Keysight U8903B Audio Analyzer

Performance audio analyzer

KEYSIGHT TECHNOLOGIES

Programmer's Reference

Notices

© Keysight Technologies 2014

No part of this manual may be reproduced in any form or by any means (including electronic storage and retrieval or translation into a foreign language) without prior agreement and written consent from Keysight Technologies as governed by United States and international copyright laws.

Manual Part Number

U8903-90052

Edition

Edition 1, November 2014

Printed in Malaysia

Keysight Technologies
Bayan Lepas Free Industrial Zone,
11900 Penang, Malaysia

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, KEYSIGHT 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. KEYSIGHT 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 KEYSIGHT 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.

Technology Licenses

The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license.

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 Keysight 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.

Safety Notices

CAUTION

A CAUTION notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in damage to the product or loss of important data. Do not proceed beyond a CAUTION notice until the indicated conditions are fully understood and met.

WARNING

A WARNING notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in personal injury or death. Do not proceed beyond a WARNING notice until the indicated conditions are fully understood and met.

Table of Contents

12 OUTPut Subsystem

14 SOURce Subsystem

20 Programming Examples

A Appendixes

List of Figures

THIS PAGE HAS BEEN INTENTIONALLY LEFT BLANK.

List of Tables

THIS PAGE HAS BEEN INTENTIONALLY LEFT BLANK.

1 U8903B Remote Operation

This chapter provides an overview on programming the U8903B Audio Analyzer.

Introduction to the SCPI Language

Standard Commands for Programmable Instruments (SCPI) is an ASCII-based instrument command language designed for test and measurement instruments. SCPI commands are based on a hierarchical structure, also known as a tree system. In this system, associated commands are grouped together under a common node or root, thus forming subsystems. A portion of the `SOURce` subsystem is shown below to illustrate the tree system.

```
SOURce:
   VOLTage:
      RATio <ratio>,(@<channel>)
```

`SOURce` is the root keyword of the command, `VOLTage` is the second-level keyword, and `RATio` is the third-level keyword. A colon ":" separates a command keyword from a lower-level keyword.

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 upper-case characters, and the rest of the keyword shown in lower-case characters. For example, the short form of `VOLTage` is `VOLT`.

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.

```
:TIM:REFC ON;:TIM:REF RIGH
```

```
:TIM:REFC ON;REF RIGH
```

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 queries (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 queries 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 without reading the response.

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 interface standard. The “*” commands are used to control the reset, clear status, identification request, self-test, wait-before-execution, and status operations in the U8903B.

Syntax conventions

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

- Square brackets “[]” indicate optional keywords or parameters and can be omitted. The brackets are not sent with the command string. If you do not specify a value for an optional parameter, the instrument chooses a default value.
- Braces “{}” enclose one or more parameters that may be included zero or more times. The brackets are not sent with the command string.
- Triangle brackets “<>” indicate that you must substitute a value for the enclosed parameter. The brackets are not sent with the command string.
- Parenthesis “()” enclose parameters that are usually a channel list.
- Vertical bars “|” can be read as “or” and are used to separate alternative parameter options.

SCPI parameter types

The SCPI language defines different parameter formats for use in program messages and response messages.

Numeric

Commands that require parameters to accept all commonly used decimal representations of numbers including optional signs, decimal points, and scientific notation. You can also send engineering unit suffixes with numeric parameters such as MHz or kHz.

Discrete

Parameters used to program settings that have a limited number of values such as `BUS`, `IMMediate`, and `EXTernal`. They have a short form and a long form just like command keywords. You can mix upper-case and lower-case letters. Query responses will always return the short form in all upper-case letters.

Boolean

Parameters that represent a single binary condition that is either true or false. For a false condition, the U8903B will accept `OFF` or `0`. For a true condition, the U8903B will accept `ON` or `1`. When you query a boolean setting, the U8903B will always return `0` or `1`.

String

Parameters that contain virtually any set of ASCII characters. A string must begin and end with matching quotes, either with a single quote or a double quote. You can include the quote delimiter as part of the string by typing it twice without any characters in between.

Block

Parameter that allows binary data (including extended ASCII codes) to be transmitted as a sequence of bytes. This is more efficient than the text format when transferring large amounts of data. Either definite length or indefinite length arbitrary data may be transmitted or returned.

Remote Interface Configurations

This section describes how to configure the GPIB (IEEE-488), LAN, and USB remote interfaces.

NOTE

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

You can choose to control the U8903B remotely using the GPIB, LAN, or USB interfaces.

GPIB interface

Each device on the GPIB interface must have a unique address. You can set the U8903B address to any value between 0 and 30. The U8903B is shipped with a default address of 28. The GPIB address is stored in nonvolatile memory, and it does not change when the U8903B is switched off or after a remote interface reset.

The GPIB bus controller has its own address. Avoid using the bus controller address for any instrument on the interface bus. Keysight controllers generally use the address of 21.

Use the following command to set the GPIB address from the remote interface.

```
SYSTem:COMMunicate:GPIB:ADDRess
```

Use the following command to query the GPIB address from the remote interface.

```
SYSTem:COMMunicate:GPIB:ADDRess?
```

LAN interface

The U8903B supports three LAN operating modes as follows.

- Dynamic IP (Dynamic Host Configuration Protocol or DHCP)
- Auto IP (local PC control or isolated LAN)
- Static IP (manual configuration)

Configuring the LAN remotely

The IP address, subnet mask, and default gateway can be changed manually or remotely. To remotely specify the LAN settings, use the following commands.

- IP address: **SYSTem:COMMunicate:LAN:ADDRess**
- Subnet mask: **SYSTem:COMMunicate:LAN:SMASk**
- Default gateway: **SYSTem:COMMunicate:LAN:DGATeway**

The 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 set 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[:NEXT]?** command.

The LAN settings are stored in nonvolatile memory.

USB interface

The USB interface does not require front panel or remote configuration. The USB address cannot be changed as it is set at the factory and is unique for each U8903B.

NOTE

- Before connecting the USB cable, make sure that the Keysight IO Libraries software is installed on your PC.
- For more information on the Keysight IO Libraries software, refer to the *Keysight USB/LAN/GPIB Interfaces Connectivity Guide*. If you have installed other I/O software, refer to the documentation that accompanies the software.

SCPI Status System

This section describes the structure of the SCPI status system used by the U8903B. Each register group is made up of several low-level registers called Condition register, Event register, Enable register, Standard Operation register, Questionable Status registers, Standard Event register, and Status Byte register which control the action of specific bits within the register group.

Condition register

A condition register continuously monitors the state of the U8903B. The bits in the condition register are updated in real time and the bits are not latched or buffered. This is a read-only register and the bits are not cleared when you read the register. A query of the condition register returns a decimal value which corresponds to the binary-weighted sum of all bits set in that register.

Event register

An event register latches the various events from the changes in the condition register. There is no buffering in this register; while an event bit is set, subsequent events corresponding to that bit are ignored. This is a read-only register. Once a bit is set, it remains set until cleared by a query or clear status (***CLS**) command. A query of this register returns a decimal value which corresponds to the binary-weighted sum of all bits set in that register.

Enable register

An enable register defines which bits in the event register will be reported to the Status Byte register group. You can write to or read from an enable register. A clear status (***CLS**) command will not clear the enable register but it clears all bits in the event register. To enable bits in the enable register to be reported to the Status Byte register, you must write a decimal value which corresponds to the binary-weighted sum of the corresponding bits.

Standard Operation register

This register record signals that occur during normal operation. The outputs of the Standard Operation register are logically-ORed into the `OPERation` summary bit (7) of the Status Byte register.

Table 1-1 Bit definitions: Standard Operation register

Bit number		Decimal value	Definition
0	Not used	1	0 is returned.
1	Recording to a wave file	2	Recording of signal to a wave file is in progress.
2	Measuring in progress for digital analyzer	4	The U8903B is initiated, and is making, or about to make a measurement.
3	Sweeping in progress	8	The U8903B is performing sweep.
4	Measuring in progress for analog analyzer	16	The U8903B is initiated, and is making, or about to make a measurement.
5	Waiting for trigger	32	The U8903B is waiting for an external or bus trigger.
8	Generator setup in progress	256	The U8903B is configuring the generator.
6–15	Not used	Not used	0 is returned.

The **STATus:PRESet** command will clear all bits in the Negative-Transition (NTR) and enable registers.

Questionable Status registers

These registers record signals that indicate abnormal operation. The Questionable Data and Questionable Voltage registers are used for the U8903B. The outputs of the Questionable Voltage register are logically-ORed into the Voltage Overload bit (0) of the Questionable Data register. The outputs of the Questionable Data register are logically-ORed into the QUEStionable summary bit (3) of the Status Byte register.

Table 1-2 Bit definitions: Questionable Data register

Bit number		Decimal value	Definition
0	Voltage overload	1	The voltage of one of the input signals is over the limit.
1–15	Not used	Not used	0 is returned.

Table 1-3 Bit definitions: Questionable Voltage register

Bit number		Decimal value	Definition
0	Channel 1 voltage overload	1	The voltage of channel 1 is over the limit.
1	Channel 2 voltage overload	2	The voltage of channel 2 is over the limit.
2	Channel 3 voltage overload	4	The voltage of channel 3 is over the limit.
3	Channel 4 voltage overload	8	The voltage of channel 4 is over the limit.
4	Channel 5 voltage overload	16	The voltage of channel 5 is over the limit.
5	Channel 6 voltage overload	32	The voltage of channel 6 is over the limit.
6	Channel 7 voltage overload	64	The voltage of channel 7 is over the limit.
7	Channel 8 voltage overload	128	The voltage of channel 8 is over the limit.
8–15	Not used	Not used	0 is returned.

Standard Event register

The Standard Event register reports the following types of instrument events: command syntax errors, command execution errors, device errors (self-test or calibration), query errors, or when the ***OPC** command is executed. All of these conditions can be reported in the Standard Event summary bit through the enable register.

Table 1-4 Bit definitions: Standard Event register

Bit number		Decimal value	Definition
0	Operation complete	1	All commands prior to and including ***OPC** have been executed.
1	Not used	Not used	0 is returned.
2	Query error	4	The U8903B tried to read the output buffer but it was empty. Or, a new command line was received before a previous query has been read. Or, both the input and output buffers are full.
3	Device-dependent error	8	A self-test, calibration, or other device-specific error has occurred.
4	Execution error	16	A command execution error occurred.
5	Command error	32	A command syntax error occurred.
6–7	Not used	Not used	0 is returned.

The event register in the Standard Event is cleared when:

– you execute the clear status (***CLS**) command

– you query the event register using the event status register (***ESR?**) command

The Standard Event enable register is cleared when you execute the `*ESE 0` command.

Status Byte register

The Status Byte register reports the conditions from the other status registers. Clearing an event register from one of the other registers will clear the corresponding bits in the Status Byte condition register.

Data that is waiting in the U8903B output buffer is immediately reported on the “Message Available” bit (bit 4).

Table 1-5 Bit definitions: Status Byte register

Bit number		Decimal value	Definition
0–1	Not used	Not used	0 is returned.
2	Error queue	4	One or more error messages are stored in the error queue.
3	Questionable Data summary	8	One or more bits are set in the Questionable Data register (bits must be enabled in the enable register).
4	Message available	16	Data is available in the U8903B output buffer.
5	Event Status Byte summary	32	One or more bits are set in the Standard Event register (bits must be enabled in the enable register).
6	Master Status summary	64	One or more bits are set in the Status Byte register (bits must be enabled in the enable register). Also used to indicate a Request for Service (RQS).
7	Standard Operation summary	128	One or more bits are set in the Standard Operation register (bits must be enabled in the enable register).

The Status Byte condition register will be cleared when:

– you execute the clear status (***CLS**) command

– you read the event register from one of the other registers, only the corresponding bits are cleared in the condition register

The Status Byte enable register is cleared when you execute the `*SRE 0` command.

Status system diagram

The U8903B uses the Operation, Questionable, Standard Event, and Status Byte register groups to record a variety of instrument conditions. The relationship between various registers in the U8903B SCPI status system is shown in **Figure 1-1**.

