# Programming Guide

## P-Series Power Meters

**N1911A and N1912A**

Agilent Technologies

**Manufacturing Part Number: N1912-90008**

**Printed in UK**

**February 2005**

# Notices



Station Road, South Queensferry, Scotland, EH30 9TG, UK.

## General Warranty

The material contained in this document is provided “as is,” and is subject to being changed, without notice, in future editions. Further, to the maximum extent permitted by applicable law, Agilent disclaims all warranties, either express or implied with regard to this manual and any information contained herein, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. Agilent shall not be liable for errors or for incidental or consequential damages in connection with the furnishing, use, or performance of this document or any information contained herein. Should Agilent and the user have a separate written agreement with warranty terms covering the material in this document that conflict with these terms, the warranty terms in the separate agreement will control.

## Restricted Rights Legend

If software is for use in the performance of a U.S. Government prime contract or subcontract, Software is delivered and licensed as “Commercial computer software” as defined in DFAR 252.227-7014 (June 1995), or as a “commercial item” as defined in FAR 2.101(a) or as “Restricted computer software” as defined in FAR 52.227-19 (June 1987) or any equivalent agency regulation or contract clause. Use, duplication or disclosure of Software is subject to Agilent Technologies’ standard commercial license terms, and non-DOD Departments and Agencies of the U.S. Government will receive no greater than Restricted Rights as defined in FAR 52.227-19(c)(1-2) (June 1987). U.S. Government users will receive no greater than Limited Rights as defined in FAR 52.227-14 (June 1987) or DFAR 252.227-7015 (b)(2) (November 1995), as applicable in any technical data.

# Equipment Operation

### Warnings and Cautions

This guide uses warnings and cautions to denote hazards.

**WARNING**

**A warning calls attention to a procedure, practice or the like, which, if not correctly performed or adhered to, could result in injury or the loss of life. Do not proceed beyond a warning until the indicated conditions are fully understood and met.**

**CAUTION**

A caution calls attention to a procedure, practice or the like which, if not correctly performed or adhered to, could result in damage to or the destruction of part or all of the equipment. Do not proceed beyond a caution until the indicated conditions are fully understood and met.

### Personal Safety Considerations

**WARNING**

**This is a Safety Class I product (provided with a protective earthing ground incorporated in the power cord). The mains plug shall only be inserted in a socket outlet provided with a protective earth contact. Any interruption of the protective conductor, inside or outside the instrument, is likely to make the instrument dangerous. Intentional interruption is prohibited.**
**If this instrument is not used as specified, the protection provided by the equipment could be impaired. This instrument must be used in a normal condition (in which all means of protection are intact) only.**
**No operator serviceable parts inside. Refer servicing to qualified personnel. To prevent electrical shock, do not remove covers.**
**For continued protection against fire hazard, replace the line fuse(s) only with fuses of the same type and rating (for example, normal blow, time delay, etc.). The use of other fuses or material is prohibited.**

# General Safety Considerations

**WARNING**

**Before this instrument is switched on, make sure it has been properly grounded through the protective conductor of the ac power cable to a socket outlet provided with protective earth contact.**
**Any interruption of the protective (grounding) conductor, inside or outside the instrument, or disconnection of the protective earth terminal can result in personal injury.**

**CAUTION**

Any adjustments or service procedures that require operation of the instrument with protective covers removed should be performed only by trained service personnel.

# User Environment

The product is suitable for indoor use only.

# About this Guide

- **Chapter 1: Power Meter Remote Operation**

  This chapter describes the parameters which configure the power meter and helps you determine settings to optimize performance.

- **Chapter 2: MEASurement Instructions**

  This chapter explains how to use the MEASure group of instructions to acquire data using a set of high level instructions.

- **Chapter 3: CALCulate Subsystem**

  This chapter explains how to use the CALCulate subsystem to perform post acquisition data processing.

- **Chapter 4: CALibration Subsystem**

  This chapter explains how to use the CALibration command subsystem to zero and calibrate the power meter.

- **Chapter 5: DISPlay Subsystem**

  This chapter explains how the DISPlay subsystem is used to control the selection and presentation of the windows used on the power meter's display.

- **Chapter 6: FORMat Subsystem**

  This chapter explains how the FORMat subsystem is used to set a data format for transferring numeric information.

- **Chapter 7: MEMory Subsystem**

  This chapter explains how the MEMory command subsystem is used to create, edit and review sensor calibration tables.

- **Chapter 8: OUTput Subsystem**

  This chapter explains how the OUTput command subsystem is used to switch the POWER REF output on and off.

- **Chapter 9: SENSe Subsystem**

  This chapter explains how the SENSe command subsystem directly affects device specific settings used to make measurements.

- **Chapter 10: STATus Subsystem**

  This chapter explains how the STATus command subsystem enables you to examine the status of the power meter by monitoring the "Device Status Register", "Operation Status Register" and the "Questionable Status Register".

- **Chapter 11: SYSTem Subsystem**

  This chapter explains how to use the SYSTem command subsystem to return error numbers and messages from the power meter, preset the power meter, set the remote address, and query the SCPI version.

- **Chapter 12: TRACe Subsystem**

  This chapter explains how to use the TRACe command subsystem to configure and read back the measured power trace.

- **Chapter 13: TRIGger Subsystem**

  This chapter explains how the TRIGger command subsystem is used synchronize device actions with events.

- **Chapter 14: UNIT Subsystem**

  This chapter explains how to use the UNIT command subsystem to set the power meter measurement units to Watts and % (linear), or dBm and dB (logarithmic).

- **Chapter 15: SERVice Subsystem**

  This chapter explains how to use the SERVice command subsystem to obtain and set information useful for servicing the power meter.

- **Chapter 16: IEEE488.2 Command Reference**

  This chapter contains information about the IEEE488.2 Common Commands that the power meter supports.

# Related Publications

The *P-Series Power Meters User's Guide* is available on the CD-ROM and printed in the following languages:

- English Language User's Guide - ABA
- French Language User's Guide - ABF
- Japanese Language User's Guide - ABJ

Useful information on SCPI (Standard Commands for Programmable Instruments) can be found in:

- *A Beginner's Guide to SCPI*, which is available by ordering Agilent Part Number 5010-7166.
- The SCPI reference manuals which are available from:
  SCPI Consortium,
  8380 Hercules Drive, Suite P3,
  La Mesa, CA 91942, USA.
  Telephone: 619-697-4301
  Fax: 619-697-5955

# Contents

# Contents

# Contents

# 1 Power Meter Remote Operation

# Introduction

This chapter describes the parameters which configure the power meter and help you determine settings to optimize performance. It contains the following sections:

# Configuring the Remote Interface

This section briefly describes how to configure the GPIB, LAN and USB remote interfaces.

**NOTE** For more information on configuring the remote interface connectivity, refer to the *Agilent Technologies USB/LAN/GPIB Interfaces Connectivity Guide*. If you have installed the *IO Libraries Suite*, you can access the *Connectivity Guide* via the Agilent IO Libraries Control icon. Alternatively, you can access the *Connectivity Guide* via the Web at www.agilent.com/find/connectivity.

## Interface Selection

You can choose to control the power meter remotely using the GPIB, LAN or USB interfaces.

For information on selecting and configuring the remote interface manually from the front panel, refer to the *P-Series Power Meters Installation Guide*.

**NOTE** It is expected that most users will use the front panel keys to set up the remote interfaces. The remote interface commands are provided for completeness (for the front panel operation).

## GPIB Address

Each device on the GPIB (IEEE-488) interface must have a unique address. You can set the power meter's address to any value between 0 and 30. The power meter is shipped with a default address set to 13. The GPIB address is stored in non-volatile memory, and does not change when the power meter is switched off, or after a remote interface reset.

Your GPIB bus controller has its own address. Avoid using the bus controller's address for any instrument on the interface bus. Agilent Technologies controllers generally use address 21.

For information on setting the GPIB address manually from the front panel, refer to the *P-Series Power Meters Installation Guide*.

- To set the GPIB address from the remote interface use the:

  `SYSTem:COMMunicate:GPIB:ADDRess` command.

- To query the GPIB address from the remote interface use the:

  `SYSTem:COMMunicate:GPIB:ADDRess?` query.

## LAN Configuration

The power meter has three LAN operating modes:

- Dynamic IP (Dynamic Host Configuration Protocol or DHCP)
- Auto IP (Local PC Control or isolated (non-site) LAN)
- Static IP (Manual mode)

These three modes can be set up from the front panel. For front panel operation refer to the *P-Series Power Meter Installation Guide*.

### Configuring the LAN Remotely

To automatically configure the LAN settings, enable DHCP operation using the `SYSTem:COMMunicate:LAN:DHCP[:STATe]` command.

In this Dynamic IP mode the IP Address, Subnet Mask, and Default Gateway values are obtained from a DHCP server. Using this Dynamic IP mode does not require a detailed knowledge of your network configuration.

The IP Address, Subnet Mask, Default Gateway, and Host settings can be changed manually or remotely. To individually specify the LAN settings use the following commands:

- IP Address - `SYSTem:COMMuniucate:LAN:ADDRess`
- Subnet Mask - `SYSTem:COMMunicate:LAN:SMASk`
- Default Gateway - `SYSTem:COMMunicate:LAN:DGATeway`
- Domain Name - `SYSTem:COMMunicate:LAN:DNAMe`
- Hostname - `SYSTem:COMMunicate:LAN:HNAMe`
- Restart Network - `SYSTem:COMMunicate:LAN:RESTart`

The `character_data` values for the IP address, Subnet Mask, and Default Gateway can range between 0.0.0.0 and 255.255.255.255.

**NOTE** If you configure an invalid IP Address or an IP address that is used by another device or host, an error message is generated. This error can be read by using the `SYSTem:ERRor?` command.

The LAN setting values are stored in non-volatile memory and are not part of the save-recall function.

## USB Configuration

The USB interface requires no front panel or remote configuration.

The USB address cannot be changed - it is set at the factory and is unique for each power meter.

**NOTE** For further information about the USB configuration refer to the *P-Series Power Meters Installation Guide*.

**NOTE** Before connecting the USB cable, make sure that I/O software is installed on your computer.

**NOTE** For information about *Agilent IO Libraries* software refer to the *Connectivity Guide*.

If you have installed other I/O software, refer to documentation that accompanies the software.

# Zeroing and Calibrating the P-Series Power Sensor

P-series Wideband Power Sensor's do not need manual calibration and zero routines performed. These are performed without removing the power sensor from the source.

## Zeroing

Zeroing adjusts the power meter's specified channel for a zero power reading.

The command `CALibration[1]|2:ZERO:AUTO [ONCE|ON|OFF|0|1]` causes the power meter to perform its zeroing routine on the specified channel when enabled. This adjusts the power meter for a zero power reading with no power supplied to the power sensor.

`1|ON` can only be used with a P-series sensor. When `1|ON` is enabled the the zero is maintained by a combination of zero *on-the-fly* for measurements and temperature compensation.

Zeroing of the power meter happens automatically:

- when a 5°C change in temperature occurs.
- when you change the power sensor.
- every 24 hours.
- prior to measuring low level signals. For example, 10 dB above the lowest specified power for your power sensor.

## Calibration

This command used to calibrate the power meter is:

`CALibration[1|2]:AUTO ONCE`

. It is recommended that you zero the power meter before calibrating.

### Calibration Sequence

This feature allows you to perform a complete calibration sequence with a single query. The query is:

`CALibration[1|2][:ALL]?`

The query assumes that the power sensor is connected to the power reference oscillator. It turns the power reference oscillator on, then after calibrating, returns the power reference oscillator to the same state it was in prior to the command being received. The calibration sequence consists of:

1. Zeroing the power meter (`CALibration[1|2]:ZERO:AUTO ONCE`).
2. calibrating the power meter (`CALibration[1|2]:AUTO ONCE`).

The query enters a number into the output buffer when the sequence is complete. If the result is 0 the sequence was successful. If the result is 1 the sequence failed. Refer to "CALibration[1]|2[:ALL]?" on page 176 for further information.

**NOTE** The `CALibration[1|2][:ALL]` command is identical to the `CALibration[1|2][:ALL]?` query except that no number is returned to indicate the outcome of the sequence. You can examine the Questionable Status Register or the error queue to discover if the sequence has passed or failed. Refer to "Status Reporting" on page 46 for further information.

# Making Measurements

The MEASure? and CONFigure commands provide a straight-forward method to program the power meter for measurements. You can select the measurement's expected power level, resolution and with the N1912A the measurement type (that is single channel, difference or ratio measurements) all in one command. The power meter automatically presets other measurement parameters to default values as shown in Table 1-1.

**Table 1-1 MEASure? and CONFigure Preset States**

| Command | MEASure? and CONFigure Setting |
|---|---|
| Trigger source (TRIGger:SOURce) | Immediate |
| Filter (SENSe:AVERage:COUNt:AUTO) | On |
| Filter state(SENSe:AVERage:STATe) | On |
| Trigger cycle (INITiate:CONTinuous) | Off |
| TriggerDelay (TRIGger:DELay:AUTO) | On |

An alternative method to program the power meter is to use the lower level commands. The advantage of using the lower level commands over the CONFigure command is that they give you more precise control of the power meter. As shown in Table 1-1 the CONFigure command presets various states in the power meter. It may be likely that you do not want to preset these states. Refer to "Using the Lower Level Commands" on page 23 for further information.

## Using MEASure?

The simplest way to program the power meter for measurements is by using the `MEASure?` query. However, this command does not offer much flexibility. When you execute the command, the power meter selects the best settings for the requested configuration and immediately performs the measurement. You cannot change any settings (other than the expected power value, resolution and with the N1912A the measurement type) before the measurement is taken. This means you cannot fine tune the measurement, for example, you cannot change the filter length. To make more flexible and accurate measurements use the `CONFIGure` command. The measurement results are sent to the output buffer. `MEASure?` is a compound command which is equivalent to an `ABORT`, followed by a `CONFigure`, followed by a `READ?`.

### MEASure? Examples

The following commands show a few examples of how to use the `MEASure?` query to make a measurement. It is advisable to read through these examples in order as they become increasingly more detailed. These examples configure the power meter for a measurement (as described in each individual example), automatically place the power meter in the "wait-for-trigger" state, internally trigger the power meter to take one reading, and then sends the reading to the output buffer.

These examples give an overview of the `MEASure?` query. For further information on the `MEASure?` commands refer to the section "MEASure[1]|2|3|4 Commands" on page 125.

### Example 1 - The Simplest Method

The following commands show the simplest method of making single channel (for example A or B) measurements. Using `MEAS1?` results in an upper window measurement, and `MEAS2?` in a lower window measurement. The channel associated with the window can be set using the source list parameter (see Example 2 - Specifying the Source List Parameter), or defaults as in this example (See Agilent N1912A Only on page 13).

specifies window

```
MEAS1?
MEAS2?
```

### Example 2 - Specifying the Source List Parameter

The `MEASure` command has three optional parameters, an expected power value, a resolution and a source list. These parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter `DEFault` is used as a place holder.

The following example uses the source list parameter to specify the measurement channel as channel A. The expected power and resolution parameters are defaulted, leaving them at their current settings. The measurement is carried out on the upper window.

```
          specifies window          specifies channel
                       |                      |
                       v                      v
                MEAS1? DEF,DEF,(@1)
```

The operation of the `MEAS1?` command when the source list parameter is defaulted is described in the note Agilent N1912A Only on page 13.

**NOTE** For the N1911A it is not necessary to specify a channel as only one channel is available.

### Example 3 - Specifying the Expected Power Parameter

The previous example details the three optional parameters which can be used with the `MEASure?` command. The first optional parameter is used to enter an expected power value. Entering this parameter is only relevant if you are using an E-series power sensor. The value entered determines which of the power sensor's two ranges is used for the measurement. If the current setting of the power sensor's range is no longer valid for the new measurement, specifying the expected power value decreases the time taken to obtain a result.

The following example uses the expected value parameter to specify a value of -50 dBm. This selects the power sensor's lower range (refer to "Range" on page 33 for details of the range breaks). The resolution parameter is defaulted, leaving it at its current setting. The source list parameter specifies a channel B measurement. The measurement is displayed on the lower window.

specifies expected power value
specifies window
specifies channel

```
MEAS2? -50,DEF,(@2)
```

### Example 4 - Specifying the Resolution Parameter

The previous examples detailed the use of the expected value and source list parameters. The resolution parameter is used to set the resolution of the specified window. This parameter does not affect the resolution of the data, however it does affect the auto averaging setting (refer to Figure 1-2 on page 32).

Since the filter length used for a channel with auto-averaging enabled is dependent on the window resolution setting, a conflict arises when a given channel is set up in both windows and the resolution settings are different. In this case, the higher resolution setting is used to determine the filter length.

The following example uses the resolution parameter to specify a resolution setting of 3. This setting represents 3 significant digits if the measurement suffix is W or %, and 0.01 dB if the suffix is dB or dBm. Refer to Chapter 2, "Measurement Commands," on page 79, for further details on the resolution parameter. The expected power and source list parameters are defaulted in the example. The expected power value remains unchanged at its current setting. The source list parameter defaults as described in the note Agilent N1912A Only on page 13. Note that as the source list parameter is the last specified parameter you do not have to specify DEF. The measurement is carried out on the upper window.

specifies window
specifies resolution setting

```
MEAS1? DEF,3
```

Example 5 - Making a Difference Measurement

The following command is performed on the N1912A. It queries the lower window to make a difference measurement of channel B - channel A. The expected power and resolution parameters are defaulted, leaving them at their current settings.

specifies window

specifies between which channels the difference is calculated

```
MEAS2:POW:AC:DIFF? DEF,DEF,(@2),(@1)
```

Channel B - A

### Example 6 - Making a Ratio Measurement

The following command is performed on the N1912A. It queries the upper window to make a ratio measurement of channel A/B. The expected power and resolution parameters are defaulted, leaving them at their current settings.

specifies window

specifies the relationship of the channels in the ratio

```
MEAS1:POW:AC:RAT? DEF,DEF,(@1),(@2)
```

Channel A / B

**NOTE**

**Agilent N1912A Only**

The operation of the `MEASure?` command when the source list parameter is defaulted depends on the current setup of the window concerned (for example, A, B, A/B, A-B etc.) and on the particular command used (for example, `MEAS[:POW][:AC]?` and `MEAS:POW:AC:RAT?`).

This means that when the source list parameter is defaulted, there are a number of possibilities.

| Command | Current Window Setup | | Measurement |
|---|---|---|---|
| `MEAS1[:POW][AC]?` | Upper Window: | A | A |
| | | B | B |
| | Any Other | Any Other | A |
| `MEAS2[:POW][AC]?` | Lower Window: | A | A |
| | | B | B |
| | | Any Other | B |
| `MEAS1:POW:AC:RAT` | Upper Window: | A/B | A/B |
| | | B/A | B/A |
| | | Any Other | A/B |
| `MEAS2:POW:AC:RAT` | Lower Window: | A/B | A/B |
| | | B/A | B/A |
| | | Any Other | A/B |
| `MEAS1:POW:AC:DIFF?` | Upper Window: | A-B | A-B |
| | | B-A | B-A |
| | | Any Other | A-B |
| `MEAS2:POW:AC:DIFF?` | Lower Window: | A-B | A-B |
| | | B-A | B-A |
| | | Any Other | A-B |

## Using the CONFigure Command

When you execute this command, the power meter presets the optimum settings for the requested configuration (like the MEASure? query). However, the measurement is not automatically started and you can change measurement parameters before making measurements. This allows you to change the power meter's configuration from the preset conditions. The power meter offers a variety of low-level commands in the SENSe, CALCulate, and TRIGger subsystems. For example, if you want to change the averaging use the [SENSe[1]]|SENSe2:AVERage:COUNt command.

***Use the INITiate or READ? query to initiate the measurement.***

### Using READ?

CONFigure does not take the measurement. One method of obtaining a result is to use the READ? query. The READ? query takes the measurement using the parameters set by the CONFigure command then sends the reading to the output buffer. Using the READ? query obtains new data.

### Using INITiate and FETCh?

CONFigure does not take the measurement. One method of obtaining the result is to use the INITiate and FETCh? commands. The INITiate command causes the measurement to be taken. The FETCh? query retrieves a reading when the measurement is complete, and sends the reading to the output buffer. FETCh? can be used to display the measurement results in a number of different formats (for example, A/B and B/A) without taking fresh data for each measurement.

### CONFigure Examples

The following program segments show how to use the commands READ?, INITiate and FETCh? and CONFigure to make measurements.

It is advisable to read through these examples in order as they become increasingly more detailed.

These examples give an overview of the CONFigure command. For further information on the CONFigure commands refer to Chapter 2, “Measurement Commands,” on page 79.

### Example 1 - The Simplest Method

The following program segments show the simplest method of querying the upper and lower window’s measurement results respectively.

#### Using READ?

| | |
|---|---|
| `*RST` | *Reset instrument* |
| `CONF1` | *Configure upper window - defaults to a channel A measurement* |
| `READ1?` | *Take upper window (channel A) measurement* |

| | |
|---|---|
| `*RST` | *Reset instrument* |
| `CONF2` | *Configure lower window - defaults to a channel A (N1911A), Channel B (N1912A) measurement* |
| `READ2?` | *Take lower window measurement (channel A on N1911A, B on N1912A)* |

#### Using INITiate and FETCh?

| | |
|---|---|
| `*RST` | *Reset instrument* |
| `CONF1` | *Configure upper window - defaults to a channel A measurement* |
| `INIT1?` | *Causes channel A to make a measurement* |
| `FETC1?` | *Retrieves the upper window’s measurement* |

For the N1911A only:

| | |
|---|---|
| `*RST` | *Reset instrument* |
| `CONF2` | *Configure lower window - N1911A defaults to channel A* |
| `INIT1` | *Causes channel A to make a measurement* |
| `FETC2?` | *Retrieves the lower window’s measurement* |

For the N1912A only:

| | |
|---|---|
| `*RST` | *Reset instrument* |
| `CONF2` | *Configure lower window* |
| `INIT2?` | *Causes channel B to make a measurement* |
| `FETC2?` | *Retrieves the lower window's measurement* |

### Example 2 - Specifying the Source List Parameter

The `CONFigure` and `READ?` commands have three optional parameters, an expected power value, a resolution and a source list. These parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter `DEFault` is used as a place holder.

The following examples use the source list parameter to specify the measurement channel as channel A. The expected power and resolution parameters are defaulted, leaving them at their current settings. The measurement is carried out on the upper window.

Although the `READ?` and `FETCh?` queries have three optional parameters it is not necessary to define them as shown in these examples. If they are defined they must be identical to those defined in the `CONFigure` command otherwise an error occurs.

**NOTE** For the N1911A it is not necessary to specify a channel as only one channel is available.

### Using READ?

| | |
|---|---|
| `ABOR1` | *Aborts channel A* |
| `CONF1 DEF,DEF,(@1)` | *Configures the upper window to make a channel A measurement using the current expected power and resolution settings.* |
| `READ1?` | *Takes the upper window's measurement.* |

### Using INITiate and FETCh?

| | |
|---|---|
| `ABOR1` | *Aborts channel A* |
| `CONF1 DEF,DEF,(@1)` | *Configures the upper window to make a channel A measurement using the current expected power and resolution settings.* |
| `INIT1` | *Causes channel A to make a measurement.* |
| `FETC1? DEF,DEF,(@1)` | *Retrieves the upper window's measurement.* |

### Example 3 - Specifying the Expected Power Parameter

The previous example details the three optional parameters which can be used with the `CONFigure` and `READ?` commands. The first optional parameter is used to enter an expected power value. Entering this parameter is only relevant if you are using an E-series power sensor. The value entered determines which of the power sensor's two ranges is used for the measurement. If the current setting of the power sensor's range is no longer valid for the new measurement, specifying the expected power value decreases the time taken to obtain a result.

The following example uses the expected value parameter to specify a value of -50 dBm. This selects the power meter's lower range (refer to "Range" on page 33 for details of the range breaks). The resolution parameter is defaulted, leaving it at its current setting. The source list parameter specifies a channel B measurement. The measurement is carried out on the upper window.

### Using READ?

| | |
|---|---|
| `ABOR2` | *Aborts channel B* |
| `CONF1 -50,DEF,(@2)` | *Configures the upper window to make a channel B measurement using an expected power of -50 dBm and the current resolution setting.* |
| `READ1?` | *Takes the upper window's measurement.* |

Some fine tuning of measurements can be performed using the `CONFigure` and `READ?` commands. For example, in the above program segment some fine tuning can be performed by setting the filter length to 1024 and the trigger delay off.

1. `ABOR2`
2. `CONF1 -50,DEF,(@2)`
3. `SENS2:AVER:COUN 1024`
4. `TRIG2:DEL:AUTO OFF`
5. `READ1?`

### Using INITiate and FETCh?

| | |
|---|---|
| `ABOR2` | *Aborts channel B* |
| `CONF1 -50,DEF,(@2)` | *Configures the upper window to make a channel B measurement using an expected power of -50 dBm and the current resolution setting.* |
| `INIT2` | *Causes channel B to make a measurement.* |
| `FETC1? -50,DEF,(@2)` | *Retrieves the upper window's measurement.* |

Some fine tuning of measurements can be carried out using the `CONFigure` command and `INITiate` and `FETCh?` commands. For example, in the above program segment some fine tuning can be carried out by setting the filter length to 1024 and the trigger delay off.

1. `ABOR2`
2. `CONF1 -50,DEF,(@2)`

3. `SENS2:AVER:COUN 1024`
4. `TRIG2:DEL:AUTO OFF`
5. `INIT2`
6. `FETC1? -50,DEF,(@2)`

### Example 4 - Specifying the Resolution Parameter

The previous examples detailed the use of the expected value and source list parameters. The resolution parameter is used to set the resolution of the specified window. This parameter does not affect the resolution of the data, however it does affect the auto averaging setting (refer to Figure 1-2 on page 32).

Since the filter length used for a channel with auto-averaging enabled is dependent on the window resolution setting, a conflict arises when a given channel is set up in both windows and the resolution settings are different. In this case, the higher resolution setting is used to determine the filter length.

The following example uses the resolution parameter to specify a resolution setting of 3. This setting represents 3 significant digits if the measurement suffix is W or %, and 0.01 dB if the suffix is dB or dBm (for further details on the resolution parameter refer to the commands in Chapter 2, "Measurement Commands," on page 79). Also, in this example the expected power and source list parameters are defaulted. The expected power value is left unchanged at its current setting. The source list parameter is defaulted as described in the note Agilent N1912A Only on page 13. Note that as the source list parameter is the last specified parameter you do not have to specify `DEF`.

### Using READ?

| | |
|---|---|
| `ABOR1` | *Aborts channel A.* |
| `CONF1 DEF,3` | *Configures the upper window to make a measurement using the current setting of the expected power and source list and a resolution setting of 3.* |
| `READ1?` | *Takes the upper window's measurement. This is channel A or B measurement depending on current window setup* |

Some fine tuning of the above program segment can be carried out for example, by setting the trigger delay off. The following program segment assumes that channel A is currently being measured on the upper window.

1. `ABOR1`
2. `CONF1 DEF,3`
3. `TRIG1:DEL:AUTO OFF`
4. `READ1?`

**Using INITiate and FETCh?**

The following program segment assumes that channel A is currently being measured on the upper window.

| | |
|---|---|
| `ABOR1` | *Aborts channel A.* |
| `CONF1 DEF,3` | *Configures the upper window to make a measurement using the current setting of the expected power and source list and a resolution setting of 3.* |
| `INIT1` | *Causes channel A to make a measurement.* |
| `FETC1? DEF,3` | *Retrieves the upper window's measurement.* |

Some fine tuning of the above program segment can be carried out for example, by setting the trigger delay off.

1. `ABOR1`
2. `CONF1 DEF,3`
3. `TRIG1:DEL:AUTO OFF`
4. `INIT1:IMM`
5. `FETC1? DEF,3`

### Example 5 - Making a Difference Measurement

The following program segment can be carried out on the N1912A. It queries the lower window to make a difference measurement of channel A - channel B. The expected power level and resolution parameters are defaulted, leaving them at their current settings. Some fine tuning of the measurement is carried out by setting the averaging, and the trigger delay to off.

#### Using READ?

```
ABOR1
ABOR2
CONF2:POW:AC:DIFF DEF,DEF,(@1),(@2)
SENS1:AVER:COUN 1024
SENS2:AVER:COUN 1024
TRIG1:DEL:AUTO OFF
TRIG2:DEL:AUTO OFF
READ2:POW:AC:DIFF?
READ2:POW:AC:DIFF? DEF,DEF,(@2),(@1)
```

*(A second `READ?` query is sent to make a channel B - channel A measurement using fresh measurement data).*

#### Using INITiate and FETCh?

```
ABOR1
ABOR2
CONF2:POW:AC:DIFF DEF,DEF,(@1),(@2)
SENS1:AVER:COUN 1024
SENS2:AVER:COUN 1024
TRIG1:DEL:AUTO OFF
TRIG2:DEL:AUTO OFF
INIT1:IMM
INIT2:IMM
FETC2:POW:AC:DIFF?
FETC2:POW:AC:DIFF? DEF,DEF,(@2),(@1)
```

*(A second `FETCh?` query is sent to make a channel B - channel A measurement using the current measurement data).*

### Example 6 - Making a Ratio Measurement

The following program segment can be carried out on the N1912A. It queries the lower window to make a ratio measurement of channel A/B. The expected power level and resolution parameters are defaulted, leaving them at their current settings. Some fine tuning of the measurement is carried out by setting the averaging.

#### Using READ?

```
ABOR1
ABOR2
CONF2:POW:AC:RAT DEF,DEF,(@1),(@2)
SENS1:AVER:COUN 512
SENS2:AVER:COUN 256
READ2:POW:AC:RAT?
```

`READ2:POW:AC:RAT? DEF,DEF,(@2),(@1)` *(A second `READ?` query is sent to make a channel B - channel A ratio measurement using fresh measurement data.)*

#### Using INITiate and FETCh?

```
ABOR1
ABOR2
CONF2:POW:AC:RAT DEF,DEF,(@1),(@2)
SENS1:AVER:COUN 512
SENS2:AVER:COUN 256
INIT1:IMM
INIT2:IMM
FETC2:POW:AC:RAT?
```

`FETC2:POW:AC:RAT? DEF,DEF,(@2),(@1)` *(A second `FETCh?` query is sent to make a channel B - channel A measurement using the current measurement data.)*

## Using the Lower Level Commands

An alternative method of making measurements is to use the lower level commands to set up the expected range and resolution. This can be done using the following commands:

```
[SENSe[1]]|SENSe2:POWER:AC:RANGe
DISPlay[:WINDow[1|2]]:RESolution
```

The measurement type can be set using the following commands in the `CALCulate` subsystem:

```
CALCulate[1|2]:MATH[:EXPRession]
CALCulate[1|2]:RELative[:MAGNitude]
```

The advantage of using the lower level commands over the `CONFigure` command is that they give you more precise control of the power meter. As shown in Table 1-1 the `CONFigure` command presets various states in the power meter. It may be likely that you do not want to preset these states.

### Example

The following example sets the expected power value to -50 dBm and the resolution setting to 3 using the lower level commands. The measurement is a single channel A measurement carried out on the lower window.

| | |
|---|---|
| `ABOR1` | *Aborts channel A.* |
| `CALC2:MATH:EXPR "(SENS1)"` | *Displays channel A on lower window.* |
| `SENS1:POW:AC:RANG -50` | *Sets lower range (E-series sensors only).* |
| `DISP:WIND2:RES 3` | *Sets the lower window's resolution to setting 3.* |
| `INIT1` | *Causes channel A to make a measurement.* |
| `FETC2?` | *Retrieves the lower window's measurement.* |

# Using Frequency Dependent Offset Tables

This section describes how to use frequency dependent offset tables. These tables give you the ability to compensate for frequency effects in your test setup.

## Overview

If the `[SENSe[1]]|SENSe2:CORRection:CSET2:STATe` command is `OFF`, the frequency dependent offset tables are not used. When `[SENSe[1]]|SENSe2:CORRection:CSET2:STATe` is `ON`, the frequency dependent offset tables are used, providing you with a quick and convenient method of compensating for your external test setup over a range of frequencies. Note that when selected, frequency dependent offset correction is IN ADDITION to any correction applied for sensor frequency response. The power meter is capable of storing 10 frequency dependent offset tables of 80 frequency points each.

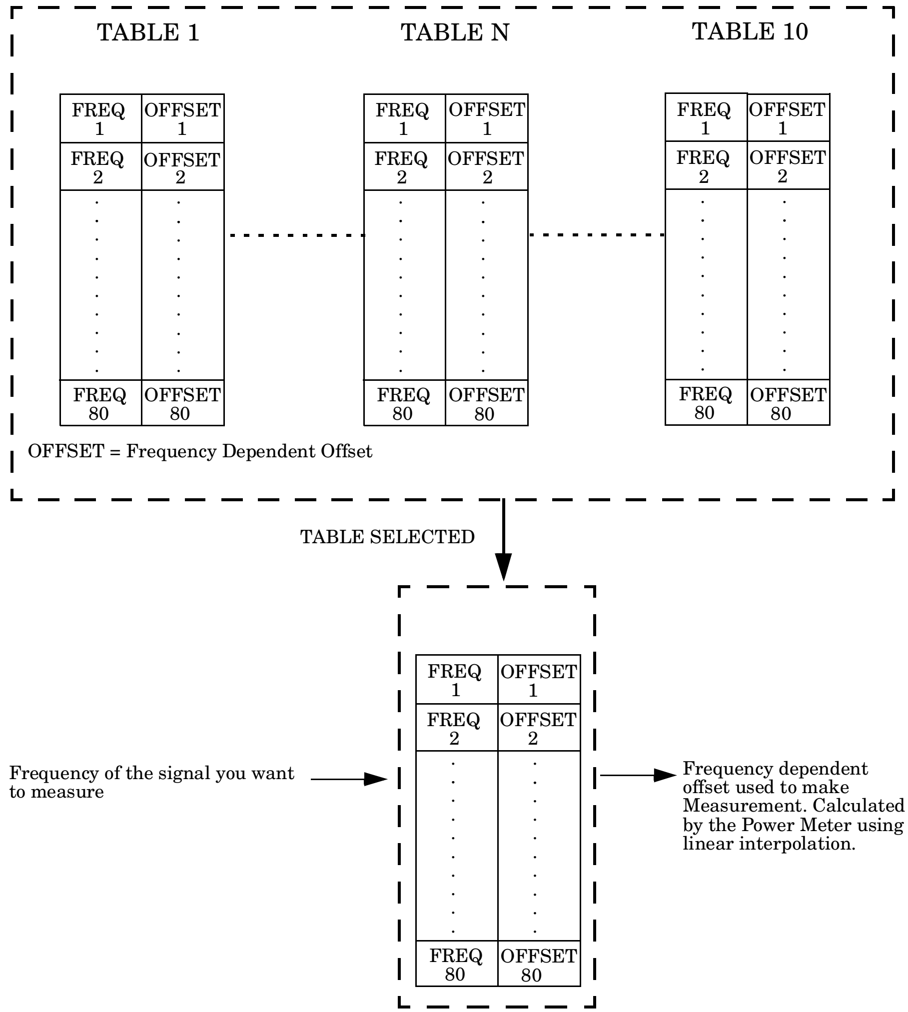
To use frequency dependent offset tables you:

1. Edit a frequency dependent offset table if necessary.
2. Select the frequency dependent offset table.
3. Enable the frequency dependent offset table.
4. Zero and calibrate the power meter.

   If you are using an 8480 series sensor the reference calibration factor used during the calibration must be entered manually.
5. Specify the frequency of the signal you want to measure. The required offset is automatically set by the power meter from the frequency dependent offset table.
6. Make the measurement.

Figure 1-1 illustrates how frequency dependent offset tables operate.

**Figure 1-1** **Frequency Dependent Offset Tables**



## Editing Frequency Dependent Offset Tables

It is not possible to create any additional frequency dependent offset tables. However, the 10 existing tables can be edited using the MEMory subsystem. To do this:

1. Select one of the existing tables using:
   MEMory:TABle:SELect <string>.
   For information on naming frequency dependent offset tables see “Naming Frequency Dependent Offset Tables” on page 28. For information on the current names which you can select refer to “Listing the Frequency Dependent Offset Table Names” on page 27.
2. Enter the frequency data using:
   MEMory:TABle:FREQuency <numeric_value> {,<numeric_value>}
3. Enter the offset factors as shown in the table below using:
   MEMory:TABle:GAIN <numeric_value> {,<numeric_value>}

| Frequency | Offset |
|---|---|
| Frequency 1 | Offset 1 |
| Frequency 2 | Offset 2 |
| " | " |
| Frequency n | Offset n |

4. If required, rename the frequency dependent offset table using:
   MEMory:TABLe:MOVE <string>,<string>. The first <string> parameter identifies the existing table name, and the second identifies the new table name.

**NOTE**

The legal frequency suffix multipliers are any of the IEEE suffix multipliers, for example, KHZ, MHZ and GHZ. If no units are specified the power meter assumes the data is Hz.

PCT is the only legal unit for offset factors and can be omitted.

The frequency and offset data must be within range. Refer to the individual commands in Chapter 4 for their specified ranges.

Any offset values entered into the table should exclude the effect of the sensor. Characterization of the test setup independently of the sensor allows the same table to be used with any sensor.

Ensure that the frequency points you use cover the frequency range of the signals you want to measure. If you measure a signal with a frequency outside the frequency range defined in the frequency dependent offset table, then the power meter uses the highest or lowest frequency point in the table to calculate the offset. To make subsequent editing of a frequency dependent offset table simpler, it is recommended that you retain a copy of your data in a program.

### Listing the Frequency Dependent Offset Table Names

To list the frequency dependent offset tables currently stored in the power meter, use the following command:
`MEMory:CATalog:TABLe?`

Note that all tables are listed; including sensor calibration tables.

The power meter returns the data in the form of two numeric parameters and a string list representing all stored tables.

- `<numeric_value>,<numeric_value>{,<string>}`
  The first numeric parameter indicates the amount of memory, in bytes, used for storage of tables. The second parameter indicates the memory, in bytes, available for tables.

Each string parameter returned indicates the name, type and size of a stored frequency dependent offset table:

- `<string>,<type>,<size>`
  The `<string>`, `<type>` and `<size>` are all character data. The `<type>` is always `TABL`. The `<size>` is displayed in bytes.

For example, a sample of the response may look like:

```
560,8020,“Offset_1,TABL,220”,”Offset_2,TABL,340” ....
```

### Naming Frequency Dependent Offset Tables

To rename a frequency dependent offset table use:
`MEMory:TABLe:MOVE <string>,<string>`

The first `<string>` parameter identifies the existing table name, and the second identifies the new table name.

The following rules apply to frequency dependent offset table names:

1. Table names use a maximum of 12 characters.
2. All characters must be upper or lower case alphabetic characters, or numeric (0-9), or an underscore (_).
3. No spaces are allowed in the name.

### Reviewing Table Data

To review the data stored in a frequency dependent offset table, use the following commands:

`MEMory:TABLe:SELect "Offset1"`
*Select the sensor calibration table named “Offset1”.*
`MEMory:TABLe:SELect?`
*Query command which returns the name of the currently selected table.*
`MEMory:TABLe:FREQuency:POINTs?`
*Query command which returns the number of stored frequency points.*
`MEMory:TABLe:FREQuency?`
*Query command which returns the frequencies stored in the frequency dependent offset table (in Hz).*
`MEMory:TABLe:GAIN[:MAGNitude]:POINTs?`
*Query command which returns the number of offset factor points stored in the frequency dependent offset table.*
`MEMory:TABLe:GAIN[:MAGNitude]?`
*Query command which returns the offset factors stored in the frequency dependent offset table.*

### Modifying Data

If you need to modify the frequency and offset factor data stored in a frequency dependent offset table you need to resend the complete data lists.

If you have retained the original data in a program, edit the program and resend the data.

## Selecting a Frequency Dependent Offset Table

After you have created the frequency dependent offset table, you can select it using the following command:

`[SENSe[1]]|SENSe2:CORRection:CSET2[:SELect] <string>`

To find out which frequency dependent offset table is currently selected, use the query:

`[SENSe[1]]|SENSe2:CORRection:CSET2[:SELect]?`

## Enabling A Frequency Dependent Offset Table

To enable the frequency dependent offset table, use the following command:

`[SENSe[1]]|SENSe2:CORRection:CSET2:STATe ON`

If you set `[SENSe[1]]|SENSe2:CORRection:CSET2:STATe` to `ON` and no frequency dependent offset table is selected error -221, “Settings conflict” occurs.

## Making The Measurement

To make the power measurement, set the power meter for the frequency of the signal you want to measure. The power meter automatically sets the calibration factor. Use either the `INITiate`, `FETCh?` or the `READ?` query to initiate the measurement as shown in the following program segments:

### INITiate Example

```
ABORt1
CONFigure1:POWer:AC DEF,1,(@1)
SENS1:CORR:CSET2:SEL "Offset1"
SENS1:CORR:CSET2:STAT ON
SENSe1:FREQuency 500KHZ
INITiate1:IMMediate
FETCh1?
```

### READ? Example

```
ABORt1
CONFigure1:POWer:AC DEF,2,(@1)
SENS1:CORR:CSET2:SEL "Offset1"
SENS1:CORR:CSET2:STAT ON
SENSe1:FREQuency 500KHZ
READ1?
```

**NOTE**

If the measurement frequency does not correspond directly to a frequency in the frequency dependent offset table, the power meter calculates the offset using linear interpolation.

If you enter a frequency outside the frequency range defined in the frequency dependent offset table, then the power meter uses the highest or lowest frequency point in the table to set the offset.

To find out the value of the offset being used by the power meter to make a measurement, use the query command:
`SENSe:CORRection:GAIN4|FDOFfset[:INPut][MAGNITUDE]?`
The response may be an interpolated value.

# Setting the Range, Resolution and Averaging

This section provides an overview of setting the range, resolution and averaging. For more detailed information about these features refer to the individual commands in Chapter 9.

## Resolution

You can set the window's resolution using the following command:

```
DISPlay[:WINDow[1]|2][:NUMeric[1]|2]
:RESolution <numeric_value>
```

There are four levels of resolution available (1 through 4).

When the measurement suffix is W or % this parameter represents the number of significant digits. When the measurement suffix is dB or dBM, 1 through 4 represents 1, 0.1, 0.01, and 0.001 dB respectively.

Refer to the `:RESolution` command on page 207 for further information.

## Averaging

The power meter has a digital filter to average power readings. The number of readings averaged can range from 1 to 1024. This filter is used to reduce noise, obtain the desired resolution and to reduce the jitter in the measurement results. However, the time to take the measurement is increased. You can select the filter length or you can set the power meter to auto filter mode. To enable and disable averaging use the following command:

```
[SENSe[1]]|SENSe2:AVERage[:STATe] <boolean>
```

### Auto Averaging Mode

To enable and disable auto filter mode, use the following command:

```
[SENSe[1]]|SENSe2:AVERage:COUNt:AUTO <boolean>
```

When the auto filter mode is enabled, the power meter automatically sets the number of readings averaged together to satisfy the filtering requirements for most power measurements. The number of readings averaged together depends on the resolution and the power level currently being measured. Figure 1-2 lists the number of readings averaged for each range and resolution when the power meter is in auto filter mode.

**NOTE** Figure 1-2 applies to 8480 series sensors only.

**Figure 1-2** **Typical Averaged Readings on 8480 Series Sensors**

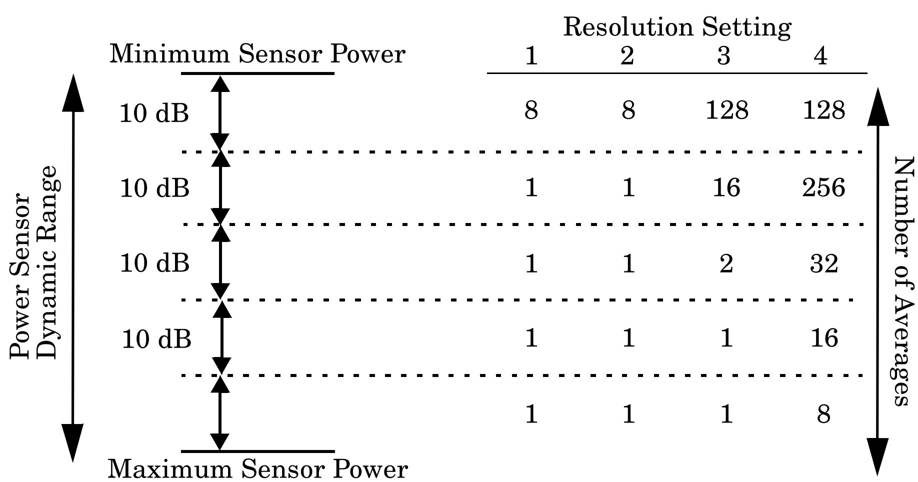
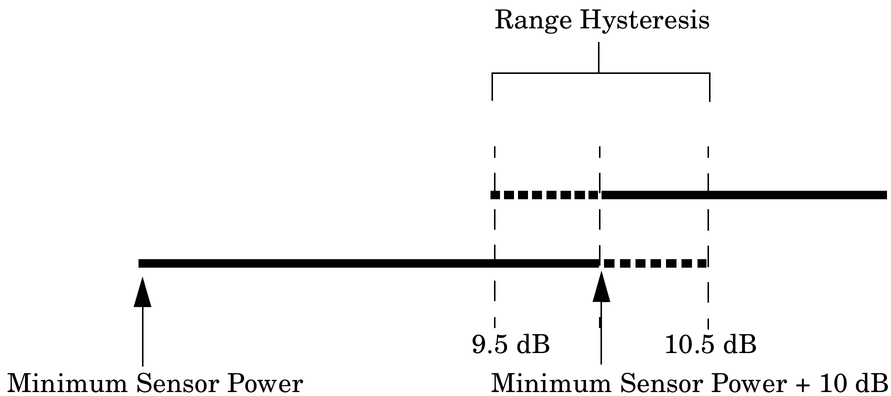



Figure 1-3 illustrates part of the power sensor dynamic range hysteresis.

**Figure 1-3** **Averaging Range Hysteresis**




## Filter Length

You specify the filter length using the following command:

```
[SENSe[1]]|SENSe2:AVERage:COUNt <numeric_value>
```

The range of values for the filter length is 1 to 1024. Specifying this command disables automatic filter length selection. Increasing the value of the filter length reduces measurement noise. However, the time to take the measurement is increased.

## Range

The power meter has no internal ranges which can be set. The only ranges that can be set are those of the E-series power sensor. With an E-series power sensor the range can be set either automatically or manually. Use autoranging when you are not sure of the power level you will be measuring.

### Setting the Range

To set the range manually use the following command:
`[SENSe[1]]|SENSe2:POWer:AC:RANGe <numeric_value>`

If the `<numeric_value>` is set to:

- `0`, the sensor's lower range is selected. (For example, this range is -70 to -13.5 dBm for the E4412A power sensor.)
- `1`, the sensor's upper range is selected. (For example, this range is -14.5 to +20 dBm for the E4412A power sensor.)

For details on the range limits of other E-series power sensor refer to the appropriate power sensor manual.

For further information on this command refer to page 303.

To enable autoranging use the following command:
`[SENSe[1]]|SENSe2:POWer:AC:RANGe:AUTO ON`

Use autoranging when you are not sure of the power level you will be measuring.

# Setting Offsets

## Channel Offsets

The power meter can be configured to compensate for signal loss or gain in your test setup (for example, to compensate for the loss of a 10 dB attenuator). You use the `SENSe` command subsystem to configure the power meter. Gain and loss correction are a coupled system. This means that a gain set by `[SENSe[1]]|SENSe2:CORRection:GAIN2` is represented in the `[SENSe[1]]|SENSe2:CORRection:LOSS2?` command. If you enter an offset value the state is automatically enabled. However it can be enabled and disabled using either the `[SENSe[1]]|SENSe2:CORRection:GAIN2:STATe` or `[SENSe[1]]|SENSe2:CORRection:LOSS2:STATe` commands.

`LOSS2` is coupled to `GAIN2` by the equation $Loss = \frac{1}{Gain}$ when the default unit is linear, and $Gain = -Loss$ when the default is logarithmic.

**NOTE** You can only use `LOSS2` and `GAIN2` for external losses and gains. `LOSS1` and `GAIN1` are specifically for calibration factors.

## Display Offsets

Display offset values can be entered using the `CALCulate[1|2]:GAIN[:MAGNitude]` command. `CALCulate[1|2]:GAIN:STATe` must be set to `ON` to enable the offset value. If you enter an offset value the state is automatically enabled. This offset is applied after any math calculations (refer to Figure 1-6 on page 45).

### Example

The following example program, in HP Basic, details how to use the channel and display offsets on an N1912A making a channel A/B ratio measurement. The final result is:

$$\left(\left(\frac{A_{dBm}-10}{B_{dBm}-10}\right)-20\right)_{dB}$$

```
10 !Create I/O path name
20 ASSIGN @POWER TO 713
30 !Clear the power meter’s interface
40 CLEAR @POWER
50 !Set the power meter to a known state
60 OUTPUT @POWER;"*RST"
70 !Configure the Power Meter to make the measurement
80 OUTPUT @Power;"CONF:POW:AC:RAT 20DBM,2,(@1),(@2)"
90 !Set the measurement units to dBm
100 OUTPUT @POWER;"UNIT:POW DBM"
110 !Set the power meter for channel offsets of -10 dB
120 OUTPUT @POWER;"SENS1:CORR:GAIN2 -10"
130 OUTPUT @POWER;"SENS2:CORR:GAIN2 -10"
140 !Enable the gain correction
150 OUTPUT @POWER;"SENS:CORR:GAIN2:STATe ON"
160 OUTPUT @POWER;"SENS2:CORR:GAIN2:STATe ON"
170 !Set the power meter for a display offset of -20 dB
180 OUTPUT @POWER;"CALC1:GAIN -20 DB"
190 PRINT "MAKING THE MEASUREMENT"
200 !Initiate the measurement
210 OUTPUT @Power;"INIT1:IMM"
220 OUTPUT @Power;"INIT2:IMM"
230 ! ... and get the result
240 OUTPUT @Power;"FETC:POW:AC:RAT? 20DBM,2,(@1),(@2)"
250 ENTER @Power;Reading
260 !
270 PRINT "The measurement result is ";Reading;"dB."
280 END
```

For further information on channel offsets refer to page 292. For further information on display offsets refer to page 145.

# Setting Measurement Limits

You can configure the power meter to detect when a measurement is outside of a predefined upper and/or lower limit value.

Limits are window or measurement display line based and can be applied to power, ratio or difference measurements.

## Setting Limits

The power meter can be configured to verify the power being measured against an upper and/or lower limit value. The range of values that can be set for lower and upper limits is -150.00 dBm to +230.00 dBm. The default upper limit is +90.00 dBm and the default lower limit is -90.00 dBm.

A typical application for this feature is shown in Figure 1-4.

**Figure 1-4** **Limits Checking Application**

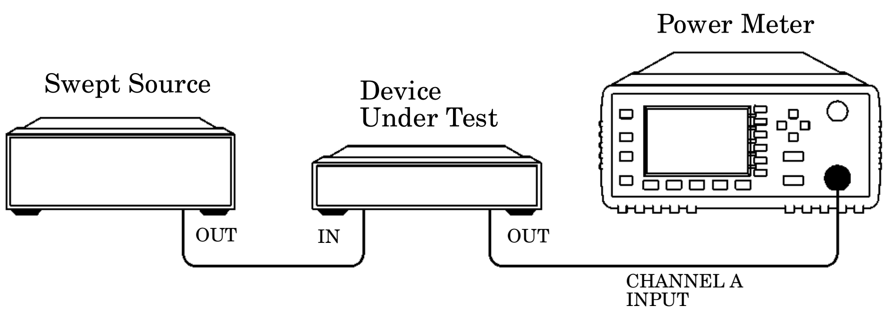
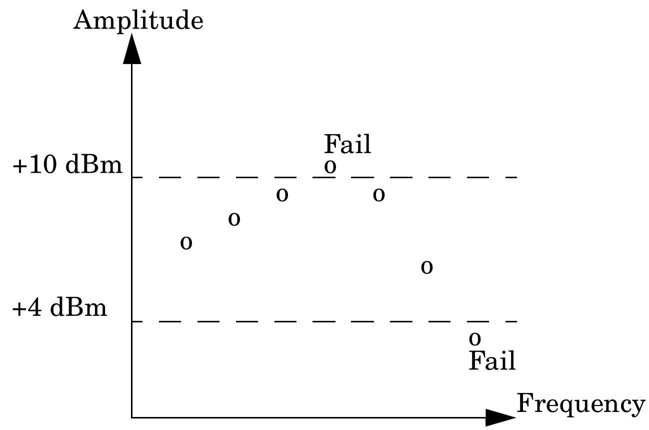


**Figure 1-5** **Limits Checking Results**




## Setting Limits (Check Header - same as previous)

The power meter can be configured to verify the current measurement in any measurement line against predefined upper and/or lower limit values. The range of values that can be set for the upper and lower limits and the default values depends on the measurement units in the currently measurement line - see Table 1-2.

**Table 1-2** **Range of Values for Window Limits**

| Window Units | | | Default | |
|---|---|---|---|---|
| | Maximum | Minimum | Maximum | Minimum |
| dB | +200 dB | -180 dB | 60 dB | -120 dB |
| dBm | +230 dBm | -150 dBm | 90 dBm | -90 dBm |
| % | 999.9 X% | 100.0 a% | 100.0 M% | 100.0 p% |
| W | 100.000 XW | 1.000 aW | 1.000 MW | 1.000 pW |

## Checking for Limit Failures

There are two ways to check for limit failures:

1. Use the `SENSe:LIMit:FAIL? and SENSe:LIMit:FCOunt?` commands for channel limits or the `CALCulate[1|2]:LIMit:FAIL?` and the `CALCulate[1|2]:LIMit:FCOunt?` for window limits.
2. Use the `STATus` command subsystem.

### Using SENSe and CALCulate

Using `SENSe` to check the channel limit failures in Figure 1-5 would return the following results:

| | |
|---|---|
| `SENSe:LIMit:FAIL?` | *Returns 1 if there has been 1 or more limit failures or 0 if there have been no limit failures. In this case 1 is returned.* |
| `SENSe:LIMit:FCOunt?` | *Returns the total number of limit failures, in this case 2.* |

Use the equivalent `CALCulate` commands for checking window limit failures.

**NOTE** If `TRIGger:DELay:AUTO` is set to `ON`, then the number of failures returned by `SENSe:LIMit:FCOunt?` or `CALCulate[1|2]:LIMit:FCOunt?` is affected by the current filter settings.

## Using STATus

If using GPIB, you can use the STATus subsystem to generate an SRQ to interrupt your program when a limit failure occurs. This is a more efficient method than using SENSe or CALCulate, since you do not need to check the limit failures after every power measurement.

Refer to “Status Reporting” on page 46 and “STATus Subsystem” on page 324 for further information.

# Getting the Best Speed Performance

This section discusses the factors that influence the speed of operation (number of readings/sec) of a P-Series power meter.

The following factors are those which have the greatest effect upon measurement speed (in no particular order):

- The selected measurement rate, i.e. `NORMal`, `DOUBle`, `FAST`.
- The sensor being used.
- The trigger mode (for example, free run, trigger with delay etc.).
- The output format: `ASCii` or `REAL`.
- The units used for the measurement.
- The command used to take a measurement.

In addition, in `FAST` mode there are other influences which are described in “Fast Mode” on page 44.

The following paragraphs give a brief description of the above factors and how they are controlled from SCPI.

## Measurement Rate

There are three possible speed settings `NORMal`, `DOUBle` and `FAST`. These are set using the `SENSe:MRATe` command and can be applied to each channel independently (N1912A only).

In `NORMal` and `DOUBle` modes, full instrument functionality is available and these settings can be used with all sensors. `FAST` mode is only available for the P- series and E-series sensors. Also, in `FAST` mode averaging, limits and ratio/difference math functions are disabled.

Refer to “Specifications” in the *P-Series Power Meters User’s Guide* to see the influence of these speed settings on the accuracy and noise performance of the power meter.

## Sensor

Different measurement rates are achievable depending on the sensor type being used, as shown in Table 1-3:

**Table 1-3** **Model of Sensor and Measurement Rates**

| Sensor | Measurement Rate | | |
|---|---|---|---|
| | **NORMal** | **DOUBle** | **FAST** |
| 8480 series | 20 reading/s | 40 reading/s | NA |
| E-series E4410 and E9300 | 50 ms<br>20 reading/s | 25 ms<br>40 reading/s | Up to 400 |
| E-series E9320, AVERage only mode | 50 ms<br>20 reading/s | 25 ms<br>40 reading/s | Up to 400 |
| E-series E9320, NORMal mode | 50 ms<br>20 reading/s | 25 ms<br>40 reading/s | Up to 1000 |
| P-series | 50 ms<br>20 reading/s | 25 ms<br>40 reading/s | Up to 1500 |

## Trigger Mode

The power meter has a very flexible triggering system. For simplicity, it can be described as having three modes:

- Free Run: When a channel is in Free Run, it continuously takes measurements on this channel. A channel is in free run when `INITiate:CONTinuous` is set to `ON` and `TRIGger:SOURce` is set to `IMMediate`.
- Triggered Free Run: When a channel is in Triggered Free Run Continuous Trigger, it takes a new measurement each time a trigger even is detected. A channel is in Triggered Free Run Continuous Trigger when `INITiate:CONTinuous` is set to `ON` and `TRIGger:SOURce` is not set to `IMMediate`.

- Single Shot: When a channel is in Single Shot, it takes a new measurement when a trigger event is detected and then returns to the idle state. A channel is in Single Shot when `INITiate:CONTinuous` is set to `OFF`. Note that a measurement can take several INT/EXT triggers depending on the filter settings. Refer to “TRIGger[1]|2:DELay:AUTO <boolean>” on page 463 for further information.

**NOTE** A trigger event can be any of the following:

- The input signal meeting the trigger level criteria.
- Auto-level triggering being used.
- A `TRIGger GET` or `*TRG` command being sent.
- An external TTL level trigger being detected.

### Trigger with delay

This can be achieved using the same sequences above (apart from the second) with `TRIG:DEL:AUTO` set to `ON`. Also, the `MEAS?` command operates in trigger with delay mode.

In trigger with delay mode, a measurement is not completed until the power meter filter is full. In this way, the reading returned is guaranteed to be settled. In all other modes, the result returned is simply the current result from the filter and may or may not be settled. This depends on the current length of the filter and the number of readings that have been taken since a change in power level.

With trigger with delay enabled, the measurement speed can be calculated roughly using the following equation:

readings/sec = speed (as set by `SENSe:SPEed`) / filter length

For example, with a filter length of 4 and `SENS:SPE` set to `20`, approximately 5 readings/sec is calculated by the power meter.

Typically, free run mode provides the best speed performance from the power meter (especially in 200 readings/sec mode).

## Output Format

The power meter has two output formats for measurement results: `ASCii` and `REAL`. These formats are selected using the `FORMat` command. When `FORMat` is set to `REAL`, the returned result is in IEEE 754 floating-point format (note that the byte order can be changed using `FORMat:BORDer`) plus <LF> as an end sentinel of the block.

The `REAL` format is likely to be required only for `FAST` mode as it reduces the amount of bus traffic.

## Units

The power meter can output results in either linear or log units. The internal units are linear, therefore optimal performance is achieved when the results output are also in linear units (since the overhead of performing a log function is removed).

## Command Used

In Free Run mode, `FETCh?` must be used to return a result.

In other trigger modes, there are a number of commands which can be used, for example, `MEASure?`, `READ?`, `FETCh?` Note that the `MEAS?` and `READ?` commands are compound commands—they perform a combination of other lower level commands. Typically, the best speed performance is achieved using the low level commands directly.

### Trigger Count

To get the fastest measurement speed the a `TRIG:COUNT` must be set to return multiple measurements for each `FETCh` command. For average only measurements a count of 4 is required, however, 10 is recommended. In normal mode (peak measurements) a count of 50 is required to attain 1000 readings per second.

## Fast Mode

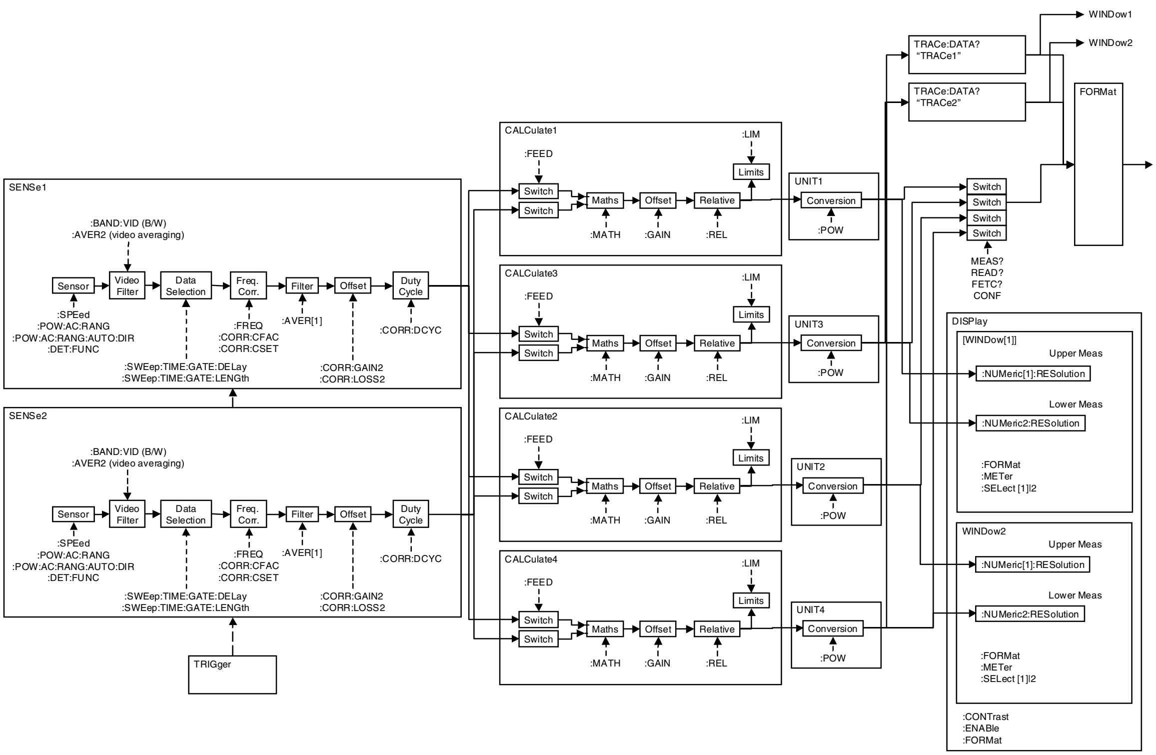
In the highest speed setting, the limiting factor tends to be the speed of the controller being used to retrieve results from the power meter, and to a certain extent, the volume of remote traffic. The latter can be reduced using the `FORMat REAL` command to return results in binary format. The former is a combination of two factors:

- the hardware platform being used.
- the programming environment being used.

# How Measurements are Calculated

Figure 1-6 details how measurements are calculated. It shows the order in which the various power meter functions are implemented in the measurement calculation.

**Figure 1-6 How Measurements are Calculated**




The `MEASure` commands in this figure can be replaced with the `FETCh?` and `READ?` commands.

**NOTE**

All references to channel B in the above diagram refer to the N1912A only.

# Status Reporting

Status reporting is used to monitor the power meter to determine when events have occurred. Status reporting is accomplished by configuring and reading status registers.

The power meter has the following main registers:

- Status Register
- Standard Event Register
- Operation Status Register
- Questionable Status Register
- Device Status Register

There are other registers that exist “behind” the main registers, and are described later in this chapter.

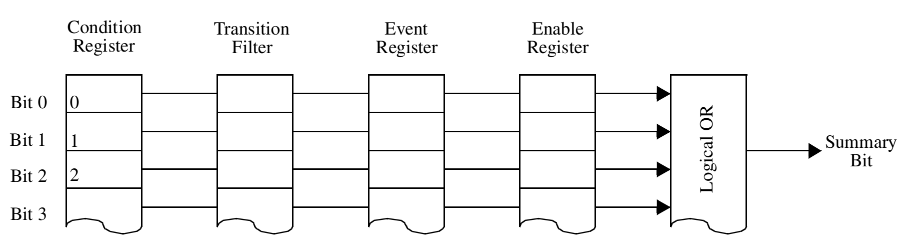
Status and Standard Event registers are read using the IEEE-488.2 common commands.

Operation and Questionable Status registers are read using the SCPI STATus command subsystem.

## The General Status Register Model

The generalized status register model shown in Figure 1-7 is the building block of the SCPI status system. This model consists of a condition register, a transition filter, an event register and an enable register. A set of these registers is called a status group.

**Figure 1-7** **Generalized Status Register Model**



When a status group is implemented in an instrument, it always contains all of the component registers. However, there is not always a corresponding command to read or write to every register.

### Condition Register

The condition register continuously monitors the hardware and firmware status of the power meter. There is no latching or buffering for this register, it is updated in real time. Condition registers are read-only.

### Transition Filter

The transition filter specifies which types of bit state changes in the condition registers and set corresponding bits in the event register. Transition filter bits may be set for positive transitions (PTR), negative transitions (NTR), or both. Transition filters are read-write. They are unaffected by `*CLS` or queries. After `STATus:PRESet` the NTR register is set to `0` and all bits of the PTR are set to `1`.

### Event Register

The event register latches transition events from the condition register as specified by the transition filter. Bits in the event register are latched and on setting they remain set until cleared by a query or a `*CLS`. Also on setting, an event bit is no longer affected by condition changes. It remains set until the event register is cleared; either when you read the register or when you send the `*CLS` (clear status) command. Event registers are read-only.

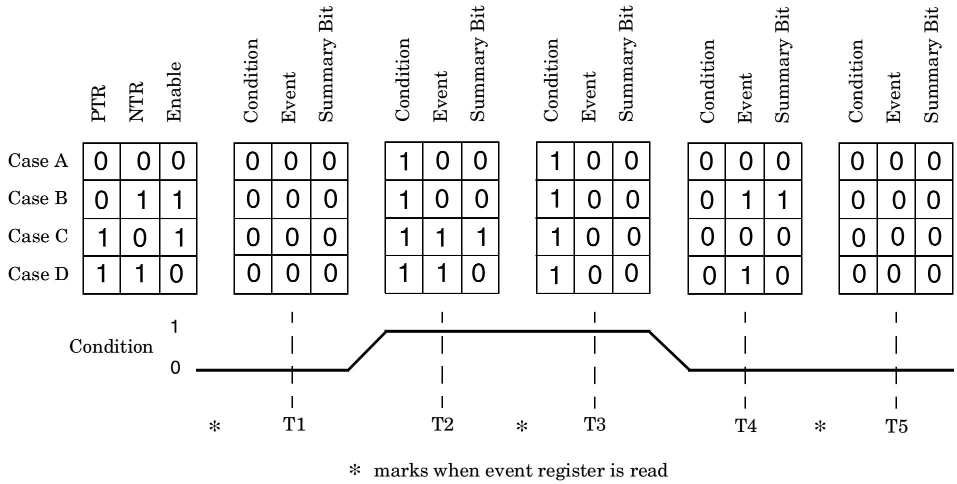
### Enable Register

The enable register specifies the bits in the event register that can generate a summary bit. The instrument logically ANDs corresponding bits in the event and enable registers and ORs all the resulting bits to obtain a summary bit. Enable registers are read-write. Querying an enable register does not affect it.

### An Example Sequence

Figure 1-8 illustrates the response of a single bit position in a typical status group for various settings. The changing state of the condition in question is shown at the bottom of the figure. A small binary table shows the state of the chosen bit in each status register at the selected times T1 to T5.

**Figure 1-8** **Typical Status Register Bit Changes**




## How to Use Registers

There are two methods to access the information in status groups:

- the polling method, or
- the service request (SRQ) method. (GPIB mode only)

Use the polling method when:

- your language/development environment does not support SRQ interrupts.
- you want to write a simple, single purpose program and do not want to add the complexity of setting an SRQ handler.

Use the SRQ method when you:

- need time critical notification of changes.
- are monitoring more than one device which supports SRQ interrupts.
- need to have the controller do something else while it is waiting.
- cannot afford the performance penalty inherent to polling.

## The Condition Polling Method

In this polling method, the power meter has a passive role. It only informs the controller that conditions have changed when the controller asks. When you monitor a condition with the polling method, you must:

1. Determine which register contains the bit that monitors the condition.
2. Send the unique query that reads that register.
3. Examine the bit to see if the condition has changed.

The polling method works well if you do not need to know about the changes the moment they occur. The SRQ method is more effective if you must know immediately when a condition changes. Detecting an immediate change in a condition using the polling method requires your program to continuously read the registers at very short intervals. This is not particularly efficient and there is a possibility that an event may be missed.

## The SRQ Method

When a bit of the Status Register is set and has been enabled to assert SRQ (*SRE command), the power meter sets the GPIB SRQ line true. This interrupt can be used to interrupt your program, suspending its current operation, and find out what service the power meter requires. Refer to your computer and language manuals for information on how to program the computer to respond to the interrupt.

To allow any of the Status Register bits to set the SRQ line true, you must enable the appropriate bit(s) with the *SRE command. For example, if your application requires an interrupt whenever a message is available in the output queue (Status Register bit 4, decimal weight 16). To enable bit 4 to assert SRQ, use the command *SRE 16

**NOTE** You can determine which bits are enabled in the Status Register using *SRE?. This command returns the decimal weighted sum of all the bits.

### Procedure

- Send a bus device clear message.
- Clear the event registers with the *CLS (clear status) command.
- Set the *ESE (standard event register) and *SRE (status byte register) enable masks.
- Enable your bus controller's IEEE-488 SRQ interrupt.

### Examples

The following two examples are written in HP BASIC and illustrate possible uses for SRQ. In both cases, it is assumed that the power meter has been zeroed and calibrated.

### Example 1:

```
10   ! Program to generate an SRQ when a channel A sensor
20   ! connect or disconnect occurs
30   !
40    ASSIGN @Pm TO 713 ! Power meter GPIB address
50   ON ON INTR 7 GOTO Srq_i! Define service request handler
60    CLEAR @Pm              ! Selective device clear
70   OUTPUT @Pm;"*CLS;*RST" ! Clear registers and resetmeter
80   !
90   ! Configure the device status register so that a sensor
100   ! connect or disconnect on channel A will cause an SRQ.
110   !
120   OUTPUT @Pm;"STAT:DEV:ENAB 2"
130   OUTPUT @Pm;"STAT:DEV:NTR 2"
140   OUTPUT @Pm;"STAT:DEV:PTR 2"
150   OUTPUT @Pm;"*SRE 2"
160   !
170   ENABLE INTR 7;2  ! Enable an SRQ to cause an interrupt
180   LOOP             ! Idle loop
190   ! Forever
200   END LOOP
210   !
220   ! When a SRQ is detected, the following routine will
service it.
230   !
240 Srq_i:     !
250   St=SPOLL(@Pm)  ! Serial Poll (reads status byte)
```

```
260   IF BIT(St,1)=1 THEN    ! Device status reg bit set ?
270     OUTPUT @Pm;"STAT:DEV:EVEN?"  ! Yes , read register
280     ENTER @Pm;Event              ! (this also clears it)
290     OUTPUT @Pm;"STAT:DEV:COND?"
300     ENTER @Pm;Cond
310     IF Cond=0 THEN
320       PRINT "Sensor disconnected"
330     ELSE
340       PRINT "Sensor connected"
350     END IF
360   END IF
370   GOTO 170               ! Return to idle loop
380   END
```

**Example 2:**

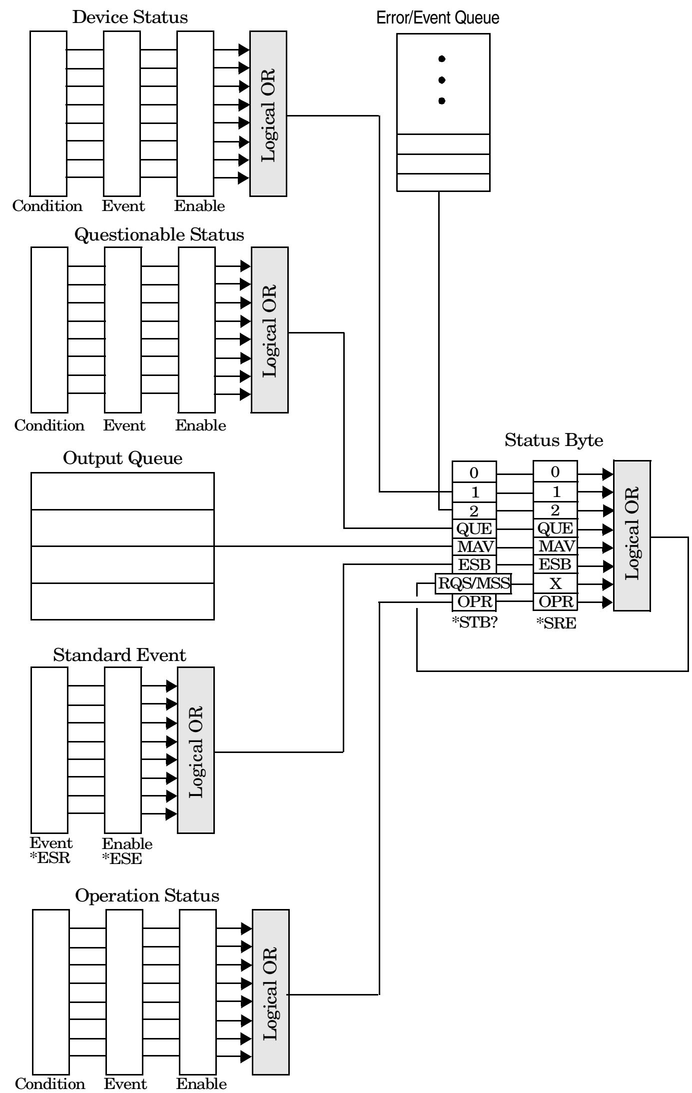
```
10    ! Program to generate an SRQ when an over limit
20    ! condition occurs.
30    !
40    ASSIGN @Pm TO 713      ! Power meter GPIB address
50   ON INTR 7 GOTO Srq_i   ! Define service request handler
60    CLEAR @Pm              ! Selective device clear
70    OUTPUT @Pm;"*CLS"      ! Clear registers
80    OUTPUT @Pm;"SYST:PRES" ! Preset meter
90    !
100   ! Set upper limit to 2dBm and configure the operation
status
110   ! so that an over limit condition will cause an SRQ.
120   !
130   OUTPUT @Pm;"CALC:LIM:UPP 2DBM"
140   OUTPUT @Pm;"CALC:LIM:STAT ON"
150   OUTPUT @Pm;"STAT:OPER:PTR 4096"
160   OUTPUT @Pm;"STAT:OPER:ENAB 4096"
170   OUTPUT @Pm;"*SRE 128"
180   !
190   ENABLE INTR 7;2  ! Enable an SRQ to cause an interrupt
200   LOOP             ! Idle loop
210   ! Forever
220   END LOOP
230   !
240   ! When a SRQ is detected, the following routine will
service it.
250   !
```

```
260 Srq_i:     !
270   St=SPOLL(@Pm) ! Serial Poll (reads status byte)
280   IF BIT(St,7)=1 THEN      ! Operation status bit set?
290     OUTPUT @Pm;"STAT:OPER?"! Yes , read register
300     ENTER @Pm;Oper         ! (this also clears it)
310     OUTPUT @Pm;"STAT:OPER:ULF?"
320     ENTER @Pm;Ulf
330     IF Ulf=2 THEN PRINT "Over limit detected"
340   END IF
350   GOTO 190                 ! Return to idle loop
360   END
```

## Status Registers

The Status System in the power meter is shown in Figure 1-9. The Operation Status and Questionable Status groups are 16 bits wide, while the Status Byte and Standard Event groups are 8 bits wide. In all 16-bit groups, the most significant bit (bit 15) is not used and is always set to 0.

