the sample storage is used. The sample storage is available in all CYCLMOD [9.10.355→385] settings *except* SCOPE.

Command form
Short form: SAMPLESTORAGEMODE␣<NRi>val
SCPI form: :SENSe:SCOPe:SAMPlestoragemode␣<NRi>val

$\boxed{\text{<NRi>val}}$ Config Value

*RST value: 0

Possible values:

0 or NONe: None
1 or EVENt: Storage for event-triggered recordings; requires option L6-OPT-EVT (see OPTN [9.10.455→428])

Query form
Short form: SAMPLESTORAGEMODE?
SCPI form: :SENSe:SCOPe:SAMPlestoragemode?

Response
Config Value

Type: <NRi>

Possible values:

0: None
1: Storage for event-triggered recordings; requires option L6-OPT-EVT (see OPTN [9.10.455→428])

9.10.351 SCSR
Scope sampling rate

This command sets the sample rate of a particular scope. If the desired sample rate is not available, the next higher one is applied, if possible.

Suffix: sp (1, 2, 21 and 22)

Command form
Short form: SCSR␣<NRf>val
SCPI form: :SENSe:SCOPe:SRATe␣<NRf>val

$\boxed{\text{<NRf>val}}$ Config Value

*RST value: 8000
Unit: Hz

Query form
Short form: SCSR?
SCPI form: :SENSe:SCOPe:SRATe?

Response
Config Value

Type: <NRf>
Unit: Hz

9.10.352 SCTRAC
Tracks' signal sources

Each element of the array specifies the signal for the scope track. For a description of possible signals see SAMPLE SOURCES [6.14.4→124].

Suffix: sp (1, 2, 21 and 22)

Command form
Short form: SCTRAC␣<numeric_list>listVal, <string program data>val
SCPI form: :SENSe:SCOPe:TRACk␣<numeric_list>listVal, <string program data>val

| <numeric_list>listVal | Specifies the positions of the elements to modify.

      List elements: 0 ... 15
      *RST value: Positions 0 ... 15.

| <string program data>val | Scope track source

      *RST value: ""

Query form
Short form: SCTRAC?␣<numeric_list>listVal
SCPI form: :SENSe:SCOPe:TRACk?␣<numeric_list>listVal

| <numeric_list>listVal | Specifies the positions of the elements to query.

      List elements: 0 ... 15

Response
Config Value

Type: <string program data>,...
List elements: 0 ... 15

9.10.353 WIRE
Physical wiring

Specifies the physical wiring of a group's measuring channels. See GROUPS [6.1→101] for details.

Display ID: Wiring
Suffix: g (1 ... 7)

Command form

Short form: WIRE␣<NRi>val
SCPI form: :SENSe:SCOPe:WIRing␣<NRi>val

$\boxed{<NRi>val}$ Config Value

    *RST value: 0

    Possible values:

      0 or DIRect: Direct wiring
      1 or ARON: Aron wiring
      2 or STARstar: U and I are both wired as star
      3 or DELTastar: U is wired as delta and I is wired as star

Query form

Short form: WIRE?
SCPI form: :SENSe:SCOPe:WIRing?

Response

Config Value

Type: <NRi>

Possible values:

0: Direct wiring
1: Aron wiring
2: U and I are both wired as star
3: U is wired as delta and I is wired as star

9.10.354 SCRIPT
The script code

Specifies the script to run.

Command form

Short form: SCRIPT␣<string program data>val
SCPI form: :SENSe:SCRIPT:LISTing␣<string program data>val

$\boxed{<\text{string program data}>val}$ This variable holds the currently running script. Changing this variable also changes the running script. Usually, this script is created in the tab "Script Editor" in the "Custom" menu.

    *RST value: ""

Query form

Short form: SCRIPT?
SCPI form: :SENSe:SCRIPT:LISTing?

Response

This variable holds the currently running script. Changing this variable also changes the running script. Usually, this script is created in the tab "Script Editor" in the "Custom" menu.

Type: <string program data>

9.10.355 CYCLMOD
Device cycle mode

Specifies when values are computed. This setting also affects the configuration of the
SYNCHRONIZATION CONNECTOR [4.3→46] cycle I/O pin. For details about the HARMn modes see
PARALLEL CALCULATIONS [6.7→113].

Display ID: Cyclemode

Command form
Short form: CYCLMOD␣<NRi>val
SCPI form: :SENSe:SWEep:MODe␣<NRi>val

| <NRi>val | Config Value

*RST value: 0

Possible values:

0 or CYCLetime: Fixed interval, specified by CYCL [9.10.356→386]
1 or EXTern: External signal
2 or HARM1: Harmonics of group 1
3 or HARM2: Harmonics of group 2
4 or HARM3: Harmonics of group 3
5 or HARM4: Harmonics of group 4
6 or HARM5: Harmonics of group 5
7 or HARM6: Harmonics of group 6
8 or HARM7: Harmonics of group 7
9 or SCOPE: Completion of scope. Note that this disables the sample storage (see
SAMPLESTORAGEMODE [9.10.350→382]).
10 or FLICKER: Flicker and halfwave-based values

Query form
Short form: CYCLMOD?
SCPI form: :SENSe:SWEep:MODe?

Response
Config Value

Type: <NRi>

Possible values:

0: Fixed interval, specified by CYCL [9.10.356→386]
1: External signal
2: Harmonics of group 1
3: Harmonics of group 2
4: Harmonics of group 3
5: Harmonics of group 4
6: Harmonics of group 5
7: Harmonics of group 6
8: Harmonics of group 7
9: Completion of scope. Note that this disables the sample storage (see
SAMPLESTORAGEMODE [9.10.350→382]).
10: Flicker and halfwave-based values

9.10.356 CYCL

—— **Cycle time**

Specifies the device cycle time. This setting only takes effect if the corresponding cycle mode is selected (see CYCLMOD [9.10.355→385]).

Display ID: Cycle

Command form
Short form: CYCL␣<Time>val
SCPI form: :SENSe:SWEep:TIME␣<Time>val

| <Time>val | Config Value

Possible range: 0.01 ... 60
*RST value: 0.5
Unit: s

Query form
Short form: CYCL?
SCPI form: :SENSe:SWEep:TIME?

Response
Config Value

Type: <Time>
Possible range: 0.01 ... 60
Unit: s

9.10.357 TRCOND

—— **Transient trigger condition**

Specifies the transient trigger condition for comparison with the limits TRLIMITA [9.10.361→388] and TRLIMITB [9.10.362→389]. An external trigger condition can be enabled via TREXTERN [9.10.360→388].

Display ID: Transient trigger condition
Suffix: tr (1 ... 3)

Command form
Short form: TRCOND␣<NRi>val
SCPI form: :SENSe:TRANsient:COND␣<NRi>val

| <NRi>val | Config Value

*RST value: 0
Unit: V

Possible values:

0 or DISabled:	Disabled
1 or GTA:	Signal must be greater than TRLIMITA [9.10.361→388]
2 or LEA:	Signal must be less than or equal to TRLIMITA [9.10.361→388]
3 or GTB:	Signal must be greater than TRLIMITA [9.10.361→388]
4 or LEB:	Signal must be less than or equal to TRLIMITA [9.10.361→388]
5 or INBetween:	Signal must be in between TRLIMITA [9.10.361→388] and TRLIMITB [9.10.362→389]
6 or OUTSide:	Signal must be outside of (not in between) TRLIMITA [9.10.361→388] and TRLIMITB [9.10.362→389]

Query form
Short form: TRCOND?
SCPI form: :SENSe:TRANsient:COND?

Response

Config Value

Type: <NRi>
Unit: V

Possible values:

0: Disabled
1: Signal must be greater than ᴛʀʟɪᴍɪᴛᴀ [9.10.361→388]
2: Signal must be less than or equal to ᴛʀʟɪᴍɪᴛᴀ [9.10.361→388]
3: Signal must be greater than ᴛʀʟɪᴍɪᴛᴀ [9.10.361→388]
4: Signal must be less than or equal to ᴛʀʟɪᴍɪᴛᴀ [9.10.361→388]
5: Signal must be in between ᴛʀʟɪᴍɪᴛᴀ [9.10.361→388] and ᴛʀʟɪᴍɪᴛʙ [9.10.362→389]
6: Signal must be outside of (not in between) ᴛʀʟɪᴍɪᴛᴀ [9.10.361→388] and ᴛʀʟɪᴍɪᴛʙ [9.10.362→389]

9.10.358 DURTR
Duration of the Transient

This command reads the duration of the recorded samples by the transient.

Display ID: Δt_{tr}

Query form

Short form: DURTR?
SCPI form: :SENSe:TRANsient:DURation?

Response

The measured value

Type: <Time>
Unit: s

9.10.359 TREVENTDUR
Transient trigger minimum duration

Specifies the required minimum duration the trigger condition(s) must hold true for the transient trigger to assert.

Display ID: Upper limit

Command form

Short form: TREVENTDUR␣<Time>val
SCPI form: :SENSe:TRANsient:EVENtduration␣<Time>val

<Time>val	Config Value

> Possible range: 0 ... 31536000
> *RST value: 0.0
> Unit: s

Query form

Short form: TREVENTDUR?
SCPI form: :SENSe:TRANsient:EVENtduration?

Response
Config Value

Type: <Time>
Possible range: 0 ... 31536000
Unit: s

9.10.360 TREXTERN
External transient trigger

This setting enables an external signal to assert the transient trigger. The signal has to be applied to the Transient I/O pin of the SYNCHRONIZATION CONNECTOR [4.3→46]. If TREXTERN is enabled, the Transient I/O pin is automatically configured as an input, otherwise it is configured as an output which is asserted along with the internal transient trigger. See the pin description for further details regarding the signal parameters. The conditions for the internal trigger can be configured via TRCOND [9.10.357→386].

Display ID: External Transient Trigger

Command form
Short form: TREXTERN␣<Boolean>val
SCPI form: :SENSe:TRANsient:EXTernal␣<Boolean>val

| <Boolean>val | Config Value |

　　　*RST value: 0

Query form
Short form: TREXTERN?
SCPI form: :SENSe:TRANsient:EXTernal?

Response
Config Value

Type: <Boolean>

9.10.361 TRLIMITA
Transient trigger limit A

Specifies the transient trigger upper level. Use TRCOND to specify the condition for comparisons with the limits (A and TRLIMITB [9.10.362→389]).

Display ID: Limit A
Suffix: tr (1 ... 3)

Command form
Short form: TRLIMITA␣<NRf>val
SCPI form: :SENSe:TRANsient:LIMit:A␣<NRf>val

| <NRf>val | Config Value |

　　　*RST value: 0.0
　　　Unit: V

Query form
Short form: TRLIMITA?
SCPI form: :SENSe:TRANsient:LIMit:A?

Response
Config Value

Type: <NRf>
Unit: V

9.10.362 TRLIMITB
Transient trigger level B

Specifies the transient trigger limit B. Use TRCOND to specify the condition for comparisons with the limits (B and TRLIMITA [9.10.361→388]).

Display ID: Limit B
Suffix: tr (1 ... 3)

Command form
Short form: TRLIMITB␣<NRf>val
SCPI form: :SENSe:TRANsient:LIMit:B␣<NRf>val

| <NRf>val | Config Value

>   *RST value: 0.0
>   Unit: V

Query form
Short form: TRLIMITB?
SCPI form: :SENSe:TRANsient:LIMit:B?

Response
Config Value

Type: <NRf>
Unit: V

9.10.363 TRCHLINK
Transient trigger channel linkage

Specifies how the trigger events from different channels are linked.