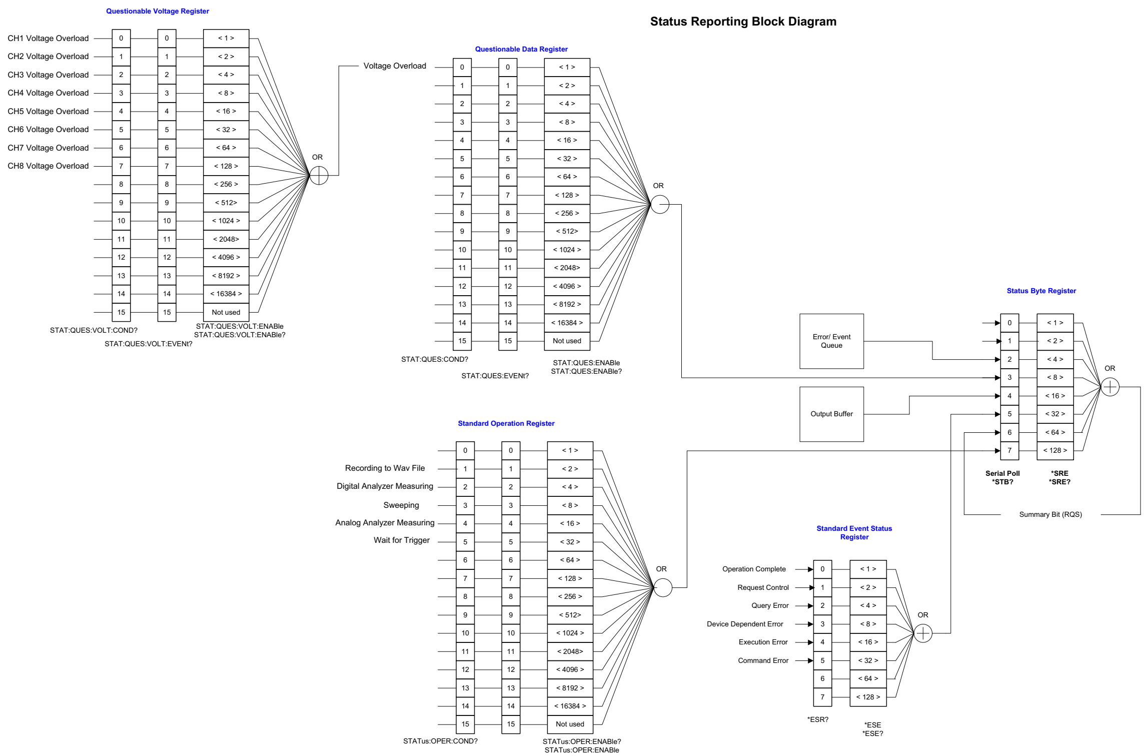


Figure 1-1 Status system diagram

Keysight U8903B
Audio Analyzer

Programmer's Reference

2 IEEE-488 Common Commands

This chapter describes the IEEE-488 common commands.

KEYSIGHT TECHNOLOGIES

*CLS

Syntax

```
*CLS
```

Description

Clears the event registers in all register groups, and also clears the error queue.

Example

The following command is used to clear all event registers and the error queue.

```
*CLS
```

*ESE

Syntax

```
*ESE <value>
*ESE?
```

Description

Sets the bits in the Standard Event enable register. The selected bits are then reported to bit 5 of the Status Byte register. The query reads the enable register and returns a decimal value which corresponds to the binary-weighted sum of all bits set in the register.

Parameter

Item	Type	Range of values	Default value
`<value>`	Numeric	A decimal value which corresponds to the binary-weighted sum of the bits in the register.	0

Remarks

- The bit definitions for the Standard Event register are listed in "**Standard Operation register**" on page 29.
- Use the `<value>` parameter to specify which bits will be enabled. The specified decimal value corresponds to the binary-weighted sum of the bits you wish to enable in the register. For example, to enable bit 2 (decimal value = 4), bit 3 (decimal value = 8), and bit 7 (decimal value = 128), the corresponding decimal value would be 140 (4 + 8 + 128).
- The clear status (***CLS**) command will not clear the enable register, but it clears all bits in the event register.
- The ***RST** or **SYSTem:PRESet** command does not affect the settings enabled by this command. However, cycling the U8903B power will reset this register to 0.

Examples

The following command enables bit 4 (decimal value = 16) in the enable register. If an Execution Error occurs, this condition will be reported to the Status Byte register (bit 5 will be set to high).

```
*ESE 16
```

The following query returns the bits set in the register.

```
*ESE?
```

Typical response:

```
16
```

*ESR?

Syntax

```
*ESR?
```

Description

Reads the event register of the Standard Event register group, and returns a decimal value which corresponds to the binary-weighted sum of all bits set in the register.

Remarks

- The bit definitions for the Standard Event register are listed in "**Standard Event register**" on page 30.
- Once a bit is set, it remains set until cleared by a clear status (***CLS**) command or queried by this command.

Example

The following query reads the event register (bits 3 and 4 are set).

```
*ESR?
```

Typical response:

```
24
```

*IDN?

Syntax

```
*IDN?
```

Description

Reads the U8903B identification string which contains four comma-separated fields. The first field is the manufacturer's name, the second field is the instrument model number, the third field is the serial number, and the fourth field is the firmware revision. This query returns an ASCII string with the following format.

<Manufacturer's_name>,<model_number>,<serial_number>,<firmware_revision>

Manufacturer:	Keysight Technologies
Instrument model number:	U8903B
Instrument serial number if available, or 0:	MYxxxxxxxx
Firmware revision levels:	x.x.x.x

Example

The following query returns the U8903B identification string.

```
*IDN?
```

Typical response:

```
KEYSIGHT TECHNOLOGIES,U8903B,MY00123456,1.0.0.0
```

*OPC

Syntax

```
*OPC
```

```
*OPC?
```

Description

Sets the “Operation Complete” bit (bit 0) in the Standard Event register when all pending operations have completed. This query sends 1 to the output buffer when all pending operations have completed.

Remark

This command is used to synchronize your application with the U8903B.

Examples

The following command sets the “Operation Complete” bit.

```
*OPC
```

The following query waits until the completion of the current command and then sends 1 to the output buffer.

```
*OPC?
```

Typical response:

```
1
```

*OPT?

Syntax

```
*OPT?
```

Description

Returns an ASCII string identifying the U8903B option configuration.

Remarks

This command is used to verify the installed option in the U8903B.

Example

The following query returns the U8903B installed option string.

```
*OPT?
```

Typical response:

```
“None”
```

*RST

Syntax

```
*RST
```

Description

Resets the U8903B to its startup state.

Remarks

- This command does not affect any user-defined files in the U8903B memory.
- The time taken to reset all settings for all channels of the U8903B is approximately 5 s. To reset the settings for only a particular mode such as the generator mode, use the **SYSTem:RESet[:MODE]** command.

Example

The following command resets the U8903B to its startup state.

```
*RST
```

*SRE

Syntax

```
*SRE <value>
*SRE?
```

Description

Enables the bits in the Status Byte enable register. The selected enabled bits are summarized in the “Master Summary” bit (bit 6) of the Status Byte register. If any of the selected bit condition changes from 0 to 1, a Service Request is generated. The query reads the enable register and returns a decimal value that corresponds to the binary-weighted sum of all bits set in the register.

Parameter

Item	Type	Range of values	Default value
`<value>`	Numeric	A decimal value which corresponds to the binary-weighted sum of the bits in the register.	0

Remarks

– The bit definitions for the Status Byte register are listed in “**Status Byte register**” on page 31.

– Use the `<value>` parameter to specify which bits to enable. The specified decimal value corresponds to the binary-weighted sum of the bits you wish to enable in the register. For example, to enable bit 2 (decimal value = 4) and bit 5 (decimal value = 32), the corresponding decimal value would be 36 (4 + 32).

– The **STATus:PRESet**, **SYSTem:PRESet**, ***CLS**, or ***RST** command does not clear the bits in the Status Byte enable register.

– Cycling the U8903B power will reset it to 0.

Examples

The following command enables bit 4 (decimal value = 16) in the enable register.

```
*SRE 16
```

The following query returns which bits are enabled in the register.

```
*SRE?
```

Typical response:

```
16
```

*STB?

Syntax

```
*STB?
```

Description

Queries the condition register for the Status Byte register, and returns a decimal value which corresponds to the binary-weighted sum of all bits set in the register. This query is similar to a Serial Poll, but it is processed like any other instrument command. This is a read-only register, and the bits are not cleared when you read the register.

Remarks

- The bit definitions for the Status Byte register are listed in the **Status Byte register**.
- This query returns the same results as a Serial Poll, but the “Master Summary” bit (bit 6) is not cleared if a Serial Poll has occurred.
- A power-on cycle will clear all bits in the condition register.

Example

The following query reads the condition register (bits 2 and 5 are set).

```
*STB?
```

Typical response:

```
36
```

*TRG

Syntax

```
*TRG
```

Description

This command is used in conjunction with the **TRIGger Subsystem** command to trigger the U8903B from the remote interface.

Remarks

- For analog module, use the **TRIGger Subsystem** command to select the BUS trigger source.
- For HP8903B mode, this command will trigger a T3 command. Refer to **Table A-9** for more information on the T3 command.

Example

The following command sequence is used to trigger the U8903B in the analog analyzer mode.

```
TRIG:ANAL:SOUR BUS
INIT:ANAL
*TRG
```

*TST?

Syntax

```
*TST?
```

Description

Initiates an internal self-test of the U8903B, and returns a pass or fail indication. The self-test runs a series of tests and will take approximately 30 s to complete.

Remarks

- If one or more tests fail, 1 is returned and the errors are stored in the error queue. For a complete listing of the error messages related to self-test failures, refer to "**Error messages**" on page 611. Use the **SYSTem:ERRor[:NEXT]?** command to read the error queue.
- If all tests pass, 0 is returned.

NOTE Do not operate the U8903B while the self-test is in progress as doing so might cause unexpected results.

Example

The following query performs a self-test and returns a pass or fail indication.

```
*TST?
```

Typical response:

```
0
```

*WAI

Syntax

```
*WAI
```

Description

The Wait-to-Continue (WAI) command causes the U8903B to wait until all pending operations have completed before executing any other command.

Example

The following command waits until all pending operations have completed.

```
*WAI
```

THIS PAGE HAS BEEN INTENTIONALLY LEFT BLANK.

3 ABORt Subsystem

This chapter describes the `ABORt` subsystem commands.

ABORt:ANALyzer

Syntax

```
ABORt:ANALyzer
```

Description

Cancels any initiated analyzer measurement trigger and measurement trigger for an array of graph points, and returns the trigger state to idle.

Remark

This command will abort all the initiated triggered channels.

Example

The following command aborts the initiated analyzer measurement trigger and measurement trigger for an array of graph points.

```
ABOR:ANAL
```

ABORt:GRAPh

Syntax

```
ABORt:GRAPh
```

Description

Cancels any initiated analyzer measurement trigger and measurement trigger for an array of graph points, and returns the trigger state to idle.

Remark

This command will abort all the initiated triggered channels.

Example

The following command aborts the initiated analyzer measurement trigger and measurement trigger for an array of graph points.

```
ABOR:GRAP
```

ABORt:SWEep

Syntax

```
ABORt:SWEep
```

Description

Cancels any initiated measurement trigger for sweep, and returns the trigger state to idle.

Remark

This command will abort all the initiated triggered channels.

Example

The following command aborts the initiated measurement trigger for sweep.

```
ABOR:SWE
```

ABORt:WAVFile:RECord

Syntax

```
ABORt:WAVFile:RECord
```

Description

Stops any initiated wave file recording.

Remark

This command will abort all the initiated wave file recording.

Example

The following command aborts the initiated wave file recording.

```
ABOR:WAVF:REC
```

ABORt:DIGital:ANALyzer

Syntax

```
ABORt:DIGital:ANALyzer
```

Description

Cancels any initiated digital analyzer measurement trigger and returns the trigger state to Idle.

Remark

This command will abort all the initiated triggered channels.

Examples

The following command aborts the initiated analyzer measurement trigger.

```
ABOR:DIG:ANAL
```

ABORt:DIGital:GRAPh

Syntax

```
ABORt:DIGital:GRAPh
```

Description

Cancels any initiated digital analyzer measurement trigger and measurement trigger for an array of graph points. Returns the trigger state to idle.

Remark

This command will abort all the initiated triggered channels.

Examples

The following command aborts the initiated digital analyzer measurement trigger and measurement trigger for an array of graph points.

```
ABOR:DIG:GRAP
```

ABORt:DIGital:BERT

Syntax

```
ABORt:DIGital:BERT
```

Description

Stops the Bit Error Rate Test (BERT).

Examples

The following command stops the initiated BERT.

```
ABOR:DIG:BERT
```

Keysight U8903B
Audio Analyzer

Programmer's Reference

4 CALCulate Subsystem

This chapter describes the `CALCulate` subsystem commands.

CALCulate[:ANALog]:HARMonic:AMPL?

Syntax

```
CALCulate[:ANALog]:HARMonic: AMPL  <value>
```

Description

Configures the value of the amplification factor to be applied in dB. The query returns the amplification factor value.

Parameter

Item	Type	Range of values	Default value
`<value>`	Numeric	-200 to 200	6

Remarks

– Only applicable for HAMP, HADI, and HAFU analysis mode.

Examples

The following command configures the harmonics amplification to 9 dB.

```
CALC:HARM:AMPL 9
```

The following query returns the value of harmonic amplification.

```
CALC:HARM:AMPL?
```

Typical response:

```
9
```

CALCulate[:ANALog]:HARMonic:COUNt

Syntax

```
CALCulate[:ANALog]:HARMonic:COUNt <count>
```

```
CALCulate[:ANALog]:HARMonic:COUNt?
```

Description

Sets the number of harmonic components in the frequency domain (MAGnitude) display. The query returns the number of harmonic components.