**Figure 1-9** **Status System**




## The Status Byte Summary Register

The status byte summary register reports conditions from other status registers. Query data waiting in the power meter’s output buffer is immediately reported through the “message available” bit (bit 4). Clearing an event register clears the corresponding bits in the status byte summary register. Reading all messages in the output buffer, including any pending queries, clears the message available bit.

**Table 1-4 Bit Definitions - Status Byte Register**

| Bit Number | Decimal Weight | Definition |
|---|---|---|
| 0 | 1 | Not Used (Always set to 0) |
| 1 | 2 | Device Status Register summary bit.<br>One or more bits are set in the Device Status Register (bits must be “enabled” in enable register) |
| 2 | 4 | Error/Event Queue |
| 3 | 8 | Questionable Status Register summary bit.<br>One or more bits are set in the Questionable Status Register (bits must be “enabled” in enable register). |
| 4 | 16 | Data Available<br>Data is available in the power meter’s output buffer. |
| 5 | 32 | Standard Event<br>One or more bits are set in the Standard Event register (bits must be “enabled” in enable register). |
| 6 | 64 | Request Service<br>The power meter is requesting service (serial poll). |
| 7 | 128 | Operation Status Register summary bit.<br>One or more bits are set in the Operation Status Register (bits must be “enabled” in enable register). |

Particular bits in the status byte register are cleared when:

- The standard event, Questionable status, operation status and device status are queried.
- The error/event queue becomes empty.
- The output queue becomes empty.

The status byte enable register (`SRE`, service request enable) is cleared when you:

- cycle the instrument power.
- execute a `*SRE 0` command.

### Using *STB? to Read the Status Byte

The *STB? (status byte query) command is similar to a serial poll except it is processed like any other power meter command. The *STB? command returns the same result as an IEEE-488 serial poll except that the request service bit (bit 6) *is not* cleared if a serial poll has occurred. The *STB? command is not handled automatically by the IEEE-488 bus interface hardware and the command is executed only after previous commands have completed. Using the *STB? command does not clear the status byte summary register.

### The Standard Event Register

The standard event register reports the following types of instrument events: power-on detected, command and syntax errors, command execution errors, self-test or calibration errors, query errors, or when an overlapped command completes following a *OPC command. Any or all of these conditions can be reported in the standard event summary bit through the enable register. You must write a decimal value using the *ESE (event status enable) command to set the enable register mask.

**Table 1-5 Bit Definitions - Standard Event Register**

| Bit Number | Decimal Value | Definition |
|---|---|---|
| 0 | 1 | Operation Complete<br>All overlapped commands following an *OPC command have been completed. |
| 1 | 2 | Not Used. (Always set to 0.) |
| 2 | 4 | Query Error<br>A query error occurred, refer to error numbers 410 to 440 in the *User's Guide*. |
| 3 | 8 | Device Error<br>A device error occurred, refer to error numbers 310 to 350 in the *User's Guide*. |
| 4 | 16 | Execution Error<br>An execution error occurred, refer to error numbers 211 to 241 in the *User's Guide*. |
| 5 | 32 | Command Error<br>A command syntax error occurred, refer to error numbers 101 to 161 in the *User's Guide*. |

**Table 1-5** **Bit Definitions - Standard Event Register**

| Bit Number | Decimal Value | Definition |
|---|---|---|
| 6 | 64 | User request. |
| 7 | 128 | Power On<br>Power has been turned off and on since the last time the event register was read or cleared. |

The standard event register is cleared when you:

- send a `*CLS` (clear status) command.
- query the event register using the `*ESR?` (event status register) command.

The standard event enable register is cleared when you:

- cycle the instrument power.
- execute a `*ESE 0` command.

### Questionable Status Register

The questionable status register provides information about the quality of the power meter's measurement results. Any or all of these conditions can be reported in the questionable data summary bit through the enable register. You must write a value using the `STATus:QUEStionable:ENABle` command to set the enable register mask.

The questionable status model is shown in the pullout at the end of this chapter.

The following bits in these registers are used by the power meter.

**Table 1-6** **Bit Definitions - Questionable Status Registers**

| Bit Number | Decimal Weight | Definition |
|---|---|---|
| 0 to 2 | - | Not used |
| 3 | 8 | POWer Summary |
| 4 to 7 | - | Not used |

**Table 1-6 Bit Definitions - Questionable Status Registers**

| Bit Number | Decimal Weight | Definition |
|---|---|---|
| 8 | 256 | CALibration Summary |
| 9 | 512 | Power On Self Test |
| 10 to 14 | - | Not Used |
| 15 | - | Not used (always 0) |

The condition bits are set and cleared under the following conditions:

| Bit Number | Meaning | EVENts Causing Bit Changes |
|---|---|---|
| 3 | POWer Summary | This is a summary bit for the Questionable POWer Register.<br>• **SET:**<br>Error -230, “Data corrupt or stale”<br>Error -231, “Data questionable;Input Overload”<br>Error -231, “Data questionable;Input Overload ChA”[a]<br>Error -231, “Data questionable;Input Overload ChB”[a]<br>Error -231, “Data questionable;PLEASE ZERO”<br>Error -231, “Data questionable;PLEASE ZERO ChA”[a]<br>Error -231, “Data questionable;PLEASE ZERO ChB”[a]<br>Error -231, ”Data questionable;Lower window log error”[a]<br>Error -231, ”Data questionable;Upper window log error”[a]<br>• **CLEARED:** When no errors are detected by the power meter during a measurement covering the causes given for it to set. |

| Bit Number | Meaning | EVENts Causing Bit Changes |
|---|---|---|
| 8 | CALibration Summary | This is a summary bit for the Questionable CALibration Register.<br>• **SET:**<br>These may be caused by<br>`CALibration[1\|2]:ZERO:AUTO ONCE` or<br>`CALibration[1\|2]:AUTO ONCE` or<br>`CALibration[1\|2][:ALL]` or<br>`CALibration[1\|2][:ALL]?`.<br>Error -231, “Data questionable; ZERO ERROR”<br>Error -231, “Data questionable; ZERO ERROR ChA”[a]<br>Error -231, “Data questionable; ZERO ERROR ChB”[a]<br>Error -231, “Data questionable; CAL ERROR”<br>Error -231, “Data questionable; CAL ERROR ChA”[a]<br>Error -231, “Data questionable; CAL ERROR ChB”[a]<br>• **CLEARED:** When any of the commands listed above succeed and no errors are placed on the error queue. |
| 9 | Power On Self Test | • **SET:** This bit is set when the power on self test fails.<br>• **CLEARED:** When the power on self test passes. |

a. N1912A only

### Operation Status

The Operation Status group monitors conditions in the power meter’s measurement process.

The Operation status model is shown in the pullout at the end of this chapter.

The following bits in these registers are used by the power meter:

| Bit Number | Decimal Weight | Definition |
|---|---|---|
| 0 | 1 | CALibrating Summary |
| 1 - 3 | - | Not used |
| 4 | 16 | MEASuring Summary |
| 5 | 32 | Waiting for TRIGger Summary |
| 6 - 9 | - | Not used |

| Bit Number | Decimal Weight | Definition |
|---|---|---|
| 10 | 1024 | SENSe Summary |
| 11 | 2048 | Lower Limit Fail Summary |
| 12 | 4096 | Upper Limit Fail Summary |
| 13 to 14 | - | Not used |
| 15 | - | Not used (always 0) |

The condition bits are set and cleared under the following conditions:

| Bit Number | Meaning | EVENts Causing Bit Changes |
|---|---|---|
| 0 | CALibrating | This is a summary bit for the Operation CALibrating Register.<br>• **SET:** At beginning of zeroing (`CALibration:ZERO:AUTO ONCE`) and at the beginning of calibration (`CALibration:AUTO ONCE`). Also for the compound command/query `CALibration[:ALL]?`, this bit is set when sensor zeroing begins.<br>• **CLEARED:** At the end of zeroing or calibration. |
| 4 | MEASuring | This is a summary bit for the Operation MEASuring Register.<br>• **SET:** When the power meter is taking a measurement.<br>• **CLEARED:** When the measurement is finished. |
| 5 | Waiting for TRIGger | This is a summary bit for the Operation TRIGger Register.<br>• **SET:** When the power meter enters the “wait for trigger” state.<br>• **CLEARED:** When the power meter enters the “idle” state. |
| 10 | SENSe | This is a summary bit for the Operation SENSe Register.<br>• **SET:** When the power meter is reading data from the power sensor’s EEPROM.<br>• **CLEARED:** When the power meter is not reading data from the power sensor’s EEPROM. |

| Bit Number | Meaning | EVENts Causing Bit Changes |
|---|---|---|
| 11 | Lower Limit Fail | This is a summary bit for the Lower Limit Fail Register.<br>• **SET:** If a measurement is made and either a channel or window lower limit test fails.<br>• **CLEARED:** If a measurement is made and the lower limit test is not enabled or the test is enabled and passes. |
| 12 | Upper Limit Fail | This is a summary bit for the Upper Limit Fail Register.<br>• **SET:** If a measurement is made and either a channel or window upper limit test fails.<br>• **CLEARED:** If a measurement is made and the upper limit test is not enabled or the test is enabled and passes. |

## Device Status Register

The device status register set contains bits which give device dependent information.

The following bits in these registers are used by the power meter:

| Bit Number | Decimal Weight | Definition |
|---|---|---|
| 0 | - | Not used |
| 1 | 2 | Channel A sensor connected |
| 2 | 4 | Channel B sensor connected[a] |
| 3 | 8 | Channel A sensor error |
| 4 | 16 | Channel B sensor error[a] |
| 5 | 32 | Channel A sensor Front/Rear |
| 6 | 64 | Channel B sensor Front/Rear[a] |
| 14 | 16384 | Front Panel key press |

a. N1912A only

The condition bits are set and cleared under the following conditions:

| Bit Number | Meaning | EVENts Causing Bit Changes |
|---|---|---|
| 1 | Channel A sensor connected | • **SET:** When a power sensor is connected to the Channel A input.<br>• **CLEARED:** When no power sensor is connected to the Channel A input. |
| 2 | Channel B sensor connected | • **SET:** When a power sensor is connected to the Channel B input.<br>• **CLEARED:** When no power sensor is connected to the Channel B input. |
| 3 | Channel A error | • **SET:** If the power sensor EEPROM on Channel A has failed or if there are power sensors connected to both the rear and front panel Channel A connectors.<br>• **CLEARED:** In every other condition. |
| 4 | Channel B error | • **SET:** If the power sensor EEPROM on Channel B has failed or if there are power sensors connected to both the rear and front panel Channel B connectors.<br>• **CLEARED:** In every other condition. |
| 5 | Channel A Front/Rear | • **SET:** If a power sensor is connected to the Channel A rear panel.<br>• **CLEARED:** If a power sensor is connected to the Channel A front panel. |
| 6 | Channel B Front/Rear | • **SET:** If a power sensor is connected to the Channel B rear panel.<br>• **CLEARED:** If a power sensor is connected to the Channel B front panel. |
| 14 | Front Panel Key Press | This is an event, and DOES NOT set the condition register. The bit is set in the event register which is cleared when read. Note that the transition registers are of no use for this bit. |

## Using the Operation Complete Commands

The *OPC? and *OPC commands allow you to maintain synchronization between the computer and the power meter. The *OPC? query command places an ASCII character 1 into the power meter's output queue when all pending power meter commands are complete. If your program reads this response before continuing program execution, you can ensure synchronization between one or more instruments and the computer.

The *OPC command sets bit 0 (Operation Complete) in the Standard Event Status Register when all pending power meter operations are complete. By enabling this bit to be reflected in the Status Register, you can ensure synchronization using the GPIB serial poll.

**NOTE** For LAN and USB use the *STB? command. See "Using *STB? to Read the Status Byte" on page 55.

### Procedure

- Send a device clear message to clear the power meter's output buffer.
- Clear the event registers with the *CLS (clear status) command.
- Enable operation complete using the *ESE 1 command (standard event register).
- Send the *OPC? (operation complete query) command and enter the result to assure synchronization.
- Send your programming command string, and place the *OPC (operation complete) command as the last command.
- Send the *STB? (status byte query) command to poll the register. This command does not clear the status byte summary register.

  In GPIB mode only you can use a serial poll to check to see when bit 5 (standard event) is set in the status byte summary register. You could also configure the power meter for an SRQ interrupt by sending *SRE 32 (status byte enable register, bit 5).

### Examples

This example program uses the *OPC? command to determine when the power meter has finished calibrating.

```
CAL:AUTO ONCE
*OPC?
MEAS:POW:AC?
```

This example GPIB program, in HP Basic, uses the `*OPC` command and serial poll to determine when the power meter has finished calibrating. The advantage to using this method over the `*OPC?` command is that the computer can perform other operations while it is waiting for the power meter to finish calibrating.

```
10 ASSIGN @Power TO 713
20 OUTPUT @Power;"*CLS"
30 OUTPUT @Power;"*ESE 1"
40 OUTPUT @Power;"CAL:AUTO ONCE;*OPC"
50 WHILE NOT BIT(SPOLL(@Power),5)
60 !(Computer carries out other operations here)
70 END WHILE
80 OUTPUT @Power;"MEAS:POW:AC?"
90 ENTER @Power;Result
100 PRINT Result
110 END
```

# Saving and Recalling Power Meter Configurations

To reduce repeated programming, up to ten power meter configurations can be stored in the power meter's non-volatile memory. The error list, remote addresses, sensor calibration table data, zeroing and calibration information are not stored.

## How to Save and Recall a Configuration

Power meter configurations are saved and recalled with the following commands:

```
*SAV <NRf>
*RCL <NRf>
```

The range of values for `<NRf>` in the above commands is 1 to 10.

### Example Program

```
10 ASSIGN @POWER TO 713
20 !Configure the power meter
30 OUTPUT @POWER;"UNIT:POW W"
40 OUTPUT @POWER;"SENS:CORR:LOSS2 -10"
50 OUTPUT @POWER;"SENS:CORR:LOSS2:STAT ON"
60 !Save the configuration
70 OUTPUT @POWER;"*SAV 5"
80 PRINT "Configuration Saved"
90 !Now reset the power meter
100 OUTPUT @POWER;"*RST"
110 ! Recall the configuration
120 OUTPUT @POWER;"*RCL 5"
130 PRINT "Configuration Recalled"
140 PRINT "Save and Recall complete"
150 END
```

## Using Device Clear to Halt Measurements

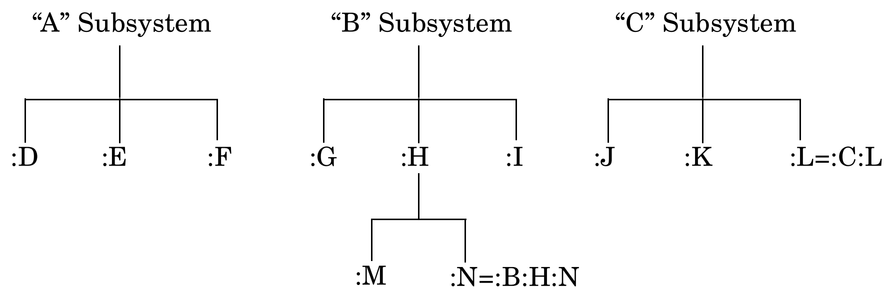
Device clear is an IEEE-488 low-level bus message which can be used to halt measurements in progress. Different programming languages and IEEE-488 interface cards provide access to this capability through their own unique commands. The status registers, the error queue, and all configuration states are left unchanged when a device clear message is received. Device clear performs the following actions.

- All measurements in progress are aborted.
- The power meter returns to the trigger “idle state”.
- The power meter’s input and output buffers are cleared.
- The power meter is prepared to accept a new command string.

**NOTE** For interfaces the that do not support a low-level device clear, use the `ABORt` command.

# An Introduction to the SCPI Language

Standard Commands for Programmable Instruments (SCPI) defines how you communicate with an instrument from a bus controller. The SCPI language uses a hierarchical structure similar to the file systems used by many bus controllers. The command tree is organized with root-level commands (also called subsystems) positioned at the top, with multiple levels below each root-level command. You must specify the complete path to execute the individual lower-level commands.




## Mnemonic Forms

Each keyword has both a long and a short form. A standard notation is used to differentiate the short form keyword from the long form keyword. The long form of the keyword is shown, with the short form portion shown in uppercase characters, and the rest of the keyword shown in lowercase characters. For example, the short form of `TRIGger` is `TRIG`.

## Using a Colon (:)

When a colon is the first character of a command keyword, it indicates that the next command mnemonic is a root-level command. When a colon is inserted between two command mnemonics, the colon moves the path down one level in the present path (for the specified root-level command) of the command tree. You *must* separate command mnemonics from each other using a colon. *You can omit the leading colon if the command is the first of a new program line.*

## Using a Semicolon (;)

Use a semicolon to separate two commands within the same command string. The semicolon does not change the present path specified. For example, the following two statements are equivalent. Note that in the first statement the first colon is optional but the third is compulsory.

```
:DISP:FORM DIG;:DISP:RES 2
:DISP:FORM DIG;RES 2
```

## Using a Comma (,)

If a command requires more than one parameter, you must separate adjacent parameters using a comma.

## Using Whitespace

You *must* use whitespace characters, [tab], or [space] to separate a parameter from a command keyword. Whitespace characters are generally ignored *only* in parameter lists.

## Using "?" Commands

The bus controller may send commands at any time, but a SCPI instrument may only send responses when *specifically* instructed to do so. Only query commands (commands that end with a "?") instruct the instrument to send a response message. Queries return either measured values or internal instrument settings.

**NOTE** If you send two query commands without reading the response from the first, then attempt to read the second response, you may receive some data from the first response followed by the complete second response. To avoid this, do not send a query command without reading the response. When you cannot avoid this situation, send a device clear before sending the second query command.

## Using “*” Commands

Commands starting with a “*” are called common commands. They are required to perform the identical function for *all* instruments that are compliant with the IEEE-488.2 interface standard. The “*” commands are used to control reset, self-test, and status operations in the power meter.

## Syntax Conventions

Throughout this guide, the following conventions are used for SCPI command syntax.

- Square brackets ([]) indicate optional keywords or parameters.
- Braces ({}) enclose one or more parameters that may be included zero or more times.
- Triangle brackets (<>) indicate that you must substitute a value for the enclosed parameter.
- Bars ( | ) can be read as “or” and are used to separate alternative parameter options.

### Syntax Diagram Conventions

- Solid lines represent the recommended path.
- Ovals enclose command mnemonics. The command mnemonic must be entered exactly as shown.
- Dotted lines indicate an optional path for bypassing secondary keywords.
- Arrows and curved intersections indicate command path direction.

## SCPI Data Types

The SCPI language defines different data formats for use in program messages and response messages. Instruments are flexible listeners and can accept commands and parameters in various formats. However, SCPI instruments are precise talkers. This means that SCPI instruments *always* respond to a particular query in a predefined, rigid format.

### <boolean> Definition

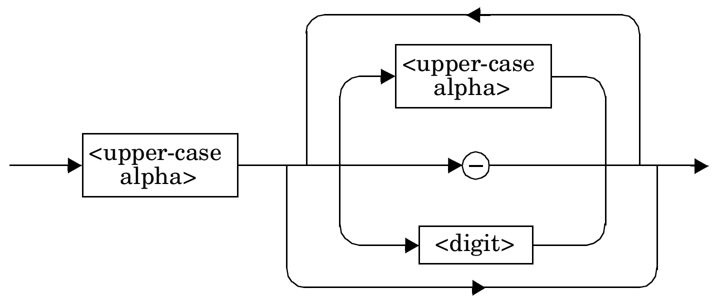
Throughout this document <boolean> is used to represent ON|OFF|<NRf>. boolean parameters have a value of 0 or 1 and are unitless. ON corresponds to 1 and OFF corresponds to 0.

On input, an <NRf> is rounded to an integer. A nonzero result is interpreted as 1.

Queries always return a 1 or 0, never ON or OFF.

### <character_data> Definition

Throughout this document <character_data> is used to represent character data, that is, A - Z, a - z, 0 - 9 and _ (underscore). For example: START and R6_5F. The format is defined as:

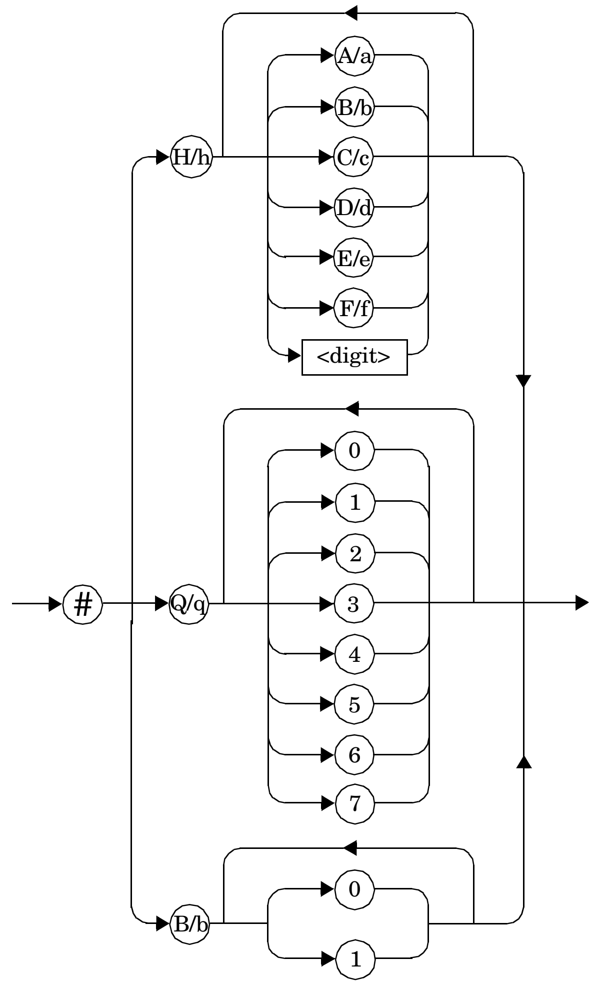



### <NAN> Definition

Not a number (NAN) is represented as 9.91 E37. Not a number is defined in IEEE 754.

### <non-decimal numeric> Definition

Throughout this document `<non-decimal numeric>` is used to represent numeric information in bases other than ten (that is, hexadecimal, octal and binary). The following syntax diagram shows the standard for these three data structures. For example, #HA2F, #ha4e, #Q62, #q15, #B01011.



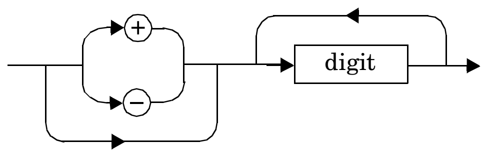

Refer to section 7.7.4.1 of IEEE 488.2 for further details.

### <NRf> Definition

Throughout this document `<NRf>` is used to denote a flexible numeric representation. For example: +200; -56; +9.9E36. Refer to section 7.7.2.1 of IEEE 488.2 for further details.

### <NR1> Definition

Throughout this document `<NR1>` numeric response data is defined as:



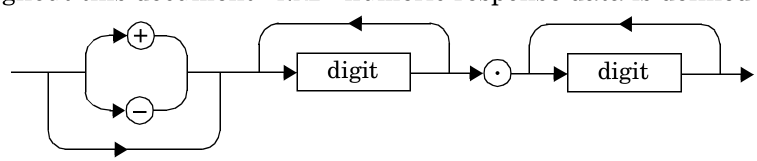

For example:

- 146
- +146
- -12345

Refer to section 8.7.2 of IEEE 488.2 for further details.

### <NR2> Definition

Throughout this document `<NR2>` numeric response data is defined as:



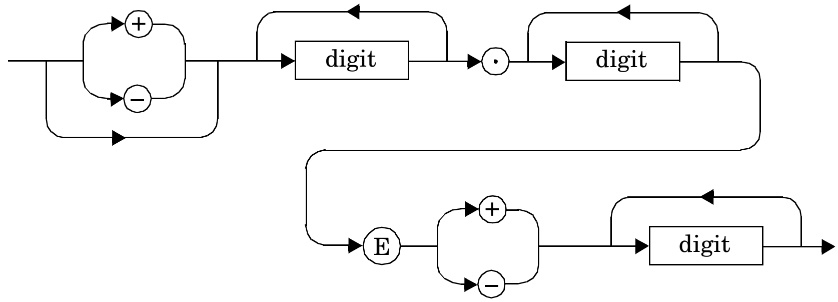

For example:

- 12.3
- +1.2345
- -0.123

Refer to section 8.7.3 of IEEE 488.2 for further details.

### <NR3> Definition

Throughout this document `<NR3>` numeric response data is defined as:




For example:

- 1.23E+6
- 123.4E-54
- -1234.567E+90

Refer to section 8.7.4 of IEEE 488.2 for further details.

### <numeric_value> Definition

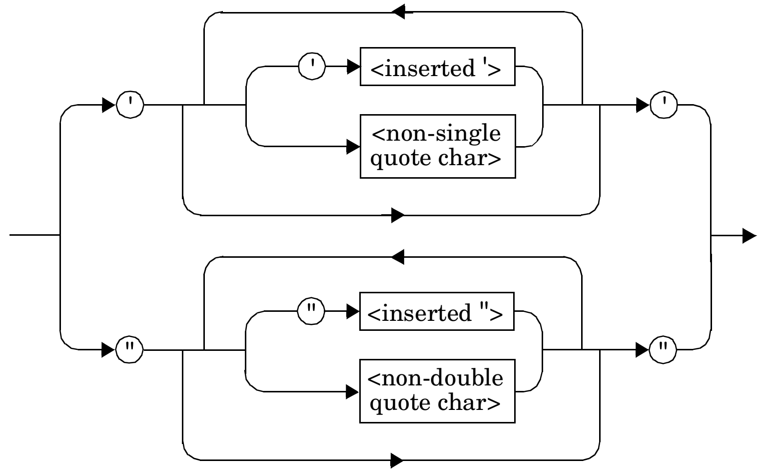
Throughout this document the decimal numeric element is abbreviated to `<numeric_value>`. For example, `<NRf>`, `MINimum`, `MAXimum`, `DEFault` or Not A Number (`NAN`).

### <string> Definition

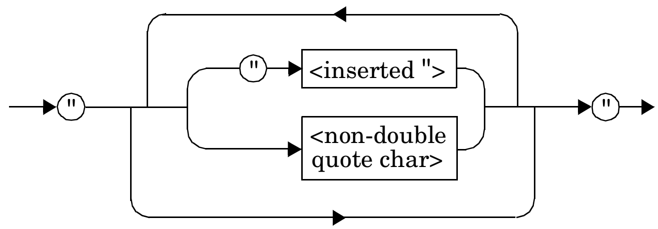
Throughout this document `<string>` is used to represent 7-bit ASCII characters.

The format is defined as:

**Program Data**




**Response Data**




## Input Message Terminators

Program messages sent to a SCPI instrument *must* terminate with a <newline> character. The IEEE.488 EOI (end or identify) signal is interpreted as a <newline> character and may also be used to terminate a message in place of the <newline> character. A <carriage return> followed by a <newline> is also accepted. Many programming languages allow you to specify a message terminator character or EOI state to be automatically sent with each bus transaction. Message termination *always* sets the current path back to the root-level.

# SCPI Compliance Information

The power meter complies with the rules and regulations of the present version of SCPI (Standard Commands for Programmable Instruments). You can determine the SCPI version with which the power meter's is in compliance by sending the SYSTem:VERSion? command from the remote interface.

The following commands are device-specific to the power meter. They are not included in the 1999.0 version of the SCPI standard. However, these commands are designed with the SCPI format in mind and they follow all of the syntax rules of the standard.

```
CALibration[1|2]:RCALibration
CALibration[1|2]:RCFactor
DISPlay[:WINDow[1|2]]:FORMat
DISPlay[:WINDow[1|2]]:METer:LOWer
DISPlay[:WINDow[1|2]]:METer:UPPer
DISPlay[:WINDow[1|2]]:RESolution
DISPlay[:WINDow[1|2]]:SELect
MEMory:CLEar[:NAME]
MEMory:TABLe:SELect
MEMory:STATe:DEFine
MEMory:TABLe:GAIN[:MAGNitude]
MEMory:TABLe:GAIN:POINts?
MEMory:TABLe:MOVE
[SENSe[1]]|SENSe2:AVERage:SDETect
[SENSe[1]]|SENSe2:CORRection:CFACtor
[SENSe[1]]|SENSe2:CORRection:DCYCle
[SENSe[1]]|SENSe2:CORRection:FDOFfset
[SENSe[1]]|SENSe2:SPEed
[SENSe[1]]|SENSe2:POWer:AC:RANGe
SERVice:SENSor[1|2]:CDATE?
SERVice:SENSor[1|2]:CPLace?
SERVice:SENSor[1|2]:SNUMber?
SERVice:SENSor[1|2]:TYPE?
SYSTem:COMMunicate:LAN:AIP
SYSTem:COMMunicate:LAN:CURRent:ADDRess?
SYSTem:COMMunicate:LAN:CURRent:DGATeway?
SYSTem:COMMunicate:LAN:CURRent:DNAMe?
SYSTem:COMMunicate:LAN:CURRent:SMASk?
```

```
SYSTem:COMMunicate:LAN:ADDRess
SYSTem:COMMunicate:LAN:DGATeway
SYSTem:COMMunicate:LAN:DHCP
SYSTem:COMMunicate:LAN:HNAMe
SYSTem:COMMunicate:LAN:RESTart
SYSTem:COMMunicate:LAN:SMASk
SYSTem:LOCal
SYSTem:REMote
SYSTem:RWLock
UNIT[1|2]:POWer:RATio
```

# Summary Of Commands

For detail of each SCPI (Standard Commands for Programmable Instruments) command available to program the power meter, refer to later chapters for more details on each command.

**NOTE** This Guide details the commands available for both the N1911A and the N1912A power meters. As the N1911A is a single channel power meter only channel A can be selected. Where instances of channel selection are detailed in this document they are only relevant for the N1912A.

In different subsystems the numeric suffix of program mnemonics can represent either a channel selection or a window selection. Refer to the appropriate command description to verify the meaning of the numeric suffix.

With commands that require you to specify a channel, Channel A is represented by a 1 and Channel B by a 2. If you omit the channel number, Channel A is assumed.

With commands that require you to specify a window, the upper window is represented by a 1 and the lower window by a 2. If you omit the window number, the upper window is assumed.

All the commands listed also have queries unless otherwise stated in the “Notes” column.

# Making Measurements on Wireless Communication Standards

The following sections describe typical measurements you may want to make. They are also described, for front panel operation, in the *User's Guide*.

The optimum method of measuring these Wireless Communication Standards is to use the SYSTem:PRESet <character_data> command and use one of the following values.

- `GSM900` - See "GSM900" on page 375 for greater detail.
- `EDGE`- See "EDGE" on page 378 for greater detail.
- `CDMAone`- See "CDMAone" on page 382 for greater detail.
- `CDMA2000`- See "CDMA2000" on page 386 for greater detail.
- `WCDMA`- See "W-CDMA" on page 390 for greater detail.
- `BLUetooth`- See "BLUetooth" on page 394 for greater detail.
- `MCPa`- See "MCPA" on page 397 for greater detail.
- `RADar`- See "RADAR" on page 400 for greater detail.
- `WL802DOT11A`- See "802.11a and HiperLan2" on page 404 for greater detail.
- `WL802DOT11B`- See "892.11b/g" on page 407 for greater detail.
- `XEVDO`- See "1xeV-DO" on page 410 for greater detail.
- `XEVDV`- See "1xeV-DV" on page 413 for greater detail.
- `TDSCdma`- See "TD-SCDMA" on page 415 for greater detail.
- `NADC`- See "NADC" on page 419 for greater detail.
- `IDEN`- See "IDEN" on page 423 for greater detail.
- `DVB`- See "DVB" on page 427 for greater detail.
- `HIPERLAN2`- See "802.11a and HiperLan2" on page 404 for greater detail.

## Starting a Preset Example

```
10 *CLS !Clears error queue
20 *RST !Resets meter settings to their default states
30 :SYST:ERR? <read string> !The system error query should
!return “0: No Error”
40 SERV:SENS:TYPE? !The sensor type query should return one
!of the following:E9321A|E9322A|E9323A|E9325A|E9326A|E9327A|
!N1921A|N1922A The GSM setup is only valid with these
!sensors
50 SYSTem:PRESet “GSM900”
```

# 2 Measurement Commands

# Measurement Commands

Measurement commands are high level commands used to acquire data. They enable you to trade interchangeability against fine control of the measurement process.

| Measurement Command | Description |
|---|---|
| MEASure? | Provides the simplest way to program a power meter for measurements. MEASure? is a compound command which is equivalent to an ABORT followed by a CONFigure and a READ?. It does not enable much flexibility or control over measurement settings. |
| CONFigure | Used to change the power meter's configuration values. CONFigure must then be followed by another command which takes the measurement—for example, a READ? followed by a FETCh?. |
| READ? | Takes a measurement using parameters previously set up using either CONFigure or lower level commands. READ? is equivalent to an ABORt followed by an INITiate1 (which performs the data acquisition) and a FETCh? |
| FETCh? | Retrieves measurements taken by INITiate[a]. |

a. INITiate is described in Chapter 13, "TRIGger Subsystem," on page 449.

The `CONFigure`, `FETCh?`, `READ?` and `MEASure?` commands all have a numeric suffix which refers to a specific window/measurement. Figure 2-1 shown an example of the configuration returned result windows.

**Figure 2-1 Measurement Display CALCulate Block Window**

| |
|---|
| CONfigure1?<br>upper window/upper measurement |
| CONFigure3?<br>upper window/lower measurement |

| |
|---|
| CONFigure2?<br>lower window/upper measurement |
| CONFigure4?<br>lower window/lower measurement |

## Optional Parameters

`CONFigure`, `FETCh?`, `READ?` and `MEASure?` have the following three optional parameters:

- An expected power value.
- A resolution.
- A source list.

### Expected Power Value

An `<expected_value>` parameter is only required if you are using an E-series power sensor. It has no effect on P-Series or 8480 series power sensor. The value entered determines which of the power sensor's two ranges is used for the measurement. If the current setting of the power sensor's range is no longer valid for the new measurement, specifying the expected power value decreases the time taken to obtain a result.

### Resolution

The `<resolution>` parameter sets the resolution of the specified window. This parameter does not affect the resolution of the remote data but it does affect the auto averaging setting. Where a channel is set up in both the upper and lower window and the `<resolution>` parameter settings for these windows are different, the highest resolution setting is taken to calculate the averaging. If you are making a ratio or difference measurement the `<resolution>` parameters are applied to both channels.

### Source List

The <source list> parameter is used to define:

- What channels the measurements will be made on, for a dual channel measurement.
- Whether the calculation is A-B or B-A, for a dual channel difference measurement.
- Whether the calculation is A/B or B/A, for a ratio measurement.

Entering a <source list> is only required if you are using an N1912A. As the N1911A has a single channel only, the source list can only be channel A.

The following commands are described in this chapter:

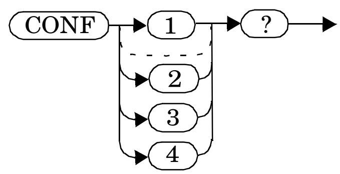
| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
| CONFigure[1]\|2\|3\|4 | | [query only] | page 84 |
| CONFigure[1]\|2\|3\|4 | | | |
| [:SCALar] | | | |
| [:POWer:AC] | [<expected_value>[,<resolution>[,<source list>]]] | [no query] | page 87 |
| :RELative | [<expected_value>[,<resolution>[,<source list>]]] | [no query] [non-SCPI] | page 89 |
| :DIFFerence | [<expected_value>[,<resolution>[,<source list>]]] | [no query] [non-SCPI] | page 91 |
| :RELative | [<expected_value>[,<resolution>[,<source list>]]] | [no query] [non-SCPI] | page 93 |
| :RATio | [<expected_value>[,<resolution>[,<source list>]]] | [no query] | page 95 |
| :RELative | [<expected_value>[,<resolution>[,<source list>]]] | [no query] [non-SCPI] | page 97 |
| FETCh[1]\|2\|3\|4 | | | |
| [:SCALar] | | | |
| [:POWer:AC]? | [<expected_value>[,<resolution>[,<source list>]]] | [query only] | page 100 |
| :RELative? | [<expected_value>[,<resolution>[,<source list>]]] | [query only] [non-SCPI] | page 102 |
| :DIFFerence? | [<expected_value>[,<resolution>[,<source list>]]] | [query only] [non-SCPI] | page 104 |
| :RELative? | [<expected_value>[,<resolution>[,<source list>]]] | [query only] [non-SCPI] | page 106 |

| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
| :RATio? | [<expected_value>[,<resolution>[,<source list>]]] | [query only] | page 108 |
| :RELative? | [<expected_value>[,<resolution>[,<source list>]]] | [query only] [non-SCPI] | page 110 |
| READ[1]\|2\|3\|4 | | | |
| [:SCALar] | | | |
| [:POWer:AC]? | [<expected_value>[,<resolution>[,<source list>]]] | [query only] | page 113 |
| :RELative? | [<expected_value>[,<resolution>[,<source list>]]] | [query only] [non-SCPI] | page 115 |
| :DIFFerence? | [<expected_value>[,<resolution>[,<source list>]]] | [query only] [non-SCPI] | page 117 |
| :RELative? | [<expected_value>[,<resolution>[,<source list>]]] | [query only] [non-SCPI] | page 119 |
| :RATio? | [<expected_value>[,<resolution>[,<source list>]]] | [query only] | page 121 |
| :RELative? | [<expected_value>[,<resolution>[,<source list>]]] | [query only] [non-SCPI] | page 123 |
| MEASure[1]\|2\|3\|4 | | | |
| [:SCALar] | | | |
| [:POWer:AC]? | [<expected_value>[,<resolution>[,<source list>]]] | [query only] | page 126 |
| :RELative? | [<expected_value>[,<resolution>[,<source list>]]] | [query only] [non-SCPI] | page 128 |
| :DIFFerence? | [<expected_value>[,<resolution>[,<source list>]]] | [query only] [non-SCPI] | page 130 |
| :RELative? | [<expected_value>[,<resolution>[,<source list>]]] | [query only] [non-SCPI] | page 132 |
| :RATio? | [<expected_value>[,<resolution>[,<source list>]]] | [query only] | page 134 |
| :RELative? | [<expected_value>[,<resolution>[,<source list>]]] | [query only] [non-SCPI] | page 136 |

# CONFigure[1]|2|3|4?

This query returns the present configuration of the specified window/measurement.

## Syntax




The string returned depends on the setting of the `CALCulate:MATH` and `CALCulate:RELative:STATe` commands.

The configuration is returned as a quoted string in the following format:

"<function> <expected_value>,<resolution>,<source list>"

| CALCulate:MATH | CALCulate: RELative: STATe | Function | <source list> |
|---|---|---|---|
| (SENSe1) | OFF | :POW:AC | (@1) |
| (SENSe2)[a] | OFF | :POW:AC | (@2) |
| (SENSe1) | ON | :POW:AC:REL | (@1) |
| (SENSe2)[a] | ON | :POW:AC:REL | (@2) |
| (SENSe1 - SENSe2)[a] | OFF | :POW:AC:DIFF | (@1),(@2) |
| (SENSe2 - SENSe1)[a] | OFF | :POW:AC:DIFF | (@2),(@1) |
| (SENSe1 - SENSe2)[a] | ON | :POW:AC:DIFF:REL | (@1),(@2) |
| (SENSe2 - SENSe1)[a] | ON | :POW:AC:DIFF:REL | (@2),(@1) |
| (SENSe1 - SENSe1) | OFF | :POW:AC:DIFF | (@1),(@1) |
| (SENSe2 - SENSe2)[a] | OFF | :POW:AC:DIFF | (@2),(@2) |
| (SENSe1 - SENSe1) | ON | :POW:AC:DIFF:REL | (@1),(@1) |
| (SENSe2 - SENSe2)[a] | ON | :POW:AC:DIFF:REL | (@2),(@2) |
| (SENSe2 / SENSe1)[a] | OFF | :POW:AC:RAT | (@1),(@2) |

| CALCulate:MATH | CALCulate: RELative: STATe | Function | <source list> |
|---|---|---|---|
| (SENSe2 / SENSe1)[a] | OFF | :POW:AC:RAT | (@2),(@1) |
| (SENSe1 / SENSe2)[a] | ON | :POW:AC:RAT:REL | (@1),(@2) |
| (SENSe2 / SENSe1)[a] | ON | :POW:AC:RAT:REL | (@2),(@1) |
| (SENSe1/SENSe1) | OFF | POW:AC:RAT | (@1),(@1) |
| (SENSe2/SENSe2)[a] | OFF | POW:AC:RAT | (@2),(@2) |
| (SENSe1/SENSe1) | ON | POW:AC:RAT:REL | (@1),(@1) |
| (SENSe2/SENSe2)[a] | ON | POW:AC:RAT:REL | (@2),(@2) |

a. N1912A only.

`<expected_value>` returns the expected value sent by the last `CONFigure` command or +20 dBm by default. Note that when the display is showing dual windows this value is meaningless.

The `<resolution>` returned is the same as the value returned by `DISPlay:WINDow:RESolution?`. The format of the return is `<NR1>` in the range 1 through 4.

**Example**

| | |
|---|---|
| `CONF2?` | *This command queries the current configuration of the lower window / upper measurement.* |

**Reset Condition**

On reset:

- The command function is set to `:POWer:AC`.
- The expected power level is set to +20 dBm.
- The resolution is set to 3.
- The source list on the N1911A is set to channel A on both windows and their measurements.
- The source list on the N1912A is set to channel A for the upper measurement on both windows and channel B for the lower measurement on both windows.

# CONFigure[1]|2|3|4 Commands

The CONFigure commands are used on the specified window/measurement to set:

- The expected power level being measured.
- The resolution of the window/measurement.
- The channel(s) on which the measurement is to be made.

The CONFigure commands do not make the power measurement after setting the configuration. Use READ?, or alternatively use INITiate followed by a FETCh? to make the measurement.

The CONFigure command also applies the following defaults to the channel(s) which are in the specified window (the channel(s) in the window are specified in the <source list> parameter):

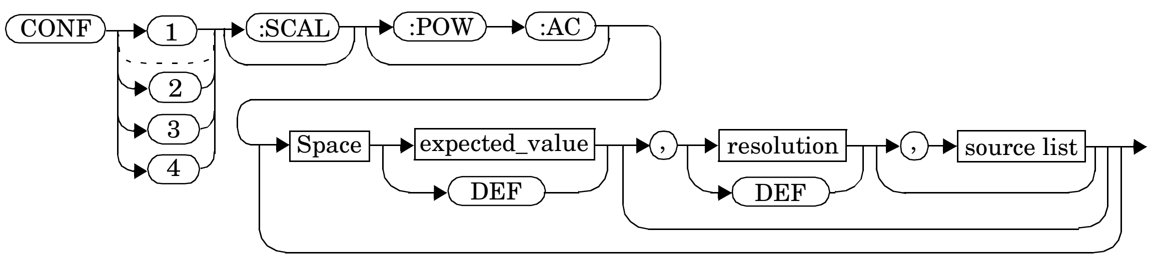
| Default Settings | Description |
|---|---|
| INITiate:CONTinuous OFF | Sets the power meter to make one trigger cycle when INITiate is sent. |
| TRIGger:SOURce IMMediate | When TRIG:SOUR is set to BUS or HOLD, sets the power meter to make the measurement immediately a trigger is received. |
| TRIGger:DELay:AUTO ON | Enables automatic delay before making the measurement. |
| SENSE:AVERage:COUNt:AUTO ON | Enables automatic filter length selection. |
| SENSE:AVERage:STATe ON | Enables averaging. |

# CONFigure[1]|2|3|4[:SCALar][:POWer:AC] [<expected_value>[,<resolution>[,<source list>]]]

This command is used on the specified window/measurement to set:

- The expected power level of the measurement.
- The resolution of the window/measurement.
- The channel on which the measurement will be made.

### Syntax




### Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value | A numeric value for the expected power level. The units of measurement are dBm and W. The default units are defined by `UNIT:POWer`. | Sensor dependent.<br>DEF[a] |
| resolution | A numeric value for the resolution. If unspecified the current resolution setting is used. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>DEF[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| source list | The channel which the command is implemented on.<br>If unspecified the current window setup is used. However, on the N1912A, if the window shows a ratio or difference measurement, the upper window defaults to channel A and the lower window to channel B. | (@1)<br>(@2)[c] |

a. The mnemonic DEF means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying DEF leaves the parameter value unchanged.
b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.
c. N1912A only.

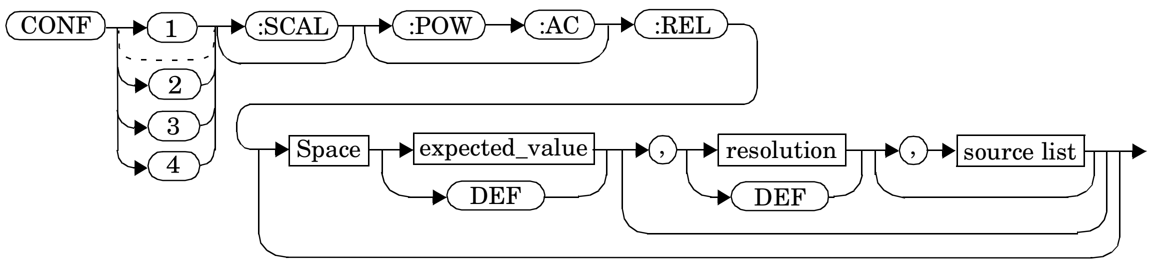
### Example

| | |
|---|---|
| `CONF1:POW:AC DEF,2,(@1)` | *This command configures the upper window/upper measurement to measure the power of channel A, using the current sensor range and a resolution setting of 2.* |

# CONFigure[1]|2|3|4[:SCALar][:POWer:AC]:RELative [<expected_value>[,<resolution>[,<source list>]]]

This command sets the measurement function, range and resolution of the specified window. It sets the measurement function to single channel with relative mode on. The relative value used is that set by the `CALCulate:RELative:MAGNitude:AUTO` command.

### Syntax




### Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | A numeric value for the expected power level. The units of measurement are dBm and W. The default units are defined by `UNIT:POWer`. | sensor dependent<br>DEF[a] |
| resolution | A numeric value for the resolution. If unspecified the current resolution setting is used. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>DEF[a] |

**CONFigure[1]|2|3|4[:SCALar][:POWer:AC]:RELative [<expected_value>[,<resolution>[,<source list>]]]**

| Item | Description/Default | Range of Values |
|---|---|---|
| source list | The channel which the command is implemented on. If unspecified the current window setup is used. However, on the N1912A, if the window shows a ratio or difference measurement, the upper window defaults to channel A and the lower window to channel B. | (@1)<br>(@2)[c] |

a. The mnemonic `DEF` means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying `DEF` leaves the parameter value unchanged.

b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.

c. N1912A only.

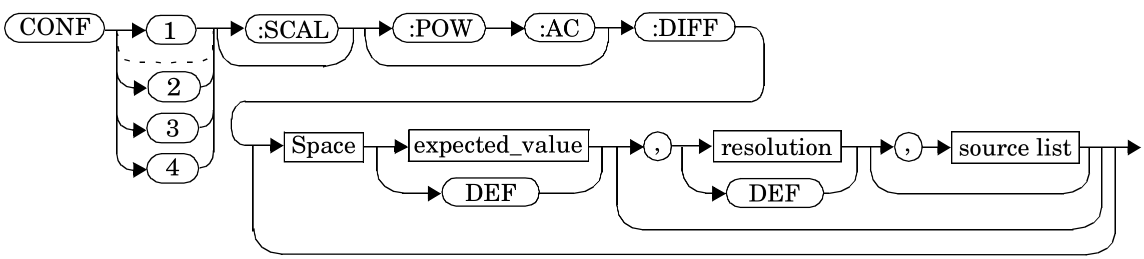
**Example**

`CONF2:REL -50DBM,3,(@1)` *This command configures the lower window / upper measurement to measure the relative power of channel A, using an expected power level of -50 dBm and a resolution setting of 3.*

# CONFigure[1]|2|3|4[:SCALar][:POWer:AC]:DIFFerence [<expected_value>[,<resolution>[,<source list>]]]

This command sets the measurement function and resolution of the specified window. It sets the measurement function to difference with relative mode off.

### Syntax




### Parameters

Refer to “Optional Parameters” on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The power meter ignores the numeric value entered in this parameter. Any value entered is treated like DEF. | sensor dependent<br>DEF[a] |
| resolution | A numeric value for the resolution. If unspecified the current resolution setting is used. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>DEF[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| source list | This channel list specifies between which channels the difference is calculated.<br>If unspecified and the current window setup is a difference measurement then this difference setup is used, otherwise it defaults to channel A-B (N1912A) or A-A (N1911A). | (@1),(@2)[c]<br>(@2),(@1)[c]<br>(@1),(@1)<br>(@2),(@2)[c] |

a. The mnemonic DEF means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying DEF leaves the parameter value unchanged.
b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.
c. N1912A only.

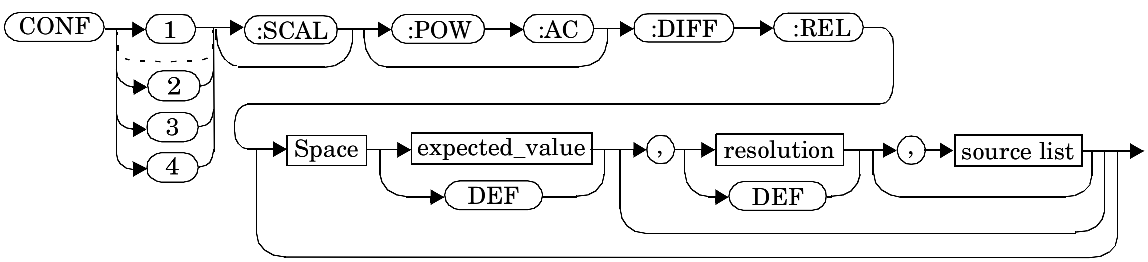
**Example**

| | |
|---|---|
| `CONF2:DIFF DEF,1,(@2),(@1)` | *This command configures the lower window/upper measurement to make a difference measurement of channel B - channel A, using the current sensor range and a resolution of 1 on both channels.* |

# CONFigure[1]|2|3|4[:SCALar][:POWer:AC]:DIFFerence: RELative [<expected_value>[,<resolution>[,<source list>]]]

This command sets the measurement function, range and resolution of the specified window. It sets the measurement function to difference with relative mode on. The relative value used is set by the `CALCulate:RELative:MAGNitude:AUTO` command.

### Syntax




### Parameters

Refer to “Optional Parameters” on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The power meter ignores the numeric value entered in this parameter. Any value entered is treated like DEF. | sensor dependent<br>DEF[a] |
| resolution | A numeric value for the resolution. If unspecified the current resolution setting is used. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>DEF[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| source list | This channel list specifies the channels used to calculate the difference.<br><br>If unspecified and the current window setup is a difference measurement then this difference setup is used, otherwise it defaults to channel A-B (N1912A) or A-A (N1911A). | (@1),(@2)[c]<br>(@2),(@1)[c]<br>(@1),(@1)<br>(@2),(@2)[c] |

a. The mnemonic `DEF` means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying `DEF` leaves the parameter value unchanged.
b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.
c. N1912A only.

**Example**

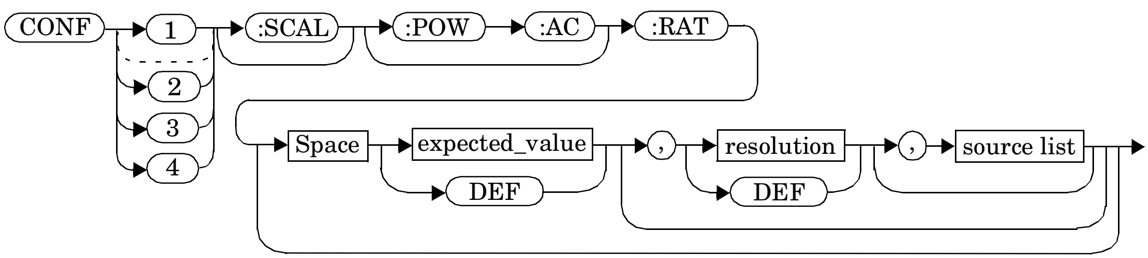
`CONF1:DIFF:REL DEF,1,(@1),(@2)`

*This command configures the upper window / upper measurement to make a difference measurement of channel A - channel B with relative mode on, using the current sensor range and a resolution of 1 on both channels.*

# CONFigure[1]|2|3|4[:SCALar][:POWer:AC]:RATio [<expected_value>[,<resolution>[,<source list>]]]

This command sets the measurement function, range and resolution of the specified window. It sets the measurement function to ratio with relative mode off.

**Syntax**




**Parameters**

Refer to “Optional Parameters” on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The power meter ignores the numeric value entered in this parameter. Any value entered is treated like DEF. | sensor dependent<br>DEF[a] |
| resolution | A numeric value for the resolution. If unspecified the current resolution setting is used. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>DEF[a] |
| source list | This channel list specifies the channels used to calculate the ratio. If unspecified and the current window setup is a ratio measurement then this ratio setup is used, otherwise it defaults to channel A/B (N1912A) or A/A (N1911A). | (@1),(@2)[c]<br>(@2),(@1)[c]<br>(@1),(@1)<br>(@2),(@2)[c] |

a. The mnemonic `DEF` means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying `DEF` leaves the parameter value unchanged.
b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.
c. N1912A only.

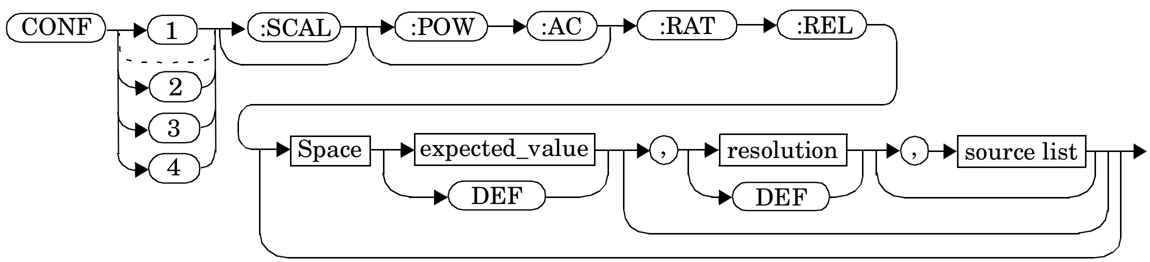
**Example**

| | |
|---|---|
| `CONF1:RAT DEF,4,(@1),(@2)` | *This command configures the upper window/upper measurement to make a ratio measurement of channel A over channel B, using the current sensor range and a resolution setting of 4 on both channels.* |

# CONFigure[1]|2|3|4[:SCALar][:POWer:AC]:RATio: RELative[<expected_value>[,<resolution>[,<source list>]]]

This command sets the measurement function, range and resolution of the specified window. It sets the measurement function to ratio with relative mode on. The relative value used is that set by the `CALCulate:RELative:MAGNitude:AUTO` command.

### Syntax




### Parameters

Refer to “Optional Parameters” on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The power meter ignores the numeric value entered in this parameter. Any value entered is treated like DEF. | sensor dependent<br>DEF[a] |
| resolution | A numeric value for the resolution. If unspecified the current resolution setting is used. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>DEF[a] |

**CONFigure[1]|2|3|4[:SCALar][:POWer:AC]:RATio:**
**RELative[<expected_value>[,<resolution>[,<source list>]]]**

| Item | Description/Default | Range of Values |
|---|---|---|
| source list | This channel list specifies the channels used to calculate the ratio.<br><br>If unspecified and the current window setup is a ratio measurement then this ratio setup is used, otherwise it defaults to channel A/B (N1912A) or A/A (N1911A). | (@1),(@2)[c]<br>(@2),(@1)[c]<br>(@1),(@1)<br>(@2),(@2)[c] |

a. The mnemonic DEF means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying DEF leaves the parameter value unchanged.
b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.
c. N1912A only.

**Example**

```
CONF1:RAT:REL DEF,1,(@1),(@2)
```

*This command configures the upper window / upper measurement to make a ratio measurement of channel A over channel B with relative mode on, using the current sensor range and a resolution setting of 1 on both channels.*

# FETCh[1]|2|3|4 Queries

The FETCh? queries set the specified window's measurement function. This can be set to either single channel, difference or ratio measurements, with relative mode either off or on. They then recalculate the measurement and place the result on the bus. The format of the result is set by FORM[:READ][:DATA]. Refer to Chapter 6, "FORMat Subsystem," for further information.

The query returns a measurement result when it is valid. The measurement result is invalid under the following conditions:

- When *RST is executed.
- Whenever a measurement is initiated.
- When any SENSe parameter, such as frequency, is changed.

If data is invalid, the FETCh? query is not completed until all data becomes valid. The exceptions to this are, if the power meter is in the idle state and the data is invalid, or the power meter has been reconfigured as defined above and no new measurement has been initiated. In such cases, the FETCh? routine generates the error -230, "Data corrupt or stale" and no result is returned. A common cause for this error is receiving a FETCh? after a *RST. If the expected value and resolution parameters are not the same as those that were used to collect the data, error -221, "Settings conflict" occurs.

**NOTE**

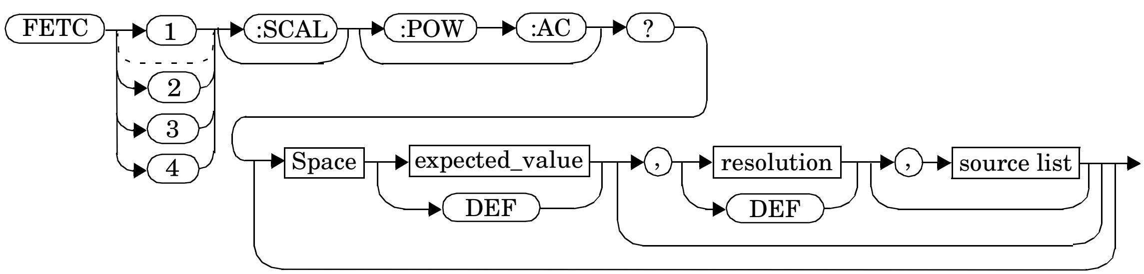
When TRIG:SOUR is INT1, INT2 or EXT and a new acquisition has been initiated (using the INIT command for example), FETCH? waits until the trigger takes place before executing. If trigger conditions are not satisfied—when the trigger level differs greatly from the signal level for example—this can give the impression that the power meter has hung.

To 'unlock' the power meter and adjust trigger settings, an execute clear (*CLS) must be performed.

# FETCh[1]|2|3|4[:SCALar][:POWer:AC]? [<expected_value>[,<resolution>[,<source list>]]]

This command sets the specified window's measurement function to single channel with relative mode off, recalculates the measurement and places the result on the bus. The result is a power based measurement and is expressed in the units defined by `UNIT[1]|2|3|4:POWer`.

### Syntax




### Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The expected power level parameter can be set to `DEF` or a numeric value. If a value is entered it should correspond to that set by `CONFigure` otherwise an error occurs. The units of measurement are dBm and W. The default units are defined by `UNIT:POWer`. | sensor dependent<br>`DEF`[a] |
| resolution | A numeric value for the resolution. If it is unspecified the current resolution setting is used. If a value is entered it should correspond to the current resolution setting otherwise an error occurs. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>`DEF`[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| source list | The channel which the command is implemented on. If unspecified the current window setup is used. However, on the N1912A, if the window shows a ratio or difference measurement, the upper window defaults to channel A and the lower window to channel B. | (@1)<br>(@2) (N1912A only) |

a. The mnemonic `DEF` means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying `DEF` leaves the parameter value unchanged.

b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.

### Example

| | |
|---|---|
| `FETC2:POW:AC?` | *This command queries the lower window/upper measurement result.* |

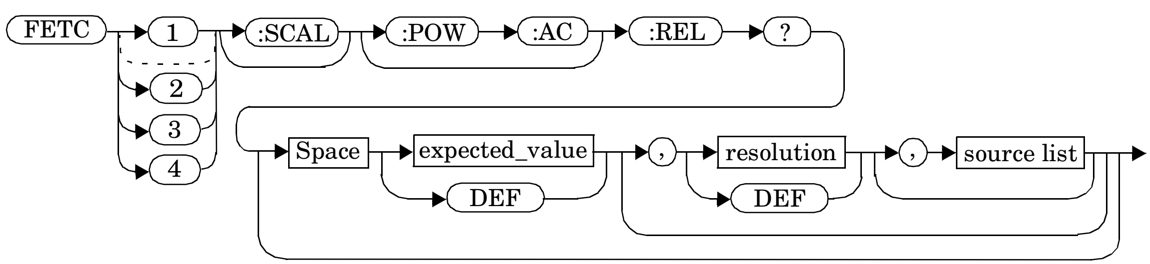
### Error Messages

- If the last measurement is not valid error -230, “Data corrupt or stale” occurs. A measurement is valid after it has been initiated. It becomes invalid when either a reset occurs or any measurement parameter, for example frequency, is changed.
- If the expected_value and resolution parameters are not the same as the current expected value and resolution setting on the specified window, error -221, “Settings conflict” occurs.

# FETCh[1]|2|3|4[:SCALar][:POWer:AC]:RELative? [<expected_value>[,<resolution>[,<source list>]]]

This command sets the specified window's measurement function to single channel with relative mode on, recalculates the measurement and places the results on the bus. The result is a ratio based measurement and is expressed in the units defined by `UNIT[1]|2|3|4:POWer:RATio`. The relative value used is that set by the `CALCulate:RELative:MAGNitude:AUTO` command.

### Syntax




### Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The expected power level parameter can be set to `DEF` or a numeric value. If a value is entered it should correspond to that set by `CONFigure` otherwise an error occurs. The units of measurement are dBm and W. The default units are defined by `UNIT:POWer`. | sensor dependent<br>`DEF`[a] |
| resolution | A numeric value for the resolution. If it is unspecified the current resolution setting is used. If a value is entered it should correspond to the current resolution setting otherwise an error occurs. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>`DEF`[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| source list | The channel which the command is implemented on. If unspecified the current window setup is used. However, on the N1912A, if the window shows a ratio or difference measurement, the upper window defaults to channel A and the lower window to channel B. | (@1)<br>(@2) (N1912A only) |

a. The mnemonic `DEF` means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying `DEF` leaves the parameter value unchanged.
b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.

### Example

`FETC1:REL? DEF,2,(@2)` *This command queries the upper window/upper measurement relative measurement of channel B, using the current sensor range and a resolution setting of 2.*

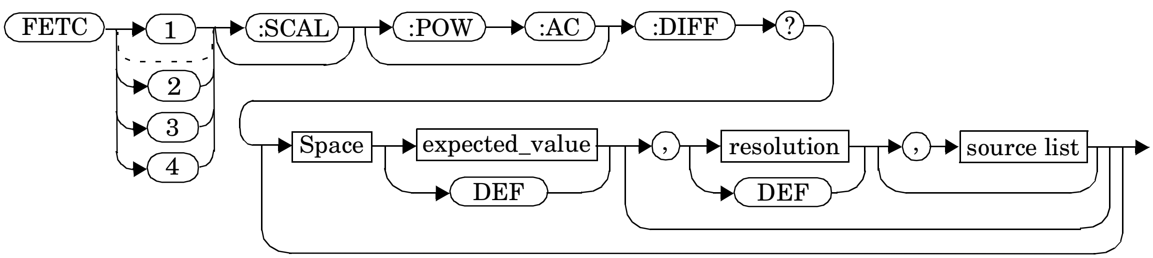
### Error Messages

- If the last measurement is not valid error -230, “Data corrupt or stale” occurs. A measurement is valid after it has been initiated. It becomes invalid when either a reset occurs or any measurement parameter, for example frequency, is changed.
- If the expected_value and resolution parameters are not the same as the current expected value and resolution settings on the specified window, error -221, “Settings conflict” occurs.

# FETCh[1]|2|3|4[:SCALar][:POWer:AC]:DIFFerence? [<expected_value>[,<resolution>[,<source list>]]]

This command sets the specified window's measurement function to power difference with relative mode off, recalculates the measurement and places the results on the bus. The result is a power based measurement and is expressed in the units defined by `UNIT[1]|2|3|4:POWer`.

### Syntax




### Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The expected power level parameter can be set to `DEF` or a numeric value. If a value is entered it should correspond to that set by `CONFigure` otherwise an error occurs.The units of measurement are dBm and W. The default units are defined by `UNIT:POWer`. | sensor dependent<br>`DEF`[a] |
| resolution | A numeric value for the resolution. If it is unspecified the current resolution setting is used. If a value is entered it should correspond to the current resolution setting otherwise an error occurs. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>`DEF`[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| source list | This channel list specifies the channels used to calculate the difference. If unspecified and the current window setup is a difference measurement then this difference setup is used, otherwise it defaults to channel A-B (N1912A) or A-A (N1911A). | (@1),(@2)[c]<br>(@2),(@1)[c]<br>(@1),(@1)<br>(@2),(@2)[c] |

a. The mnemonic DEF means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying DEF leaves the parameter value unchanged.
b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.
c. N1912A only.

### Example

| | |
|---|---|
| `FETC2:DIFF?` | *This command queries the difference measurement on the lower window/lower measurement* |

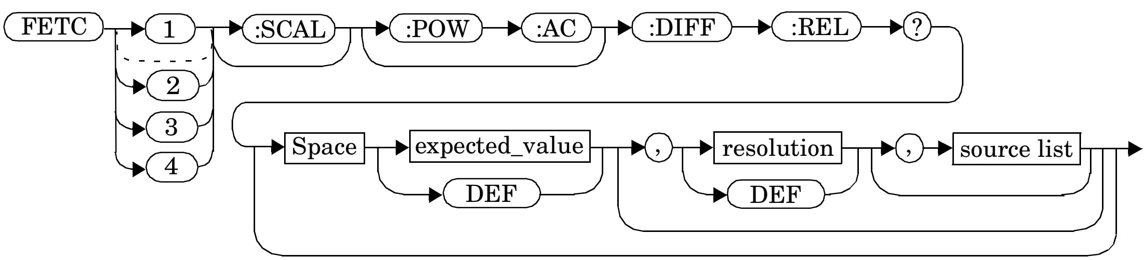
### Error Messages

- If the last measurement on either channel is not valid error -230, “Data corrupt or stale” occurs. A measurement is valid after it has been initiated. It becomes invalid when either a reset occurs or any measurement parameter, for example frequency, is changed.
- If the expected_value and resolution parameters are not the same as the current expected value and resolution settings on the specified window, error -221, “Settings conflict” occurs.

# FETCh[1]|2|3|4[:SCALar][:POWer:AC]:DIFFerence: RELative? [<expected_value>[,<resolution>[,<source list>]]]

This command sets the specified window's measurement function to power difference with relative mode on, recalculates the measurement and places the results on the bus. The result is a ratio based measurement and is expressed in the units defined by `UNIT[1]|2|3|4:POWer:RATio`. The relative value used is that set by the `CALCulate:RELative:MAGNitude:AUTO` command.

## Syntax




## Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The expected power level parameter can be set to `DEF` or a numeric value. If a value is entered it should correspond to that set by `CONFigure` otherwise an error occurs.The units of measurement are dBm and W. The default units are defined by `UNIT:POWer`. | sensor dependent<br>`DEF`[a] |
| resolution | A numeric value for the resolution. If it is unspecified the current resolution setting is used. If a value is entered it should correspond to the current resolution setting otherwise an error occurs. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>`DEF`[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| source list | This channel list specifies the channels used to calculate the difference.<br>If unspecified and the current window setup is a difference measurement then this difference setup is used, otherwise it defaults to channel A-B (N1912A) or A-A (N1911A). | (@1),(@2)[c]<br>(@2),(@1)[c]<br>(@1),(@1)<br>(@2),(@2)[c] |

a. The mnemonic DEF means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying DEF leaves the parameter value unchanged.
b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.
c. N1912A only.

### Example

```
FETC1:DIFF:REL? DEF,3,(@2),(@1)
```

*This command queries the upper window/upper measurement relative difference measurement of channel B - channel A, using the current sensor range and a resolution setting of 3 on both channels.*

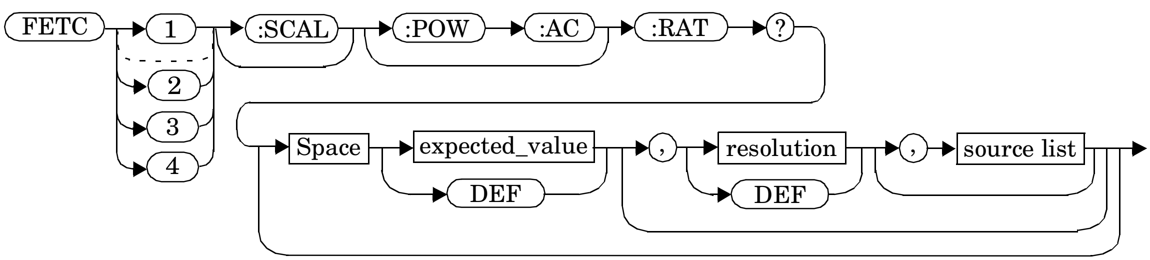
### Error Messages

- If the last measurement on either channel is not valid error -230, “Data corrupt or stale” occurs. A measurement is valid after it has been initiated. It becomes invalid when either a reset occurs or any measurement parameter, for example frequency, is changed.
- If the expected_value and resolution parameters are not the same as the current expected value and resolution settings on the specified window, error -221, “Settings conflict” occurs.

# FETCh[1]|2|3|4[:SCALar][:POWer:AC]:RATio? [<expected_value>[,<resolution>[,<source list>]]]

This command sets the specified window's measurement function to power ratio with relative mode off, recalculates the measurement and places the results on the bus. The result is a ratio based measurement and is expressed in the units defined by `UNIT[1]|2|3|4:POWer:RATio`.

### Syntax




### Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The expected power level parameter can be set to `DEF` or a numeric value. If a value is entered it should correspond to that set by `CONFigure` otherwise an error occurs.The units of measurement are dBm and W. The default units are defined by `UNIT:POWer`. | sensor dependent<br>`DEF`[a] |
| resolution | A numeric value for the resolution. If it is unspecified the current resolution setting is used. If a value is entered it should correspond to the current resolution setting otherwise an error occurs. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>`DEF`[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| source list | This channel list specifies the channels used to calculate the difference.<br>If unspecified and the current window setup is a difference measurement then this difference setup is used, otherwise it defaults to channel A-B (N1912A) or A-A (N1911A). | (@1),(@2)[c]<br>(@2),(@1)[c]<br>(@1),(@1)<br>(@2),(@2)[c] |

a. The mnemonic DEF means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying DEF leaves the parameter value unchanged.
b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.
c. N1912A only.