Display ID: channel linkage

Command form
Short form: TRCHLINK␣<NRi>val
SCPI form: :SENSe:TRANsient:LOGic␣<NRi>val

| <NRi>val | Config Value

>   *RST value: 0

>   Possible values:

>   **0 or OR:**   OR linkage: *At least one channel* must trigger for the transient trigger to assert.
>   **1 or AND:** AND linkage: *All channels* must trigger for the transient trigger to assert.

Query form
Short form: TRCHLINK?
SCPI form: :SENSe:TRANsient:LOGic?

Response
Config Value

Type: <NRi>

Possible values:

0: OR linkage: *At least one channel* must trigger for the transient trigger to assert.
1: AND linkage: *All channels* must trigger for the transient trigger to assert.

9.10.364 TRCPTRT

Pretrigger Time of the Transient Scope

This command sets the time interval between the begin of the sample recording and the transient trigger event. Note, that this command does not change the total recording time of the transient scope.

Display ID: Transient Pretrigger Time

Command form
Short form: TRCPTRT␣<Time>val
SCPI form: :SENSe:TRANsient[:PRETrigger]␣<Time>val

| <Time>val | Config Value

> *RST value: 0.0
> Unit: s

Query form
Short form: TRCPTRT?
SCPI form: :SENSe:TRANsient[:PRETrigger]?

Response
Config Value

Type: <Time>
Unit: s

9.10.365 TRRECLEN

recording time

This command specifies the recording time of the transient scope. The maximal recoding time is the maximal number of samples per track devided by the sample rate TRCSR [9.10.367→391]/TRPSR [9.10.184→307].

Display ID: Upper limit

Command form
Short form: TRRECLEN␣<Time>val
SCPI form: :SENSe:TRANsient:RECordlength␣<Time>val

| <Time>val | Config Value

> Possible range: 0 ... 158400
> *RST value: 1.0
> Unit: s

Query form
Short form: TRRECLEN?
SCPI form: :SENSe:TRANsient:RECordlength?

Response

Config Value

Type: <Time>
Possible range: 0 ... 158400
Unit: s

9.10.366 TRSOURCE
Transient trigger signal source

The signal source for each trigger

Suffix: tr (1 ... 3)

Command form
Short form: TRSOURCE␣<string program data>val
SCPI form: :SENSe:TRANsient:SOURce␣<string program data>val

| <string program data>val | Config Value |

    *RST value: ""

Query form
Short form: TRSOURCE?
SCPI form: :SENSe:TRANsient:SOURce?

Response
Config Value

Type: <string program data>

9.10.367 TRCSR
Transient sampling rate

This command sets the sample rate of the transient scope. If the desired sample rate is not available, the next higher one is applied, if possible.

Command form
Short form: TRCSR␣<NRf>val
SCPI form: :SENSe:TRANsient:SRATe␣<NRf>val

| <NRf>val | Config Value |

    *RST value: 8000
    Unit: Hz

Query form
Short form: TRCSR?
SCPI form: :SENSe:TRANsient:SRATe?

Response
Config Value

Type: <NRf>
Unit: Hz

9.10.368 TSTR
Time of the First Sample

This command reads the point in time of the first sample in the transient.

Display ID: ts_{tr}

Query form
Short form: TSTR?
SCPI form: :SENSe:TRANsient:TIMestamp?

Response
The measured value

Type: <Date>

9.10.369 TRCTRAC
Tracks' signal sources

Each element of the array specifies the signal for the track. For a description of possible signals see SAMPLE SOURCES [6.14.4→124].

Command form
Short form: TRCTRAC␣<numeric_list>listVal, <string program data>val
SCPI form: :SENSe:TRANsient:TRACk␣<numeric_list>listVal, <string program data>val

| <numeric_list>listVal | Specifies the positions of the elements to modify.

> List elements: 0 ... 15
> *RST value: Positions 0 ... 15.

| <string program data>val | Transient track source

> *RST value: ""

Query form
Short form: TRCTRAC?␣<numeric_list>listVal
SCPI form: :SENSe:TRANsient:TRACk?␣<numeric_list>listVal

| <numeric_list>listVal | Specifies the positions of the elements to query.

> List elements: 0 ... 15

Response
Config Value

Type: <string program data>,...
List elements: 0 ... 15

9.10.370 TRTRIGMODE
Transient Triggering Mode

Sets whether the event trigger triggers on the occurence of a new event or also on an already present event when event searching starts.

Display ID: Triggering Mode

Command form
Short form: TRTRIGMODE␣<NRi>val
SCPI form: :SENSe:TRANsient:TRIGgermode␣<NRi>val

$\boxed{\text{<NRi>val}}$ Config Value

*RST value: 0

Possible values:

0 or Edge: Triggers on new occurence of event.
1 or Level: Triggers on already present event.

Query form
Short form: TRTRIGMODE?
SCPI form: :SENSe:TRANsient:TRIGgermode?

Response
Config Value

Type: <NRi>

Possible values:

0: Triggers on new occurence of event.
1: Triggers on already present event.

9.10.371 UDLY

U channel delay correction

Configures a (positive or negative!) *correction* time to compensate for external delays in the signal path, e.g. due to a sensor. Note that delay compensation for compatible ZES sensors is done automatically by the device (using data stored in the sensor's nonvolatile memory). Gain correction is configured via USCA [9.10.387→400].

Display ID: U Delay
Suffix: p (1 ... 7)

Command form
Short form: UDLY␣<Time>val
SCPI form: :SENSe:VOLTage:DELay␣<Time>val

$\boxed{\text{<Time>val}}$ Config Value

Possible range: -5.5e-6 ... 5.5e-6
*RST value: 0
Unit: s

Query form
Short form: UDLY?
SCPI form: :SENSe:VOLTage:DELay?

Response
Config Value

Type: <Time>
Possible range: -5.5e-6 ... 5.5e-6
Unit: s

9.10.372 IDNU

Human-readable U sensor identification

If no sensor is attached to the channel the string is empty. Otherwise its value consists of 4 comma-separated fields identifying the sensor:

- Manufacturer
- Model
- Serial Number
- Firmware Version

Suffix: p (1 ... 7)

Query form
Short form: IDNU?
SCPI form: :SENSe:VOLTage:IDENtify?

Response
Config Value

Type: <string program data>

9.10.373 UJACK

U channel jack

Selects a U channel jack. UJLS [9.10.374→395] provides a list of all currently available jacks or more precisely, their numeric IDs.

Display ID: U Jack
Suffix: p (1 ... 7)

Command form
Short form: UJACK␣<NRi>val
SCPI form: :SENSe:VOLTage:JACK␣<NRi>val

| <NRi>val | Config Value

    *RST value: 0

    Possible values:

    0 or USTar: U* jack
    1 or USENsor: U_{sensor} jack

Query form
Short form: UJACK?
SCPI form: :SENSe:VOLTage:JACK?

Response
Config Value

Type: <NRi>

Possible values:

0: U* jack
1: U_{sensor} jack

9.10.374 UJLS

―――― **U channel jack list**

Specifies the currently available U channel jacks or more preciselym their numeric IDs. See the documentation for UJACK [9.10.373→394] to decode the values; use UJACK with one of the values to select the appropriate jack.

Suffix: p (1 ... 7)

Query form
Short form: UJLS?
SCPI form: :SENSe:VOLTage:LJACk?

Response
Config Value

Type: <NRi>,...
List elements: 0 ... (dynamic size)

9.10.375 UAUTO

―――― **U channel auto range**

In manual mode the configured range, if valid, is taken as given and stays fixed until changed by a configuration command, for instance explicitly by URNG [9.10.386→399] or implicitly by UJACK [9.10.373→394]. With auto range enabled, the instrument monitors the range usage and if necessary automatically selects an appropriate smaller or larger range in order to best measure the signal applied to the selected jack, with the same set of ranges available as in manual mode (see URNLS [9.10.377→396]).

Display ID: U Auto Range
Suffix: p (1 ... 7)

Command form
Short form: UAUTO␣<Boolean>val
SCPI form: :SENSe:VOLTage:RANGe:AUTo␣<Boolean>val

| <Boolean>val | Config Value |

      *RST value: 1

Query form
Short form: UAUTO?
SCPI form: :SENSe:VOLTage:RANGe:AUTo?

Response
Config Value

Type: <Boolean>

9.10.376 URNMAXLS

―――― **List of nominal maximum RMS values of U channel ranges**

Specifies the maximum RMS values for the nominal range values listed in URNLS [9.10.377→396].

Suffix: p (1 ... 7)

Query form
Short form: URNMAXLS?
SCPI form: :SENSe:VOLTage:RANGe:LNMax?

Response
Config Value

Type: <NRf>,...
List elements: 0 ... (dynamic size)

9.10.377 URNLS
List of nominal U channel ranges

Specifies the currently available U channel measuring ranges or more precisely, their nominal values. In general, this list changes when a different jack is selected (via UJACK [9.10.373→394]) or a sensor is connected to the channel, etc. For S channels the values listed also depend on URNGMOD [9.10.384→398]. Use URNG [9.10.386→399] with one of the values to select the appropriate range. The corresponding maximum RMS and peak values are listed in URNMAXLS [9.10.376→395] and URNPKLS [9.10.378→396], respectively.

Suffix: p (1 ... 7)

Query form
Short form: URNLS?
SCPI form: :SENSe:VOLTage:RANGe:LNOMinal?

Response
Config Value

Type: <NRf>,...
List elements: 0 ... (dynamic size)

9.10.378 URNPKLS
List of nominal peak values of U channel ranges

Specifies the peak values corresponding to the nominal range values listed in URNLS [9.10.377→396].

Suffix: p (1 ... 7)

Query form
Short form: URNPKLS?
SCPI form: :SENSe:VOLTage:RANGe:LNPeak?

Response
Config Value

Type: <NRf>,...
List elements: 0 ... (dynamic size)

9.10.379 URNULS
List of units of nominal U channel ranges

Specifies the unit of each currently available U channel nominal measuring range (as given by URNLS [9.10.377→396]):

- "V" (Volt)

In general, this list changes whenever URNLS changes.

Suffix: p (1 ... 7)

Query form

Short form: URNULS?
SCPI form: :SENSe:VOLTage:RANGe:LNUNit?

Response

Config Value

Type: <string program data>,...
List elements: 0 ... (dynamic size)

9.10.380 URSLS
List of scaled U channel ranges

Specifies the currently available U channel measuring ranges including custom (USCA [9.10.387→400]) and possibly sensor scaling. In general, this list changes when USCA changes, a different jack is selected (UJACK [9.10.373→394]) or a sensor is connected to the channel, etc. For S channels the listed values also depende on URNGMOD [9.10.384→398]. Be aware that when using one of these *scaled* values to select a range via URNG [9.10.386→399], a match with a *nominal* range value takes precedence. To unambiguously select a range, use its *nominal* value (see URNLS [9.10.377→396]). The corresponding scaled maximum RMS and peak values are listed in URSMAXLS [9.10.381→397] and URSPKLS [9.10.382→398], respectively.

Suffix: p (1 ... 7)

Query form

Short form: URSLS?
SCPI form: :SENSe:VOLTage:RANGe:LSCaled?

Response

Config Value

Type: <NRf>,...
List elements: 0 ... (dynamic size)

9.10.381 URSMAXLS
List of scaled maximum RMS values of U channel ranges

Specifies the maximum RMS values corresponding to the scaled range values listed in URSLS [9.10.380→397].