Parameter

Item	Type	Range of values	Default value
`<count>`	Numeric	1 to 8	8

Remarks

- The graph analysis mode must be set to the frequency domain (MAGnitude) using the **DISPlay:ANALysis:TYPE** command.
- This command is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command.

Examples

The following command sets eight harmonic components.

```
CALC:HARM:COUN 8
```

The following query returns the number of harmonic components.

```
CALC:HARM:COUN?
```

Typical response:

```
8
```

CALCulate[:ANALog]:HARMonic:FREQuencies?

Syntax

```
CALCulate[:ANALog]:HARMonic:FREQuencies? (@<channel>)
```

Description

Returns the harmonic frequency values for the specified channel. Multiple responses are separated by commas.

Parameter

Item	Type	Range of values	Default value
<channel>	Numeric	1 to 8	Required parameter

Remarks

- The graph analysis mode must be set to frequency domain (MAGnitude) using the **DISPlay:ANALysis:TYPE** command.
- This query is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command at the particular input channel.
- The number of harmonic frequency values returned is based on the harmonic count specified in the **CALCulate[:ANALog]:HARMonic:AMPL?** command.

Example

The following command sequence is used to obtain the harmonic frequency values of channel 2.

```
INIT:GRAP (@2)
CALC:HARM:FREQ? (@2)
```

Typical response:

```
9.91821E+02,2.00272E+03,2.99454E+03,4.00543E+03,4.99725E+03,
6.00815E+03,6.99997E+03,7.99179E+03
```

CALCulate[:ANALog]:HARMonic:FUNDamental?

Syntax

```
CALCulate[:ANALog]:HARMonic:FUNDamental? (@<channel>)
```

Description

Returns the signal harmonic fundamental frequency in Hz for the specified channel.

Parameter

Item	Type	Range of values	Default value
`<channel>`	Numeric	1 to 8	Required parameter

Remarks

- The graph analysis mode must be set to frequency domain (MAGnitude) using the **DISPlay:ANALysis:TYPE** command.
- This query is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command at the particular input channel.

Example

The following command sequence is used to obtain the harmonic fundamental frequency of channel 2.

```
INIT:GRAP (@2)
CALC:HARM:FUND? (@2)
```

Typical response:

```
1.000000E+03
```

CALCulate[:ANALog]:HARMonic:STATe

Syntax

```
CALCulate[:ANALog]:HARMonic:STATe <state>,(@<channel>)
CALCulate[:ANALog]:HARMonic:STATe? (@<channel>)
```

Description

Turns on or off the harmonic analysis status. The query returns the harmonic analysis state as `0` if the harmonic analysis state is `OFF`, or `1` if the harmonic analysis state is `ON`.

Parameters

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	OFF
`<channel>`	Numeric	1 to 8	Required parameter

Remarks

– This command should be turned on before sending the **INITiate[:IMMediate]:GRAPh** command.

– Turning on the harmonic analysis will switch the graph analysis page GUI to the harmonic analysis view page.

Examples

The following command turns on harmonic analysis for channel 1.

```
CALC:HARM:STAT ON,(@1)
```

The following query returns the state for the channel 1 harmonic analysis.

```
CALC:HARM:STAT? (@1)
```

Typical response:

```
1
```

CALCulate[:ANALog]:HARMonic:VALue?

Syntax

```
CALCulate[:ANALog]:HARMonic:VALue? (@<channel>)
```

Description

Returns the harmonic component results of the trace for the specified channel. Multiple responses are separated by commas.

Parameter

Item	Type	Range of values	Default value
<channel>	Numeric	1 to 8	Required parameter

Remarks

- The graph analysis mode must be set to frequency domain (MAGnitude) using the **DISPlay:ANALysis:TYPE** command.
- This query is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command at the particular input channel.
- The number of harmonic component results returned is based on the harmonic count specified in the **CALCulate[:ANALog]:HARMonic:AMPL?** command.

Example

The following command sequence is used to obtain the harmonic component results of channel 2.

```
INIT:GRAP (@2)
CALC:HARM:VAL? (@2)
```

Typical response:

```
-1.44019E+00,-6.48722E+01,-7.28213E+01,-7.76705E+01,-8.12592E+01,
-8.39658E+01,-8.62497E+01,-8.79064E+01
```

CALCulate[:ANALog]:READing:FORMat

Syntax

```
CALCulate[:ANALog]:READing:FORMat
<format>,<function>,(@<channel_list>)
```

```
CALCulate[:ANALog]:READing:FORMat? <function>,(@<channel_list>)
```

Description

Sets the reading format of the measurement to be returned for the specified measurement function and channel of the analog analyzer. The query returns the measurement reading format.

Parameters

Item	Type	Range of values	Default value
`<format>`	Discrete	LINear|LOGaritmic|DELTa|OFF	OFF
`<function>`	Discrete	FUNCtion1|FUNCtion2|FUNCtion3|FUNCtion4	FUNCtion2
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- `LINear` will return the measurement reading in unit x. This format is only applicable if the measurement function is level-based and ratio-based measurements. The result in unit x is calculated by dividing the acquired measurement reading by the reference value (Measured Value/Reference Value).
- `LOGaritmic` will return the measurement result in unit dBr. This format is only applicable if the measurement function is level-based type. The result is calculated using formula below:

  $$20\mathrm{Log}_{10}\left(\frac{\text{Measured rms voltage}}{\text{Reference level}}\right)$$

- `DELTa` will return the measurement data, calculated by deducting the reference value from the measured value (Measured Value – Reference Value). For level-based measurements, the unit is in ΔV (represented as dV in SCPI). For frequency-based measurements, the unit is in ΔHz (represented as dHz in SCPI). For ratio-based measurements, the unit is in ΔdB (represented as ddB in SCPI).

– `OFF` will turn off the measurement reading formatting. The returned measurement reading will be in its default unit.

Examples

The following command sets the measurement reading format for function 1 of channel 1 to `DELTa`.

```
CALC:READ:FORM DELT,FUNC1,(@1)
```

The following query returns the measurement reading format for function 1 of channel 1.

```
CALC:READ:FORM? FUNC1,(@1)
```

Typical response:

```
DELT
```

CALCulate[:ANALog]:STATistics:COUNt

Syntax

```
CALCulate[:ANALog]:STATistics:COUNt
<number_of_readings>,(@<channelist>)
```

```
CALCulate[:ANALog]:STATistics:COUNt? (@<channel_list>)
```

Description

Specifies the number of measurement readings to be taken in order to calculate the statistics values for the specified channels.

Parameters

Item	Type	Range of values	Default value
`<number_of_readings>`	Numeric	2 to 20	10
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

Refer to "**Example 12: Obtain Statistics Data from the Analyzer**" on page 657 for the example in using this command.

Examples

The following command sets the number of readings to be taken for statistics calculation to 20.

```
CALC:STAT:COUN 20,(@1)
```

The following query returns the number of readings to be taken for statistics calculation.

```
CALC:STAT:COUN? (@1)
```

Typical response:

```
20
```

CALCulate[:ANALog]:STATistics:DATA<i>?

Syntax

```
CALCulate[:ANALog]:STATistics:DATA<i>?
<function>,(@<channel_list>)
```

Description

Queries for the statistics data calculated for the selected statistics number and function of the selected channel. The query returns the calculated statistics data of the selected analog analyzer channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<i>`	Numeric	1 to 3 – 1 for the first statistics type – 2 for the second statistics type – 3 for the third statistics type	1
`<function>`	Numeric	FUNCtion1|FUNCtion2|FUNCtion3|FUNCtion4	FUNCtion1
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- This command can only return valid results if the analog analyzer measurements have been made and there are sufficient measurement readings data for the statistics calculation.
- Refer to "**Example 12: Obtain Statistics Data from the Analyzer**" on page 657 for the example in using this command.

Example

The following query returns the calculated first statistics data of the function 2 of analog analyzer channel 1.

```
CALC:STAT:DATA1? FUNC2,(@1)
```

Typical response:

```
9.304861E+01
```

CALCulate[:ANALog]:STATistics:RESet

Syntax

```
CALCulate[:ANALog]:STATistics:RESet (@<channel_list>)
```

Description

Resets the statistics values of the specified analog analyzer channels.

Parameter

Item	Type	Range of values	Default value
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– After this command is sent, the following statistics data will be reset.

  MAX = $-\infty$

  MIN = $+\infty$

  DMMAX = 0

  SD = 0

  AVG = 0

– This is not a reset command. To reset the statistic state and count, use the ***RST**, **SYSTem:PRESet**, or **SYSTem:RESet[:MODE]** command.

– Refer to "**Example 12: Obtain Statistics Data from the Analyzer**" on page 657 for the example in using this command.

Example

The following command resets the statistics values of analog analyzer channel 1.

```
CALC:STAT:RES (@1)
```

CALCulate[:ANALog]:STATistics:STATe

Syntax

```
CALCulate[:ANALog]:STATistics:STATe <state>,(@<channel_list>)
CALCulate[:ANALog]:STATistics:STATe? (@<channel_list>)
```

Description

Enables or disables the average and standard deviation calculation of the measurement result. The query returns the statistics state of the selected analog analyzer channels. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	OFF
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– The MAX, MIN, and DMMAX will still be updated when the analyzer measurement is running with the statistics state disabled.

– When the statistics state is enabled, the average and standard deviation will only be updated when the number of readings reach the number set by the **CALCulate[:ANALog]:STATistics:COUNt** command.

– The statistics values are accumulative for each of the measurement function. You may reset the values by sending the **CALCulate[:ANALog]:STATistics:RESet** command.

– Refer to "**Example 12: Obtain Statistics Data from the Analyzer**" on page 657 for the example in using this command.

Examples

The following command enables the average and standard deviation calculation for analog analyzer channel 1.

```
CALC:STAT:STAT ON,(@1)
```

The following query returns the statistics state for the analog analyzer channel 1.

```
CALC:STAT:STAT? (@1)
```

Typical response:

```
1
```

CALCulate[:ANALog]:STATistics:TYPE<i>

Syntax

```
CALCulate[:ANALog]:STATistics:TYPE<i>
<statistics_type>,(@<channel_list>)
```

```
CALCulate[:ANALog]:STATistics:TYPE<i>? (@<channel_list>)
```

Description

Specifies the type of statistics of the selected channel(s). You are allowed to make calculation for up to 3 statistics data for each function. The query returns the statistics type of the selected analog analyzer channel(s). Multiple responses are separated by commas.

The types of statistics are listed as follows:

MIN	Minimum
MAX	Maximum
AVG	Average
SD	Standard deviation
DMMax	Delta value of the maximum and minimum (Result = Maximum – Minimum)

Parameters

Item	Type	Range of values	Default value
<i>	Numeric	1 to 3 – 1 for the first statistics type – 2 for the second statistics type – 3 for the third statistics type	1
<statistic_type>	Discrete	MIN|MAX|AVG|SD|DMMaz	– MIN for statistics 1 – MAX for statistics 2 – AVG for statistics 3
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- The three statistics types selected for the particular channel are applied for all functions in that channel. You are not allowed to select different statistics types for every single function in the same channel. For example, if you select `MIN` as the first statistics type for channel 1, all analog analyzer functions (Function 1, Function 2, Function 3, and Function 4) will do the calculation for minimum. In other words, you are not allowed to select MIN as the first statistics type for function 1 and then select `SD` as the first statistics type for function 2 of channel 1.
- Refer to "**Example 12: Obtain Statistics Data from the Analyzer**" on page 657 for the example in using this command.

Examples

The following command sets the second statistics type to `SD` for analog analyzer channel 1.

```
CALC:STAT:TYPE2 SD,(@1)
```

The following query returns the second statistics type for analog analyzer channel 1.

```
CALC:STAT:TYPE2? (@1)
```

Typical response:

```
SD
```

CALCulate[:ANALog]:THDistortion?

Syntax

```
CALCulate[:ANALog]:THDistortion? <unit>,(@<channel_list>)
```

Description

Queries for the Total Harmonic Distortion (THD) value of the input signal at the particular channel. The query will return the THD result in the unit as specified. You can select the result to be returned in decibel (dB) or percentage (%) by changing the unit to `DB` or `PCT` respectively.

Parameter

Item	Type	Range of values	Default value
`<unit>`	Discrete	DB|PCT	PCT
`<channel>`	Numeric	1 to 8	Required parameter

Remarks

- The graph analysis mode must be set to the frequency domain (`MAGnitude`) using the **DISPlay:ANALysis:TYPE** command.
- This command is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command at the particular input channel.