**Example**

| | |
|---|---|
| `FETC2:RAT? DEF,1,(@1),(@2)` | *This command queries the lower window / upper measurement ratio measurement of channel A over channel B, using the current sensor range and a resolution of 1 on both channels.* |

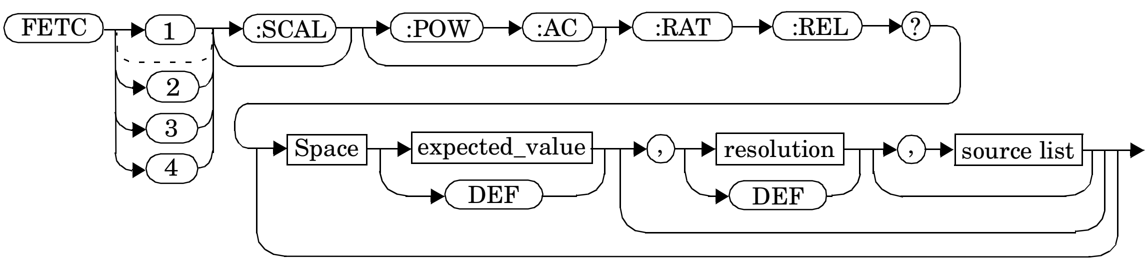
**Error Messages**

- If the last measurement on either channel is not valid error -230, “Data corrupt or stale” occurs. A measurement is valid after it has been initiated. It becomes invalid when either a reset occurs or any measurement parameter, for example frequency, is changed.
- If the expected_value and resolution parameters are not the same as the current expected value and resolution settings on the specified window, error -221, “Settings conflict” occurs.

# FETCh[1]|2|3|4[:SCALar][:POWer:AC]:RATio:RELative? [<expected_value>[,<resolution>[,<source list>]]]

This command sets the specified window's measurement function to power ratio with relative mode on, recalculates the measurement and places the results on the bus. The result is a ratio based measurement and is expressed in the units defined by `UNIT[1]|2|3|4:POWer:RATio`. The relative value used is that set by the `CALCulate:RELative:MAGNitude:AUTO` command.

### Syntax




### Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The expected power level parameter can be set to `DEF` or a numeric value. If a value is entered it should correspond to that set by `CONFigure` otherwise an error occurs.The units of measurement are dBm and W. The default units are defined by `UNIT:POWer`. | sensor dependent<br>`DEF`[a] |
| resolution | A numeric value for the resolution. If it is unspecified the current resolution setting is used. If a value is entered it should correspond to the current resolution setting otherwise an error occurs. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>`DEF`[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| source list | This channel list specifies the channels used to calculate the difference.<br>If unspecified and the current window setup is a difference measurement then this difference setup is used, otherwise it defaults to channel A-B (N1912A) or A-A (N1911A). | (@1),(@2)[c]<br>(@2),(@1)[c]<br>(@1),(@1)<br>(@2),(@2)[c] |

a. The mnemonic DEF means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying DEF leaves the parameter value unchanged.
b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.
c. N1912A only.

### Example

| | |
|---|---|
| `FETC:RAT:REL?` | *This command queries the relative ratio measurement on the upper window / upper measurement.* |

### Error Messages

- If the last measurement on either channel is not valid error -230, “Data corrupt or stale” occurs. A measurement is valid after it has been initiated. It becomes invalid when either a reset occurs or any measurement parameter, for example frequency, is changed.
- If the expected_value and resolution parameters are not the same as the current expected value and resolution settings on the specified window, error -221, “Settings conflict” occurs.

# READ[1]|2|3|4 Commands

The READ? commands are most commonly used with the CONFigure command to cause a new power measurement to be taken and the result returned to the output buffer. The format of the result is set by FORM[:READ][:DATA]. Refer to Chapter 6, "FORMat Subsystem," on page 213 for further information.

- For the N1911A the READ? query is equivalent to:

```
ABORt
INITiate
FETCh?
```

- For the N1912A carrying out a single channel measurement the READ? queries are equivalent to:

```
ABORt1
INITiate1
FETCh1?
```

*or*

```
ABORt2
INITiate2
FETCh2?
```

- For the N1912A carrying out a difference measurement the READ:DIFFerence? queries are equivalent to:

ABORt1
and
ABORt2
INITiate1
INITiate2
FETCh:DIFFerence?

- For the N1912A carrying out a ratio measurement the READ:RATio? queries are equivalent to:

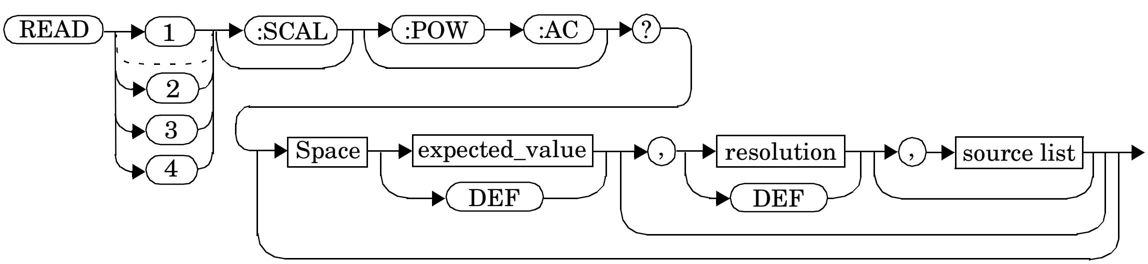
```
ABORt1
ABORt2
INITiate1
INITiate2
FETCh:RATio?
```

# READ[1]|2|3|4[:SCALar][:POWer:AC]? [<expected_value>[,<resolution>[,<source list>]]]

This command sets the specified window's measurement function to single channel with relative mode off, aborts then initiates the specified channel, calculates the measurement result and places the result on the bus. The result is a power based measurement and is expressed in the units defined by `UNIT[1]|2|3|4:POWer`.

**NOTE** `INITiate:CONTinuous` must be set to OFF, otherwise error -213, "INIT ignored" occurs. If `TRIGger:SOURce` is set to `BUS`, error -214, "Trigger deadlock" occurs.

### Syntax




### Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The expected power level parameter can be set to `DEF` or a numeric value. If a value is entered it should correspond to that set by `CONFigure` otherwise an error occurs. | sensor dependent<br>`DEF`[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| resolution | A numeric value for the resolution. If it is unspecified the current resolution setting is used. If a value is entered it should correspond to the current resolution setting otherwise an error occurs. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>DEF[a] |
| source list | The channel which the command is implemented on.<br>If unspecified the current window setup is used. However, on the N1912A, if the window shows a ratio or difference measurement, the upper window defaults to channel A and the lower window to channel B. | (@1)<br>(@2) (N1912A only) |

a. The mnemonic DEF means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying DEF leaves the parameter value unchanged.
b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.

### Example

| | |
|---|---|
| `READ2:POW:AC?` | *This command queries the lower window / upper measurement.* |

### Error Messages

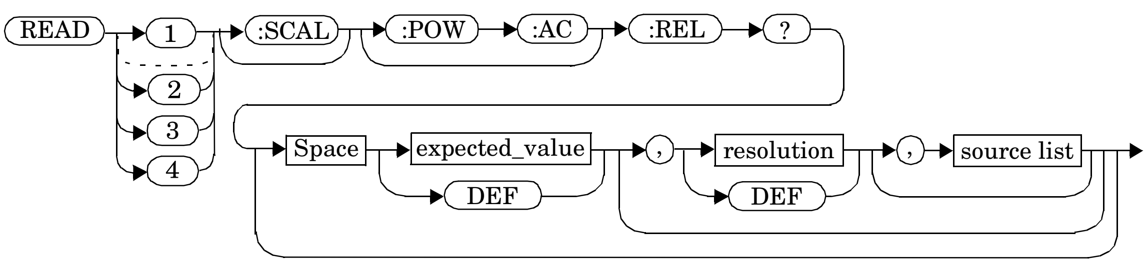
- `INITiate:CONTinuous` must be set to `OFF`, otherwise error -213, “INIT ignored” occurs.
- If `TRIGger:SOURce` is set to `BUS` or `HOLD`, error -214, “Trigger deadlock” occurs.
- If the expected value and resolution parameters are not the same as the current expected value and resolution settings on the specified window, error -221, “Settings conflict” occurs.

# READ[1]|2|3|4[:SCALar][:POWer:AC]:RELative? [<expected_value>[,<resolution>[,<source list>]]]

This command sets the specified window's measurement function to single channel with relative mode on, aborts then initiates the specified channel, calculates the measurement result and places the result on the bus. The result is a ratio based measurement and is expressed in the units defined by `UNIT[1]|2|3|4:POWer:RATio`. The relative value used is that set by the `CALCulate:RELative:MAGNitude:AUTO` command.

**NOTE** `INITiate:CONTinuous` must be set to OFF, otherwise error -213, "INIT ignored" occurs. If `TRIGger:SOURce` is set to `BUS`, error -214, "Trigger deadlock" occurs.

### Syntax




### Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The expected power level parameter can be set to `DEF` or a numeric value. If a value is entered it should correspond to that set by `CONFigure` otherwise an error occurs. | sensor dependent<br>`DEF`[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| resolution | A numeric value for the resolution. If it is unspecified the current resolution setting is used. If a value is entered it should correspond to the current resolution setting otherwise an error occurs. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>DEF[a] |
| source list | The channel which the command is implemented on.<br>If unspecified the current window setup is used. However, on the N1912A, if the window shows a ratio or difference measurement, the upper window defaults to channel A and the lower window to channel B. | (@1)<br>(@2) (N1912A only) |

a. The mnemonic DEF means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying DEF leaves the parameter value unchanged.

b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.

## Example

| | |
|---|---|
| `READ1:REL? DEF,1,(@2)` | *This command queries the upper window/upper measurement relative measurement of channel B, using the current sensor range and a resolution of 1.* |

## Error Messages

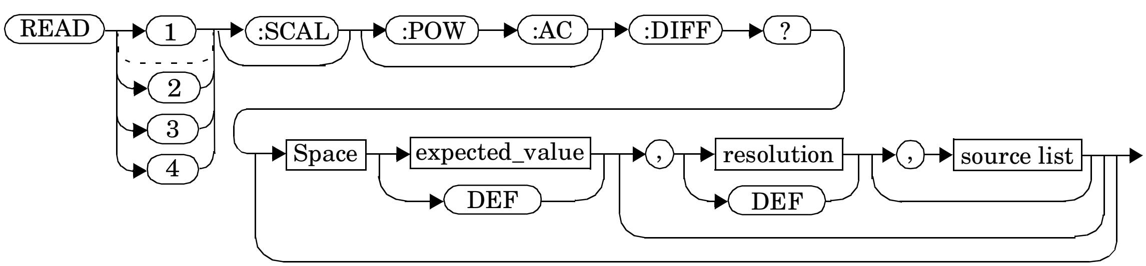
- `INITiate:CONTinuous` must be set to `OFF`, otherwise error -213, “INIT ignored” occurs.
- If `TRIGger:SOURce` is set to `BUS` or `HOLD`, error -214, “Trigger deadlock” occurs.
- If the expected value and resolution parameters are not the same as the current expected value and resolution settings on the specified window, error -221, “Settings conflict” occurs.

# READ[1]|2|3|4[:SCALar][:POWer:AC]:DIFFerence? [<expected_value>[,<resolution>[,<source list>]]]

This command sets the specified window's measurement function to difference mode with relative mode off, aborts then initiates both channel A and B, calculates the difference measurement result and places the result on the bus. The result is a power based measurement and is expressed in the units defined by `UNIT[1]|2|3|4:POWer`.

**NOTE** `INITiate:CONTinuous` must be set to `OFF` on both channels, otherwise error -213, "INIT ignored" occurs. If `TRIGger:SOURce` is set to `BUS` on either channel, error -214, "Trigger deadlock" occurs.

### Syntax




### Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The power meter ignores the numeric value entered in this parameter. Any value entered is treated like DEF. | sensor dependent<br>`DEF`[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| resolution | A numeric value for the resolution. If it is unspecified the current resolution setting is used. If a value is entered it should correspond to the current resolution setting otherwise an error occurs. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>DEF[a] |
| source list | This channel list specifies the channels used to calculate the difference.<br>If unspecified and the current window setup is a difference measurement then this difference setup is used, otherwise it defaults to channel A-B (N1912A) or A-A (N1911A). | (@1),(@2)[c]<br>(@2),(@1)[c]<br>(@1),(@1)<br>(@2),(@2)[c] |

a. The mnemonic DEF means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying DEF leaves the parameter value unchanged.
b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.
c. N1912A only.

### Example

| | |
|---|---|
| `READ2:DIFF?` | *This command queries difference measurement on the lower window/upper measurement.* |

### Error Messages

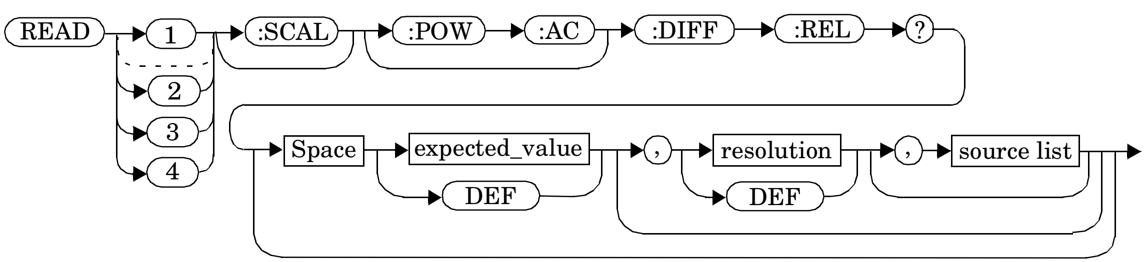
- `INITiate:CONTinuous` must be set to OFF on both channels, otherwise error -213, “INIT ignored” occurs.
- If `TRIGger:SOURce` is set to `BUS` or `HOLD` on either channel, error -214, “Trigger deadlock” occurs.
- If the resolution parameter is not the same as the current resolution setting on the specified window, error -221, “Settings conflict” occurs.

# READ[1]|2|3|4[:SCALar][:POWer:AC]:DIFFerence: RELative? [<expected_value>[,<resolution>[,<source list>]]]

This command sets the specified window's measurement function to difference mode with relative mode on, aborts then initiates both channel A and B, calculates the difference measurement result and places the result on the bus. The result is a ratio based measurement and is expressed in the units defined by `UNIT[1]|2|3|4:POWer:RATio`. The relative value used is that set by the `CALCulate:RELative:MAGNitude:AUTO` command.

**NOTE** `INITiate:CONTinuous` must be set to OFF on both channels, otherwise error -213, "INIT ignored" occurs. If `TRIGger:SOURce` is set to `BUS` on either channel, error -214, "Trigger deadlock" occurs.

### Syntax




### Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The power meter ignores the numeric value entered in this parameter. Any value entered is treated like DEF. | sensor dependent<br>`DEF`[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| resolution | A numeric value for the resolution. If it is unspecified the current resolution setting is used. If a value is entered it should correspond to the current resolution setting otherwise an error occurs. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>DEF[a] |
| source list | This channel list specifies the channels used to calculate the difference.<br>If unspecified and the current window setup is a difference measurement then this difference setup is used, otherwise it defaults to channel A-B (N1912A) or A-A (N1911A). | (@1),(@2)[c]<br>(@2),(@1)[c]<br>(@1),(@1)<br>(@2),(@2)[c] |

a. The mnemonic DEF means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying DEF leaves the parameter value unchanged.
b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.
c. N1912A only.

## Example

```
READ1:DIFF:REL? DEF,4,(@2),(@1)
```

*This command queries the upper window / upper measurement relative difference measurement of channel B - channel A, using the current sensor range and a resolution setting of 4 on both channels.*

## Error Messages

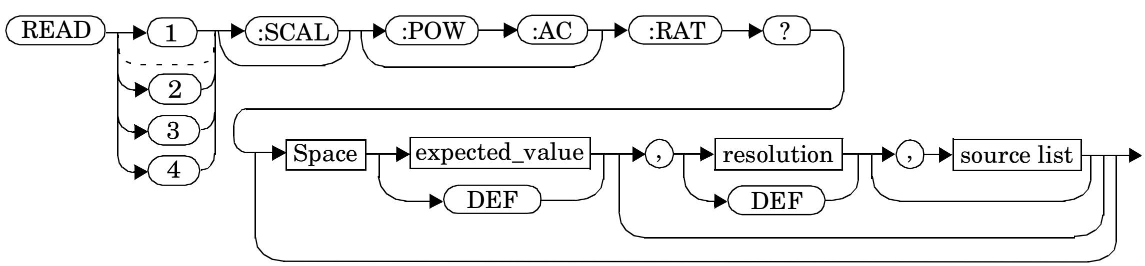
- `INITiate:CONTinuous` must be set to OFF on both channels, otherwise error -213, “INIT ignored” occurs.
- If `TRIGger:SOURce` is set to `BUS` or `HOLD` on either channel, error -214, “Trigger deadlock” occurs.
- If the resolution parameter is not the same as the current resolution setting on the specified window, error -221, “Settings conflict” occurs.

# READ[1]|2|3|4[:SCALar][:POWer:AC]:RATio? [<expected_value>[,<resolution>[,<source list>]]]

This command sets the specified window's measurement function to ratio mode with relative mode off, aborts then initiates both channel A and B, calculates the ratio measurement result and places the result on the bus. The result is a ratio based measurement and is expressed in the units defined by `UNIT[1]|2|3|4:POWer:RATio`.

**NOTE** `INITiate:CONTinuous` must be set to OFF on both channels, otherwise error -213, "INIT ignored" occurs. If `TRIGger:SOURce` is set to `BUS` on either channel, error -214, "Trigger deadlock" occurs.

### Syntax




### Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The power meter ignores the numeric value entered in this parameter. Any value entered is treated like DEF. | sensor dependent<br>`DEF`[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| resolution | A numeric value for the resolution. If it is unspecified the current resolution setting is used. If a value is entered it should correspond to the current resolution setting otherwise an error occurs. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>DEF[a] |
| source list | This channel list specifies the channels used to calculate the difference.<br>If unspecified and the current window setup is a difference measurement then this difference setup is used, otherwise it defaults to channel A-B (N1912A) or A-A (N1911A). | (@1),(@2)[c]<br>(@2),(@1)[c]<br>(@1),(@1)<br>(@2),(@2)[c] |

a. The mnemonic DEF means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying DEF leaves the parameter value unchanged.
b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.
c. N1912A only.

## Example

```
READ2:RAT? DEF,1,(@1),(@2)
```

*This command queries the lower window / upper measurement ratio measurement of channel A over channel B, using the current sensor range and a resolution of 1 on both channels.*

## Error Messages

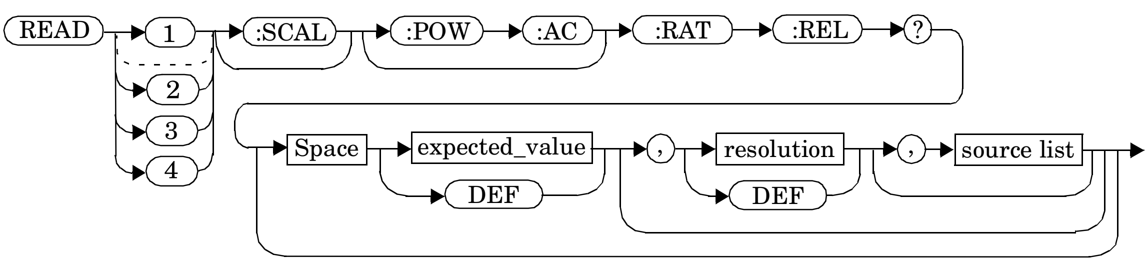
- `INITiate:CONTinuous` must be set to OFF on both channels, otherwise error -213, “INIT ignored” occurs.
- If `TRIGger:SOURce` is set to `BUS` or `HOLD` on either channel, error -214, “Trigger deadlock” occurs.
- If the resolution parameter is not the same as the current resolution setting on the specified window, error -221, “Settings conflict” occurs.

# READ[1]|2|3|4[:SCALar][:POWer:AC]:RATio:RELative? [<expected_value>[,<resolution>[,<source list>]]]

This command sets the specified window's measurement function to ratio mode with relative mode on, aborts then initiates both channel A and B, calculates the ratio measurement result using the new sensor data and places the result on the bus. The result is a ratio based measurement and is expressed in the units defined by `UNIT[1]|2|3|4:POWer:RATio`. The relative value used is that set by the `CALCulate:RELative:MAGNitude:AUTO` command.

**NOTE** `INITiate:CONTinuous` must be set to OFF on both channels, otherwise error -213, "INIT ignored" occurs. If `TRIGger:SOURce` is set to `BUS` on either channel, error -214, "Trigger deadlock" occurs.

### Syntax




### Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The power meter ignores the numeric value entered in this parameter. Any value entered is treated like DEF. | sensor dependent<br>`DEF`[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| resolution | A numeric value for the resolution. If it is unspecified the current resolution setting is used. If a value is entered it should correspond to the current resolution setting otherwise an error occurs. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>DEF[a] |
| source list | This channel list specifies the channels used to calculate the difference.<br>If unspecified and the current window setup is a difference measurement then this difference setup is used, otherwise it defaults to channel A-B (N1912A) or A-A (N1911A). | (@1),(@2)[c]<br>(@2),(@1)[c]<br>(@1),(@1)<br>(@2),(@2)[c] |

a. The mnemonic DEF means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying DEF leaves the parameter value unchanged.
b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.
c. N1912A only.

## Example

| | |
|---|---|
| `READ:RAT:REL?` | *This command queries the relative ratio measurement on the upper window / upper measurement.* |

## Error Messages

- `INITiate:CONTinuous` must be set to OFF on both channels, otherwise error -213, “INIT ignored” occurs.
- If `TRIGger:SOURce` is set to `BUS` or `HOLD` on either channel, error -214, “Trigger deadlock” occurs.
- If the resolution parameter is not the same as the current resolution setting on the specified window, error -221, “Settings conflict” occurs.

# MEASure[1]|2|3|4 Commands

The `MEASure?` commands configure the power meter to perform a power measurement with the given measurement function, relative mode setting, range and resolution then makes the measurement. The format of the result is set by `FORM[:READ][:DATA]`. Refer to Chapter 6, “FORMat Subsystem,” on page 213 for further information.

`MEASure?` is a compound command which is equivalent to:

- For the N1911A the `MEASure?` query is equivalent to:

```
ABORt
CONFigure
READ?
```

- For the N1912A carrying out a single channel measurement the `MEASure?` queries are equivalent to:

```
ABORt1
CONFigure
READ1?
```

*or*

```
ABORt2
CONFigure
READ2?
```

- For the N1912A carrying out a difference measurement the `READ:DIFFerence?` queries are equivalent to:

```
ABORt1
ABORt2
CONFigure:DIFFerence
READ:DIFFerence?
```

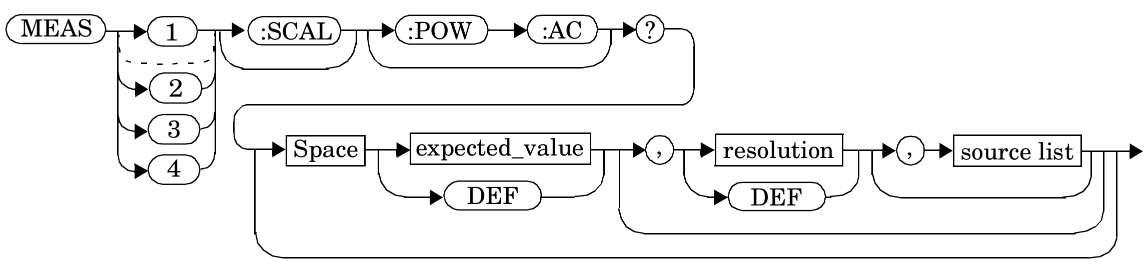
- For the N1912A carrying out a ratio measurement the `READ:RATio?` queries are equivalent to:

```
ABORt1
ABORt2
CONFigure:RATio
READ:RATio?
```

# MEASure[1]|2|3|4[:SCALar][:POWer:AC]? [<expected_value>[,<resolution>[,<source list>]]]

This command sets the specified window's measurement function to single channel with relative mode off, aborts, configures the window then initiates channel A or B, calculates the measurement result and places the result on the bus.

### Syntax




### Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | A numeric value for the expected power level. The units of measurement are dBm and W. The default units are defined by `UNIT:POWer`. | sensor dependent<br>`DEF`[a] |
| resolution | A numeric value for the resolution. If unspecified the current resolution setting is used. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>`DEF`[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| source list | The channel which the command is implemented on.<br>If unspecified the current window setup is used. However, on the N1912A, if the window shows a ratio or difference measurement, the upper window defaults to channel A and the lower window to channel B. | (@1)<br>(@2) (N1912A only) |

a. The mnemonic `DEF` means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying `DEF` leaves the parameter value unchanged.

b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.

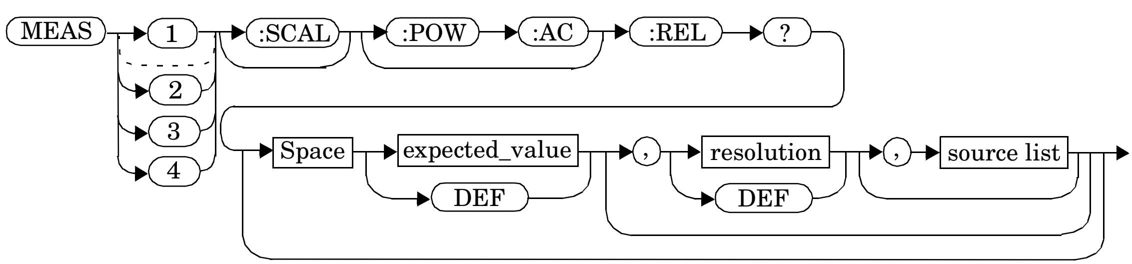
### Example

| | |
|---|---|
| `MEAS2:POW:AC? -70DBM,1,(@1)` | *This command queries the lower window / upper measurement of channel A, using an expected power level of -70 dBm and a resolution setting of 1.* |

# MEASure[1]|2|3|4[:SCALar][:POWer:AC]:RELative? [<expected_value>[,<resolution>[,<source list>]]]

This command sets the specified window's measurement function to single channel with relative mode on, aborts, configures then initiates the specified channel, calculates the measurement result and places the result on the bus. The result is a ratio based measurement and is expressed in the units defined by `UNIT[1]|2|3|4:POWer:RATio`. The relative value used is that set by the `CALCulate:RELative:MAGNitude:AUTO` command.

### Syntax




### Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | A numeric value for the expected power level. The units of measurement are dBm and W. The default units are defined by `UNIT:POWer`. | sensor dependent<br>`DEF`[a] |
| resolution | A numeric value for the resolution. If unspecified the current resolution setting is used. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>`DEF`[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| source list | The channel which the command is implemented on.<br>If unspecified the current window setup is used. However, on the N1912A, if the window shows a ratio or difference measurement, the upper window defaults to channel A and the lower window to channel B. | (@1)<br>(@2) (N1912A only) |

a. The mnemonic `DEF` means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying `DEF` leaves the parameter value unchanged.

b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.

## Example

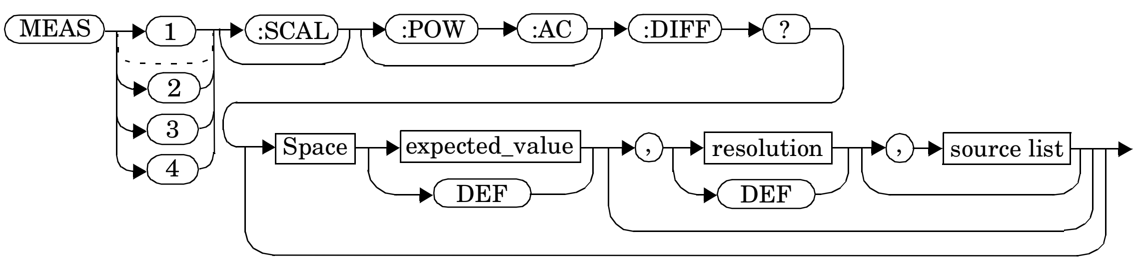
| | |
|---|---|
| `MEAS1:REL? -50DBM,2,(@2)` | *This command queries the upper window/upper measurement relative measurement of channel B, using an expected power level of -50 dBm and a resolution setting of 2.* |

# MEASure[1]|2|3|4[:SCALar][:POWer:AC]:DIFFerence? [<expected_value>[,<resolution>[,<source list>]]]

This command applies to the N1912A power meter only, as it needs two measurement channels to make sense.

This command sets the specified window's measurement function to difference mode with relative mode off, aborts, configures then initiates both channel A and B, calculates the difference measurement result and places the result on the bus. The result is a power based measurement and is expressed in the units defined by `UNIT[1]|2|3|4:POWer`.

### Syntax




### Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The power meter ignores the numeric value entered in this parameter. Any value entered is treated like DEF. | sensor dependent<br>DEF[a] |
| resolution | A numeric value for the resolution. If unspecified the current resolution setting is used. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>DEF[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| source list | This channel list specifies the channels used to calculate the difference.<br>If unspecified and the current window setup is a difference measurement then this difference setup is used, otherwise it defaults to channel A-B (N1912A) or A-A (N1911A). | (@1),(@2)[c]<br>(@2),(@1)[c]<br>(@1),(@1)<br>(@2),(@2)[c] |

a. The mnemonic `DEF` means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying `DEF` leaves the parameter value unchanged.
b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.
c. N1912A only.

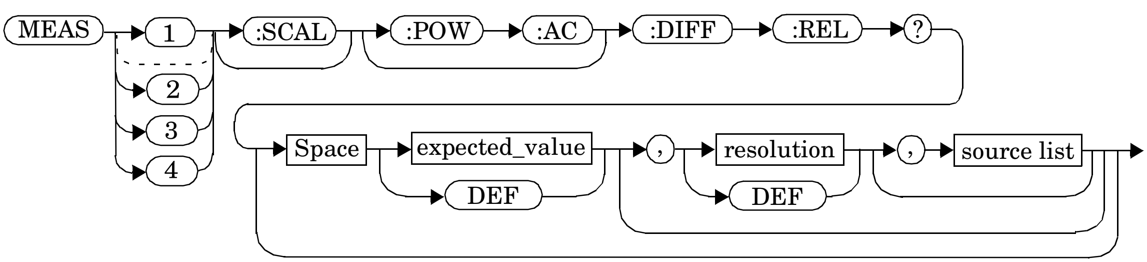
### Example

| | |
|---|---|
| `MEAS2:DIFF?` | *This command queries the difference measurement on the lower window / upper measurement.* |

# MEASure[1]|2|3|4[:SCALar][:POWer:AC]:DIFFerence: RELative? [<expected_value>[,<resolution>[,<source list>]]]

This command sets the specified window's measurement function to difference mode with relative mode on, aborts, configures then initiates both channel A and B, calculates the difference measurement result and places the result on the bus. The result is a ratio based measurement and is expressed in the units defined by `UNIT[1]|2|3|4:POWer:RATio`. The relative value used is that set by the `CALCulate:RELative:MAGNitude:AUTO` command.

## Syntax




## Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The power meter ignores the numeric value entered in this parameter. Any value entered is treated like DEF. | sensor dependent<br>DEF[a] |
| resolution | A numeric value for the resolution. If unspecified the current resolution setting is used. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>DEF[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| source list | This channel list specifies the channels used to calculate the difference.<br>If unspecified and the current window setup is a difference measurement then this difference setup is used, otherwise it defaults to channel A-B (N1912A) or A-A (N1911A). | (@1),(@2)[c]<br>(@2),(@1)[c]<br>(@1),(@1)<br>(@2),(@2)[c] |

a. The mnemonic DEF means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying DEF leaves the parameter value unchanged.
b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.
c. N1912A only.

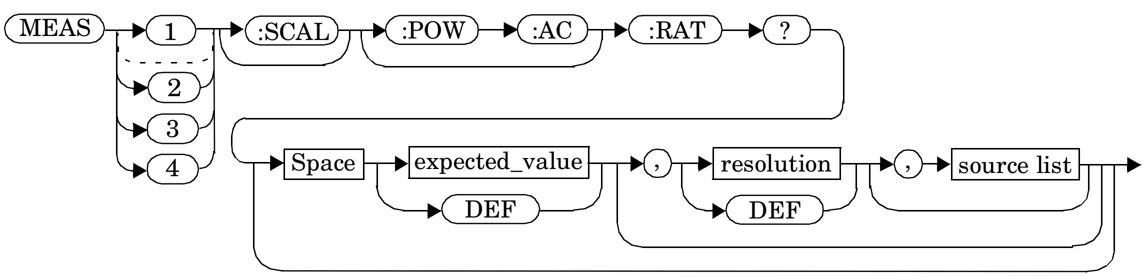
### Example

| | |
|---|---|
| `MEAS1:DIFF:REL? DEF,3,(@2),(@1)` | *This command queries the upper window/upper measurement relative difference measurement of channel B - channel A, using the current sensor range and a resolution setting of 3 on both channels.* |

# MEASure[1]|2|3|4[:SCALar][:POWer:AC]:RATio? [<expected_value>[,<resolution>[,<source list>]]]

This command sets the specified window's measurement function to ratio mode with relative mode off, aborts, configures then initiates both channel A and B, calculates the ratio measurement result and places the result on the bus. The result is a ratio based measurement and is expressed in the units defined by `UNIT[1]|2|3|4:POWer:RATio`.

**Syntax**




**Parameters**

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The power meter ignores the numeric value entered in this parameter. Any value entered is treated like DEF. | sensor dependent<br>DEF[a] |
| resolution | A numeric value for the resolution. If unspecified the current resolution setting is used. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>DEF[a] |

| Item | Description/Default | Range of Values |
| --- | --- | --- |
| source list | This channel list specifies the channels used to calculate the difference.<br>If unspecified and the current window setup is a difference measurement then this difference setup is used, otherwise it defaults to channel A-B (N1912A) or A-A (N1911A). | (@1),(@2)[c]<br>(@2),(@1)[c]<br>(@1),(@1)<br>(@2),(@2)[c] |

a. The mnemonic DEF means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying DEF leaves the parameter value unchanged.
b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.
c. N1912A only.

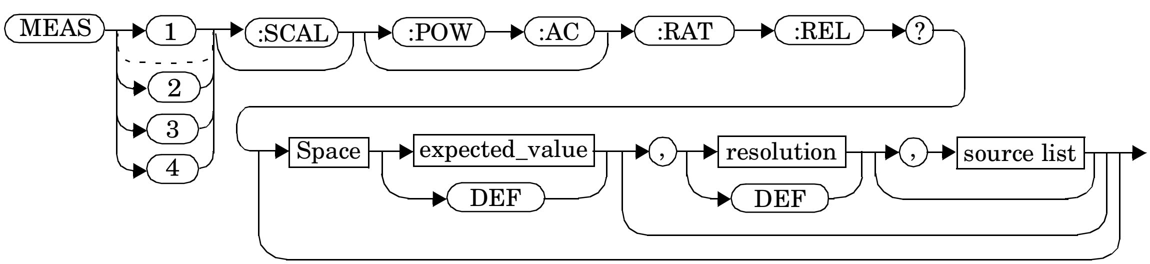
### Example

| | |
| --- | --- |
| `MEAS2:RAT? DEF,1,(@1),(@2)` | *This command queries the lower window / upper measurement ratio measurement of channel A over channel B, using the current sensor range and a resolution of 1 on both channels.* |

# MEASure[1]|2|3|4[:SCALar][:POWer:AC]:RATio:RELative? [<expected_value>[,<resolution>[,<source list>]]]

This command sets the specified window's measurement function to ratio mode with relative mode on, aborts, configures then initiates both channel A and B, calculates the ratio measurement and places the result on the bus. The result is a ratio based measurement and is expressed in the units defined by `UNIT[1]|2|3|4:POWer:RATio`. The relative value used is that set by the `CALCulate:RELative:MAGNitude:AUTO` command.

### Syntax




### Parameters

Refer to "Optional Parameters" on page 81 for additional details on the parameters in this command.

| Item | Description/Default | Range of Values |
|---|---|---|
| expected_value (for the expected power level) | The power meter ignores the numeric value entered in this parameter. Any value entered is treated like DEF. | sensor dependent<br>DEF[a] |
| resolution | A numeric value for the resolution. If unspecified the current resolution setting is used. | 1 to 4[b]<br>1.0, 0.1, 0.01, 0.001<br>DEF[a] |

| Item | Description/Default | Range of Values |
|---|---|---|
| source list | This channel list specifies the channels used to calculate the difference. If unspecified and the current window setup is a difference measurement then this difference setup is used, otherwise it defaults to channel A-B (N1912A) or A-A (N1911A). | (@1),(@2)[c]<br>(@2),(@1)[c]<br>(@1),(@1)<br>(@2),(@2)[c] |

a. The mnemonic DEF means DEFault. This is not equivalent to the DEFault parameter used in the command sub-systems. The parameters must be entered in the specified order. If parameters are omitted, they default from the right. The parameter DEFault is used as a place holder. Specifying DEF leaves the parameter value unchanged.

b. When the measurement result is linear this parameter represents the number of significant digits. When the measurement result is logarithmic 1 to 4 represents a resolution of 1, 0.1, 0.01 and 0.001 respectively.

c. N1912A only.

## Example

| | |
|---|---|
| `MEAS:RAT:REL?` | *This command queries the relative ratio measurement on the upper window / upper measurement.* |

**MEASure[1]|2|3|4[:SCALar][:POWer:AC]:RATio:RELative? [<expected_value>[,<resolution>[,<source list>]]]**

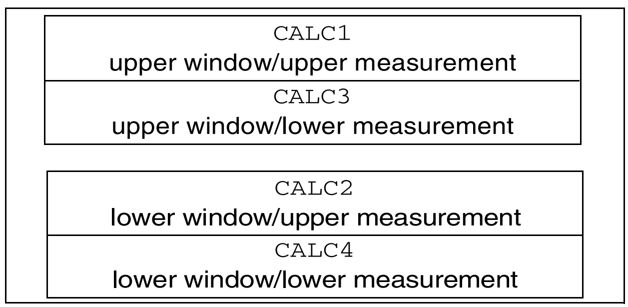
# 3 CALCulate Subsystem

# CALCulate Subsystem

The `CALCulate` subsystem performs post acquisition data processing. Functions in the `SENSe` subsystem are related to data acquisition, while the `CALCulate` subsystem operates on the data acquired by a `SENSe` function.

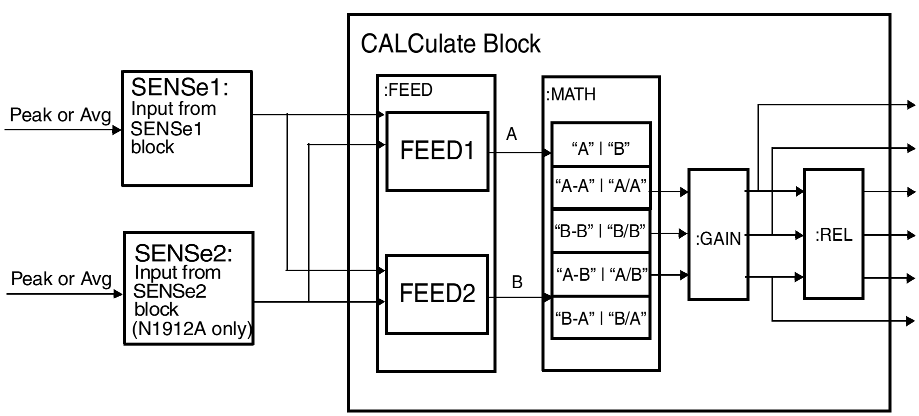
There are four independent `CALCulate` blocks in the power meter: two for each window, as shown in Figure 3-1. The numeric suffix of the CALCulate command determines which CALCulate block is used and where the measurement result is displayed.

**Figure 3-1 Measurement Display CALCulate Block Window**




Data from both `SENSe` blocks may feed any or all of the `CALCulate` blocks via the `MATH` command. Figure 3-2 details where the commands are applied within the `CALCulate` block.

**Figure 3-2 CALCulate Block**



<table>
<tr><th>Keyword</th><th>Parameter Form</th><th>Notes</th><th>Page</th></tr>
<tr><td>CALCulate[1]|2|3|4</td><td></td><td></td><td></td></tr>
<tr><td>    :FEED[1]|2</td><td><data_handle></td><td></td><td>page 142</td></tr>
<tr><td>    :GAIN</td><td></td><td></td><td></td></tr>
<tr><td>        [:MAGNitude]</td><td><numeric_value></td><td></td><td>page 146</td></tr>
<tr><td>        :STATe</td><td><boolean></td><td></td><td>page 148</td></tr>
<tr><td>    :LIMit</td><td></td><td></td><td></td></tr>
<tr><td>        :CLEar</td><td></td><td></td><td></td></tr>
<tr><td>            :AUTO</td><td><boolean></td><td></td><td>page 150</td></tr>
<tr><td>            [:IMMediate]</td><td></td><td></td><td>page 152</td></tr>
<tr><td>        :FAIL?</td><td></td><td>[query only]</td><td>page 153</td></tr>
<tr><td>        :FCOunt?</td><td></td><td>[query only]</td><td>page 154</td></tr>
<tr><td>        :LOWer</td><td></td><td></td><td></td></tr>
<tr><td>            [:DATA]</td><td><numeric_value></td><td></td><td>page 156</td></tr>
<tr><td>        :STATe</td><td><boolean></td><td></td><td>page 160</td></tr>
<tr><td>        :UPPer</td><td></td><td></td><td></td></tr>
<tr><td>            [:DATA]</td><td><numeric_value></td><td></td><td>page 158</td></tr>
<tr><td>    :MATH</td><td></td><td></td><td></td></tr>
<tr><td>        [:EXPRession]</td><td><string></td><td></td><td>page 162</td></tr>
<tr><td>            :CATalog?</td><td></td><td>[query only]</td><td>page 165</td></tr>
<tr><td>    :PHOLd</td><td></td><td></td><td></td></tr>
<tr><td>        :CLEar</td><td></td><td>[no query]</td><td>page 166</td></tr>
<tr><td>    :RELative</td><td></td><td></td><td></td></tr>
<tr><td>        [:MAGNitude]</td><td></td><td></td><td></td></tr>
<tr><td>            :AUTO</td><td><boolean></td><td></td><td>page 168</td></tr>
<tr><td>        :STATe</td><td><boolean></td><td></td><td>page 170</td></tr>
</table>

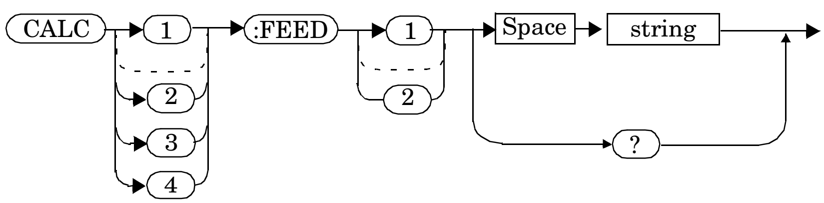
# CALCulate[1]|2|3|4:FEED[1]|2 <string>

This command sets the input measurement mode to be fed to the specified input on the CALC block. It is applied to the measurement after the `CALC:MATH:EXPR` command has been used to specify which channel the feed is taken from.

Measurement modes are coupled for combination measurements (for example, ratio measurements). For example, if one feed is changed to `PTAV`, the other is automatically changed to `PTAV`.

Under certain circumstances the measurement mode is changed by the `CALC:MATH:EXPR` command. Refer to page 162 for further information.

### Syntax



### Parameters

| Item | Description | Range of Values |
|---|---|---|
| string | The input measurement type to be fed to the specific input on the CALC block:<br>• `PEAK`: peak power.<br>• `PTAV`: peak to average.<br>• `AVER`: average.<br>Values may be followed by `ON SWEEP[1]|2|3|4` where the numeric specifies the gate to be used for the feed. For example: `"POW:PEAK ON SWEEP2"`<br>If ON `SWEEP[1]|2|3|4` is not supplied, the gate used is left unchanged.<br>A feed of "" (empty string) disables the CALC block and switches off that display line. | `"POW:PEAK"`<br>`"POW:PTAV"`<br>`"POW:AVER"` |

### Example

`CALC3:FEED2 "POW:AVER ON SWEEP2"` *This command selects the input for FEED2 of CALC block CALC3 to be average power, using gate 2. The channel from which the feed is taken is determined by* `CALC:MATH:EXPR`*.*

**Reset Condition** On reset, data_handle is set to :POW:AVER.

**Query** `CALCulate[1]|2|3|4:FEED[1]2?`

The query returns the current value of the string.

### Query Example

`CALC1:FEED2?` *This command queries the current setting of the data_handle on FEED2 of the upper window/upper measurement.*

**Error Message**

- If the command is used when no sensor is attached, error -241 “Hardware missing” occurs.
- If `<string>` contains `ON SWEEP[1]|2|3|4` and the feed’s `TRIG:SOUR` is not `INT` or `EXT` (for single channel power meters) or `INT1`, `INT2` or `EXT` (for dual channel power meters), error -221 “Settings conflict” occurs.
- If the command changes the measurement mode to `PEAK` or `PTAV` when a sensor other than a P-series or E9320 power sensor is connected or a P-series or E9320 sensor is connected and set to `AVERage` mode rather than `NORMal` mode, error -221, “Settings Conflict” occurs.

# CALCulate[1]|2|3|4:GAIN Commands

These commands are used to enter and enable a display offset on the specified window/measurement. The display offset is applied to the measurement signal after any math calculation.

The following commands are detailed in this section:

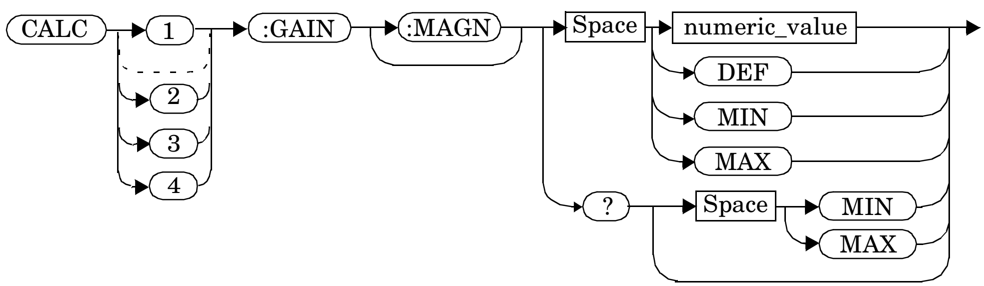
```
CALCulate[1]|2|3|4:GAIN[:MAGNitude] <numeric value>
CALCulate[1]|2|3|4:GAIN:STATe <boolean>
```

# CALCulate[1]|2|3|4:GAIN[:MAGNitude] <numeric_value>

This command is used to enter a value for the display offset on the specified window/measurement. The display offset is applied to the measurement signal after any math calculation.

Entering a value using this command automatically turns the `CALCulate[1]|2|3|4:GAIN:STATe` command to `ON`.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value for the display offset:<br>• `DEF`: the default value is 0 dB.<br>• `MIN`: -100.000 dB.<br>• `MAX`: +100.000 dB. | -100.000 to +100.000 dB<br>`DEF`<br>`MIN`<br>`MAX` |

### Example

| | |
|---|---|
| `CALC2:GAIN 20` | *This command enters a display offset of 20 dB to the lower window/lower measurement.* |

**Reset Condition** On reset, the display offset is set to 0 dB (`DEF`).

**Query** `CALCulate[1]|2|3|4:GAIN[:MAGNitude]? [MIN|MAX]`

The query returns the current setting of the display offset or the value associated with `MIN` and `MAX`.

**Query Example**

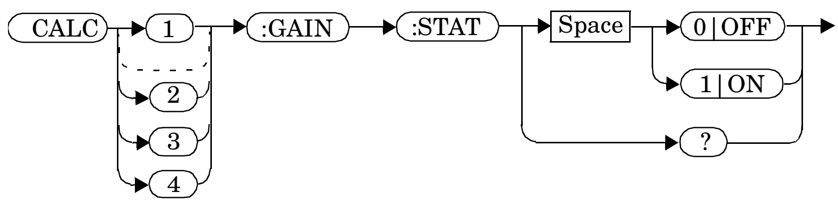
| | |
|---|---|
| `CALC1:GAIN?` | *This command queries the current setting of the display offset on the upper window / upper measurement.* |

**Error Message** If `CALCulate[1]|2|3|4:GAIN[:MAGNitude]` is set to `ON` while `SENSe:SPEed` is set to 200, error -221, “Settings Conflict” occurs.

# CALCulate[1]|2|3|4:GAIN:STATe <boolean>

This command is used on the specified window/measurement to enable and disable the display offset set by the `CALCulate[1]|2|3|4:GAIN[:MAGNitude]` command.

**Syntax**




**Example**

| | |
|---|---|
| `CALC2:GAIN:STAT 1` | *This command enables the display offset for the lower window / upper measurement.* |

**Reset Condition**

On reset, the gain is disabled.

**Query**

`CALCulate[1]|2|3|4:GAIN:STATe?`

The query enters a `1` or `0` into the output buffer indicating the status of the display offset.

- `1` is returned when the display offset feature is enabled.
- `0` is returned when the display offset feature is disabled.

**Query Example**

| | |
|---|---|
| `CALC1:GAIN:STAT?` | *This command queries whether the display offset in the upper window / upper measurement is on or off.* |

**Error Message**

If `CALCulate[1]|2|3|4:GAIN:STATe` is set to `ON` while `SENSe:SPEed` is set to 200, error -221, "Settings Conflict" occurs.

# CALCulate[1]|2|3|4:LIMit Commands

These commands set the limits on both the upper and lower windows/measurements enabling you to:

- Set upper and lower level limits.
- Query if there has been a failure.
- Count the number of failures.
- Clear the counter.

The following commands are detailed in this section:

`CALCulate[1]|2|3|4:LIMit:CLEar:AUTo <boolean>`

`CALCulate[1]|2|3|4:LIMit:CLEar[IMMediate]`

`CALCulate[1]|2|3|4:LIMit:FAIL?`

`CALCulate[1]|2|3|4:LIMit:FCOunt?`

`CALCulate[1]|2|3|4:LIMit:LOWer[:DATA]`

`CALCulate[1]|2|3|4:LIMit:UPPer[:DATA]`

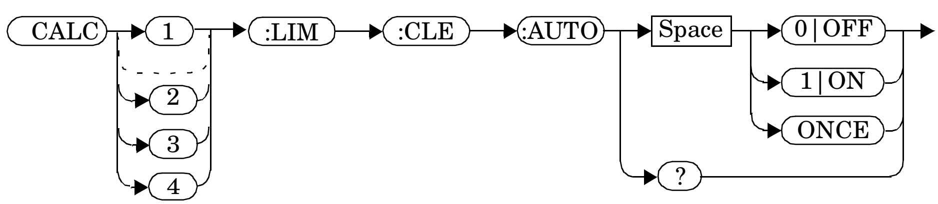
`CALCulate[1]|2|3|4:LIMit:STATe <boolean>`

# CALCulate[1]|2|3|4:LIMit:CLEar:AUTo <boolean>|ONCE

This command controls when the FCO (fail counter) is cleared of any limit failures. The FCO is used to determine the results returned by the `CALCulate[1]|2|3|4:LIMit:FAIL?` query.

- If `ON` is specified, the FCO is set to `0` each time a measurement is:
  - Initiated using `INITiate[:IMMediate]`.
  - Initiated using `INITiate:CONTinuous ON`.
  - Measured using `MEASure?`
  - Read using `READ?`
- If `OFF` is specified, the FCO is not cleared by the above commands.
- If `ONCE` is specified, the FCO is cleared only after the first initialization then starts accumulating any limit failures.

**Syntax**




**Example**

| | |
|---|---|
| `CALC1:LIM:CLE:AUTO 1` | *This command switches on automatic clearing of the FCO for the upper window/upper measurement.* |

**Reset Condition**

On reset, both windows and their measurements are set to `ON`.

### Query

`CALCulate[1]|2|3|4:LIMit:CLEar:AUTO?`

The query command enters a `1` or `0` into the output buffer indicating whether limit failures are cleared automatically when a new measurement is initiated on the specified window section.

- `1` is entered into the output buffer when limit failures are cleared automatically when a new measurement is initiated.
- `0` is entered into the output buffer when limit failures are not cleared automatically when a new measurement is initiated.

In the case where limit failures are cleared once, when a query occurs a `1` is entered into the output buffer if no measurement is initiated. If a measurement is initiated then `0` is entered.

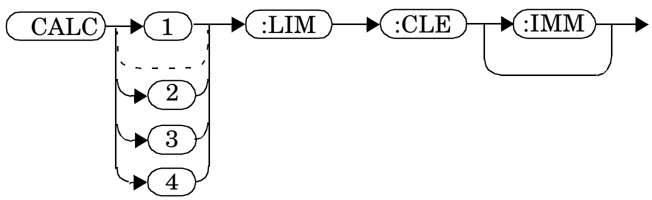
### Query Example

| | |
|---|---|
| `CALC1:LIM:CLE:AUTO?` | *This command queries when the FCO is cleared for the upper window / upper measurement.* |

# CALCulate[1]|2|3|4:LIMit:CLEar[:IMMediate]

This command immediately clears the FCO (fail counter) of any limit failures for the specified window. The FCO is used to determine the results returned by the `CALCulate[1]|2|3|4:LIMit:FAIL?` query

### Syntax




### Example

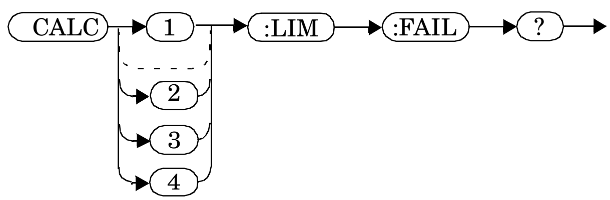
| | |
|---|---|
| CALC2:LIM:CLE:IMM | This command clears the FCO for the lower window/upper measurement. |

# CALCulate[1]|2|3|4LIMit:FAIL?

This query enters a `1` or `0` into the output buffer indicating whether there have been any limit failures for the specified window. A limit failure is defined as `CALC[1]|2|3|4:LIMit:FCO?` being non-zero. The FCO (fail counter) can be zeroed using the `CALC[1]|2|3|4:LIMit:CLEar` command.

- `1` is returned when one or more limit failures have occurred.
- `0` is returned when no limit failures have occurred.

### Syntax




### Example

| | |
|---|---|
| CALC1:LIM:FAIL? | This command queries if there have been any limit failures on the upper window/upper measurement. |

### Reset Condition

On reset, the buffer is set to zero for both upper and lower window measurements.

# CALCulate[1]|2|3|4:LIMit:FCOunt?

This query returns the total number of limit failures for the specified window/measurement.

If the appropriate `STATe` commands are set to `ON`, each time a measurement is initiated on the specified window/measurement and the result is outside the limits, the counter is incremented by one.

If the measured value is equal to a limit, this is a limit pass.

The counter is reset to zero by any of the following commands:

- `*RST`
- `CALCulate[1]|2|3|4:LIMit:CLEar:IMMediate`
- `CALCulate[1]|2|3|4:LIMit:CLEar:AUTO ON`

When `CALCulate[1]|2|3|4:LIMit:CLEar:AUTO` is set to `ON`, the counter is set to zero *each* time a measurement is:

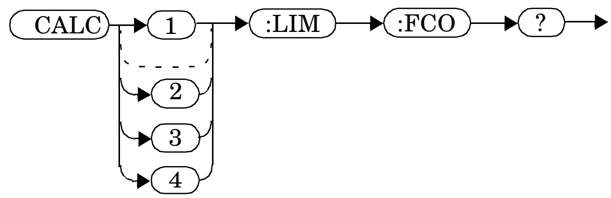
- measured using MEASure?
- read using READ?
- initiated using:
  - — `INITiate[:IMMediate]` or,
  - — `INITiate:CONTinuous ON`

When `CALCulate[1]|2|3|4:LIMit:CLEar:AUTO` is set to `ONCE`, the counter is set to zero the *first* time a measurement is:

- measured using `MEASure?`
- read using `READ?`
- initiated using:
  - — `INITiate[:IMMediate]` or,
  - — `INITiate:CONTinuous ON`

The maximum number of errors is $2^{16}$-1. If more than $2^{16}$-1 errors are detected the counter returns to zero.

### Syntax




### Example

`CALC1:LIM:FCO?`

*This command queries the number of limit failures on the upper window / upper measurement.*

**Reset Condition**

On reset, the counter is set to zero for both measurements of the upper and lower windows

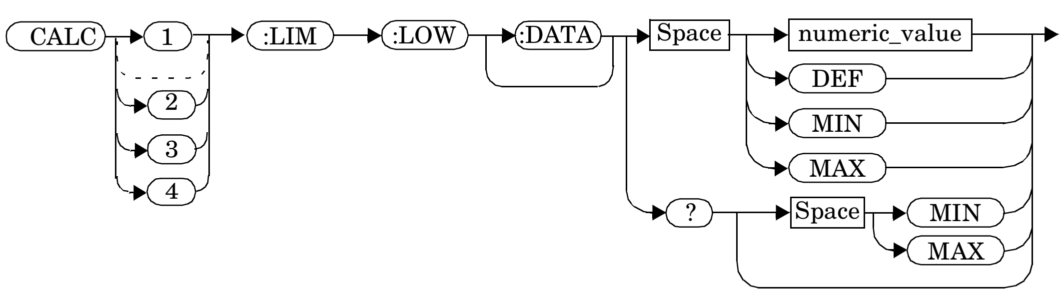
# CALCulate[1]|2|3|4:LIMit:LOWer[:DATA] <numeric_value>

This command enters a value for the lower test limit for the specified window/measurement used in the `CALCulate[1]|2|3|4:LIMit:FAIL?` test. The units used are dependent on the current setting of `UNIT:POWer` and `CALCulate:RELative:STATe` as shown in Table 3-1. When the measured value is less than the value specified in `CALCulate[1]|2|3|4:LIMit:LOWer[:DATA]`, `CALCulate[1]|2|3|4:LIMit:FAIL?` reports a fail. When the measured value is greater than or equal to the limit, a fail is not reported.

**Table 3-1 Measurement Units**

| **Measurement Mode** | **Measurement Type** | **CALC:REL:STAT OFF** | | **CALC:REL:STAT ON** | |
|---|---|---|---|---|---|
| | | **Linear** | **Log** | **Linear** | **Log** |
| **Single Channel** | Avg, Pk | Watt | dBm | % | dB |
| | Pk-Avg | % | dB | % | dB |
| **Ratio** | Avg, Pk, Pk-Avg | % | dB | % | dB |
| **Difference** | Avg, Pk | Watt | dBm | % | dB |
| | Pk-Avg | % | dB | % | dB |

### Syntax



### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value for the lower test limit:<br>• `DEF`: the default is -90.00 dBm or -90 db.<br>• `MIN`: -150 dBm or -180 dB.<br>• `MAX`: +230 dBm or +200 dB. | -150 to +230 dBm or<br>-180 to +200 dB<br>`DEF`<br>`MIN`<br>`MAX` |

### Example

`CALC2:LIM:LOW:DATA 0.1`

*This command enters a lower limit for the lower window / upper measurement depending on the window's units as follows:*
*dBm = 0.1 dBm*
*W = 100 mW*
*dB = 0.1 dB*
*% = 0.1%*

### Reset Condition

On reset, both measurements of the upper and lower windows are set to -90.00 dBm or -90 dB (`DEF`).

### Query

`CALCulate[1]|2|3|4:LIMit:LOWer[:DATA]? [MIN|MAX]`

The query returns the current setting of the lower limit or the values associated with `MIN` and `MAX` for the specified window.

### Query Example

`CALC2:LIM:LOW:DATA?`

*This command queries the lower limit set for the lower window upper measurement.*

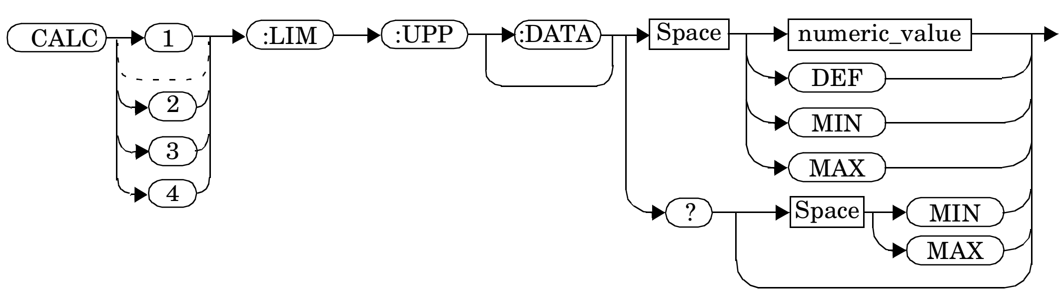
# CALCulate[1]|2|3|4:LIMit:UPPer[:DATA] <numeric_value>

This command enters a value for the upper test limit for the specified window/measurement used in the `CALCulate[1]|2|3|4:LIMit :FAIL?` test. The units used are dependent on the current setting of `UNIT:POWer` and `CALCulate:RELative:STATe` as shown in Table 3-2. When the measured power is greater than the value specified in `CALCulate[1]|2|3|4:LIMit:UPPer[:DATA]`, `CALCulate[1]|2|3|4:LIMit:FAIL?` reports a fail. When the measured level is less than or equal to the limit, a fail is not reported.

**Table 3-2** **Measurement Units**

| Measurement Mode | Measurement Type | CALC:REL:STAT OFF | | CALC:REL:STAT ON | |
|---|---|---|---|---|---|
| | | Linear | Log | Linear | Log |
| **Single Channel** | Avg, Pk | Watt | dBm | % | dB |
| | Pk-Avg | % | dB | % | dB |
| **Ratio** | Avg, Pk, Pk-Avg | % | dB | % | dB |
| **Difference** | Avg, Pk | Watt | dBm | % | dB |
| | Pk-Avg | % | dB | % | dB |

### Syntax



**Parameters**

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value for the lower test limit:<br>• `DEF`: the default is -90.00 dBm or -90 db.<br>• `MIN`: -150 dBm or -180 dB.<br>• `MAX`: +230 dBm or +200 dB. | -150 to +230 dBm or<br>-180 to +200 dB<br>`DEF`<br>`MIN`<br>`MAX` |

**Example**

`CALC2:LIM:UPP:DATA 5`

*This command enters an upper limit for the lower window / upper measurement depending on the window's units as follows:*
*dBm = 5 dBm*
*W = 5 W*
*dB = 5 dB*
*% = 5%*

**Reset Condition** On reset, both channels are set to +90.00 dBm or +90 dB.

**Query** `CALCulate[1]|2|3|4:LIMit:UPPer[:DATA]? [MIN|MAX]`

**Query Example**

`CALC2:LIM:UPP:DATA?`

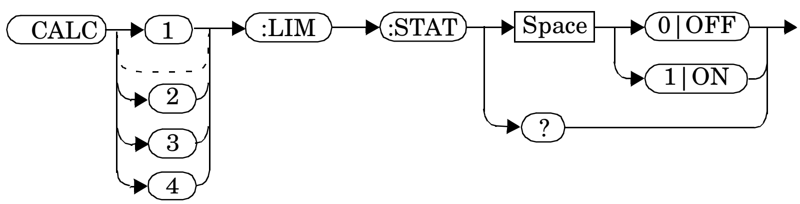
*This command queries the setting of the upper limit for the lower window / upper measurement.*

The query returns the current setting of the upper limit or the values associated with `MIN` and `MAX` for the specified window/measurement.

# CALCulate[1]|2|3|4:LIMit:STATe <boolean>

This command enables/disables the test limits for the specified window.

### Syntax




### Example

| | |
|---|---|
| `CALC2:LIM:STAT 1` | *This command enables the limit checking function for the lower window upper measurement.* |

### Reset Condition

On reset, limit checking is disabled.

### Query

`CALCulate[1]|2|3|4:LIMit:STATe?`

The query enters `1` or `0` into the output buffer indicating the status of the limits testing feature for the specified window/measurement.

- `1` is returned when limits testing is enabled.
- `0` is returned when limits testing is disabled.

### Query Example

| | |
|---|---|
| `CALC1:LIM:STAT?` | *This command queries whether the limit checking function for the upper window/upper measurement is on or off.* |

### Error Message

If `CALCulate[1|2|3|4]:LIMit:STATe` is set to `ON` while `[SENSe[1]]|SENSe2:SPEed` is set to 200, error -221, “Settings Conflict” occurs.

# CALCulate[1]|2|3|4:MATH Commands

These commands define and carry out the following mathematical transformations on `SENSe` data:

- Single channel.
- Difference.
- Ratio.

The following commands are detailed in this section:

`CALCulate[1]|2|3|4:MATH[:EXPRession] <string>`

`CALCulate[1]|2|3|4:MATH[:EXPRession]:CATalog?`

# CALCulate[1]|2|3|4:MATH[:EXPRession] <string>

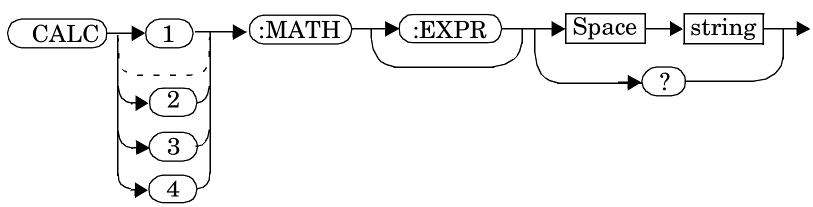
This command sets the specified window/measurement to a single channel, difference or ratio measurement.

The command may result in a change to the measurement mode set by CALC:FEED <string>. The following sequence of commands provides an example:

1. `SENS2:DET:FUN=AVERage`
2. `CALC:MATH "(SENS1)"`
3. `CALC:FEED1 "POW:PEAK"`
4. `CALC:MATH "(SENS2)"`

The `FEED1` measurement mode, set in step 3, is made invalid by step 4 and automatically changed to `"POW:AVER"`.

## Syntax



### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| string | A single string value detailing the measurement type:<br>• For the Agilent N1911A the default is SENS1.<br>• For the Agilent N1912A the default is SENS1 if the upper window is selected, or SENS2 if the lower window is selected. | "(SENS1)"[a]<br>"(SENS2)"[a,b]<br>"(SENS1-SENS1)"[a]<br>"(SENS2-SENS2)"[a,b]<br>"(SENS1/SENS1)"[a]<br>"(SENS2/SENS2)"[a,b]<br>"(SENS1-SENS2)"[a,b]<br>"(SENS2-SENS1)"[a,b]<br>"(SENS1/SENS2)"[a,b]<br>"(SENS2/SENS1)"[a,b] |

a. Quotes are mandatory. Either single or double quotes may be used.
b. N1912A only.

### Example

| | |
|---|---|
| `CALC2:MATH "(SENS2/SENS1)"` | *This command sets the lower window / upper measurement to make a channel B / A ratio measurement.* |

### Reset Condition

On reset, the Agilent N1911A upper and lower window measurements are set to channel A (`"(SENS1)"`). On the N1912A the upper window measurements are set to channel A (`"(SENS1)"`) and the lower window measurements to channel B (`"(SENS2)"`)

### Query

`CALCulate[1]|2|3|4:MATH[:EXPRession]?`

The query returns the current math measurement setting on the specified window.

### Query Example

| | |
|---|---|
| `CALC1:MATH?` | *This command queries the current setting of the math expression on the upper window / upper measurement.* |

### Error Messages

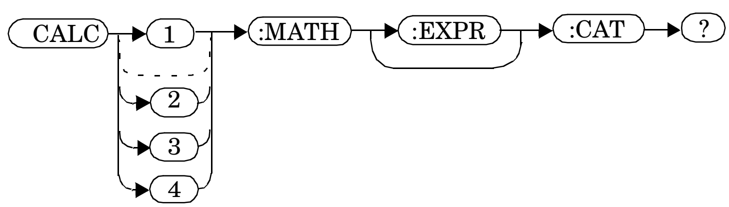
- For the single channel N1911A power meter: if `<string>` is not set to "`(SENS1)`" while `SENSe:SPEed` is set to 200, error -221, "Settings Conflict" occurs.
- For the dual channel N1912A power meter: if `<string>` is not set to "`(SENS1)`" or "`(SENS2)`" while `SENS1:SPEEd` or `SENS2:SPEEd` is set to 200, error -221, "Settings Conflict" occurs.

# CALCulate[1]|2|3|4:MATH[:EXPRession]:CATalog?

This query lists all the defined expressions. The response is a list of comma separated strings. Each string contains an expression.

- For the N1911A the string is:
  “`(SENS1)`”, “`(SENS1-SENS1)`”, “`(SENS1/SENS1)`”
- For the N1912A the string is:
  "(SENS1)","(SENS2)","(SENS1/SENS2)",
  "(SENS2/SENS1)","(SENS1-SENS2)","(SENS2-SENS1)"
  "(SENS1-SENS1)","(SENS2-SENS2)","(SENS1/SENS1)",
  "(SENS2/SENS2)"

### Syntax




### Example

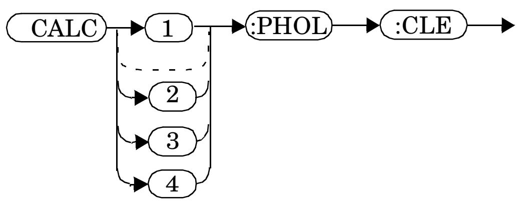
| | |
|---|---|
| `CALC1:MATH:CAT?` | *This command lists all the defined math expressions.* |

# CALCulate[1]|2|3|4:PHOLd:CLEar

This command clears the peak hold value for a specified CALC block so that a new peak hold value can be set.

**NOTE** Clearing the peak hold value for a specified CALC block may affect the peak hold value of other CALC blocks, depending on the CALC channel set up (set by `CALC:MATH:EXPR`).

### Syntax




### Example

| | |
|---|---|
| `CALC2:PHOLd:CLEar` | *This command clears the peak hold value for CALC2.* |

### Error Messages

- If no power sensor is connected, error -241 “Hardware missing” occurs.
- If a sensor, other than an P-series or E9320 power sensor, is connected, error -241 “Hardware missing” occurs.
- If `SENS:DET:FUNC` is set to `AVER` or `TRIG:SOUR` is set to `INT1`, `INT2` or `EXT`, error -221 “Settings conflict” occurs.

# CALCulate[1]|2|3|4:RELative Commands

These commands compare the measurement signal to a reference value.

Within the `CALCulate` block the relative value is applied to the measurement signal after any math calculations and display offsets have been applied.

The commands described in this section:

`CALCulate[1]|2|3|4:RELative[:MAGNitude]:AUTO <boolean>|ONCE`

`CALCulate[1]|2|3|4:RELative:STATe <boolean>`

# CALCulate[1]|2|3|4:RELative[:MAGNitude]:AUTO <boolean>|ONCE

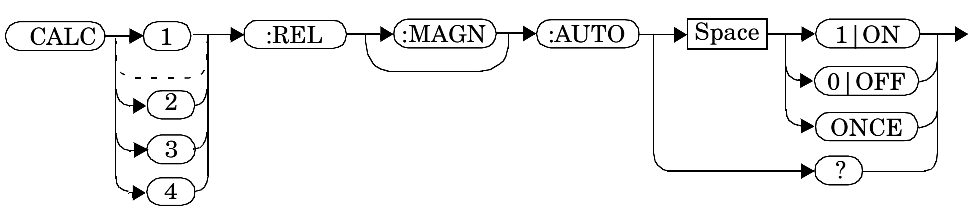
This command sets the reference value to be used in the relative measurement. Within the `CALCulate` block the relative value is applied to the measurement signal after any math calculations and display offsets have been applied.

The value should be set to `ONCE` to set the reference value to be used in relative measurements. Selecting `ONCE` sets the reference value to that of the measurement signal after any math calculations and display offsets have been applied. After the reference value has been set the command returns to `OFF`. Setting this command to `ONCE` turns the `CALCulate[1]|2|3|4:RELative:STATe` command to `ON`.

If `0|OFF` is selected, no reference value is applied to the measurement signal. There is no situation in which you would want to send this command with `OFF`. `OFF` is only available because it is required for the query response.

If `1|ON` is selected, it causes error -224, “Illegal parameter value” to occur.

### Syntax




### Example

| | |
|---|---|
| `CALC1:REL:AUTO ONCE` | *This command sets a reference value to be used in the relative measurement on the upper window/upper measurement.* |

### Query

`CALCulate[1]|2|3|4:RELative[:MAGNitude]:AUTO?`

The query always returns `OFF`.

**Error Message**

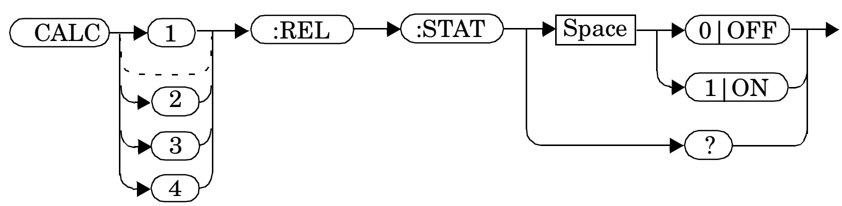
- If `CALCulate:RELative[:MAGNitude]:AUTO` is set to `ONCE` while `SENSe:SPEed` is set to 200, error -221, "Settings Conflict" occurs.
- If the value is set to `ON` error -224, "Illegal parameter value" occurs.

# CALCulate[1]|2|3|4:RELative:STATe <boolean>

This command enables/disables relative mode. If the command is:

- disabled, the measurement signal remains unchanged.
- enabled, the current relative value set by `CALCulate:RELative:MAGnitude:AUTO` is applied to the measurement signal.

**Syntax**




**Example**

| | |
|---|---|
| `CALC1:REL:STAT OFF` | *This command disables the relative mode on the upper window/upper measurement.* |

**Reset Condition** On reset, relative mode is disabled.

**Query** `CALCulate[1]|2|3|4:RELative:STATe?`

The query returns a `1` or `0` into the output buffer.

- `1` is returned when relative mode is enabled.
- `0` is returned when relative mode is disabled.