Suffix: p (1 ... 7)

Query form

Short form: URSMAXLS?
SCPI form: :SENSe:VOLTage:RANGe:LSMax?

Response

Config Value

Type: <NRf>,...
List elements: 0 ... (dynamic size)

9.10.382 URSPKLS

List of scaled peak values of U channel ranges

Specifies the peak values corresponding to the scaled range values listed in URSLS [9.10.380→397].

Suffix: p (1 ... 7)

Query form

Short form: URSPKLS?
SCPI form: :SENSe:VOLTage:RANGe:LSPeak?

Response

Config Value

Type: <NRf>,...
List elements: 0 ... (dynamic size)

9.10.383 URSULS

List of units of scaled U channel ranges

Specifies the unit of each currently available U channel scaled measuring range (as given by URSLS [9.10.380→397]):

- "V" (Volt)

In general, this list changes whenever URSLS changes.

Suffix: p (1 ... 7)

Query form

Short form: URSULS?
SCPI form: :SENSe:VOLTage:RANGe:LSUNit?

Response

Config Value

Type: <string program data>,...
List elements: 0 ... (dynamic size)

9.10.384 URNGMOD

U range mode

This setting only applies to groups of S channels. Selects U range mode selects whether the display of U range *nominal* values is geared to AC or DC measurements,

- URNG [9.10.386→399],

- URSCALED [9.10.385→399]

- URNLS [9.10.377→396], and

- URSLS [9.10.380→397].

Consequently it also affects which values may be used to select a range via URNG, that is the values listed in URNLS and URSLS.

Display ID: U range mode
Suffix: g (1 ... 7)

Command form
Short form: URNGMOD␣<NRi>val
SCPI form: :SENSe:VOLTage:RANGe:MODe␣<NRi>val

| <NRi>val | Config Value

*RST value: 0

Possible values:

0 or AC: AC
1 or DC: DC

Query form
Short form: URNGMOD?
SCPI form: :SENSe:VOLTage:RANGe:MODe?

Response
Config Value

Type: <NRi>

Possible values:

0: AC
1: DC

9.10.385 URSCALED
Scaled U channel measuring range

Gives the U channel measuring range including any scaling due to e.g. ᴜsᴄᴀ [9.10.387→400] and/or a plugged-in ZES sensor. Use ᴜʀɴɢ [9.10.386→399] to query the nominal range value without scaling or to set the active range.

Display ID: Scaled U Range
Suffix: p (1 ... 7)

Query form
Short form: URSCALED?
SCPI form: :SENSe:VOLTage:RANGe:RSCaled?

Response
Config Value

Type: <NRf>
Unit: V

9.10.386 URNG
U channel measuring range

Specifies the selected U channel measuring range. A query always returns the *nominal* range value. Note that for S channels these values depend on ᴜʀɴɢᴍᴏᴅ [9.10.384→398]. The scaled range (possibly including custom ᴜsᴄᴀ [9.10.387→400] and ZES sensor scaling, if any) is given by ᴜʀsᴄᴀʟᴇᴅ [9.10.385→399]. When setting the range, either the nominal *or* the scaled value may be used. If the given (nominal or scaled) value matches the nominal value of an available range, that one is selected; otherwise, if the given value matches the scaled value of an available range, that one is selected. A matching nominal value thus takes precedence over a matching scaled value. ᴜʀɴʟs [9.10.377→396] provides a list of currently available U channel ranges or more precisely, their nominal values; ᴜʀsʟs [9.10.380→397] provides the corresponding scaled values. The user manual includes detailed technical data for each range.

Display ID: U Range
Suffix: p (1 ... 7)

Command form
Short form: URNG␣<NRf>val
SCPI form: :SENSe:VOLTage:RANGe[:UPPer]␣<NRf>val

| <NRf>val | Config Value

    Unit: V

Query form
Short form: URNG?
SCPI form: :SENSe:VOLTage:RANGe[:UPPer]?

Response
Config Value

Type: <NRf>
Unit: V

9.10.387 USCA
U channel scaling

Specifies a custom U channel scaling factor. A scaling factor of zero is not permitted. Delay correction is configured via ᴜᴅʟʏ [9.10.371→393].

Display ID: U Scale
Suffix: p (1 ... 7)

Command form
Short form: USCA␣<NRf>val
SCPI form: :SENSe:VOLTage:SCALe␣<NRf>val

| <NRf>val | Config Value

    Possible range: -1e6 ... 1e6
    *RST value: 1

Query form
Short form: USCA?
SCPI form: :SENSe:VOLTage:SCALe?

Response
Config Value

Type: <NRf>
Possible range: -1e6 ... 1e6

9.10.388 ZSUP
Zero suppression

Controls the zero suppression for all P channels, except for harmonics values. If the RMS value of the signal is smaller than 1.5% of the max. peak value of the range, and the absolute values for MAX and MIN are smaller than 5%, then all values are set to zero.

Display ID: Zero Suppression

Command form
Short form: ZSUP␣<Boolean>val
SCPI form: :SENSe:ZPReject␣<Boolean>val

| <Boolean>val | Config Value |

*RST value: 1

Query form
Short form: ZSUP?
SCPI form: :SENSe:ZPReject?

Response
Config Value

Type: <Boolean>

9.10.389 IST
Individual Status Query

This returns the status of the 'ist' local message in the device. This command is identical to
*IST [9.10.6→236]

Query form
Short form: IST?
SCPI form: :STATus:INDividual?

Response
The IST local message

Type: <Boolean>

9.10.390 SOC
Operation Status Condition Register

Returns the status operation condition register.

Query form
Short form: SOC?
SCPI form: :STATus:OPERation:CONDition?

Response
The register's value

Type: <NRi>

Possible values:

bit 0: Calibrating
bit 1: Setting
bit 2: Ranging
bit 3: Sweeping
bit 4: Measuring
bit 5: Waiting for Trigger Summary
bit 6: Waiting for ARM Summary
bit 7: Correcting
bit 13: Instrument Summary
bit 14: Program Running

9.10.391 SOPT
Operation Status Positive Transition Register

Sets or reads the operation status positive transition register. If a bit in the SOC register changes from 0 to 1 and the corresponding bit in the SOPT register is 1, then the corresponding bit in the SOE register is set to 1.

Command form
Short form: SOPT␣<NRi>registerValue
SCPI form: :STATus:OPERation:PTRansition␣<NRi>registerValue

| <NRi>registerValue | The register's value

Query form
Short form: SOPT?
SCPI form: :STATus:OPERation:PTRansition?

Response
The register's value

Type: <NRi>

9.10.392 SPPE
parallel Poll Register Enable

Set or read the parallel poll enable register. This command is identical to *PRE [9.10.8→236]

Command form
Short form: SPPE␣<NRi>registerValue
SCPI form: :STATus:PPOLl[:ENABle]␣<NRi>registerValue

| <NRi>registerValue | The register's value

Query form
Short form: SPPE?
SCPI form: :STATus:PPOLl[:ENABle]?

Response
The register's value

Type: <NRi>

9.10.393 PRES
Reset transition registers

Resets all enable and transition registers to default values.

Command form
Short form: PRES
SCPI form: :STATus:PRESet

9.10.394 QUESTMASK

Mask for questionable values

Enables/Disables that measurement values considered questionable (see QUEST [9.10.204→314]) will be delivered to the interfaces. This affects the commands for fetching new measurement values (e.g. INIM [9.10.221→321]) and continuous output (CONT [9.10.219→320]).

Command form
Short form: QUESTMASK␣<NRi>val
SCPI form: :STATus:QMASk␣<NRi>val

$\boxed{<\text{NRi}>\text{val}}$ Set/Unset the mask. Zero means: Questionable data will be discarded. One means: Questionable data will be put through

   *RST value: 0

Query form
Short form: QUESTMASK?
SCPI form: :STATus:QMASk?

Response
Set/Unset the mask. Zero means: Questionable data will be discarded. One means: Questionable data will be put through

Type: <NRi>

9.10.395 SQC

Questionable Status Condition Register

Reads the Questionable Status Condition Register. If QUESTMASK [9.10.394→403] is set, this register indicates questionable data by setting all bits to 1. Otherwise, all bits are cleared.

Query form
Short form: SQC?
SCPI form: :STATus:QUEStionable:CONDition?

Response
The register's value

Type: <NRi>

Possible values:

bit 0: Voltage
bit 1: Current
bit 2: Time
bit 3: Power
bit 4: Temperature
bit 5: Frequency
bit 6: Phase
bit 7: Modulation
bit 8: Calibration
bit 13: Instrument Summary
bit 14: Command Warning

9.10.396 SQNT

Questionable status negitive transition register

Sets or reads the questionable status negative transition register. If a bit in the SQC register changes from 1 to 0 and the corresponding bit in the SQNT register is 1, then the corresponding bit in the SQE register is set to 1.

Command form
Short form: SQNT␣<NRi>registerValue
SCPI form: :STATus:QUEStionable:NTRansition␣<NRi>registerValue

| <NRi>registerValue | The register's value

Query form
Short form: SQNT?
SCPI form: :STATus:QUEStionable:NTRansition?

Response

The register's value

Type: <NRi>

9.10.397 SQPT

Questionable status positive transition register

Sets or reads the questionable status positive transition register. If a bit in the SQC register changes from 0 to 1 and the corresponding bit in the SQPT register is 1, then the corresponding bit in the SQE register is set to 1.

Command form
Short form: SQPT␣<NRi>registerValue
SCPI form: :STATus:QUEStionable:PTRansition␣<NRi>registerValue

| <NRi>registerValue | The register's value

Query form
Short form: SQPT?
SCPI form: :STATus:QUEStionable:PTRansition?

Response

The register's value

Type: <NRi>

9.10.398 STB

Status byte register

Sets   or   queries   the   Status   byte   register.   This   command   is   identical   to
*STB [9.10.11→238]

Query form
Short form: STB?
SCPI form: :STATus[:REGister]?

Response

The register's value

Type: <NRi>

Possible values:

bit 1: Extended Status Byte
bit 2: Error Queue not Empty
bit 3: Questionable Status Register
bit 4: Output Queue not Empty
bit 5: Event Status Register
bit 6: Status Byte Enable
bit 7: Operation Status Register

9.10.399 SRE

Service request enable register

Sets or queries the service request enable register. This command is identical to *SRE [9.10.10→237]

Command form

Short form: SRE␣<NRi>registerValue
SCPI form: :STATus:SERVice[:ENABle]␣<NRi>registerValue

| <NRi>registerValue | The register's value |

Query form

Short form: SRE?
SCPI form: :STATus:SERVice[:ENABle]?

Response

The register's value

Type: <NRi>

9.10.400 SREH

Service request enable register high byte

Sets or reads the high byte of the SRE register.

Command form

Short form: SREH␣<NRi>registerValue
SCPI form: :STATus:SERVice:HENable␣<NRi>registerValue

| <NRi>registerValue | The register's value |

Query form

Short form: SREH?
SCPI form: :STATus:SERVice:HENable?

Response

The register's value

Type: <NRi>

9.10.401 SYSDATE
System Date

Reads or sets the system date/time. Setting the date and time is only possible if the internal date/time source is selected via SYSDATESOURCE [9.10.402→406].

Display ID: System Date

Command form
Short form: SYSDATE␣<Date>val
SCPI form: :SYSTem:DATe[:DATE]␣<Date>val

| <Date>val | Config Value

Query form
Short form: SYSDATE?
SCPI form: :SYSTem:DATe[:DATE]?