Example

The following command sequence is used to obtain the THD value of the input signal at channel 2 in percentage.

```
CALC:HARM:STAT (@2),ON
INIT:GRAP (@2)
CALC:THD? PCT,(@2)
```

Typical response:

```
1.691385E+01
```

CALCulate:DIGital:READing:FORMat

Syntax

```
CALCulate:DIGital:READing:FORMat
<format>,<function>,(@<channel_list>)
```

```
CALCulate:DIGital:READing:FORMat? (@<channel_list>),<function>
```

Description

Sets the reading format of the measurement to be returned for the specified measurement function and channel of the digital analyzer. The query returns the measurement reading format.

Parameters

Item	Type	Range of values	Default value
`<format>`	Discrete	LINear|LOGaritmic|DELTa|OFF	OFF
`<function>`	Discrete	FUNCtion1|FUNCtion2|FUNCtion3|FUNCtion4	FUNCtion2
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

– `LINear` will return the measurement reading in unit x. This format is only applicable if the measurement function is level-based and ratio-based measurements. The result in unit x is calculated by dividing the acquired measurement reading by the reference value (Measured Value/Reference Value).

– `LOGaritmic` will return the measurement result in unit dBr. This format is only applicable if the measurement function is level-based type. The result is calculated using formula below:

$$20\mathrm{Log}_{10}\left(\frac{\text{Measured rms voltage}}{\text{Reference level}}\right)$$

– `DELTa` will return the measurement data, calculated by deducting the reference value from the measured value (Measured Value – Reference Value). For frequency-based measurements, the unit is in ΔHz (represented as dHz in SCPI). For ratio-based measurements, the unit is in ΔdB (represented as ddB in SCPI).

– `OFF` will turn off the measurement reading formatting. The returned measurement reading will be in its default unit.

Examples

The following command sets the measurement reading format for function 1 of channel 1 to `DELTa`.

```
CALC:DIG:READ:FORM DELT,FUNC1,(@D1)
```

The following query returns the measurement reading format for function 1 of channel 1.

```
CALC:DIG:READ:FORM? FUNC1,(@D1)
```

Typical response:

```
DELT
```

CALCulate:DIGital:STATistics:COUNt

Syntax

```
CALCulate:DIGital:STATistics:COUNt
<number_of_readings>,(@<channelist>)
```

```
CALCulate:DIGital:STATistics:COUNt? (@<channel_list>)
```

Description

Specifies the number of measurement readings to be taken in order to calculate the statistics values for the specified channels. The query returns the number of the measurement readings.

Parameters

Item	Type	Range of values	Default value
`<number_of_readings>`	Numeric	2 to 20	10
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remark

Refer to “**Example 12: Obtain Statistics Data from the Analyzer**” on page 657 for the example in using this command.

Examples

The following command sets the number of readings to be taken for statistics calculation to 20.

```
CALC:DIG:STAT:COUN 20,(@D1)
```

The following query returns the number of readings to be taken for statistics calculation.

```
CALC:DIG:STAT:COUN? (@D1)
```

Typical response:

```
20
```

CALCulate:DIGital:STATistics:STATe

Syntax

```
CALCulate:DIGital:STATistics:STATe <state>,(@<channel_list>)
CALCulate:DIGital:STATistics:STATe? (@<channel_list>)
```

Description

Enables or disables the average and standard deviation calculation of the measurement result. The query returns the statistics state of the selected digital analyzer channels. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	OFF
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

– The MAX, MIN, and DMMAX will still be updated when the analyzer measurement is running with the statistics state disabled.

– When the statistics state is enabled, the average and standard deviation will only be updated when the number of readings reach the number set by the **CALCulate:DIGital:STATistics:COUNt** command.

– The statistics values are accumulative for each of the measurement function. You may reset the values by sending the **CALCulate:DIGital:STATistics:RESet** command.

– Refer to “**Example 12: Obtain Statistics Data from the Analyzer**” on page 657 for the example in using this command.

Examples

The following command enables the average and standard deviation calculation for digital analyzer channel 1.

```
CALC:DIG:STAT:STAT ON,(@D1)
```

The following query returns the statistics state for the digital analyzer channel 1.

```
CALC:DIG:STAT:STAT? (@D1)
```

Typical response:

```
1
```

CALCulate:DIGital:STATistics:RESet

Syntax

```
CALCulate:DIGital:STATistics:RESet (@<channel_list>)
```

Description

Resets the statistics values of the specified digital analyzer channels.

Parameter

Item	Type	Range of values	Default value
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

– After this command is sent, the following statistics data will be reset.

MAX = –∞

MIN = +∞

DMMAX = 0

SD = 0

AVG = 0

– This is not a reset command. To reset the statistic state and count, use the ***RST**, **SYSTem:PRESet**, or **SYSTem:RESet[:MODE]** command.

– Refer to "**Example 12: Obtain Statistics Data from the Analyzer**" on page 657 for the example in using this command.

Example

The following command resets the statistics values of digital analyzer channel 1.

```
CALC:DIG:STAT:RES (@D1)
```

CALCulate:DIGital:STATistics:TYPE<i>

Syntax

```
CALCulate:DIGital:STATistics:TYPE<i>
<statistics_type>,(@<channel_list>)
```

```
CALCulate:DIGital:STATistics:TYPE<i>? (@<channel_list>)
```

Description

Specifies the type of statistics of the selected channel(s). You are allowed to make calculation for up to 3 statistics data for each function. The query returns the statistics type of the selected digital analyzer channel(s). Multiple responses are separated by commas.

The types of statistics are listed as follows:

`MIN`	Minimum
`MAX`	Maximum
`AVG`	Average
`SD`	Standard deviation
`DMMax`	Delta value of the maximum and minimum (Result = Maximum – Minimum)

Parameters

Item	Type	Range of values	Default value
`<i>`	Numeric	1 to 3 – 1 for the first statistics type – 2 for the second statistics type – 3 for the third statistics type	1
`<statistic_type>`	Discrete	MIN|MAX|AVG|SD|DMMaz	– MIN for statistics 1 – MAX for statistics 2 – AVG for statistics 3
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- The three statistics types selected for the particular channel are applied for all functions in that channels. You are not allowed to select different statistics types for every single function in the same channel. For example, if you select `MIN` as the first statistics type for channel 1, all analog analyzer functions (Function 1, Function 2, Function 3, and Function 4) will do the calculation for minimum. In other words, you are not allowed to select MIN as the first statistics type for function 1 and then select `SD` as the first statistics type for function 2 of channel 1.
- Refer to "**Example 12: Obtain Statistics Data from the Analyzer**" on page 657 for the example in using this command.

Examples

The following command sets the second statistics type to `SD` for digital analyzer channel 1.

```
CALC:DIG:STAT:TYPE2 SD,(@D1)
```

The following query returns the second statistics type for analog digital channel 1.

```
CALC:DIG:STAT:TYPE2? (@D1)
```

Typical response:

```
SD
```

CALCulate:DIGital:STATistics:DATA<i>?

Syntax

```
CALCulate:DIGital:STATistics:DATA<i>?
<function>,(@<channel_list>)
```

Description

Queries for the statistics data calculated for the selected statistics number and function of the selected channel. The query returns the calculated statistics data of the selected digital analyzer channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<i>`	Numeric	1 to 3 – 1 for the first statistics type – 2 for the second statistics type – 3 for the third statistics type	1
`<function>`	Numeric	FUNCtion1|FUNCtion2|FUNCtion3|FUNCtion4	FUNCtion1
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

– This command can only return valid results if the digital analyzer measurements have been made and there are sufficient measurement readings data for the statistics calculation.

– Refer to "**Example 12: Obtain Statistics Data from the Analyzer**" on page 657 for the example in using this command.

Example

The following query returns the calculated first statistics data of the function 2 of digital analyzer channel 1.

```
CALC:DIG:STAT:DATA1? FUNC2,(@D1)
```

Typical response:

```
9.304861E+01
```

CALCulate:DIGital:HARMonic:STATe

Syntax

```
CALCulate:DIGital:HARMonic:STATe <state>,(@<channel>)
CALCulate:DIGital:HARMonic:STATe? (@<channel>)
```

Description

Turns on or off the harmonic analysis status. The query returns the harmonic analysis state as `0` if the harmonic analysis state is `OFF`, or `1` if the harmonic analysis state is `ON`.

Parameters

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	OFF
`<channel>`	Numeric	D1|D2	Required parameter

Remarks

– This command should be turned on before sending the **INITiate[:IMMediate]:DIGital:GRAPh** command.

– Turning on the harmonic analysis will switch the graph analysis page GUI to the harmonic analysis view page.

Examples

The following command turns on harmonic analysis for channel 1.

```
CALC:HARM:STAT ON,(@D1)
```

The following query returns the state for the channel 1 harmonic analysis.

```
CALC:HARM:STAT? (@D1)
```

Typical response:

```
1
```

CALCulate:DIGital:HARMonic:FUNDamental?

Syntax

```
CALCulate:DIGital:HARMonic:FUNDamental?(@<channel>)
```

Description

Returns the signal fundamental frequency in Hz for the specified channel.

Parameter

Item	Type	Range of values	Default value
`<channel>`	Discrete	D1|D2	Required parameter

Remark

– This command is only applicable after the graph data has been acquired using the **INITiate[:IMMediate]:DIGital:GRAPh** command at the particular input channel.

Examples

The following commands are used to obtain the signal fundamental frequency of channel 2.

```
INIT:DIG:GRAP (@D2)
CALC:DIG:HARM:FUND? (@D2)
```

Typical response:

```
1.000000E+03
```

CALCulate:DIGital:HARMonic:VALue?

Syntax

```
CALCulate:DIGital:HARMonic:VALue? (@<channel>)
```

Description

Returns the harmonic component results of the trace for the specified channel. Multiple responses are separated by commas.

Parameter

Item	Type	Range of values	Default value
`<channel>`	Discrete	D1|D2	Required parameter

Remark

- This query is only applicable after the graph data has been acquired using the **INITiate[:IMMediate]:DIGital:GRAPh** command at the particular input channel.

Examples

The following commands are used to obtain the harmonic component results of channel 2.

```
INIT:IMM:DIG:GRAP (@D2)
```

```
CALC:DIG:HARM:VAL? (@D2)
```

Typical response:

```
-1.440191E+00,-6.487222E+01,-7.282130E+01,
-7.767053E+01,-8.125921E+01,-8.396585E+01,
-8.624970E+01,-8.790641E+01
```

CALCulate:DIGital:HARMonic:FREQuencies?

Syntax

```
CALCulate:DIGital:HARMonic:FREQuencies?(@<channel>)
```

Description

Returns the signal harmonic frequency values for the specified channel. Multiple responses are separated by commas.

Parameter

Item	Type	Range of values	Default value
`<channel>`	Discrete	D1|D2	Required parameter

Remark

– This query is only applicable after the graph data has been acquired using the **INITiate[:IMMediate]:DIGital:GRAPh** command at the particular input channel.

Examples

The following commands are used to obtain the harmonic frequency values of channel 2.

```
INIT:IMM:DIG:GRAP (@D2)
CALC:DIG:HARM:FREQ? (@D2)
```

Typical response:

```
9.918210E+02,2.002721E+03,2.994543E+03,
4.005431E+03,4.997250E+03,6.008151E+03,
6.999972E+03,7.991791E+03
```

CALCulate:DIGital:THDistortion?

Syntax

```
CALCulate:DIGital:THDistortion? <unit>,(@<channel>)
```

Description

Returns the Total Harmonic Distortion (THD) value of the input signal in the specified unit for the selected channel. The returned value can either be in dB or percentage by setting `<unit>` to DB or PCT respectively.

Parameter

Item	Type	Range of values	Default value
`<unit>`	Discrete	DB|PCT	PCT
`<channel>`	Discrete	D1|D2	Required parameter

Remark

– This query is only applicable after the graph data has been acquired using the **INITiate[:IMMediate]:DIGital:GRAPh** command at the particular input channel.

Examples

The following commands are used to obtain the distortion value of the input signal at channel 2 in percentage.

```
INIT:IMM:DIG:GRAP (@D2)
CALC:DIG:THD? PCT, (@D2)
```

Typical response:

```
1.691385E+01
```

CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:FUNCtion:FUNCtion

Syntax

```
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:FUNCtion:FUNCtion
<function>
```

```
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:FUNCtion:FUNCtion?
```

Description

Sets the marker function of the selected marker. The query returns the marker function of the selected marker.

Parameter

Item	Type	Range of values	Default value
`<function>`	Discrete	NONE|SLOPe|PSD	Required parameter

Remark

The marker has to be enabled before sending this command in obtain the correct value.

Examples

The following command sets the marker 2 to measure slope.

```
CALC:GRAP:MARK2:FUNC:FUNC SLOP
```

The following query returns the marker function of marker 3.

```
CALC:GRAP:MARK3:FUNC:FUNC?
```

Typical response:

```
BAND
```

CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:FUNCtion:VALue?

Syntax

```
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:FUNCtion:VALue?
```

Description

The query returns the marker function value of the selected marker.

Remark

If marker function is set to NONE using the **CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:FUNCtion:FUNCtion** command or the marker is not enabled, the reading returned will be not a number (NaN).