**Query Example**

| | |
|---|---|
| `CALC1:REL:STAT?` | *This command queries whether relative mode is off or on for the upper window/upper measurement.* |

**Error Message** If `CALCulate:RELative:STATe` is set to `ON` while `SENSe:SPEed` is set to 200, error -221, “Settings Conflict” occurs.

# 4 CALibration Subsystem

# CALibration Subsystem

The `CALibration` command subsystem is used to zero and calibrate the power meter. It is also used to set the reference calibration factor for the power sensor which is being used.

The numeric suffix of the `CALibration` command refers to a specific channel:

- `CALibration1` represents channel A.
- `CALibration2` represent channel B.
  This command does not apply to the single channel N1911A power meter and results in the error "Header suffix out of range."

Zeroing and calibration of the power meter is recommended:

- When a 5$^{o}$C change in temperature occurs.
- When you change the power sensor.
- Every 24 hours.
- Prior to measuring low level signals. For example, 10 dB above the lowest specified power for your sensor.

The following `CALibration` commands are overlapped commands:

- `CAL:ALL`
- `CAL:AUTO`
- `CAL:ZERO:AUTO`

An overlapped command allows the instrument to continue parsing and executing subsequent commands while it is still executing.

CALibration Subsystem

| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
| CALibration[1]\|2 | | | |
| [:ALL] | | [event; no query] | page 174 |
| [:ALL]? | | [event;query] | page 176 |
| :AUTO | <boolean>\|ONCE | | page 178 |
| :RCALibration | <boolean> | | page 180 |
| :RCFactor | <numeric_value> | [non-SCPI] | page 182 |
| :ZERO | | | |
| :AUTO | <boolean>\|ONCE | | page 184 |
| :NORMal | | | |
| :AUTO | <boolean> | | page 186 |

# CALibration[1]|2[:ALL]

**NOTE** This command is identical to `CALibration[1]|2[:ALL]?`, however, unlike the query it does not provide a response to indicate whether the calibration has been successful or not.

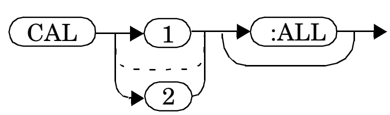
This command causes the power meter to perform a calibration sequence on the specified channel. The command assumes that the power sensor is connected to the POWER REF output. The calibration sequence consists of:

1. Zeroing the power meter (`CALibration:ZERO:AUTO ONCE`), and
2. calibrating the power meter (`CALibration:AUTO ONCE`).

For 8480 series power sensors the reference calibration factor used during this calibration can be derived from either an active sensor calibration table or the value entered using `CALibration:RCFactor`. The actual value used is the one which was most recently set. That is, a value entered using `CALibration:RCFactor` is overridden if a sensor calibration table is subsequently selected and enabled. Conversely, `CALibration:RCFactor` overrides any reference calibration factor previously set from a sensor calibration table. To determine the currently set reference calibration factor use `CALibration:RCFactor?`

E-Series power sensors have their sensor calibration tables stored in EEPROM which means that the reference calibration factor is automatically downloaded by the power meter.

## Syntax



### Example

| | |
|---|---|
| `CAL1:ALL` | *This command causes the power meter to perform a calibration sequence on channel A.* |

### Error Messages

- If the calibration was not carried out successfully the error -231, “Data Questionable; CAL ERROR” occurs. If you are using an N1912A the error message specifies which channel failed calibration.
- If zeroing was not carried out successfully the error -231, “Data Questionable; ZERO ERROR” occurs. If you are using an N1912A the error message specifies which channel failed calibration.
- If there is no sensor connected, the error -241, “Hardware Missing” occurs.

# CALibration[1]|2[:ALL]?

**NOTE** This query is identical to `CALibration[1]|2[:ALL]`, however, unlike the command, it provides a response to indicate whether the calibration has been successful or not.

This query causes the power meter to perform a calibration sequence on the specified channel. The query assumes that the power sensor is connected to the POWER REF output. The calibration sequence consists of:

1. Zeroing the power meter (`CALibration:ZERO:AUTO ONCE`), and
2. calibrating the power meter (`CALibration:AUTO ONCE`).

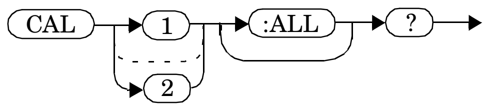
When the calibration sequence is completed, `0` or `1` is entered into the output buffer to indicate if the sequence was successful. If the result is:

- `0`, the calibration has passed.
- `1`, the calibration has failed.

For the 8480 power sensors the reference calibration factor used during this calibration can be derived from either an active sensor calibration table or the value entered using `CALibration:RCFactor`. The actual value used is the one which was most recently set. That is, a value entered using `CALibration:RCFactor` is overridden if a sensor calibration table is subsequently selected and enabled. Conversely, `CALibration:RCFactor` overrides any reference calibration factor previously set from a sensor calibration table. To determine the currently set reference calibration factor use `CALibration:RCFactor?`

The E-Series power sensors have their sensor calibration tables stored in EEPROM which means that the reference calibration factor is automatically downloaded by the power meter.

## Syntax




## Query Example

| | |
|---|---|
| `CAL1:ALL?` | *This command causes the power meter to perform a calibration sequence on channel A and return a result.* |

## Error Messages

- If the calibration was not carried out successfully the error -231, “Data Questionable; CAL ERROR” occurs. If you are using an N1912A the error message specifies which channel failed calibration.
- If zeroing was not carried out successfully the error -231, “Data Questionable; ZERO ERROR” occurs. If you are using an N1912A the error message specifies which channel failed calibration.
- If there is no sensor connected, the error -241, “Hardware Missing” occurs.

# CALibration[1]|2:AUTO [ONCE|ON|OFF|0|1]

This command calibrates the specified channel when enabled. The command assumes that an 8480 or E-series power sensor is connected to a 1 mW reference signal.

`1|ON` can only be used with a P-series sensor. When `1|ON` is enabled the calibration is updated if the meter's or sensor's temperature changes by ±5°C or the time since last calibration is greater then 1000 minutes.

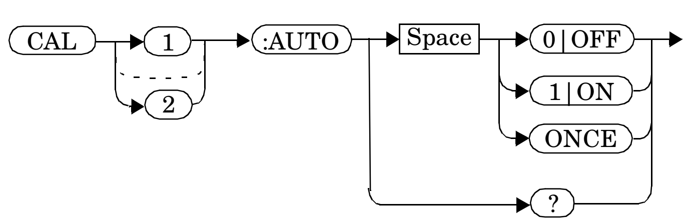
The `0|OFF` parameter is only required for the query response and is ignored in the command.

The E-series power sensors have their sensor calibration tables stored in EEPROM which means that the reference calibration factor is automatically downloaded by the power meter.

For 8480 series power sensors the reference calibration factor used during this calibration can be obtained from an active sensor calibration table or the value entered using `CALibration:RCFactor`. The actual value used is the one which was most recently set. For example, a value entered using `CALibration:RCFactor` is overridden if a sensor calibration table is subsequently selected and enabled and `CALibration:RCFactor` overrides any reference calibration factor previously set from a sensor calibration table. To determine the current reference calibration factor, use `CALibration:RCFactor?`

**NOTE** If the power meter is using an 8480 or E-series power sensor it should be zeroed before calibration using the `CALibration:ZERO:AUTO ONCE` command.

### Syntax



**Example**

| | |
|---|---|
| `CAL1:AUTO ONCE` | *This command causes the power meter to perform a calibration on channel A.* |

**Reset Condition** On reset, automatic calibration is disabled.

**Query** `CALibration[1]|2:AUTO?`

The query always returns a value of `0`.

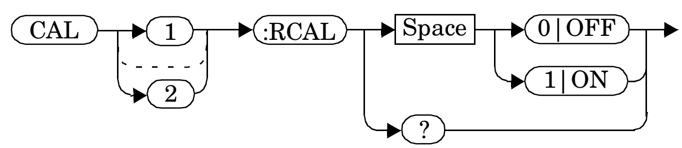
**Error Messages**

- If this command is set to `ON` and an 8480 series or E-series power sensor is connected the error -241, “Hardware missing” occurs.
- If the calibration was not carried out successfully the error -231, “Data Questionable; CAL ERROR” occurs. If you are using an N1912A the error message specifies which channel failed calibration.
- If there is no sensor connected, the error -241, “Hardware Missing” occurs.

# CALibration[1]|2:RCALibration <boolean>

This command enables and disables the zero/cal lockout facility. With the lockout facility enabled the power meter is stopped from making measurements until the connected sensor has been zeroed and calibrated.

**Syntax**




**Example**

| | |
|---|---|
| `CAL1:RCAL 1` | *This command enables the zero/cal lockout facility on channel A.* |

**Reset Condition**

On reset, the state of the zero/cal lockout is unaffected.

**Query**

`CALibration[1]|2:RCALibration?`

The query enters a `1` or `0` into the output buffer indicating whether zero/cal lockout is enabled or disabled.

- `1` is returned if zero/cal lockout is enabled.
- `0` is returned if zero/cal lockout is disabled.

**Query Example**

| | |
|---|---|
| `CAL1:RCAL?` | *This command queries whether or not the zero/cal lockout facility is enabled for channel A.* |

**Error Messages**

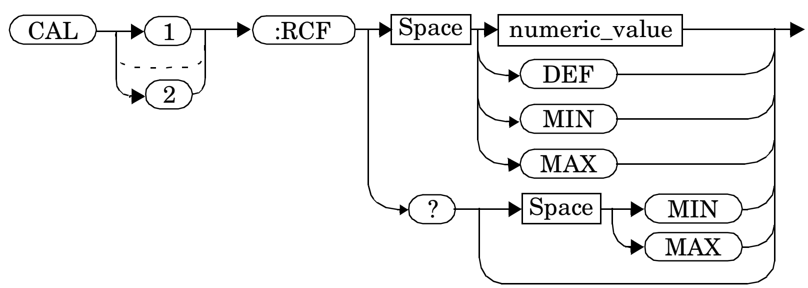
When `CAL[1]|2:RCAL` is `ON` and the sensor currently connected to the appropriate channel (A or B) has not been zeroed and calibrated, then any SCPI command which would normally return a measurement result (for example, `FETC?`, `READ?`, `MEAS?` etc) does not return a result and generates the error -230, "Data corrupt or stale; Please zero and Cal."

After the sensor has been zeroed and calibrated the return measurement results commands function normally.

# CALibration[1]|2:RCFactor <numeric_value>

This command is used with 8480 series power sensors to set the reference calibration factor of the specified channel. Reference calibration factors can also be set using sensor calibration tables. The power meter uses the most recently set reference calibration factor.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value:<br>• DEF: the default is 100%.<br>• MIN: 1%.<br>• MAX: 150%. | 1.0 to 150.0 PCT<br>DEF<br>MIN<br>MAX |

### Example

| | |
|---|---|
| `CAL1:RCF 98` | *This command enters a reference calibration factor of 98% to channel A.* |

### Reset Condition

On reset, the reference calibration factor is set to 100%.

**Query**

`CALibration[1]|2:RCFactor? [MIN|MAX]`

The query returns the current setting of the reference calibration factor or the values associated with `MIN` and `MAX`.

**Query Example**

| | |
|---|---|
| `CAL2:RCF?` | *This command queries the reference calibration factor of channel B.* |

**Error Messages**

If this command is used when a P-series or E-series power sensor is connected the error -241, “Hardware missing” occurs.

# CALibration[1]|2:ZERO:AUTO [ONCE|ON|OFF|0|1]

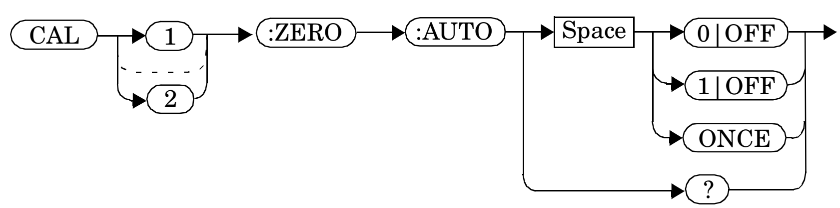
This command causes the power meter to perform its zeroing routine on the specified channel when enabled. This adjusts the power meter for a zero power reading with no power supplied to the power sensor.

`1|ON` can only be used with a P-series sensor. When `1|ON` is enabled the the zero is maintained by a combination of *on-the-fly* zero measurements and temperature compensation.

The `0|OFF` parameter is only required for the query response and is ignored in the command.

Except when using a P-series sensor, this command assumes that a power sensor is not connected to a power source.

## Syntax




## Example

`CAL2:ZERO:AUTO ONCE` *This command causes the power meter to perform a zeroing routine on channel B.*

## Reset Condition

On reset, automatic zeroing is disabled.

## Query

`CALibration[1]|2:ZERO:AUTO?`

The query always returns a value of `0`.

**Error Messages**

- If this command is set to `ON` and an 8480 series or E-series power sensor is connected the error -241, “Hardware missing” occurs.
- If zeroing was not carried out successfully the error -231, “Data Questionable; ZERO ERROR” occurs. If you are using an N1912A, the error message specifies which channel failed zeroing.
- If there is no sensor connected, the error -241, “Hardware Missing” occurs.

# CALibration[1]|2:ZERO:NORMal:AUTO <boolean>

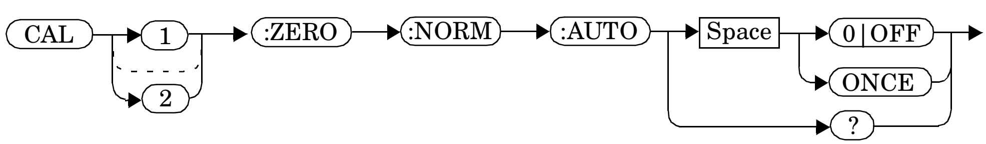
This command provides a quick way of zeroing the NORMAL path of an E9320 series sensor. The average only path is unaffected. This command can only be used to zero an E9320 series sensor and a P-series sensor.

**NOTE** The P-series sensor only has a NORMAL path. Hence, the reason this E9320 series sensor command is allowed to function.

The command causes the power meter to perform its zeroing routine, on the specified channel, when `ONCE` is selected. This adjusts the power meter for a zero power reading with no power supplied to the power sensor.
The `0|OFF` parameter is only required for the query response and is ignored in the command. If `1|ON` is selected on an E9320 series sensor, it causes the error -224, “Illegal parameter value” to occur.

Except when using a P-series sensor, this command assumes that the E9320 series sensor is not connected to a power source.

### Syntax




### Example

| | |
|---|---|
| `CAL2:ZERO:NORM:AUTO ONCE` | *This command causes the power meter to perform a zeroing routine on channel B.* |

### Reset Condition

On reset, automatic zeroing is disabled.

### Query

`CALibration[1]|2:ZERO:NORMal:AUTO?`

The query always returns a value of `0`.

**Error Messages**

- If zeroing was not carried out successfully the error -231, “Data Questionable; ZERO ERROR” occurs. If you are using a dual channel power meter, the error message specifies which channel failed zeroing.
- If this command is set to `ON` the error -224, “Illegal parameter value” occurs.
- If there is no sensor connected, or if a sensor other than an E9320 or P-series is connected, the error -241, “Hardware missing” occurs.
- If an E9320 sensor is connected and is not in NORMAL mode, the error -221 “Settings conflict” occurs.

# 5 DISPlay Subsystem

# DISPlay Subsystem

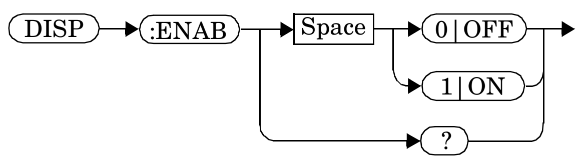
The `DISPlay` subsystem is used to control the selection and presentation of the windows used on the power meter's display.

| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
| DISPlay | | | |
| :ENABle | <boolean> | | page 191 |
| :SCReen | | | |
| :FORMat | <character_data> | | page 192 |
| [:WINDow[1]\|2] | | | |
| :ANALog | | | |
| :LOWer | <numeric_value> | | page 196 |
| :UPPer | <numeric_value> | | page 198 |
| :FORMat | <character_data> | [non-SCPI] | page 200 |
| :METer | | | |
| :LOWer | <numeric_value> | [non-SCPI] | page 203 |
| :UPPer | <numeric_value> | [non-SCPI] | page 205 |
| [:NUMeric[1]\|2] | | | |
| :RESolution | <numeric_value> | | page 207 |
| :SELect[1]\|2 | | | page 209 |
| [:STATe] | <boolean> | | page 210 |
| :TRACe | | | |
| :FEED | <character_data> | | page 211 |

# DISPlay:ENABle <boolean>

This command is used to enable and disable the display. At power-up the display is always enabled.

**Syntax**




**Example**

| | |
|---|---|
| `DISP:ENAB 0` | *This command disables the display.* |

**Reset Condition** On reset, the display is enabled.

**Query** `DISPlay:ENABle?`

The query returns a `1` or `0` into the output buffer.

- `1` is returned when the display is enabled.
- `0` is returned when the display is disabled.

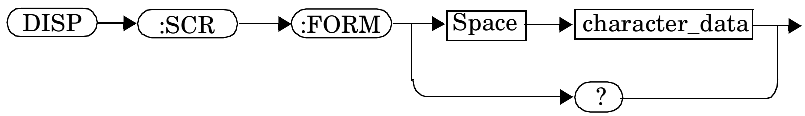
**Query Example**

| | |
|---|---|
| `DISP:ENAB?` | *This command queries whether the display is on or off.* |

# DISPlay:SCReen:FORMat <character_data>

This command sets the display format.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | Sets the display format:<br>• `WINDowed`: the windowed format provides two display windows. Each window can display two measurements.<br>• `EXPanded`: the expanded format provides one display window which can display a single measurement. The EXP display format provides access to softkeys.<br>• `FSCReen`: the full screen format provides one display window which can display a single measurement. The FSCR display format does not provide access to softkeys. | `WIND`<br>`EXP`<br>`FSCR` |

### Example

`DISP:SCReen:FORM FSCR` *This command sets the display format to full screen.*

**Reset Condition** On reset, the display format is `WIND`.

**Query** `DISPlay:SCReen:FORMat?`

The query returns `WIND`, `EXP` or `FSCR`.

**Query Example**

| | |
|---|---|
| `DISP:SCR:FORM?` | *This command queries the display format.* |

# DISPlay[:WINDow[1]|2] Commands

These commands control various characteristics of the display windows. `WINDow1` and `WINDow2` represent the upper and lower windows respectively.

The following commands are detailed in this section:

`DISPlay[:WINDow[1]|2]:ANALog:LOWer <numeric_value>`

`DISPlay[:WINDow[1]|2]:ANALog:UPPer <numeric_value>`

`DISPlay[:WINDow[1]|2]:FORMat <character_data>`

`DISPlay[:WINDow[1]|2]:METer:LOWer <numeric_value>`

`DISPlay[:WINDow[1]|2]:METer:UPPer <numeric_value>`

`DISPlay[:WINDow[1]|2][NUMeric[1|2]]:RESolution <numeric_value>`

`DISPlay[:WINDow[1]|2]:SELect[1]|2`

`DISPlay[:WINDow[1]|2][:STATe] <boolean>`

`DISPlay[:WINDow[1]|2]:TRACe:FEED <character_data>`

# DISPlay[:WINDow[1]|2]:ANALog Commands

These commands control the upper and lower scale limits of the analog meter.

The following commands are detailed in this section:

`DISPlay[:WINDow[1]|2]:ANALog:LOWer <numeric_value>`

`DISPlay[:WINDow[1]|2]:ANALog:UPPer <numeric_value>`

# DISPlay[:WINDow[1]|2]:ANALog:LOWer <numeric_value>

This command sets the analog meter lower scale limit.

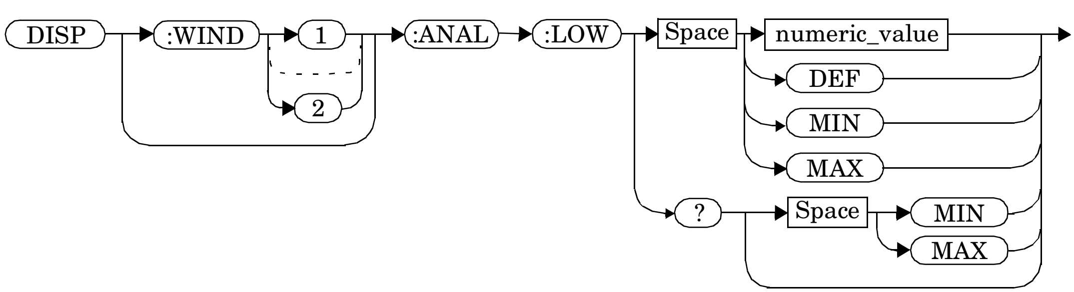
**NOTE** This command has the same purpose as `DISPlay[:WINDow[1]|2]:METer:LOWer <numeric_value>`

The units used are dependent on the current setting of `UNIT:POWer` and `CALCulate:RELative:STATe` as shown in Table 5-1.

**Table 5-1 Measurement Units**

| Measurement Mode | Measurement Type | CALC:REL:STAT OFF | | CALC:REL:STAT ON | |
|---|---|---|---|---|---|
| | | **Linear** | **Log** | **Linear** | **Log** |
| **Single Channel** | Avg, Pk | Watt | dBm | % | dB |
| | Pk-Avg | % | dB | % | dB |
| **Ratio** | Avg, Pk, Pk-Avg | % | dB | % | dB |
| **Difference** | Avg, Pk | Watt | dBm | % | dB |
| | Pk-Avg | % | dB | % | dB |

## Syntax



### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value for the analog meter lower scale limit:<br>• `DEF`: the default is -70 dBm<br>• `MIN`: -150 dBm<br>• `MAX`: 230 dBm<br>Units used are determined by the current setting of `UNIT:POWer` and `CALCulate:RELative:STATe` as shown in Table 5-1. | -150 to 230 dBm<br>`DEF`<br>`MIN`<br>`MAX` |

### Example

`DISP:WIND1:ANAL:LOW -50` *This command sets the upper window's analog meter lower scale limit to -50 dBm*

**Reset Condition** On reset, the value is set to -70 dBm for both windows.

**Query** `DISPlay:[WINDow[1]|2]:ANALog:LOW? [MIN|MAX]`

The query returns the current setting of the analog meter's lower scale limit, or the value associated with `MIN` or `MAX`. The format of the response is `<NR3>`. The units in which the results are returned are determined by the current setting of `UNIT:POWer` and `CALCulate:RELative:STATe` as shown in Table 5-1.

### Query Example

`DISP:WIND1:ANAL:LOW?` *This command queries the lower scale limit set on the analog meter in the upper window.*

# DISPlay[:WINDow[1]|2]:ANALog:UPPer <numeric_value>

This command sets the analog meter upper scale limit.

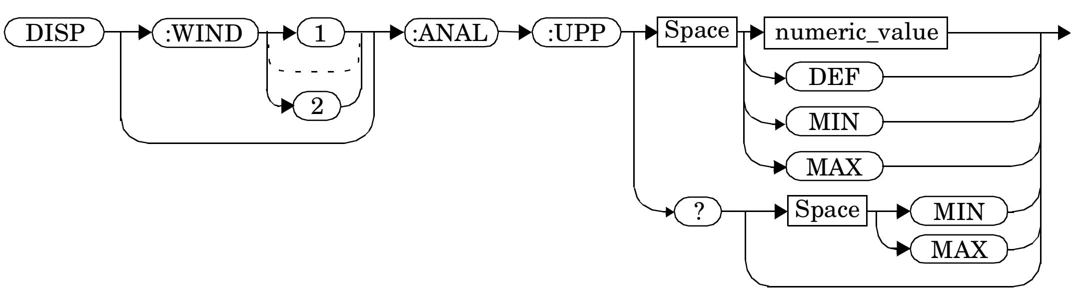
**NOTE** This command has the same purpose as `DISPlay[:WINDow[1]|2]:METer:UPPer <numeric_value>`

The units used are dependent on the current setting of `UNIT:POWer` and `CALCulate:RELative:STATe` as shown in Table 5-2.

**Table 5-2 Measurement Units**

| Measurement Mode | Measurement Type | CALC:REL:STAT OFF | | CALC:REL:STAT ON | |
|---|---|---|---|---|---|
| | | Linear | Log | Linear | Log |
| **Single Channel** | Avg, Pk | Watt | dBm | % | dB |
| | Pk-Avg | % | dB | % | dB |
| **Ratio** | Avg, Pk, Pk-Avg | % | dB | % | dB |
| **Difference** | Avg, Pk | Watt | dBm | % | dB |
| | Pk-Avg | % | dB | % | dB |

### Syntax



### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value for the analog meter upper scale limit:<br>• DEF: the default is 20 dBm<br>• MIN: -150 dBm<br>• MAX: 230 dBm<br>Units used are determined by the current setting of UNIT:POWer and CALCulate:RELative:STATe as shown in Table 5-2. | -150 to 230 dBm<br>DEF<br>MIN<br>MAX |

### Example

| | |
|---|---|
| DISP:WIND2:ANAL:UPP 50 | *This command sets the lower window's analog meter upper scale limit to 50 dBm* |

**Reset Condition** On reset, the upper scale limit is set to 20 dBm.

**Query** DISPlay:[WINDow[1]|2]:ANALog:UPPer? [MIN|MAX]

The query returns the current setting of the analog meter's upper scale limit, or the value associated with MIN or MAX. The format of the response is <NR3>. The units in which the results are returned are determined by the current setting of UNIT:POWer and CALCulate:RELative:STATe as shown in Table 5-2.

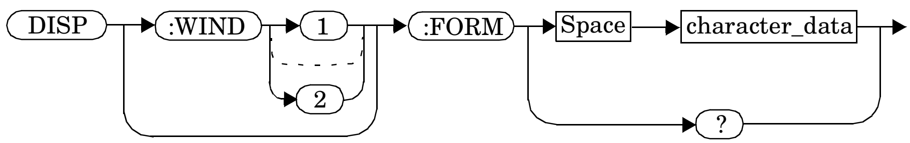
### Query Example

| | |
|---|---|
| DISP:WIND2:ANAL:UPP? | *This command queries the upper scale limit set on the analog meter in the lower window* |

# DISPlay[:WINDow[1]|2]:FORMat <character_data>

This command selects the format of the selected window.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | Sets the window format:<br>• `DIGital`: sets the window display to digital. This setting is the same as SNUMeric.<br>• `ANALog`: sets the window display to analog using the currently SELected measurement.<br>• `SNUMeric`: sets the window display to single numeric. The currently SELected measurement is displayed. This setting is the same as DIGital.<br>• `DNUMeric`: sets the window display to dual numeric.<br>• `TRACe`: trace display using the currently SELected measurement. Used to determine the channel from which the trace is taken. | `DIGital`<br>`ANALog`<br>`SNUMeric`<br>`DNUMeric`<br>`TRACe` |

**Example**

| | |
|---|---|
| `DISP:WIND2:FORM DIG` | *This command sets the lower window to a digital display.* |

**Reset Condition**

On reset, the N1911A power meter upper window is `DIGital` and the lower window `ANALog`. For the N1912A power meter, the defaults for the upper and lower windows are `DIGital`.

**Query**

`DISPlay:[WINDow[1]|2]:FORMat?`

The query returns the current format of the selected window.

**Query Example**

| | |
|---|---|
| `DISP:FORM?` | *This command queries the current format of the upper window.* |

**Error Messages**

- If the command is set to `TRACe` and the selected channel from which `TRACe` is taken has no sensor connected or has on a sensor other than a P-series or E9320 power sensor connected, error -241, “Hardware missing” occurs.
- If the command is set to `TRACe` and the selected channel has a P-series or E9320 power sensor connected in `AVERage` measurement mode, the error -221, “Settings conflict” occurs.

# DISPlay[:WINDow[1]|2]:METer Commands

These commands control the upper and lower scale limits of the analog meter.

The following commands are detailed in this section:

`DISPlay[:WINDow[1]|2]:METer:LOWer <numeric_value>`

`DISPlay[:WINDow[1]|2]:METer:UPPer <numeric_value>`

# DISPlay[:WINDow[1]|2]:METer:LOWer <numeric_value>

This command sets the analog meter lower scale limit.

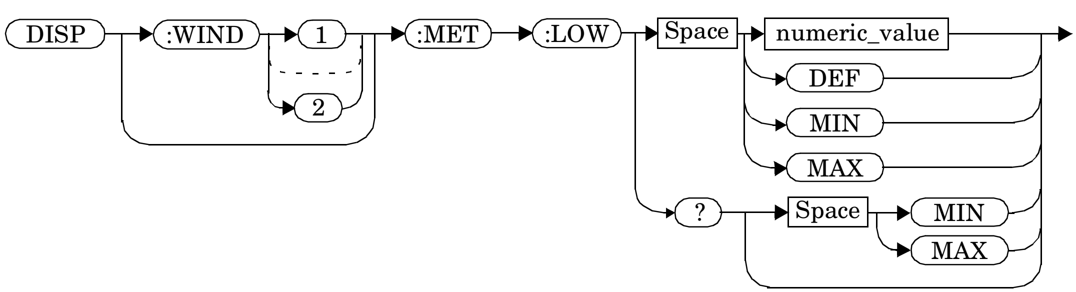
**NOTE** This command has the same purpose as `DISPlay[:WINDow[1]|2]:ANALog:LOWer <numeric_value>`

The units used are dependent on the current setting of `UNIT:POWer` and `CALCulate:RELative:STATe` as shown in Table 5-3.

**Table 5-3 Measurement Units**

| Measurement Mode | Measurement Type | CALC:REL:STAT OFF | | CALC:REL:STAT ON | |
|---|---|---|---|---|---|
| | | Linear | Log | Linear | Log |
| **Single Channel** | Avg, Pk | Watt | dBm | % | dB |
| | Pk-Avg | % | dB | % | dB |
| **Ratio** | Avg, Pk, Pk-Avg | % | dB | % | dB |
| **Difference** | Avg, Pk | Watt | dBm | % | dB |
| | Pk-Avg | % | dB | % | dB |

### Syntax



### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value for the analog meter lower scale limit:<br>• `DEF`: the default is 20 dBm<br>• `MIN`: -150 dBm<br>• `MAX`: 230 dBm<br>The default units are defined by `UNIT:POWer` and `CALCulate:RELative:STATe`. | -150 to 230 dBm<br>`DEF`<br>`MIN`<br>`MAX` |

### Example

`DISP:WIND2:MET:LOW 10` — *This command sets the lower window's analog meter lower scale limit.*

### Reset Condition

On reset, the lower scale limit is set to -70 dBm.

### Query

`DISPlay[:WINDow[1]|2]:METer:LOWer? [MIN|MAX]`

The query returns the current setting of the analog meter's lower scale limit or the value associated with `MIN` and `MAX`. The format of the response is <`NR3`>. The units in which the results are returned is dependent on the current setting of `UNIT:POWer` and `CALCulate:RELative:STATe` as shown in Table 5-3.

### Query Example

`DISP:MET:LOW?` — *This command queries the lower scale limit set on the analog meter in the upper window.*

# DISPlay[:WINDow[1]|2]:METer:UPPer <numeric_value>

This command sets the analog meter upper scale limit.

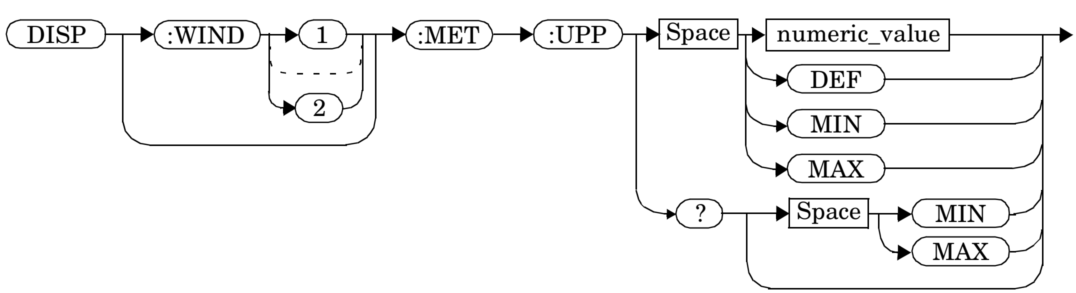
**NOTE** This command has the same purpose as `DISPlay[:WINDow[1]|2]:ANALog:UPPer <numeric_value>`

The units used are dependent on the current setting of `UNIT:POWer` and `CALCulate:RELative:STATe` as shown in Table 5-4.

**Table 5-4 Measurement Units**

| Measurement Mode | Measurement Type | CALC:REL:STAT OFF | | CALC:REL:STAT ON | |
|---|---|---|---|---|---|
| | | Linear | Log | Linear | Log |
| **Single Channel** | Avg, Pk | Watt | dBm | % | dB |
| | Pk-Avg | % | dB | % | dB |
| **Ratio** | Avg, Pk, Pk-Avg | % | dB | % | dB |
| **Difference** | Avg, Pk | Watt | dBm | % | dB |
| | Pk-Avg | % | dB | % | dB |

### Syntax



### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value for the analog meter upper scale limit:<br>• `DEF`: the default is 20 dBm<br>• `MIN`: -150 dBm<br>• `MAX`: 230 dBm<br>Units used are determined by the current setting of `UNIT:POWer` and `CALCulate:RELative:STATe` as shown in Table 5-4. | -150 to 230 dBm<br>`DEF`<br>`MIN`<br>`MAX` |

### Example

`DISP:WIND2:MET:UPP 20` *This command sets the lower window's analog meter upper scale limit.*

### Reset Condition

On reset, the upper scale limit is set to 20 dBm.

### Query

`DISPlay[:WINDow[1]|2]:METer:UPPer? [MIN|MAX]`

The query returns the current setting of the analog meter's upper scale limit or the value associated with `MIN` and `MAX`. The format of the response is <NR3>. The units in which the results are returned is dependent on the current setting of `UNIT:POWer` and `CALCulate:RELative:STATe` as shown in the previous table.

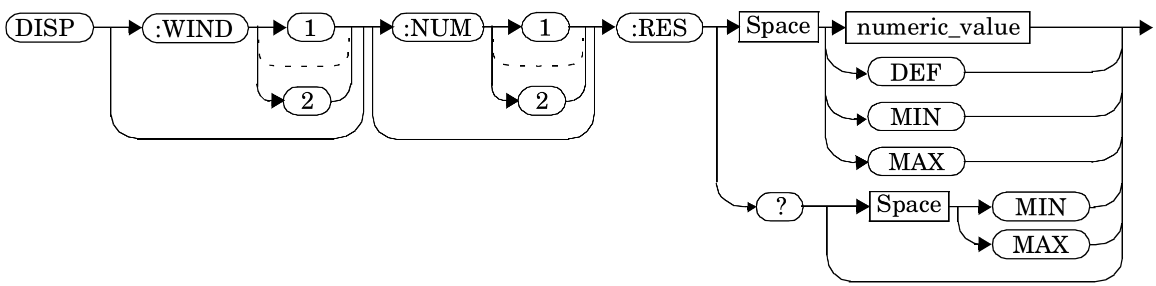
### Query Example

`DISP:WIND2:MET:UPP?` *This command queries the upper scale limit set on the analog meter in the lower window.*

# DISPlay[:WINDow[1]|2][:NUMeric[1]|2]:RESolution <numeric_value>

This command sets the resolution of the measurement result in the specified window.

## Syntax




## Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value for the window resolution:<br>• DEF: 3<br>• MIN: 1<br>• MAX: 4 | 1 to 4<br>DEF<br>MIN<br>MAX |

## Example

`DISP:WIND2:RES 4` *This command sets the lower window's resolution to four significant digits if the measurement result is linear, or to 0.001 if the measurement result is logarithmic.*

**Reset Condition** On reset, the resolution is set to 3.

**Query**

`DISPlay[:WINDow[1]|2]:RESolution? [MIN|MAX]`

The query returns the current setting of the window's resolution or the value associated with `MIN` and `MAX`. The format of the response is `<NR1>`.

**Query Example**

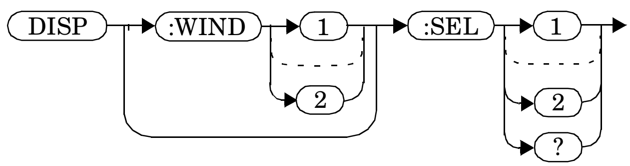
| | |
|---|---|
| `DISP:WINDow1:NUMber2RES?` | *This command queries the resolution setting of the upper window/lower measurement.* |

# DISPlay[:WINDow[1]|2]:SELect[1]|2

This command is used to select a specific measurement within a specific window.

If the second numeric value is not sent, the upper measurement of the relevant window is selected. This command is used to specify which measurement is used for the analog, trace, or single numeric display.

### Syntax




### Example

| | |
|---|---|
| `DISP:WIND2:SEL1` | *This command selects the upper measurement in the lower window.* |

### Reset Condition

On reset, the upper window upper measurement is selected.

### Query

`DISPlay[:WINDow[1]|2]:SELect[1]|2?`

The query enters a `1` or `0` into the output buffer indicating whether the window specified is currently selected.

- `1` is returned if the specified window is selected.
- `0` is returned if the specified window is not selected.

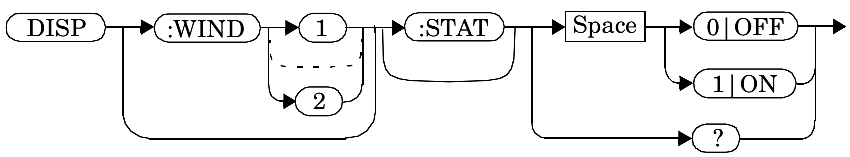
### Query Example

| | |
|---|---|
| `DISP:SEL1?` | *This command queries whether or not the upper measurement in the upper window is selected.* |

# DISPlay[:WINDow[1]|2][:STATe] <boolean>

This command enables/disables the upper or lower window (`WINDow1` and `WINDow2` respectively) so that the display shows a single window only. The displayed window is presented in expanded format, showing a single measurement only: either the single measurement that was shown on the window, or the currently selected measurement, if two measurements had been shown.

### Syntax




### Examples

| | |
|---|---|
| `DISP:WIND2:STAT OFF` | *This command disables the lower window. The upper window in shown in expanded format, displaying its currently selected measurement.* |
| `DISP:WIND2:STAT 1` | *This command enables the lower window so that a dual window display is once more provided.* |

### Reset Condition

On reset, both windows are enabled.

### Query

`DISPlay[:WINDow[1]|2]:STATe?`

This enters a `1` or `0` in the output buffer indicating the selected window.

- `1` is returned if the window is enabled.
- `0` is returned if the window is disabled.

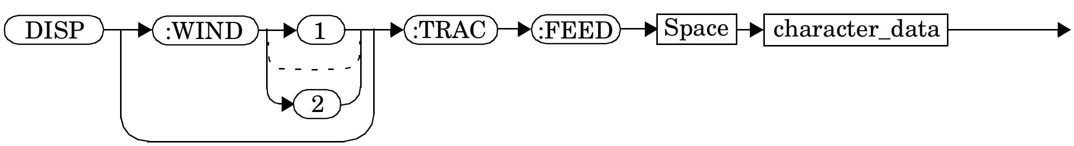
### Query Example

| | |
|---|---|
| `DISP:WIND2:STAT?` | *This command queries whether or not the lower window is displayed.* |

# DISPlay[:WINDow[1]|2]:TRACe:FEED <character_data>

This command selects which channel's trace is displayed in the specified window.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | Identifies which channel's trace is displayed.<br>• `SENS1`: channel A.<br>• `SENS2`: channel B. | "SENS1"<br>"SENS2" |

### Example

`DISP:WIND2:TRAC:FEED "SENS1"` *This command selects channel A's trace to be displayed in the lower window.*

### Reset Condition

On reset, the value is set to:

- Upper window: `SENS1`.
- Lower window (dual channel only): `SENS2`.

### Query

`DISPlay:[WINDow[1]|2]:TRACe:FEED?`

The query returns the channel of the trace currently displayed in the specified window.

### Query Example

| | |
|---|---|
| DISP:WIND2:TRAC:FEED? | *This command queries the channel of the trace currently displayed in the lower window.* |

# 6 FORMat Subsystem

# FORMat Subsystem

The FORMat subsystem sets a data format for transferring numeric information. This data format is used only for response data by commands that are affected by the FORMat subsystem.

The queries affected are:

- FETCh?
- READ?
- MEASure?

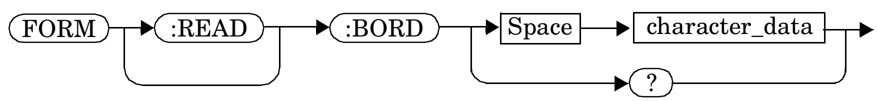
For the N1912A power meter the same FORMat is used on both channels.

| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
| FORMat | | | |
| [:READings] | | | |
| :BORDer | <character_data> | | page 215 |
| [:DATA] | <character_data> | | page 216 |

# FORMat[:READings]:BORDer <character_data>

This command controls whether the binary data is transferred in normal or swapped Byte ORDer. It is only used when `FORMat[:READings][:DATA]` is set to `REAL`.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | Byte order of binary data transfer:<br>• `NORMal`<br>• `SWAPped` | `NORMal`<br>`SWAPped` |

### Example

`FORM:BORD SWAP` *This command sets the byte order to swapped.*

### Reset Condition

On reset, this value is set to `NORMal`.

### Query

`FORMat[:READings]:BORDer?`

The query returns the current setting of the byte order. The format of the response is `NORMal` or `SWAPped`.

### Query Example

`FORM:BORD?` *This command queries the current byte order setting.*

# FORMat[:READings][:DATA] <character_data>

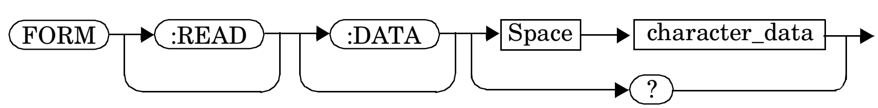
This command sets the data format for transferring numeric information to either `ASCii` or `REAL`:

- When the format type is ASCii, numeric data is output as ASCII bytes in the `<NR3>` format.
- When the format type is `REAL`, numeric data is output as IEEE 754 64 bit floating point numbers in a definite length block. The result is an 8 byte block per number. Each complete block is terminated by a line feed character.

For the N1912A power meter the same `FORMat` is used on both channels.

**NOTE** `FORMat` data formatting is not affected by `TRACe` subsystem data formatting.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | Data format for transferring data:<br>• `ASCii`<br>• `REAL` | `ASCii`<br>`REAL` |

### Example

| | |
|---|---|
| `FORM REAL` | *This command sets the format to* `REAL`. |

**Reset Condition**

On reset, the format is set to `ASCii`.

**Query**

`FORMat[:READings][:DATA]?`

The query returns the current setting of format: either `ASCii` or `REAL`.

**Query Example**

| | |
|---|---|
| `FORM?` | *This command queries the current format setting.* |

# 7 MEMory Subsystem

# MEMory Subsystem

The MEMory command subsystem is used to:

- Edit and review sensor calibration tables (8480 series sensors only).
- Store sensor calibration tables (8480 series sensors only).
- Edit and review sensor frequency dependent offset tables.
- Store sensor frequency dependent offset tables.
- Edit and review sensor save/recall registers.

Stored tables remain in the power meter's memory during power down. The power meter is capable of storing 20 sensor calibration tables and 10 frequency dependent offset tables of 80 frequency points each.

**NOTE** The MEMory subsystem is not used for E-series and P-series power sensors calibration tables. These are automatically downloaded to the power meter and cannot be reviewed or edited.

| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
| MEMory | | | |
| :CATalog | | | |
| [:ALL]? | | [query only] | page 223 |
| :STATe? | | [query only] | page 225 |
| :TABLe? | | [query only] | page 226 |
| :CLEar | | | |
| [:NAME] | <character_data> | [no query], [non-SCPI] | page 229 |
| :TABLe | | [no query] | page 230 |
| :FREE | | | |
| [:ALL]? | | [query only] | page 232 |
| :STATe? | | [query only] | page 233 |

| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
| :TABLe? | | [query only] | page 234 |
| :NSTates? | | [query only] | page 235 |
| :STATe | | | |
| :CATalog? | | [query only] | page 237 |
| :DEFine | <character_data> [,<numeric_value>] | [non-SCPI] | page 238 |
| :TABLe | | | |
| :FREQuency | <numeric_value> [,<numeric_value>] | | page 241 |
| :POINts? | | [query only] | page 241 |
| :GAIN | | | |
| [:MAGNitude] | <numeric_value> [,<numeric_value>] | [non-SCPI] | page 245 |
| :POINts? | | [query only], [non-SCPI] | page 247 |
| :MOVE | <character_data>, <character_data> | [no query], [non-SCPI] | page 248 |
| :SELect | <character_data> | [no query], [non-SCPI] | page 249 |

## MEMory:CATalog Commands

These commands are used to query information on the current contents of a power meter's:

- Sensor calibration tables (8480 series sensors only).
- Frequency dependent offset tables.
- Save/recall registers.

The following commands are detailed in this section:

```
MEMory:CATalog[:ALL]?
MEMory:CATalog:STATe?
MEMory:CATalog:TABLe?
```

# MEMory:CATalog[:ALL]?

This command lists stored sensor calibration tables (8480 series sensors only), frequency dependent offset tables and save/recall registers.

The power meter returns the data in the form of two numeric parameters and as many strings as there are stored tables and save/recall registers:

`<numeric_value>,<numeric_value>{,<string>}`

- The first numeric parameter indicates the amount of memory, in bytes, used for the storage of tables and registers.
- The second numeric parameter indicates the memory, in bytes, available for the storage of tables and registers.
- Each string parameter returned indicates the name, type and size of a stored table or save/recall register:
  - `<string>,<type>,<size>`
    - `<string>` indicates the name of the table or save/recall register.
    - `<type>` indicates `TABL` for sensor calibration and frequency dependent offset tables, or `STAT` for a save/recall register.
    - `<size>` indicates the size of the table or save/recall register in bytes.

A sample of a response may look like the following:

```
1178,26230,"DEFAULT,TABL,14","8481A,TABL,116",
"8482A,TABL,74",..........."State0,STAT,1619",
"State1,STAT,1619","State2,STAT,1619" ...........
```

The power meter is shipped with a set of predefined sensor calibration tables. The data in these sensor calibration tables is based on statistical averages for a range of Agilent Technologies power sensors. These tables can be edited. The predefined data is listed in your *User's Guide*. These power sensors and table numbers are listed in Table 7-1.

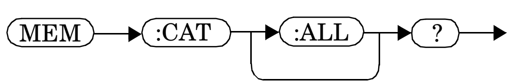
**Table 7-1 8480 Series Power Sensor Tables**

| Table | Power Sensor | Table Name |
|---|---|---|
| 0 | None | DEFAULT[a] |
| 1 | 8481A | 8481A |
| 2 | 8482A, 8482B, 8482H | 8482A |
| 3 | 8483A | 8483A |
| 4 | 8481D | 8481D |
| 5 | 8485A | 8485A |
| 6 | R8486A | R8486A |
| 7 | Q8486A | Q8486A |
| 8 | R8486D | R8486D |
| 9 | 8487A | 8487A |

a. Default is a sensor calibration table in which the reference calibration factor and calibration factors are 100%. This sensor calibration table can be used during the performance testing of the power meter.

There are also ten sensor calibration tables named CUSTOM_0 through CUSTOM_9 and ten frequency dependent offset tables named CUSTOM _A through CUSTOM _J which do not contain any data when the power meter is shipped from the factory.

### Syntax




### Example

| | |
|---|---|
| `MEM:CAT?` | *This command queries the list of tables and save/recall registers.* |

# MEMory:CATalog:STATe?

This command is used to list the save/recall registers.

The power meter returns the data in the form of two numeric parameters and as many strings as there are save/recall registers.

<numeric_value>,<numeric_value>{,<string>}

- The first numeric parameter indicates the amount of memory, in bytes, used for the storage of registers.
- The second parameter indicates the memory, in bytes, available for the storage of registers.
- Each string parameter returned indicates the name, type and size of a save/recall register:

  — <string>,<type>,<size>

    — <string> indicates the name of the save/recall register.

    — <type> indicates STAT for save/recall register.

    — <size> indicates the size of the save/recall register in bytes.

For example, a sample of a response may look like:

```
0,16190,"State0,STAT,0","State1,STAT,0" .........
```

## Syntax

MEM → :CAT → :STAT → ? →

## Example

| | |
|---|---|
| MEM:CAT:STAT? | *This command queries the list of save/recall registers.* |

# MEMory:CATalog:TABLe?

This command is used to list the stored sensor calibration (8480 series sensors only) and frequency dependent offset tables.

The power meter returns the data in the form of two numeric parameters and as many strings as there are stored tables.

<numeric_value>,<numeric_value>{,<string>}

- The first numeric parameter indicates the amount of memory, in bytes, used for the storage of tables.
- The second parameter indicates the memory, in bytes, available for the storage of tables.
- Each string parameter returned indicates the name, type and size of a stored table:

  — `<string>,<type>,<size>`

    — `<string>` indicates the name of the table.

    — `<type>` indicates `TABL` for a table.

    — `<size>` indicates the size of the table in bytes.

For example, a sample of a response may look like:

```
1178,10040,"DEFAULT,TABL,14","8481A,TABL,116",
"8482A,TABL,74","8483A,TABL,62"...........
```

The power meter is shipped with a set of predefined sensor calibration tables. The data in these sensor calibration tables is based on statistical averages for a range of Agilent Technologies power sensors. These tables can be edited. The predefined data is listed in your *User's Guide*. These power sensors and table numbers are listed in Table 7-2.

**Table 7-2 8480 Series Power Sensor Tables**

| Table | Power Sensor | Table Name |
|---|---|---|
| 0 | None | DEFAULT[a] |
| 1 | 8481A | 8481A |
| 2 | 8482A, 8482B, 8482H | 8482A |
| 3 | 8483A | 8483A |
| 4 | 8481D | 8481D |
| 5 | 8485A | 8485A |
| 6 | R8486A | R8486A |
| 7 | Q8486A | Q8486A |
| 8 | R8486D | R8486D |
| 9 | 8487A | 8487A |

a. Default is a sensor calibration table in which the reference calibration factor and calibration factors are 100%. This sensor calibration table can be used during the performance testing of the power meter.

There are also ten sensor calibration tables named CUSTOM_0 through CUSTOM_9 and ten frequency dependent offset tables named CUSTOM_A through CUSTOM_J which do not contain any data when the power meter is shipped from the factory.

### Syntax

MEM → :CAT → :TABL → ? →

### Example

`MEM:CAT:TABL?` *This command queries the list of stored tables.*

# MEMory:CLEar Commands

These commands are used to remove the contents stored in the sensor calibration tables (8480 series sensors only), frequency dependent offset tables and save/recall registers. This subsystem removes the data contents but does not affect the name of the associated table or save/recall register.

The following commands are detailed in this section:

```
MEMory:CLEar:[NAME] <character_data>
```

```
MEMory:CLEar:TABLe
```

**NOTE** The contents cleared using these commands are non-recoverable.

# MEMory:CLEar[:NAME] <character_data>

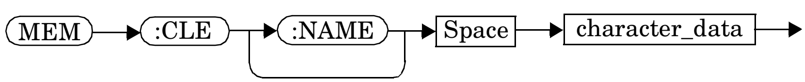
This command clears the contents of a specified sensor calibration table (8480 series sensors only), frequency dependent offset table, or save/recall register.

Although the table remains, a `MEMory:TABLe:FREQuency|GAIN:POINts?` query returns a `0` as there are no contents in the table.

For sensor calibration tables and frequency dependent offset tables, this command is an alternative form of the `MEMory:CLEar:TABLE` command, the only difference being the method in which the table is selected.

**NOTE** The contents cleared using this command are non-recoverable.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | Contains an existing table name or save/recall register. | Any existing table name or save/recall register. |

### Example

`MEM:CLE "8485A"` *This command clears the contents of sensor calibration table 8485A*

### Error Messages

If the table or save/recall register name does not exist, error -224, "Illegal parameter value" occurs.

# MEMory:CLEar:TABle

This command is used to clear the contents of the table currently selected using `MEMory:TABLe:SELect`. Although the table remains, a `MEMory:TABLe:FREQuency|GAIN:POINts?` query returns a `0` as the table contents are empty.

This command is an alternative form of the `MEMory:CLEar[:NAME]` command. The difference is the method in which the table is selected.

**NOTE** The contents cleared using this command are non-recoverable.

### Syntax

MEM → :CLE → :TABL →

### Example

| | |
|---|---|
| `MEM:CLE:TABL` | *This command clears the contents of the currently selected table.* |

### Error Message

If no table is selected, error -221, “Settings conflict” occurs.

# The MEMory:FREE Commands

These commands are used to return information on the amount of free memory space available for sensor calibration tables (8480 series sensors only), frequency dependent offset tables, and save/recall registers.

The following commands are described in this section:

`MEMory:FREE[:ALL]?`

`MEMory:FREE:STATe?`

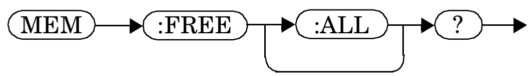
`MEMory:FREE:TABLe?`

# MEMory:FREE[:ALL]?

This query returns the amount of memory free for sensor calibration tables (8480 series sensors only), frequency dependent offset tables, and save/recall registers. The format of the response is:

<bytes_available>,<bytes_in_use>

### Syntax




### Example

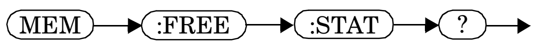
| | |
|---|---|
| `MEM:FREE?` | *This command queries the amount of free memory in total.* |

# MEMory:FREE:STATe?

This query returns the amount of memory free for save/recall registers. The format of the response is:

`<bytes_available>,<bytes_in_use>`

### Syntax




### Example

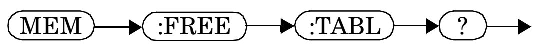
| | |
|---|---|
| `MEM:FREE:STAT?` | *This command queries the amount of free memory for save/recall registers.* |

# MEMory:FREE:TABLe?

This query returns the amount of memory free for sensor calibration tables (8480 series sensors only) and frequency dependent offset tables. The format of the response is:

`<bytes_available>,<bytes_in_use>`

### Syntax




### Example

| | |
|---|---|
| `MEM:FREE:TABL?` | *This command queries the amount of free memory for tables.* |

# MEMory:NSTates?

This query returns the number of registers that are available for save/recall. As there are ten registers this query always returns ten.

### Syntax


### Example

| | |
|---|---|
| `MEM:NST?` | *This command queries the number of registers available for save/recall.* |

# The MEMory:STATe Commands

These commands are used to query and define register names.

The following commands are described in this section:

`MEMory:STATe:CATalog?`

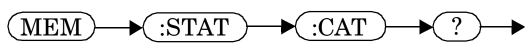
`MEMory:STATe:DEFine`

# MEMory:STATe:CATalog?

This query returns a list of the save/recall register names in ascending order of register number. The format of the response is:

`<string>,<string>,.....,<string>`

### Syntax



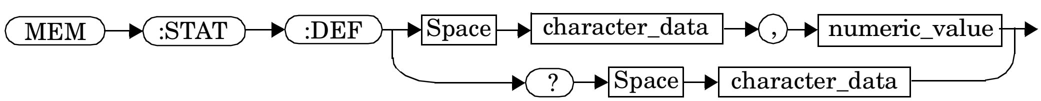

### Example

| | |
|---|---|
| `MEM:STAT:CAT?` | *This command queries the register names.* |

# MEMory:STATe:DEFine <character_data>,<numeric_value>

This command is used to associate a name with a save/recall register number.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | Details the register name. A maximum of 12 characters can be used. | A to Z (uppercase)<br>a to z (lowercase)<br>0 - 9<br>_ (underscore) |
| numeric_value | A numeric value (`<NRf>`) for the register number. | 0 to 9 |

### Example

`MEM:STAT:DEF "SETUP1",4` *This command names register 4 SETUP1.*

### Query

`MEMory:STATe:DEFine? <string>`

The query returns the register number for the given register name.

### Query Example

`MEM:STAT:DEF? "SETUP1"` *This command queries the register number of SETUP1.*

### Error Messages

- If the register number is out of range, error -222, “Data out of range” occurs.

- If the name is invalid, error -224, "Illegal parameter value" occurs.
- If a register or sensor calibration table with the same name already exists, error -257, "File name error" occurs (command only).

# MEMory:TABLe Commands

These commands are used to define a sensor calibration table (8480 series sensors only) or a frequency dependent offset table, and to write to and read data from it.

The following commands are described in this section:

```
MEMory:TABLe:FREQuency <numeric_value>{,<numeric_value>}
```

```
MEMory:TABLe:FREQuency:POINts?
```

```
MEMory:TABLe:GAIN[:MAGNitude]
<numeric_value>{,<numeric_value>}
```

```
MEMory:TABLe:GAIN[:MAGNitude]:POINts?
```

```
MEMory:TABLe:MOVE <character_data>,<character_data>
```

```
MEMory:TABLe:SELect <character_data>
```

# MEMory:TABLe:FREQuency <numeric_value>{,<numeric_value>}

This command is used to enter frequency data into the current selected table. Any previous frequency list is cleared before the new frequency list is stored. The frequencies must be entered in ascending order. Entries in the frequency lists correspond as shown in Table 7-3 with entries in the calibration/offset factor lists.

**NOTE** For sensor calibration tables only, the first calibration factor entered using the `MEMory:TABLe:GAIN` command is used as the reference calibration factor.

**Table 7-3 Frequency and Calibration/Offset Factor List**

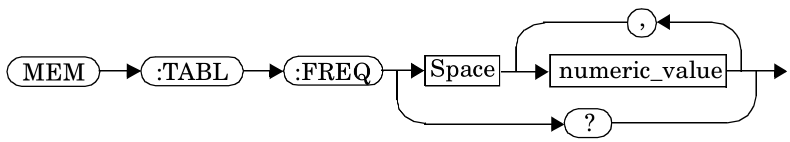
| Frequency | Calibration Factor/Offset |
|---|---|
| - | Reference Calibration Factor (For Sensor Calibration Tables) |
| Frequency 1 | Calibration Factor/Offset 1 |
| Frequency 2 | Calibration Factor/Offset 2 |
| " | " |
| Frequency 80 | Calibration Factor/Offset 80 |

For sensor calibration tables (8480 series sensors only), the number of frequency points must be one less than the number of calibration factor points. This is verified when the sensor calibration table is selected using `SENSe:CORRection:CSET:SELect <string>`.

Ensure that the frequency points you use cover the frequency range of the signals that you want to measure. If you measure a signal with a frequency outside the frequency range defined in the table, then the power meter uses the highest or lowest point in the table to calculate the calibration factor/offset.

Depending on available memory, the power meter is capable of storing 20 sensor calibration tables and 10 frequency dependent offset tables, each containing 80 points.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value for the frequency. The default units are Hz. | 1 kHz to 1000.0 GHz[a, b] |

a. The following measurement units can be used:

Hz
kHz ($10^3$)
MHz ($10^6$)
GHz ($10^9$)

b. All frequencies are truncated to a multiple of 1 kHz.

### Example

| | |
|---|---|
| `MEM:TABL:FREQ 200kHz,600kHz` | *This command enters frequencies of 200 kHz and 600 kHz into the currently selected table.* |

### Query

`MEMory:TABLe:FREQuency?`

The query returns a list of frequency points for the table currently selected. The frequencies are returned in Hz.

### Query Example

| | |
|---|---|
| `MEM:TABL:FREQ?` | *This command queries the frequency points in the currently selected table.* |

**Error Messages**

- If more than 80 frequencies are in the list, error -108, “Parameter not allowed” occurs.
- If the frequencies are not entered in ascending order, error -220, “Parameter error;Frequency list must be in ascending order” occurs.
- If a table has not been specified using the `MEMory:TABLe:SELect` command, the data cannot be entered into the table and error -221, “Settings conflict” occurs.
- If a frequency is sent which is outside of the allowed frequency range, error -222, “Data out of range” occurs.

# MEMory:TABLe:FREQuency:POINts?

This query returns the number of frequency points for the table currently selected. The response format is `<NRf>`. If no frequency values have been set, this command returns `0`. If no table is selected, this command returns `NAN`.

### Syntax

MEM → :TABL → :FREQ → :POIN → ? →

### Example

| | |
|---|---|
| `MEM:TABL:FREQ:POIN?` | *This command queries the number of frequency points in the current table.* |

# MEMory:TABLe:GAIN[:MAGNitude] <numeric_value>{,<numeric_value>}

This command is used to enter calibration factors into the sensor calibration table (8480 series sensors only) or offsets into the frequency dependent offset table, currently selected using `MEMory:TABLe:SELect`. Any previous calibration factor list, or offset list is cleared before the new calibration factors/offsets are stored.

A maximum of 81 parameters for sensor calibration tables and 80 parameters for frequency dependent offset tables can be sent with this command. For sensor calibration tables only, the first parameter is the reference calibration factor, each subsequent parameter is a calibration factor point in the sensor calibration table.

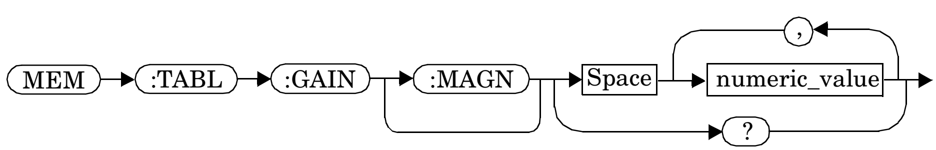
Entries in the frequency lists correspond as shown in Table 7-4 with entries in the calibration/offset factor lists.

**Table 7-4 Frequency and Calibration/Offset Factor List**

| Frequency | Calibration Factor/Offset |
|---|---|
| - | Reference Calibration Factor (For Sensor Calibration Tables) |
| Frequency 1 | Calibration Factor/Offset 1 |
| " | " |
| Frequency 80 | Calibration Factor/Offset 80 |

For sensor calibration tables the number of frequency points must be one less than the number of calibration factor data points. This is verified when the sensor calibration table is selected using `SENSe:CORRection:CSET1:SELect <string>`.

### Syntax



### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value for the calibration/ offset factors. The units are PCT. | 1.0 to 150.0 |

### Example

```
MEM:TABL:SEL "Sensor_1"
MEM:TABL:GAIN 97,99.5,97.4
```

*This command enters a reference calibration factor of 97% and calibration factors of 99.5% and 97.4% into the sensor calibration table.*

### Query

`MEMory:TABLe:GAIN[:MAGNitude]?`

The query returns a list of calibration factor/offset points for the currently selected table.

### Query Example

```
MEM:TABL:GAIN?
```

*This command queries the calibration factor/offset in the current table.*

### Error Messages

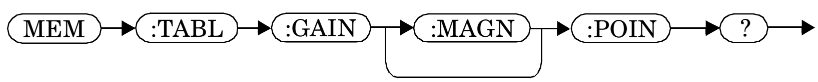
- If more than 81 calibration factors for sensor calibration tables, or 80 offsets for frequency dependent offset tables are in the list, error -108, “Parameter not allowed” occurs.
- If a table is not specified using the `MEMory:TABLe:SELect` command, the data cannot be entered and error -221, “Settings conflict” occurs.
- If any of the calibration/offset factors are outside of the allowed range, error -222, “Data out of range” occurs.

# MEMory:TABLe:GAIN[:MAGNitude]:POINts?

This query is used to return the number of calibration factor/offset points for the currently selected table. If the currently selected table is a sensor calibration table (8480 series sensors only), the reference calibration factor is included

If no values have been set, `0` is returned. If no table is selected, `NAN` is returned.

### Syntax




### Example

| | |
|---|---|
| `MEM:TABL:GAIN:POIN?` | *This command queries the number of calibration factor / offset points in the current table.* |

# MEMory:TABLe:MOVE <character_data>,<character_data>

This command is used to rename a sensor calibration table (8480 series sensors only) or a frequency dependent offset table.

### Syntax

MEM → :TABL → :MOVE → Space → character_data → , → character_data →

### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data 1st parameter) | Contains the existing table name. | existing table name |
| character_data( 2nd parameter) | Details the new table name. A maximum of 12 characters can be used. | A to Z (uppercase)<br>a to z (lowercase)<br>0 - 9<br>_ (underscore) |

### Example

```
MEM:TABL:MOVE "tab1","tab1a"
```

*This command renames a table named tab1 to tab1a.*

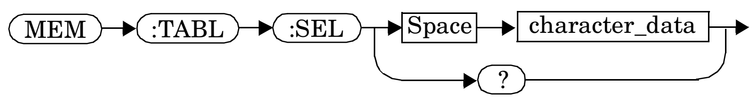
### Error Messages

- If either table name is invalid, error -224, “Illegal parameter value” occurs.
- If the first parameter does not match an existing table name, error -256, “File name not found” occurs.
- If the second parameter matches an existing table name or save/recall register, error -257, “File name error” occurs.

# MEMory:TABLe:SELect <character_data>

This command is used to activate either a sensor calibration table (8480 series sensors only), or a frequency dependent offset table. A table must be activated before any operation can be performed on it.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | Details the new table name. A maximum of 12 characters can be used. | A to Z (uppercase)<br>a to z (lowercase)<br>0 - 9<br>_ (underscore) |

### Example

| | |
|---|---|
| `MEM:TABL:SEL "Sensor1"` | *This command selects a sensor calibration table named "Sensor1".* |

### Query

`MEMory:TABLe:SELect?`

The query returns the name of the currently selected table.

# 8 OUTput Subsystem

# OUTPut Subsystem

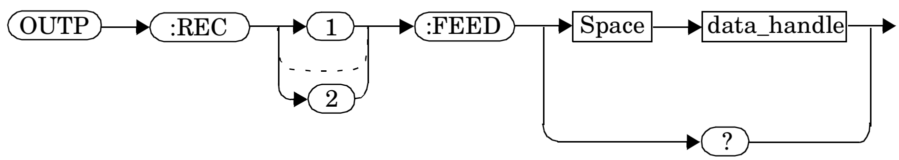
The OUTPut command subsystem is used to control the trigger output, switch on and off the POWER REF output, and controls the recorder output.

| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
| OUTPut | | | |
| :RECorder[1]\|2 | | | |
| :FEED | <data_handle> | | page 253 |
| :LIMit | | | |
| :LOWer | <numeric_value> | | page 254 |
| :UPPer | <numeric_value> | | page 255 |
| :STATe | <boolean> | | page 256 |
| :ROSCillator | | | |
| [:STATe] | <boolean> | | page 257 |
| :TRIGger | | | |
| [:STATe] | <boolean> | | page 258 |

# OUTPut:RECorder[1]|2:FEED <data_handle>

This command specifies which measurement is sent to the recorder output specified by the numeric value following `RECorder`. `RECorder1` applies to both single and dual channel power meters. `RECorder2` applies to dual channel power meters only.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| data_handle | The CALC block specifying the measurement to be sent to the recorder output. | `"CALC1"` or `"CALC"`<br>`"CALC2"`<br>`"CALC3"`<br>`"CALC4"` |

### Example

| | |
|---|---|
| `OUTP:REC2:FEED "CALC1"` | *This command sends the CALC1 measurement to recorder output 2.* |

### Reset Condition

On reset, `data_handle` is set to its previous value.

### Query

`OUTPut:RECorder[1]|2:FEED?`

The query command returns the current value of `data_handle`.

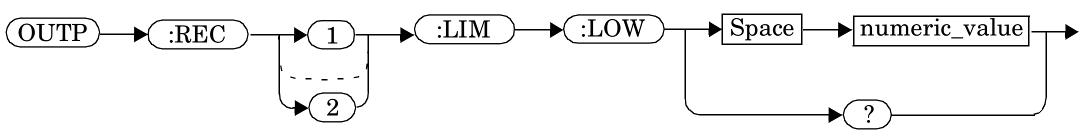
### Query Example

| | |
|---|---|
| `OUTP:REC2:FEED?` | *This command queries the value of data_handle for recorder output 2.* |

# OUTPut:RECorder[1]|2:LIMit:LOWer <numeric_value>

This command sets the minimum scaling value for the specified recorder output. The units used are dependent on the units currently set for the CALC block specified in `OUTPut:RECorder[1]|2:FEED <data_handle>`.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value for the minimum scaling value. The units used—dBm, W or %—are dependent on the units currently set for the CALC block specified in `OUTPut:RECorder[1]\|2:FEED <data_handle>`. | -150 to +230 dBm<br>1 aW to 100 XW<br>0% to 999% |

### Example

`OUTP:REC:LIM:LOW -90` *This command sets the minimum scaling value to -90.*

**Reset Condition** On reset, the minimum scaling value is set to -150 dBm.

**Query** `OUTPut:RECorder[1]|2:LIMit:LOWer?`

The query command returns the minimum scaling value.

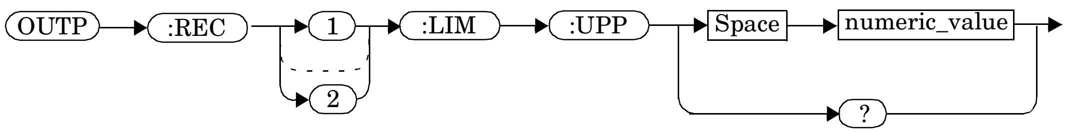
### Query Example

`OUTP:REC:LIM:LOW?` *This command returns the minimum scaling value for the specified recorder output.*

# OUTPut:RECorder[1]|2:LIMit:UPPer <numeric_value>

This command sets the maximum scaling value for the specified recorder output. The units used are dependent on the units currently set for the CALC block specified in `OUTPut:RECorder[1]|2:FEED <data_handle>`.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value for the minimum scaling value. The units used—dBm, W or %—are dependent on the units currently set for the CALC block specified in `OUTPut:RECorder[1]|2:FEED <data_handle>`. | -150 to +230 dBm<br>1 aW to 100 XW<br>0% to 999% |

### Example

| | |
|---|---|
| `OUTP:REC:LIM:UPP 10` | *This command sets the maximum scaling value to 10.* |

**Reset Condition** On reset, the maximum scaling value is set to +20 dBm.

**Query** `OUTPut:RECorder[1]|2:LIMit:UPPer?`

The query command returns the maximum scaling value.

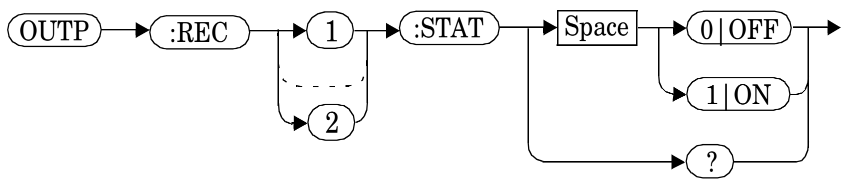
### Query Example

| | |
|---|---|
| `OUTP:REC:LIM:UPP?` | *This command returns the maximum scaling value for the specified recorder output.* |

# OUTPut:RECorder[1]|2:STATe <boolean>

This command enables or disables the specified recorder output.

**Syntax**




**Example**

| | |
|---|---|
| `OUTP:REC1:STAT 1` | *This command enables the specified recorder output.* |

**Reset Condition** On reset, the recorder output is `OFF`.

**Query** `OUTPut:RECorder[1]|2:STATe?`

The query command enters a `1` or `0` into the output buffer indicating whether or not the specified recorder is switched on.

- `1` is returned when the recorder output is switched `ON`.
- `0` is returned when the recorder output is switched `OFF`.

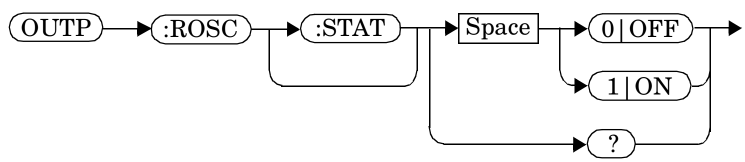
**Query Example**

| | |
|---|---|
| `OUTP:REC2:STAT?` | *This command queries the status of the recorder output.* |

# OUTPut:ROSCillator[:STATe] <boolean>

This command enables/disables the POWER REF output.

**Syntax**




**Example**

| | |
|---|---|
| `OUTP:ROSC:STAT 1` | *This command enables the POWER REF output.* |

**Reset Condition** On reset, the POWER REF output is disabled.

**Query** `OUTPut:ROSCillator[:STATe]?`

The query command enters a `1` or `0` into the output buffer indicating whether or not the POWER REF is enabled.

- `1` is returned when the POWER REF output is enabled.
- `0` is returned when the POWER REF output is disabled.

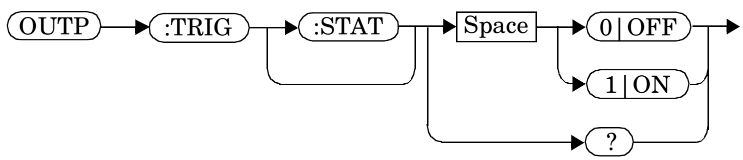
**Query Example**

| | |
|---|---|
| `OUTP:ROSC?` | *This command queries the status of the POWER REF output.* |

# OUTPut:TRIGger[:STATe] <boolean>

This command enables/disables the trigger output signal.

### Syntax




### Example

| | |
|---|---|
| `OUTP:TRIG:STAT 1` | *This command enables the trigger output signal.* |

### Reset Condition

On reset, the trigger output signal is disabled.

### Query

`OUTPut:TRIGger[:STATe]?`

The query command enters a `1` or `0` into the output buffer indicating whether or not the trigger output signal is enabled/disabled.

- `1` is returned when the trigger output signal is enabled.
- `0` is returned when the trigger output signal is disabled.