Response
Config Value

Type: <Date>

9.10.402 SYSDATESOURCE
System date/time source

Selects the source for the instrument's system date and time (SYSDATE [9.10.401→406]). The Time I/O pin of the SYNCHRONIZATION CONNECTOR [4.3→46] can be used to connect with external sinks or sources (that is another LMG or compatible device). If the internal source or NTP is selected, the pin is configured as an output and can be used as a date/time source for other LMGs. By selecting an external source the pin is configured as an input. Note that for synchronization of multiple LMGs to work, the connected devices must run a compatible firmware. See SYNCHRONIZATION CONNECTOR [4.3→46] for further details.

Display ID: System Date Source

Command form
Short form: SYSDATESOURCE␣<NRi>val
SCPI form: :SYSTem:DATe:SOURce␣<NRi>val

| <NRi>val | Config Value

　　　*RST value: 0

　　　Possible values:

　　　0 or INTernal:　LMG is master and uses the internal date/time source (RTC)
　　　1 or OTHerlmg:　LMG is slave and uses an other LMG as date/time source
　　　2 or NTP:　　　The LMG uses the NTP server specified by NTPSERVER [9.10.445→424] to synchronize the system time.

Query form
Short form: SYSDATESOURCE?
SCPI form: :SYSTem:DATe:SOURce?

Response
Config Value

Type: <NRi>

Possible values:

0: LMG is master and uses the internal date/time source (RTC)
1: LMG is slave and uses an other LMG as date/time source
2: The LMG uses the NTP server specified by NTPSERVER [9.10.445→424] to synchronize the system time.

9.10.403 UPTIME
Time since power-on of device

Returns the time that has elapsed since the device has been powered on. Going to standby or disconnecting from power resets this value to zero.

Display ID: System Running Time

Command form
Short form: UPTIME
SCPI form: :SYSTem:DATe:UPTime

Query form
Short form: UPTIME?
SCPI form: :SYSTem:DATe:UPTime?

Response
Running time

Type: <Time>
Unit: s

9.10.404 IDN
Human-readable instrument identification

IDN provides the same information as *IDN [9.10.5→235] but as a correctly quoted SCPI string.

Query form
Short form: IDN?
SCPI form: :SYSTem:DIDentify?

Response
Config Value

Type: <string program data>

9.10.405 ERRALL
Queries all error queues

Shows all errors of the error queue and clears this queue. You can find a list with valid SCPI error codes in SCPI ERROR MESSAGES [9.11→439]

Query form
Short form: ERRALL?
SCPI form: :SYSTem:ERRor:ALL?

Response
List of all errors in the queue

Type: <Error>,...
List elements: 0 ... (dynamic size)

9.10.406 ERRCNT
Queries the error count

Reads the number of errors in the error queue.

Query form
Short form: ERRCNT?
SCPI form: :SYSTem:ERRor:COUNt?

Response
Error count

Type: <NRi>

9.10.407 ERR
Fetch the next error from the errorqueue

Reads the next error from the error queue and removes it from the queue. A list with the valid SCPI error codes you can find in SCPI ERROR MESSAGES [9.11→439]

Query form
Short form: ERR?
SCPI form: :SYSTem:ERRor[:NEXT]?

Response
One error code from the queue

Type: <Errorcode>

9.10.408 GPIODATA
General-purpose I/O pin output data

Specifies the value for the general-purpose output pins of the synchronization connector (see SYNCHRONIZATION CONNECTOR [4.3→46]). Each suffix is associated with a single pin. The direction (input/output) of a general-purpose I/O pin can be configured via GPIODIR [9.10.409→409]. Values for general-purpose input pins may be set but do not take effect as long as the pin is configured as an input.

Suffix: gpio (1 ... 6)

Command form
Short form: GPIODATA␣<NRi>val
SCPI form: :SYSTem:GPIO:DATa␣<NRi>val

| <NRi>val | Config Value |

Query form
Short form: GPIODATA?
SCPI form: :SYSTem:GPIO:DATa?

Response
Config Value

Type: <NRi>

9.10.409 GPIODIR

General-purpose I/O pin direction

Specifies the direction for the synchronization connector's general-purpose I/O pins (see SYNCHRONIZATION CONNECTOR [4.3→46]). Each suffix is associated with a single pin.

Suffix: gpio (1 ... 6)

Command form
Short form: GPIODIR␣<NRi>val
SCPI form: :SYSTem:GPIO:DIRection␣<NRi>val

| <NRi>val | Config Value |

Possible values:

0 or INPUt: Input
1 or OUTPut: Output

Query form
Short form: GPIODIR?
SCPI form: :SYSTem:GPIO:DIRection?

Response
Config Value

Type: <NRi>

Possible values:

0: Input
1: Output

9.10.410 HEAD

List all SCPI command headers

Shows a list of all implemented SCPI commands.

Query form
Short form: HEAD?
SCPI form: :SYSTem:HELP:HEADers?

Response
Config Value

Type: <string program data>,...
List elements: 0 ... (dynamic size)

9.10.411 SHEAD

List all Short command headers

Shows a list of all implemented Short commands.

Query form
Short form: SHEAD?
SCPI form: :SYSTem:HELP:SHEaders?

Response
The list of command headers

Type: <string program data>,...
List elements: 0 ... (dynamic size)

9.10.412 CANBIT
CAN Bus bit-rate

Changes the bit-rate of the CAN interface.

Display ID: Bit-rate

Command form
Short form: CANBIT␣<NRi>val
SCPI form: :SYSTem:INTerface:CAN:BITRate␣<NRi>val

| <NRi>val | Config Value

Possible values:

5000 or K5:	5 kbit/s
10000 or K10:	10 kbit/s
20000 or K20:	20 kbit/s
33333 or K33_333:	33.333 kbit/s
47619 or K47_619:	47.619 kbit/s
50000 or K50:	50 kbit/s
83333 or K83_333:	83.333 kbit/s
95238 or K95_238:	95.238 kbit/s
100000 or K100:	100 kbit/s
125000 or K125:	125 kbit/s
250000 or K250:	250 kbit/s
500000 or K500:	500 kbit/s
800000 or K800:	800 kbit/s
1000000 or M1:	1 Mbit/s

Query form
Short form: CANBIT?
SCPI form: :SYSTem:INTerface:CAN:BITRate?

Response
Config Value

Type: <NRi>

Possible values:

5000:	5 kbit/s
10000:	10 kbit/s
20000:	20 kbit/s
33333:	33.333 kbit/s
47619:	47.619 kbit/s
50000:	50 kbit/s
83333:	83.333 kbit/s
95238:	95.238 kbit/s
100000:	100 kbit/s
125000:	125 kbit/s
250000:	250 kbit/s
500000:	500 kbit/s
800000:	800 kbit/s
1000000:	1 Mbit/s

9.10.413 CANCONT
Cont-On control

This command switches a cont-on via CAN on or off.

Suffix: can (1 ... 128)

Command form
Short form: CANCONT␣<string program data>ctrl
SCPI form: :SYSTem:INTerface:CAN:CONT:CONTinuous␣<string program data>ctrl

| <string program data>ctrl | The Control parameter can be: on (switches it on) allon (switches all cont-ons on) off (switches it off) alloff (switches all cont-ons off) once (switches it on, delivers output only for one measurement cycle, switches back off afterwards)

9.10.414 CANACTN
Defines a CAN Cont-On action

Defines a CAN action in SCPI/Short Syntax. The command is followed by a semicolon and a semicolon separated list of commands, which are considered to be the action to to execute.

Suffix: can (1 ... 128)

Command form
Short form: CANACTN
SCPI form: :SYSTem:INTerface:CAN:CONT[:DEFine]

9.10.415 CANCONTIDEFF
EFF bit of CAN ID for CAN output

Specifies the EFF bit of the CAN ID which shall be sent for a given CANCONT [9.10.413→411].

Display ID: CAN CONT ID EFF bit
Suffix: can (1 ... 128)

Command form
Short form: CANCONTIDEFF␣<Boolean>val
SCPI form: :SYSTem:INTerface:CAN:CONT:EFF␣<Boolean>val

| <Boolean>val | The set value

> *RST value: 0

Query form
Short form: CANCONTIDEFF?
SCPI form: :SYSTem:INTerface:CAN:CONT:EFF?

Response
The set value

Type: <Boolean>

9.10.416 CANCONTID
CAN ID for CAN output

Specifies the CAN ID which shall be sent for a given CANCONT [9.10.413→411].

Display ID: CAN CONT ID
Suffix: can (1 ... 128)

Command form
Short form: CANCONTID␣<NRi>val
SCPI form: :SYSTem:INTerface:CAN:CONT:ID␣<NRi>val

| <NRi>val | The set value

  *RST value: 0

Query form
Short form: CANCONTID?
SCPI form: :SYSTem:INTerface:CAN:CONT:ID?

Response
The set value

Type: <NRi>

9.10.417 CANDBCALL
Export to DBC file format

Exports all CAN slots in dbc-file format.

Query form
Short form: CANDBCALL?
SCPI form: :SYSTem:INTerface:CAN:DBCexport:ALL?

Response
The DBC file as blockdata

Type: <IEEE Block Program Data>

9.10.418 CANDBC
Export to DBC file format

Exports all currently running CAN slots in dbc-file format.

Query form
Short form: CANDBC?
SCPI form: :SYSTem:INTerface:CAN:DBCexport[:RUNNing]?

Response
The DBC file as blockdata

Type: <IEEE Block Program Data>

9.10.419 CANTRIGDBCALL
Export triggers to DBC file format

Exports all CAN trigger slots in dbc-file format.

Query form
Short form: CANTRIGDBCALL?
SCPI form: :SYSTem:INTerface:CAN:DBCexport:TRIGger:ALL?

Response
The DBC file as blockdata

Type: <IEEE Block Program Data>

9.10.420 CANTRIGDBC
Export triggers to DBC file format

Exports all currently running CAN trigger slots in dbc-file format.

Query form
Short form: CANTRIGDBC?
SCPI form: :SYSTem:INTerface:CAN:DBCexport:TRIGger[:RUNNing]?

Response
The DBC file as blockdata

Type: <IEEE Block Program Data>

9.10.421 CANONESHOT
CAN Bus One-shot mode

Switches One-shot mode on/off. Normally, if a CAN message fails to be sent out to the bus, the instrument retries to transmit the message and might give up eventually if the problem prevails. This will turn off the affected CANCONT [9.10.413→411]. In One-shot mode, an outgoing message will only be tried to transmit once. If a problem occurs, the LMG will silently discard the message. The affected CANCONT [9.10.413→411] will continue trying to transmit any new data.

Display ID: One-shot mode

Command form
Short form: CANONESHOT␣<Boolean>val
SCPI form: :SYSTem:INTerface:CAN:OSHot␣<Boolean>val

| <Boolean>val | Config Value

> *RST value: 0

Query form
Short form: CANONESHOT?
SCPI form: :SYSTem:INTerface:CAN:OSHot?

Response
Config Value

Type: <Boolean>

9.10.422 CANTRIGACTNSTR
CAN trigger: Trigger action string

Specifies the action string which is to be executed each time the trigger fires. Any occurences of the formatting symbol "%v" will be replaced with the currently incoming value of the CAN message which triggered this action. See SETTING THE TRIGGER-TABLE [9.9.3→231] for examples.

Suffix: can (1 ... 128)

Command form
Short form: CANTRIGACTNSTR␣<string program data>val
SCPI form: :SYSTem:INTerface:CAN:TRIGger:ACST␣<string program data>val

| <string program data>val | The action string. |

  *RST value: ""

Query form
Short form: CANTRIGACTNSTR?
SCPI form: :SYSTem:INTerface:CAN:TRIGger:ACST?