Example

The following query returns the marker function value of marker 1.

```
CALC:GRAP:MARK1:FUNC:VAL?
```

CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:MIN

Syntax

```
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:MIN <direction>
```

Description

Searches for the minimum value of the trace data by placing the specified marker at either the left or right minimum of the graph display in the graph analysis mode. Selecting all will search for the lowest minimum value from the left and right direction. The specified marker will become the active marker.

Parameter

Item	Type	Range of values	Default value
`<direction>`	Discrete	LEFT|RIGHt|ALL	RIGHt

Remarks

- This command is similar to the Minimum Search softkey at the front panel.
- This command is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command.
- The top and bottom of the graph must be scaled correctly. Any peaks out of the graph scale will not be included into the peak calculation.

Example

The following command places marker 2 at the right minimum of the graph display.

```
CALC:GRAP:MARK2:MIN RIGH
```

CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:MOVement

Syntax

```
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:MOVement <movement_type>
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:MOVement?
```

Description

Sets the marker movement characteristic on the graph display in the graph analysis mode. The query returns the marker movement characteristic. If the coupled markers move out of the graph grid, not a number (NaN) value will be returned.

The description for each <movement_type> parameter is listed as follows.

`SINGle`	The active marker will move depending on the speed and direction of the knob being turned.
`PAIR`	The active marker will move with the reference marker.
`BIN`	The active marker will move to the next/previous bin or pixel regardless of the speed of the knob.
`PEAK`	The active marker will move from peak to peak. This behavior is affected by the peak threshold settings.
`HARMonic`	The active marker will move from harmonic to harmonic. This setting is only useful if the display option is changed to Harmonics.

Parameter

Item	Type	Range of values	Default value
`<movement_type>`	Discrete	SINGle|PAIR|BIN|PEAK|HARMonic	SINGle

Remarks

– This command is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command.

– To move the markers in pairs, you need to specify the reference marker of the selected marker prior to sending this command. To define a reference marker, use the **CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:REFerence** command.

Examples

The following command sequence sets marker 1 and marker 2 as its reference marker to move together on the graph.

```
CALC:GRAP:MARK1:REF M2
CALC:GRAP:MARK1:MOV PAIR
CALC:GRAP:MARK1:X 3.2500E+02
```

The following query returns the movement characteristic of marker 1.

```
CALC:GRAP:MARK1:MOV?
```

Typical response:

```
PAIR
```

CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:PEAK

Syntax

```
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:PEAK <direction>
```

Description

Searches for the peak value of the trace data by placing the specified marker at either the left or right peak of the graph display in the graph analysis mode. Selecting all will search for the highest peak value from the left and right direction. The specified marker will become the active marker.

Parameter

Item	Type	Range of values	Default value
`<direction>`	Discrete	LEFT|RIGHt|ALL	RIGHt

Remarks

- This command is similar to the Peak Search softkey at the front panel.
- This command is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command.
- The top and bottom of the graph must be scaled correctly. Any peaks out of the graph scale will not be included into the peak calculation.

Example

The following command places marker 2 at the left peak of the graph display.

```
CALC:GRAP:MARK2:PEAK LEFT
```

CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:REFerence

Syntax

```
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:REFerence
<reference_marker_no>
```

```
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:REFerence?
```

Description

Sets the reference marker for the selected marker on the graph display in the graph analysis mode. The query returns the reference marker for the specified marker.

Parameter

Item	Type	Range of values	Default value
`<reference_marker_no>`	Discrete	M1|M2|M3|M4|M5|M6|M7|M8|OFF	OFF

Remarks

- This command is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command.
- A marker cannot be referenced to itself.
- Sending this command will turn on the reference marker. Select OFF to turn off the reference marker.
- If the selected marker has no reference marker when queried, error –200,“Execution Error;The marker has no reference marker” is generated.

Examples

The following command sets the reference marker as marker 2 for marker 1 on the graph display.

```
CALC:GRAP:MARK1:REF M2
```

The following query returns the reference marker for marker 1.

```
CALC:GRAP:MARK1:REF?
```

Typical response:

```
M2
```

CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8[:SET]:MODE

Syntax

```
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8[:SET]:MODE <marker_mode>
```

Description

Positions the marker at either the start, stop, or center points of the graph in the graph analysis mode. You may also expand the area between the selected marker and its reference marker.

The description for each <marker_mode> parameter is listed as follows.

STARt	Set the left value of the graph to the current marker location.
STOP	Set the right value of the graph to the current marker location.
CENTer	Set the center value of the graph to the current marker location.
DSPan	Set the left and right values of the graph to the current marker and the reference marker location.

Parameter

Item	Type	Range of values	Default value
<marker_mode>	Discrete	STARt|STOP|CENTer|DSPan	Required parameter

Remarks

- This command is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command.
- The DSPan mode is only applicable for a selected marker which has a reference marker.

Example

The following command positions marker 2 at the graph start point.

```
CALC:GRAP:MARK2:MODE STAR
```

CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:STATe

Syntax

```
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:STATe <state>
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:STATe?
```

Description

Turns on or off the selected marker on the graph display in the graph analysis mode. The selected marker will become the active marker when it is turned on. The query returns the marker state as `0` if the marker state is `OFF`, or `1` if the marker state is `ON`.

Parameter

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	OFF

Remark

This command is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command.

Examples

The following command turns on marker 2 on the graph display.

```
CALC:GRAP:MARK2:STAT ON
```

The following query returns the state for marker 2.

```
CALC:GRAP:MARK2:STAT?
```

Typical response:

```
1
```

CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:TRACe

Syntax

```
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:TRACe <trace_no>
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:TRACe?
```

Description

Assigns the marker to the trace of the specified channel on the graph display in the graph analysis mode. The trace number corresponds with the channel number. For example, `CHANnel1` or `TRACe1` represents the trace for analog analyzer channel 1. The selected marker will become the active marker. `R1`, `R2`, and `R3` correspond to the respective reference traces loaded into the graph. For backward compatibility with U8903A, use `CHANnel` and `R1` to `R3` parameters. The query returns the trace number for the specified marker.

Parameter

Item	Type	Range of values	Default value
<trace_no>	Discrete	CHANnel1|CHANnel2|CHANnel3|CHANnel4|CHANnel5|CHANnel6|CHANnel7|CHANnel8|TRACe1|TRACe2|TRACe3|TRACe4|TRACe5|TRACe6|TRACe7|TRACe8|R1|R2|R3	CHANnel1

Remarks

- This command is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command.
- If a marker is activated using other command without assigning a channel to it, the marker will be assigned to `AA1` by default.
- Regardless of the values being set, the return values of the marker trace will always be in the range of `TRACe1` to `TRACe8`.

Examples

The following command assigns marker 1 to the second trace on the graph display.

```
CALC:GRAP:MARK:TRAC TRAC2
```

The following query returns the trace data channel that marker 1 is assigned to.

```
CALC:GRAP:MARK:TRAC?
```

Typical response:

```
TRAC1
```

CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:X

Syntax

```
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:X <x-position>
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:X?
```

Description

Sets the marker X-axis value on the graph display in the graph analysis mode. The selected marker will become the active marker. The query returns the marker X-axis value. If the marker state is off, the response is not a number (NaN).

Parameter

Item	Type	Range of values	Default value
<x-position>	Numeric	–200000 to 200000	0

Remark

This command is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command.

Examples

The following command sets the marker 2 X-axis value to 550 Hz on the graph analysis display (assume that the graph is in the frequency domain).

```
CALC:GRAP:MARK2:X 550
```

The following query returns the marker 2 X-axis value.

```
CALC:GRAP:MARK2:X?
```

Typical response:

```
5.500000E+02
```

CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:XDELta?

Syntax

```
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:XDELta?
```

Description

Returns the difference in the X-axis value between the selected marker and its reference marker on the graph display in the graph analysis mode.

Remarks

- This query is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command.
- If the marker state is off, the response is not a number (NaN).
- If the selected marker has no reference marker, the response is also not a number (NaN).

Example

The following query returns the delta X-axis value for marker 2.

```
CALC:GRAP:MARK2:XDEL?
```

Typical response:

```
3.500000E+02
```

CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:Y?

Syntax

```
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:Y?
```

Description

Returns the marker Y-axis value on the graph display in the graph analysis mode.

Remarks

- This query is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command.
- If the marker state is off, the response is not a number (NaN).

Example

The following query returns the marker 2 Y-axis value.

```
CALC:GRAP:MARK2:Y?
```

Typical response:

```
0.000000E+00
```

CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:YDELta?

Syntax

```
CALCulate:GRAPh:MARKer[1]|2|3|4|5|6|7|8:YDELta?
```

Description

Returns the difference in the Y-axis value between the selected marker and its reference marker on the graph display in the graph analysis mode.

Remarks

- This query is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command.
- If the marker state is off, the response is not a number (NaN).
- If the selected marker has no reference marker, the response is also not a number (NaN).

Example

The following query returns the delta Y-axis value for marker 2.

```
CALC:GRAP:MARK2:YDEL?
```

Typical response:

```
5.000000E+00
```

CALCulate:GRAPh:MARKer:THReshold[:LEVel]

Syntax

```
CALCulate:GRAPh:MARKer:THReshold[:LEVel]
<threshold_level>[,<threshold_type>]
```

```
CALCulate:GRAPh:MARKer:THReshold[:LEVel]?
```

Description

Sets the threshold level that the marker can identify as a peak or minimum on the graph display in the graph analysis mode. If the trace is above the threshold level, it will be identified as a peak, whereas the trace below the threshold level will be identified as a minimum. If no threshold type is defined, the peak and mininimum threshold will be set to the same value. The query returns the specified threshold level.

Parameters

Item	Type	Range of values	Default value
`<threshold_level>`	Numeric	Within the top edge and bottom edge of the display	–100
`<threshold_type>`	Discrete	PEAK|MIN	PEAK

Remark

This command is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command.

Examples

The following command sets the threshold level to 20.

```
CALC:GRAP:MARK:THR 20
```

The following query returns the threshold level.

```
CALC:GRAP:MARK:THR?
```

Typical response:

```
2.000000E+01
```

CALCulate:GRAPh:MARKer:THReshold:STATe

Syntax

```
CALCulate:GRAPh:MARKer:THReshold:STATe <threshold_state>
```

```
CALCulate:GRAPh:MARKer:THReshold:STATe?
```

Description

Turns on or off the threshold on the graph display in the graph analysis mode. The query returns the threshold state as `0` if the state is `OFF`, or `1` if the state is `ON`.

Parameter

Item	Type	Range of values	Default value
`<threshold_state>`	Boolean	OFF(0)|ON(1)	OFF

Remark

This command is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command.

Examples

The following command turns on the threshold on the graph display.

```
CALC:GRAP:MARK:THR:STAT ON
```

The following query returns the threshold state.

```
CALC:GRAP:MARK:THR:STAT?
```

Typical response:

```
1
```

CALCulate:HARMonic:MARKer:STATe

Syntax

```
CALCulate:HARMonic:MARKer:STATe <state>
CALCulate:HARMonic:MARKer:STATe?
```

Description

Turns on or off the harmonic markers display. The query returns the harmonic markers display state as `0` if the state is `OFF`, or `1` if the state is `ON`.

Parameter

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	OFF

Remarks

- The display analysis mode must be set to harmonics in the frequency domain.
- This command is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command.

Examples

The following command turns on the harmonic markers display.

```
CALC:MARK:STAT ON
```

The following query returns the harmonic markers display state.

```
CALC:MARK:STAT?
```

Typical response:

```
1
```

CALCulate:HARMonic:MARKer:TRACe

Syntax

```
CALCulate:HARMonic:MARKer:TRACe <trace_id>
```

```
CALCulate:HARMonic:MARKer:TRACe?
```

Description

Sets the trace that the harmonic markers are attached. The selected trace source must be set to a channel or the markers will not be displayed on the screen. The query returns the trace that the harmonic markers are attached.

Parameter

Item	Type	Range of values	Default value
<trace_id>	Numeric	1 to 8	8

Remarks

- The display analysis mode must be set to harmonics in the frequency domain.
- This command is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command.

Examples

The following command attaches the harmonic marker to trace 2.

```
CALC:HARM:MARK:TRAC 2
```

The following query returns the trace that the harmonic markers are attached.

```
CALC:HARM:MARK:TRAC?
```

Typical response:

```
2
```

CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:MIN

Syntax

```
CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:MIN <direction>
```

Description

Searches for the minimum value of the trace data by placing the specified marker at either the left or right minimum of the graph display in the sweep mode. Selecting all will search for the lowest minimum value from the left and right direction. The specified marker will become the active marker.

Parameter

Item	Type	Range of values	Default value
`<direction>`	Discrete	LEFT|RIGHt|ALL	RIGHt

Remarks

- This command is similar to the Minimum Search softkey at the front panel.
- This command is only applicable after the sweep data is acquired using the **INITiate[:IMMediate]:SWEep** command.
- The top and bottom of the graph must be scaled correctly. Any peaks out of the graph scale will not be included in the peak calculation.