### Query Example

| | |
|---|---|
| `OUTP:TRIG:STAT?` | *This command queries the status of the trigger output signal.* |

# 9 SENSe Subsystem

# [SENSe] Subsystem

The SENSe command subsystem directly affects device specific settings used to make measurements. The SENSe subsystem is optional since this is the primary function of the power meter. The high level command CONFigure uses the SENSe commands to prepare the power meter for making measurements. At a lower level SENSe enables you to change the following parameters: RANGe, FREQuency, LOSS, CFACator|GAIN1 (calibration factor), GAIN2 (channel offset), DCYCle (duty cycle) and AVERage, without completely re-configuring the power meter.

The SENSe command subsystem also allows you to select the measurement speed, a sensor calibration table, and a frequency dependent offset table.

The numeric suffix of the SENSe program mnemonic in the SENSe commands refers to a channel, that is SENSe1 and SENSe2 represent channel A and channel B respectively.

**NOTE** If you are using the single channel N1911A power meter the SENSe2 commands are irrelevant and cause the error “Header suffix out of range.”

| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
| [SENSe[1]]\|SENSe2 | | | |
| :AVERage | | | |
| :COUNt | <numeric_value> | | page 264 |
| :AUTO | <boolean> | | page 266 |
| :SDETect | <boolean> | [non-SCPI] | page 269 |
| [:STATe] | <boolean> | | page 271 |
| :AVERage2 | | | |
| :COUNt | <numeric_value> | | page 273 |
| [:STATe] | <boolean> | | page 275 |

| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
| :BANDwidth\|BWIDth | | | |
| :VIDeo | <character_data> | | page 276 |
| :CORRection | | | |
| :CFACtor\|GAIN[1] | | [non-SCPI] | |
| [:INPut] | | | |
| [:MAGNitude] | <numeric_value> | | page 279 |
| :CSET[1]\|CSET2 | | | |
| [:SELect] | <string> | | page 282 |
| :STATe | <boolean> | | page 284 |
| :DCYCle\|GAIN3 | | [non-SCPI] | |
| [:INPut] | | | |
| [:MAGNitude] | <numeric_value> | | page 287 |
| :STATe | <boolean> | | page 290 |
| :FDOFfset\|GAIN4 | | | |
| [:INPut] | | | |
| [:MAGNitude] | | [query only] | page 292 |
| :GAIN2 | | | |
| :STATe | <boolean> | | page 294 |
| [:INPut] | | | |
| [:MAGNitude] | <numeric_value> | | page 295 |
| :DETector | | | |
| :FUNCtion | <character_data> | | page 297 |
| :FREQuency | | | |
| [:CW\|:FIXed] | <numeric_value> | | page 299 |
| :MRATe | <character_data> | | page 301 |
| :POWer | | | |
| :AC | | | |
| :RANGe | <numeric_value> | [non-SCPI] | page 303 |
| :AUTO | <boolean> | | page 304 |
| :SWEep[1]\|2\|3\|4 | | | |

| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
|     :OFFSet | | | |
|       :TIME | <numeric_value> | | page 307 |
|     :TIME | <numeric_value> | | page 309 |
|   :TEMPerature? | | [query only] | page 311 |
|   :TRACe | | | |
|     :LIMit | | | |
|       :LOWer | <numeric_value> | | page 313 |
|       :UPPer | <numeric_value> | | page 315 |
|     :OFFSet | | | |
|       :TIME | <numeric_value> | | page 317 |
|     :TIME | <numeric_value> | | page 319 |
|     :UNIT | <character_data> | | page 321 |
|     :X | | | |
|       :SCALe | | | |
|         :AUTO | <numeric_value> | | |
|         :LEFT | <numeric_value> | | |
|         :RIGHt | <numeric_value> | | |
|         :CENTer | <numeric_value> | | |
|         :PDIV | <numeric_value> | | |
|           :LINK | | | |
|     :Y | | | |
|       :SCALe | | | |
|         :AUTO | <numeric_value> | | |
|         :TOP | <numeric_value> | | |
|         :BOTTom | <numeric_value> | | |
|         :RLEVel | <numeric_value> | | |
|         :RPOSition | | | |
|         :PDIV | | | |
|   :V2P | ATYPe\|DTYPe | [non-SCPI] | page 322 |

# [SENSe[1]]|SENSe2:AVERage Commands

These commands control the measurement averaging which is used to improve measurement accuracy. They combine successive measurements to produce a new composite result.

The following commands are detailed in this section:

```
[SENSe[1]]|SENSe2:AVERage:COUNt <numeric_value>
[SENSe[1]]|SENSe2:AVERage:COUNt:AUTO <boolean>
[SENSe[1]]|SENSe2:AVERage:SDETect <boolean>
[SENSe[1]]|SENSe2:AVERage[:STATe] <boolean>
```

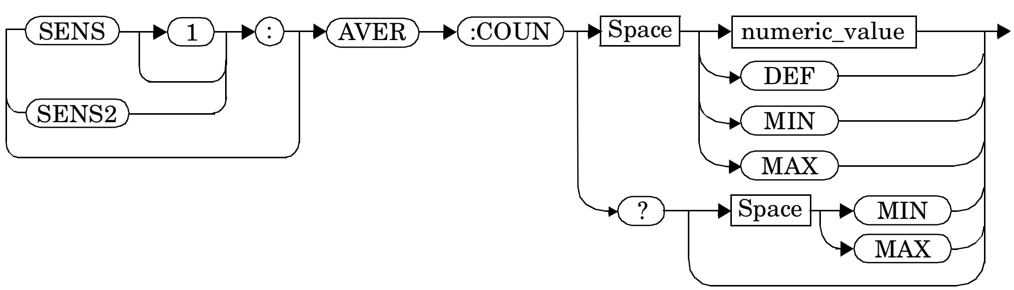
# [SENSe[1]]|SENSe2:AVERage:COUNt <numeric_value>

This command is used to enter a value for the filter length. If [SENSe[1]]|SENSe2:AVERage:COUNt:AUTO is set to ON then entering a value for the filter length automatically sets it to OFF. Increasing the value of filter length increases measurement accuracy but also increases the time taken to make a power measurement.

Entering a value using this command automatically turns the [SENSe[1]]|SENSe2:AVERage:STATe command to ON.

**NOTE** For most applications, automatic filter length selection ([SENSe[1]]|SENSe2:AVERage:COUNt:AUTO ON) is the best mode of operation. However, manual filter length selection ([SENSe[1]]|SENSe2:AVERage:COUNt <numeric_value>) is useful in applications requiring either high resolution or fast settling times, where signal variations rather than measurement noise need filtering, or when approximate results are needed quickly.

### Syntax



#### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value defining the filter length.<br>• DEF: the default value is 4.<br>• MIN: 1.<br>• MAX: 1024. | 1 to 1024<br>DEF<br>MIN<br>MAX |

#### Example

`AVER:COUN 400` *This command enters a filter length of 400 for channel A.*

**Reset Condition** On reset, the filter length is set to 4.

#### Query

`AVERage:COUNt? [MIN|MAX]`

The query returns the current setting of the filter length or the values associated with MIN and MAX. The format of the response is <NR1>.

#### Query Example

`AVER:COUN?` *This command queries the filter length for channel A.*

**Error Messages** If a filter length value is entered using `[SENSe[1]]|SENSe2:AVERage:COUNt` while `[SENSe[1]]|SENSe2:SPEed` is set to 200, the error -221, “Settings Conflict” occurs. However, the filter length value is set but the `[SENSe[1]]|SENSe2:AVERage:STATe` command is not automatically set ON.

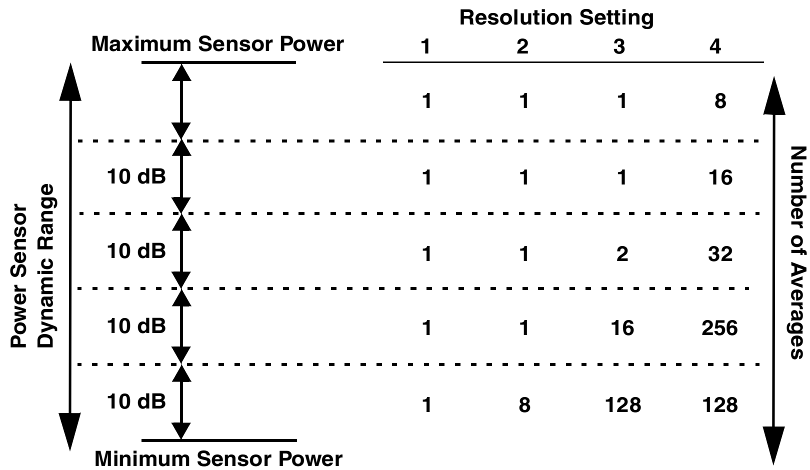
# [SENSe[1]]|SENSe2:AVERage:COUNt:AUTO <boolean>

This command enables and disables automatic averaging. `ONCE` has no affect on the power meter.

When the auto filter mode is enabled, the power meter automatically sets the number of readings averaged together to satisfy the averaging requirements for most power measurements. The number of readings averaged together depends on the resolution and the power level in which the power meter is currently operating. Figure 9-1 is an example of the averaged number of readings for each range and resolution when the power meter is in auto measurement average mode and using a P-series or E932XX power sensor.

Setting this command to `ON` automatically sets the `[SENSe[1]]|SENSe2:AVERage:STATe` command to `ON`.

**Figure 9-1** **Example of Averaged Readings**

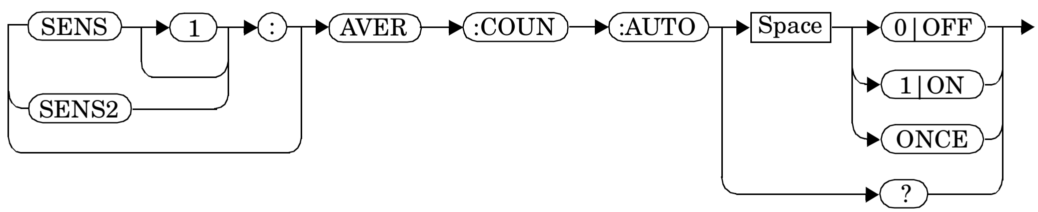



If `[SENSe[1]]|SENSe2:AVERage:COUNt:AUTO` is set to `OFF`, the filter length is set by the `[SENSe[1]]|SENSe2:AVERage:COUNt` command. Using the `[SENSe[1]]|SENSe2:AVERage:COUNt` command disables automatic averaging.

Auto averaging is enabled by the `MEASure:POWer:AC?` and `CONFigure:POWer:AC?` commands.

**NOTE** For most applications, automatic filter length selection (`[SENSe[1]]|SENSe2:AVERage:COUNt:AUTO ON`) is the best mode of operation. However, manual filter length selection (`[SENSe[1]]|SENSe2:AVERage:COUNt <numeric_value>`) is useful in applications requiring either high resolution or fast settling times, where signal variations rather than measurement noise need filtering, or when approximate results are needed quickly.

### Syntax




### Example

| | |
|---|---|
| `AVER:COUN:AUTO OFF` | *This command disables automatic filter length selection for channel A.* |

### Reset Condition

On reset, automatic averaging is enabled.

### Query

`[SENSe[1]]|SENSe2:AVERage:COUNt:AUTO?`

The query enters a `1` or `0` into the output buffer indicating whether automatic filter length is enabled or disabled.

- `1` is returned when automatic filter length is enabled.
- `0` is returned when automatic filter length is disabled.

### Query Example

| | |
|---|---|
| `AVER:COUN:AUTO?` | *This command queries whether automatic filter length selection is on or off for channel A.* |

**Error Messages**

If [SENSe[1]]|SENSe2:AVERage:COUNt:AUTO is set to ON while [SENSe[1]]|SENSe2:SPEed is set to 200, the error -221, “Settings Conflict” occurs. However, automatic averaging is enabled but the [SENSe[1]]|SENSe2:AVERage:STATe command is not automatically set ON.

# [SENSe[1]]|SENSe2:AVERage:SDETect <boolean>

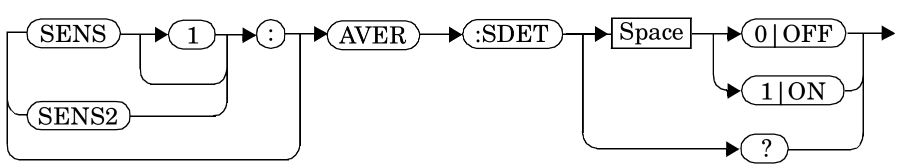
This command enables and disables step detection. In AUTO filter mode, the average of the last four values entered into the filter is compared to the average of the entire filter. If the difference between the two averages is greater than 12.5%, the digital filter is cleared. The filter then starts storing new measurement values. This feature shortens the filter time when the input power changes substantially. for the filter output to get to its final value. Note that this result appears to settle faster, although true settling to the final value is unaffected.

**NOTE** Step detection is automatically disabled when TRIG:DEL:AUTO is ON and the trigger mode is set to free run.

Under this circumstances the value of SENS:AVER:SDET is ignored. Note also that SENS:AVER:SDET is not set by the instrument (that is, SENS:AVER:SDET retains its current setting which may indicate that step detection is ON).

**NOTE** With certain pulsing signals step detect may operate on the pulses, preventing the final average being completed and making the results unstable. Under these conditions SDET should be set to OFF.

### Syntax




### Example

| | |
|---|---|
| SENS:AVER:SDET OFF | *This command disables step detection.* |

**Reset Condition**

On reset, step detection is enabled.

**Query**

`[SENSe[1]]|SENSe2:AVERage:SDETect?`

The query enters a 1 or 0 into the output buffer indicating the status of step detection.

- 1 is returned when step detection is enabled.
- 0 is returned when step detection is disabled.

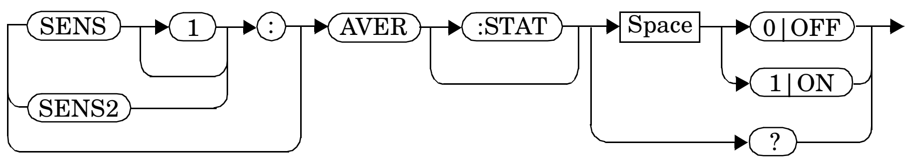
**Query Example**

| | |
|---|---|
| `SENS:AVER:SDET?` | *This command queries whether step detection is on or off.* |

# [SENSe[1]]|SENSe2:AVERage[:STATe] <boolean>

This command is used to enable and disable averaging.

## Syntax




## Example

| | |
|---|---|
| `AVER 1` | *This command enables averaging on channel A.* |

## Reset Condition

On reset, averaging is `OFF`.

## Query

`[SENSe[1]]|SENSe2:AVERage[:STATe]?`

The query enters a `1` or `0` into the output buffer indicating the status of averaging.

- `1` is returned when averaging is enabled.
- `0` is returned when averaging is disabled.

## Query Example

| | |
|---|---|
| `SENS2:AVER?` | *This command queries whether averaging is on or off for channel B.* |

## Error Messages

If `[SENSe[1]]|SENSe2:AVERage:STATe` is set to `ON` while `[SENSe[1]]|SENSe2:SPEed` is set to 200, the error -221, “Settings Conflict” occurs.

# [SENSe[1]]|SENSe2:AVERage2 Commands

These commands control video averaging, which is used to improve measurement accuracy, for the P-Series and E-Series E9320 Power Sensor. They combine successive measurements to produce a new composite result.

**NOTE** If the command is used when a sensor other than a P-series or E9320 power sensor is connected, error -241, “Hardware missing” occurs

If the commands in this section are used when an E9320 sensor is connected and set to AVERage mode rather than NORMal mode, the error -221, “Settings Conflict” occurs.

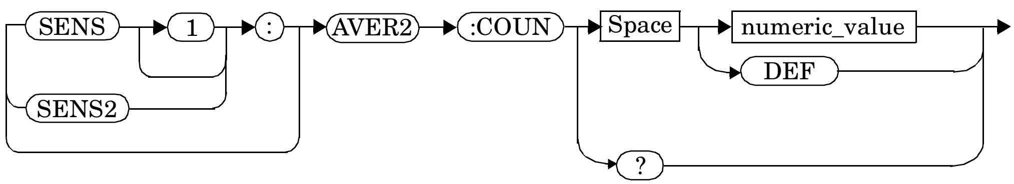
The following commands are detailed in this section:

```
[SENSe[1]]|SENSe2:AVERage2:COUNt <numeric_value>
[SENSe[1]]|SENSe2:AVERage2[:STATe] <boolean>
```

# [SENSe[1]]|SENSe2:AVERage2:COUNt <numeric_value>

This command is used to enter the video filter length for the P-Series and E9320 sensor. Video filtering is applied to the traces. Successive traces are combined to reduce noise without affecting the dynamic characteristic of the signal.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value defining the filter length.<br>• `DEF`: the default value is 4. | 1 to 256[a]<br>`DEF` |

a. This is only implemented in powers of 2 ($2^n$).

### Example

`AVER2:COUN 16` *This command enters a video filter length of 16 for channel A.*

### Reset Condition

On reset, the filter length is set to 4.

### Query

`AVERage2:COUNt?`

The query returns the current setting of the video filter length. The format of the response is `<NR1>`.

### Query Example

| | |
|---|---|
| `AVER2:COUN?` | *This command queries the video filter length for channel A.* |

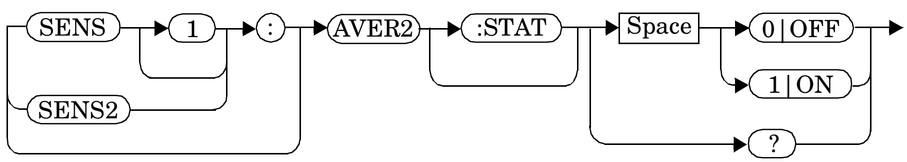
### Error Messages

- If the command is used when a sensor other than a P-series or E9320 power sensor is connected, error -241, "Hardware missing" occurs
- If the command is used when an E9320 sensor is connected and set to `AVERage` mode rather than NORMal mode, the error -221, "Settings Conflict" occurs.

# SENSe[1]]|SENSe2:AVERage2[:STATe] <boolean>

This command is used to enable and disable video averaging for the P-series or E9320 sensor.

**Syntax**




**Example**

| | |
|---|---|
| `AVER2 1` | *This command enables video averaging on channel A.* |

**Reset Condition**

On reset, averaging is enabled.

**Query**

`[SENSe[1]]|SENSe2:AVERage2[:STATe]?`

The query enters a `1` or `0` into the output buffer indicating the status of averaging.

- `1` is returned when averaging is enabled.
- `0` is returned when averaging is disabled.

**Query Example**

| | |
|---|---|
| `SENS2:AVER2?` | *This command queries whether averaging is on or off for channel B.* |

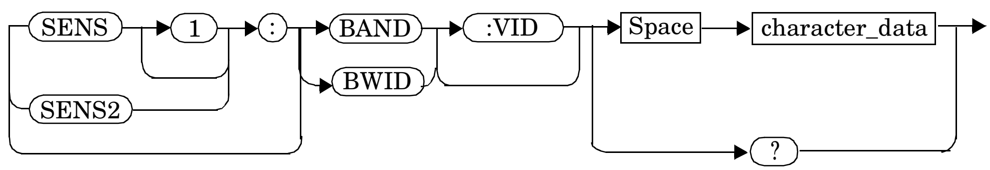
**Error Messages**

- If the command is used when a sensor other than a P-series or E9320 power sensor is connected, error -241, “Hardware missing” occurs
- If the command is used when an E9320 sensor is connected and set to `AVERage` mode rather than `NORMal` mode, the error -221, “Settings Conflict” occurs.

# [SENSe[1]]|SENSe2:BANDwidth|BWIDth:VIDeo <character_data>

This command sets the sensor bandwidth on a P-Series or an E9320 series sensor.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | Defines the sensor bandwidth. | HIGH<br>MEDium<br>LOW<br>OFF |

Values for HIGH, MEDIUM, LOW and OFF are sensor dependant as shown in the following table:

| | Video Bandwidth Settings | | | |
|---|---|---|---|---|
| **Sensor** | **LOW** | **MEDium** | **HIGH** | **OFF** |
| E9321A<br>E9325A | 30 kHz | 100 kHz | 300 kHz | 300 kHz[a] |
| E9322A<br>E9326A | 100 kHZ | 300 kHz | 1.5 MHz | 1.5 MHz[a] |
| E9323A<br>E9327A | 300 kHz | 1.5 MHz | 5 MHz | 5 MHz[a] |
| N1920A<br>N1921A | 5 MHz | 15 MHz | 30 MHz | 30 MHz |

a. At 3.0 dB roll off point.

**Example**

| | |
|---|---|
| `SENSe1:BAND:VID HIGH` | *This command sets sensor bandwidth to high for channel A.* |

**Reset Condition**

On reset, sensor bandwidth is set to `OFF`.

**Query**

[SENSe[1]] | SENSe2:BANDwidth | BWIDth:VIDeo?

The query returns the current sensor bandwidth setting.

**Query Example**

| | |
|---|---|
| `SENS2:BAND:VID?` | *This command queries the current sensor bandwidth setting for channel B.* |

**Error Messages**

- If the command is used when a sensor other than a P-series or E9320 power sensor is connected, error -241, “Hardware missing” occurs
- If the command is used when a P-Series or an E9320 sensor is connected and set to `AVERage` mode rather than `NORMal` mode, the error -221, “Settings Conflict” occurs.

**NOTE**

Selection of video bandwidth to `LOW`, `MED` or `HIGH` implements digital signal processing to ensure a flat bandwidth up to the frequency shown, bandwidths are flat to ±0.1 dB. In the `OFF` state no corrections are applied and the response has a slow roll-off.

# [SENSe[1]]|SENSe2:CORRection Commands

These commands provide for changes to be applied to the measurement result. They are used to enter duty cycle values, calibration factors and other external gains and losses.

The following commands are detailed in this section:

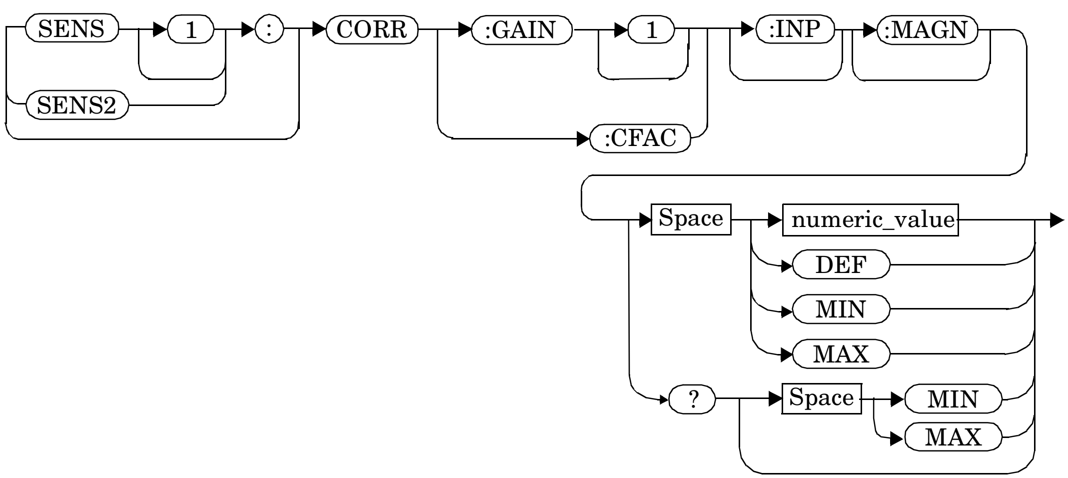
```
[SENSe[1]]|SENSe2:CORRection:CFACtor|GAIN[1][:INPut]
[:MAGNitude] <numeric_value>
[SENSe[1]]|SENSe2:CORRection:CSET[1]|CSET2
[:SELect] <string>
[SENSe[1]]|SENSe2:CORRection:CSET[1]|CSET2:STATe <boolean>
[SENSe[1]]|SENSe2:CORRection:DCYCle|GAIN3[:INPut]
[:MAGNitude] <numeric_value>
[SENSe[1]]|SENSe2:CORRection:DCYCle|GAIN3:STATe <boolean>
[SENSe[1]]|SENSe2:CORRection:FDOFfset|GAIN4[:INPut]
[:MAGNitude]?
[SENSe[1]]|SENSe2:CORRection:LOSS2[:INPut][:MAGNitude]
<numeric_value>
[SENSe[1]]|SENSe2:CORRection:LOSS2:STATe <boolean>
```

# SENSe[1]]|SENSe2:CORRection:CFACtor|GAIN[1][:INPut][:MAGNitude] <numeric_value>

This command is used to enter a gain correction value for the calibration factor. The power meter corrects every measurement by this factor to compensate for the gain.

Either `CFACtor` and `GAIN1` can be used in the command—both have an identical result. Using `GAIN1` complies with the SCPI standard, whereas `CFACtor` does not—this may make your program easier to understand.

## Syntax




## Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| `numeric_value` (for `CFACtor` and `GAIN1`) | A numeric value.<br>• `DEF`: the default value is 100%.<br>• `MIN`: 1%.<br>• `MAX`: 150%. | 1 to 150 PCT[a]<br>`DEF`<br>`MIN`<br>`MAX` |

a. For example, a gain of 60% corresponds to a multiplier of 0.6 and a gain of 150% corresponds to a multiplier of 1.5.

**Example**

| | |
|---|---|
| SENS2:CORR:GAIN1 | *This command sets a gain correction of 100% for channel B.* |

**Reset Condition**

On reset, `CFACtor|GAIN1` is set to 100%.

**Query**

`[SENSe[1]]|SENSe2:CORRection:CFACtor|GAIN[1][:INPut][:MAGNitude]? [MIN|MAX]`

The query returns the current gain correction setting or the values associated with `MIN` and `MAX`.

**Query Example**

| | |
|---|---|
| CORR:GAIN1? | *This command queries the current calibration factor setting for channel A.* |

**Error Messages**

The `SENSe[1]]|SENSe2:CORRection:CFACtor|GAIN1` command can be used for the 8480 series power sensor when no sensor calibration table has been set up. If a sensor calibration table is selected the error -221, “Settings Conflict” occurs.

# [SENSe[1]]|SENSe2:CORRection:CSET[1]|CSET2 Commands

These commands are used to select the active sensor calibration table (using CSET1) and the active frequency dependent offset table (using CSET2).

**NOTE** If any of the CSET1 commands are used when a P-series or E-series power sensor is connected, the error -241, “Hardware missing” occurs.

The following commands are detailed in this section:

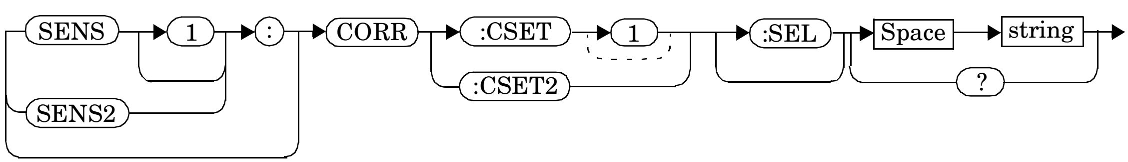
```
[SENSe[1]]|SENSe2:CORRection:CSET[1]|CSET2[:SELect] <string>
[SENSe[1]]|SENSe2:CORRection:CSET[1]|CSET2:STATe <boolean>
```

# [SENSe[1]]|SENSe2:CORRection:CSET[1]|CSET2[:SELect] <string>

This command enters the name of the sensor calibration table or frequency dependent offset table which is to be used. The `CSET1` command selects the sensor calibration table and the `CSET2` command selects the frequency dependent offset table. The calibration factor is interpolated from the table using the setting for `[SENSe[1]]|SENSe2:FREQuency`.

**NOTE** If `[SENSe[1]]|SENSe2:CORRection:CSET[1]|CSET2:STATe` is set to `OFF`, the selected sensor calibration table or frequency offset table is not being used.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| `string` | String data representing a sensor calibration table, or frequency dependent offset table name. | Any existing table name (Existing table names can be listed using `MEMory:CATalog:TABle?`). |

### Example

| | |
|---|---|
| `CORR:CSET1 'PW1'` | *This command enters the name of the sensor calibration table which is to be used on channel A.* |

**Reset Condition**

On reset the selected table is not affected.

**Query**

`[SENSe[1]]|SENSe2:CORRection:CSET[1]|CSET2:[SELect]?`

The name of the selected table is returned as a quoted string. If no table is selected an empty string is returned.

**Query Example**

| | |
|---|---|
| `CORR:CSET1?` | *This command queries the sensor calibration table currently used for channel A.* |

**Error Messages**

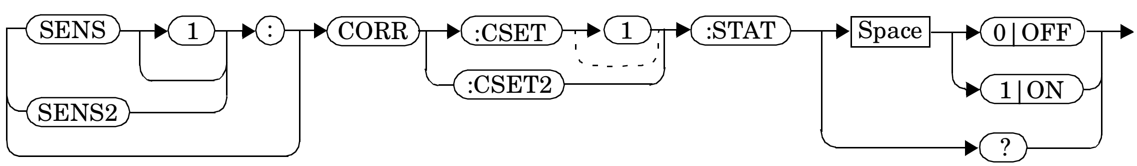
- If `<string>` is not valid, error -224, “Illegal parameter value” occurs.
- If a table called `<string>` does not exist, error -256, “File name not found” occurs.
- When a sensor calibration table is selected, the power meter verifies that the number of calibration points defined is one more than the number of frequency points defined. When a frequency dependent offset table is selected, the power meter verifies that the number of offset points defined is equal to the number of frequency points defined. If this is not the case, error -226, “Lists not the same length” occurs.
- If the `CSET1` command is used when a P-series or an E-series power sensor is connected the error -241, “Hardware missing” occurs.

# [SENSe[1]]|SENSe2:CORRection:CSET[1]|CSET2:STATe <boolean>

This command is to enable and disable the use of the currently active sensor calibration table (`CSET1`) or frequency dependent offset table (`CSET2`). When a table has been selected and enabled, the calibration factors/offsets stored in it can be used by specifying the required frequency using the `[SENSe[1]]|SENSe2:FREQuency` command.

When the `CSET1` command is set to `ON`, the reference calibration factor is taken from the sensor calibration table and is used during calibration.

### Syntax




### Example

| | |
|---|---|
| `CORR:CSET1:STAT 1` | *This command enables the use of the currently active sensor calibration table for channel A.* |

### Reset Condition

On reset, the sensor calibration table and frequency dependent offset table are not affected.

### Query

`[SENSe[1]]|SENSe2:CORRection:CSET[1]|CSET2:STATe?`

The query returns a `1` or `0` into the output buffer indicating whether a table is enabled or disabled.

- `1` is returned when the table is enabled.
- `0` is returned when the table is disabled.

**Query Example**

| | |
|---|---|
| `SENS2:CORR:CSET1:STAT?` | *This command queries whether there is currently an active sensor calibration table for channel B.* |

**Error Messages**

If you attempt to set this command to `ON` and no table has been selected using `[SENSe[1]]|SENSe2:CORRection:CSET[1]|CSET2:[SELect]` then error -221, "Settings conflict" occurs and `[SENSe[1]]|SENSe2:CORRection:CSET[1]|CSET2:STATe` remains `OFF`.

# [SENSe[1]]|SENSe2:CORRection:DCYCle|GAIN3 Commands

These commands control the pulse power measurement feature of the power meter.

The following commands are detailed in this section:

```
[SENSe[1]]|SENSe2:CORRection:DCYCle|GAIN3[:INPut]
[:MAGNitude] <numeric_value>
[SENSe[1]]|SENSe2:CORRection:DCYCle|GAIN3:STATe <boolean>
```

**NOTE** You can use either `DCYCLe` or `GAIN3` in these commands, both do the same. Using `GAIN3` complies with the SCPI standard whereas `DCYCle` does not, but may make your program more understandable.

# [SENSe[1]]|SENSe2:CORRection:DCYCle|GAIN3[:INPut] [:MAGNitude] <numeric_value>

This command is used to set the duty cycle for the pulse power measurement feature of the power meter. Pulse power measurements average out any deviations in the pulse, such as, overshoot or ringing. The result returned for a pulse power measurement is a mathematical representation of the pulse power rather than an actual measurement. The power meter measures the average power in the pulsed input signal and then divides the result by the duty cycle value to obtain a pulse power reading.

Entering a value using this command automatically turns the `[SENSe[1]]|SENSe2:CORRection:DCYCle|GAIN3:STATe` command to `ON`.

**NOTE** Pulse measurements are not recommended using E-Series power sensors at power levels above -20 dBm.

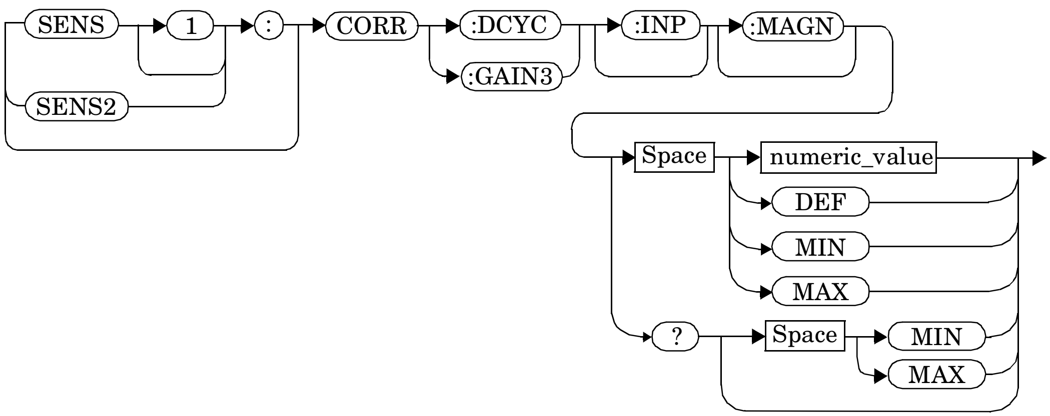
Pulse power averages out any deviations in the pulse such as overshoot or ringing. Hence, it is called pulse power and not peak power or peak pulse power.

In order to ensure accurate pulse power readings, the input signal must be pulsed with a rectangular pulse. Other pulse shapes (such as triangle, chirp or Gaussian) cause incorrect results.

The pulse power on/off ratio must be much greater than the duty cycle ratio.

## [SENSe[1]]|SENSe2:CORRection:DCYCle|GAIN3[:INPut] [:MAGNitude] <numeric_value>

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value for the duty cycle.<br>• `DEF`: the default value is 1%.<br>• `MIN`: 0.001%.<br>• `MAX`: 99.999%.<br>The units are PCT, and are optional. | 0.001 to 99.999 PCT<br>`DEF`<br>`MIN`<br>`MAX` |

### Example

| | |
|---|---|
| `CORR:DCYC 90PCT` | *This command sets a duty cycle of 90% for channel A.* |

### Reset Condition

On reset, the duty cycle is set to 1% (`DEF`).

### Query

```
[SENSe[1]]|SENSe2:CORRection:DCYCle|GAIN3[:INPut]
[:MAGNitude]? [MIN|MAX]
```

The query returns the current setting of the duty cycle or the values associated with `MIN` and `MAX`.

### Query Example

| | |
|---|---|
| `CORR:GAIN3?` | *This command queries the current setting of the duty cycle for channel A.* |

### Error Messages

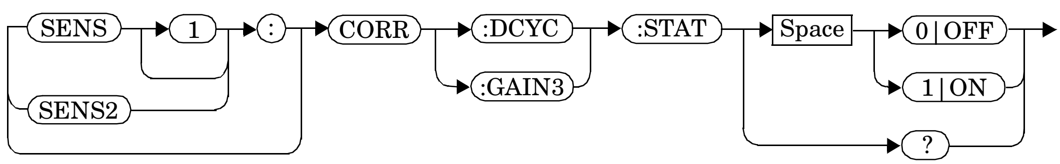
- If a duty cycle value is entered using `[SENSe[1]]|SENSe2:CORRection:DCYCle|GAIN3` while `[SENSe[1]]|SENSe2:SPEed` is set to 200, the error -221, “Settings Conflict” occurs. However, the duty cycle value is set but the `[SENSe[1]]|SENSe2:CORRection:DCYCle|GAIN3`:STATe command is not automatically set `ON`.
- If this command is used when an P-Series or E-series power sensor is connected, the error -310, “System error;Dty Cyc may impair accuracy with ECP sensor” occurs. If you are using a dual channel power meter the error message specifies the channel.

# [SENSe[1]]|SENSe2:CORRection:DCYCle|GAIN3:STATe <boolean>

This command is used to enable and disable the pulse power measurement feature.

The `[SENSe[1]]|SENSe2:CORRection:DCYCle|GAIN3` command should be used to enter the duty cycle of the signal you want to measure.

### Syntax




### Example

`CORR:DCYC:STAT 1` *This command enables the pulse measurement feature on channel A.*

### Reset Condition

On reset, the pulse power measurement feature is disabled.

### Query

`[SENSe[1]]|SENSe2:CORRection:DCYCle|GAIN3:STATe?`

The query enters a `1` or `0` into the output buffer indicating the status of the pulse power measurement feature.

- `1` is returned when the pulse power measurement feature is enabled.
- `0` is returned when the pulse power measurement feature is disabled.

### Query Example

`CORR:GAIN3:STAT?` *This command queries whether the pulse measurement feature is on or off.*

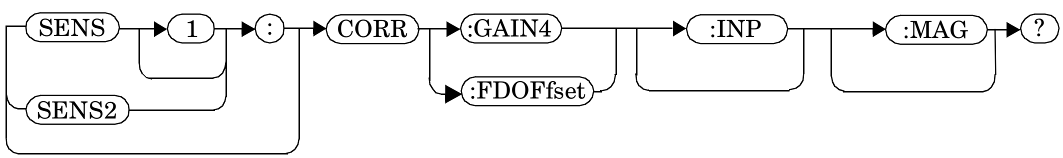
**Error Messages**

- If `[SENSe[1]]|SENSe2:CORRection:DCYCle:STATus` is set to `ON` while `[SENSe[1]]|SENSe2:SPEed` is set to 200, the error -221, "Settings Conflict" occurs.
- If this command is used when an P-Series or E-series power sensor is connected, the error -310, "System error;Dty Cyc may impair accuracy with ECP sensor" occurs. If you are using a dual channel power meter the error message specifies the channel.

# [SENSe[1]]|SENSe2:CORRection:FDOFfset|GAIN4[:INPut][:MAGNitude]?

This command is used to return the frequency dependent offset currently being applied.

### Syntax




### Example

| | |
|---|---|
| `CORR:GAIN4?` | *This command queries the current frequency dependent offset being applied to channel A.* |

### Reset Condition

On reset, the frequency dependent offset is not affected.

# [SENSe[1]]|SENSe2:CORRection:GAIN2 Commands

These commands provide a simple correction to a measurement for an external gain/loss.

The following commands are detailed in this section:

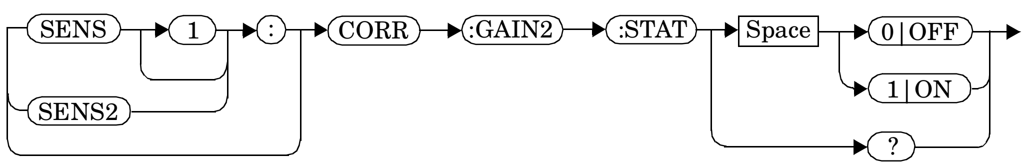
`[SENSe[1]]|SENSe2:CORRection:GAIN2:STATe <boolean>`

`[SENSe[1]]|SENSe2:CORRection:GAIN2[:INPut][:MAGNitude] <numeric_value>`

# [SENSe[1]]|SENSe2:CORRection:GAIN2:STATe <boolean>

This command is used to enable/disable a channel offset for the power meter setup. The `[SENSe[1]]|SENSe2:CORRection:GAIN2[:INPut][:MAGNitude]` command is used to enter the loss/gain value.

### Syntax




### Example

| | |
|---|---|
| `CORR:GAIN2:STAT ON` | *This command enables a channel offset on channel A.* |

**Reset Condition** On reset, channel offsets are disabled.

**Query** `[SENSe[1]]|SENSe2:CORRection:GAIN2:STATe?`

The query enters `1` or `0` into the output buffer indicating the status of the channel offsets.

- `1` is returned if a channel offset is enabled.
- `0` is returned if a channel offset is disabled.

### Query Example

| | |
|---|---|
| `CORR:GAIN2:STAT?` | *This command queries whether or not there is a channel offset applied to channel A.* |

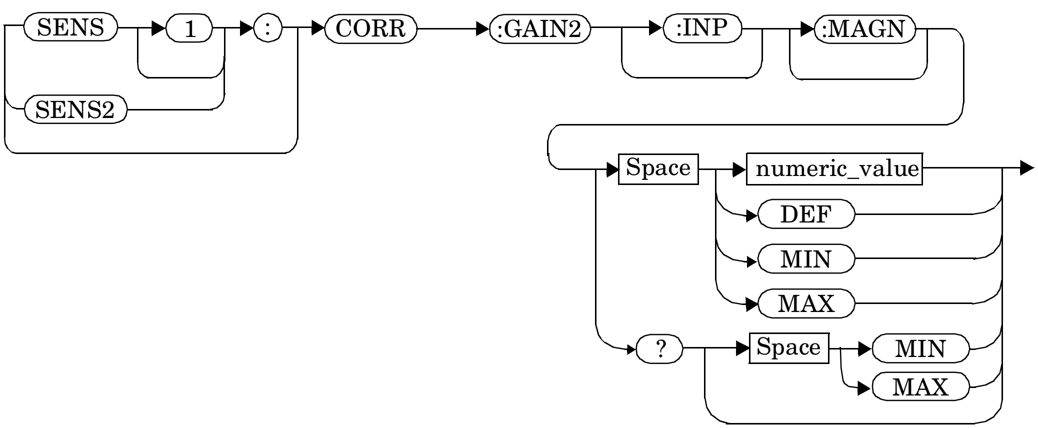
**Error Messages** If `[SENSe[1]]|SENSe2:CORRection:GAIN2:STATe` is set to `ON` while `[SENSe[1]]|SENSe2:SPEed` is set to 200, the error -221, “Settings Conflict” occurs.

# [SENSe[1]]|SENSe2:CORRection:GAIN2[:INPut] [:MAGNitude] <numeric_value>

This command is used to enter a channel offset value for the power meter setup, for example cable loss. The power meter then corrects every measurement by this factor to compensate for the gain/loss.

Entering a value for `GAIN2` using this command automatically turns the `[SENSe[1]]|SENSe2:CORRection:GAIN2:STATe` command to `ON`.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value:<br>• `DEF`: the default is 0.00 dB.<br>• `MIN`: -100 dB.<br>• `MAX`: +100 dB. | -100 to +100 dB<br>`DEF`<br>`MIN`<br>`MAX` |

**Example**

| | |
|---|---|
| `CORR:GAIN2 50` | *This command sets a channel offset of 50 dB for channel A.* |

**Reset Condition** On reset, `GAIN2` is set to 0.00 dB.

**Query** `[SENSe[1]]|SENSe2:CORRection:GAIN2[:INPut][:MAGNitude]? [MIN|MAX]`

The query returns the current setting of the channel offset or the values associated with `MIN` and `MAX`.

**Query Example**

| | |
|---|---|
| `CORR:GAIN2?` | *This command queries the current setting of the channel offset on channel A.* |

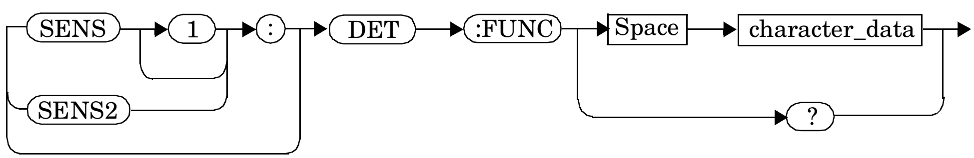
**Error Messages**

- If a loss/gain correction value is entered using `[SENSe[1]]|SENSe2:CORRection:GAIN2[:INPut][:MAGNitude]` while `[SENSe[1]]|SENSe2:SPEed` is set to 200, the error -221, "Settings Conflict" occurs. However, the correction value is set but the `[SENSe[1]]|SENSe2:CORRection:GAIN2:STATe` command is not automatically set `ON`.
- The `SENSe[1]]|SENSe2:CORRection:GAIN2[:INPut][:MAGNitude]` command can be used for the 8480 series power sensor when no sensor calibration table has been set up.

# [SENSe[1]]|SENSe2:DETector:FUNCtion <character_data>

This command sets the measurement mode for the E9320 sensor only.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | Defines the measurement mode:<br>• AVERage: sets the E9320 sensor to average only mode.<br>• NORMal: sets the E9320 sensor to normal mode. | AVERage[a]<br>NORMal[b] |

a. 1. When measurement mode is set to average:

- If TRIG:SOUR is set to INT1, INT2 or EXT, it is set automatically to IMM.
- INIT:CONT is set automatically to ON.
- SENS:AVER2:STAT is set automatically to OFF.
- CALC:FEED is set automatically to "POW:AVG" for all CALC blocks using the specified channel in their CALC:MATH:EXPR.

b. 2. When measurement mode is set to NORMal:

- SENS:CORR:DCYC:STAT is set automatically to OFF.

### Example

| | |
|---|---|
| `SENS1:DET:FUNC NORM` | *This command sets the sensor to normal mode for channel A.* |

**Reset Condition**

On reset, the mode is set to `NORMal`.

**Query**

[SENSe[1]] | SENSe2:DETector:FUNCtion?

The query returns the current sensor mode setting.

**Query Example**

| | |
|---|---|
| `SENS:DET:FUNC?` | *This command queries the current sensor mode setting for channel A.* |

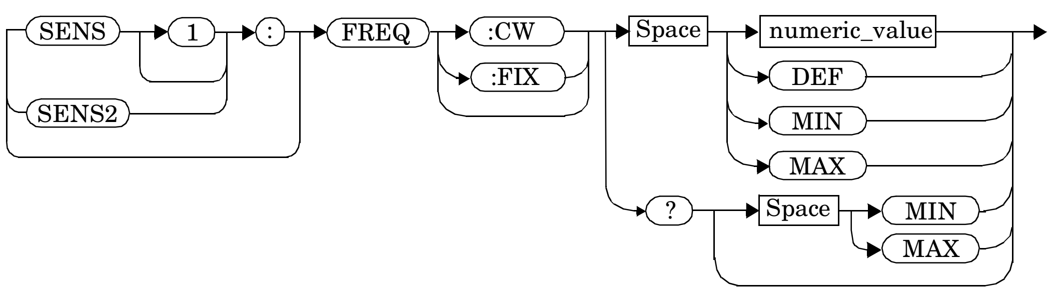
**Error Messages**

- If the command is used when a non E9320 sensor is connected, the error -241, "Hardware missing" occurs.
- If the command is used when an E9320 sensor is connected and set to `AVERage` mode rather than `NORMal` mode, the error -221, "Settings Conflict" occurs.

# [SENSe[1]]|SENSe2:FREQuency[:CW|:FIXed] <numeric_value>

This command is used to enter a frequency. If the frequency does not correspond directly to a frequency in the sensor calibration table, the power meter calculates the calibration factor using linear interpolation. For 8480 series power sensor the power meter uses linear interpolation to calculate the calibration factor for the frequency entered if `[SENSe[1]]|SENSe2:CORRection:CSET:STATe` is ON. For P-series and E-series power sensor, the appropriate corrections are applied for the frequency selected, dependant on the calibration data stored in the sensor's EEPROM.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value for the frequency:<br>• `DEF`: the default value is 50 MHz.<br>• `MIN`: 1 kHz.<br>• `MAX`: 1000.0 GHz.<br>The default units are Hz. | 1 kHz to 1000.0 GHz[a]<br>`DEF`<br>`MIN`<br>`MAX` |

a. The following measurement units can be used:

- Hz
- kHz ($10^3$)
- MHz ($10^6$)
- GHz ($10^9$)

**Example**

| | |
|---|---|
| `FREQ 500kHz` | *This command enters a channel A frequency of 500 kHz.* |

**Reset Condition** On reset, the frequency is set to 50 MHz (`DEF`).

**Query** `[SENSe[1]]|SENSe2:FREQuency[:CW|:FIXed]? [MIN|MAX]`

The query returns the current frequency setting or the values associated with `MIN` and `MAX`. The units in which the results are returned are Hz.

**Query Example**

| | |
|---|---|
| `SENS2:FREQ?` | *This command queries the channel B frequency setting.* |

# [SENSe[1]]|SENSe2:MRATe <character_data>

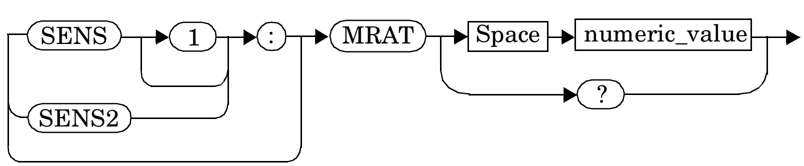
This command sets the measurement speed on the selected channel.

When a channel is set to FAST, the following couplings occur:

| Command | Status |
|---|---|
| [SENSe[1]]\|SENSe2:AVERage:STATe | OFF[a] |
| [SENSe[1]]\|SENSe2:CORRection:DCYCle:STATe | OFF[a] |
| [SENSe[1]]\|SENSe2:CORRection:GAIN2:STATe | OFF[a] |
| CALCulate[1\|2\|3\|4]:GAIN:STATe | OFF[b] |
| CALCulate[1\|2\|3\|4]:RELative:STATe | OFF[b] |
| CALCulate1\|3:MATH:EXPRession | "(SENSe1)" |
| CALCulate2\|4:MATH:EXPRession | "(SENSe2)"[c] |

a. This change only occurs on the channel specified in the SENSe:MRATe command. When the specified channel is changed from FAST to NORMal or DOUBle, the settings that were in place when FAST was entered are restored.
b. This change occurs when either channel is set to FAST. When both channels are changed from FAST to NORMal or DOUBle, the settings that were in place when FAST was entered are restored.
c. Applicable to the N1912A dual channel power meter only.

### Syntax



### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | A numeric value for the measurement speed:<br>• `NORMal`: 20 readings/second.<br>• `DOUBle`: 40 readings/second.<br>• `FAST`: up to 1000 readings/second.<br>The default is `NORMal`. | `NORMal`[a]<br>`DOUBle`[a]<br>`FAST` |

a. When a channel is set to `NORMal` or `DOUBle`, `TRIG:COUNt` is set automatically to 1.

### Example

| | |
|---|---|
| `MRAT DOUBle` | *This command sets the channel A speed to 40 readings/second.* |

### Reset Condition

On reset, the speed is set to `NORMal`.

### Query

`[SENSe[1]]|SENSe2:MRAT?`

The query returns the current speed setting, either `NORMal`, `DOUBle` or `FAST`.

### Query Example

| | |
|---|---|
| `MRAT?` | *This command queries the current speed setting for channel A.* |

### Error Messages

- If `<character_data>` is not set to `NORMal`, `DOUBle` or `FAST`, error -224 “Illegal parameter value” occurs.
- If a P-Series or an E-series power sensor is not connected and `<character_data>` is set to `FAST`, error -241 “Hardware missing” occurs.

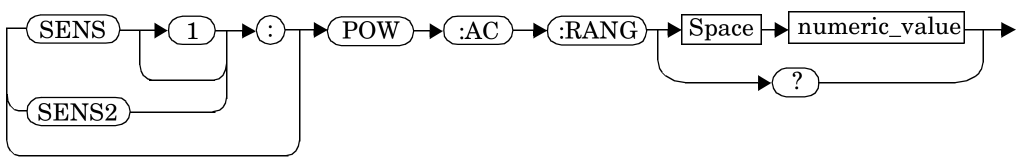
# [SENSe[1]]|SENSe2:POWer:AC:RANGe <numeric_value>

This command is only valid when used with an E-series power sensor. Its purpose is to select one of two power ranges.

- If `0` is selected, the power sensor's lower range is selected.
- If `1` is selected, the power sensor's upper range is selected.

Setting a range with this command automatically switches `[SENSe[1]]|SENSe2:POWer:AC:RANGe:AUTO` to `OFF`.

**Syntax**




**Example**

| | |
|---|---|
| `POW:AC:RANG 0` | *This command sets the power sensor to it's lower range.* |

**Reset Condition** On reset, the upper range is selected.

**Query** `[SENSe[1]]|SENSe2:POWer:AC:RANGe?`

The query enters a `1` or `0` into the output buffer indicating the status of the power sensor's range.

- `1` is returned when the upper range is selected.
- `0` is returned when the lower range is selected.

**Query Example**

| | |
|---|---|
| `POW:AC:RANG?` | *This command queries the current setting of the power sensor range.* |

**Error Messages** This command is used with the E-series power sensor. If one is not connected the error -241, "Hardware missing" occurs.

# [SENSe[1]]|SENSe2:POWer:AC:RANGe:AUTO <boolean>

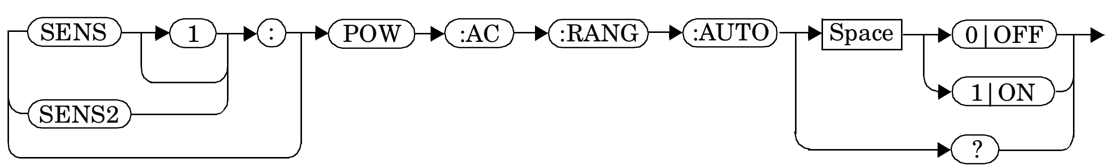
This command is only valid when used with an E-series power sensor. Its purpose is to enable and disable autoranging. When autoranging is ON, the power meter selects the best measuring range for the measurement. When autoranging is set to `OFF`, the power meter remains in the currently set range.

The `[SENSe[1]]|SENSe2:POWer:AC:RANGe` command disables autoranging.

If `INITiate:CONTinuous` is set to `ON` and `TRIGger:SOURce` is set to `IMMediate`, the range tracks the input power if `[SENSe[1]]|SENSe2:POWer:AC:RANGe:AUTO` is `ON`.

If the power meter is not making measurements then autoranging only occurs when the power meter is triggered.

### Syntax




### Example

| | |
|---|---|
| `POW:AC:RANG:AUTO 0` | *This command disables autoranging.* |

### Reset Condition

On reset, autoranging is enabled.

### Query

`[SENSe[1]]|SENSe2:POWer:AC:RANGe:AUTO?`

The query enters a `1` or `0` into the output buffer indicating the status of autoranging.

- `1` is returned when autoranging is enabled.
- `0` is returned when autoranging is disabled.

### Query Example

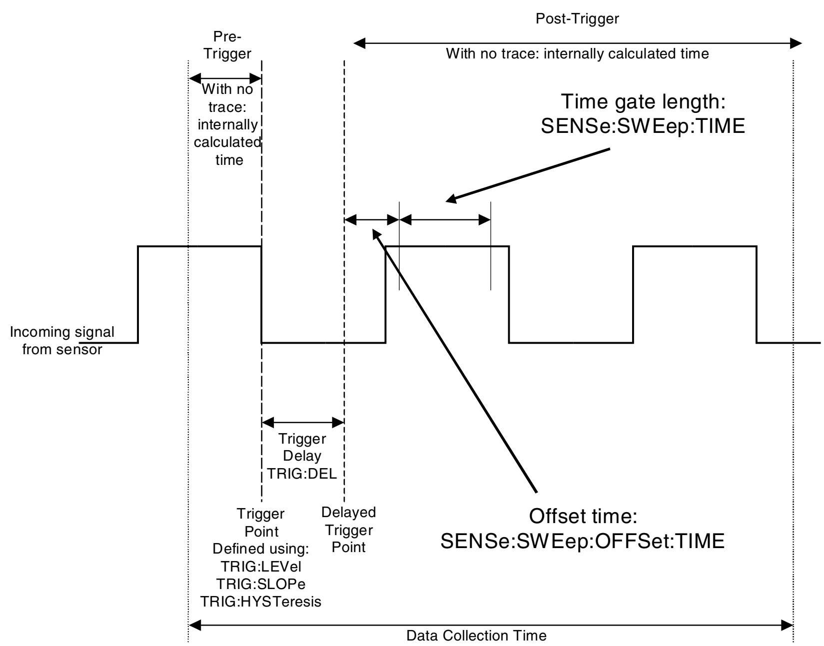
| | |
|---|---|
| POW:AC:RANG:AUTO? | *This command queries whether auto ranging is on or off.* |

### Error Messages

If this command is set to OFF when there is not an E-series power sensor connected, the error, -241, “Hardware missing” occurs.

# SENSe[1]]|SENSe2:SWEep[1]|2|3|4 Commands

These commands set offset time and time gate length as illustrated in the following diagram:




Offset time and time gate length values can be set for up to four measurement gates per channel. Measurement gate number is defined by the numeric value following the SWEep component of the command.

**NOTE** These commands can only be used with P-Series and E9320 sensors. The E9320 sensor must be set to `NORMal` mode.

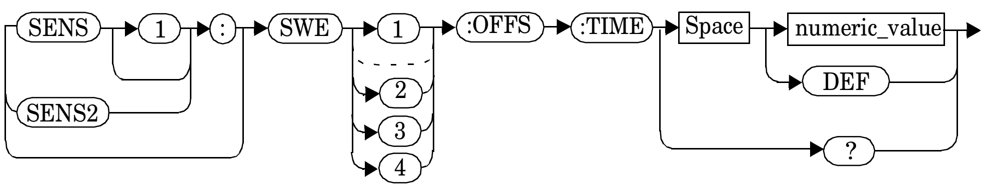
The following commands are detailed in this section:

```
[SENSe[1]]|SENSe2:SWEep[1]|2|3|4:OFFSet:TIME <numeric_value>
[SENSe[1]]|SENSe2:SWEep[1]|2|3|4:TIME <numeric_value>
```

# [SENSe[1]]|SENSe2:SWEep[1]|2|3|4:OFFSet:TIME <numeric_value>

This command sets the delay between the delayed trigger point and the start of the time-gated period (the offset time) for a P-series sensor or a E9320 sensors set to `NORMal` mode. To set an E9320 sensor to `NORMal` mode, refer to the command `[SENSe[1]]|SENSe2:DETector:FUNCtion <character_data>` on page 297.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | The delay between the trigger point and the start of the time-gated period.<br>• `DEF`: the default value is 0 seconds.<br>Units are resolved to 1 ns. | -1 to 1 seconds<br>`DEF` |

### Example

| | |
|---|---|
| `SENS2:SWE3:OFFS:TIME 0.001` | *This command sets the delay to 0.001 seconds.* |

### Reset Condition

On reset, the value is set to 0 seconds.

**Query**

`SENSe[1]]|SENSe2:SWEep[1]|2|3|4:OFFSet:TIME?`

The query returns the current delay between the trigger point and the start of the time-gated period.

**Query Example**

| | |
|---|---|
| `SENS2:SWE2:OFFS:TIME?` | *The query returns the current delay between the trigger point and the start of the time-gated period for channel B and gate 2.* |

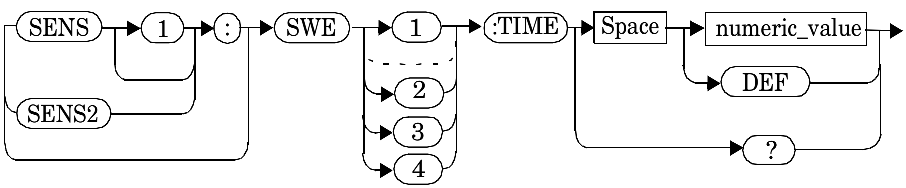
**Error Messages**

- If the command is used when a sensor other than a P-series or E9320 power sensor is connected, error -241, “Hardware missing” occurs
- If the command is used when an E9320 sensor is connected and set to `AVERage` mode rather than `NORMal` mode, the error -221, “Settings Conflict” occurs.

# [SENSe[1]]|SENSe2:SWEep[1]|2|3|4:TIME <numeric_value>

This command sets the length of the time-gated period (time-gate length) for time-gated measurements for the P-Series and E9320 sensors which are set to NORMal mode. To set an E9320 sensor to NORMal mode, refer to the command [SENSe[1]]|SENSe2:DETector:FUNCtion <character_data> on page 297.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | The length of the time gated period in seconds.<br>• DEF: the default value is 100 us<br>Units are resolved to 1 ns. | 0 to 1 second<br>DEF |

### Example

SENS2:SWE3:TIME 0.001 *This command sets the length to 0.001 seconds.*

### Reset Condition

On reset, gate 1 is set to 100us and other gates to 0s.

### Query

SENSe[1]]|SENSe2:SWEep[1]|2|3|4:TIME?

The query returns the current length of the time-gated period.

### Query Example

| | |
|---|---|
| `SENS2:SWE2:TIME?` | *This command queries the length of the time-gated period for channel B and gate 2.* |

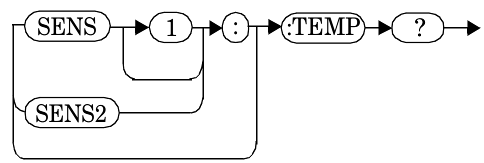
### Error Messages

- If the command is used when a sensor other than a P-series or E9320 power sensor is connected, error -241, “Hardware missing” occurs
- If the command is used when an E9320 sensor is connected and set to `AVERage` mode rather than `NORMal` mode, the error -221, “Settings Conflict” occurs.

# [SENSe[1]]|SENSe2:TEMPerature?

This this command to returns the P-series power sensor's temperature in degrees Celsius.

**Syntax**




**Reset Condition** On reset, this parameter is not affected.

**Parameters**

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value defining sensor's temperature in degrees Celsius. | -50 to 100 |

**Example**

| | |
|---|---|
| `SENS2:TEMP?` | *This command returns the current sensor temperature found on channel B.* |

**Error Messages**

- If a P-series sensor is not connected, error -241, “Hardware missing” occurs.

# SENSe[1]]|SENSe2:TRACe Commands

These commands are used to set:

- The upper and lower limits for the trace display.
- The delay between the delayed trigger point and the start of the trace.
- The duration of the trace.
- The trace units.

**NOTE** These commands can only be used with P-Series and E9320 sensors. The E9320 sensor must be set to `NORMal` mode.

The following commands are detailed in this section:

`[SENSe[1]]|SENSe2:TRACe:LIMit:LOWer <numeric_value>`

`[SENSe[1]]|SENSe2:TRACe:LIMit:UPPer <numeric_value>`

`[SENSe[1]]|SENSe2:TRACe:OFFSet:TIME <numeric_value>`

`[SENSe[1]]|SENSe2:TRACe:TIME <numeric_value>`

`[SENSe[1]]|SENSe2:TRACe:UNIT <character_data>`

# SENSe[1]|2:TRACe:LIMit:LOWer <numeric_value>

This command sets the lower scale limit of the trace for the specified channel.

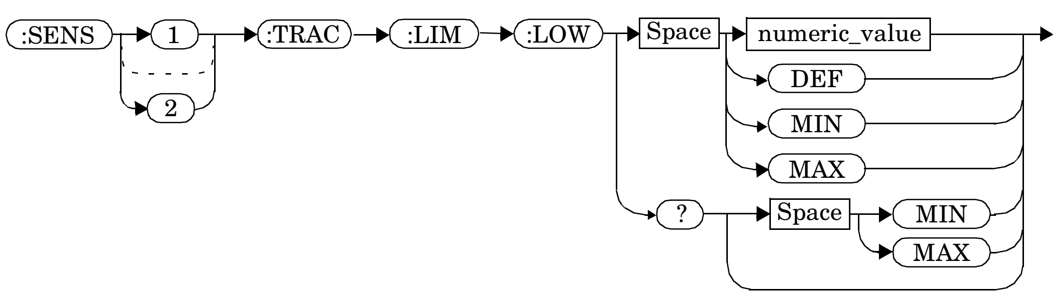
The units used are dependent on the current setting of SENS:TRAC:UNIT as shown in Table 9-1.

**Table 9-1** **Measurement Units**

| Units: SENS:TRAC:UNIT | Units: SENS:TRAC:LIM:LOW |
|---|---|
| dBm | dBm |
| W | W |

**NOTE** The trace lower scale limit is maintained at a lower power than the upper scale limit which is adjusted to be slightly greater than the lower scale limit if necessary. Refer to SENSe[1]|2:TRACe:LIMit:UPPer <numeric_value> for further information on setting the trace upper scale limit.

### Syntax



**Parameters**

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value for the trace lower scale limit.<br>• DEF: the default is 20 dBm<br>• MIN: -150 dBm<br>• MAX: 230 dBm | -150 to 230 dBm<br>DEF<br>MIN<br>MAX |

**Example**

| | |
|---|---|
| `SENS2:TRAC:LIM:LOW 10` | *This command sets the trace lower scale limit to 10 dBm for channel B.* |

**Reset Condition** On reset, the value is set to -50 dBm.

**Query** `SENSe[1]|2:TRACe:LIMit:LOWer [MIN|MAX]`

The query returns the current setting of the trace lower scale limit or the value associated with MIN or MAX. The format of the response is <NR3>.

**Query Example**

| | |
|---|---|
| `SENSe:TRAC:LIM:LOW?` | *This command queries the trace lower scale limit of channel A.* |

# SENSe[1]|2:TRACe:LIMit:UPPer <numeric_value>

This command sets the upper scale limit of the trace for the specified channel.

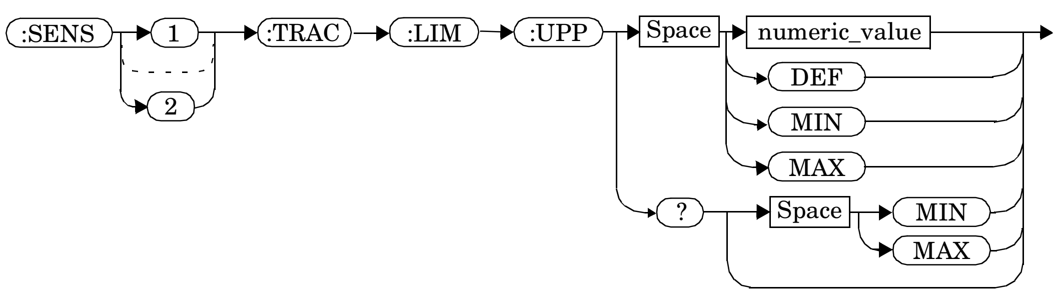
The units used are dependent on the current setting of `SENS:TRAC:UNIT` as shown in Table 9-2.

**Table 9-2** **Measurement Units**

| Units: **SENS:TRAC:UNIT** | Units: **SENS:TRAC:LIM:UPP** |
|---|---|
| dBm | dBm |
| W | W |

**NOTE** The trace lower scale limit is maintained at a lower power than the upper scale limit which is adjusted to be slightly greater than the lower scale limit if necessary. Refer to `SENSe[1]|2:TRACe:LIMit:LOWer <numeric_value>` for further information on setting the trace lower scale limit.

### Syntax



### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value for the trace lower scale limit.<br>• `DEF`: the default is 20 dBm<br>• `MIN`: -150 dBm<br>• `MAX`: 230 dBm | -150 to 230 dBm<br>`DEF`<br>`MIN`<br>`MAX` |

### Example

`SENS:TRAC:LIM:UPP 100` — *This command sets the trace upper scale limit to 100 dBm for channel A.*

**Reset Condition** On reset, the value is set to `DEF`.

**Query** `SENSe[1]|2:TRACe:LIMit:LOWer [MIN|MAX]`

The query returns the current setting of the trace upper scale limit or the value associated with `MIN` or `MAX`. The format of the response is <`NR3`>.

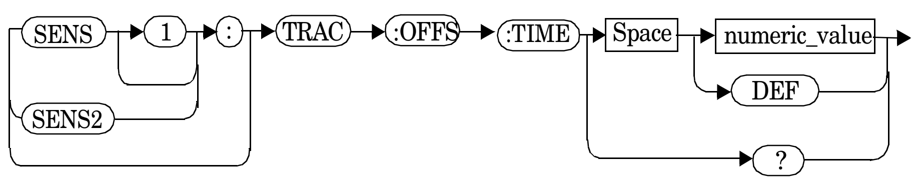
### Query Example

`SENS:TRAC:LIM:UPP?` — *This command queries the trace upper scale limit of channel A.*

# [SENSe[1]]|SENSe2:TRACe:OFFSet:TIME <numeric_value>

This command sets the delay between the delayed trigger point and the start of the trace for P-series or E9320 sensors are set to `NORMal` mode. To set an E9320 sensor to `NORMal` mode, refer to the command `[SENSe[1]]|SENSe2:DETector:FUNCtion <character_data>` on page 297.

### syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | The length of the delay in seconds.<br>• `DEF`: the default value is 0 seconds.<br>Units are resolved to 1 ns. | -1 to 1 seconds<br>`DEF` |

### Example

| | |
|---|---|
| `SENS:TRAC:OFFS:TIME TIME 0.05` | *This command sets the delay to 0.05 seconds.* |

**Reset Condition**

On reset, the delay is set to 0 seconds.

**Query**

`SENSe[1]]|SENSe2:TRACe:OFFSet:TIME?`

The query returns the current delay between the delayed trigger point and the start of the trace.

**Query Example**

| | |
|---|---|
| `SENS:TRAC:OFFS:TIME?` | *This command queries the current delay between the delayed trigger point and the start of the trace for channel A.* |

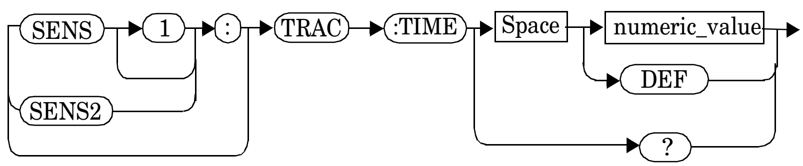
**Error Messages**

- If the command is used when a sensor other than a P-series or E9320 power sensor is connected, error -241, "Hardware missing" occurs
- If the command is used when an E9320 sensor is connected and set to `AVERage` mode rather than `NORMal` mode, the error -221, "Settings Conflict" occurs.

# [SENSe[1]]|SENSe2:TRACe:TIME <numeric_value>

This command sets the duration of the trace for a P-series sensor and a E9320 sensors set to `NORMal` mode. To set the E9320 sensor to `NORMal` mode, refer to the command `SENSe[1]]|SENSe2:DETector:FUNCtion <character_data>`
on page 297.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | The duration of the trace in seconds.<br>• `DEF`: the default value is 100 us.<br>Units are resolved to 1 ns. | 10 us to 1 second<br>`DEF` |

### Example

| | |
|---|---|
| `SENS2:TRAC:TIME 0.5` | *This command sets the duration of the trace to 0.5 seconds for channel B.* |

**Reset Condition** On reset, the duration is set to 100 us.

**Query** `SENSe[1]]|SENSe2:TRACe:TIME?`

The query returns the current duration of the trace.

**Query Example**

| | |
|---|---|
| `SENS2:TRAC:TIME?` | *This command queries the current duration of the trace.* |

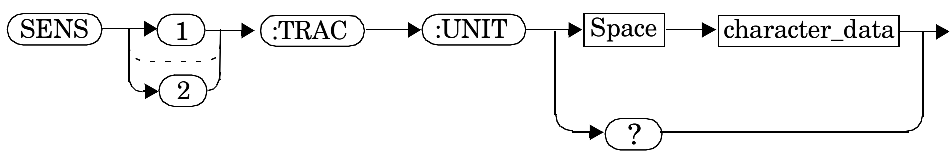
**Error Messages**

- If the command is used when a sensor other than a P-series or E9320 power sensor is connected, error -241, “Hardware missing” occurs
- If the command is used when an E9320 sensor is connected and set to `AVERage` mode rather than `NORMal` mode, the error -221, “Settings Conflict” occurs.

# [SENSe[1]]|SENSe2:TRACe:UNIT <character_data>

This command sets the units for the trace for the specified channel.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | • DBM: dBm.<br>• W: Watts. | DBM<br>W |

### Example

| | |
|---|---|
| SENS2:TRAC:UNIT W | *This command sets the trace units for channel B to Watts.* |

**Reset Condition** On reset the units are set to dBm.

**Query** [SENSe[1]]|SENSe2:TRACe:UNIT?

The query command returns the current value of character_data.

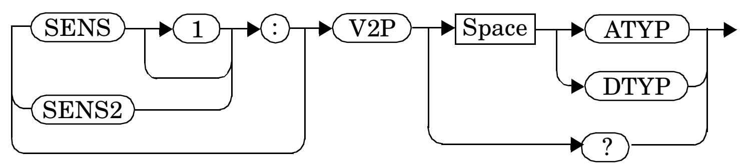
### Query Example

| | |
|---|---|
| SENS2:TRAC:UNIT? | *This command queries the current trace units for channel B.* |

# [SENSe[1]]|SENSe2:V2P ATYPe|DTYPe

This command is used to select the type of linearity correction that is applied to the channel sensors being used. For most 8480 series sensors the correct (A type or D type) linearity correction table is automatically selected. However, for the V8486A and W8486A sensors the automatic selection must be overridden and the D type (diode) correction selected.

**Syntax**




**Example**

| | |
|---|---|
| `SENS2:V2P DTYP` | *This command selects the D type linearity correction to be applied to channel B.* |

**Reset Condition** On reset, the linearity correction is set for A type.

**Query** `[SENSe[1]]|SENSe2:V2P?`

The query returns the current type of linearity correction being used (A type or D type).

**Query Example**

| | |
|---|---|
| `SENS:V2P?` | *This command queries which linearity correction type is currently being used on channel A.* |

**Error Messages** If no sensor is connected or the sensor is not an A type, the error -241, “Hardware missing” occurs.

# 10 STATus Subsystem

# STATus Subsystem

The `STATus` command subsystem enables you to examine the status of the power meter by monitoring the following status registers:

- Device status register.
- Operation status register.
- Questionable status register.

The contents of these and other registers in the power meter are determined by one or more status registers.