Response
The action string.

Type: <string program data>

9.10.423 CANTRIGBITLEN
CAN trigger: Bit length of data to inspect

Specifies bit-length of the data-range within the dataframe to look for. Offset plus bit-length must be between 0 and 64. If an incoming frame does not have sufficient data within that range, this frame will be ignored.

Suffix: can (1 ... 128)

Command form
Short form: CANTRIGBITLEN␣<NRi>val
SCPI form: :SYSTem:INTerface:CAN:TRIGger:BITlength␣<NRi>val

| <NRi>val | The bit-length |

  *RST value: 0

Query form
Short form: CANTRIGBITLEN?
SCPI form: :SYSTem:INTerface:CAN:TRIGger:BITlength?

Response
The bit-length

Type: <NRi>

9.10.424 CANTRIGCHECK
CAN trigger: Sanity check

Checks if trigger entry makes sense. If not, an error will be placed in the error queue.

Suffix: can (1 ... 128)

Command form
Short form: CANTRIGCHECK
SCPI form: :SYSTem:INTerface:CAN:TRIGger:CHecK

9.10.425 CANTRIGCOMP
CAN trigger: Comparison operator

Sets the trigger's comparison operator.

Suffix: can (1 ... 128)

Command form
Short form: CANTRIGCOMP␣<NRi>val
SCPI form: :SYSTem:INTerface:CAN:TRIGger:COMParison␣<NRi>val

| <NRi>val | The comparison operator.

    *RST value: 1

    Possible values:

0 or EQ: Test if the incoming data is equal to the comparison value
1 or NE: Test if the incoming data is not equal to the comparison value
2 or LE: Test if the incoming data is less than or equal to the comparison value
3 or LT: Test if the incoming data is less than the comparison value
4 or GE: Test if the incoming data is greater than or equal to the comparison value
5 or GT: Test if the incoming data is greater than the comparison value
6 or TRUE: This evaluates always to true. Useful when receiving a frame is the only information neccesary.

Query form
Short form: CANTRIGCOMP?
SCPI form: :SYSTem:INTerface:CAN:TRIGger:COMParison?

Response
The comparison operator.

Type: <NRi>

Possible values:

0: Test if the incoming data is equal to the comparison value
1: Test if the incoming data is not equal to the comparison value
2: Test if the incoming data is less than or equal to the comparison value
3: Test if the incoming data is less than the comparison value
4: Test if the incoming data is greater than or equal to the comparison value
5: Test if the incoming data is greater than the comparison value
6: This evaluates always to true. Useful when receiving a frame is the only information neccesary.

9.10.426 CANTRIGIDEFF
CAN trigger: CAN ID EFF bit

Specifies if this trigger is intended for CAN messages in extended frame format (EFF) or standard frame format.

Display ID: CAN TRIGGER EFF bit
Suffix: can (1 ... 128)

Command form
Short form: CANTRIGIDEFF␣<Boolean>val
SCPI form: :SYSTem:INTerface:CAN:TRIGger:EFF␣<Boolean>val

| <Boolean>val | The set value

    *RST value: 0

Query form
Short form: CANTRIGIDEFF?
SCPI form: :SYSTem:INTerface:CAN:TRIGger:EFF?

Response
The set value

Type: <Boolean>

9.10.427 CANTRIGOPFLT
CAN trigger: Float operand

Sets up the floating point number comparison operand.

Suffix: can (1 ... 128)

Command form
Short form: CANTRIGOPFLT␣<NRf>val
SCPI form: :SYSTem:INTerface:CAN:TRIGger:FLOat␣<NRf>val

| <NRf>val | The floating point number operand

    *RST value: 0

Query form
Short form: CANTRIGOPFLT?
SCPI form: :SYSTem:INTerface:CAN:TRIGger:FLOat?

Response
The floating point number operand

Type: <NRf>

9.10.428 CANTRIGID
CAN trigger: CAN ID

Sets up to the CAN ID on which the trigger shall react

Suffix: can (1 ... 128)

Command form
Short form: CANTRIGID␣<NRi>val
SCPI form: :SYSTem:INTerface:CAN:TRIGger:ID␣<NRi>val

| <NRi>val | The CAN ID

    *RST value: 0

Query form
Short form: CANTRIGID?
SCPI form: :SYSTem:INTerface:CAN:TRIGger:ID?

Response

The CAN ID

Type: <NRi>

9.10.429 CANTRIGOPINT
CAN trigger: Integer operand

Sets up the integer comparison operand.

Suffix: can (1 ... 128)

Command form
Short form: CANTRIGOPINT␣<NRi>val
SCPI form: :SYSTem:INTerface:CAN:TRIGger:INTeger␣<NRi>val

| <NRi>val | The integer operand

> *RST value: 0

Query form
Short form: CANTRIGOPINT?
SCPI form: :SYSTem:INTerface:CAN:TRIGger:INTeger?

Response

The integer operand

Type: <NRi>

9.10.430 CANTRIGOFFSET
CAN trigger: Offset of data to inspect

The bit-offset within the dataframe to look for. Offset plus bit-length must be between 0 and 64. If an incoming frame does not have sufficient data within that range, this frame will be ignored.

Suffix: can (1 ... 128)

Command form
Short form: CANTRIGOFFSET␣<NRi>val
SCPI form: :SYSTem:INTerface:CAN:TRIGger:OFFset␣<NRi>val

| <NRi>val | The bit-offset

> *RST value: 0

Query form
Short form: CANTRIGOFFSET?
SCPI form: :SYSTem:INTerface:CAN:TRIGger:OFFset?

Response

The bit-offset

Type: <NRi>

9.10.431 CANTRIGSWITCH
CAN trigger: Status

Switches a given trigger on or off.

Suffix: can (1 ... 128)

Command form
Short form: CANTRIGSWITCH␣<NRi>val
SCPI form: :SYSTem:INTerface:CAN:TRIGger:SWiTCh␣<NRi>val

$\boxed{<\text{NRi}>\text{val}}$ The activation status.

> *RST value: 0

> Possible values:

> **0 or OFF:** Trigger is inactive
> **1 or ON:** Trigger is active

Query form
Short form: CANTRIGSWITCH?
SCPI form: :SYSTem:INTerface:CAN:TRIGger:SWiTCh?

Response
The activation status.

Type: <NRi>

Possible values:

0: Trigger is inactive
1: Trigger is active

9.10.432 CANTRIGOPTYPE
CAN trigger: Switch operand type

Defines the type of comparison operand to use.

Suffix: can (1 ... 128)

Command form
Short form: CANTRIGOPTYPE␣<NRi>val
SCPI form: :SYSTem:INTerface:CAN:TRIGger:TYPE␣<NRi>val

$\boxed{<\text{NRi}>\text{val}}$ The type

> *RST value: 0

> Possible values:

> **0 or INTEGER:** Integer type
> **1 or FLOAT:** Floating point type

Query form
Short form: CANTRIGOPTYPE?
SCPI form: :SYSTem:INTerface:CAN:TRIGger:TYPE?

Response
The type

Type: <NRi>

Possible values:

0: Integer type
1: Floating point type

9.10.433 COMBD
Baud rate

Set the baud rate of RS232 (COM) interface. A successfull changing will be confirmed by the device with a break sent to the computer. After that the controlling programme must change the baud rate at the PC and continue. Note: use this command on a single command line.

Display ID: Baud rate

Command form
Short form: COMBD␣<NRi>val
SCPI form: :SYSTem:INTerface:COM:BAUD␣<NRi>val

| <NRi>val | Config Value

       Possible values:

1200 or B1200:	1200 bit/s
2400 or B2400:	2400 bit/s
4800 or B4800:	4800 bit/s
9600 or B9600:	9600 bit/s
19200 or B19200:	19200 bit/s
38400 or B38400:	38400 bit/s
57600 or B57600:	57600 bit/s
115200 or B115200:	115200 bit/s
230400 or B230400:	230400 bit/s

Query form
Short form: COMBD?
SCPI form: :SYSTem:INTerface:COM:BAUD?

Response
Config Value

Type: <NRi>

Possible values:

1200:	1200 bit/s
2400:	2400 bit/s
4800:	4800 bit/s
9600:	9600 bit/s
19200:	19200 bit/s
38400:	38400 bit/s
57600:	57600 bit/s
115200:	115200 bit/s
230400:	230400 bit/s

9.10.434 COMECHO
Echo mode

De-/Activate character echoing.

Display ID: Echo mode

Command form
Short form: COMECHO␣<Boolean>val
SCPI form: :SYSTem:INTerface:COM:ECHo␣<Boolean>val

| <Boolean>val | The echo mode status

Query form
Short form: COMECHO?
SCPI form: :SYSTem:INTerface:COM:ECHo?

Response
The echo mode status

Type: <Boolean>

9.10.435 COMHFC

Hardware flow control

De-/Activate Hardware Flow Control (RTS-CTS-Handshake)

Display ID: Hardware flow control

Command form
Short form: COMHFC␣<Boolean>val
SCPI form: :SYSTem:INTerface:COM:HFControl␣<Boolean>val

| <Boolean>val | Config Value

Query form
Short form: COMHFC?
SCPI form: :SYSTem:INTerface:COM:HFControl?

Response
Config Value

Type: <Boolean>

9.10.436 COMTERM

Termination character

Defines the termination character sent after each SCPI message and expected to be read after each SCPI command. This is useful for terminal-programs in order to get a proper newline after each message.

Display ID: Termination character

Command form
Short form: COMTERM␣<NRi>val
SCPI form: :SYSTem:INTerface:COM:TERM␣<NRi>val

| <NRi>val | The echo mode status

Possible values:

0 or LF: Line feed, the default character at the end of a SCPI message
1 or CR: Carriage return
2 or CRLF: Carriage return, followed by a line feed
3 or TERM: Terminal mode. Sends Carriage return and line feed and accepts carriage
                return as input delimiter.

Query form
Short form: COMTERM?
SCPI form: :SYSTem:INTerface:COM:TERM?

Response
The echo mode status

Type: <NRi>

Possible values:

0: Line feed, the default character at the end of a SCPI message
1: Carriage return
2: Carriage return, followed by a line feed
3: Terminal mode. Sends Carriage return and line feed and accepts carriage return as input delimiter.

9.10.437 IPADDR
IP Address

Specifies the device's static IP address if DHCP (IPDHCP [9.10.439→422]) is disabled. Otherwise this provides the setting as determined by DHCP.

Display ID: IP Address

Command form
Short form: IPADDR␣<string program data>val
SCPI form: :SYSTem:INTerface:LAN:ADDRess␣<string program data>val

| <string program data>val | Config Value |

Query form
Short form: IPADDR?
SCPI form: :SYSTem:INTerface:LAN:ADDRess?

Response
Config Value

Type: <string program data>

9.10.438 IPBCAST
IP broadcast address

Specifies the IP broadcast address if DHCP (IPDHCP [9.10.439→422]) is disabled. Otherwise provides the setting as determined by DHCP.

Display ID: Broadcast Address

Command form
Short form: IPBCAST␣<string program data>val
SCPI form: :SYSTem:INTerface:LAN:BROadcast␣<string program data>val

| <string program data>val | Config Value |

Query form
Short form: IPBCAST?
SCPI form: :SYSTem:INTerface:LAN:BROadcast?

Response
Config Value

Type: <string program data>

9.10.439 IPDHCP

IP DHCP

Specifies whether DHCP (Dynamic Host Configuration Protocol) or a static configuration (see IPADDR [9.10.437→421]) shall be used to determine the device's IP network settings.