Example

The following command places marker 2 at the right minimum of the graph display.

```
CALC:SWE:MARK2:MIN RIGH
```

CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:MOVement

Syntax

```
CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:MOVement
<movement_characteristic>
```

```
CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:MOVement?
```

Description

Sets the marker movement characteristic of either single or in pair on the graph display in the sweep mode. The query returns the marker movement characteristic in the form of either `SING` or `PAIR`.

The description for each `<movement_characteristic>` parameter is listed as follows.

SINGle	Move only the selected marker on the graph.
PAIR	Move both the selected and reference markers in the same direction on the graph.

Parameter

Item	Type	Range of values	Default value
<movement_characteristic>	Discrete	SINGle|PAIR	SINGle

Remarks

- This command is only applicable after the sweep data is acquired using the **INITiate[:IMMediate]:SWEep** command.
- To move the markers in pairs, you need to specify the reference marker of the selected marker prior to sending this command. To define a reference marker for the selected marker, use the **CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:REFerence** command.

Examples

The following command sequence sets marker 1 and marker 2 as its reference marker to move together on the graph.

```
CALC:SWE:MARK1:REF M2
CALC:SWE:MARK1:MOV PAIR
CALC:SWE:MARK1:X 3.2500E+02
```

The following query returns the movement characteristic of marker 1.

```
CALC:SWE:MARK1:MOV?
```

Typical response:

```
PAIR
```

CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:PEAK

Syntax

```
CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:PEAK <direction>
```

Description

Searches for the peak value of the trace data by placing the specified marker at either the left or right peak of the graph display in the sweep mode. Selecting all will search for the highest peak value from the left and right direction. The specified marker will become the active marker.

Parameter

Item	Type	Range of values	Default value
`<direction>`	Discrete	LEFT|RIGHt|ALL	RIGHt

Remarks

- This command is similar to the Peak Search softkey at the front panel.
- This command is only applicable after the sweep data is acquired using the **INITiate[:IMMediate]:SWEep** command.
- The top and bottom of the graph must be scaled correctly. Any peaks out of the graph scale will not be included in the peak calculation.

Example

The following command places marker 2 at the left peak of the graph display.

```
CALC:SWE:MARK2:PEAK LEFT
```

CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:REFerence

Syntax

```
CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:REFerence
<reference_marker_no>
```

```
CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:REFerence?
```

Description

Sets the reference marker for the selected marker on the graph display in the sweep mode. The query returns the reference marker for the specified marker.

Parameter

Item	Type	Range of values	Default value
`<reference_marker_no>`	Discrete	M1|M2|M3|M4|M5|M6|M7|M8|OFF	OFF

Remarks

- This command is only applicable after the sweep data is acquired using the **INITiate[:IMMediate]:SWEep** command.
- A marker cannot be referenced to itself.
- Sending this command will turn on the reference marker. Select `OFF` to turn off the reference marker.
- If the selected marker has no reference marker when queried, error –200,“Execution Error;The marker has no reference marker” is generated.

Examples

The following command sets the reference marker as marker 2 for marker 1 on the | graph display.

```
CALC:SWE:MARK1:REF M2
```

The following query returns the reference marker for marker 1.

```
CALC:SWE:MARK1:REF?
```

Typical response:

```
M2
```

CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8[:SET]:MODE

Syntax

```
CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8[:SET]:MODE <marker_mode>
```

Description

Positions the marker at either the start, stop, or center points of the graph in the sweep mode. You may also expand the area between the selected marker and its reference marker.

The description for each `<marker_mode>` parameter is listed as follows.

`STARt`	Position the marker at the graph start point.
`STOP`	Position the marker at the graph stop point.
`CENTer`	Position the marker at the graph center point.
`DSPan`	Expand the area of the graph between the selected marker and its reference marker.

Parameter

Item	Type	Range of values	Default value
`<marker_mode>`	Discrete	STARt|STOP|CENTer|DSPan	Required parameter

Remarks

- This command is only applicable after the sweep data is acquired using the **INITiate[:IMMediate]:SWEep** command.
- The `DSPan` mode is only applicable for a selected marker which has a reference marker.

Example

The following command positions marker 2 at the graph start point.

```
CALC:SWE:MARK2:MODE STAR
```

CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:STATe

Syntax

```
CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:STATe <state>
```

```
CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:STATe?
```

Description

Turns on or off the selected marker on the graph display in the sweep mode. The selected marker will become the active marker when it is turned on. The query returns the marker state as `0` if the marker state is `OFF`, or `1` if the marker state is `ON`.

Parameter

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	OFF

Remark

This command is only applicable after the sweep data is acquired using the **INITiate[:IMMediate]:SWEep** command.

Examples

The following command turns on marker 2 on the graph display.

```
CALC:SWE:MARK2:STAT ON
```

The following query returns the state for marker 2.

```
CALC:SWE:MARK2:STAT?
```

Typical response:

```
1
```

CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:TRACe

Syntax

```
CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:TRACe <trace_no>
CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:TRACe?
```

Description

Assigns the marker to the trace of the specified channel on the graph display in the sweep mode. The trace number corresponds with the channel number. For example, trace `CHANnel1` represents the trace for channel 1. The selected marker will become the active marker. `R1`, `R2`, and `R3` correspond to the respective reference traces loaded into the graph. The query returns the trace number for the specified marker.

Parameter

Item	Type	Range of values	Default value
`<trace_no>`	Discrete	CHANnel1|CHANnel2|R1|R2|R3	CHANnel1

Remarks

- This command is only applicable after the sweep data is acquired using the **INITiate[:IMMediate]:SWEep** command.
- When the legacy sweep mode is activated, the trace value is automatically changed according to the sweep channel and cannot be configured to another channel. However, the trace can still be configured to the reference traces.

Examples

The following command assigns marker 1 to the channel 2 trace on the sweep display.

```
CALC:SWE:MARK:TRAC CHAN2
```

The following query returns the trace number for marker 1.

```
CALC:SWE:MARK:TRAC?
```

Typical response:

```
CHAN2
```

CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:X

Syntax

```
CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:X <x-position>
CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:X?
```

Description

Sets the marker X-axis value on the graph display in the sweep mode. The selected marker will become the active marker. The query returns the marker X-axis value. If the marker state is off, the response is not a number (NaN).

Parameter

Item	Type	Range of values	Default value
`<x-position>`	Numeric	–200000 to 200000	0

Remark

This command is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:SWEep** command.

Examples

The following command sets the marker 2 X-axis value to 550 Hz on the graph display.

```
CALC:SWE:MARK2:X 550
```

The following query returns the marker 2 X-axis value.

```
CALC:SWE:MARK2:X?
```

Typical response:

```
5.500000E+02
```

CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:XDELta?

Syntax

```
CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:XDELta?
```

Description

Returns the difference in the X-axis value between the selected marker and its reference marker on the graph display in the sweep mode.

Remarks

- This query is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:GRAPh** command.
- If the marker state is off, the response is not a number (NaN).
- If the selected marker has no reference marker, the response is also not a number (NaN).

Examples

The following query returns the delta X-axis value for marker 2.

```
CALC:SWE:MARK2:XDEL?
```

Typical response:

```
3.500000E+02
```

CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:Y?

Syntax

```
CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:Y?
```

Description

Returns the marker Y-axis value on the graph display in the sweep mode.

Remarks

- This query is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:SWEep** command.
- If the marker state is off, the response is not a number (NaN).

Example

The following query returns the marker 2 Y-axis value.

```
CALC:SWE:MARK2:Y?
```

Typical response:

```
0.000000E+00
```

CALCulate:SWEep:MARKer[1]|2|3|4|5|6|7|8:YDELta?

Syntax

```
CALCulate:SWE:MARKer[1]|2|3|4|5|6|7|8:YDELta?
```

Description

Returns the difference in the Y-axis value between the selected marker and its reference marker on the graph display in the sweep mode.

Remarks

- This query is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:SWEep** command.
- If the marker state is off, the response is not a number (NaN).
- If the selected marker has no reference marker, the response is also not a number (NaN).

Example

The following query returns the delta Y-axis value for marker 2.

```
CALC:SWE:MARK2:YDEL?
```

Typical response:

```
5.000000E+00
```

CALCulate:SWEep:MARKer:THReshold[:LEVel]

Syntax

```
CALCulate:SWEep:MARKer:THReshold[:LEVel] <threshold_level>
CALCulate:SWEep:MARKer:THReshold[:LEVel]?
```

Description

Sets the threshold level that the marker can identify as a peak or minimum on the graph display in the sweep mode. If the trace is above the threshold level, it will be identified as a peak, whereas the trace below the threshold level will be identified as a minimum. The query returns the specified threshold level.

Parameter

Item	Type	Range of values	Default value
<threshold_level>	Numeric	Within the top edge and bottom edge of the display	0

Remark

This command is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:SWEep** command.

Examples

The following command sets the threshold level to 20.

```
CALC:SWE:MARK:THR 20
```

The following query returns the threshold level.

```
CALC:SWE:MARK:THR?
```

Typical response:

```
2.000000E+01
```

CALCulate:SWEep:MARKer:THReshold:STATe

Syntax

```
CALCulate:SWEep:MARKer:THReshold:STATe <threshold_state>
```

```
CALCulate:SWEep:MARKer:THReshold:STATe?
```

Description

Turns on or off the threshold on the graph display in the sweep mode. The query returns the threshold state as `0` if the state is `OFF`, or `1` if the state is `ON`.

Parameter

Item	Type	Range of values	Default value
`<threshold_state>`	Boolean	OFF(0)|ON(1)	OFF

Remark

This command is only applicable after the graph data is acquired using the **INITiate[:IMMediate]:SWEep** command.

Examples

The following command turns on the threshold on the graph display.

```
CALC:SWE:MARK:THR:STAT ON
```

The following query returns the threshold state.

```
CALC:SWE:MARK:THR:STAT?
```

Typical response:

```
1
```

THIS PAGE HAS BEEN INTENTIONALLY LEFT BLANK.

5 DATA Subsystem

This chapter describes the `DATA` subsystem commands.

DATA:DIGital:FILTer

Syntax

```
DATA:DIGital:FILTer <filter_category>, <no_of_section>,
<no_of_group_delay>, <data>
```

Description

Downloads the 32-bit floating point user-defined filter data into the U8903B volatile memory allocated for the user-defined filter coefficients.

NOTE There is only one memory slot allocated for this function.

The `<data>` parameter represents the filter coefficients in the IEEE-488.2 binary block program data format, where the minimum number of bytes is 16 and maximum number of bytes is 1008. The maximum number of filter coefficients is 252 with 32 bits for each coefficient.

If IIR filter is selected:

- Minimum number of coefficient is 7 (1 section)
- Maximum number of coefficient is 252 (36 section)

If FIR filter is selected :

- Minimum number of coefficient is 4.
- Maximum number of coefficient is 252.

Parameter

Item	Type	Range of values	Default value
`<filter_category>`	Numeric	IIR|FIR	Required parameter
`<no_of_section>`	Numeric	1 to 36	Required parameter
`<no_of_group_delay>`	Numeric	0 to 65535	Required parameter

Remarks

- Refer to “**Appendix H: Using the IEEE-488.2 Binary Block Format**” on page 671 for details on the `<data>` format.
- This command must be sent prior to sending the **SENSe:DIGital:FILTer:LPASs** `CUSTom`, **SENSe:DIGital:FILTer:HPASs** `CUSTom`, **SENSe:DIGital:FILTer:WEIGhting** `CUSTom`, or **SENSe:DIGital:FILTer:DEEMphasis** `CUSTom commands`.
- The **DATA:DIGital:FILTer** command overwrites the previous filter data in the U8903B volatile memory.
- The downloaded filter data will remain in the U8903B volatile memory if the CUSTom filter is not changed to any other preset filters for the selected channel.
- Sending the **SYSTem:PRESet**, ***RST**, or **SYSTem:RESet[:MODE]** `DANalyzer` command, or cycling the U8903B power, will delete the downloaded filter data.
- If the filter type is Finite Impulse Response (FIR), the coefficients are arranged in the following manner.

  Coefficient[0] = A[0]
  Coefficient[1] = A[1]
  Coefficient[2] = A[2]
  Coefficient[3] = A[3]
  Coefficient[4] = A[4]
  Coefficient[5] = A[5]
  Coefficient[6] = A[6]
  ...

NOTE

The FIR filter transfer function, H(z), is defined as:

$H(z) = A[0] + A[1]z^{-1} + A[2]z^{-2} + A[3]z^{-3} + \ldots$

where z = complex variable

- If the filter type is Infinite Impulse Response (IIR), the coefficients are arranged in the following manner.