Table 10-1 summarizes the effects of various commands and events on these status registers:

**Table 10-1 Commands and events affecting Status Registers**

| Status Register | *RST | *CLS | Power On | STATus: PRESet |
|---|---|---|---|---|
| SCPI Transition Filters (NTR and PTR registers) | none | none | preset | preset |
| SCPI Enable Registers | none | none | preset | preset |
| SCPI Event Registers | none | clear | clear | none |
| SCPI Error/Event Queue enable | none | none | preset | preset |
| SCPI Error/Event Queue | none | clear | clear | none |
| IEEE488.2 Registers ESE SRE | none | none | clear | none |
| IEEE488.2 Registers SESR STB | none | clear | clear | none |

The contents of the status registers are examined using the following status register set commands:

```
:CONDition?
:ENABle <NRf>|<non-decimal numeric>
[:EVENt?]
:NTRansition <NRf>|<non-decimal numeric>
:PTRansition <NRf>|<non-decimal numeric>
```

Each of these can be used to examine any of the following eleven status registers:

**Examples:**

- To use the :CONDition? command to examine the STATus:DEVice register:

  STATus:DEVice:CONDition?

- To use the :NTRansition command to examine the STATus:OPERation:SENSe[:SUMMary] register:

  STATus:OPERation:SENSe[:SUMMary]:NTRansition

This chapter describes the status register set commands and the status registers which they are used to examine.

# Status Register Set Commands

This section describes the five status register set commands. Each can be used to examine all of the eleven status registers listed on page 325.

To apply a command to a specific register, prefix the command with the name of the appropriate register. For example, to apply the `:ENABle` command to the `STATus:QUEStionable register, use the following command:`

`STATus:QUEStionable:ENABle.`

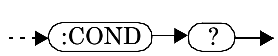
The Status Register Set commands detailed in this section are:

<table>
<tr><th>Keyword</th><th>Parameter Form</th><th>Notes</th><th>Page</th></tr>
<tr><td>:CONDition?</td><td></td><td>[query only]</td><td>page 326</td></tr>
<tr><td>:ENABle</td><td><NRf>|<non-decimal numeric></td><td></td><td>page 327</td></tr>
<tr><td>[:EVENt?]</td><td></td><td>[query only]</td><td>page 327</td></tr>
<tr><td>:NTRansition</td><td><NRf>|<non-decimal numeric></td><td></td><td>page 328</td></tr>
<tr><td>:PTRansition</td><td><NRf>|<non-decimal numeric></td><td></td><td>page 328</td></tr>
</table>

## :CONDition?

This query returns a 16 bit decimal-weighted number representing the bits set in the Condition Register of the SCPI Register Set you require to control. The format of the return is `<NR1>` in the range of 0 to 32767 ($2^{15}$-1). The contents of the Condition Register remain unchanged after it is read.

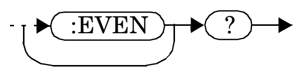
### Syntax



## [:EVENt]?

This query returns a 16 bit decimal-weighted number representing the bits set in the Event Register of the SCPI Register Set you require to control. The format of the return is `<NR1>` in the range of 0 to 32767 ($2^{15}$-1). This query clears all bits in the register to 0.

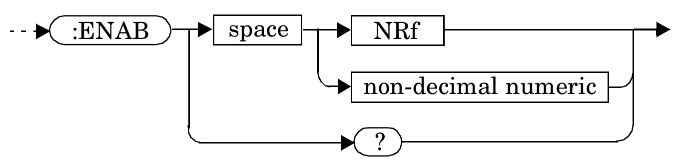
**Syntax**




## :ENABle <NRf>|<non-decimal numeric>

This command sets the Enable Register of the particular SCPI Register Set you require to control. The parameter value, when rounded to an integer and expressed in base 2 has its first 15 bits written into the Enable Register of the SCPI Register Set concerned. The last bit (bit 15) is always set to 0.

**Syntax**




**Parameters**

| Type | Description | Range of Values |
|---|---|---|
| `NRf` | The value used to set the Enable Register. | 0 to $2^{16}$-1 |
| `non-decimal numeric` | | |

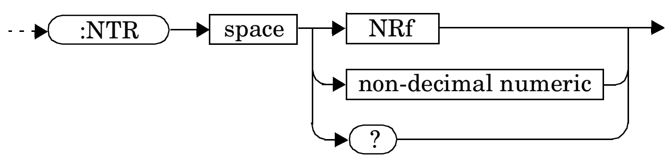
**Query** `:ENABle?`

The query returns a 15 bit decimal-weighted number representing the contents of the Enable Register of the SCPI Register Set being queried. The format of the return is `<NR1>` in the range of 0 to 32767 ($2^{15}$-1).

## :NTRansition <NRf>|<non-decimal numeric>

This command sets the Negative Transition Register of the SCPI Register Set you require to control. The parameter value, when rounded to an integer and expressed in base 2 has its first 15 bits written into the Negative Transition Register of the SCPI Register Set concerned. The last bit (bit 15) is always set to 0.

**Syntax**




**Parameters**

| Type | Description | Range of Values |
|---|---|---|
| `NRf` | The value used to set the NTR Register. | 0 to $2^{16}$-1 |
| `non-decimal numeric` | | |

**Query**

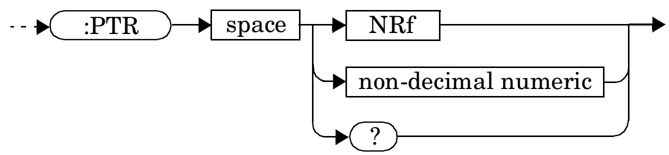
`:NTRansition?`

The query returns a 15 bit decimal-weighted number representing the contents of the Negative Transition Register of the SCPI register set being queried. The format of the return is `<NR1>` in the range of 0 to 32767 ($2^{15}$-1).

## :PTRansition <NRf>|<non-decimal numeric>

This command is used to set the Positive Transition Register of the SCPI Register Set you require to control. The first 15 bits of the input parameter are written into the Positive Transition Register of the SCPI Register Set concerned. The last bit (bit 15) is always set to 0.

### Syntax




### Parameters

| Type | Description | Range of Values |
|---|---|---|
| `NRf` | The value used to set the PTR Register. | 0 to $2^{16}$-1 |
| `non-decimal numeric` | | |

### Query

`:PTRansition?`

The query returns a 15 bit decimal-weighted number representing the contents of the Positive Transition Register of the SCPI register set being queried. The format of the return is `<NR1>` in the range of 0 to 32767 ($2^{15}$-1).

# Device Status Register Sets

The status registers contain information which give device status information. The contents of the individual registers of these register sets may be accessed by appending the commands listed in "Status Register Set Commands".

The following command descriptions detail the SCPI register you require to control but do not detail the register set commands.

The one device status register set is:

`STATus:DEVice:`

The following bits in these registers are used by the power meter:

| Bit Number | Decimal Weight | Definition |
|---|---|---|
| 0 | - | Not used |
| 1 | 2 | Channel A sensor connected |
| 2 | 4 | Channel B sensor connected (N1912A only) |
| 3 | 8 | Channel A sensor error |
| 4 | 16 | Channel B sensor error (N1912A only) |
| 7-15 | - | Not used |
| 14 | 16384 | Front Panel key press |
| 15 | - | Bit 15 always 0 |

The Channel A and B sensor connected bits (bits 1 and 2), when queried with the `STATus:DEVice:CONDition?` query are set to:

- `1`, when a power sensor is connected.
- `0`, when no power sensor is connected.

The Channel A and B sensor connected bits (bits 1 and 2), when queried with the `STATus:DEVice:EVENt?` query indicate whether a power sensor has been connected or disconnected depending on the state of the corresponding bits of `STATus:DEVice:NTRansition` and `STATus:DEVice:PTRansition`. If the corresponding bit in:

- `STATus:DEVice:NTRansition` is `1`, then `STATus:DEVice:EVENt?` is set when a power sensor is disconnected.
- `STATus:DEVice:PTRansition` is `1`, then `STATus:DEVice:EVENt?` is set when a power sensor is connected.

**NOTE** Querying `STATus:DEVice:EVENt?` clears the `STATus:DEVice:EVENt?` register.

The Channel A and B sensor error bits (3 and 4) are set to:

- `1`, if the P-series or E-series power sensor EEPROM has failed or if there are power sensors connected to both the rear and front panel connectors.
- `0`, for every other condition.

The Front Panel key press bit (bit 14), when queried with the `STATus:DEVice:EVENt?` query indicates whether any front panel keys have been pressed since power up or since you last queried the device status register. This bit ignores the `:NTRansition`, and `:PTRansition registers` and a `:CONDition?` query always returns a `0`.

# Operation Register Sets

The following registers contain information which is part of the power meter's normal operation. The contents of the individual registers of these register sets may be accessed by appending the commands listed in "Status Register Set Commands".

The following command descriptions detail the SCPI register you require to control but do not detail the Register Set commands.

The seven Operation Register Sets are:

`STATUS:OPERation`

`STATus:OPERation:CALibrating[:SUMMary]`

`STATus:OPERation:LLFail[:SUMMary]`

`STATus:OPERation:MEASuring[:SUMMary]`

`STATus:OPERation:SENSe[:SUMMary]`

`STATus:OPERation:TRIGger[:SUMMary]`

`STATus:OPERation:ULFail[:SUMMary]`

Further information on these register sets is provided on the following pages.

# STATus:OPERation

The operation status register set contains conditions which are a part of the operation of the power meter as a whole.

The following bits in these registers are used by the power meter:

| Bit Number | Decimal Weight | Definition |
|---|---|---|
| 0 | 1 | CALibrating Summary |
| 1 - 3 | - | Not used |
| 4 | 16 | MEASuring Summary |
| 5 | 32 | Waiting for TRIGger Summary |
| 6 - 9 | - | Not used |
| 10 | 1024 | SENSe Summary |
| 11 | 2048 | Lower Limit Fail Summary |
| 12 | 4096 | Upper Limit Fail Summary |
| 13 to 15 | - | Not used (bit 15 always 0) |

### Syntax

STAT → :OPER →

# STATus:OPERation:CALibrating[:SUMMary]

The operation status calibrating summary register set contains information on the calibrating status of the power meter.

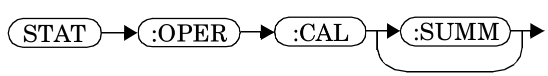
The following bits in these registers are used by the power meter:

| Bit Number | Decimal Weight | Definition |
|---|---|---|
| 0 | - | Not used |
| 1 | 2 | Channel A CALibrating Status |
| 2 | 4 | Channel B CALibrating Status (N1912A only) |
| 3-15 | - | Not used |

These bits are set at the beginning of zeroing (`CALibration:ZERO:AUTO ONCE`) and at the beginning of calibration (`CALibration:AUTO ONCE`). Also for the compound command/query `CALibration[:ALL]?`, this bit is set at the beginning of the calibration sequence.

These bits are cleared at the end of zeroing or calibration.

### Syntax



# STATus:OPERation:LLFail[:SUMMary]

The operation status lower limit fail summary register set contains information on the lower limit fail status of the power meter.

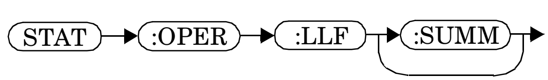
The following bits in these registers are used by the power meter:

| Bit Number | Decimal Weight | Definition |
|---|---|---|
| 0 | - | Not used |
| 1 | 2 | Channel A LLFail Status |
| 2 | 4 | Channel B LLFail Status (N1912A only) |
| 3 | 8 | Upper window LLFail Status |
| 4 | 16 | Lower widow LLFail Status |
| 5 | 32 | Upper window lower measurement LLFail Status |
| 6 | 64 | Lower window lower measurement LLFail Status |
| 7-15 | - | Not used |

The appropriate bits are set if a channel lower limit test fails or a window lower limit test fails.

These bits are cleared if a measurement is made and the test is enabled and passes.

### Syntax



# STATus:OPERation:MEASuring[:SUMMary]

The operation status measuring summary register set contains information on the measuring status of the power meter.

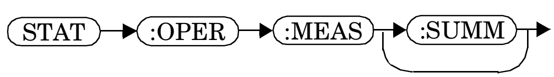
The following bits in these registers are used by the power meter:

| Bit Number | Decimal Weight | Definition |
|---|---|---|
| 0 | - | Not used |
| 1 | 2 | Channel A MEASuring Status |
| 2 | 4 | Channel B MEASuring Status (N1912A only) |
| 3-15 | - | Not used |

These bits are set when the power meter is taking a measurement.

These bits are cleared when the measurement is finished.

### Syntax



# STATus:OPERation:SENSe[:SUMMary]

The operation status sense summary register set contains information on the status of the power sensors.

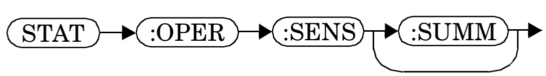
The following bits in these registers are used by the power meter:

| Bit Number | Decimal Weight | Definition |
|---|---|---|
| 0 | - | Not used |
| 1 | 2 | Channel A SENSe Status |
| 2 | 4 | Channel B SENSe Status (N1912A only) |
| 3-15 | - | Not used |

These bits are set when the power meter is reading data from the E-series power sensor EEPROM.

These bits are cleared when the power meter is not reading data from the E-series power sensor EEPROM.

## Syntax



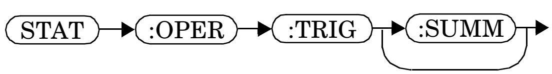
# STATus:OPERation:TRIGger[:SUMMary]

The operation status trigger summary register set contains information on the trigger status of the power meter.

The following bits in these registers are used by the power meter:

| Bit Number | Decimal Weight | Definition |
|---|---|---|
| 0 | - | Not used |
| 1 | 2 | Channel A TRIGger Status |
| 2 | 4 | Channel B TRIGger Status (N1912A only) |
| 3-15 | - | Not used |

### Syntax



# STATus:OPERation:ULFail[:SUMMary]

The operation status upper limit fail summary register set contains information on the upper limit fail status of the power meter.

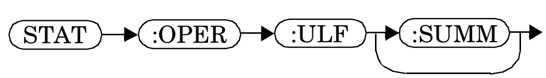
The following bits in these registers are used by the power meter:

| Bit Number | Decimal Weight | Definition |
|---|---|---|
| 0 | - | Not used |
| 1 | 2 | Channel A ULFail Status |
| 2 | 4 | Channel B ULFail Status (N1912A only) |
| 3 | 8 | Upper window ULFail Status |
| 4 | 16 | Lower window ULFail Status |
| 5 | 32 | Upper window lower measurement LLFail Status |
| 6 | 64 | Lower window lower measurement LLFail Status |
| 7-15 | - | Not used |

The appropriate bits are set if a channel upper limit test fails or a window upper limit test fails.

These bits are cleared if a measurement is made and the test is enabled and passes.

### Syntax

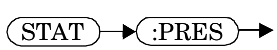


# STATus:PRESet

`PRESet` sets a number of the status registers to their preset values as shown below - all other registers are unaffected. Bit 15 is always `0`.

| Register | Filter/Enable | PRESet Value |
|---|---|---|
| OPERational | ENABle | all zeros |
| | PTR | all ones |
| | NTR | all zeros |
| QUEStionable | ENABle | all zeros |
| | PTR | all ones |
| | NTR | all zeros |
| All Others | ENABle | all ones |
| | PTR | all ones |
| | NTR | all zeros |

### Syntax



# Questionable Register Sets

The questionable register sets contain information which gives an indication of the quality of the data produced by the power meter. The contents of the individual registers in these register sets may be accessed by appending the commands listed in “Status Register Set Commands”.

The following command descriptions detail the SCPI register you require to control but do not detail the register set commands.

The three questionable register sets are:

`STATus:QUEStionable`

`STATus:QUEStionable:CALibration[:SUMMary]`

`STATus:QUEStionable:POWer[:SUMMary]`

# STATus:QUEStionable

The questionable register set contains bits that indicate the quality of various aspects of signals processed by the power meter.

The following bits in these registers are used by the power meter:

| Bit Number | Decimal Weight | Definition |
|---|---|---|
| 0 to 2 | - | Not used |
| 3 | 8 | POWer Summary |
| 4 to 7 | - | Not used |
| 8 | 256 | CALibration Summary |
| 9 | 512 | Power On Self Test |
| 10 to 15 | - | Not Used (bit 15 always 0) |

Bit 3 is set by the logical OR outputs of the `STATus:QUEStionable:POWer:SUMMary` register set.

Bit 8 is set by the logical OR outputs of the `STATus:QUEStionable:CALibration:SUMMary` register set.

Bit 9 is set if power-on self-test fails, and cleared if it passes.

### Syntax

```
STAT → :QUES →
```

# STATus:QUEStionable:CALibration[:SUMMary]

The questionable calibration summary register set contains bits which give an indication of the quality of the data produced by the power meter due to its calibration status.

The following bits in these registers are used by the power meter:

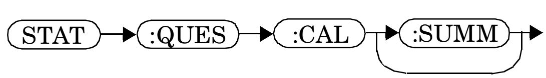
| Bit Number | Decimal Weight | Definition |
|---|---|---|
| 0 | - | Not used |
| 1 | 2 | Summary of Channel A CALibration |
| 2 | 4 | Summary of Channel B CALibration (N1912A only) |
| 3-15 | - | Not used |

These bits are set by the following:

- Error -231, “Data questionable; CH<A|B>:ZERO ERROR”
- Error -231, “Data questionable; CAL ERROR”
- Error -231, “Data questionable; CAL ERROR ChA”
- Error -231, “Data questionable; CAL ERROR ChB”

These bits are cleared when any of the three commands listed above succeed and no errors are placed on the error queue.

## Syntax



# STATus:QUEStionable:POWer[:SUMMary]

The questionable power summary register set contain bits that indicate the quality of the power data being acquired by the power meter.

The following bits in these registers shall be used by the power meter:

| Bit Number | Decimal Weight | Definition |
|---|---|---|
| 0 | - | Not used |
| 1 | 2 | Channel A Power |
| 2 | 4 | Channel B Power (N1912A only) |
| 3 | 8 | Upper Window Power |
| 4 | 16 | Lower Window Power |
| 5 | 32 | Channel A Please Zero |
| 6 | 64 | Channel B Please Zero (N1912A only) |
| 7 | 128 | Upper Window Lower Measurement Power |
| 8 | 256 | Lower Window Lower Measurement Power |

Bit 1 is set when any of the following errors occur:

- Error -231, “Data questionable;Input Overload” (N1912A only)
- Error -231, “Data questionable;Input Overload ChA” (N1912A only)

Bit 2 is set when the following error occurs:

- Error -231, “Data questionable;Input Overload ChB” (N1912A only)

Bits 3 is set when the following error occurs:

- Error -230, “Data corrupt or stale”
- Error -231, “Data questionable;Upper window log error”

Bit 4 is set when the following error occurs:

- Error -230, “Data corrupt or stale”
- Error -231, “Data questionable;Lower window log error”

Bit 5 is set when the following condition occurs:

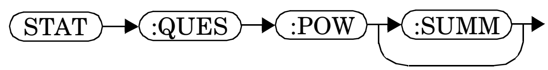
- Channel A requires zeroing

Bit 6 is set when the following condition occurs (N1912A only):

- Channel B requires zeroing

These bits are cleared when no errors or events are detected by the power meter during a measurement covering the causes given for it to set.

**Syntax**



# 11 SYSTem Subsystem

# SYSTem Subsystem

The SYStem command subsystem is used to:

- Return error numbers and messages from the power meter.
- Preset the power meter.
- Set the GPIB address.
- Set the LAN address.
- Set the command language.
- Query the SCPI version.

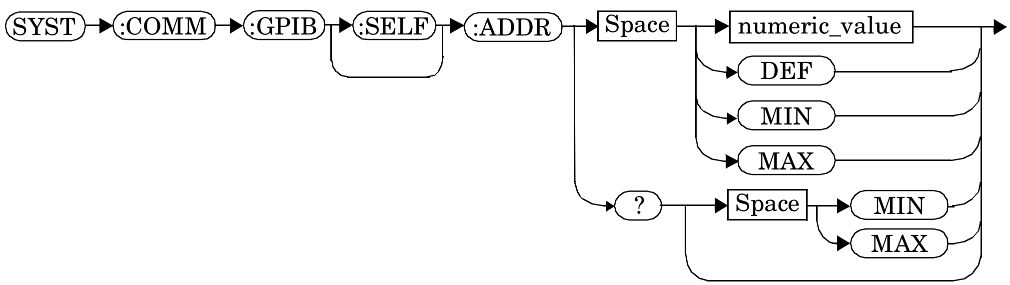
| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
| SYSTem | | | |
| :COMMunicate | | | |
| :GPIB | | | |
| [:SELF] | | | |
| :ADDRess | <numeric_value> | | page 350 |
| :LAN | | | |
| :AIP | | | |
| [:STATe] | <Boolean> | | page 352 |
| :CURRent | | | |
| :ADDRess? | | [query only] | page 353 |
| :DGATeway? | | [query only] | page 354 |
| :DNAMe? | | [query only] | page 355 |
| :SMASk? | | [query only] | page 356 |
| :ADDRess | <character_data> | | page 357 |
| :DGATeway | <character_data> | | page 358 |
| :DHCP | | | |
| [:STATe] | <Boolean> | | page 359 |
| :DNAMe | <character_data> | | page 360 |
| :HNAMe | <character_data> | | page 361 |

| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
| :MAC? | | [query only] | page 362 |
| :RESTart | | [no query] | page 363 |
| :SMASk | <character_data> | | page 364 |
| :DISPLAY | | | |
| :BMP? | | [query only] | page 365 |
| :HELP | | | |
| :HEADers? | | [query only] | page 366 |
| :LOCal | | | page 367 |
| :PRESet | character_data | [event; no query] | page 368 |
| :REMote | | | page 430 |
| :RWLock | | | page 431 |
| :VERSion? | | [query only] | page 432 |

# SYSTem:COMMunicate:GPIB[:SELF]:ADDRess <numeric_value>

This command sets the GPIB address of the power meter.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | A numeric value for the address.<br>• DEF: the default value is 13.<br>• MIN: 0.<br>• MAX: 30. | 0 to 30<br>DEF<br>MIN<br>MAX |

### Example

| | |
|---|---|
| `SYST:COMM:GPIB:ADDR 13` | *This command sets the GPIB address to 13.* |

### Query

`SYSTem:COMMunicate:GPIB[:SELF]:ADDRess? MIN|MAX`

The query returns the current setting of the GPIB address or the values associated with MIN and MAX.

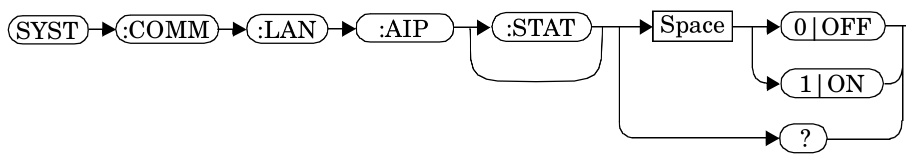
### Query Example

| | |
|---|---|
| `SYST:COMM:GPIB:ADDR?` | *This command queries the setting of the GPIB address.* |

# SYSTem:COMMunicate:LAN:AIP[:STATe] <Boolean>

This command enables the AutoIP protocol to dynamically assign the IP address when connecting to the power meter in an isolated (non-site) LAN network (for example, laptop to power meter).

### Syntax




### Example

| | |
|---|---|
| `SYST:COMM:LAN:AIP ON` | *This command enables the AutoIP* |

### Query

`SYSTem:COMMunicate:LAN:AIP?`

- `1` is returned if AutoIP is enabled.
- `0` is returned if AutoIP is disabled.

### Query Example

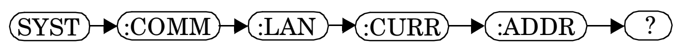
| | |
|---|---|
| `SYST:COMM:LAN:AIP?` | *This command queries the state of the AutoIP.* |

# SYSTem:COMMunicate:LAN:CURRent:ADDRess?

This command returns the current setting of the IP address in use by the power meter.

**NOTE** If DHCP or AutoIP are enabled and successful, then one of these IP address modes assigns the IP address, otherwise it is the static IP address.

### Syntax




### Example

| | |
|---|---|
| `SYST:COMM:LAN:CURR:ADDR?` | *This command queries the current setting of the IP address.* |

# SYSTem:COMMunicate:LAN:CURRent:DGATeway?

This command returns the current setting of the LAN IP router/gateway address in use by the power meter.

**NOTE** If DHCP or AutoIP are enabled and successful, then one of these IP address modes assigns the LAN IP router/gateway address, otherwise it is the static LAN IP router/gateway address.

### Syntax

SYST → :COMM → :LAN → :CURR → :DGAT → ?

### Example

| | |
|---|---|
| `SYST:COMM:LAN:CURR:DGAT?` | *This command queries the current setting of the LAN IP router/gateway address.* |

# SYSTem:COMMunicate:LAN:CURRent:DNAMe?

This command returns the current setting of the LAN domain name in use by the power meter.

**NOTE** If DHCP or AutoIP are successfully enabled, then one of these IP address modes assign the LAN domain name, otherwise it is the static LAN domain name.

### Syntax

SYST → :COMM → :LAN → :CURR → :DNAM → ?

### Example

| | |
|---|---|
| `SYST:COMM:LAN:CURR:DNAM?` | *This command queries the current setting of the LAN domain name.* |

# SYSTem:COMMunicate:LAN:CURRent:SMASk?

This command returns the current setting of the LAN subnet mask in use by the power meter.

**NOTE** If DHCP or AutoIP are successfully enabled, then one of these IP address modes assign the LAN subnet mask, otherwise it is the static LAN subnet mask.

### Syntax

SYST → :COMM → :LAN → :CURR → :SMAS → ?

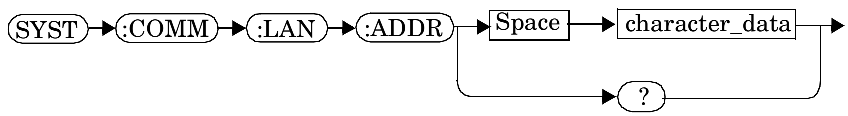
### Example

| | |
|---|---|
| `SYST:COMM:LAN:CURR:SMAS?` | *This command queries the current setting of the LAN subnet mask.* |

# SYSTem:COMMunicate:LAN:ADDRess <character_data>

This command sets the LAN (IP) address of the power meter.

### Syntax




### Parameters

| Item | Description | Range of Values |
|---|---|---|
| character_data | Numeric character values for the address. Up to 15 characters, formatted as follows: A.B.C.D where A, B, C, D = 0 to 225 | 0 to 225 (no embedded spaces) |

### Example

| | |
|---|---|
| `SYST:COMM:LAN:ADDR '130.015.156.255'` | *This command sets the LAN IP address to 130.015.156.255.* |

### Query

`SYSTem:COMMunicate:LAN:ADDRess?`

The query returns the current setting of the LAN address.

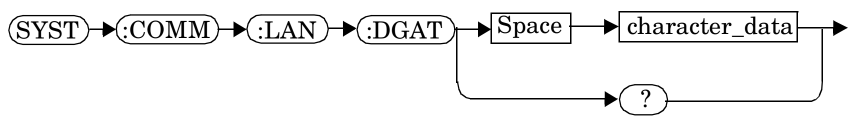
### Query Example

| | |
|---|---|
| `SYST:COMM:LAN:ADDR?` | *This command queries the setting of the LAN IP address.* |

# SYSTem:COMMunicate:LAN:DGATeway <character_data>

This command sets the LAN IP router/gateway address for the power meter.

### Syntax




### Parameters

| Item | Description | Range of Values |
|---|---|---|
| character_data | Numeric character values for the address. Up to 15 characters, formatted as follows: A.B.C.D where A, B, C, D = 0 to 225 | 0 to 225 (no embedded spaces) |

### Example

| | |
|---|---|
| `SYST:COMM:LAN:DGAT '130.2.6.200'` | *This command sets the gateway address to 130.2.6.200.* |

### Query

`SYSTem:COMMunicate:LAN:DGAT?`

The query returns the current setting of the LAN gateway address.

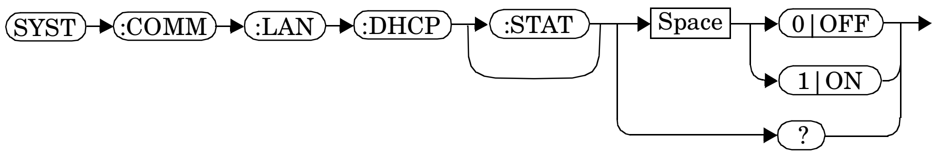
### Query Example

| | |
|---|---|
| `SYST:COMM:LAN:DGAT?` | *This command queries the setting of the gateway address.* |

# SYSTem:COMMunicate:LAN:DHCP[:STATe] <Boolean>

This command enables the dynamic host configuration protocol.

### Syntax




### Example

| | |
|---|---|
| `SYST:COMM:LAN:DHCP ON` | *This command enables the DHCP* |

### Query

`SYSTem:COMMunicate:LAN:DHCP?`

- `1` is returned if DHCP is enabled.
- `0` is returned if DHCP is disabled.

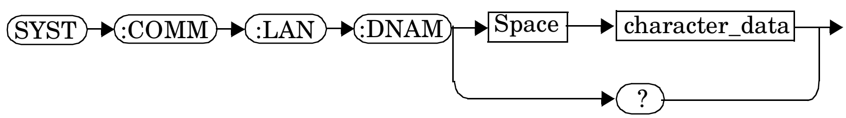
### Query Example

| | |
|---|---|
| `SYST:COMM:LAN:DHCP?` | *This command queries the state of the DHCP.* |

# SYSTem:COMMunicate:LAN:DNAMe <character_data>

This command sets the domain name for the power meter.

### Syntax




### Parameters

| Item | Description | Range of Values |
|---|---|---|
| character_data | Character values of up to 16 characters | Maximum of 16 characters |

### Example

| | |
|---|---|
| `SYST:COMM:LAN:DNAM 'myco.com'` | *This command sets the hostname to myco.com.* |

### Query

`SYSTem:COMMunicate:LAN:DNAM?`

The query returns the current setting of the LAN domain name.

### Query Example

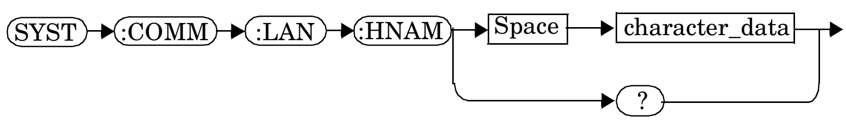
| | |
|---|---|
| `SYST:COMM:LAN:DNAM?` | *This command queries the setting of the domain name.* |

# SYSTem:COMMunicate:LAN:HNAMe <character_data>

This command sets the hostname for the power meter.

The factory default setting of hostname is in this format:
**A-** + **product number** + **-** + **suffix 5 digits of serial number**.
For example, **A-N1911A-00204**

### Syntax




### Parameters

| Item | Description | Range of Values |
|---|---|---|
| character_data | Character values of up to 15 characters | Maximum of 15 characters |

### Example

`SYST:COMM:LAN:HNAM 'PowerMeter1'` — *This command sets the hostname to PowerMeter1.*

### Query

`SYSTem:COMMunicate:LAN:HNAM?`

The query returns the current setting of the LAN hostname.

### Query Example

`SYST:COMM:LAN:HNAM?` — *This command queries the setting of the hostname.*

# SYSTem:COMMunicate:LAN:MAC?

This query returns the LAN MAC address.

### Syntax

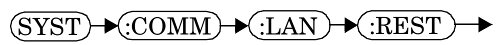

### Example

| | |
|---|---|
| `SYST:COMM:LAN:MAC?` | *This command queries the current MAC address.* |

# SYSTem:COMMunicate:LAN:RESTart

This command restarts the power meter's network stack; any LAN configuration changes can only take effect after this is performed.

### Syntax



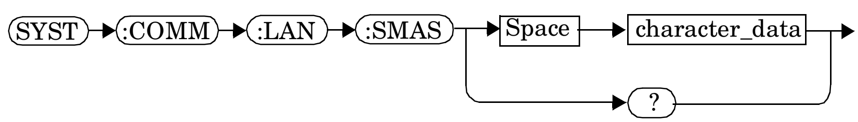

### Example

| | |
|---|---|
| `SYST:COMM:LAN:REST` | *This command restarts the LAN network with new configuration.* |

# SYSTem:COMMunicate:LAN:SMASk <character_data>

This command sets the subnet mask of the power meter.

### Syntax




### Parameters

| Item | Description | Range of Values |
|---|---|---|
| character_data | Numeric character values for the address. Up to 15 characters, formatted as follows: A.B.C.D where A, B, C, D = 0 to 225 | 0 to 225 (no embedded spaces) |

### Example

| | |
|---|---|
| `SYST:COMM:LAN:SMAS '255.255.248.0'` | *This command sets the subnet mask to 255.255.248.0* |

### Query

`SYSTem:COMMunicate:LAN:SMASk?`

The query returns the current setting of the LAN subnet mask.

### Query Example

| | |
|---|---|
| `SYST:COMM:LAN:SMAS?` | *This command queries the setting of the LAN subnet mask.* |

# SYSTem:DISPlay:BMP

This command returns the display image in bitmap format.

This command is limited to a maximum of five image returns per second.

**NOTE** It is not recommended to use this command in Fast Mode, as it slows down the measurement rate.

### Syntax

SYST → :DISP → :BMP → ?

### Example

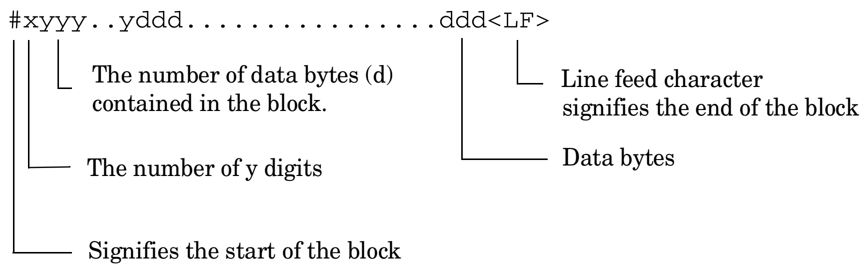
| | |
|---|---|
| `SYST:DISP:BMP?` | *This command returns the display image in bitmap format.* |

# SYSTem:HELP:HEADers?

This query returns a list of all SCPI commands supported by the instrument.

Data is returned in IEEE 488.2 arbitrary block program data format as shown in Figure 11-1.

**Figure 11-1** **IEEE 488.2 Arbitrary Block Program Data Format**

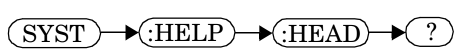



Example: if there are 12435 data bytes, y = 12435 and x = 5

Each point in the trace is represented as an IEEE 754 32 bit floating point number, made up of four bytes in the data block. The MS byte is transmitted first. Each complete block is terminated by a line feed.

Commands are listed in alphabetical order.

### Syntax



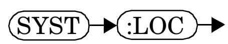

### Example

| | |
|---|---|
| `SYST:HELP:HEAD?` | *This command returns the SCPI commands supported by the instrument.* |

# SYSTem:LOCal

This command unlocks the front panel keypad and enables the power meter to be controlled from the front panel. The power meter display status reporting line shows "LCL".

### Syntax




### Example

| | |
|---|---|
| `SYST:LOC` | *This command unlocks the power meter front panel keypad and enables local front panel control.* |

# SYSTem:PRESet <character_data>

This command presets the power meter to values appropriate for measuring the communications format specified by <character_data>. The power meter is preset to default values if no value or the value DEFault is supplied.

**NOTE** DEFault settings apply to both `*RST` and to `SYSTem:PREset DEFault` unless stated otherwise.

For further information on preset configurations, refer to Table 11-1 through to Table 11-49.

Command results differ according to the sensor(s) connected to the power meter:

- If connected to a dual channel power meter, a P-series or E9320 sensor are connected to one channel, and another model sensor is connected to the other channel, the channel connected to the P-series or E9320 sensor is set up according to the <character_data> value and the other channel is set to DEFault values.
- If two P-series or E9320 sensor sensors are connected to a dual channel power meter, both channels are set to the same values except for bandwidth which is set to an appropriate value for each sensor.

### Primary and secondary channels

Dual channel meter channels are defined as either primary or secondary. The primary channel is always the trigger master and primary channel measurements occupy a greater share of the display space than secondary channel measurements.

- If a dual channel meter has a P-series sensor connected, the P-series or E9320 sensor channel is the primary channel. In such cases the primary channel could be either channel A or channel B. The other model’s channel is the secondary channel.
- If a dual channel meter has two, P-series or E9320 sensors, connected to it, the primary channel is always channel A and the secondary channel is channel B.

### Syntax

SYST → :PRES → Space → character_data →

### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | A communications format which determines the preset values. Refer to Table 11-1 through to Table 11-49 for the preset values for each format. | `DEFault`<br>`GSM900`<br>`EDGE`<br>`NADC`<br>`BLUetooth`<br>`CDMAone`<br>`WCDMA`<br>`CDMA2000`<br>`IDEN`<br>`MCPa`<br>`RADar`<br>`WL802DOT11A`<br>`WL802DOT11B`<br>`XEVDO`<br>`XEVDV`<br>`TDSCdma`<br>`DVB`<br>`HIPERLAN2` |

### Example

`SYST:PRES DEF`

*This command presets the power meter with default values. The same default values are set when the parameter is omitted.*

**Error messages**

- If a non-E-series power sensor is connected, the command can be used to set the power meter to Default settings. Attempts to set the power meter to any of the other settings result in error -241 “Hardware missing: E9320 series sensor required” occurring.
- If BLUetooth or CDMAone is selected and an E9322/6A (1.5 MHz bandwidth) or E9323/7A (5 MHz bandwidth) power sensor is not connected, error -241 “Hardware missing: Higher bandwidth E9320 sensor required on channel X. Measurements on channel X may be inaccurate” occurs.
- If WCDMA or CDMA2000 is selected and an E9323/7A (5 MHz bandwidth) power sensor is not connected, error -241 “Hardware missing: Higher bandwidth E9320 sensor required on channel X. Measurements on channel X may be inaccurate” occurs.
- If two E9320 power sensors are connected to a dual channel power meter and only one is of sufficient bandwidth to support the selected format, error -241 “Hardware missing: Higher bandwidth E9320 sensor required on channel X. Measurements on channel X may be inaccurate” occurs.

## Preset Values

### DEFault

Table 11-1 shows the power meter presets when `<character_data>` is set to `DEFault` or omitted. Values are shown for all SCPI commands:

**Table 11-1 DEFault: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| CALC[1]\|2\|3\|4:FEED[1]\|2 | "POW:AVER" | Select average measurement type |
| CALC[1]\|2\|3\|4:GAIN[:MAGN] | 0.000 dB | Display offset value |
| CALC[1]\|2\|3\|4:GAIN:STAT | OFF | Display offset disabled |
| CALC[1]\|2\|3\|4:LIM:CLE:AUTO | ON | Clear limit data at INIT |
| CALC[1]\|2\|3\|4:LIM:LOW[:DATA] | -90 dBm | Lower limit |
| CALC[1]\|2\|3\|4:LIM:STAT | OFF | Window limits checking disabled |
| CALC[1]\|2\|3\|4:LIM:UPP[:DATA] | +90 dBm | |
| CALC[1]\|2\|3\|4:MATH[:EXPR] | Agilent N1911A:<br>Upper - channel A<br>Lower - channel A<br>Agilent N1912A<br>Upper - channel A<br>Lower - channel B | Math expression |
| CALC[1]\|2\|3\|4:REL[:MAGN]:AUTO | OFF | Reference value disabled |
| CALC[1]\|2\|3\|4:REL:STAT | OFF | Relative offset disabled |
| CAL[1]\|2:ECON:STAT | OFF | TTL zero/calibration inputs disabled |
| CAL[1]\|2:RCAL | not affected | zero/cal lockout |
| CAL[1]\|2:RCF | 100.0% | Reference calibration factor |
| DISP:CONT | not affected | Display contrast |
| DISP:ENAB | ON | Display enabled |
| DISP:SCR:FORM | WIND | Display format set to windowed |
| DISP[:WIND[1]\|2]:ANAL:LOW | -70 dBm | Lower scale limit |

**Table 11-1 DEFault: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| DISP[:WIND[1]\|2]:ANAL:UPP | **20 dBm** | Upper scale limit |
| DISP[:WIND[1]\|2]:FORM | **Agilent N1911A:<br>Upper - digital<br>Lower - analog**<br><br>**Agilent N1912A<br>Upper - digital<br>Lower - digital** | Display format |
| DISP[:WIND[1]\|2]:MET:LOW | **-70.000 dBm** | Analog meter lower limit |
| DISP[:WIND[1]\|2]:MET:UPP | **+20.000 dBm** | Analog meter upper limit |
| DISP[:WIND[1]\|2\|][:NUM[1]\|2]:RES | **3** | Window resolution |
| DISP[:WIND[1]\|2]:SEL[1]\|2 | **upper window** | Window selected |
| DISP[:WIND[1]\|2][:STAT] | **ON** | Both windows enabled on display |
| DISP[:WIND[1]\|2]:TRAC:LOW | **DEF** | Maximum power |
| DISP[:WIND[1]\|2]:TRAC:UPP | **DEF** | Minimum power |
| FORM[:READ]:BORD | **normal** | Binary order |
| FORM[:READ][:DATA] | **ascii** | Data format |
| INIT[1]\|2:CONT | *RST: OFF<br>SYS:PRES ON | Power Meter in idle state<br>Power Meter in wait for trigger state |
| MEM:TABL:SEL | **not affected** | Active sensor calibration table |
| OUTP:REC[1]\|2:FEED | **not affected** | Previous measurement |
| OUTP:REC[1]\|2:LIM:LOW | **-150 dBm** | Minimum scaling value |
| OUTP:REC[1]\|2:LIM:UPP | **20 dBm** | Maximum scaling value |
| OUTP:ROSC:STAT | **OFF** | 50 MHz reference disabled |
| OUTP:TRIG:STAT | **OFF** | Trigger output signal disabled |
| [SENS[1]]\|SENS2:AVER:COUN | **4** | Filter length |
| [SENS[1]]\|SENS2:AVER:COUN:AUTO | **ON** | Auto-filtering enabled |
| [SENS[1]]\|SENS2:AVER:SDET | **1** | Step detection enabled |

**Table 11-1 DEFault: Power Meter Presets**

<table>
<tr><th>Command</th><th>Setting</th><th>Comments</th></tr>
<tr><td>[SENS[1]]|SENS2:AVER[:STAT]</td><td>ON</td><td>Averaging enabled</td></tr>
<tr><td>[SENS[1]]|SENS2:AVER2:COUN</td><td>4</td><td>Video average length</td></tr>
<tr><td>[SENS[1]]|SENS2:AVER2[:STAT]</td><td>ON</td><td>Video averaging enabled</td></tr>
<tr><td>[SENS[1]]|SENS2:BAND|BWID:VID</td><td>OFF</td><td>Sensor video bandwidth set to off</td></tr>
<tr><td>[SENS[1]]|SENS2:CORR:CFAC|GAIN[1][:INPut][:MAGNitude]</td><td>100.0%</td><td>Calibration factor</td></tr>
<tr><td>[SENS[1]]|SENS2:CORR:CSET[1]|CSET2[:SEL]</td><td>not affected</td><td>Selected sensor calibration table</td></tr>
<tr><td>[SENS[1]]|SENS2:CORR:CSET[1]|CSET2:STAT</td><td>not affected</td><td>Sensor calibration table disabled</td></tr>
<tr><td>[SENS[1]]|SENS2:CORR:DCYC|GAIN3[:INP][:MAGN]</td><td>1.000%</td><td>Duty cycle factor</td></tr>
<tr><td>[SENS[1]]|SENS2:CORR:DCYC|GAIN3:STAT</td><td>OFF</td><td>Duty cycle correction disabled</td></tr>
<tr><td>[SENS[1]]|SENS2:CORR:FDOF|GAIN4[:INP][:MAGN]</td><td>not affected</td><td>Return frequency dependent offset</td></tr>
<tr><td>[SENS[1]]|SENS2:CORR:GAIN2:STAT</td><td>OFF</td><td>Channel offset disabled</td></tr>
<tr><td>[SENS[1]]|SENS2:CORR:GAIN2:STAT[:INPut][:MAGNitude]</td><td>0.0 dB</td><td>Enter channel offset value</td></tr>
<tr><td>[SENS[1]]|SENS2:DET:FUNC</td><td>NORM</td><td>Measurement mode</td></tr>
<tr><td>[SENS[1]]|SENS2:FREQ[:CW|:FIX]</td><td>+50.000 MHz</td><td>Frequency setting</td></tr>
<tr><td>[SENSe[1]]|SENS2:MRAT</td><td>NORM</td><td>Measurement speed</td></tr>
<tr><td>[SENS[1]]|SENS2:POW:AC:RANG</td><td>upper</td><td>Upper range selected</td></tr>
<tr><td>[SENS[1]]|SENS2:POW:AC:RANG:AUTO</td><td>ON</td><td>Auto-ranging selected</td></tr>
<tr><td>[SENS[1]]|SENS2:SPE</td><td>20 readings/second</td><td>Speed</td></tr>
<tr><td>[SENS[1]]|SENS2:SWE[1]|2|3|4:OFFS:TIME</td><td>0</td><td>Set delay</td></tr>
<tr><td>[SENS[1]]|SENS2:SWE[1]|2|3|4:TIME</td><td>Gate 1: 100 us<br>Other gates: 0 sec</td><td>Set time gated period</td></tr>
</table>

**Table 11-1 DEFault: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| [SENS[1]]\|SENS2:TRACe:OFFSet:TIME | 0 | Delay |
| [SENS[1]]\|SENS2:TRACe:TIME | 100 us | Duration of trace |
| [SENS[1]]\|SENS2:V2P | ATYP | Select linearity correction |
| SYST:GPIB[:SELF]ADDR | not affected | Power meter address |
| TRAC[1]\|2:STAT | OFF | Disable trace capture |
| TRAC[1]\|2:UNIT | dBm | Trace units |
| TRIG[1]\|2:DEL:AUTO | ON | Insert settling time delay |
| TRIG[:SEQ]:DEL | 0 | Delay between recognition of trigger event and start of a measurement |
| TRIG[:SEQ]:HOLD | 1 us | Trigger holdoff |
| TRIG[:SEQ]:HYST | 0 dB | Fall/rise below/above TRIG:LEV |
| TRIG[:SEQ]:LEV | 0 dB | Power level |
| TRIG[:SEQ]:LEV:AUTO | ON | Enable automatic setting of trigger level |
| TRIG[:SEQ]:SLOP | POS | Trigger event recognized on rising edge |
| TRIG[:SEQ[1]\|2]:COUN | 1 | Trigger events for measurement cycle |
| TRIG[:SEQ[1]\|2]:DEL:AUTO | ON | Enable settling time delay |
| TRIG[:SEQ[1]\|2]:SOUR | IMM | Trigger source set up |
| UNIT:POW | dBm | Power units |
| UNIT:POW:RAT | dB | Ratio units |

### GSM900

Table 11-2 shows the power meter presets when `<character_data>` is set to `GSM900`.

The GSM900 set-up provides the following:

- Average power measurement in one GSM timeslot.
- Trace display showing “on” timeslot.

A GSM900 measurement is started by detecting the rising edge of a GSM RF burst—for example the burst emitted by a GSM mobile—using the internal RF level trigger. The trigger level is set to –20dBm. Time-gating is used to measure the average power in the useful part of a GSM burst.

Commands not listed are preset according to their `DEFault` values (for further information refer to Table 11-1.

**Table 11-2 GSM900: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| *Frequency* | | |
| `[SENS[1]]|SENS2:FREQ[:CW|:FIX]` | **+900.000 MHz** | Frequency setting |
| *Sensor measurement mode* | | |
| `[SENS[1]]|SENS2:DET:FUNC` | **NORM** | Measurement mode |
| *Sensor video bandwidth setup* | | |
| `[SENS[1]]|SENS2:BAND|BWID:VID` | **E9321A/25A: HIGH**<br>**E9322A/26A: MED**<br>**E9323A/27A: LOW**<br>**N1921/2A: LOW** | Sensor video bandwidth |
| *Gate Setup* | | |
| `[SENS[1]]|SENS2:SWE[1]|2|3|4 :OFF:TIME` | **Gate 1: 20 us**<br>**Gates 2 - 4: 0** | Delay between trigger point and time gated period. |
| `[SENS[1]]|SENS2:SWE[1]|2|3|4 :TIME` | **Gate 1: 520 us**<br>**Gates 2 - 4: 0** | Length of time gated period for time gated measurements. |
| *Trigger setup* | | |
| `TRIG[:SEQ[1]|2]:SOUR` | **INT1** | Trigger source set up and acquisition mode continuous triggering |
| `INIT:CONT` | **ON** | |

**Table 11-2 GSM900: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| TRIG[:SEQ]:LEV:AUTO | **OFF** | Disable automatic setting of the trigger level |
| TRIG[:SEQ]:LEV | **-20 dBm** | Power level |
| TRIG[:SEQ]:SLOP | **POS** | Trigger event recognized on the rising edge of a signal |
| TRIG[:SEQ]:DEL | **20 us** | Delay between recognition of trigger event and start of a measurement |
| TRIG[:SEQ]:HOLD | **4275 us** | Trigger holdoff |
| *Step detection* | | |
| [SENSe[1]]\|SENS2:AVER:SDET | **1** | Step detection enabled |
| *Trace setup* | | |
| DISP[:WIND[1]\|2]:TRAC:LOW | **+20 dBm** | Maximum power |
| DISP[:WIND[1]\|2]:TRAC:UPP | **-35 dBm** | Minimum power |
| [SENS[1]]\|SENS2:TRAC:OFFS :TIME <numeric_value> | **-40 us** | Delay between delayed trigger point and the start of the trace |
| [SENS[1]]\|SENS2:TRAC:TIME <numeric_value> | **700 us** | Length of the trace |

**Table 11-3 GSM900: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | **No Sensor** | **Non E9320 Sensor** |
| *Display setup* | | |
| Upper window | Channel A trace | Primary channel[a] trace |
| Lower window | LU single numeric | See Table 11-4 |
| *Window / measurement setup* | | |
| *Upper window / upper measurement (UU)* | | |
| Feed | N/A | N/A |

**Table 11-3 GSM900: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | **No Sensor** | **Non E9320 Sensor** |
| Measurement | N/A | N/A |
| *Upper window / lower measurement (UL)* | | |
| Feed | N/A | N/A |
| Measurement | N/A | N/A |
| *Lower window / upper measurement (LU)* | | |
| Feed | Gate 1 Channel A | Gate 1 primary channel[a] |
| Measurement | Avg | Avg |
| *Lower window / lower measurement (LL)* | | |
| Feed | DEF | See Table 11-4 |
| Measurement | DEF | See Table 11-4 |

a. For further information refer to “Primary and secondary channels” on page 368.

**Table 11-4 GSM900: Power Meter Presets For Secondary Channel Sensors**

| Function | Secondary Channel Sensor | | |
|---|---|---|---|
| | **No Sensor** | **Non P-series or E9320 Sensor** | **P-series and E9320 Sensor** |
| *Display setup* | | | |
| Lower window | LU single numeric | Dual numeric | Dual numeric |
| *Lower window / lower measurement (LL)* | | | |
| Feed | DEF | Secondary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | DEF | Avg | Avg |

a. For further information refer to “Primary and secondary channels” on page 368.

### EDGE

EDGE (Enhanced Data for Global Evolution or Enhanced Data for GSM Evolution) is an enhancement of the GSM standard. Whereas the GSM modulation scheme is GMSK which has constant amplitude, the EDGE modulation scheme is 8PSK which has variable amplitude.

The EDGE set-up provides:

- Average power measurement in an EDGE burst.
- Peak-to-average ratio in an EDGE burst.
- A trace display of the burst profile

An EDGE measurement is started by detecting the rising edge of the EDGE RF burst—for example the burst emitted by a mobile—using the internal RF level trigger. The internal level trigger is set to –20dBm. Trigger level hysteresis is used to prevent the power meter re-triggering on the varying power levels within the EDGE burst. Time-gating is used to measure the average power and the peak-to-average ratio in the useful part of the RF burst.

The following table shows the power meter presets when <character_data> is set to `EDGE`. Commands not listed are preset according to their `DEFault` values (for further information refer to Table 11-1).

**Table 11-5 EDGE: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| *Frequency* | | |
| `[SENS[1]]|SENS2:FREQ[:CW|:FIX]` | **+900.000 MHz** | Frequency setting |
| *Sensor measurement mode* | | |
| `[SENS[1]]|SENS2:DET:FUNC` | **NORM** | Measurement mode |
| *Sensor video bandwidth setup* | | |
| `[SENS[1]]|SENS2:BAND|BWID:VID` | **E9321A/25A: HIGH**<br>**E9322A/26A: MED**<br>**E9323A/27A: LOW**<br>**N1921/2A: LOW** | Sensor video bandwidth |
| *Gate Setup* | | |
| `[SENS[1]]|SENS2:SWE[1]|2|3|4 :OFF:TIME` | **Gate 1: 20 us**<br>**Gates 2 - 4: 0** | Delay between trigger point and time gated period. |

**Table 11-5 EDGE: Power Meter Presets**

<table>
<tr><th>Command</th><th>Setting</th><th>Comments</th></tr>
<tr><td>[SENS[1]]|SENS2:SWE[1]|2|3|4 :TIME</td><td>Gate 1: 520 us<br>Gates 2 - 4: 0</td><td>Length of time gated period for time gated measurements.</td></tr>
<tr><td colspan="3">Trigger setup</td></tr>
<tr><td>TRIG[:SEQ[1]|2]:SOUR</td><td>INT1</td><td rowspan="2">Trigger source set up and acquisition mode continuous triggering</td></tr>
<tr><td>INIT:CONT</td><td>ON</td></tr>
<tr><td>TRIG[:SEQ]:LEV:AUTO</td><td>OFF</td><td>Disable automatic setting of the trigger level</td></tr>
<tr><td>TRIG[:SEQ]:LEV</td><td>-20 dBm</td><td>Power level</td></tr>
<tr><td>TRIG[:SEQ]:SLOP</td><td>POS</td><td>Trigger event recognized on the rising edge of a signal</td></tr>
<tr><td>TRIG[:SEQ]:DEL</td><td>0 s</td><td>Delay between recognition of trigger event and start of a measurement</td></tr>
<tr><td>TRIG[:SEQ]:HOLD</td><td>4275 us</td><td>Trigger holdoff</td></tr>
<tr><td>TRIG[:SEQ]:HYST</td><td>3 dB</td><td>Hysteresis</td></tr>
<tr><td colspan="3">Averaging</td></tr>
<tr><td>[SENSe[1]]|SENSe2:AVER[:STATe]</td><td>ON</td><td>Averaging On</td></tr>
<tr><td>[SENSe[1]]|SENSe2:AVER:COUN</td><td>64</td><td>Averaging set to 64</td></tr>
<tr><td colspan="3">Step detection</td></tr>
<tr><td>[SENSe[1]]|SENS2:AVER:SDET</td><td>1</td><td>Step detection enabled</td></tr>
<tr><td colspan="3">Trace setup</td></tr>
<tr><td>DISP[:WIND[1]|2]:TRAC:LOW</td><td>+20 dBm</td><td>Maximum power</td></tr>
<tr><td>DISP[:WIND[1]|2]:TRAC:UPP</td><td>-35 dBm</td><td>Minimum power</td></tr>
<tr><td>[SENS[1]]|SENS2:TRAC:OFFS :TIME <numeric_value></td><td>-40 us</td><td>Delay between delayed trigger point and the start of the trace</td></tr>
<tr><td>[SENS[1]]|SENS2:TRAC:TIME <numeric_value></td><td>700 us</td><td>Length of the trace</td></tr>
</table>

**Table 11-6 EDGE: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | **No Sensor** | **Non E9320 Sensor** |
| *Display setup* | | |
| Upper window | Channel A trace | Primary channel[a] trace |
| Lower window | Dual numeric | See Table 11-7 |
| *Window / measurement setup* | | |
| *Upper window / upper measurement (UU)* | | |
| Feed | N/A | N/A |
| Measurement | N/A | N/A |
| *Upper window / lower measurement (UL)* | | |
| Feed | N/A | N/A |
| Measurement | N/A | N/A |
| *Lower window / upper measurement (LU)* | | |
| Feed | Gate 1 Channel A | Gate 1 primary channel[a] |
| Measurement | Avg | Avg |
| *Lower window / lower measurement (LL)* | | |
| Feed | Gate 1 Channel A | See Table 11-7 |
| Measurement | Pk-to-Avg | See Table 11-7 |

a. For further information refer to “Primary and secondary channels” on page 368.

**Table 11-7 EDGE: Power Meter Presets For Secondary Channel Sensors**

| Function | Secondary Channel Sensor | | |
|---|---|---|---|
| | **No Sensor** | **Non P-series or E9320 Sensor** | **P-series and E9320 Sensor** |
| *Display setup* | | | |
| Lower window | Dual numeric | Dual numeric | Dual numeric |
| *Lower window / lower measurement (LL)* | | | |
| Feed | Gate 1 primary channel[a] | Secondary channel[a] | Gate1 secondary channel[a] (channel B) |

**Table 11-7 EDGE: Power Meter Presets For Secondary Channel Sensors**

| Function | Secondary Channel Sensor | | |
|---|---|---|---|
| | No Sensor | Non P-series or E9320 Sensor | P-series and E9320 Sensor |
| Measurement | Pk-to-Avg | Avg | Avg |

a. For further information refer to “Primary and secondary channels” on page 368.

### CDMAone

The cdmaOne set-up provides:

- Average power in an IS-95 cdmaOne signal (bandwidth is less than 1.5MHz).
- Peak power and peak-to-average ratio of the signal over a defined, statistically valid number of samples. The reading is continuously refreshed. This gives an indication of how cdmaOne channel loading affects peak power and power distribution.

The measurement is a continuously gated measurement on a cdmaOne signal. Its aim is to measure the peak and average power corresponding to a <0.01% probability that there are no peaks above the returned peak reading. Time gating is therefore set to 10ms, corresponding to 200000 samples. Triggering is set to occur continuously internally to the meter. The internal trigger is set to AutoLevel. A reading over the 10ms period is returned and the reading is then re-initiated for the next 10ms period. In this way the reading always relates to a position beyond 0.01% on the CCDF curve and will refresh to track any signal or DUT changes.

The following table shows the power meter presets when <character_data> is set to `CDMAone`. Commands not listed are preset according to their `DEFault` values (for further information refer to Table 11-1):

**Table 11-8 CDMAone: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| *Frequency* | | |
| `[SENS[1]]|SENS2:FREQ[:CW|:FIX]` | **+850.000 MHz** | Frequency setting |
| *Sensor measurement mode* | | |
| `[SENS[1]]|SENS2:DET:FUNC` | **NORM** | Measurement mode |
| *Sensor video bandwidth setup* | | |
| `[SENS[1]]|SENS2:BAND|BWID:VID` | **E9321A/25A: DEF**<br>**E9322A/26A: OFF**<br>**E9323A/27A: OFF**<br>**N1921/2A: OFF** | Sensor video bandwidth |
| *Gate Setup* | | |
| `[SENS[1]]|SENS2:SWE[1]|2|3|4 :OFF:TIME` | **Gate 1: 0 s**<br>**Gates 2 - 4: 0** | Delay between trigger point and time gated period. |

**Table 11-8 CDMAone: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| [SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :TIME | **Gate 1: 10 ms**<br>**Gates 2 - 4: 0** | Length of time gated period for time gated measurements. |
| *Trigger setup* | | |
| TRIG[:SEQ[1]\|2]:SOUR | **INT1** | Trigger source set up and acquisition mode continuous triggering |
| INIT:CONT | **ON** | |
| TRIG[:SEQ]:LEV:AUTO | **ON** | Enable automatic setting of the trigger level |
| TRIG[:SEQ]:LEV | **AUTO** | Automatic Power level |
| TRIG[:SEQ]:SLOP | **POS** | Trigger event recognized on the rising edge of a signal |
| TRIG[:SEQ]:DEL | **0 s** | Delay between recognition of trigger event and start of a measurement |
| TRIG[:SEQ]:HOLD | **MIN** | Trigger holdoff |
| *Step detection* | | |
| [SENSe[1]]\|SENS2:AVER:SDET | **0** | Step detection disabled |

**Table 11-9 CDMAone: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | **No Sensor** | **Non E9320 Sensor** |
| *Display setup* | | |
| Upper window | UU single numeric | See Table 11-10 |
| Lower window | Dual numeric | See Table 11-10 |
| *Window / measurement setup* | | |
| *Upper window / upper measurement (UU)* | | |
| Feed | Gate 1 channel A | Gate 1 primary channel[a] |
| Measurement | Avg | Avg |
| *Upper window / lower measurement (UL)* | | |

**Table 11-9 CDMAone: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | No Sensor | Non E9320 Sensor |
| Feed | DEF | See Table 11-10 |
| Measurement | DEF | See Table 11-10 |
| *Lower window / upper measurement (LU)* | | |
| Feed | Gate 1 Channel A | See Table 11-10 |
| Measurement | Peak | See Table 11-10 |
| *Lower window / lower measurement (LL)* | | |
| Feed | Gate 1 Channel A | See Table 11-10 |
| Measurement | Peak to avg | See Table 11-10 |

a. For further information refer to “Primary and secondary channels” on page 368.

**Table 11-10 CDMAone: Power Meter Presets For Secondary Channel Sensors**

| Function | Secondary Channel Sensor | | |
|---|---|---|---|
| | No Sensor | Non P-series or E9320 Sensor | P-series and E9320 Sensor |
| *Display setup* | | | |
| Upper window | UU single numeric | Dual numeric | Dual numeric |
| Lower window | Dual numeric | Dual numeric | Dual numeric |
| *Upper window / lower measurement (UL)* | | | |
| Feed | DEF | Gate 1 primary channel[a] | Gate 1 primary channel[a] (channel A) |
| Measurement | DEF | Peak | Peak |
| *Lower window / upper measurement (LU)* | | | |
| Feed | Gate 1 primary channel[a] | Gate 1 primary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Peak | Pk-to-Avg | Avg |
| *Lower window / lower measurement (LL)* | | | |

**Table 11-10 CDMAone: Power Meter Presets For Secondary Channel Sensors**

| Function | Secondary Channel Sensor | | |
|---|---|---|---|
| | **No Sensor** | **Non P-series or E9320 Sensor** | **P-series and E9320 Sensor** |
| Feed | Gate 1 primary channel[a] | Secondary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Pk-to-Avg | Avg | Pk-to-Avg |

a. For further information refer to "Primary and secondary channels" on page 368.

## CDMA2000

The cdma2000 set-up provides:

- Average power in a cdma2000 signal (bandwidth <=5MHz).
- Peak power and peak-to-average ratio of the signal over a defined, statistically valid number of samples. The reading is continuously refreshed. This indicates how cdma2000 channel loading affects peak power and power distribution.

The measurement is a continuously gated measurement on a 3GPP cdma2000 signal. Its aim is to measure the peak and average power corresponding to a <0.01% probability that there are no peaks above the returned peak reading. Time gating is set to 10ms, corresponding to 200,000 samples. Triggering is set to occur continuously internally to the meter. The internal trigger is set to AutoLevel. A reading over the 10ms period is returned, then the reading is re-initiated for the next 10ms period. In this way the reading always relates to a position beyond 0.01% on the CCDF curve and will refresh to track any signal or DUT changes.

The following table shows the power meter presets when <character_data> is set to `CDMA2000`. Commands not listed are preset according to their `DEFault` values (for further information refer to Table 11-1):

**Table 11-11 cdma2000: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| *Frequency* | | |
| `[SENS[1]]\|SENS2:FREQ[:CW\|:FIX]` | **+1900.000 MHz** | Frequency setting |
| *Sensor measurement mode* | | |
| `[SENS[1]]\|SENS2:DET:FUNC` | **NORM** | Measurement mode |
| *Sensor video bandwidth setup* | | |
| `[SENS[1]]\|SENS2:BAND\|BWID:VID` | **E9321A/25A: DEF**<br>**E9322A/26A: DEF**<br>**E9323A/27A: OFF**<br>**N1921/2A: OFF** | Sensor video bandwidth |
| *Gate Setup* | | |
| `[SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :OFF:TIME` | **Gate 1: 0 s**<br>**Gates 2 - 4: 0** | Delay between trigger point and time gated period. |

**Table 11-11 cdma2000: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| [SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :TIME | **Gate 1: 10 ms**<br>**Gates 2 - 4: 0** | Length of time gated period for time gated measurements. |
| *Trigger setup* | | |
| TRIG[:SEQ[1]\|2]:SOUR | **INT1** | Trigger source set up and acquisition mode continuous triggering |
| INIT:CONT | **ON** | |
| TRIG[:SEQ]:LEV:AUTO | **ON** | Enable automatic setting of the trigger level |
| TRIG[:SEQ]:LEV | **AUTO** | Automatic Power level |
| TRIG[:SEQ]:SLOP | **POS** | Trigger event recognized on the rising edge of a signal |
| TRIG[:SEQ]:DEL | **0 s** | Delay between recognition of trigger event and start of a measurement |
| TRIG[:SEQ]:HOLD | **MIN** | Trigger holdoff |
| *Step detection* | | |
| [SENSe[1]]\|SENS2:AVER:SDET | **0** | Step detection disabled |

**Table 11-12 cdma2000: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | No Sensor | Non E9320 Sensor |
| *Display setup* | | |
| Upper window | UU single numeric | UU single numeric |
| Lower window | Dual numeric | See Table 11-13 |
| *Window / measurement setup* | | |
| *Upper window / upper measurement (UU)* | | |
| Feed | Gate 1 channel A | Gate 1 primary channel[a] |
| Measurement | Avg | Avg |
| *Upper window / lower measurement (UL)* | | |

**Table 11-12 cdma2000: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | **No Sensor** | **Non E9320 Sensor** |
| Feed | DEF | DEF |
| Measurement | DEF | DEF |
| *Lower window / upper measurement (LU)* | | |
| Feed | Gate 1 Channel A | Gate 1 primary channel[a] |
| Measurement | Peak | Peak |
| *Lower window / lower measurement (LL)* | | |
| Feed | Gate 1 Channel A | See Table 11-13 |
| Measurement | Peak to avg | See Table 11-13 |

a. For further information refer to “Primary and secondary channels” on page 368.

**Table 11-13 cdma2000: Power Meter Presets For Secondary Channel Sensors**

| Function | Secondary Channel Sensor | | |
|---|---|---|---|
| | **No Sensor** | **Non P-series or E9320 Sensor** | **P-series and E9320 Sensor** |
| *Display setup* | | | |
| Upper window | UU single numeric | Dual numeric | Dual numeric |
| Lower window | Dual numeric | Dual numeric | Dual numeric |
| *Upper window / lower measurement (UL)* | | | |
| Feed | DEF | Gate 1 primary channel[a] | Gate 1 primary channel[a] (channel A) |
| Measurement | DEF | Peak | Peak |
| *Lower window / upper measurement (LU)* | | | |
| Feed | Gate 1 primary channel[a] | Gate 1 primary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Peak | Pk-to-Avg | Avg |
| *Lower window / lower measurement (LL)* | | | |

**Table 11-13 cdma2000: Power Meter Presets For Secondary Channel Sensors**

| Function | Secondary Channel Sensor | | |
|---|---|---|---|
| | No Sensor | Non P-series or E9320 Sensor | P-series and E9320 Sensor |
| Feed | Gate 1 primary channel[a] | Secondary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Pk-to-Avg | Avg | Pk-to-Avg |

a. For further information refer to “Primary and secondary channels” on page 368.

### W-CDMA

The W-CDMA set-up provides:

- Average power in a W-CDMA signal (bandwidth <=5MHz)
- Peak power and peak-to-average ratio of the signal over a defined, statistically valid number of samples. The reading is continuously refreshed. This indicates how W-CDMA channel loading affects peak power and power distribution.

The measurement is a continuously gated measurement on a 3GPP W-CDMA signal. Its aim is to measure the peak and average power corresponding to a <0.01% probability that there are no peaks above the returned peak reading. Time gating is set to 10ms, corresponding to 200000 samples. Triggering is set to occur continuously internally to the meter. The internal trigger is set to AutoLevel. A reading over the 10ms period is returned then re-initiated for the next 10ms period. In this way the reading always relates to a position beyond 0.01% on the CCDF curve and will refresh to track any signal or DUT changes.

The following table shows the power meter presets when <character_data> is set to `WCDMA`. Commands not listed are preset according to their `DEFault` values (for further information refer to Table 11-1):

**Table 11-14 W-CDMA: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| *Frequency* | | |
| `[SENS[1]]|SENS2:FREQ[:CW|:FIX]` | **+1900.000 MHz** | Frequency setting |
| *Sensor measurement mode* | | |
| `[SENS[1]]|SENS2:DET:FUNC` | **NORM** | Measurement mode |
| *Sensor video bandwidth setup* | | |
| `[SENS[1]]|SENS2:BAND|BWID:VID` | **E9321A/25A: DEF**<br>**E9322A/26A: DEF**<br>**E9323A/27A: OFF**<br>**N1921/2A: OFF** | Sensor video bandwidth |
| *Gate Setup* | | |
| `[SENS[1]]|SENS2:SWE[1]|2|3|4 :OFF:TIME` | **Gate 1: 0 s**<br>**Gates 2 - 4: 0** | Delay between trigger point and time gated period. |

**Table 11-14 W-CDMA: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| [SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :TIME | **Gate 1: 10 ms**<br>**Gates 2 - 4: 0** | Length of time gated period for time gated measurements. |
| *Trigger setup* | | |
| TRIG[:SEQ[1]\|2]:SOUR | **INT1** | Trigger source set up and acquisition mode continuous triggering |
| INIT:CONT | **ON** | |
| TRIG[:SEQ]:LEV:AUTO | **ON** | Enable automatic setting of the trigger level |
| TRIG[:SEQ]:LEV | **AUTO** | Automatic Power level |
| TRIG[:SEQ]:SLOP | **POS** | Trigger event recognized on the rising edge of a signal |
| TRIG[:SEQ]:DEL | **0 s** | Delay between recognition of trigger event and start of a measurement |
| TRIG[:SEQ]:HOLD | **MIN** | Trigger holdoff |
| *Step detection* | | |
| [SENSe[1]]\|SENS2:AVER:SDET | **0** | Step detection disabled |

**Table 11-15 W-CDMA: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | **No Sensor** | **Non E9320 Sensor** |
| *Display setup* | | |
| Upper window | UU single numeric | See Table 11-16 |
| Lower window | Dual numeric | See Table 11-16 |
| *Window / measurement setup* | | |
| *Upper window / upper measurement (UU)* | | |
| Feed | Gate 1 channel A | Gate 1 primary channel[a] |
| Measurement | Avg | Avg |
| *Upper window / lower measurement (UL)* | | |

**Table 11-15 W-CDMA: Power Meter Presets: Window/Measurement Settings**

| **Function** | **Setting** | |
|---|---|---|
| | **No Sensor** | **Non E9320 Sensor** |
| Feed | DEF | See Table 11-16 |
| Measurement | DEF | See Table 11-16 |
| *Lower window / upper measurement (LU)* | | |
| Feed | Gate 1 Channel A | See Table 11-16 |
| Measurement | Peak | See Table 11-16 |
| *Lower window / lower measurement (LL)* | | |
| Feed | Gate 1 Channel A | See Table 11-16 |
| Measurement | Pk-to-Avg | See Table 11-16 |

a. For further information refer to “Primary and secondary channels” on page 368.

**Table 11-16 W-CDMA: Power Meter Presets For Secondary Channel Sensors**

| **Function** | **Secondary Channel Sensor** | | |
|---|---|---|---|
| | **No Sensor** | **Non P-series or E9320 Sensor** | **P-series and E9320 Sensor** |
| *Display setup* | | | |
| Upper window | UU single numeric | Dual numeric | Dual numeric |
| Lower window | Dual numeric | Dual numeric | Dual numeric |
| *Upper window / lower measurement (UL)* | | | |
| Feed | DEF | Gate 1 primary channel[a] | Gate 1 primary channel[a] (channel A) |
| Measurement | DEF | Peak | Peak |
| *Lower window / upper measurement (LU)* | | | |
| Feed | Gate 1 primary channel[a] | Gate 1 primary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Peak | Pk-to-Avg | Avg |
| *Lower window / lower measurement (LL)* | | | |

**Table 11-16 W-CDMA: Power Meter Presets For Secondary Channel Sensors**

| **Function** | **Secondary Channel Sensor** | | |
|---|---|---|---|
| | **No Sensor** | **Non P-series or E9320 Sensor** | **P-series and E9320 Sensor** |
| Feed | Gate 1 primary channel[a] | Secondary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Pk-to-Avg | Avg | Pk-to-Avg |

a. For further information refer to “Primary and secondary channels” on page 368.

### BLUetooth

The Bluetooth set-up provides:

- Average power in a Bluetooth DH1 data burst.
- Peak power in the same burst.
- Display of RF pulse in one timeslot.

The measurement is started by detecting the Bluetooth RF burst using the internal RF level trigger. The internal trigger is set to –20dBm. Time-gating is used to measure the peak and average power in a single Bluetooth DHI data burst which lasts for 366us. The DHI burst does not occupy a full Bluetooth timeslot, which lasts for 625us.