Display ID: DHCP

Command form
Short form: IPDHCP␣<Boolean>val
SCPI form: :SYSTem:INTerface:LAN:DHCProtocol␣<Boolean>val

| <Boolean>val | Config Value |

Query form
Short form: IPDHCP?
SCPI form: :SYSTem:INTerface:LAN:DHCProtocol?

Response
Config Value

Type: <Boolean>

9.10.440 IPDNSA

IP domain name server 1

Specifies the domain server 1 if DHCP (IPDHCP [9.10.439→422]) is disabled. Otherwise this provides the setting as determined by DHCP.

Display ID: Domain Server 1

Command form
Short form: IPDNSA␣<string program data>val
SCPI form: :SYSTem:INTerface:LAN:DNSA␣<string program data>val

| <string program data>val | Config Value |

Query form
Short form: IPDNSA?
SCPI form: :SYSTem:INTerface:LAN:DNSA?

Response
Config Value

Type: <string program data>

9.10.441 IPDNSB

IP domain name server 2

Specifies the domain server 2 if DHCP (IPDHCP [9.10.439→422]) is disabled. Otherwise this provides the setting as determined by DHCP.

Display ID: Domain Server 2

Command form
Short form: IPDNSB␣<string program data>val
SCPI form: :SYSTem:INTerface:LAN:DNSB␣<string program data>val

| <string program data>val | Config Value |

Query form
Short form: IPDNSB?
SCPI form: :SYSTem:INTerface:LAN:DNSB?

Response
Config Value

Type: <string program data>

9.10.442 IPGATE
IP gateway address

Specifies the IP address of the default gateway if DHCP (IPDHCP [9.10.439→422]) is disabled. Otherwise provides the setting as determined by DHCP.

Display ID: Gateway

Command form
Short form: IPGATE␣<string program data>val
SCPI form: :SYSTem:INTerface:LAN:GATeway␣<string program data>val

| <string program data>val | Config Value |

Query form
Short form: IPGATE?
SCPI form: :SYSTem:INTerface:LAN:GATeway?

Response
Config Value

Type: <string program data>

9.10.443 IPMAC
MAC address

Provides the MAC address of the LAN interface.

Display ID: MAC Address

Query form
Short form: IPMAC?
SCPI form: :SYSTem:INTerface:LAN:MACad?

Response
Config Value

Type: <string program data>

9.10.444 IPMASK
IP netmask

Specifies the IP network mask if DHCP (IPDHCP [9.10.439→422]) is disabled. Otherwise provides the setting as determined by DHCP.

Display ID: Netmask

Command form
Short form: IPMASK␣<string program data>val
SCPI form: :SYSTem:INTerface:LAN:NETMask␣<string program data>val

| <string program data>val | Config Value |

Query form
Short form: IPMASK?
SCPI form: :SYSTem:INTerface:LAN:NETMask?

Response
Config Value

Type: <string program data>

9.10.445 NTPSERVER

List of NTP server addresses

This list specifies the URLs (or IP addresses) of NTP servers to use when NTP is used as SYSDATESOURCE [9.10.402→406]. Note that name resolution requires a working DNS server, see IPDNSA [9.10.440→422]/IPDNSB [9.10.441→422].

Command form
Short form: NTPSERVER␣[<numeric_list>listVal], <string program data>val
SCPI form: :SYSTem:INTerface:LAN:NTP:LSERver␣[<numeric_list>listVal],
　　　　　　<string program data>val

| [<numeric_list>listVal] | Specifies the positions of the elements to modify. |

　　List elements: 0 ... 0
　　*RST value: Positions 0 ... 0.

| <string program data>val | NTP server URL or IP address |

　　*RST value: ""

Query form
Short form: NTPSERVER?␣[<numeric_list>listVal]
SCPI form: :SYSTem:INTerface:LAN:NTP:LSERver?␣[<numeric_list>listVal]

| [<numeric_list>listVal] | Specifies the positions of the elements to query. |

　　List elements: 0 ... 0

Response
NTP server URL or IP address

Type:　　　　　<string program data>,...
List elements: 0 ... 0

9.10.446 NTPMAXPOLL

NTP maximum poll interval

Specifies the maximum poll interval for the corresponding entry in NTPSERVER [9.10.445→424]. If both NTPMAXPOLL and NTPMINPOLL [9.10.447→425] are set simultaneously then NTPMAXPOLL must be greater than or equal to NTPMINPOLL. Otherwise, if the new NTPMAXPOLL value is less than NTPMINPOLL, the latter is adjusted accordingly.

Command form
Short form: NTPMAXPOLL␣[<numeric_list>listVal], <NRi>val
SCPI form: :SYSTem:INTerface:LAN:NTP:PMAX␣[<numeric_list>listVal], <NRi>val

[<numeric_list>listVal] Specifies the positions of the elements to modify.

List elements: 0 ... 0

<NRi>val NTP maximum poll interval

Possible range: 4 ... 17

Query form
Short form: NTPMAXPOLL?␣[<numeric_list>listVal]
SCPI form: :SYSTem:INTerface:LAN:NTP:PMAX?␣[<numeric_list>listVal]

[<numeric_list>listVal] Specifies the positions of the elements to query.

List elements: 0 ... 0

Response
NTP maximum poll interval

Type: <NRi>,...
List elements: 0 ... 0
Possible range: 4 ... 17

9.10.447 NTPMINPOLL
NTP minimum poll interval

Specifies the minimum poll interval for the corresponding entry in NTPSERVER [9.10.445→424]. If both NTPMINPOLL and NTPMAXPOLL [9.10.446→424] are set simultaneously then NTPMINPOLL must be less than or equal to NTPMAXPOLL. Otherwise, if the new NTPMINPOLL value exceeds NTPMAXPOLL, the latter is adjusted accordingly.

Command form
Short form: NTPMINPOLL␣[<numeric_list>listVal], <NRi>val
SCPI form: :SYSTem:INTerface:LAN:NTP:PMIN␣[<numeric_list>listVal], <NRi>val

[<numeric_list>listVal] Specifies the positions of the elements to modify.

List elements: 0 ... 0

<NRi>val NTP minimum poll interval

Possible range: 4 ... 17

Query form
Short form: NTPMINPOLL?␣[<numeric_list>listVal]
SCPI form: :SYSTem:INTerface:LAN:NTP:PMIN?␣[<numeric_list>listVal]

[<numeric_list>listVal] Specifies the positions of the elements to query.

List elements: 0 ... 0

Response
NTP minimum poll interval

Type: <NRi>,...
List elements: 0 ... 0
Possible range: 4 ... 17

9.10.448 INTSELECT
Select active interface

Selects which peripheral interface shall be activated.

Display ID: Active interface

Command form
Short form: INTSELECT⌴<NRi>val
SCPI form: :SYSTem:INTerface:SELEct⌴<NRi>val

| <NRi>val | Config Value

Possible values:

0 or None: Don't activate any of the interfaces selectable
1 or LAN: LAN interface
2 or SERIAL: Serial interface

Query form
Short form: INTSELECT?
SCPI form: :SYSTem:INTerface:SELEct?

Response
Config Value

Type: <NRi>

Possible values:

0: Don't activate any of the interfaces selectable
1: LAN interface
2: Serial interface

9.10.449 LANG
Changes the interface language

Changes the interface language. Available languages are SCPI and Short.

Command form
Short form: LANG⌴<NRi>val
SCPI form: :SYSTem:LANGuage⌴<NRi>val

| <NRi>val | The interface language

Possible values:

0 or SCPI: The IEEE standard SCPI language. This is the default on startup.
1 or SHORt: The Short language uses the same syntax as SCPI, but shorter commands

9.10.450 TZONELS
List of available time zones

Lists the available time zones. To select a time zone, set TZONE [9.10.456→429] to one of these values.

Query form
Short form: TZONELS?
SCPI form: :SYSTem:LTIMezone?

Response
Config Value

Type: <string program data>,...
List elements: 0 ... (dynamic size)

9.10.451 CLKDISESTERR
Clock discipline estimated error

Clock   discipline   estimated   error.   Please   refer   to   ntp.org   for   further
details.

Query form
Short form: CLKDISESTERR?
SCPI form: :SYSTem:MONitor:CDIScipline:EESTimated?

Response
The measured value

Type: <Time>
Unit: s

9.10.452 CLKDISMAXERR
Clock discipline maximum error

Clock   discipline   maximum   error.   Please   refer   to   ntp.org   for   further
details.

Query form
Short form: CLKDISMAXERR?
SCPI form: :SYSTem:MONitor:CDIScipline:EMAXimum?

Response
The measured value

Type: <Time>
Unit: s

9.10.453 CLKDISPPSDIR
Pulse-Per-Second I/O Direction

Specifies the direction of the PPS (Pulse-Per-Second) I/O pin of the Synchronization Connector
[4.3→46]. When set to output, a short pulse is generated with the rising edge at the change of
seconds.

Command form
Short form: CLKDISPPSDIR␣<NRi>val
SCPI form: :SYSTem:MONitor:CDIScipline:PPSDirection␣<NRi>val

<NRi>val   Config Value

    Possible values:

0 or INPUt: Input
1 or OUTPut: Output

Query form
Short form: CLKDISPPSDIR?
SCPI form: :SYSTem:MONitor:CDIScipline:PPSDirection?

Response
Config Value

Type: <NRi>

Possible values:

0: Input
1: Output

9.10.454 CLKDISSTATE
Clock synchronization state

State of the clock synchronization.

Query form
Short form: CLKDISSTATE?
SCPI form: :SYSTem:MONitor:CDIScipline:STATe?

Response
The measured value

Type: <NRi>

Possible values:

0: The clock is not synchronized to a reliable source.
1: OK

9.10.455 OPTN
Current options

This bit field indicates the currently enabled/disabled optional features of the instrument and supplementary software. Each bit corresponds to an optional feature; if a bit is set (1), the respective feature is installed/enabled, otherwise (0) it is not installed or disabled. The number and types of available channels can be queried separately via CTYP [9.10.294→356].

Query form
Short form: OPTN?
SCPI form: :SYSTem:OPTions?

Response
Config Value

Type: <NRi>

Possible values:

bit 0: DVI-I port (analog and/or digital); L6(X1)-OPT-DVI
bit 1: CAN bus interface; L6-OPT-CAN
bit 3: Processing Signal Interface (PSI); L6-OPT-PSI
bit 8: Star/Delta transformation; L6-OPT-SDC

bit 9: Harmonics; L6-OPT-HRM
bit 10: Flicker; L6-OPT-FLK
bit 11: Event Trigger; L6-OPT-EVT
bit 14: Large data storage; L6-OPT-MSD
bit 15: Smart Vision; L6-OPT-SMV
bit 21: CE Harm module (external software); LMG-TEST-CE-HRM
bit 22: Sample Vision (external software); L6-OPT-SPV
bit 23: CE Standby module; LMG-TEST-CE-STBY
bit 24: CE Flicker module (external software); LMG-TEST-CE-FLK

9.10.456 TZONE
Time zone

Specifies the time zone. The identifiers of all available time zone are listed in TZONELS [9.10.450→426].

Command form
Short form: TZONE␣<string program data>val
SCPI form: :SYSTem:TIMezone␣<string program data>val

| <string program data>val | Config Value |

Query form
Short form: TZONE?
SCPI form: :SYSTem:TIMezone?