  Coefficient[0] = Section 1:Gain1
  Coefficient[1] = Section 1:A1[0]
  Coefficient[2] = Section 1:A1[1]
  Coefficient[3] = Section 1:A1[2]
  Coefficient[4] = Section 1:B1[0]
  Coefficient[5] = Section 1:B1[1]
  Coefficient[6] = Section 1:B1[2]
  Coefficient[0] = Section 2:Gain2
  Coefficient[1] = Section 2:A2[0]

Coefficient[2] = Section 2:A2[1]
Coefficient[3] = Section 2:A2[2]
Coefficient[4] = Section 2:B2[0]
Coefficient[5] = Section 2:B2[1]
Coefficient[6] = Section 2:B2[2]
...

where Ax = Denominator and Bx = Numerator

NOTE

The IIR filter transfer function, H(z), is defined as:

$$H(z) = \prod_{x=1}^{N} Gain_x \left(\frac{B_x[0] + B_x[1]z^{-1} + B_x[2]z^{-2}}{A_x[0] + A_x[1]z^{-1} + A_x[2]z^{-2}} \right)$$

where z = complex variable, N = number of sections, x = section number

– Each section must contain second-order filter coefficients.

Examples

The following command downloads the user- defined FIR low-pass filter data into the U8903B volatile memory.

```
DATA:DIG:FILT FIR, 1, 0, <data>
```

DATA:FILTer

Syntax

```
DATA:FILTer
<filter_category>,<no_of_section>,<no_of_group_delay>,<data>
```

Description

Downloads the 32-bit floating point user-defined filter data into the U8903B volatile memory allocated for the user-defined filter coefficients.

NOTE There is only one memory slot allocated for this function.

The `<data>` parameter represents the filter coefficients in the IEEE-488.2 binary block program data format, where the minimum number of bytes is 16 and maximum number of bytes is 1024. The maximum number of filter coefficients is 256 with 32 bits for each coefficient.

Parameters

Item	Type	Range of values	Default value
<filter_category>	Numeric	IIR|FIR	Required parameter
<no_of_section>	Numeric	1 to 36	Required parameter
<no_of_group_delay>	Numeric	0 to 65535	Required parameter

Remarks

– Refer to "**Appendix H: Using the IEEE-488.2 Binary Block Format**" on page 671 for details on the `<data>` format.

– IIR filter is Infinite Impulse Response and FIR filter is Finite Impulse Response.

– This command must be sent prior to sending the **SENSe[:ANALog]:FILTer:LPASs** `CUSTom`, **SENSe[:ANALog]:FILTer:HPASs** `CUSTom`, or **SENSe[:ANALog]:FILTer:WEIGhting** `CUSTom`.

- The **DATA:FILTer** command overwrites the previous filter data in the U8903B volatile memory.
- The downloaded filter data will remain in the U8903B volatile memory if the `CUSTom` filter is not changed to any other preset filters for the selected channel.
- Sending the **SYSTem:PRESet**, ***RST**, or **SYSTem:RESet[:MODE]** `AANalyzer` command, or cycling the U8903B power, will delete the downloaded filter data.
- If the filter type is `IIR`, the coefficients are arranged in the following manner.

  Coefficient[0] = Section 1:Gain1
  Coefficient[1] = Section 1:A1[0]
  Coefficient[2] = Section 1:A1[1]
  Coefficient[3] = Section 1:A1[2]
  Coefficient[4] = Section 1:B1[0]
  Coefficient[5] = Section 1:B1[1]
  Coefficient[6] = Section 1:B1[2]
  Coefficient[0] = Section 2:Gain2
  Coefficient[1] = Section 2:A2[0]
  Coefficient[2] = Section 2:A2[1]
  Coefficient[3] = Section 2:A2[2]
  Coefficient[4] = Section 2:B2[0]
  Coefficient[5] = Section 2:B2[1]
  Coefficient[6] = Section 2:B2[2]
  ...

  where Ax = Denominator and Bx = Numerator

NOTE

The IIR filter transfer function, H(z), is defined as:

$$\mathbf{H(z)} = \prod_{x=1}^{N} \mathrm{Gain}_x \left(\frac{B_x[0] + B_x[1]z^{-1} + B_x[2]z^{-2}}{A_x[0] + A_x[1]z^{-1} + A_x[2]z^{-2}} \right)$$

where z = complex variable, N = number of sections, x = section number

---

- If the filter type is `FIR`, the coefficients are arranged in the following manner.

  Coefficient[0] = A[0]
  Coefficient[1] = A[1]
  Coefficient[2] = A[2]
  Coefficient[3] = A[3]
  Coefficient[4] = A[4]
  Coefficient[5] = A[5]
  Coefficient[6] = A[6]
  ...

NOTE

The FIR filter transfer function, H(z), is defined as:

$H(z) = A[0] + A[1]z^{-1} + A[2]z^{-2} + A[3]z^{-3} + \ldots$

where z = complex variable

– Each section must contain second-order filter coefficients.

– Refer to “**Example 9: Use the User-Defined Filter Data**” on page 652 for the user-defined filter example.

Example

The following command downloads the user-defined FIR low-pass filter data into the U8903B volatile memory.

```
DATA:FILT FIR,1,0,<data>
```

DATA:FIRMware

Syntax

```
DATA:FIRMware <data>
```

Description

Performs the firmware update process based on the the downloaded `<data>`. The `<data>` parameter is in the IEEE-488.2 binary block program data format.

Remarks

- Refer to "**Appendix H: Using the IEEE-488.2 Binary Block Format**" on page 671 for details on the `<data>` format.
- The `<data>` is the firmware update file in the binary block data byte format.
- All the measurement and waveform generation will be stopped automatically before the update process.
- The time taken for this command to return depends on the firmware file, and it may take around 20 to 40 minutes to transfer and update. It is advisable to adjust the time-out accordingly.
- After the firmware update process is completed, the U8903B will restart automatically and the connection will be lost until the U8903B is turned on.
- It is advisable to send the **SYSTem:ERRor[:NEXT]?** command after this command to query the error in case the update process failed.

Example

The following command performs the firmware update process.

```
DATA:FIRM <data>
```

DATA:SWEep

Syntax

```
DATA:SWEep <data>
```

Description

Downloads the 32-bit floating point sweep data into the U8903B internal sweep memory. The `<data>` parameter is in the IEEE-488.2 binary block program data format.

Remarks

– Refer to ““**Appendix H: Using the IEEE-488.2 Binary Block Format**” on page 671 for details on the `<data>` format.

– The maximum number of allowable sweep points is 1024.

– Refer to “**Example 8: Perform Sweep**” on page 649 for the example of the List sweep mode.

– Sending the **SYSTem:PRESet**, ***RST**, or **SYSTem:RESet[:MODE]** `SWEep` command, or cycling the U8903B power, will delete the downloaded sweep data.

– If you select amplitude as the sweep parameter, your downloaded amplitude points are assumed in unit Vp.

– The command **SOURce:SWEep:CHANnel** must be sent prior to sending the **DATA:SWEep** command.

– When you send this command, the custom sweep points in the file will be loaded into the U8903B based on the source sweep channel set in the **SOURce:SWEep:CHANnel** command. The sweep mode will also be set to Automatic List or Manual list, depending on the previous sweep mode.

Example

The following command downloads the sweep data points into the U8903B internal sweep memory.

```
DATA:SWE <data>
```

DATA:TSEQuence:PROJect

Syntax

```
DATA:TSEQuence:PROJect <data>
```

```
DATA:TSEQuence:PROJect?
```

Description

Downloads the test sequence project settings into the U8903B. The current test sequence project settings in the U8903B will be overwritten. The query will return the current test sequence project settings in a definite length arbitrary block response data. The `<data>` parameter is in the IEEE-488.2 binary block program data format.

Remarks

– Refer to "**Appendix H: Using the IEEE-488.2 Binary Block Format**" on page 671 for details on the `<data>` format.

– The acceptable file type is restricted to the XML file format (.xml).

– The test sequence must be in the STOP state before this command is allowed.

Examples

The following command downloads the XML file data into the test sequence mode.

```
DATA:TSEQ:PROJ <data>
```

The following query returns the current test sequence project settings.

```
DATA:TSEQ:PROJ?
```

DATA:TSEQuence:REPort

Syntax

```
DATA:TSEQuence:REPort?
```

Description

Queries for the test sequence report. The query will return the existing test sequence report in a DOCX file format.

Remarks

- The returned file type is restricted to the DOCX file format (.docx).
- The test sequence must be in the STOP state before this command is allowed.
- If the test sequence report is not available when queried, error –200, “Execution Error” will be generated.

Example

The following query returns the existing test sequence report.

```
DATA:TSEQ:REP?
```

DATA:WAVeform

Syntax

```
DATA:WAVeform <Vpeak>,<DC_offset>,<data>
```

Description

Downloads the 32-bit floating point arbitrary waveform data into the U8903B internal waveform memory.

You can download from 32 to 32768 (32K) points per waveform. The data value must be the normalized data between –1 to 1. The values of –1 and +1 correspond to the peak values of the waveform (if the offset is 0 V). For example, if you set the Vpeak to 5 Vp (0 V offset), +1 corresponds to +5 Vp. The `<data>` parameter is in the IEEE-488.2 binary block program data format.

Parameters

Item	Type	Range of values	Default value
`<Vpeak>`	Numeric	– 0 to 22.6 Vp (balanced output connection) – 0 to 11.3 Vp (unbalanced or common output connection)	Required parameter
`<DC_offset>`	Numeric	–11.3 V to 11.3 V	Required parameter

Remarks

– Refer to “**Appendix H: Using the IEEE-488.2 Binary Block Format**” on page 671 for details on the `<data>` format.

– Refer to “**Example 3: Generate Arbitrary Waveform**” on page 644 for the arbitrary waveform example.

– The **DATA:WAVeform** command overwrites the previous waveform data in the U8903B volatile memory.

– Sending the **SYSTem:PRESet**, ***RST**, or **SYSTem:RESet[:MODE]** `AGENerator` command, or cycling the U8903B power, will delete the downloaded waveform data.

Example

The following command downloads the arbitrary waveform data into the instrument

```
DATA:WAV 5,0,<data>
```

DATA:WAVFile

Syntax

```
DATA:WAVFile <data>
```

Description

Downloads a Microsoft compatible wave file (.wav) to the U8903B buffer. The `<data>` parameter is in the IEEE-488.2 binary block program data format.

Remarks

- Refer to "**Appendix H: Using the IEEE-488.2 Binary Block Format**" on page 671 for details on the `<data>` format.
- The acceptable file type is restricted to the wave file format (.wav).
- The size of the wave file is limited to 5 MB, and the supported data resolution is 8, 16, and 24 bits per sample.
- According to the definition of the wave file format, the PCM data is two's-complement except for resolutions of 1 to 8 bits, which are represented as offset binary. Therefore, for a wave file with 8 bits per sample resolution, the data will automatically be converted to two's-complement at the output.

Example

The following command downloads the wave file data into the analog generator arbitrary waveform.

```
DATA:WAVF <data>
```

THIS PAGE HAS BEEN INTENTIONALLY LEFT BLANK.

6 DISPlay Subsystem

This chapter describes the `DISPlay` subsystem commands.

DISPlay:ANALysis:TYPE

Syntax

```
DISPlay:ANALysis:TYPE <mode>[,<mode2>]
DISPlay:ANALysis:TYPE?
```

Description

Sets the graph display panel analysis mode as either time domain, frequency domain (magnitude), frequency domain (phase), or frequency domain (power spectral density). The query returns the graph display mode in the form of `MAGN`, `PHAS`, `TIME`, and `PSD` . Multiple responses are separated by commas.

The types of configuration are listed as follows:

`MAGNitude`	Magnitude in frequency domain
`PHASe`	Phase in frequency domain
`TIME`	Signal waveform in time domain
`PSD`	Power Spectral Density in frequency domain

Parameter

Item	Type	Range of values	Default value
`<mode>`	Discrete	MAGNitude|PHASe|TIME|PSD	MAGNitude

Remarks

- Sample size is not changed when this command is sent.
- The first `<mode>` parameter will be set to the first graph display panel, while the second `<mode>` parameter will be set to the second graph display panel. If only one mode parameter is specified, the parameter will be set to the current active panel.
- The query will always return two values when queried.
- PSD analysis is not applicable for digital channels.

Examples

The following command sets the first graph display panel as frequency domain (magnitude) and the second graph display panel as time domain.

```
DISP:ANAL:TYPE MAGN,TIME
```

The following query returns the analysis mode for the graph display panels.

```
DISP:ANAL:TYPE?
```

Typical response:

```
MAGN,TIME
```

DISPlay[:WINDow]:GRAPh:HARMonic:CHANnel

Syntax

```
DISPlay[:WINDow]:GRAPh:HARMonic:CHANnel <state>,(@<channel_list>)
DISPlay[:WINDow]:GRAPh:HARMonic:CHANnel? (@<channel_list>)
```

Description

Displays or hides the specified channel(s) information in the bar chart of the harmonic view. The query returns the state of the selected channel(s) information in the bar chart of the harmonic view. This command is only useful when the display option in the graph is set to the harmonic view and a set of valid data exists in the buffer.