The following table shows the power meter presets when <character_data> is set to `BLUetooth`. Commands not listed are preset according to their `DEFault` values (for further information refer to Table 11-1):

**Table 11-17 BLUetooth: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| *Frequency* | | |
| `[SENS[1]]|SENS2:FREQ[:CW|:FIX]` | **+2400.000 MHz** | Frequency setting |
| *Sensor measurement mode* | | |
| `[SENS[1]]|SENS2:DET:FUNC` | **NORM** | Measurement mode |
| *Sensor video bandwidth setup* | | |
| `[SENS[1]]|SENS2:BAND|BWID:VID` | **E9321A/25A: DEF**<br>**E9322A/26A: OFF**<br>**E9323A/27A: OFF**<br>**N1921/2A: OFF** | Sensor video bandwidth |
| *Gate Setup* | | |
| `[SENS[1]]|SENS2:SWE[1]|2|3|4 :OFF:TIME` | **Gate 1: 0.2 us**<br>**Gates 2 - 4: 0** | Delay between trigger point and time gated period. |
| `[SENS[1]]|SENS2:SWE[1]|2|3|4 :TIME` | **Gate 1: 366 us**<br>**Gates 2 - 4: 0** | Length of time gated period for time gated measurements. |
| *Trigger setup* | | |

**Table 11-17 BLUetooth: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| TRIG[:SEQ[1]\|2]:SOUR | INT1 | Trigger source set up and acquisition mode continuous triggering |
| INIT:CONT | ON | |
| TRIG[:SEQ]:LEV:AUTO | OFF | Disable automatic setting of the trigger level |
| TRIG[:SEQ]:LEV | -20 dBm | Power level |
| TRIG[:SEQ]:SLOP | POS | Trigger event recognized on the rising edge of a signal |
| TRIG[:SEQ]:DEL | 0 s | Delay between recognition of trigger event and start of a measurement |
| TRIG[:SEQ]:HOLD | 650 us | Trigger holdoff |
| *Step detection* | | |
| [SENSe[1]]\|SENS2:AVER:SDET | 1 | Step detection enabled |
| *Trace setup* | | |
| DISP[:WIND[1]\|2]:TRAC:LOW | +20 dBm | Maximum power |
| DISP[:WIND[1]\|2]:TRAC:UPP | -35 dBm | Minimum power |
| [SENS[1]]\|SENS2:TRAC:OFFS :TIME <numeric_value> | -50 us | Delay between delayed trigger point and the start of the trace |
| [SENS[1]]\|SENS2:TRAC:TIME <numeric_value> | 3.8 ms | Length of the trace |

**Table 11-18 BLUetooth: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | No Sensor | Non E9320 Sensor |
| *Display setup* | | |
| Upper window | Channel A trace | Primary channel[a] trace |
| Lower window | Dual numeric | See Table 11-19 |
| *Window / measurement setup* | | |

**Table 11-18 BLUetooth: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | **No Sensor** | **Non E9320 Sensor** |
| *Upper window / upper measurement (UU)* | | |
| Feed | N/A | N/A |
| Measurement | N/A | N/A |
| *Upper window / lower measurement (UL)* | | |
| Feed | N/A | N/A |
| Measurement | N/A | N/A |
| *Lower window / upper measurement (LU)* | | |
| Feed | Gate 1 Channel A | Gate 1 primary channel[a] |
| Measurement | Avg | Avg |
| *Lower window / lower measurement (LL)* | | |
| Feed | Gate 1 Channel A | See Table 11-19 |
| Measurement | Peak | See Table 11-19 |

a. For further information refer to “Primary and secondary channels” on page 368.

**Table 11-19 BLUetooth: Power Meter Presets For Secondary Channel Sensors**

| Function | Secondary Channel Sensor | | |
|---|---|---|---|
| | **No Sensor** | **Non P-series or E9320 Sensor** | **E9320 Sensor** |
| *Display setup* | | | |
| Lower window | Dual numeric | Dual numeric | Dual numeric |
| *Lower window / lower measurement (LL)* | | | |
| Feed | Gate 1 primary channel1 | Secondary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Peak | Avg | Avg |

a. For further information refer to “Primary and secondary channels” on page 368.

### MCPA

The following table shows the power meter presets when <character_data> is set to `MCPa`. Commands not listed are preset according to their `DEFault` values (for further information refer to Table 11-1).

**Table 11-20 MPCA: Power Meter Presets**

<table>
<tr><th>Command</th><th>Setting</th><th>Comments</th></tr>
<tr><td colspan="3">Frequency</td></tr>
<tr><td>[SENS[1]]|SENS2:FREQ[:CW|:FIX]</td><td>+1900.000 MHz</td><td>Frequency setting</td></tr>
<tr><td colspan="3">Sensor measurement mode</td></tr>
<tr><td>[SENS[1]]|SENS2:DET:FUNC</td><td>NORM</td><td>Measurement mode</td></tr>
<tr><td colspan="3">Sensor video bandwidth setup</td></tr>
<tr><td>[SENS[1]]|SENS2:BAND|BWID:VID</td><td>E9321A/25A: DEF<br>E9322A/26A: DEF<br>E9323A/27A: DEF<br>N1921/2A: High</td><td>Sensor video bandwidth</td></tr>
<tr><td colspan="3">Gate Setup</td></tr>
<tr><td>[SENS[1]]|SENS2:SWE[1]|2|3|4 :OFF:TIME</td><td>Gate 1: 0 s<br>Gates 2 - 4: 0</td><td>Delay between trigger point and time gated period.</td></tr>
<tr><td>[SENS[1]]|SENS2:SWE[1]|2|3|4 :TIME</td><td>Gate 1: 1 ms<br>Gates 2 - 4: 0</td><td>Length of time gated period for time gated measurements.</td></tr>
<tr><td colspan="3">Trigger setup</td></tr>
<tr><td>TRIG[:SEQ[1]|2]:SOUR</td><td>INT1</td><td rowspan="2">Trigger source set up and acquisition mode continuous triggering</td></tr>
<tr><td>INIT:CONT</td><td>ON</td></tr>
<tr><td>TRIG[:SEQ]:LEV:AUTO</td><td>ON</td><td>Enable automatic setting of the trigger level</td></tr>
<tr><td>TRIG[:SEQ]:LEV</td><td>AUTO</td><td>Automatic Power level</td></tr>
<tr><td>TRIG[:SEQ]:SLOP</td><td>POS</td><td>Trigger event recognized on the rising edge of a signal</td></tr>
<tr><td>TRIG[:SEQ]:DEL</td><td>0 s</td><td>Delay between recognition of trigger event and start of a measurement</td></tr>
</table>

**Table 11-20 MPCA: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| `TRIG[:SEQ]:HOLD` | **MIN** | Trigger holdoff |
| *Step detection* | | |
| `[SENSe[1]]\|SENS2:AVER:SDET` | **0** | Step detection disabled |

**Table 11-21 MPCA: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | No Sensor | Non E9320 Sensor |
| *Display setup* | | |
| Upper window | UU single numeric | UU single numeric |
| Lower window | Dual numeric | See Table 11-22 |
| *Window / measurement setup* | | |
| *Upper window / upper measurement (UU)* | | |
| Feed | Gate 1 channel A | Gate 1 primary channel[a] |
| Measurement | Avg | Avg |
| *Upper window / lower measurement (UL)* | | |
| Feed | DEF | DEF |
| Measurement | DEF | DEF |
| *Lower window / upper measurement (LU)* | | |
| Feed | Gate 1 Channel A | Gate 1 primary channel[a] |
| Measurement | Peak | Peak |
| *Lower window / lower measurement (LL)* | | |
| Feed | Gate 1 Channel A | See Table 11-22 |
| Measurement | Peak to Avg | See Table 11-22 |

a. For further information refer to “Primary and secondary channels” on page 368.

**Table 11-22 MCPA: Power Meter Presets For Secondary Channel Sensors**

| **Function** | **Secondary Channel Sensor** | | |
|---|---|---|---|
| | **No Sensor** | **Non P-series or E9320 Sensor** | **P-series and E9320 Sensor** |
| *Display setup* | | | |
| Upper window | UU single numeric | Dual numeric | Dual numeric |
| Lower window | Dual numeric | Dual numeric | Dual numeric |
| *Upper window/lower measurement (UL)* | | | |
| Feed | DEF | Gate 1 primary channel[a] | Gate 1 primary channel[a] (channel A) |
| Measurement | DEF | Peak | Peak |
| *Lower window/upper measurement (LU)* | | | |
| Feed | Gate 1 primary channel[a] | Gate 1 primary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Peak | Pk-to-Avg | Avg |
| *Lower window/lower measurement (LL)* | | | |
| Feed | Gate 1 primary channel[a] | Secondary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Pk-to-Avg | Avg | Pk-to-Avg |

a. For further information refer to “Primary and secondary channels” on page 368.

## RADAR

The following table shows the power meter presets when <character_data> is set to `RADar`. Commands not listed are preset according to their `DEFault` values (for further information refer to Table 11-1).

**Table 11-23 RADAR: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| *Frequency* | | |
| `[SENS[1]]\|SENS2:FREQ[:CW\|:FIX]` | **+10.000 GHz** | Frequency setting |
| *Sensor measurement mode* | | |
| `[SENS[1]]\|SENS2:DET:FUNC` | **NORM** | Measurement mode |
| *Sensor video bandwidth setup* | | |
| `[SENS[1]]\|SENS2:BAND\|BWID:VID` | **E9321A/25A: DEF**<br>**E9322A/26A: DEF**<br>**E9323A/27A: OFF**<br>**N1921/2A: OFF** | Sensor video bandwidth |
| *Gate Setup* | | |
| `[SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :OFF:TIME` | **Gate 1: 0**<br>**Gate 2: 0**<br>**Gate 3: 750 ns**<br>**Gate 4: 0** | Delay between trigger point and time gated period. |
| `[SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :TIME` | **Gate 1: 1.0 us**<br>**Gate 2: 250 ns**<br>**Gate 3: 250 ns**<br>**Gate 4: 0** | Length of time gated period for time gated measurements. |
| *Trigger setup* | | |
| `TRIG[:SEQ[1]\|2]:SOUR` | **INT1** | Trigger source set up and acquisition mode continuous triggering |
| `INIT:CONT` | **ON** | |
| `TRIG[:SEQ]:LEV:AUTO` | **OFF** | Disable automatic setting of the trigger level |
| `TRIG[:SEQ]:LEV` | **-20 dBm** | Power level |
| `TRIG[:SEQ]:SLOP` | **POS** | Trigger event recognized on the rising edge of a signal |

**Table 11-23 RADAR: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| TRIG[:SEQ]:DEL | **0 s** | Delay between recognition of trigger event and start of a measurement |
| TRIG[:SEQ]:HOLD | **MIN** | Trigger holdoff |
| *Range* | | |
| [SENS[1]]\|SENS2:POW:AC:RANG:AUTO | **OFF** | Auto range off |
| [SENS[1]]\|SENS2:POW:AC:RANG | **UPPER** | Range set to upper |
| *Step detection* | | |
| [SENS[1]]\|SENS2:AVER:SDET | **0** | Step detection disabled |
| *Trace setup* | | |
| [SENS[1]]\|SENS2:TRAC:OFFS :TIME <numeric_value> | **-250 us** | Delay between delayed trigger point and the start of the trace |
| [SENS[1]]\|SENS2:TRAC:TIME <numeric_value> | **1.5 us** | Length of the trace |

**Table 11-24 RADAR: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | **No Sensor** | **Non E9320 Sensor** |
| *Display setup* | | |
| Upper window | Channel A trace | See Table 11-25 |
| Lower window | Dual numeric | Dual numeric |
| *Window / measurement setup* | | |
| *Upper window / upper measurement (UU)* | | |
| Feed | Gate 1 Channel A | See Table 11-25 |
| Measurement | Pk-to-Avg | See Table 11-25 |
| *Upper window / lower measurement (UL)* | | |
| Feed 1 | Gate 2 Channel A - Avg | See Table 11-25 |

**Table 11-24 RADAR: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | No Sensor | Non E9320 Sensor |
| Feed 2 | | See Table 11-25 |
| Measurement | Feed 1/ Feed 2 | See Table 11-25 |
| *Lower window / upper measurement (LU)* | | |
| Feed | Gate 1 Channel A | See Table 11-25 |
| Measurement | Peak | See Table 11-25 |
| *Lower window / lower measurement (LL)* | | |
| Feed | Gate 1 Channel A | See Table 11-25 |
| Measurement | Avg | See Table 11-25 |

**Table 11-25 RADAR: Power Meter Presets For Secondary Channel Sensors**

| Function | Secondary Channel Sensor | | |
|---|---|---|---|
| | No Sensor | Non P-series or E9320 Sensor | P-series and E9320 Sensor |
| *Display setup* | | | |
| Upper window | Primary Channel Trace | Dual numeric | Dual numeric |
| Lower window | Dual numeric | Dual numeric | Dual numeric |
| *Upper window / upper measurement (UU)* | | | |
| Feed | Gate 1 primary channel[a] | Gate 1 primary channel[a] | Gate 1 primary channel[a] (channel A) |
| Measurement | Pk-to-Avg | Peak | Peak |
| *Upper window / lower measurement (UL)* | | | |
| Feed 1 | Gate 2 primary channel[a] | Gate 1 primary channel[a] | Gate1 secondary channel[a] (channel B) |
| Feed 2 | | | |
| Measurement | Avg | Peak | Avg |
| *Lower window / upper measurement (LU)* | | | |

**Table 11-25 RADAR: Power Meter Presets For Secondary Channel Sensors**

| **Function** | **Secondary Channel Sensor** | | |
|---|---|---|---|
| | **No Sensor** | **Non P-series or E9320 Sensor** | **P-series and E9320 Sensor** |
| Feed 1 | Gate 1 primary channel[a] | Gate 2 primary channel[a] | Gate1 secondary channel[a] (channel B) |
| Feed 2 | | | |
| Measurement | Peak | Avg | Peak |
| *Lower window/lower measurement (LL)* | | | |
| Feed | Gate 1 primary channel[a] | Secondary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Avg | Avg | Avg |

a. For further information refer to “Primary and secondary channels” on page 368.

### 802.11a and HiperLan2

The following table shows the power meter presets when <character_data> is set to 802DOT11A and HIPERLAN2. Commands not listed are preset according to their DEFault values (for further information refer to Table 11-1).

**Table 11-26 802.11a and HiperLan2: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| *Frequency* | | |
| [SENS[1]]\|SENS2:FREQ[:CW\|:FIX] | **+5200.000 MHz** | Frequency setting |
| *Sensor measurement mode* | | |
| [SENS[1]]\|SENS2:DET:FUNC | **NORM** | Measurement mode |
| *Sensor video bandwidth setup* | | |
| [SENS[1]]\|SENS2:BAND\|BWID:VID | **E9321A/25A: DEF**<br>**E9322A/26A: DEF**<br>**E9323A/27A: High**<br>**N1921/2A: High** | Sensor video bandwidth |
| *Gate Setup* | | |
| [SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :OFF:TIME | **Gate 1: 0**<br>**Gates 2 - 4: 0** | Delay between trigger point and time gated period. |
| [SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :TIME | **Gate 1: 25 us**<br>**Gates 2 - 4: 0** | Length of time gated period for time gated measurements. |
| *Trigger setup* | | |
| TRIG[:SEQ[1]\|2]:SOUR | **INT1** | Trigger source set up and acquisition mode continuous triggering |
| INIT:CONT | **ON** | |
| TRIG[:SEQ]:LEV:AUTO | **ON** | Enable automatic setting of the trigger level |
| TRIG[:SEQ]:LEV | **AUTO** | Power level |
| TRIG[:SEQ]:SLOP | **POS** | Trigger event recognized on the rising edge of a signal |
| TRIG[:SEQ]:DEL | **0 s** | Delay between recognition of trigger event and start of a measurement |

**Table 11-26 802.11a and HiperLan2: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| TRIG[:SEQ]:HOLD | **MIN** | Trigger holdoff |
| TRIG[:SEQ]:HYST | **3 dB** | Hysteresis |
| *Step detection* | | |
| [SENSe[1]]\|SENS2:AVER:SDET | **0** | Step detection disabled |

**Table 11-27 802.11a and HiperLan2: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | **No Sensor** | **Non E9320 Sensor** |
| *Display setup* | | |
| Upper window | UU Single Numeric | UU Single Numeric |
| Lower window | Dual numeric | See Table 11-28 |
| *Window / measurement setup* | | |
| *Upper window / upper measurement (UU)* | | |
| Feed | Gate 1 Channel A | Gate 1 primary channel[a] |
| Measurement | Avg | Avg |
| *Upper window / lower measurement (UL)* | | |
| Feed | DEF | DEF |
| Measurement | DEF | DEF |
| *Lower window / upper measurement (LU)* | | |
| Feed | Gate 1 Channel A | Gate 1 primary channel[a] |
| Measurement | Peak | Peak |
| *Lower window / lower measurement (LL)* | | |
| Feed | Gate 1 Channel A | See Table 11-28 |
| Measurement | Pk-to-Avg | See Table 11-28 |

a. For further information refer to “Primary and secondary channels” on page 368.

**Table 11-28 802.11a and HiperLan2: Power Meter Presets For Secondary Channel Sensors**

| Function | Secondary Channel Sensor | | |
|---|---|---|---|
| | **No Sensor** | **Non P-series or E9320 Sensor** | **P-series and E9320 Sensor** |
| *Display setup* | | | |
| Upper window | UU Single numeric | Dual numeric | Dual numeric |
| Lower window | Dual numeric | Dual numeric | Dual numeric |
| *Upper window / lower measurement (UL)* | | | |
| Feed | DEF | Gate 1 primary channel[a] | Gate1 primary channel[a] (channel A) |
| Measurement | DEF | Peak | Peak |
| *Lower window / upper measurement (LU)* | | | |
| Feed | Gate 1 primary channel[a] | Gate 1 primary channel[a] | Gate 1 secondary channel[a] (channel B) |
| Measurement | Peak | Pk-to-Avg | Avg |
| *Lower window / lower measurement (LL)* | | | |
| Feed | Gate 1 primary channel[a] | Secondary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Pk-to-Avg | Avg | Pk-to-Avg |

a. For further information refer to “Primary and secondary channels” on page 368.

### 892.11b/g

The following table shows the power meter presets when <character_data> is set to `802DOT11B`. Commands not listed are preset according to their `DEFault` values (for further information refer to Table 11-1).

**Table 11-29 802.11b/g: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| *Frequency* | | |
| `[SENS[1]]\|SENS2:FREQ[:CW\|:FIX]` | **+2.400 GHz** | Frequency setting |
| *Sensor measurement mode* | | |
| `[SENS[1]]\|SENS2:DET:FUNC` | **NORM** | Measurement mode |
| *Sensor video bandwidth setup* | | |
| `[SENS[1]]\|SENS2:BAND\|BWID:VID` | **E9321A/25A: DEF**<br>**E9322A/26A: DEF**<br>**E9323A/27A: High**<br>**N1921/2A: High** | Sensor video bandwidth |
| *Gate Setup* | | |
| `[SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :OFF:TIME` | **Gate 1: 0**<br>**Gates 2 - 4: 0** | Delay between trigger point and time gated period. |
| `[SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :TIME` | **Gate 1: 100 us**<br>**Gates 2 - 4: 0** | Length of time gated period for time gated measurements. |
| *Trigger setup* | | |
| `TRIG[:SEQ[1]\|2]:SOUR` | **INT1** | Trigger source set up and acquisition mode continuous triggering |
| `INIT:CONT` | **ON** | |
| `TRIG[:SEQ]:LEV:AUTO` | **ON** | Enable automatic setting of the trigger level |
| `TRIG[:SEQ]:LEV` | **AUTO** | Power level |
| `TRIG[:SEQ]:SLOP` | **POS** | Trigger event recognized on the rising edge of a signal |
| `TRIG[:SEQ]:DEL` | **0 s** | Delay between recognition of trigger event and start of a measurement |

**Table 11-29 802.11b/g: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| `TRIG[:SEQ]:HOLD` | **MIN** | Trigger holdoff |
| *Step detection* | | |
| `[SENSe[1]]|SENS2:AVER:SDET` | **0** | Step detection disabled |

**Table 11-30 802.11b/g: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | No Sensor | Non E9320 Sensor |
| *Display setup* | | |
| Upper window | UU Single Numeric | UU Single Numeric |
| Lower window | Dual numeric | See Table 11-31 |
| *Window / measurement setup* | | |
| *Upper window / upper measurement (UU)* | | |
| Feed | Gate 1 Channel A | Gate 1 primary channel[a] |
| Measurement | Avg | Avg |
| *Upper window / lower measurement (UL)* | | |
| Feed | DEF | DEF |
| Measurement | DEF | DEF |
| *Lower window / upper measurement (LU)* | | |
| Feed | Gate 1 Channel A | Gate 1 primary channel[a] |
| Measurement | Peak | Peak |
| *Lower window / lower measurement (LL)* | | |
| Feed | Gate 1 Channel A | See Table 11-31 |
| Measurement | Pk-to-Avg | See Table 11-31 |

a. For further information refer to “Primary and secondary channels” on page 368.

**Table 11-31 802.11b/g: Power Meter Presets For Secondary Channel Sensors**

| Function | Secondary Channel Sensor | | |
|---|---|---|---|
| | **No Sensor** | **Non P-series or E9320 Sensor** | **P-series and E9320 Sensor** |
| *Display setup* | | | |
| Upper window | UU Single numeric | Dual numeric | Dual numeric |
| Lower window | Dual numeric | Dual numeric | Dual numeric |
| *Upper window / lower measurement (UL)* | | | |
| Feed | DEF | Gate 1 primary channel[a] | Gate1 primary channel[a] (channel A) |
| Measurement | DEF | Peak | Peak |
| *Lower window / upper measurement (LU)* | | | |
| Feed | Gate 1 primary channel[a] | Gate 1 primary channel[a] | Gate 1 secondary channel[a] (channel B) |
| Measurement | Peak | Pk-to-Avg | Avg |
| *Lower window / lower measurement (LL)* | | | |
| Feed | Gate 1 primary channel[a] | Secondary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Pk-to-Avg | Avg | Pk-to-Avg |

a. For further information refer to “Primary and secondary channels” on page 368.

### 1xeV-DO

The following table shows the power meter presets when <character_data> is set to XEVDO. Commands not listed are preset according to their DEFault values (for further information refer to Table 11-1).

**Table 11-32 1xeV-DO: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| *Frequency* | | |
| [SENS[1]]\|SENS2:FREQ[:CW\|:FIX] | **+1900.000 MHz** | Frequency setting |
| *Sensor measurement mode* | | |
| [SENS[1]]\|SENS2:DET:FUNC | **NORM** | Measurement mode |
| *Sensor video bandwidth setup* | | |
| [SENS[1]]\|SENS2:BAND\|BWID:VID | **E9321A/25A: DEF**<br>**E9322A/26A: DEF**<br>**E9323A/27A: High**<br>**N1921/2A: LOW** | Sensor video bandwidth |
| *Gate Setup* | | |
| [SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :OFF:TIME | **Gate 1: 10 us**<br>**Gates 2 - 4: 0** | Delay between trigger point and time gated period. |
| [SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :TIME | **Gate 1: 810 us**<br>**Gates 2 - 4: 0** | Length of time gated period for time gated measurements. |
| *Trigger setup* | | |
| TRIG[:SEQ[1]\|2]:SOUR | **INT1** | Trigger source set up and acquisition mode continuous triggering |
| INIT:CONT | **ON** | |
| TRIG[:SEQ]:LEV:AUTO | **ON** | Enable automatic setting of the trigger level |
| TRIG[:SEQ]:LEV | **AUTO** | Power level |
| TRIG[:SEQ]:SLOP | **POS** | Trigger event recognized on the rising edge of a signal |
| TRIG[:SEQ]:DEL | **0 s** | Delay between recognition of trigger event and start of a measurement |

**Table 11-32 1xeV-DO: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| TRIG[:SEQ]:HOLD | **1 ms** | Trigger holdoff |
| *Step detection* | | |
| [SENSe[1]]\|SENS2:AVER:SDET | **0** | Step detection disabled |
| *Trace setup* | | |
| [SENS[1]]\|SENS2:TRAC:OFFS :TIME <numeric_value> | **-40 us** | Delay between delayed trigger point and the start of the trace |
| [SENS[1]]\|SENS2:TRAC:TIME <numeric_value> | **1 ms** | Length of the trace |

**Table 11-33 1xeV-DO: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | **No Sensor** | **Non E9320 Sensor** |
| *Display setup* | | |
| Upper window | Channel A trace | Primary channel[a] trace |
| Lower window | Dual numeric | See Table 11-34 |
| *Window / measurement setup* | | |
| *Upper window / upper measurement (UU)* | | |
| Feed | N/A | N/A |
| Measurement | N/A | N/A |
| *Upper window / lower measurement (UL)* | | |
| Feed | N/A | N/A |
| Measurement | N/A | N/A |
| *Lower window / upper measurement (LU)* | | |
| Feed | Gate 1 Channel A | Gate 1 primary channel[a] |
| Measurement | Avg | Avg |
| *Lower window / lower measurement (LL)* | | |
| Feed | Gate 1 Channel A | See Table 11-34 |

**Table 11-33 1xeV-DO: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | No Sensor | Non E9320 Sensor |
| Measurement | Pk-to-Avg | See Table 11-34 |

a. For further information refer to "Primary and secondary channels" on page 368.

**Table 11-34 1xeV-DO: Power Meter Presets For Secondary Channel Sensors**

| Function | Secondary Channel Sensor | | |
|---|---|---|---|
| | No Sensor | Non P-series or E9320 Sensor | P-series and E9320 Sensor |
| *Display setup* | | | |
| Lower window | Dual numeric | Dual numeric | Dual numeric |
| *Lower window/lower measurement (LL)* | | | |
| Feed | Gate 1 primary channel[a] | Secondary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Pk-to-Avg | Avg | Avg |

a. For further information refer to "Primary and secondary channels" on page 368.

### 1xeV-DV

The following table shows the power meter presets when <character_data> is set to XEVDV. Commands not listed are preset according to their DEFault values (for further information refer to Table 11-1).

**Table 11-35 1xeV-DV: Power Meter Presets**

<table>
<tr><th>Command</th><th>Setting</th><th>Comments</th></tr>
<tr><td colspan="3">Frequency</td></tr>
<tr><td>[SENS[1]]|SENS2:FREQ[:CW|:FIX]</td><td>+1900.000 MHz</td><td>Frequency setting</td></tr>
<tr><td colspan="3">Sensor measurement mode</td></tr>
<tr><td>[SENS[1]]|SENS2:DET:FUNC</td><td>NORM</td><td>Measurement mode</td></tr>
<tr><td colspan="3">Sensor video bandwidth setup</td></tr>
<tr><td>[SENS[1]]|SENS2:BAND|BWID:VID</td><td>E9321A/25A: DEF<br>E9322A/26A: DEF<br>E9323A/27A: High<br>N1921/2A: LOW</td><td>Sensor video bandwidth</td></tr>
<tr><td colspan="3">Gate Setup</td></tr>
<tr><td>[SENS[1]]|SENS2:SWE[1]|2|3|4<br>:OFF:TIME</td><td>Gate 1: 10 us<br>Gates 2 - 4: 0</td><td>Delay between trigger point and time gated period.</td></tr>
<tr><td>[SENS[1]]|SENS2:SWE[1]|2|3|4<br>:TIME</td><td>Gate 1: 810 us<br>Gates 2 - 4: 0</td><td>Length of time gated period for time gated measurements.</td></tr>
<tr><td colspan="3">Trigger setup</td></tr>
<tr><td>TRIG[:SEQ[1]|2]:SOUR</td><td>INT1</td><td rowspan="2">Trigger source set up and acquisition mode continuous triggering</td></tr>
<tr><td>INIT:CONT</td><td>ON</td></tr>
<tr><td>TRIG[:SEQ]:LEV:AUTO</td><td>ON</td><td>Enable automatic setting of the trigger level</td></tr>
<tr><td>TRIG[:SEQ]:LEV</td><td>AUTO</td><td>Power level</td></tr>
<tr><td>TRIG[:SEQ]:SLOP</td><td>POS</td><td>Trigger event recognized on the rising edge of a signal</td></tr>
<tr><td>TRIG[:SEQ]:DEL</td><td>0 s</td><td>Delay between recognition of trigger event and start of a measurement</td></tr>
</table>

**Table 11-35 1xeV-DV: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| TRIG[:SEQ]:HOLD | **1 ms** | Trigger holdoff |
| *Step detection* | | |
| [SENSe[1]]\|SENS2:AVER:SDET | **0** | Step detection disabled |
| *Trace setup* | | |
| [SENS[1]]\|SENS2:TRAC:OFFS :TIME <numeric_value> | **-40 us** | Delay between delayed trigger point and the start of the trace |
| [SENS[1]]\|SENS2:TRAC:TIME <numeric_value> | **1 ms** | Length of the trace |

**Table 11-36 1xeV-DV: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | No Sensor | Non E9320 Sensor |
| *Display setup* | | |
| Upper window | Channel A trace | Primary channel[a] trace |
| Lower window | Dual numeric | See Table 11-37 |
| *Window / measurement setup* | | |
| *Upper window / upper measurement (UU)* | | |
| Feed | N/A | N/A |
| Measurement | N/A | N/A |
| *Upper window / lower measurement (UL)* | | |
| Feed | N/A | N/A |
| Measurement | N/A | N/A |
| *Lower window / upper measurement (LU)* | | |
| Feed | Gate 1 Channel A | Gate 1 primary channel[a] |
| Measurement | Avg | Avg |
| *Lower window / lower measurement (LL)* | | |
| Feed | Gate 1 Channel A | See Table 11-37 |

**Table 11-36 1xeV-DV: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | No Sensor | Non E9320 Sensor |
| Measurement | Pk-to-Avg | See Table 11-37 |

a. For further information refer to “Primary and secondary channels” on page 368.

**Table 11-37 1xeV-DV: Power Meter Presets For Secondary Channel Sensors**

| Function | Secondary Channel Sensor | | |
|---|---|---|---|
| | No Sensor | Non P-series or E9320 Sensor | P-series and E9320 Sensor |
| *Display setup* | | | |
| Lower window | Dual numeric | Dual numeric | Dual numeric |
| *Lower window/lower measurement (LL)* | | | |
| Feed | Gate 1 primary channel[a] | Secondary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Pk-to-Avg | Avg | Avg |

a. For further information refer to “Primary and secondary channels” on page 368.

### TD-SCDMA

The following table shows the power meter presets when <character_data> is set to `TDSCdma`. Commands not listed are preset according to their `DEFault` values (for further information refer to Table 11-1).

**Table 11-38 TD-SCDMA: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| *Frequency* | | |
| `[SENS[1]]|SENS2:FREQ[:CW|:FIX]` | **+1900.000 MHz** | Frequency setting |
| *Sensor measurement mode* | | |
| `[SENS[1]]|SENS2:DET:FUNC` | **NORM** | Measurement mode |

**Table 11-38 TD-SCDMA: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| *Sensor video bandwidth setup* | | |
| `[SENS[1]]\|SENS2:BAND\|BWID:VID` | **E9321A/25A: DEF**<br>**E9322A/26A: DEF**<br>**E9323A/27A: High**<br>**N1921/2A: LOW** | Sensor video bandwidth |
| *Gate Setup* | | |
| `[SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :OFF:TIME` | **Gate 1: 10 us**<br>**Gates 2 - 4: 0** | Delay between trigger point and time gated period. |
| `[SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :TIME` | **Gate 1: 810 us**<br>**Gates 2 - 4: 0** | Length of time gated period for time gated measurements. |
| *Trigger setup* | | |
| `TRIG[:SEQ[1]\|2]:SOUR` | **INT1** | Trigger source set up and acquisition mode continuous triggering |
| `INIT:CONT` | **ON** | |
| `TRIG[:SEQ]:LEV:AUTO` | **ON** | Enable automatic setting of the trigger level |
| `TRIG[:SEQ]:LEV` | **AUTO** | Power level |
| `TRIG[:SEQ]:SLOP` | **POS** | Trigger event recognized on the rising edge of a signal |
| `TRIG[:SEQ]:DEL` | **0 s** | Delay between recognition of trigger event and start of a measurement |
| `TRIG[:SEQ]:HOLD` | **1 ms** | Trigger holdoff |
| *Step detection* | | |
| `[SENSe[1]]\|SENS2:AVER:SDET` | **0** | Step detection disabled |
| *Trace setup* | | |
| `[SENS[1]]\|SENS2:TRAC:OFFS :TIME <numeric_value>` | **-40 us** | Delay between delayed trigger point and the start of the trace |
| `[SENS[1]]\|SENS2:TRAC:TIME <numeric_value>` | **1 ms** | Length of the trace |

**Table 11-39 TD-SCDMA: Power Meter Presets: Window/Measurement Settings**

| **Function** | **Setting** | |
|---|---|---|
| | **No Sensor** | **Non E9320 Sensor** |
| *Display setup* | | |
| Upper window | Channel A trace | Primary channel[a] trace |
| Lower window | Dual numeric | See Table 11-40 |
| *Window / measurement setup* | | |
| *Upper window / upper measurement (UU)* | | |
| Feed | N/A | N/A |
| Measurement | N/A | N/A |
| *Upper window / lower measurement (UL)* | | |
| Feed | N/A | N/A |
| Measurement | N/A | N/A |
| *Lower window / upper measurement (LU)* | | |
| Feed | Gate 1 Channel A | Gate 1 primary channel[a] |
| Measurement | Avg | Avg |
| *Lower window / lower measurement (LL)* | | |
| Feed | Gate 1 Channel A | See Table 11-40 |
| Measurement | Pk-to-Avg | See Table 11-40 |

a. For further information refer to "Primary and secondary channels" on page 368.

**Table 11-40 TD-SCDMA: Power Meter Presets For Secondary Channel Sensors**

| **Function** | **Secondary Channel Sensor** | | |
|---|---|---|---|
| | **No Sensor** | **Non P-series or E9320 Sensor** | **P-series and E9320 Sensor** |
| *Display setup* | | | |
| Lower window | Dual numeric | Dual numeric | Dual numeric |
| *Lower window / lower measurement (LL)* | | | |

**Table 11-40 TD-SCDMA: Power Meter Presets For Secondary Channel Sensors**

| **Function** | **Secondary Channel Sensor** | | |
|---|---|---|---|
| | **No Sensor** | **Non P-series or E9320 Sensor** | **P-series and E9320 Sensor** |
| Feed | Gate 1 primary channel[a] | Secondary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Pk-to-Avg | Avg | Avg |

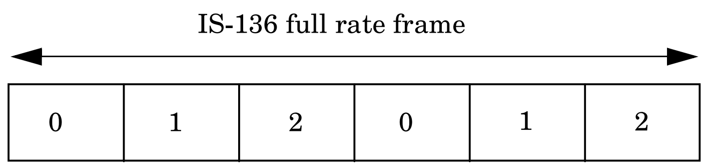
a. For further information refer to “Primary and secondary channels” on page 368.

### NADC

The NADC set-up provides:

- Average power measurement of both active timeslots in NADC or IS-136 “full rate” transmission. This assumes that there are two timeslots to be measured in each frame as for example with timeslots 0 in the following diagram:

**Figure 11-2** **A Trace Display Of The Active Timeslots**




- A trace display of the active timeslots.

The measurement is started by detecting the RF burst—for example the burst emitted by a mobile—using the internal RF level trigger. The internal level trigger is set to –20dBm. Time-gating is used to measure the average power in two active timeslots which are separated by two inactive timeslots

The following table shows the power meter presets when <character_data> is set to `NADC`. Commands not listed are preset according to their `DEFault` values (for further information refer to Table 11-1):

**Table 11-41** **NADC: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| *Frequency* | | |
| `[SENS[1]]|SENS2:FREQ[:CW|:FIX]` | **+800.000 MHz** | Frequency setting |
| *Sensor measurement mode* | | |
| `[SENS[1]]|SENS2:DET:FUNC` | **NORM** | Measurement mode |
| *Sensor video bandwidth setup* | | |

**Table 11-41 NADC: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| [SENS[1]]\|SENS2:BAND\|BWID:VID | E9321A/25A: OFF<br>E9322A/26A: OFF<br>E9323A/27A: OFF<br>N1921/2A: OFF | Sensor video bandwidth |
| *Gate Setup* | | |
| [SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :OFF:TIME | Gate 1: 123.5 us<br>Gate 2: 20.123 ms<br>Gates 3 - 4: 0 | Delay between trigger point and time gated period. |
| [SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :TIME | Gate 1: 6.46 ms<br>Gate 2: 6.46 ms<br>Gates 3 - 4: 0 | Length of time gated period for time gated measurements. |
| *Trigger setup* | | |
| TRIG[:SEQ[1]\|2]:SOUR | INT1 | Trigger source set up and acquisition mode continuous triggering |
| INIT:CONT | ON | |
| TRIG[:SEQ]:LEV:AUTO | OFF | Disable automatic setting of the trigger level |
| TRIG[:SEQ]:LEV | -20 dBm | Power level |
| TRIG[:SEQ]:SLOP | POS | Trigger event recognized on the rising edge of a signal |
| TRIG[:SEQ]:DEL | 0 s | Delay between recognition of trigger event and start of a measurement |
| TRIG[:SEQ]:HOLD | 30 ms | Trigger holdoff |
| *Step detection* | | |
| [SENSe[1]]\|SENS2:AVER:SDET | 1 | Step detection enabled |
| *Trace setup* | | |
| DISP[:WIND[1]\|2]:TRAC:LOW | +20 dBm | Maximum power |
| DISP[:WIND[1]\|2]:TRAC:UPP | -35 dBm | Minimum power |
| [SENS[1]]\|SENS2:TRAC:OFFS :TIME <numeric_value> | -0.2 ms | Delay between delayed trigger point and the start of the trace |

**Table 11-41 NADC: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| [SENS[1]]\|SENS2:TRAC:TIME <numeric_value> | **28 ms** | Length of the trace |

**Table 11-42 NADC: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | No Sensor | Non E9320 Sensor |
| *Display setup* | | |
| Upper window | Channel A trace | Primary channel[a] trace |
| Lower window | Dual numeric | See Table 11-43 |
| *Window / measurement setup* | | |
| *Upper window / upper measurement (UU)* | | |
| Feed | N/A | N/A |
| Measurement | N/A | N/A |
| *Upper window / lower measurement (UL)* | | |
| Feed | N/A | N/A |
| Measurement | N/A | N/A |
| *Lower window / upper measurement (LU)* | | |
| Feed | Gate 1 Channel A | Gate 1 primary channel[a] |
| Measurement | Avg | Avg |
| *Lower window / lower measurement (LL)* | | |
| Feed | Gate 2 Channel A | See Table 11-43 |
| Measurement | Avg | See Table 11-43 |

a. For further information refer to “Primary and secondary channels” on page 368.

**Table 11-43 NADC: Power Meter Presets For Secondary Channel Sensors**

| **Function** | **Secondary Channel Sensor** | | |
|---|---|---|---|
| | **No Sensor** | **Non P-series or E9320 Sensor** | **P-series and E9320 Sensor** |
| *Display setup* | | | |
| Lower window | Dual numeric | Dual numeric | Dual numeric |
| *Lower window/lower measurement (LL)* | | | |
| Feed | Gate 2 primary channel[a] | Secondary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Avg | Avg | Avg |

a. For further information refer to “Primary and secondary channels” on page 368.

### IDEN

The iDEN set-up provides:

- Average power in one iDEN training and data pulse.
- Peak-to-average one iDEN training and data pulse.
- Average power in a 90ms iDEN frame.

The measurement is started by detecting the iDEN training burst—for example the burst emitted by a mobile—using the internal RF level trigger. Time gating is used to measure the average power in the following 15ms (data pulse). Gate 1 is used to measure this data pulse. The 90ms frame is also captured to measure the average power in the entire frame. Gate 2 is used to measure the 90ms frame.

The following table shows the power meter presets when <character_data> is set to `IDEN`. Commands not listed are preset according to their `DEFault` values (for further information refer to Table 11-1):

**Table 11-44 iDEN: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| *Frequency* | | |
| `[SENS[1]]\|SENS2:FREQ[:CW\|:FIX]` | **+800.000 MHz** | Frequency setting |
| *Sensor measurement mode* | | |
| `[SENS[1]]\|SENS2:DET:FUNC` | **NORM** | Measurement mode |
| *Sensor video bandwidth setup* | | |
| `[SENS[1]]\|SENS2:BAND\|BWID:VID` | **E9321A/25A: OFF**<br>**E9322A/26A: OFF**<br>**E9323A/27A: OFF**<br>**N1921/2A: OFF** | Sensor video bandwidth |
| *Gate Setup* | | |
| `[SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :OFF:TIME` | **Gate 1: 0 us**<br>**Gates 2 - 4: 0** | Delay between trigger point and time gated period. |
| `[SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :TIME` | **Gate 1: 15 ms**<br>**Gate 2: 90 ms**<br>**Gate 3: 160 us**<br>**Gate 4: 0** | Length of time gated period for time gated measurements. |
| *Trigger setup* | | |

**Table 11-44 iDEN: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| TRIG[:SEQ[1]\|2]:SOUR | **INT1** | Trigger source set up and acquisition mode continuous triggering |
| INIT:CONT | **ON** | |
| TRIG[:SEQ]:LEV:AUTO | **OFF** | Disable automatic setting of the trigger level |
| TRIG[:SEQ]:LEV | **-20** | Automatic Power level |
| TRIG[:SEQ]:SLOP | **POS** | Trigger event recognized on the rising edge of a signal |
| TRIG[:SEQ]:DEL | **0 s** | Delay between recognition of trigger event and start of a measurement |
| TRIG[:SEQ]:HOLD | **20 ms** | Trigger holdoff |
| *Averaging* | | |
| [SENSe[1]]\|SENSe2:AVER[:STATe] | **ON** | Averaging On |
| [SENSe[1]]\|SENSe2:AVER:COUN | **64** | Averaging set to 64 |
| *Step detection* | | |
| [SENSe[1]]\|SENS2:AVER:SDET | **1** | Step detection enabled |
| *Trace setup* | | |
| DISP[:WIND[1]\|2]:TRAC:LOW | **+20 dBm** | Maximum power |
| DISP[:WIND[1]\|2]:TRAC:UPP | **-30 dBm** | Minimum power |
| [SENS[1]]\|SENS2:TRAC:OFFS :TIME <numeric_value> | **0** | Delay between delayed trigger point and the start of the trace |
| [SENS[1]]\|SENS2:TRAC:TIME <numeric_value> | **100 ms** | Length of the trace |

**Table 11-45 iDEN: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | No Sensor | Non E9320 Sensor |
| *Display setup* | | |

**Table 11-45 iDEN: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | No Sensor | Non E9320 Sensor |
| Upper window | UU single numeric | See Table 11-46 |
| Lower window | Dual numeric | See Table 11-46 |
| *Window / measurement setup* | | |
| *Upper window / upper measurement (UU)* | | |
| Feed | Gate 1 channel A | Gate 1 primary channel[a] |
| Measurement | Avg | Avg |
| *Upper window / lower measurement (UL)* | | |
| Feed | DEF | See Table 11-46 |
| Measurement | DEF | See Table 11-46 |
| *Lower window / upper measurement (LU)* | | |
| Feed | Gate 1 Channel A | See Table 11-46 |
| Measurement | Peak | See Table 11-46 |
| *Lower window / lower measurement (LL)* | | |
| Feed | Gate 1 Channel A | See Table 11-46 |
| Measurement | Pk-to-Avg | See Table 11-46 |

a. For further information refer to “Primary and secondary channels” on page 368.

**Table 11-46 iDEN: Power Meter Presets For Secondary Channel Sensors**

| Function | Secondary Channel Sensor | | |
|---|---|---|---|
| | No Sensor | Non P-series or E9320 Sensor | P-series and E9320 Sensor |
| *Display setup* | | | |
| Upper window | UU single numeric | Dual numeric | Dual numeric |
| Lower window | Dual numeric | Dual numeric | Dual numeric |
| *Upper window / lower measurement (UL)* | | | |
| Feed | DEF | Gate 1 primary channel[a] | Gate 1 primary channel[a] (channel A) |

**Table 11-46 iDEN: Power Meter Presets For Secondary Channel Sensors**

| Function | Secondary Channel Sensor | | |
|---|---|---|---|
| | **No Sensor** | **Non P-series or E9320 Sensor** | **P-series and E9320 Sensor** |
| Measurement | DEF | Peak | Peak |
| *Lower window / upper measurement (LU)* | | | |
| Feed | Gate 1 primary channel[a] | Gate 1 primary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Peak | Pk-to-Avg | Avg |
| *Lower window / lower measurement (LL)* | | | |
| Feed | Gate 1 primary channel[a] | Secondary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Pk-to-Avg | Avg | Pk-to-Avg |

a. For further information refer to “Primary and secondary channels” on page 368.

### DVB

The following table shows the power meter presets when <character_data> is set to `DVB`. Commands not listed are preset according to their `DEFault` values (for further information refer to Table 11-1).

**Table 11-47 DVB: Power Meter Presets**

| **Command** | **Setting** | **Comments** |
|---|---|---|
| *Frequency* | | |
| [SENS[1]]\|SENS2:FREQ[:CW\|:FIX] | **+660.000 MHz** | Frequency setting |
| *Sensor measurement mode* | | |
| [SENS[1]]\|SENS2:DET:FUNC | **NORM** | Measurement mode |
| *Sensor video bandwidth setup* | | |
| [SENS[1]]\|SENS2:BAND\|BWID:VID | **E9321A/25A: DEF**<br>**E9322A/26A: DEF**<br>**E9323A/27A: DEF**<br>**N1921/2A: OFF** | Sensor video bandwidth |
| *Gate Setup* | | |
| [SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :OFF:TIME | **Gate 1: 10 us**<br>**Gate 2: 0**<br>**Gates 3 - 4: 0** | Delay between trigger point and time gated period. |
| [SENS[1]]\|SENS2:SWE[1]\|2\|3\|4 :TIME | **Gate 1: 15 ms**<br>**Gate 1: 90 ms**<br>**Gates 2 - 4: 0** | Length of time gated period for time gated measurements. |
| *Trigger setup* | | |
| TRIG[:SEQ[1]\|2]:SOUR | **INT1** | Trigger source set up and acquisition mode continuous triggering |
| INIT:CONT | **ON** | |
| TRIG[:SEQ]:LEV:AUTO | **NORM** | Enable automatic setting of the trigger level |
| TRIG[:SEQ]:LEV | **-20 dBm** | Power level |
| TRIG[:SEQ]:SLOP | **POS** | Trigger event recognized on the rising edge of a signal |

**Table 11-47** **DVB: Power Meter Presets**

| Command | Setting | Comments |
|---|---|---|
| TRIG[:SEQ]:DEL | **0 s** | Delay between recognition of trigger event and start of a measurement |
| TRIG[:SEQ]:HOLD | **0 s** | Trigger holdoff |
| *Step detection* | | |
| [SENSe[1]]\|SENS2:AVER:SDET | **1** | Step detection enabled |

**Table 11-48** **DVB: Power Meter Presets: Window/Measurement Settings**

| Function | Setting | |
|---|---|---|
| | No Sensor | Non E9320 Sensor |
| *Display setup* | | |
| Upper window | UU single numeric | See Table 11-49 |
| Lower window | Dual numeric | See Table 11-49 |
| *Window / measurement setup* | | |
| *Upper window / upper measurement (UU)* | | |
| Feed | Gate 1 channel A | Gate 1 primary channel[a] |
| Measurement | Avg | Avg |
| *Upper window / lower measurement (UL)* | | |
| Feed | DEF | See Table 11-49 |
| Measurement | DEF | See Table 11-49 |
| *Lower window / upper measurement (LU)* | | |
| Feed | Gate 1 Channel A | See Table 11-49 |
| Measurement | Pk-to-Avg | See Table 11-49 |
| *Lower window / lower measurement (LL)* | | |
| Feed | Gate 2 Channel A | See Table 11-49 |
| Measurement | Avg | See Table 11-49 |

a. For further information refer to “Primary and secondary channels” on page 368.

**Table 11-49 DVB: Power Meter Presets For Secondary Channel Sensors**

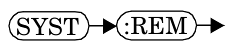
| **Function** | **Secondary Channel Sensor** | | |
|---|---|---|---|
| | **No Sensor** | **Non P-series or E9320 Sensor** | **P-series and E9320 Sensor** |
| *Display setup* | | | |
| Upper window | UU single numeric | Dual numeric | Dual numeric |
| Lower window | Dual numeric | Dual numeric | Dual numeric |
| *Upper window/lower measurement (UL)* | | | |
| Feed | DEF | Gate 1 primary channel[a] | Gate 1 primary channel[a] (channel A) |
| Measurement | DEF | Pk-to-Avg | Pk-to-Avg |
| *Lower window/upper measurement (LU)* | | | |
| Feed | Gate 1 primary channel[a] | Gate 1 primary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Pk-to-Avg | Avg | Avg |
| *Lower window/lower measurement (LL)* | | | |
| Feed | Gate 1 primary channel[a] | Secondary channel[a] | Gate1 secondary channel[a] (channel B) |
| Measurement | Avg | Avg | Pk-to-Avg |

a. For further information refer to “Primary and secondary channels” on page 368.

# SYSTem:REMote

This command locks the power meter front panel keypad excepting the Local key. The power meter display status reporting line shows “RMT”. Local front panel operation of the power meter is inhibited but can be enabled by pressing the Local key.

### Syntax




### Example

| | |
|---|---|
| `SYST:REM` | *This command locks the power meter front panel keypad excepting the Local key.* |

# SYSTem:RWLock

This command locks out the front panel keypad - including the front panel Local key. The power meter display status reporting line shows `"RMT"`. In this state the power meter cannot be returned to manual control from the front panel.

### Syntax

SYST → :RWL →

### Example

| | |
|---|---|
| `SYST:RWL` | *This command locks the power meter front panel keypad - including the* ***Local*** *key.* |

# SYSTem:VERSion?

This query returns the version of SCPI used in the power meter. The response is in the form of XXXX.Y, where XXXX is the year and Y is the version number.

### Syntax

SYST → :VERS → ?

### Example

| | |
|---|---|
| `SYST:VERS?` | *This command queries which version of SCPI is used in the power meter.* |

# 12 TRACe Subsystem

# TRACe Subsystem

The `TRACe` subsystem is used to:

- Specify the type of trace to be captured.
- Enable/disable trace capture.
- Specify the trace units.

There are two pre-defined TRACE blocks:

- `TRACe1`: associated with channel A.
- `TRACe2`: associated with channel B.

The following commands are described in this chapter:

.

| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
| TRACe[1]\|2 | | | |
| [:DATA]? | <character_data> | [query only] | page 436 |
| :MEASurement | | | |
| PULse | [1]\|...\|10 | | |
| DURation? | | [query only] | page 438 |
| PERiod? | | [query only] | page 439 |
| SEParation? | | [query only] | page 440 |
| TRANsition | [1]\|...\|10 | | |
| NEGative | | | |
| DURation? | | [query only] | page 441 |
| OCCurrence? | | [query only] | page 442 |
| POSitive | | | |
| DURation? | | [query only] | page 443 |
| OCCurrence? | | [query only] | page 444 |
| REFerence | | [query only] | page 445 |
| :STATe | <boolean> | | page 446 |
| :UNIT | <character_data> | | page 447 |

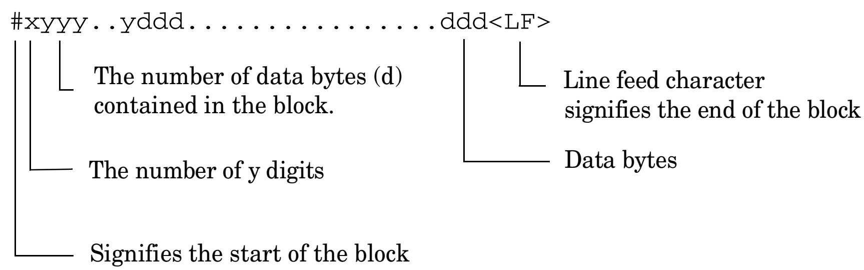
**NOTE** When making trace measurements, use the following command sequence to synchronize the returned trace data with the measurement:

| Command | Comment |
|---|---|
| INIT:CONT OFF | Trace data can only be retrieved with INIT:CONT OFF |
| TRAC:STAT ON | Enables trace capture |
| AVER:STAT OFF<br>or<br>TRIG:DEL:AUTO OFF | No settling time delays for digital filter to fill |
| INIT | Initiates a new measurement |
| FETCH? | Fetch the result (waits for the measurement to complete) |
| TRACE:DATA? MRES | Retrieves the trace data once the measurement has completed |

# TRACe[1]|2[:DATA]? <character_data>

This query returns trace data from the specified channel. The trace resolution is determined by <character_data>.

Data is returned in IEEE 488.2 arbitrary block program data format as follows:

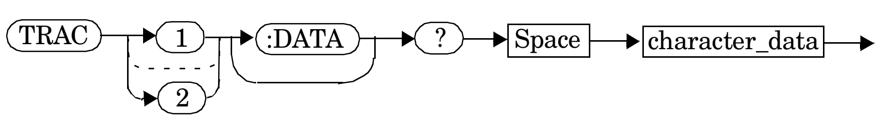



Example: if there are 12435 data bytes, y = 12435 and x = 5

Each point in the trace is represented as an IEEE 754 32 bit floating point number, made up of four bytes in the data block. The MS byte is transmitted first. Each complete block is terminated by a line feed.

**NOTE** `TRACe` data formatting is not affected by `FORMat` subsystem formatting.

### Syntax



### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | • `HRESolution`: high resolution. The complete capture buffer at the internal sample rate. The number of points in this trace is not fixed, as it is affected by the `SENS:TRACe:TIMe` setting. As a general principle, the power meter decimates[a] the data from 20Msamples/second to approximately 80,000 data points - from this, the effective rate is estimated at 12.5us/point. This, combined with the `SENS:TRACe:TIMe` setting, allows the number of `HRES` points to be estimated<br>• `MRESolution`: medium resolution. A subset of the capture buffer - the buffer contents are decimated[a] to 1000 data points.<br>• `LRESolution`: low resolution. A subset of the capture buffer - the buffer contents are decimated[a] to provide 234 data points. This is the same number of data points as the power meter uses to display the trace on the front panel. Hence, the `LRES` command can be used to replicate the power meter's display. | `HRES`<br>`MRES`<br>`LRES` |

a. This term **Decimate** is used to describe the way the power meter reduces the amount of data from 20Msamples/second by only reporting every 'nth' piece of data. Trace data that lies between each 'nth' point is discarded.

### Example

| | |
|---|---|
| `TRAC:DATA? HRES` | *This command returns the trace data for channel A at high resolution.* |

**Error Messages** If `TRAC:STAT` is off, the error -221, "Settings Conflict" occurs.

## TRACe[1]|2:MEASurement:PULse[1]|...|10:DURation?

This command returns the difference between the a pulse and next transition occurrence instants. As power pulses are by definition positive pulses, the pulse duration is the time difference between positive and negative transitions of one pulse.

**Algorithm**

If the first transition in the trace is positive,

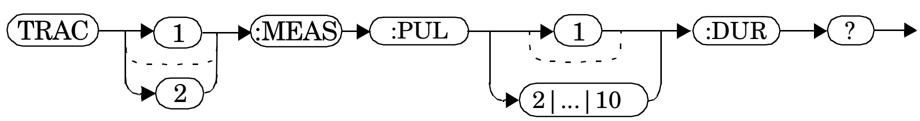
then

`PULse:DURation` = *time the first negative transition occurs* - *time the first positive transition occurs*

else

`PULse:DURation` = *time the second negative transition occurs* - *time the first positive transition occurs*.

**Syntax**




**Example**

| | |
|---|---|
| `TRAC2:MEAS:PUL3:DUR?` | *This command returns the duration of the 3rd pulse found on trace 2.* |

**Error Messages**

- If a P-series sensor is not connected, error -241, “Hardware missing” occurs.
- If a P-series sensor is connected and Free Run trigger acquisition is selected, error –221, “Settings conflict” occurs.

**NOTE** If you attempt to measure a pulse out of the range of the capture, for example, measure the 5th pulse and there are only 4 pulses displayed, the power meter returns #0##9.91E37 as the result.

# TRACe[1]|2:MEASurement:PULse[1]|...10:PERiod?

This command returns the pulse period. This is the time difference between two consecutive transition occurrences of the same polarity. The period is equal to the sum of the pulse separation and the pulse duration.

**Algorithm**

If the first transition in the trace is positive,

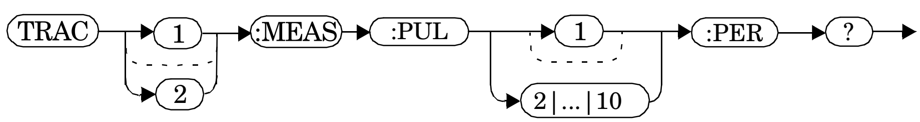
then

`PULse:PERiod` = *time the second positive transition occurrence - time the first positive transition occurs*

else

`PULse:PERiod` = *time the second negative transition occurs - time the first negative transition occurs*.

**Syntax**




**Example**

| | |
|---|---|
| `TRAC:MEAS:PUL:PER?` | *This command returns the period of the pulse found on trace 1.* |

**Error Messages**

- If a P-series sensor is not connected, error -241, “Hardware missing” occurs.
- If a P-series sensor is connected and Free Run trigger acquisition is selected, error –221, “Settings conflict” occurs.

**NOTE**

If you attempt to measure a pulse out of the range of the capture, for example, measure the 5th pulse and there are only 4 pulses displayed, the power meter returns #0##9.91E37 as the result.

# TRACe[1]|2:MEASurement:PULse[1]|...|10:SEParation?

This command returns the time difference of the $n^{th}$ and $(n+1)^{th}$ pulses found on a trace. As power pulses are by definition positive pulses, the pulse separation is the time difference between negative transition of one pulse and the positive transition of the next pulse.

**Algorithm**

If the first transition in the trace is positive,

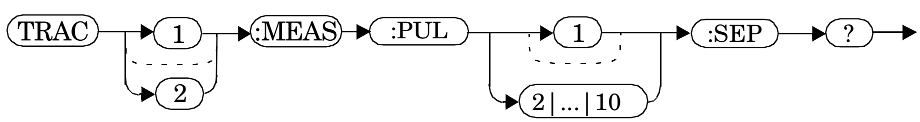
then

`PULse:SEParation` = *time the second positive transition occurs* - *time the first negative transition occurs*

else

`PULse:SEParation` = *time the first positive transition occurs* - *time the first negative transition occurs*.

**Syntax**




**Example**

| | |
|---|---|
| `TRAC1:MEAS:PUL:SEP?` | *This command returns the time separation of the 1st and 2nd pulses found on trace 1.* |

**Error Messages**

- If a P-series sensor is not connected, error -241, "Hardware missing" occurs.
- If a P-series sensor is connected and Free Run trigger acquisition is selected, error –221, "Settings conflict" occurs.

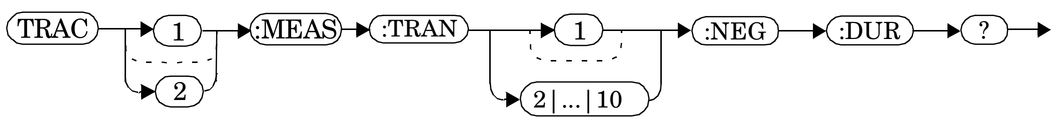
**NOTE**

If you attempt to measure a pulse out of the range of the capture, for example, measure the 5th pulse and there are only 4 pulses displayed, the power meter returns #0##9.91E37 as the result.

# TRACe[1]|2:MEASurement: TRANsition[1]|...|10:NEGative:DURation?

This command returns the $n^{th}$ negative transition duration found on a trace.

### Syntax




**Reset Condition** On reset, this parameter is not affected.

### Example

| | |
|---|---|
| `TRAC:MEAS:TRAN8:NEG:DUR?` | *This command returns the 8th negative transition duration found on trace 1.* |

**Error Messages**

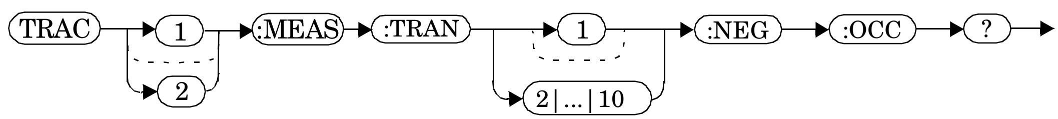
- If a P-series sensor is not connected, error -241, “Hardware missing” occurs.
- If a P-series sensor is connected and Free Run trigger acquisition is selected, error –221, “Settings conflict” occurs.

**NOTE** If you attempt to measure a pulse out of the range of the capture, for example, measure the 5th pulse and there are only 4 pulses displayed, the power meter returns #0#0#0 as the result.

# TRACe[1]|2:MEASurement: TRANsition[1]|...|10:NEGative:OCCurrence?

This command returns the position, relative to the trigger instant, of the n^th occurrence of a negative transition found on a trace.

**Syntax**




**Reset Condition** On reset, this parameter is not affected.

**Example**

| | |
|---|---|
| `TRAC2:MEAS:TRAN7:NEG:OCC?` | *This command returns the position, relative to the trigger instant, of the 7th occurrence of a negative transition found on trace 2.* |

**Error Messages**

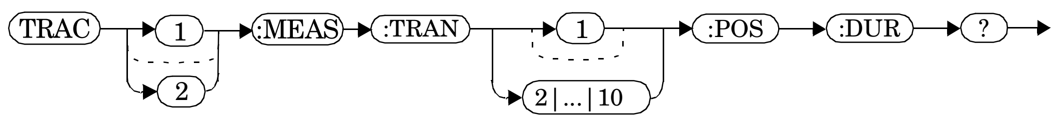
- If a P-series sensor is not connected, error -241, “Hardware missing” occurs.
- If a P-series sensor is connected and Free Run trigger acquisition is selected, error –221, “Settings conflict” occurs.

**NOTE** If you attempt to measure a pulse out of the range of the capture, for example, measure the 5th pulse and there are only 4 pulses displayed, the power meter returns #0#0#0 as the result.

# TRACe[1]|2:MEASurement: TRANsition[1]|...|10:POSitive:DURation?

This command returns the n$^{th}$ positive transition duration found on a trace.

### Syntax




**Reset Condition** On reset, this parameter is not affected.

### Example

| | |
|---|---|
| `TRAC:MEAS:TRAN10:POS:DUR?` | *This command returns the 10th positive transition duration found on trace 1.* |

**Error Messages**

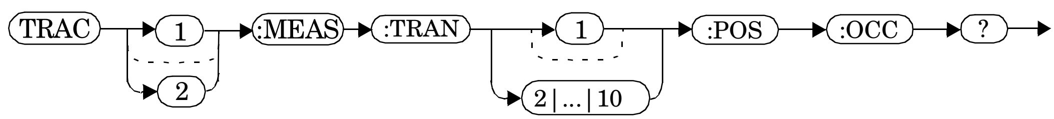
- If a P-series sensor is not connected, error -241, "Hardware missing" occurs.
- If a P-series sensor is connected and Free Run trigger acquisition is selected, error –221, "Settings conflict" occurs.

**NOTE** If you attempt to measure a pulse out of the range of the capture, for example, measure the 5th pulse and there are only 4 pulses displayed, the power meter returns #0#0#0 as the result.

# TRACe[1]|2:MEASurement: TRANsition[1]|...|10:POSitive:OCCurrence?

This command returns the position, relative to the trigger instant, of the $n^{th}$ occurrence of a positive transition found on a trace.

**Syntax**




**Reset Condition** On reset, this parameter is not affected.

**Example**

| | |
|---|---|
| `TRAC2:MEAS:TRAN:POS:OCC?` | *This command returns the position, relative to the trigger instant, of the 1st occurrence of a positive transition found on trace 2.* |

**Error Messages**

- If a P-series sensor is not connected, error -241, “Hardware missing” occurs.
- If a P-series sensor is connected and Free Run trigger acquisition is selected, error –221, “Settings conflict” occurs.

**NOTE** If you attempt to measure a pulse out of the range of the capture, for example, measure the 5th pulse and there are only 4 pulses displayed, the power meter returns #0#0#0 as the result.

# TRACe[1]|2:MEASurement: REFerence? <numeric_value>

This command is used to find the reference power level. This provides the reference power level to calculate the pulse parameters.

Commonly used reference levels are 0%, 10%, 50%, 90%, and 100%. You can set the reference level to measure overshoot at 125% and undershoot at -25%.

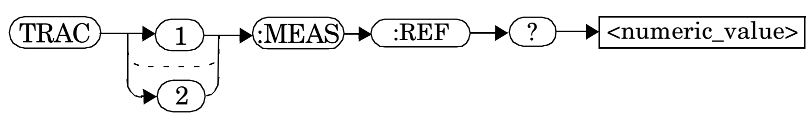
**Algorithm**

$P_{x\%} = P_{0\%} + x/100\ (P_{100\%} - P_{0\%})$

where:

- $0\% <= x <= 100\%$
- $P_{0\%}$ = level of low state
- $P_{100\%}$ = level of high state
- $P_{0\%}$ , $P_{100\%}$ and $P_{x\%}$ are all in the same unit of measurement, for example, Watts.

**Syntax**




**Reset Condition**

On reset, this parameter is not affected.

**Example**

| | |
|---|---|
| `TRAC2:MEAS:REF? 100` | *This command returns the high state power for trace 2.* |

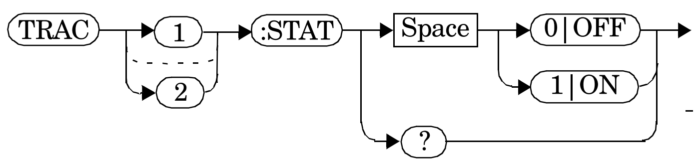
**Error Messages**

- If a P-series sensor is not connected, error -241, "Hardware missing" occurs.
- If a P-series sensor is connected and Free Run trigger acquisition is selected, error –221, "Settings conflict" occurs.

# TRACe[1]|2:STATe <boolean>

This command enables or disables trace capture for the specified channel.

**Syntax**




**Example**

| | |
|---|---|
| `TRAC2:STAT 1` | *This command enables trace capture for channel B.* |

**Reset Condition** On reset trace capture is set to OFF.

**Query** `TRACe[1]|2:STATe?`

The query command enters a 1 or 0 into the output buffer indicating whether or not trace capture is enabled or disabled.

- `1` is returned when trace capture is enabled.
- `0` is returned when trace capture is disabled.

**Query Example**

| | |
|---|---|
| `TRAC1:STAT?` | *This command queries the current state of trace capture for channel A.* |

**Error Messages**

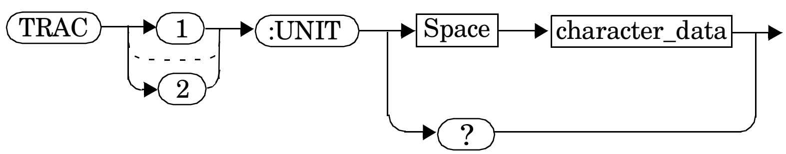
- If a P-series or E-series E9320 sensor is not connected, error -241, “Hardware missing” occurs.
- If an E-series E9320 sensor is connected and set to `AVERage` mode rather than `NORMal` mode, error –221, “Settings conflict” occurs.

# TRACe[1]|2:UNIT <character_data>

This command sets the units for the trace for the specified channel

**NOTE** This command is included for compatibility purposes only. It has the same purpose as `[SENSe[1]]|SENSe2:TRACe:UNIT <character_data>`, which should be the preferred command.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | • DBM: dBm.<br>• W: Watts. | DBM<br>W |

### Example

`TRAC2:UNIT W` *This command sets the trace units for channel B Watts.*

**Reset Condition** On reset the units are set to dBm.

**Query** `TRACe[1]|2:UNIT?`

The query command returns the current value of `character_data`.

### Query Example

`TRAC2:UNIT?` *This command queries the current trace units for channel B.*

# 13 TRIGger Subsystem

# TRIGger Subsystem

The `TRIGger` subsystem is used to synchronize device actions with events. It includes the `ABORt`, `INITiate` and `TRIGger` commands. These are all at the root level in the command hierarchy but they are grouped here because of their close functional relationship.

| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
| `ABORt[1]\|2` | | [no query] [non-SCPI] | page 452 |
| `INITiate[1]\|2` | | | |
| `:CONTinuous` | <boolean> | | page 454 |
| `[:IMMediate]` | | [no query] | page 456 |
| `INITiate` | | | |
| `:CONTinuous` | | | |
| `:ALL` | <boolean> | | page 457 |
| `:SEQuence[1]\|2` | <boolean> | | page 458 |
| `[:IMMediate]` | | | |
| `:ALL` | | [no query] | page 460 |
| `:SEQuence[1]\|2` | | [no query] | page 461 |
| `TRIGger[1]\|2` | | | |
| `:DELay` | | | |
| `:AUTO` | <boolean> | | page 463 |
| `[:IMMediate]` | | [no query] | page 465 |
| `:SOURce` | BUS\|EXTernal\|HOLD\| IMMediate\|INTernal[[1]\|2] | | page 466 |
| `TRIGger` | | | |
| `[:SEQuence]` | | | |
| `:DELay` | <numeric_value> | | page 469 |
| `:HOLDoff` | <numeric_value> | | page 471 |
| `:HYSTeresis` | <numeric_value> | | page 473 |
| `:LEVel` | <numeric_value> | | page 475 |

| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
| :AUTO | <boolean> | | page 477 |
| :SLOPe | <character_data> | | page 479 |
| [:SEQuence[1]\|2] | | | |
| :COUNt | <numeric_value> | | page 480 |
| :DELay | | | |
| :AUTO | <boolean> | | page 482 |
| :IMMediate | | [no query] | page 484 |
| :SOURce | BUS\|EXTernal\|HOLD\|IMMediate\|INTernal[[1]\|2] | | page 485 |

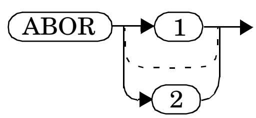
Many of the above commands contain a numeric which represents a channel number. For example `TRIGger1` and `TRIGger2` represent channel A and channel B respectively. Channel B commands cannot be used with the single channel N1911A power meter and result in the error “Header suffix out of range.”

# ABORt[1]|2]

This command removes the specified channel from the wait for trigger state and places it in the idle state. It does not affect any other settings of the trigger system. When the `INITiate` command is sent, the trigger system responds as it did before `ABORt` was executed.

If `INITiate:CONTinuous` is `ON`, then after `ABORt` the specified channel immediately goes into the wait for trigger state.

### Syntax




### Example

| | |
|---|---|
| `ABOR` | *This command places channel A in the idle state.* |

# INITiate Commands

Initiate commands allow you to place the power meter in the wait for trigger state.

The `INITiate` commands are overlapped, that is, the power meter can continue parsing and executing subsequent commands while initiated. Note that the pending operation flag is set, when the power meter enters an idle state and the flag is cleared when it re-enters the idle state.

The following commands are described in this section:

`INITiate[1]|2:CONTinuous <boolean>`

`INITiate[1]|2[:IMMediate]`

`INITiate:CONTinuous:ALL <boolean>`

`INITiate:CONTinuous:SEQuence[1]|2 <boolean>`

`INITiate[:IMMediate]:ALL`

`INITiate[:IMMediate]:SEQuence[1]|2`

# INITiate[1]|2:CONTinuous <boolean>

This command sets the power meter for either a single trigger cycle or continuous trigger cycles. A trigger cycle means that the power meter exits the wait for trigger state and starts a measurement.

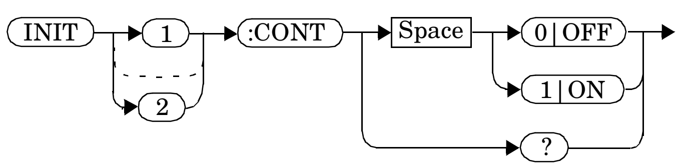
When entering local mode, if `TRIGger[:SEQuence[1]|2]:SOURce` is set to `INT[[1]|2]` or `EXT`, `INITiate:CONTinuous` is not changed. For other trigger sources, `INITiate:CONTinuous` is set to `ON`.

If `INITiate:CONTinuous` is set to:

- `OFF`, the trigger system remains in the idle state until it is set to `ON`, or `INITiate:IMMediate` is received. Once this trigger cycle is complete the trigger system returns to the idle state.
- `ON`, the trigger system is initiated and exits the idle state. On completion of each trigger cycle, the trigger system immediately commences another trigger cycle without entering the idle state.

**NOTE** This command performs the same function as `INITiate:CONTinuous:SEQuence[1]|2 <boolean>`.

### Syntax




### Example

| | |
|---|---|
| `INIT2:CONT ON` | *This command places channel B in the wait for trigger state.* |

### Reset Condition

On reset (`*RST`), this command is set to `OFF`.

On preset (SYSTem:PRESet) and instrument power-up, when entering local mode, if TRIGger[:SEQuence[1]|2]:SOURce is set to INT[[1]|2] or EXT, INITiate:CONTinuous is not changed. For other trigger sources, INITiate:CONTinuous is set to ON.

**Query**

INITiate[1]|2:CONTinuous?

The query enters a 1 or 0 into the output buffer.

- 1 is returned when there is continuous triggering.
- 0 is returned when there is only a single trigger.

**Query Example**

| | |
|---|---|
| INIT2:CONT? | *This command queries whether channel B is set for single or continuous triggering.* |

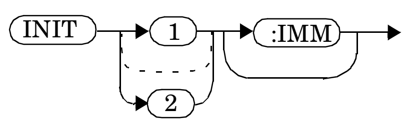
# INITiate[1]|2[:IMMediate]

This command sets the power meter in the wait for trigger state. When a trigger is received, the measurement is taken and the result placed in the power meter memory. If `TRIGger:SOURce` is set to `IMMediate` the measurement begins as soon as `INITiate:IMMediate` is executed.