Response
Config Value

Type: <string program data>

9.10.457 ACTN
Defines a Cont-On action

Defines an action in SCPI/Short Syntax. The command is followed by a semicolon and a semicolon separated list of commands, which are considered to be the action to to execute. Please note that actions are triggered with a measurement buffer to work on. Using any buffer-changing commands like INIM and SCPI-Read commands result in undefined behaviour.

Command form
Short form: ACTN␣[<string program data>name]
SCPI form: :TRIGger:ACTion[:DEFine]␣[<string program data>name]

| [<string program data>name] | Name of the Action. If none given, the default name "cont" is used. |

9.10.458 IINC
Reset inrush current

This command resets the inrush current IINR [9.10.34→248].

Command form
Short form: IINC
SCPI form: :TRIGger:ICURrent

9.10.459 ENERGYMASK
Group mask for energy measurement

Specifies the groups that are currently selected for energy measurement. The energy mask can be changed using the ENERGYSTART [9.10.461→430] and ENERGYSTOP [9.10.462→431] commands.

Query form
Short form: ENERGYMASK?
SCPI form: :TRIGger:INTerval[:ENERgy]:MASK?

Response
Config Value

Type: <NRi>

9.10.460 ENERGYRESET
Energy measurement reset

This command resets energy measurements for the specified groups.

Command form
Short form: ENERGYRESET␣<NRi>value
SCPI form: :TRIGger:INTerval[:ENERgy]:RESet␣<NRi>value

<NRi>value Specifies the groups whose energy measurements are to be reset.

Possible values:

bit 0 or GRoup1: Group 1
bit 1 or GRoup2: Group 2
bit 2 or GRoup3: Group 3
bit 3 or GRoup4: Group 4
bit 4 or GRoup5: Group 5
bit 5 or GRoup6: Group 6
bit 6 or GRoup7: Group 7

9.10.461 ENERGYSTART
Select groups for energy measurement

Selects the specified groups for energy measurement (in ENERGYMASK [9.10.459→430]). For an energy measurement to actually run the Energy I/O pin must also be asserted, which is controlled by ENERGYMOD [9.10.254→334]. Groups can be deselected via ENERGYSTOP [9.10.462→431]; to reset energy measurements use ENERGYRESET [9.10.460→430].

Command form
Short form: ENERGYSTART␣<NRi>val
SCPI form: :TRIGger:INTerval[:ENERgy]:STARt␣<NRi>val

<NRi>val Specifies which groups to select for energy measurement.

Possible values:

bit 0 or GRoup1: Group 1
bit 1 or GRoup2: Group 2
bit 2 or GRoup3: Group 3

bit 3 or GRoup4: Group 4
bit 4 or GRoup5: Group 5
bit 5 or GRoup6: Group 6
bit 6 or GRoup7: Group 7

9.10.462 ENERGYSTOP
Deselect groups for energy measurement

Deselects the specified groups for energy measurement (in ENERGYMASK [9.10.459→430]). A running energy measurement for the specified groups is stopped immediately. Depending on the ENERGYMOD [9.10.254→334] setting this may or may not affect the value output at the Energy I/O pin of the SYNCHRONIZATION CONNECTOR [4.3→46]. Groups can be selected via ENERGYSTART [9.10.461→430]; to reset energy measurements use ENERGYRESET [9.10.460→430].

Command form
Short form: ENERGYSTOP␣<NRi>val
SCPI form: :TRIGger:INTerval[:ENERgy]:STOP␣<NRi>val

<NRi>val Specifies which groups to deselect for energy measurement.

Possible values:

bit 0 or GRoup1: Group 1
bit 1 or GRoup2: Group 2
bit 2 or GRoup3: Group 3
bit 3 or GRoup4: Group 4
bit 4 or GRoup5: Group 5
bit 5 or GRoup6: Group 6
bit 6 or GRoup7: Group 7

9.10.463 TRANSIENTABORT
Abort transient search

This command stops the transient system.

Command form

Short form: TRANSIENTABORT
SCPI form: :TRIGger:INTerval:TRANSient:TABORT

9.10.464 TRANSIENTNOW
Trigger transient now

This command triggers an immediate transient recording, if the pretrigger sample are already recorded.

Command form

Short form: TRANSIENTNOW
SCPI form: :TRIGger:INTerval:TRANSient:TNOW

9.10.465 TRANSIENTRESTART
Start transient search

This command starts the transient system and the filling of the pretrigger buffer begins.

Command form
Short form: TRANSIENTRESTART
SCPI form: :TRIGger:INTerval:TRANSient:TRESTART

9.10.466 SCSYNC
Scope synchronization group

This command attaches the oscilloscope to the synchronization of the group specified. See also SYNC [9.10.477→437].

Suffix: sp (1, 2, 21 and 22)

Command form
Short form: SCSYNC␣<NRi>val
SCPI form: :TRIGger:SCOPe:SYNC␣<NRi>val

| <NRi>val | Config Value |

   Possible range: 1 ... 7
   *RST value: 1

Query form
Short form: SCSYNC?
SCPI form: :TRIGger:SCOPe:SYNC?

Response
Config Value

Type: <NRi>
Possible range: 1 ... 7

9.10.467 SYNCBW
Synchronization signal bandwidth

When a U or I channel is selected as a group's SYNC [9.10.477→437] source, this setting specifies the sample/signal path to use for synchronization. The actual signal measured by that channel is then (after dedicated preprocessing as appropriate) also used for synchronization. The possible signal/sample paths differ in their available bandwidth and the fixed and/or optional preprocessing.

- The Narrow Bandwidth synchronization derives from the measured narrow bandwidth signal. Preprocessing includes amplitude demodulation (TDEM [9.10.468→433]) and/or filtering (TLPCOF [9.10.474→436], THPCOF [9.10.469→434]). In addition, the sync level (TLEV [9.10.473→436]) and hysteresis (THYS [9.10.471→435]) can be configured.

- The Wide Bandwidth synchronization derives from the measured wide bandwidth signal. The sync level (TLEV [9.10.473→436]) and hysteresis (THYS [9.10.471→435]) can be configured.

- The Full Bandwidth synchronization is provided by an HF comparator which directly generates the sync from the unfiltered input signal applied to the instrument's input jacks.

See SYNCHRONIZATION [6.9→118] for further explanations; Figure 6.8 [→109] illustrates the various (synchronization) signal/sample paths of a U/I channel.

Display ID: Synchronization Bandwidth
Suffix: g (1 ... 7)

Command form
Short form: SYNCBW␣<NRi>val
SCPI form: :TRIGger[:SEQuence]:BANDwidth␣<NRi>val

| <NRi>val | Config Value

    *RST value: 0

Possible values:

0 or NARRow: Use configurable software filters to find sync signal
1 or WIDe: Use configurable software comparator to find sync signal
2 or FULL: Use simple hardware comparator to find sync signal

Query form
Short form: SYNCBW?
SCPI form: :TRIGger[:SEQuence]:BANDwidth?

Response
Config Value

Type: <NRi>

Possible values:

0: Use configurable software filters to find sync signal
1: Use configurable software comparator to find sync signal
2: Use simple hardware comparator to find sync signal

9.10.468 TDEM
Demodulation

Controls the optional demodulation in the narrow bandwidth sync path. Note that this setting only takes effect when a U or I channel is selected as the group's sync source (via SYNC [9.10.477→437]) *and* the narrow bandwidth path is selected (via SYNCBW [9.10.467→432]). See SYNCHRONIZATION [6.9→118] for further explanations; BANDWIDTH [6.6→110] illustrates the various (synchronization) signal/sample paths of a U/I channel.

Display ID: Demodulation
Suffix: g (1 ... 7)

Command form
Short form: TDEM␣<Boolean>val
SCPI form: :TRIGger[:SEQuence]:DEModulator␣<Boolean>val

| <Boolean>val | Config Value

    *RST value: 0

Query form
Short form: TDEM?
SCPI form: :TRIGger[:SEQuence]:DEModulator?

Response
Config Value

Type: <Boolean>

9.10.469 THPCOF

High-pass sync filter cutoff frequency

This command sets the cutoff frequency for the highpass filter of the sync signal. The lowpass and highpass filter of the sync signal work together as a bandpass. The lower cutoff frequency of this bandpass can be set by THPCOF [9.10.469→434] and the upper by TLPCOF [9.10.474→436]. This setting only takes effect if a custom highpass filter is selected via THPFILT [9.10.470→434].

Display ID: Sync HP Cut Off
Suffix: g (1 ... 7)

Command form
Short form: THPCOF␣<NRf>val
SCPI form: :TRIGger[:SEQuence]:HPASs[:COFRequency]␣<NRf>val

| <NRf>val | Config Value

      Possible range: 0.1 ... 15000
      *RST value: 1
      Unit: Hz

Query form
Short form: THPCOF?
SCPI form: :TRIGger[:SEQuence]:HPASs[:COFRequency]?

Response
Config Value

Type: <NRf>
Possible range: 0.1 ... 15000
Unit: Hz

9.10.470 THPFILT

High-pass sync filter mode

Selects the highpass sync filter when using narrow bandwidth synchronization (see SYNCBW [9.10.467→432]). The cutoff frequency for a custom digital highpass filter can be specified via THPCOF [9.10.469→434]. A lowpass sync filter can be selected via TLPFILT [9.10.475→437].

Display ID: Sync HP Mode
Suffix: g (1 ... 7)

Command form
Short form: THPFILT␣<NRi>val
SCPI form: :TRIGger[:SEQuence]:HPASs:FSTate␣<NRi>val

| <NRi>val | Config Value

      *RST value: 0

      Possible values:

      0 or OFF: Filter off
      1 or CUSTom: Custom digital filter settings

Query form
Short form: THPFILT?
SCPI form: :TRIGger[:SEQuence]:HPASs:FSTate?

Response
Config Value

Type: <NRi>

Possible values:

0: Filter off
1: Custom digital filter settings

9.10.471 THYS
Trigger hysteresis

Sets the hysteresis of the extended trigger.

Display ID: Sync Hysteresis
Suffix: g (1 ... 7)

Command form
Short form: THYS␣<NRf>val
SCPI form: :TRIGger[:SEQuence]:HYSTeresis␣<NRf>val

| <NRf>val | Config Value

> Possible range: 0.0 ... 100.0
> *RST value: 2
> Unit: %

Query form
Short form: THYS?
SCPI form: :TRIGger[:SEQuence]:HYSTeresis?

Response
Config Value

Type: <NRf>
Possible range: 0.0 ... 100.0
Unit: %

9.10.472 SYNCBWLS
List of available synchronization paths

List of all currently valid (that is selectable via SYNCBW [9.10.467→432]) synchronization paths.

Suffix: g (1 ... 7)

Query form
Short form: SYNCBWLS?
SCPI form: :TRIGger[:SEQuence]:LBandwith?

Response
Config Value

Type: <NRi>,...
List elements: 0 ... (dynamic size)

9.10.473 TLEV
Trigger level

Reads and sets the level of the extended triggers.

Display ID: Sync Level
Suffix: g (1 ... 7)

Command form
Short form: TLEV␣<NRf>val
SCPI form: :TRIGger[:SEQuence]:LEVel␣<NRf>val

| <NRf>val | Config Value

     Possible range: -100.1 ... 100.0
     *RST value: 0
     Unit: %

Query form
Short form: TLEV?
SCPI form: :TRIGger[:SEQuence]:LEVel?

Response
Config Value

Type: <NRf>
Possible range: -100.1 ... 100.0
Unit: %

9.10.474 TLPCOF
Low-pass sync filter cutoff frequency

This command sets the cutoff frequency for the lowpass filter of the sync signal. The lowpass and highpass filter of the sync signal work together as a bandpass. The lower cutoff frequency of this bandpass can be set by THPCOF [9.10.469→434] and the upper by TLPCOF [9.10.474→436]. This setting only takes effect if a custom lowpass filter is selected via TLPFILT [9.10.475→437].