Use `FETCh?` to transfer a measurement from memory to the output buffer. Refer to "FETCh[1]|2|3|4 Queries" on page 99 for further details.

**NOTE** This command performs the same function as `INITiate:[IMMediate]:SEQuence[1]|2`.

### Syntax




### Example

| | |
|---|---|
| `INIT2:IMM` | *This command places channel B in the wait for trigger state.* |

### Error Messages

If the power meter is not in the idle state or `INITiate:CONTinuous` is `ON`, error -213, "INIT ignored" occurs.

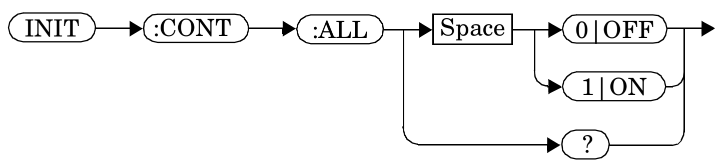
# INITiate:CONTinuous:ALL <boolean>

Sets all trigger sequences to be continuously initiated.

If `INITiate:CONTinuous:ALL` is set to:

- `ON`, trigger sequences are set to be continuously initiated.
- `OFF`, trigger sequences are not set to be continuously initiated.

**Syntax**




**Example**

| | |
|---|---|
| `INIT:CONT:ALL ON` | *This command sets all trigger sequences to be continuously initiated.* |

**Reset Condition**

On reset (`*RST`), this command is set to `OFF`.

On preset (`SYSTem:PRESet`) and instrument power-up, when entering local mode, if `TRIGger[:SEQuence[1]|2]:SOURce` is set to `INT[[1]|2]` or `EXT`, `INITiate:CONTinuous` is not changed. For other trigger sources, `INITiate:CONTinuous` is set to `ON`.

**Query**

`INITiate:CONTinuous:ALL?`

The query enters a `1` or `0` into the output buffer.

- `1` is returned when trigger sequences are set to be continuous.
- `0` is returned when trigger sequences are not set to be continuous.

**Query Example**

| | |
|---|---|
| `INIT:CONT:ALL?` | *This command queries whether both channels are in a wait for trigger state.* |

# INITiate:CONTinuous:SEQuence[1]|2 <boolean>

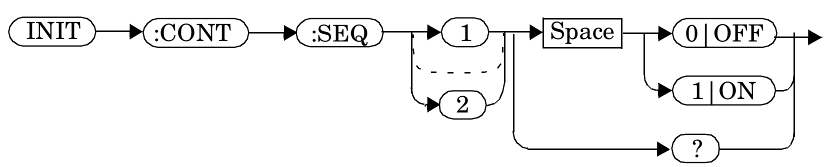
This command sets the power meter for either a single trigger cycle or continuous trigger cycles. A trigger cycle means that the power meter exits the wait for trigger state and starts a measurement. When entering local mode, `INITiate:CONTinuous` is set to `ON`.

If `INITiate:CONTinuous:SEQuence[1|2] <boolean>` is set to:

- `OFF`, the trigger system remains in the idle state until it is set to `ON`, or `INITiate:IMMediate` is received. Once this trigger cycle is complete the trigger system returns to the idle state.
- `ON`, the trigger system is initiated and exits the idle state. On completion of each trigger cycle, the trigger system immediately commences another trigger cycle without entering the idle state.

**NOTE** This command performs the same functions as `INITiate[1]|2:CONTinuous <boolean>`.

**Syntax**




**Example**

| | |
|---|---|
| `INIT:CONT:SEQ2 ON` | *This command places channel B in a wait for trigger state.* |

**Reset Condition** On reset (`*RST`), this command is disabled.

On preset (`SYSTem:PRESet`) and instrument power-up, this command is enabled.

**Query** `INITiate[1]|2:CONTinuous:SEQuence?`

The query enters a `1` or `0` into the output buffer.

- `1` is returned when there is continuous triggering.
- `0` is returned when there is only a single trigger.

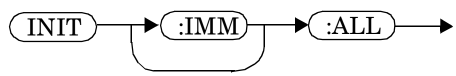
**Query Example**

| | |
|---|---|
| `INIT2:CONT:SEQ?` | *This command queries whether channel B is set for single or continuous triggering.* |

# INITiate[:IMMediate]:ALL

This command initiates all trigger sequences.

### Syntax




### Example

| | |
|---|---|
| `INIT:IMM:ALL` | *This command initiates all trigger sequences.* |

### Error Messages

If the power meter is not in the idle state or `INITiate:CONTinuous` is `ON`, error -213, “INIT ignored” occurs.

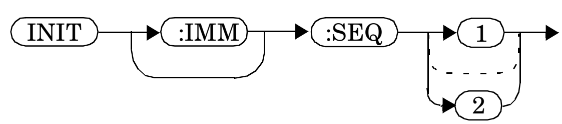
# INITiate[:IMMediate]:SEQuence[1]|2

This command sets the power meter in the wait for trigger state. When a trigger is received, the measurement is taken and the result placed in the power meter memory. If `TRIGger:SOURce` is set to `IMMediate` the measurement begins as soon as `INITiate:IMMediate` is executed.

Use `FETCh?` to transfer a measurement from memory to the output buffer. Refer to "FETCh[1]|2|3|4 Queries" on page 99 for further information.

**NOTE** This command performs the same function as `INITiate[1]|2:[IMMediate]`.

### Syntax




### Example

| | |
|---|---|
| `INIT:IMM:SEQ1` | *This command places channel A in the wait for trigger state.* |

### Error Messages

If the power meter is not in the "idle" state or `INITiate:CONTinuous` is `ON`, error -213, "INIT ignored" occurs.

# TRIGger Commands

TRIGger commands control the behavior of the trigger system.

The following commands are described in this section:

```
TRIGger[1]|2:DELay:AUTO <boolean>

TRIGger[1]|2:SOURce BUS|IMMediate|HOLD

TRIGger[1]|2[:IMMediate]

TRIGger[:SEQuence]:DELay <numeric_value>

TRIGger[:SEQuence]:HOLDoff <numeric_value>

TRIGger[:SEQuence]:HYSTeresis <numeric_value>

TRIGger[:SEQuence]:LEVel <numeric_value>

TRIGger[:SEQuence]:LEVel:AUTO <boolean>

TRIGger[:SEQuence]:SLOPe <character_data>

TRIGger[:SEQuence[1]|2]:COUNt <numeric_value>

TRIGger[:SEQuence[1]|2]:DELay:AUTO <boolean>

TRIGger[:SEQuence[1]|2]:IMMediate

TRIGger[:SEQuence[1]|2]:SOURce
BUS|EXTernal|HOLD|IMMediate|INTernal[[1]|2
```

# TRIGger[1]|2:DELay:AUTO <boolean>

This command is used to determine whether or not there is a settling-time delay before a measurement is made.

When this command is set to:

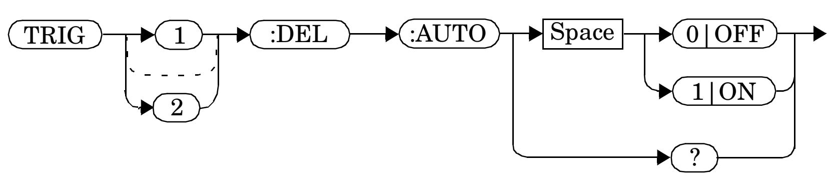
- ON, the power meter inserts a settling-time delay before taking the requested measurement. This settling time allows the internal digital filter to be updated with new values to produce valid, accurate measurement results. The trigger with delay command allows settling time for the internal amplifiers and filters. It does not allow time for power sensor delay

  In cases of large power changes, the delay may not be sufficient for complete settling. Accurate readings can be assured by taking two successive measurements for comparison.

- OFF, the power meter makes the measurement immediately a trigger is received.

TRIGger[1]|2:DELay:AUTO is ignored if TRIGger[1]|2[:IMMediate] is set to ON.

**Syntax**




**Example**

| | |
|---|---|
| TRIG:DEL:AUTO ON | *This command enables a delay on channel A.* |

**Reset Condition**

On reset, TRIGger:DELay:AUTO is set to ON.

**Query**

`TRIGger:DELay:AUTO?`

The query enters a `1` or `0` into the output buffer indicating the status of `TRIGger:DELay:AUTO`.

- `1` is returned when it is `ON`.
- `0` is returned when it is `OFF`.

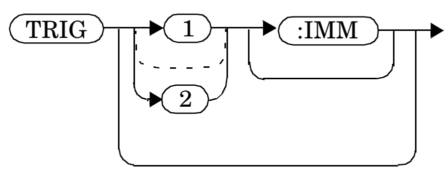
# TRIGger[1]|2[:IMMediate]

This command causes a trigger to occur immediately, provided the specified channel is in the wait for trigger state. When this command is executed, the measurement result is stored in the power meter's memory. Use `FETCh?` to place the measurement result in the output buffer.

`TRIGger[1]|2:DELay:AUTO` is ignored if `TRIGger[1]|2[:IMMediate]` is set to `ON`.

**NOTE** This command performs the same function as `INITiate[1]|2:[IMMediate]`.

### Syntax




### Example

| | |
|---|---|
| `TRIG` | *This command causes a channel A trigger to occur immediately.* |

### Error Messages

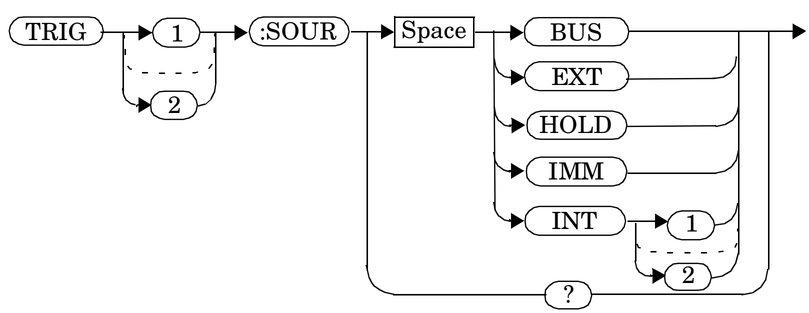
If the power meter is not in the wait for trigger state, then `TRIGger:IMMediate` causes error -211, "Trigger ignored".

# TRIGger[1]|2:SOURce BUS|EXTernal|HOLD|IMMediate|INTernal[[1]|2]

This command configures the trigger system to respond to the specified source. This command only selects the trigger source. Use the `INITiate` command to place the power meter in the wait for trigger state.

**NOTE** This command has been included for compatibility purposes. It has the same purpose as `TRIGger[:SEQuence[1]|2]:SOURce BUS|EXTernal|HOLD|IMMediate|INTernal[[1]|2]` which should be used in preference.

### Syntax



### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| source | Available trigger sources:<br>• BUS: the trigger source is the group execute trigger <GET> bus command, a *TRG common command or the TRIGGER:IMMediate SCPI command.<br>• EXTernal: the trigger source is the trigger input in the back panel.<br>• HOLD: triggering is suspended. The only way to trigger the power meter is to use TRIGger:IMMediate.<br>• IMMediate: the trigger system is always true. If INITiate:CONTinuous is ON the power meter is continually triggering free (free run mode). If an INITiate:IMMediate command is sent a measurement is triggered then the power meter returns to the idle state.<br>• INTernal: either INT1 (channel A) or INT2 (channel B). | BUS<br>EXTernal<br>HOLDIMMediate<br>INTernal[[1]\|2] |

**NOTE**

The trigger source is set to IMMediate on instrument power-up and when entering local mode.

The MEASure and CONFigure commands automatically set the trigger source to IMMediate.

The READ? or MEASure commands should not be used if the trigger source is set to BUS or HOLD.

### Example

| | |
|---|---|
| TRIG:SOUR IMM | *This command configures channel A for immediate triggering.* |

**Reset Condition** On reset, the trigger source is set to IMMediate.

**Query**

`TRIGger:SOURce?`

The query returns the current trigger source, either `IMM`, `BUS` or `HOLD`.

**Query Example**

| | |
|---|---|
| `TRIG:SOUR?` | *This command queries channel A's trigger source.* |

**Error Messages**

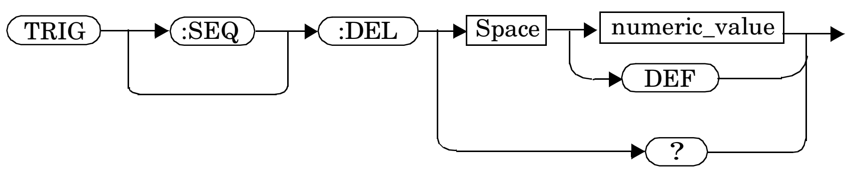
- For dual channel power meters: if the master is changed to `IMM`, `BUS` or `HOLD`, error -221 "Settings Conflict" occurs. In such situations the slave's `TRIG:SOUR` must be changed so that it is no longer a slave.
- If the source is changed to `INT1`, `INT2` or `EXT` and `SENS:SPEED` has a value of 200, error -221 "Settings Conflict" occurs.
- If the source is changed to `INT1`, `INT2` or `EXT` and `SENS:DET:FUNC` is set to `AVERage`, error -221 "Settings Conflict" occurs.

# TRIGger[:SEQuence]:DELay <numeric_value>

This command sets the delay between the recognition of a trigger event and the start of a measurement.

**NOTE** The command is accepted for `TRIGger[:SEQuence[1]]` (channel A) only, for both single and dual channel power meters.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | The delay between the recognition of a trigger event and the start of the measurement.<br>• `DEF`: the default value is 0 seconds.<br>Units are resolved to 1.25 ns. | -1 to 1 seconds<br>`DEF` |

### Example

| | |
|---|---|
| `TRIG:SEQ:DEL 0.001` | *This command sets a delay of 1 ms for channel A.* |

### Reset Condition

On reset, the trigger delay is set to 0 seconds.

**Query**

`TRIGger[:SEQuence]:DELay?`

The query returns the current setting of the trigger delay.

**Query Example**

| | |
|---|---|
| `TRIG:SEQ:DEL?` | *This command queries the trigger delay of channel A.* |

**Reset Condition**

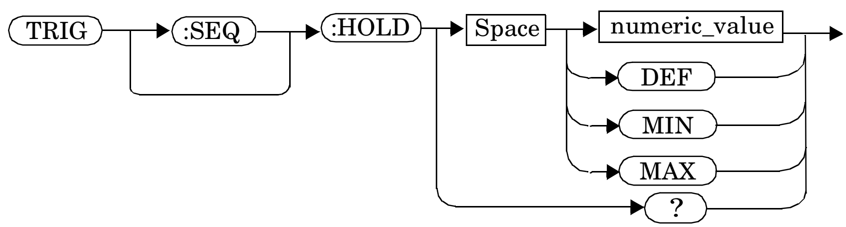
On reset, trigger delay is set to 0 seconds.

# TRIGger[:SEQuence]:HOLDoff <numeric_value>

This command sets the trigger υ holdoff in seconds.

**NOTE** The command is accepted for `TRIGger[:SEQuence[1]]` (channel A) only, for both single and dual channel power meters.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | The trigger holdoff in seconds.<br>• `DEF:` the default value is 0.1 µs.<br>• `MIN:` 0.1 µs.<br>• `MAX:` 400 ms.<br>Units are resolved to 1 ns. | 1 us to 0.4 seconds<br>`DEF`<br>`MIN`<br>`MAX` |

### Example

| | |
|---|---|
| `TRIG:SEQ1:HOLD 0.1` | *This command sets the trigger holdoff to 100 ms for channel A.* |

### Reset Condition

On reset the trigger holdoff is set to 1 us.

**Query** `TRIGger[:SEQuence]:HOLDoff?`

The query returns the current trigger holdoff setting.

**Query Example**

| | |
|---|---|
| `TRIG:SEQ:HOLD?` | *This command queries the trigger holdoff setting for channel A.* |

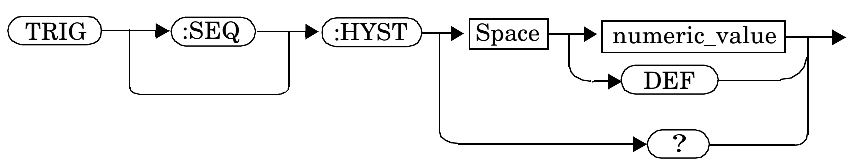
# TRIGger[:SEQuence]:HYSTeresis <numeric_value>

This command sets:

- How far a signal must fall below `TRIG:LEVel` before a rising edge can be detected.
- How far a signal must rise above `TRIG:LEVel` before a falling edge can be detected.

**NOTE** The command is accepted for `TRIGger[:SEQuence[1]]` (channel A) only, for both single and dual channel power meters.

### Syntax




### Example

`TRIG:SEQ:HYST 0.1` *This command sets the value to 2 dB for channel A.*

### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | How far a signal must fall/rise before a rising or falling edge can be detected.<br>• `DEF`: the default value is 0 dB.<br>Units are resolved to 0.05 dB. | 0 to 3 dB<br>`DEF` |

**Reset Condition** On reset the value is set to 0 dB.

**Query**

`TRIGger[:SEQuence]:HYSTeresis?`

The query returns the current value in dB.

**Query Example**

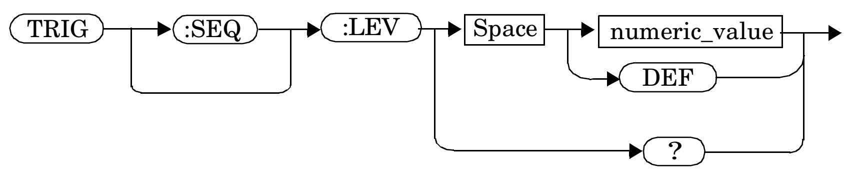
| | |
|---|---|
| `TRIG:SEQ:HYST?` | *This command queries the value for channel A.* |

# TRIGger[:SEQuence]:LEVel <numeric_value>

This command sets the power level at which a trigger event is recognized.

**NOTE** The command is accepted for `TRIGger[:SEQuence[1]]` (channel A) only, for both single and dual channel power meters.

## Syntax




## Example

| | |
|---|---|
| `TRIG:SEQ:LEV 10` | *This command sets the power level for a trigger event to 10 dBm* |

## Parameters

| Item | Description/Default | Range of Values[a] |
|---|---|---|
| numeric_value | The power level at which a trigger event is recognized.<br>• `DEF:` the default value is 0 dBm.<br>Units are resolved to 0.1 dBm. | -40 to 20 dBm<br>`DEF` |

a. If a channel offset has been previously set, a higher numeric value is permitted. See “Channel Offsets” on page 35 for more information.

## Reset Condition

On reset the power level is set to 0 dBm.

**Query**

`TRIGger[:SEQuence]:LEVel?`

The query returns the current power level setting.

**Query Example**

| | |
|---|---|
| `TRIG:SEQ1:LEV?` | *This command queries the power level setting for channel A.* |

# TRIGger[:SEQuence]:LEVel:AUTO <boolean>

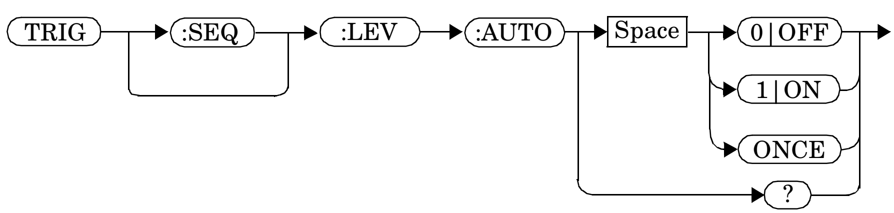
This command enables/disables automatic setting of the trigger level.

When this command is set to:

- `ON`, automatic setting of the trigger level is enabled.
- `OFF`, automatic setting of the trigger level is disabled.
- `ONCE`, automatic setting of the trigger level is enabled for one trigger event only. The value is then set to `OFF`.

**NOTE** The command is accepted for `TRIGger[:SEQuence[1]]` (channel A) only, for both single and dual channel power meters.

**Syntax**




**Example**

| | |
|---|---|
| `TRIG:SEQ:LEV:AUTO 0` | *This command disables the automatic setting of the trigger level for channel A.* |

**Reset Condition** On reset the value is set to ON.

**Query** `TRIGger[:SEQuence]:LEVel:AUTO?`

The query enters a `1` or `0` into the output buffer indicating the status of `TRIGger[:SEQuence]:LEVel:AUTO`.

- `1` is returned when it is `ON`.
- `0` is returned when it is `OFF`.

### Query Example

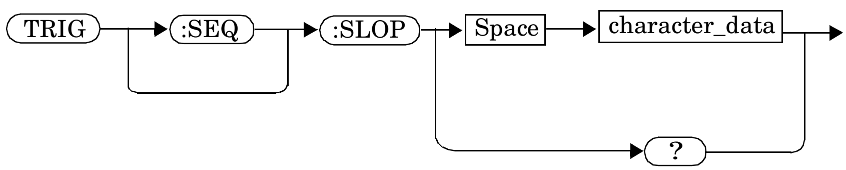
`TRIG:SEQ:LEV:AUTO?` *This command queries the setting for channel A.*

# TRIGger[:SEQuence]:SLOPe <character_data>

This command specifies whether a trigger event is recognized on the rising or falling edge of a signal.

**NOTE** The command is accepted for `TRIGger[:SEQuence[1]]` (channel A) only, for both single and dual channel power meters.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | How a trigger event is recognized:<br>• `POSitive`: a trigger event is recognized on the rising edge of a signal.<br>• `NEGative`: a trigger event is recognized on the falling edge of a signal. | `POSitive`<br>`NEGative` |

### Reset Condition

On reset the value is set to `POSitive`.

### Query

`TRIGger[:SEQuence]:SLOPe?`

The query returns the current value of `<character_data>`.

### Query Example

| | |
|---|---|
| `TRIG:SEQ:SLOP?` | *This command queries the current value of <character_data> for channel A.* |

# TRIGger[:SEQuence[1]|2]:COUNt <numeric_value>

This command controls the path of the trigger subsystem in the upward traverse of the wait for trigger state. `COUNt` loops through the event detection/measurement cycle are performed. That is, `COUNt` measurements are performed in response to `COUNt` trigger events.

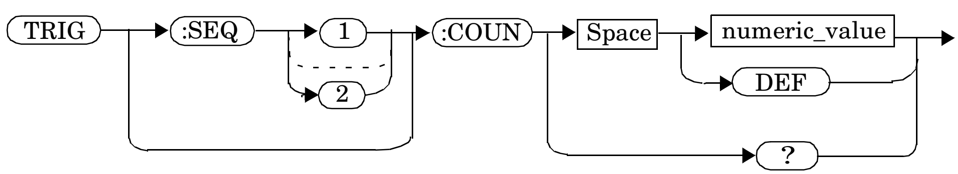
`COUNt` can be set to a value >1 only when:

- `[SENSe[1]]|SENSe2:MRATe <character_data>` is set to `FAST`

and

- `TRIGger[1]|2:SOURce` set to `BUS`, `IMMediate` or `HOLD`.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| numeric_value | The number of triggered events for the measurement cycle.<br>• `DEF`: the default value is 1. | 1 to 50<br>`DEF` |

### Example

| | |
|---|---|
| `TRIG:SEQ1:COUN 10` | *This command sets the number of triggered events to 10 for the channel A measurement cycle.* |

### Reset Condition

On reset, the value is set to 1.

**Query**

`TRIGger[1]|2[:SEQuence[1]|2]:COUNt?`

The query returns the current setting of trigger events for a specified channel.

**Query Example**

| | |
|---|---|
| `TRIG:SEQ2:COUN?` | *This command queries the number of triggered events for the channel B measurement cycle.* |

**Error Messages**

If `COUNt` >1 when `[SENSe[1]]|SENSe2:MRATe <character_data>` is set to `NORMal` or `DOUBle`, Error -221, “Settings Conflict” occurs.

# TRIGger[:SEQuence[1]|2]:DELay:AUTO <boolean>

This command is used to determine whether or not there is a settling-time delay before a measurement is made.

When this command is set to:

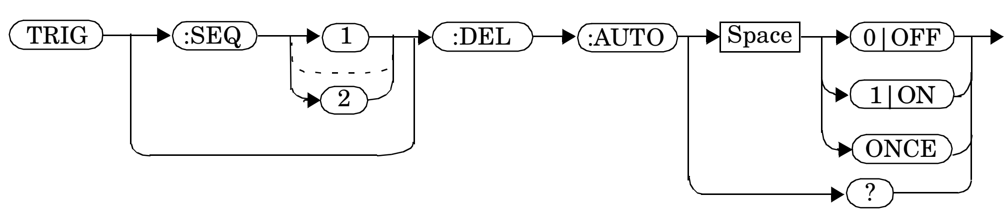
- ON, the power meter inserts a settling-time delay before taking the requested measurement and for subsequent measurements. This settling time allows the internal digital filter to be updated with new values to produce valid, accurate measurement results. The trigger with delay command allows settling time for the internal amplifiers and filters. It does not allow time for power sensor delay.

  In cases of large power changes, the delay may not be sufficient for complete settling. Accurate readings can be assured by taking two successive measurements for comparison.

- OFF, no settling-time delay is inserted and the power meter makes the measurement immediately a trigger is received.
- ONCE, a settling-time delay is inserted before taking the requested measurement, for one measurement only.

TRIGger[1]|2:DELay:AUTO is ignored if TRIGger[1]|2[:IMMediate] is set to ON.

### Syntax




### Example

`TRIG:SEQ:DEL:AUTO ON` *This command enables a delay on channel A.*

### Reset Condition

On reset, TRIGger:DELay:AUTO is set to ON.

### Query

`TRIGger:DELay:AUTO?`

The query enters a `1` or `0` into the output buffer indicating the status of `TRIGger:DELay:AUTO`.

- `1` is returned when it is `ON`.
- `0` is returned when it is `OFF`.

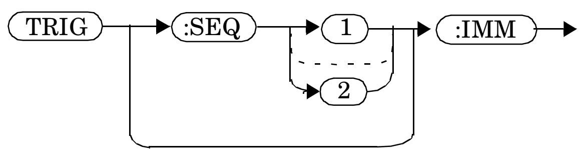
### Query Example

| | |
|---|---|
| `TRIG:SEQ2:DEL:AUTO?` | *This command queries the settling-time delay of channel B.* |

# TRIGger[:SEQuence[1]|2]:IMMediate

This command provides a one time over-ride of the normal process of the downward path through the wait for trigger state. It causes the immediate exit of the event detection layer if the trigger system is in this layer when the command is received. In other words, the instrument stops waiting for a trigger and takes a measurement ignoring any delay set by `TRIG:DELay`.

### Syntax




### Example

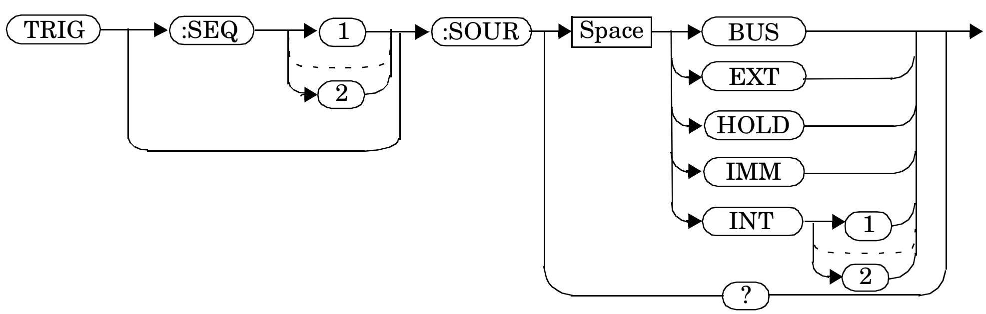
| | |
|---|---|
| `TRIG:SEQ:IMM` | *This command initiates a measurement on channel A.* |

# TRIGger[:SEQuence[1]|2]:SOURce BUS|EXTernal|HOLD|IMMediate|INTernal[[1]|2]

This command configures the trigger system to respond to the specified source. This command only selects the trigger source. Use the `INITiate` command to place the power meter in the wait for trigger state.

**NOTE** This command has the same purpose as `TRIGger[1]|2:SOURce BUS|EXTernal|HOLD|IMMediate|INTernal[[1]|2]`

### Syntax



### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| source | Available trigger sources:<br>• `BUS`: the trigger source is the group execute trigger `<GET>` bus command, a `*TRG` common command or the `TRIGGER:IMMediate` SCPI command.<br>• `EXTernal`: the trigger source is the trigger input in the back panel.<br>• `HOLD`: triggering is suspended. The only way to trigger the power meter is to use `TRIGger:IMMediate`.<br>• `IMMediate`: the trigger system is always true. If `INITiate:CONTinuous` is `ON` the power meter is continually triggering free (free run mode). If an `INITiate:IMMediate` command is sent a measurement is triggered then the power meter returns to the idle state.<br>• `INTernal`: either `INT1` (channel A) or `INT2` (channel B). | `BUS`<br>`EXTernal`<br>`HOLD`<br>`IMMediate`<br>`INTernal[[1]|2]` |

**NOTE** The trigger source is set to `IMMediate` on instrument power-up and when entering local mode.

The `MEASure` and `CONFigure` commands automatically set the trigger source to `IMMediate`.

The `READ?` or `MEASure` commands should not be used if the trigger source is set to `BUS` or `HOLD`.

### Example

| | |
|---|---|
| `TRIG:SOUR IMM` | *This command configures channel A for immediate triggering.* |

**Reset Condition** On reset, the trigger source is set to `IMMediate`.

**Query**

`TRIGger[:SEQuence[1]|2]:SOURce?`

The query returns the current trigger source.

**Query Example**

| | |
|---|---|
| `TRIG:SEQ1:SOUR?` | *This command queries the current trigger source for channel A.* |

**Error Messages**

- For dual channel power meters: if the master is changed to `IMM`, `BUS` or `HOLD`, error -221 "Settings Conflict" occurs. In such situations the slave's `TRIG:SOUR` must be changed so that it is no longer a slave.
- If the source is changed to `INT1`, `INT2` or `EXT` and `SENS:SPEED` has a value of 200, error -221 "Settings Conflict" occurs.
- If the source is changed to `INT1`, `INT2` or `EXT` and `SENS:DET:FUNC` is set to `AVERage`, error -221 "Settings Conflict" occurs.

# 14 UNIT Subsystem

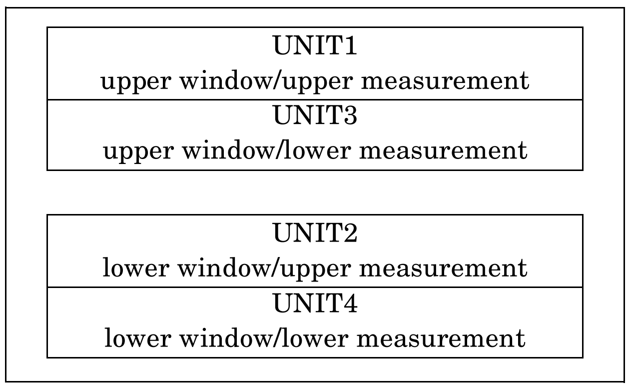
# UNIT Subsystem

The `UNIT` command subsystem:

- Sets power measurement units to dBm or Watts.
- Sets measurement ratio units to dB or % (linear).

Both `UNIT` commands have a numeric suffix which determines which window/measurement is set:

**Figure 14-1 Measurement Display UNIT Block Window**




The following commands are described in this section:

| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
| `UNIT[1]|2|3|4` | | | |
| `:POWer` | <amplitude unit> | | page 491 |
| `:RATio` | <ratio_unit> | [non-SCPI] | page 493 |

The `UNIT:POWer` and `UNIT:POWer:RATio` commands are coupled as follows:

- `If UNIT:POWer` is set to `dBm` then `UNIT:POWer:RATio` is `dB`.
- `If UNIT:POWer` is set to `W` then `UNIT:POWer:RATio` is `%`.

# UNIT[1]|2|3|4:POWer <amplitude_unit>

This command sets the power measurement units for a specified window/measurement. The power suffix set by `UNIT:POWer` is used for any command which accepts a numeric value in more than one unit

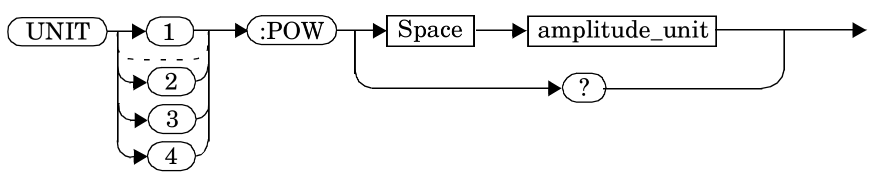
For the N1911A:

- `UNIT1:POWer` sets the power measurement units for the upper window/upper measurement.
- `UNIT2:POWer` sets the power measurement units for the lower window/upper measurement.
- `UNIT3:POWer` sets the power measurement units for the upper window/lower measurement.
- `UNIT4:POWer` sets the power measurement units for the lower window/lower measurement.

For ratio and relative power measurements:

- If `UNIT:POWer` is `W`, the measurement units are percentage.
- If `UNIT:POWer` is `DBM`, the measurement units are dB relative.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| amplitude_unit | The measurement unit.<br>• The default unit is dBm. | `W`<br>`DBM` |

**Example**

| | |
|---|---|
| `UNIT1:POW DBM` | *This command sets the power measurement units for the upper window / upper measurement.* |

**Reset Condition** On reset, all windows/measurements are set to `DBM`.

**Query** `UNIT[1]|2|3|4:POWer?`

The query returns the current setting of the power measurement units.

**Query Example**

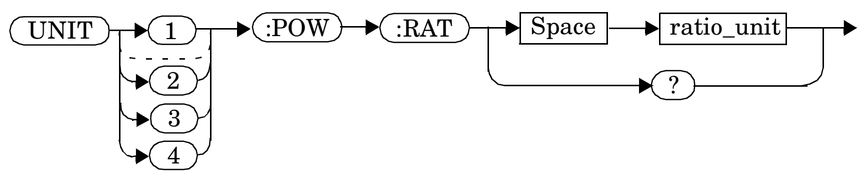
| | |
|---|---|
| `UNIT2:POW?` | *This command queries which measurement units are being used on the lower window / upper measurement.* |

# UNIT[1]|2|3|4:POWer:RATio <ratio_unit>

This command sets the window/measurement ratio units.

- `UNIT1:POWer:RATio` sets the ratio measurement units for the upper window/upper measurement.
- `UNIT2:POWer:RATio` sets the ratio measurement units for the lower window/upper measurement.
- `UNIT3:POWer:RATio` sets the ratio measurement units for the upper window/lower measurement.
- `UNIT4:POWer:RATio` sets the ratio measurement units for the lower window/lower measurement.

## Syntax




## Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| ratio_unit | The ratio measurement unit.<br>• The default unit is DB. | DB<br>PCT |

## Example

| | |
|---|---|
| `UNIT1:POW:RAT DB` | *This command sets the ratio measurement units for the upper window / upper measurement.* |

## Reset Condition

On reset, the value is set to `DB`.

**Query**

`UNIT[1]|2|3|4]:POWer:RATio?`

The query returns the current setting of the ratio measurement units.

**Query Example**

| | |
|---|---|
| `UNIT2:POW:RAT?` | *This command queries which ratio measurement units are being used on the lower window/upper measurement.* |

# 15 SERVice Subsystem

# SERVice Subsystem

The SERVice command subsystem is used to load information such as the power meter processor board revision version and obtain information such as the serial number of the current sensor(s) being used.

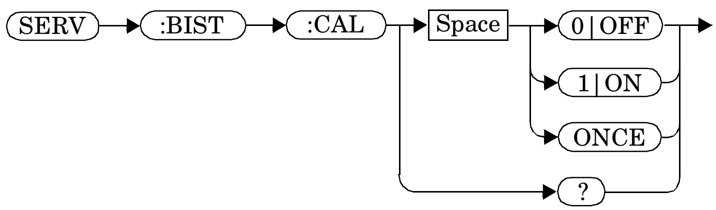
| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
| SERVice | | | |
| :BIST | | | |
| :CALibrator | <boolean> | | page 498 |
| :TBASe | | | |
| :STATe | <boolean> | | page 499 |
| :T? | | [query only] | page 501 |
| :LAN | | | |
| PHOStname | | | page 502 |
| :OPTion | <character_data> | | page 503 |
| :SECure | | | |
| ERASe | | | page 504 |
| :SENSor[1]\|2 | | | |
| :CALFactor | <cal_factor_data> | | page 505 |
| :CDATe? | | [query only] | page 507 |
| :CORRections | | | |
| :STATe | <boolean> | | page 508 |
| :CPLace? | | [query only] | page 510 |
| :FREQuency | | | |
| :MAXimum? | | [query only] | page 511 |
| :MINimum? | | [query only] | page 512 |
| :PCALfactor | <cal_factor_data> | | page 513 |
| :POWer | | | |
| :AVERage | | | |
| :MAXimum? | | [query only] | page 514 |

| Keyword | Parameter Form | Notes | Page |
|---|---|---|---|
| :PEAK | | | |
| :MAXimum? | | [query only] | page 515 |
| :USABle | | | |
| :MAXimum? | | [query only] | page 516 |
| :MINimum? | | [query only] | page 517 |
| :RADC? | | [query only] | page 518 |
| :SNUMber? | | [query only] | page 519 |
| :TNUMber? | | [query only] | page 520 |
| :TYPE? | | [query only] | page 521 |
| :SNUMber | <character_data> | | page 522 |
| :VERSion | | | |
| :PROCessor | <character_data> | | page 523 |
| :SYSTem | <character_data> | | page 524 |

# SERVice:BIST:CALibrator <boolean>

This command enables/disables the calibrator self-test during power-up. It can be used to disable the self-test if it incorrectly indicates a failure. If a load, for example, a sensor, is connected to the calibrator port this could cause the self-test to fail. Also, if it fails the self-test, a Pop-up is displayed for 5 seconds, stating - *If Ref Calibrator test fails disconnect any load attached to it and re-try test.*

### Syntax




### Example

| | |
|---|---|
| `SERV:BIST:CAL OFF` | *This command disables the calibrator self-test during power-up.* |

### Query

`SERVice:BIST:CALibrator?`

The query enters a `1` or `0` into the output buffer indicating the status of the self-test.

- `1` is returned when the self-test is enabled.
- `0` is returned when the self-test is disabled.

### Query Example

| | |
|---|---|
| `SERV:BIST:CAL?` | *This command queries whether the self-test is enabled or disabled.* |

## SERVice:BIST:TBASe:STATe <boolean>

This command sends a 10 MHz time base signal to the rear panel trig out for testing purposes.

**NOTE** This command overrides the `OUTPut:TRIGger[:STATe]` command.

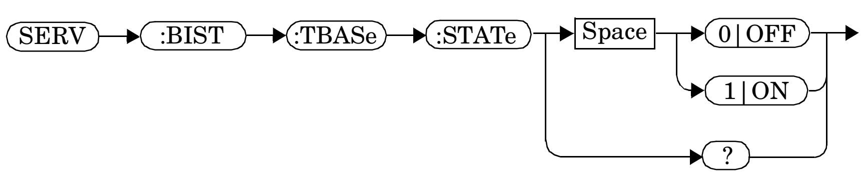
For example, if `OUTPut:TRIGger[:STATe]` is `ON` and the command `SERV:BIST:TBAS ON` is sent, this command overrides the Trigger state and sets it to `OFF`. However, the 10 MHz remains out the Trig out port.

If the `SERV:BIST:TBAS ON` has been sent, the 10 MHz is on and the `OUTPut:TRIGger[:STATe]` is then toggled to `ON`, the channel trigger is now routed to the Trig out overriding the service command turning the 10 MHz to off.

If the command is set to:

- `ON`, the 10 MHz time base signal is sent to the rear panel trigger out connector.
- `OFF`, the 10 MHz time base signal is disabled.

### Syntax




### Example

`SERV:BIST:TBAS:STAT OFF` *This command disables the signal.*

### Reset Condition

On reset, the signal is disabled.

### Query

`SERVice:BIST:TBASe:STAT?`

The query enters a `1` or `0` into the output buffer indicating the status of the 10 MHz time base testing.

- `1` is returned when the signal is enabled.
- `0` is returned when the signal is disabled.

**Query Example**

| | |
|---|---|
| `SERV:BIST:TBASe:STAT?` | *This command queries whether the test is enabled or disabled.* |

# SERVice:BIST:TRIGger:TEST?

This command queries trigger in and out.

- 1 is returned if the test passes.
- 0 is returned if the test fails.

**NOTE** Before running this command, the read panel trigger out must be jumpered to the rear panel trigger in.

### Syntax

SERV → :BIST → :TRIG → :TEST → ? →

### Example

| | |
|---|---|
| `SERV:BIST:TRIG:TEST?` | *This command queries trigger in and out.* |

# SERVice:LAN:PHOStname

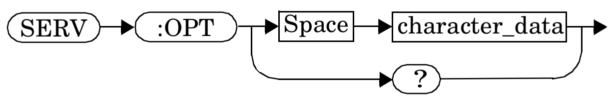
This command preset the LAN hostname to its default value. It requires the serial number to be set-up.

### Syntax

# SERVice:OPTion <character_data>

This command loads the power meter memory with the options fitted. The query form of the command can be used to determine which options are fitted to the unit.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | Details the option number in a comma separated list. A maximum of 30 characters can be used. | A to Z (uppercase)<br>a to z (lowercase)<br>0 - 9<br>_ (underscore) |

### Example

| | |
|---|---|
| `SERV:OPT "003"` | *This command loads the power meter memory with 003 indicating that the unit is a rear panel option.* |

### Query

`SERVice:OPTion?`

The query returns the current option string. For example, if the string "003" is returned, the power meter is fitted with a sensor input and power reference on the back panel.

# SERVice:SECure:ERASe

This command erases the P-series power meter's memory, for example, before you return it to Agilent Technologies for repair or calibration, of all data stored in it.

The memory data erased, includes the save/recall states and power on last states.

### Syntax

# SERVice:SENSor[1]|2:CALFactor <cal_factor_data>

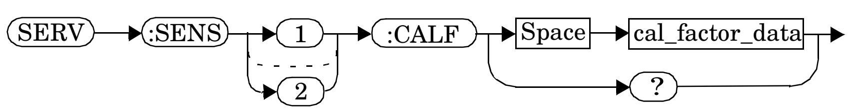
This command writes calibration factor data to, or reads calibration factor data from, the currently connected sensor. The whole calibration factor block must be written at once as a checksum is generated. The new block must not be larger than the existing block.

This command applies to the following sensors:

- E4410 series.
- E9300 series.
- E9320 series, average path data.

For E9320 series sensors, peak path, refer to SERVice:SENSor[1]|2:PCALFactor <cal_factor_data>.

### Syntax




### Parameters

| Item | Description/Default |
|---|---|
| cal_factor_data | A binary data block. Refer to Appendix A for further information. |

### Query

SERVice:SENSor[1]|2:CALFactor?

The query returns the current calibration factor block.

### Query Example

| | |
|---|---|
| SERV:SENS:CALF? | *This command returns the calibration factor block for channel A.* |

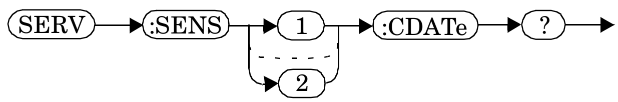
**Error Messages**

- If no power sensor is connected, error -241 “Hardware missing” occurs.
- If a a sensor other than a P-series or E-series power sensor is connected, error -241 “Hardware missing” occurs.
- If an E9320 series sensor is connected and `SERVice:SENSor[1]|2:CORRections:STATe` is set to `ON`, error -221, “Settings conflict” occurs.
- If `INIT:CONT` is not set to `OFF`, error -221, “Settings conflict” occurs.

# SERVice:SENSor[1]|2:CDATe?

This query returns the calibration date in E-series sensors. Calibration date information is stored in the sensor's EEPROM.

### Syntax




### Example

| | |
|---|---|
| `SERV:SENS2:CDATe?` | *This query returns the calibration date of the E-series sensor connected to channel B.* |

### Error Messages

- If no power sensor is connected, error -241 "Hardware missing" occurs.
- If a a sensor other than a P-series or E-series power sensor is connected, error -241 "Hardware missing" occurs.

# SERVice:SENSor[1]|2:CORRections:STATe <boolean>

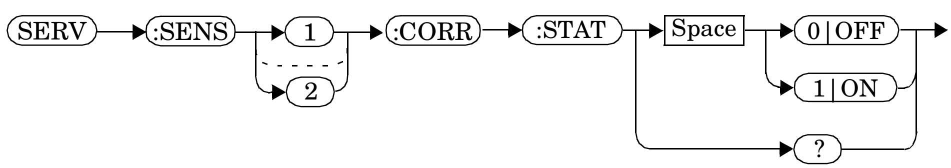
This command enables/disables the voltage to corrected power conversion. It applies to E9320 series power sensors only.

**NOTE**

Before setting this command to OFF, you must set the INIT:CONF command to OFF.

After setting this command to OFF, you must only run commands relating to the gathering of ADC values—for example, the SERV:SENS:RADC command.

### Syntax




### Example

| | |
|---|---|
| `SERV:SENS2:CORR:STAT ON` | *This command enables the voltage to corrected power conversion for channel B.* |

**Reset Condition**

On reset, the value is set to OFF.

**Query**

`SERVice:SENSor[1]|2:CORRections:STATe?`

The query enters a `1` or `0` into the output buffer indicating the status of the voltage to corrected power conversion.

- `1` is returned when voltage to corrected power conversion is enabled.
- `0` is returned when voltage to corrected power conversion is disabled.

### Query Example

| | |
|---|---|
| SERV:SENS:CORR:STAT? | *This command queries whether voltage to corrected power conversion is enabled for channel A.* |

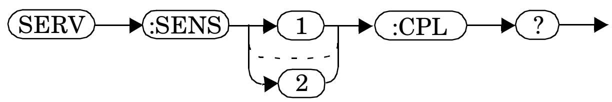
### Error Messages

- If INIT:CONT is not set to off, error -221, “Settings conflict” occurs.
- If the command is used when a sensor other than an E9320 power sensor is connected, error -241, “Hardware missing” occurs

# SERVice:SENSor[1]|2:CPLace?

This query returns the calibration place in E-series sensors. Calibration place information is stored in the sensor's EEPROM.

### Syntax




### Example

| | |
|---|---|
| `SERV:SENS2:CPL?` | *This query returns the place of calibration of the E-series sensor connected to channel B.* |

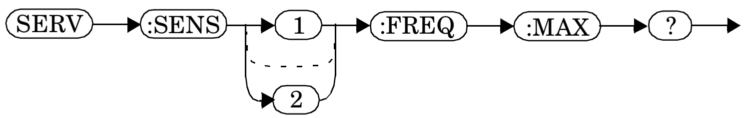
### Error Messages

- If no power sensor is connected, error -241 "Hardware missing" occurs.
- If a sensor other than a P-series or E-series power sensor is connected, error -241 "Hardware missing" occurs.

# SERVice:SENSor[1]|2:FREQuency:MAXimum?

This query returns the maximum frequency that can be measured by the currently connected sensor. It is applicable to E-series sensors only. Maximum frequency information is stored in the sensor's EEPROM.

### Syntax




### Example

| | |
|---|---|
| SERV:SENS2:FREQ:MAX? | This query returns the maximum frequency that can be measured by the E-series sensor currently connected to channel B. |

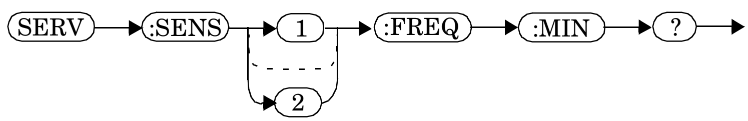
### Error Messages

- If no sensor is connected, error -241, "Hardware missing" occurs.
- If a a sensor other than an E-series power sensor is connected, error -241 "Hardware missing" occurs.
- If the sensor, currently connected, does not contain the necessary information in EEPROM, error -241 "Hardware missing" occurs.

# SERVice:SENSor[1]|2:FREQuency:MINimum?

This query returns the minimum frequency that can be measured by the currently connected sensor. It is applicable to E-series sensors only. Minimum frequency information is stored in the sensor's EEPROM.

### Syntax




### Example

| | |
|---|---|
| `SERV:SENS1:FREQ:MIN?` | *This query returns the minimum frequency that can be measured by the E-series sensor currently connected to channel A.* |

### Error Messages

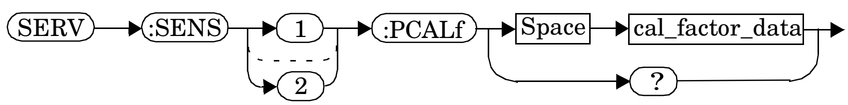
- If no sensor is connected, error -241, "Hardware missing" occurs.
- If a a sensor other than an E-series power sensor is connected, error -241 "Hardware missing" occurs.
- If the E-series sensor currently connected does not contain the necessary information in EEPROM, error -241 "Hardware missing" occurs.

# SERVice:SENSor[1]|2:PCALfactor <cal_factor_data>

This command writes calibration factor data to, or reads calibration factor data from, the currently connected sensor. The whole calibration factor block must be written at once as a checksum is generated. The new block must not be larger than the existing block.

This command applies to E9320 series sensors for peak path data only. For E4410 series, E9300 series and E9320 series sensors, average path data, refer to `SERVice:SENSor[1]|2:CALFactor <cal_factor_data>`.

### Syntax




### Parameters

| Item | Description/Default |
|---|---|
| cal_factor_data | A binary data block. Refer to Appendix , “Calibration Factor Block Layout,” on page 566 for further information. |

### Query

`SERVice:SENSor[1]|2:PCALfactor?`

The query returns the current peak path calibration factor block.

### Query Example

| | |
|---|---|
| `SERV:SENS:PCALf?` | *This command returns the peak path calibration factor block for channel A.* |

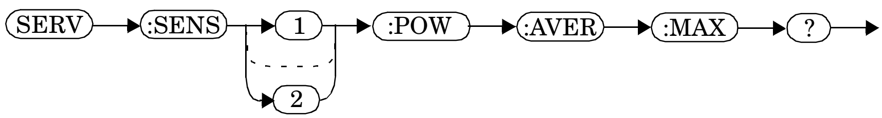
### Error Messages

- If no power sensor is connected, error -241 “Hardware missing” occurs.
- If a a sensor other than an E9320 power sensor is connected, error -241 “Hardware missing” occurs.
- If `INIT:CONT` is not set to `OFF`, error -221, “Settings conflict” occurs.

# SERVice:SENSor[1]|2:POWer:AVERage:MAXimum?

This query returns the maximum average power that can be measured by the currently connected sensor. It is applicable to E-series sensors only. Maximum average power information is stored in the sensor's EEPROM.

### Syntax




### Example

| | |
|---|---|
| `SERV:SENS:POW:AVER:MAX?` | *This query returns the maximum average power that can be measured by the E-series sensor currently connected to channel A.* |

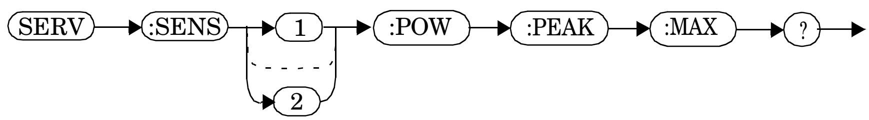
### Error Messages

- If no sensor is connected, error -241, “Hardware missing” occurs.
- If a a sensor other than an E-series power sensor is connected, error -241 “Hardware missing” occurs.
- If the E-series sensor currently connected does not contain the necessary information in EEPROM, error -241 “Hardware missing” occurs.

## SERVice:SENSor[1]|2:POWer:PEAK:MAXimum?

This query returns the maximum peak power that can be measured by the currently connected sensor. It is applicable to E-series sensors only. Maximum peak power information is stored in the sensor's EEPROM.

### Syntax




### Example

| | |
|---|---|
| `SERV:SENS2:POW:PEAK:MAX?` | *This query returns the maximum peak power that can be measured by the E-series sensor currently connected to channel B.* |

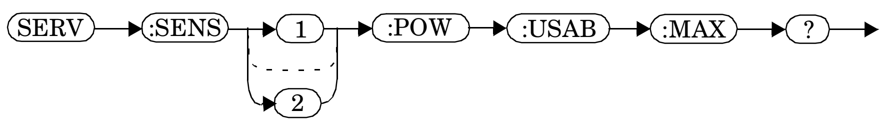
### Error Messages

- If no sensor is connected, error -241, "Hardware missing" occurs.
- If a a sensor other than an E-series power sensor is connected, error -241 "Hardware missing" occurs.
- If the E-series sensor currently connected does not contain the necessary information in EEPROM, error -241 "Hardware missing" occurs.

# SERVice:SENSor[1]|2:POWer:USABle:MAXimum?

This query returns the maximum power that can be accurately measured by the currently connected sensor. It is applicable to E-series sensors only. Maximum power information is stored in the sensor's EEPROM.

### Syntax




### Example

| | |
|---|---|
| `SERV:SENS1:POW:USAB:MAX?` | *This query returns the maximum power that can be accurately measured by the E-series sensor currently connected to channel A.* |

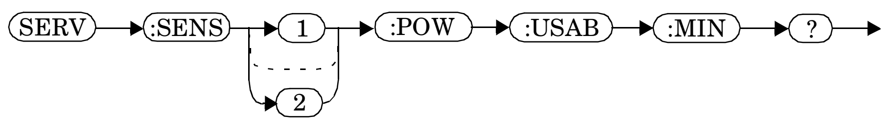
### Error Messages

- If no sensor is connected, error -241, "Hardware missing" occurs.
- If a a sensor other than an E-series power sensor is connected, error -241 "Hardware missing" occurs.
- If the E-series sensor currently connected does not contain the necessary information in EEPROM, error -241 "Hardware missing" occurs.

# SERVice:SENSor[1]|2:POWer:USABle:MINimum?

This query returns the minimum power that can be accurately measured by the currently connected sensor. It is applicable to E-series sensors only. Maximum power information is stored in the sensor's EEPROM.

### Syntax




### Example

| | |
|---|---|
| `SERV:SENS:POW:USAB:MIN?` | *This query returns the minimum power that can be accurately measured by the E-series sensor currently connected to channel A.* |

### Error Messages

- If no sensor is connected, error -241, "Hardware missing" occurs.
- If a a sensor other than an E-series power sensor is connected, error -241 "Hardware missing" occurs.
- If the E-series sensor currently connected does not contain the necessary information in EEPROM, error -241 "Hardware missing" occurs.

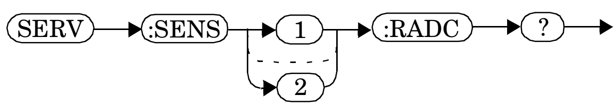
# SERVice:SENSor[1]|2:RADC?

This query returns a new raw uncorrected measurement in volts, as a 32 bit signed integer.

**NOTE**

For E9320 series sensors:
before running this query, the voltage to corrected power conversion must be disabled using the `SERVice:SENSor[1]|2:CORRections:STATe` command.

### Syntax




### Example

| | |
|---|---|
| `SERV:SENS2:RADC?` | *This query returns a new raw uncorrected measurement for the sensor connected to channel B.* |

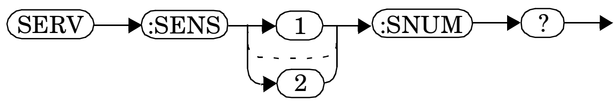
### Error Messages

- If `INIT:CONT` is set to `ON`, error -221 “Settings Conflict” occurs.
- If an E9320 series sensor is connected and `SERVice:SENSor[1]|2:CORRections:STATe` is set to `ON`, error -221 “Settings Conflict” occurs.

# SERVice:SENSor[1]|2:SNUMber?

This query returns the serial number for E-series sensors. Serial number information is stored in the sensor's EEPROM.

### Syntax




### Example

| | |
|---|---|
| `SERV:SENS2:SNUM?` | *This query returns the serial number of the E-series sensor connected to channel B.* |

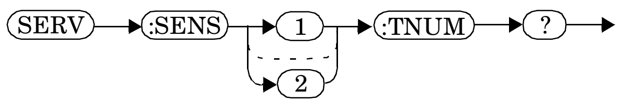
### Error Messages

- If no sensor is connected, error -241, "Hardware missing" occurs.
- If a a sensor other than a P-series or E-series power sensor is connected, error -241 "Hardware missing" occurs.

# SERVice:SENSor[1]|2:TNUMber?

This query returns the tracking number for E-series sensors. Tracking number information is stored in the sensor's EEPROM.

### Syntax




### Example

| | |
|---|---|
| `SERV:SENS2:TNUM?` | *This query returns the serial number of the E-series sensor connected to channel B.* |

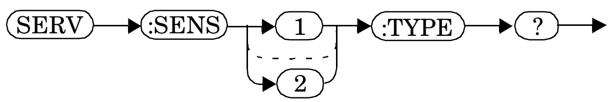
### Error Messages

- If no sensor is connected, error -241, "Hardware missing" occurs.
- If a a sensor other than a P-series or E-series power sensor is connected, error -241 "Hardware missing" occurs.

# SERVice:SENSor[1]|2:TYPE?

This query identifies the sensor type connected to the power meter input channel(s). For Agilent 8480 series sensors, either “A”, “B”, “D”, or “H” is returned. For E-series sensors, the model number stored in EEPROM is returned

### Syntax




### Example

`SERV:SENS2:TYPE?`

*This query returns either, “A”, “B”, “D”, or “H” if an Agilent 8480 series sensor is connected to channel B, or the sensor model number if an E-series sensor is connected to channel B.*

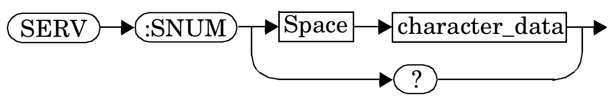
### Error Messages

- If no sensor is connected, error -241, “Hardware missing” occurs.

# SERVice:SNUMber <character_data>

This command loads the power meter with a serial number in the form GB12345678 or US12345678.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | Details the power meter serial number in the form GB12345678 or US12345678. A maximum of 30 characters can be used. | A to Z (uppercase)<br>a to z (lowercase)<br>0 - 9 |

### Example

`SERV:SNUM GB12345678` *This command loads the power meter with the serial number GB12345678.*

### Query

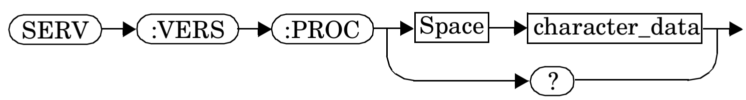
`SERVice:SNUMber?`

The query returns the power meter serial number in the form GB12345678 or US12345678.

# SERVice:VERSion:PROCessor <character_data>

This command loads the power meter with the processor board revision version.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | Details the processor board revision version. A maximum of 20 characters can be used. | A to Z (uppercase)<br>a to z (lowercase)<br>0 - 9<br>_ (underscore) |

### Example

| | |
|---|---|
| `SERV:VERS:PROC "C"` | *This command loads the power meter with processor board revision version C.* |

### Query

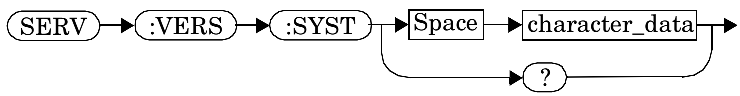
`SERVice:VERSion:PROCessor?`

The query returns the current processor board revision version.

# SERVice:VERSion:SYSTem <character_data>

This command loads the power meter with the system version number.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| character_data | Details the system version number. A maximum of 20 characters can be used. | A to Z (uppercase)<br>a to z (lowercase)<br>0 - 9<br>_ (underscore) |

### Example

`SERV:VERS:SYST “1”` *This command loads the power meter with system version number 1.*

### Query

`SERVice:VERSion:SYSTem?`

The query returns the current power meter system version number.

# 16 Command Reference

# SCPI Compliance Information

This chapter contains information about the SCPI Common (*) Commands that the power meter supports. It also describes the GPIB Universal Command statements which form the nucleus of GPIB programming; they are understood by all instruments in the network. When combined with programming language codes, they provide all management and data communication instructions for the system.

The IEEE-488.2 Common Command descriptions are listed below in alphabetical order.

# *CLS

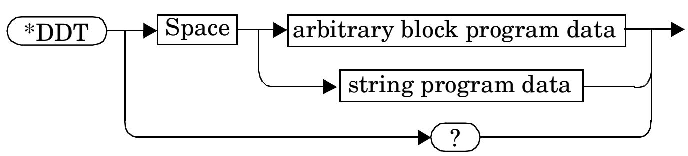
The `*CLS` (CLear Status) command clears the status data structures. The SCPI registers (Questionable Status, Operation Status and all the other SCPI registers), the Standard Event Status Register, the Status Byte, and the Error/Event Queue are all cleared.

### Syntax

# *DDT <arbitrary block program data> | <string program data>

The `*DDT` (Define Device Trigger) command determines the power meter's response to a `GET` (Group Execute Trigger) message or `*TRG` common command. This command effectively turns `GET` and `*TRG` into queries, with the measured power being returned.

### Syntax




### Parameters

| Type | Description | Range of Values |
|---|---|---|
| arbitrary block program data | The command which is executed on a GET or *TRG. | #nN<action>[a,b] |
| string program data | | "<action>"[a] |

a. The <action> field of the parameter may contain:

`FETC?`
`FETC1?`
`FETC2?` (N1912A only)
`*TRG`
`TRIG1`
`TRIG2` (N1912A only)

b. The first digit after the # indicates the number of following digits. The following digits indicate the length of the data.

Examples of <arbitrary block program data> parameters are:

- `#15FETC?` and `#206FETCh?`

Examples of <string program data> are:

- `"FETCh1?"`, `"FETCh?"` and `"TRIG1;FETC1"`

**Reset Condition** On reset, the <action> field of `*DDT` is set to `*TRG`.

**Query** `*DDT?`

The query returns the action which is performed on receipt of a `GET` or `*TRG`. This is returned as a <definite length arbitrary block response data> value which is in the form of #nN<action> as described on page 528 .

**Error Message**

- If an invalid parameter is received, error -224, “Illegal parameter value” occurs.

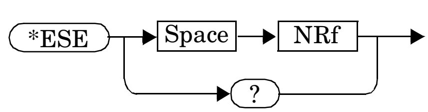
# *ESE <NRf>

The *ESE (Event Status Enable) <NRf> command sets the Standard Event Status Enable Register. This register contains a mask value for the bits to be enabled in the Standard Event Status Register. A 1 in the Enable Register enables the corresponding bit in the Status Register, a 0 disables the bit. The parameter value, when rounded to an integer and expressed in base 2, represents the bit values of the Standard Event Status Enable Register. Table 16-1 shows the contents of this register.

**Table 16-1** ***ESE Mapping**

| Bit | Weight | Meaning |
|---|---|---|
| 0 | 1 | Operation Complete |
| 1 | 2 | Request Control (not used) |
| 2 | 4 | Query Error |
| 3 | 8 | Device Dependent Error |
| 4 | 16 | Execution Error |
| 5 | 32 | Command Error |
| 6 | 64 | Not used |
| 7 | 128 | Power On |

### Syntax




### Parameters

| Type | Description/Default | Range of Values |
|---|---|---|
| NRf | A value used to set the Standard Event Status Enable Register. | 0 - 255 |

### Query

*ESE?

The query returns the current contents of the Standard Event Status Enable Register. The format of the return is `<NR1>` in the range of 0 to 255.

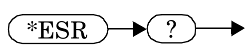
# *ESR?

The `*ESR?` query returns the contents of the Standard Event Status Register then clears it. The format of the return is `<NR1>` in the range of 0 to 255. Table 16-2 shows the contents of this register.

**Table 16-2** ***ESR? Mapping**

| Bit | Weight | Meaning |
|---|---|---|
| 0 | 1 | Operation Complete |
| 1 | 2 | Request Control (not used) |
| 2 | 4 | Query Error |
| 3 | 8 | Device Dependent Error |
| 4 | 16 | Execution Error |
| 5 | 32 | Command Error |
| 6 | 64 | Not used |
| 7 | 128 | Power On |

### Syntax



# *IDN?

The `*IDN?` query allows the power meter to identify itself. The string returned is either:

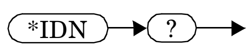
`Agilent Technologies,N1911A,<serial number>,A1.XX.YY`

`Agilent Technologies,N1912A,<serial number>,A2.XX.YY`

where:

- `<serial number>` uniquely identifies each power meter.
- `A1.XX.YY` and `A2.XX.YY` represents the firmware revision with `XX` and `YY` representing the major and minor revisions respectively.

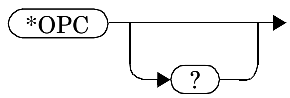
## Syntax



# *OPC

The `*OPC` (OPeration Complete) command causes the power meter to set the operation complete bit in the Standard Event Status Register when all pending device operations have completed.

**Syntax**




**Query**

`*OPC?`

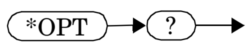
The query places an ASCII 1 in the output queue when all pending device operations have completed.

# *OPT?

The `*OPT?` query reports the options installed in the power meter and returns:

- " " empty string for a standard instrument.
- "003" for an option 003 instrument.

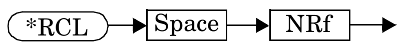
### Syntax



# *RCL <NRf>

The `*RCL <NRf>` (ReCaLl) command restores the state of the power meter from the specified save/recall register. An instrument setup must have been stored previously in the specified register.

**Syntax**




**Parameters**

| Type | Description/Default | Range of Values |
|---|---|---|
| NRf | The number of the register to be recalled. | 1 - 10 |

**Error Message**

- If the register does not contain a saved state, error -224, “Illegal parameter value” occurs.

# *RST

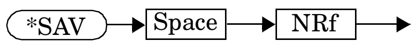
The *RST (ReSeT) command places the power meter in a known state. Refer to “SYSTem:PRESet <character_data>” on page 368 for information on reset values.

### Syntax

# *SAV <NRf>

The `*SAV <NRf>` (SAVe) command stores the current state of the power meter in the specified register.

### Syntax




### Parameters

| Item | Description/Default | Range of Values |
|---|---|---|
| `NRf` | The number of the register that the current state of the power meter is to be saved to. | 1 - 10 |

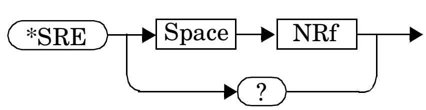
# *SRE <NRf>

The `*SRE <NRf>` command sets the Service Request Enable register bits. This register contains a mask value for the bits to be enabled in the Status Byte Register. A `1` in the Enable Register enables the corresponding bit in the Status Byte Register; a `0` disables the bit. The parameter value, when rounded to an integer and expressed in base 2, represents the bits 0 to 5 and bit 7 of the Service Request Enable Register. Bit 6 is always 0. Table 16-3 shows the contents of this register. Refer to the pullout at the end of Chapter 10 for further information.

**Table 16-3** ***SRE Mapping**

| Bit | Weight | Meaning |
|---|---|---|
| 0 | 1 | Not used |
| 1 | 2 | Not used |
| 2 | 4 | Device Dependent |
| 3 | 8 | QUEStionable Status Summary |
| 4 | 16 | Message Available |
| 5 | 32 | Event Status Bit |
| 6 | 64 | Not used |
| 7 | 128 | OPERation Status Summary |

### Syntax




### Parameters

| Type | Description/Default | Range of Values |
|---|---|---|
| `NRf` | A value used to set the Service Request Enable Register. | 0 - 255 |

**Query**

`*SRE?`

The query returns the contents of bits 0 to 5 and bit 7 of the Service Request Enable Register. The format of the return is `<NR1>` in the ranges of 0 to 63 or 128 to 191 (that is, bit 6 is always 0).

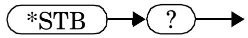
# *STB?

The `*STB?` (STatus Byte) query returns bit 0 to 5 and bit 7 of the power meter's status byte and returns the Master Summary Status (MSS) as bit 6. The MSS is the inclusive OR of the bitwise combination (excluding bit 6) of the Status Byte and the Service Request Enable registers. The format of the return is `<NR1>` in the ranges of 0 to 255. Table 16-4 shows the contents of this register. Refer to the pullout at the end of Chapter 10 for further information.

**Table 16-4 *STB? Mapping**

| Bit | Weight | Meaning |
|---|---|---|
| 0 | 1 | Not used |
| 1 | 2 | Device Dependent<br>0 - No device status conditions have occurred<br>1 - A device status condition has occurred |
| 2 | 4 | Error/Event Queue<br>0 - Queue empty<br>1 - Queue not empty |
| 3 | 8 | Questionable Status Summary<br>0 - No QUEStionable status conditions have occurred<br>1 - A QUEStionable status condition has occurred |
| 4 | 16 | Message Available<br>0 - no output messages are ready<br>1 - an output message is ready |
| 5 | 32 | Event Status Bit<br>0 - no event status conditions have occurred<br>1 - an event status condition has occurred |
| 6 | 64 | Master Summary Status<br>0 - power meter not requesting service<br>1 - there is at least one reason for requesting service |
| 7 | 128 | Operation Status Summary<br>0 - No OPERation status conditions have occurred<br>1 - An OPERation status condition has occurred |

### Syntax


*STB
?

# *TRG

The *TRG (TRiGger) command triggers all channels that are in the wait for trigger state. It has the same effect as Group Execute Trigger (GET).

Using the *DDT command may change the function of the *TRG command.

**Syntax**

*TRG →

**Error Message**

- If TRIGger:SOURce is not set to BUS, error -211, “Trigger ignored” occurs.
- If the power meter is not in the wait-for-trigger state, error -211, “Trigger ignored” occurs.

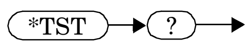
# *TST?

The `*TST?` (TeST) query causes the power meter to perform the self test. The test takes approximately 30 seconds.

The result of the test is placed in the output queue.

- `0` is returned if the test passes.
- `1` if the test fails.

## Syntax



# *WAI

The `*WAI` (WAIt) command causes the power meter to wait until either:

- All pending operations are complete.
- The device clear command is received.
- Power is cycled.

before executing any subsequent commands or queries.

## Syntax

# GPIB Universal Commands

## DCL

The DCL (Device Clear) command causes all GPIB instruments to assume a cleared condition. The definition of device clear is unique for each instrument. For the power meter:

- All pending operations are halted, that is, *OPC? and *WAI.
- The parser (the software that interprets the programming codes) is reset and now expects to receive the first character of a programming code.
- The output buffer is cleared.

## GET

The GET (Group Execute Trigger) command triggers all channels that are in the "wait-for-trigger" state.

Using the *DDT command may change the function of the GET command.

### Error Message

If TRIGger:SOURce is not set to BUS, an error -211, "Trigger ignored" occurs.

If the power meter is not in the "wait-for-trigger" state then error -211, "Trigger ignored" occurs.

## GTL

The GTL (Go To Local) command is the complement to remote. It causes the power meter to return to local control with a fully enabled front panel. When reverting to local mode the power meter triggering is set to free run.

## LLO

The LLO (Local Lock Out) command can be used to disable the front panel local key. With this key disabled, only the controller (or a hard reset by the line power switch) can restore local control.

## PPC

When addressed to listen, the PPC (Parallel Poll Configure) command causes the power meter to be configured according to the parallel poll enable secondary command which should follow this command.

## PPD

Sending the PPC command followed by the PPD (Parallel Poll Disable) command disables the power meter from responding to a parallel poll. This is effectively a selective disable.

**Table 16-5 PPD Mapping**

| Bit | Weight | Meaning |
|---|---|---|
| 0 | 1 | Always 0 |
| 1 | 2 | Always 0 |
| 2 | 4 | Always 0 |
| 3 | 8 | Always 0 |
| 4 | 16 | Always 1 |
| 5 | 32 | Always 1 |
| 6 | 64 | Always 1 |
| 7 | 128 | Always 0 |

## PPE

Once the power meter has received a `PPC` command, the `PPE` (Parallel Poll Enable) secondary command configures the power meter to respond to a parallel poll on a particular data line with a particular level.

**Table 16-6 PPE Mapping**

<table>
<tr><th>Bit</th><th>Weight</th><th>Meaning</th></tr>
<tr><td>0</td><td>1</td><td rowspan="3">Bit positions for response:<br>000 (bit 0), 001 (bit 1), 010 (bit 2), 011 (bit 3), 100 (bit 4), 101 (bit 5), 110 (bit 6), 111 (bit 7)</td></tr>
<tr><td>1</td><td>2</td></tr>
<tr><td>2</td><td>4</td></tr>
<tr><td>3</td><td>8</td><td>Sense bit<br>0 - response bit is cleared during a parallel poll if requesting service.<br>1 - response bit is set during a parallel poll if requesting service.</td></tr>
<tr><td>4</td><td>16</td><td>Always 0</td></tr>
<tr><td>5</td><td>32</td><td>Always 1</td></tr>
<tr><td>6</td><td>64</td><td>Always 1</td></tr>
<tr><td>7</td><td>128</td><td>Always 0</td></tr>
</table>

## PPU

The `PPU` (Parallel Poll Unconfigure) command disables the power meter from responding to a parallel poll. This is effectively a universal disable.

## SDC

The SDC (Selected Device Clear) command causes instruments using GPIB in the listen state, to assume a cleared condition. The definition of a selected device clear is unique for each instrument. For the power meter:

- All pending operations are halted, that is, *OPC? and *WAI.
- The parser (the software that interprets the programming codes) is reset and now expects to receive the first character of a programming code.
- The output buffer is cleared.

## SPD

The SPD (Serial Poll Disable) command terminates the serial poll mode for the power meter and returns it to its normal talker state where device dependent data is returned rather than the status byte.

## SPE

The SPE (Serial Poll Enable) command establishes the serial poll mode for the power meter. When the power meter is addressed to talk, a single eight bit status byte is returned.