Display ID: Sync LP Cut Off
Suffix: g (1 ... 7)

Command form
Short form: TLPCOF␣<NRf>val
SCPI form: :TRIGger[:SEQuence]:LPASs[:COFRequency]␣<NRf>val

| <NRf>val | Config Value

     Possible range: 0.1 ... 15000
     *RST value: 2000
     Unit: Hz

Query form
Short form: TLPCOF?
SCPI form: :TRIGger[:SEQuence]:LPASs[:COFRequency]?

Response
Config Value

Type: <NRf>
Possible range: 0.1 ... 15000
Unit: Hz

9.10.475 TLPFILT

Low-pass sync filter mode

Selects the lowpass sync filter when using narrow bandwidth synchronization (see sᴙɴᴄʙᴡ [9.10.467→432]). The cutoff frequency for a custom digital lowpass filter can be specified via ᴛʟᴘᴄᴏꜰ [9.10.474→436]. A highpass sync filter can be selected via ᴛʜᴘꜰɪʟᴛ [9.10.470→434].

Display ID: Sync LP Mode
Suffix: g (1 ... 7)

Command form
Short form: TLPFILT␣<NRi>val
SCPI form: :TRIGger[:SEQuence]:LPASs:FSTate␣<NRi>val

| <NRi>val | Config Value

    *RST value: 0

    Possible values:

    0 or NARRow: Narrowband converter
    1 or CUSTom: Custom digital filter settings

Query form
Short form: TLPFILT?
SCPI form: :TRIGger[:SEQuence]:LPASs:FSTate?

Response
Config Value

Type: <NRi>

Possible values:

0: Narrowband converter
1: Custom digital filter settings

9.10.476 SYNCLS

List of available synchronization sources

List of all currently valid (that is selectable via sᴙɴᴄ [9.10.477→437]) synchronization sources.

Suffix: g (1 ... 7)

Query form
Short form: SYNCLS?
SCPI form: :TRIGger[:SEQuence]:LSource?

Response
Config Value

Type: <NRi>,...
List elements: 0 ... (dynamic size)

9.10.477 SYNC

Synchronization source

Selects the synchronization source for a group. An overview of the synchronization subsystem with further pointers to related sections can be found in Sʏɴᴄʜʀᴏɴɪᴢᴀᴛɪᴏɴ [6.9→118]. The valid choices are either one of the following:

- Any U/I channel in the *same* group. U/I channels of other groups are not permitted (at least not directly, see next item).

- Any other group that does not synchronize (directly or indirectly) on the current group. The actual source is determined by the SYNC setting for the selected group, that is it depends on and follows any changes to that setting! Note that the selected source group may itself synchronize on another group, and so forth, until a genuine signal source is found.

- Synchronize on an external signal applied to the Sync Frequency I/O pin of the synchronization connector (see SYNCHRONIZATION CONNECTOR [4.3→46]).

This settings also has an effect on the Sync Frequency I/O pin of the SYNCHRONIZATION CONNECTOR [4.3→46]. If any group selects an external sync source (EXTS), that pin is configured as an input; it is configured as an output otherwise, providing the sync of group 1.

Display ID: Sync Source
Suffix: g (1 ... 7)

Command form
Short form: SYNC␣<NRi>val
SCPI form: :TRIGger[:SEQuence]:SOURce␣<NRi>val

| <NRi>val | Config Value

        *RST value: 23

        Possible values:

1 or EXTS:	Sync Frequency I/O pin of synchronization connector (see SYNCHRONIZATION CONNECTOR [4.3→46])
2 or U1:	U channel 1
3 or U2:	U channel 2
4 or U3:	U channel 3
5 or U4:	U channel 4
6 or U5:	U channel 5
7 or U6:	U channel 6
8 or U7:	U channel 7
9 or I1:	I channel 1
10 or I2:	I channel 2
11 or I3:	I channel 3
12 or I4:	I channel 4
13 or I5:	I channel 5
14 or I6:	I channel 6
15 or I7:	I channel 7
16 or G1:	Group 1
17 or G2:	Group 2
18 or G3:	Group 3
19 or G4:	Group 4
20 or G5:	Group 5
21 or G6:	Group 6
22 or G7:	Group 7
23 or RU1:	1st U channel of group
24 or RU2:	2nd U channel of group
25 or RU3:	3rd U channel of group
26 or RU4:	4th U channel of group
27 or RU5:	5th U channel of group
28 or RU6:	6th U channel of group
29 or RU7:	7th U channel of group
30 or RI1:	1st I channel of group
31 or RI2:	2nd I channel of group
32 or RI3:	3rd I channel of group

33 or RI4: 4th I channel of group
34 or RI5: 5th I channel of group
35 or RI6: 6th I channel of group
36 or RI7: 7th I channel of group

Query form
Short form: SYNC?
SCPI form: :TRIGger[:SEQuence]:SOURce?

Response
Config Value

Type: <NRi>

Possible values:

1: Sync Frequency I/O pin of synchronization connector (see SYNCHRONIZATION CONNECTOR [4.3→46])
2: U channel 1
3: U channel 2
4: U channel 3
5: U channel 4
6: U channel 5
7: U channel 6
8: U channel 7
9: I channel 1
10: I channel 2
11: I channel 3
12: I channel 4
13: I channel 5
14: I channel 6
15: I channel 7
16: Group 1
17: Group 2
18: Group 3
19: Group 4
20: Group 5
21: Group 6
22: Group 7
23: 1st U channel of group
24: 2nd U channel of group
25: 3rd U channel of group
26: 4th U channel of group
27: 5th U channel of group
28: 6th U channel of group
29: 7th U channel of group
30: 1st I channel of group
31: 2nd I channel of group
32: 3rd I channel of group
33: 4th I channel of group
34: 5th I channel of group
35: 6th I channel of group
36: 7th I channel of group

9.11 SCPI error messages

-101 Invalid character
Indicates, that an invalid character was sent, propably a '(' or ')' is missing in a <list>

-103 Invalid separator

Valid separators are only ',', ';', ':' and <EOS>

-110 Command header error

Indicates, that the command does not exist or is misspelled

-200 Execution error

A generic execution error has occured.

-220 Parameter error

An illegal parameter value was encountered.

-221 Settings conflict

Setting could not be executed due to the current device state.

-222 Data out of range

The interpreted value was outside the legal range as defined by the device.

-224 Illegal parameter value

An illegal parameter value was encountered. Expected an exact value from a list of possibles.

-283 Illegal variable name

Variable does not exist.

-310 System error

A system error occured. This generally indicates some unexpected low-level hardware or software failure that prevents the instrument from working as intended. The device-dependent part of the error response should provide additional hints as to the underlying cause. Please contact support for further assistance.

9.12 Remote examples

In the following section one can find examples how a typical communication can look like. The examples base on excerpts of Python code without for example showing opening or closing of the interface.

9.12.1 Reading sample values of an event

The following example sets up, that the smallband voltage of channel 1 in group 1 as well as the corresponding current should be recorded with maximum record rate and no pretrigger.

The recording starts as soon as the voltage on channel 1 rises above 120V.

```
### NOTE: The following code is programmed in python. ###

# switch to short language
device.write('LANG SHORt')

# set the signal for the track
device.write('TRCTRAC 0, "U1111"')
device.write('TRCTRAC 1, "I1111"')

# set record rate to 1.2 MS/s
device.write('TRCSR 1200000')

# set pretrigger to 0 s
device.write('TRCPTRT 0')

# make sure the CYCLMOD is not SCOPE
# use e.g. CYCLetime instead
device.write('CYCLMOD CYCLetime')

# provide the transient with the sample memory
```

```
device.write('SAMPLESTORAGEMODE␣EVENt')

# set the signal to be monitored
# doesn't need to be recorded itself,
# but will be recorded in this case on track 1 (see above)
# up to 3 signals and conditions can be set up (TRSOURCE2 & TRSOURCE3)
device.write('TRSOURCE1␣'u1111'')

# set limit A to 120
device.write('TRLIMITA1␣120')

# set up trigger condition to trigger if signal (TRSOURCE1) is higher than TRLIMITA
device.write('TRCOND1␣GTA')

# ensure that the trigger fires only if the condition changes from false to true
# thus, the trigger occurs always on the rising edge
device.write('TRTRIGMODE␣Edge')

# start the transient system
device.write('TRANSIENTRESTART')

# wait until TRPSTAT is 2 (searching) - max. 5 seconds
timeout = 0
while timeout < 5:
    device.write('INIM␣NOW')
    device.write('TRPSTAT?')
    stat = int(device.read())
    if stat == 2:
        break
    time.sleep(1)
    timeout += 1

# start recording
device.write('TRANSIENTNOW')

# wait until TRPSTAT is >= 5 (finished) - max. 5 seconds
timeout = 0
while timeout < 5:
    device.write('INIM␣NOW')
    device.write('TRPSTAT?')
    stat = int(device.read())
    if stat >= 5:
        break
    time.sleep(1)
    timeout += 1

# read from track 0, samples 0 to 49
device.write('TRPVAL?␣0,␣(0:49)')
# read the samples and fill u
u = device.read()

# read from track 1, samples 0 to 49
device.write('TRPVAL?␣1,␣(0:49)')
# read the samples and fill i
i = device.read()
```

Instead of 50 values per each track also the maximum possible value per track can be requested.

9.12.2 Continuous Output of scope values

This example shows how to set up and query continous, gapless output of sample values. Please note that this excludes the usage of the transient and the flicker.

```
### NOTE: The following code is programmed in python. ###

# switch to short language
device.write('LANG␣SHORt')

# set the signal for the track
device.write('GLCTRAC␣0,␣'U1111'')
```

```
# set record rate to 100 samples per second
device.write('GLCSR␣100')

# adapt the cycle length to the record rate
device.write('CYCLMOD␣SCOPE')

# wait until next measurement cycle
device.write('INIM')
# fetch the number of samples per cycle
device.write('GLPTLEN?')
# read the number of samples per cycle
gaplen = int(device.read())

# defines the action, which queries scope values from the first track
device.write('ACTN;␣GLPVAL?␣0,␣(0:%d)' % (gaplen-1))

# starts the continuous output
device.write('CONT␣ON')

# read data
data = device.read()

# stops the continuous output
device.write('CONT␣OFF')
# read the '1' from CONT OFF command
ret = device.read()
```

9.12.3 Mass Storge Subsystem

This example shows how to read a file from the LMG and transfer it to a local file on the PC.

```
### NOTE: The following code is programmed in python. ###

# switch to short language
device.write('LANG␣SHORt')

# set current working directory to the documentation subpath of the library path,
device.write('MMCDIR␣''/library/documentation''')

# request all data of the manual
device.write('MMDATA?␣''LMG600_en_US.pdf''')

# data of MMDATA is replied by LMG always in binary format.
data = device.readbinaryblock()

# open a local file in binary mode and write all data to it
manual = open('LMG600_en_US.pdf', 'wb')
manual.write(data)
manual.close()
```

Refer to BINARY OUTPUT [9.4.2→222] for the documentation of the binary format. The answer of MMDATA [9.10.228→323] used in the example must be read by decoding the binary blocks returned by it. Reading till the end of line is not enough, because the file to be read may contain linefeeds itself.

Hint: you can use frmt 1 to switch to binary output, which is faster to provide and to transfer.

Index