Parameters

Item	Type	Range of values	Default value
`<state>`	Discrete	ON|OFF	ON
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Examples

The following command displays channel 2 information in the bar chart.

```
DISP:GRAP:HARM:CHAN ON,(@2)
```

The following query returns the state of channel 3 and channel 4 in the bar chart.

```
DISP:GRAP:HARM:CHAN? (@3,4)
```

Typical response:

```
ON
```

DISPlay[:WINDow]:GRAPh:STATe

Syntax

```
DISPlay[:WINDow]:GRAPh    <state>
DISPlay[:WINDow]:GRAPh?
```

Description

Enables or disables the refresh of the graph display in the graph page. The query returns a 1 or 0.

Parameters

Item	Type	Range of values	Default value
`<state>`	Boolean	(ON|1|OFF|0)	ON

Remark

- Sending the **SYSTem:PRESet** or ***RST** command, or cycling the U8903B power, will enable the auto refresh.
- When disabled, other DISPlay subsystem commands may not function correctly.

Examples

The following command enables the graph refresh on the graph page.

```
DISP:GRAP:STAT ON
```

The following query returns the graph refresh state of the graph page.

```
DISP:GRAP:STAT
```

Typical response:

```
1
```

DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:AXIS

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:AXIS <axis_type>
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:AXIS?
```

Description

Assigns the trace to use the primary or secondary axis for plotting purpose. The query returns the axis of the specified trace.

Parameter

Item	Type	Range of values	Default value
<axis_type>	Discrete	PRIMary|SECondary	PRIMary

Examples

The following command displays trace 1 with the primary axis.

```
DISP:GRAP:TRAC:AXIS PRIM
```

The following query returns the axis of trace 2.

```
DISP:GRAP:TRAC2:AXIS?
```

Typical response:

```
PRIM
```

DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:COLor

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:COLor <color_type>
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:COLor?
```

Description

Assigns the trace color. The query returns the color of the specified trace.

Parameter

Item	Type	Range of values	Default value
`<color_type>`	Discrete	YELLow|CYAN|WHITe|PINK|GREEn|ORANGe|RED|PURPle	YELLow

Examples

The following command sets the trace 1 with the color cyan.

```
DISP:GRAP:TRAC:COL CYAN
```

The following query returns the color of trace 2.

```
DISP:GRAP:TRAC2:COL?
```

Typical response:

```
YELL
```

DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:MATH:FUNCtion

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:MATH:FUNCtion
<function>
```

```
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:MATH:FUNCtion?
```

Description

Sets the math function for the specified trace. The query returns the math function of the specified trace.

Parameter

Item	Type	Range of values	Default value
<function>	Discrete	NONE|ADD|SUBTract|MULTiplay|DIVision	NONE

Examples

The following command sets the trace 2 math function to ADD.

```
DISP:GRAP:TRAC2:MATH:FUNC ADD
```

The following query returns the math function of trace 2.

```
DISP:GRAP:TRAC2:MATH:FUNC?
```

Typical response:

```
ADD
```

DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:MATH:STATe

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:MATH:STATe <state>
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:MATH:STATe?
```

Description

Turns on or off the math function for the specified trace. The query returns the state of the math function of the specified trace.

Parameter

Item	Type	Range of values	Default value
<state>	Discrete	ON|OFF	OFF

Examples

The following command turns on the trace 2 math function.

```
DISP:GRAP:TRAC2:MATH:STAT ON
```

The following query returns the math function state of trace 2.

```
DISP:GRAP:TRAC2:MATH:STAT?
```

Typical response:

```
ON
```

DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:MATH:VARiable

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:MATH:VARiable
<variable>
```

```
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:MATH:VARiable?
```

Description

Sets the math function variable for the specified trace. The query returns the math function variable of the specified trace.

Parameter

Item	Type	Range of values	Default value
<variable>	Numeric	–∞ to ∞	0

Examples

The following command sets the trace 2 math function variable to 10.

```
DISP:GRAP:TRAC2:MATH:VAR 10
```

The following query returns the math function variable of trace 2.

```
DISP:GRAP:TRAC2:MATH:VAR?
```

Typical response:

```
10
```

DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:PERSistence:COUNt

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:PERSistence:COUNt
<count>
```

```
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:PERSistence:COUNt?
```

Description

Sets the number of trace to persist for the specified trace. The query returns the number of trace to persist for the specified trace

Parameter

Item	Type	Range of values	Default value
<count>	Numeric	1 to 10	5

Examples

The following command persists 5 sets of data for trace 1.

```
DISP:GRAP:TRAC:PERS:COUN 5
```

The following query returns the number of sets configured to persist for trace 2.

```
DISP:GRAP:TRAC2:PERS:COUN?
```

Typical response:

```
5
```

DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:PERSistence:STATe

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:PERSistence:STATe
<state>
```

```
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:PERSistence:STATe?
```

Description

Turns on or off the persistence function for the specified trace. The query returns the state of the persistence function of the specified trace.

Parameter

Item	Type	Range of values	Default value
<state>	Discrete	ON|OFF	OFF

Examples

The following command turns on the persistence function for trace 1.

```
DISP:GRAP:TRAC:PERS:STAT ON
```

The following query returns the persistence function state of trace 1.

```
DISP:GRAP:TRAC:PERS:STAT?
```

Typical response:

```
ON
```

DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:SOURce

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:SOURce <source>
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:SOURce?
```

Description

Assigns the source for the specified trace slot. The query returns the source of the specified trace slot.

Parameter

Item	Type	Range of values	Default value
<source>	Discrete	AA1|AA2|AA3|AA4|AA5|AA6|AA7|AA8|DA1|DA2|T1|T2|T3|T4|T5|T6|T7|T8|FILE	AA1

Examples

The following command sets AA1 as the source for trace 1.

```
DISP:GRAP:TRAC:SOUR AA1
```

The following query returns the source of trace 2.

```
DISP:GRAP:TRAC2:SOUR?
```

Typical response:

```
AA1
```

DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:STATe

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:STATe <state>
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:STATe?
```

Description

Displays or hides the trace in the graph. The query returns the state of the specified trace.

Parameter

Item	Type	Range of values	Default value
`<state>`	Discrete	ON|OFF	OFF

Examples

The following command displays the trace 2 in the graph analysis mode.

```
DISP:GRAP:TRAC2:STAT ON
```

The following query returns the state of trace 2.

```
DISP:GRAP:TRAC3:STAT?
```

Typical response:

```
ON
```

DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:UNIT

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:UNIT <unit_type>
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:UNIT?
```

Description

Assigns the unit of the data for the specified trace. The query returns the unit of the data for the specified trace.

Parameter

Item	Type	Range of values	Default value
<unit_type>	Discrete	V|dBV|W|dBU|dBm|DEGree|RADian|dBHZ	dBV (Magnitude) V (Time) RADian (Phase) dBHz (PSD)

Examples

The following command sets the unit of the data for trace 2 to dBV.

```
DISP:GRAP:TRAC2:UNIT dBV
```

The following query returns the unit of the data for trace 2.

```
DISP:GRAP:TRAC3:UNIT?
```

Typical response:

```
dBV
```

DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:X:VALues?

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:X:VALues?
```

Description

Returns the X-axis point of the specified trace in the IEEE-488.2 binary block format..

Examples

The following query returns the X-axis point of trace 3 in the graph analysis mode.

```
DISP:GRAP:TRAC3:X:VAL?
```

DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:Y:VALues?

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe[1]|2|3|4|5|6|7|8:Y:VALues?
```

Description

Returns the Y-axis point of the specified trace in the IEEE-488.2 binary block format..

Examples

The following query returns the Y-axis point of trace 1 in the graph analysis mode.

```
DISP:GRAP:TRAC:Y:VAL?
```

DISPlay[:WINDow]:GRAPh:TRACe:HOLD

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe:HOLD <hold_type>,(@<channel>)
DISPlay[:WINDow]:GRAPh:TRACe:HOLD? (@<channel>)
```

Description

Sets the hold configuration for the specified channel(s). The query returns the hold configuration type of the selected channel(s). Multiple responses are separated by commas.

The types of hold configuration are listed as follows:

NONE	Clear and write Each update replaces the old data with the new data.
AVERage	Trace average A cumulative average on a point-by-point basis of the new data and previous averaged data is displayed.
MAX	Maximum hold A representation of a maximum data value on a point-by-point basis of the new data and previous data is displayed.
MIN	Minimum hold A representation of a minimum data value on a point-by-point basis of the new data and previous data is displayed.

Parameters

Item	Type	Range of values	Default value
<hold_type>	Discrete	NONE|AVERage|MAX|MIN	NONE
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Examples

The following command sets the hold configuration to MAX for channel 1 in the graph analysis mode.

```
DISP:GRAP:TRAC:HOLD MAX,(@1)
```

The following query returns the hold configuration for channel 1 and channel 2 in the graph analysis mode.

```
DISP:GRAP:TRAC:HOLD? (@1,2)
```

Typical response:

```
MAX,NONE
```

DISPlay[:WINDow]:GRAPh:TRACe:AUTO

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe:AUTO
```

Description

Performs an autoscale to automatically scale the graph display according to the signal each time this command is sent.

Remarks

Autoscaling may affect the values of any parameters under the SCALe node.

Examples

The following command performs an autoscale on the graph.

```
DISP:GRAP:TRAC:AUTO
```

DISPlay[:WINDow]:GRAPh:TRACe:AXIS:ACTive

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe:AXIS:ACTive <axis_type>
DISPlay[:WINDow]:GRAPh:TRACe:AXIS:ACTive?
```

Description

Sets the active axis type of the selected graph panel. The query returns the active x-axis type of the selected graph panel.

Parameter

Item	Type	Range of values	Default value
<axis_type>	Discrete	PRIMary|SECondary	PRIMary

Examples

The following command selects the secondary axis of the selected graph panel.

```
DISP:GRAP:TRAC:AXIS:ACT SEC
```

The following query returns the active axis type of the selected graph panel.

```
DISP:GRAP:TRAC:AXIS:ACT?
```

Typical response:

```
SEC
```

DISPlay[:WINDow]:GRAPh:TRACe:AXIS:STATe

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe:AXIS:STATe <state>
DISPlay[:WINDow]:GRAPh:TRACe:AXIS:STATe?
```

Description

Sets the active axis visible state of the selected graph panel. The query returns the active axis visible state of the selected graph panel.

Parameter

Item	Type	Range of values	Default value
<state>	Discrete	ON|OFF	ON

Remark

The visible state of the primary axis will always be set to ON.

Examples

The following command sets the active axis visible state of the selected graph panel to ON.

```
DISP:GRAP:TRAC:AXIS:STAT ON
```

The following query returns the active axis visible state of the selected graph panel.

```
DISP:GRAP:TRAC:AXIS:STAT?
```

Typical response:

```
ON
```

DISPlay[:WINDow]:GRAPh:TRACe:REFerence:SLOT

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe:REFerence:SLOT <slot_number>
DISPlay[:WINDow]:GRAPh:TRACe:REFerence:SLOT?
```

Description

Sets the reference trace slot in the graph analysis mode. The query returns the reference trace slot.

Parameter

Item	Type	Range of values	Default value
<slot_number>	Numeric	1|2|3	1

Examples

The following command sets the reference trace slot to 2.

```
DISP:GRAP:TRAC:REF:SLOT 2
```

The following query returns the reference trace slot.

```
DISP:GRAP:TRAC:REF:SLOT?
```

Typical response:

```
2
```

DISPlay[:WINDow]:GRAPh:TRACe:REFerence:SOURce

Syntax

```
DISPlay:[:WINDow]:GRAPh:TRACe:REFerence:SOURce
<source_type>,<filename>
```

```
DISPlay:[:WINDow]:GRAPh:TRACe:REFerence:SOURce?
```

Description

Sets the source for the current reference trace slot. The query returns the source for the current reference trace slot.

Parameters

Item	Type	Range of values	Default value
<source_type>	Discrete	NONE|FILE|CH1|CH2	NONE
<filename>	String	Full file path name. Only used if source_type is FILE. For example, "\Storage 1\ file.csv". Left blank if source type is channel 1 or channel 2.	Required parameter

Examples

The following command sets the source for the current reference slot to File and named 'file.csv' in the '\Storage 1' directory.

```
DISP:GRAP:TRAC:REF:SOUR FILE,"\Storage 1\file.csv"
```

The following query returns the source for the current reference slot.

```
DISP:GRAP:TRAC:REF:SOUR?
```

Typical response:

```
FILE
```

DISPlay[:WINDow]:GRAPh:TRACe:REFerence:STATe

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe:REFerence:STATe <state>
DISPlay[:WINDow]:GRAPh:TRACe:REFerence:STATe?
```

Description

Enables or disables the reference trace in the graph analysis mode. The query returns the reference trace state.

Parameter

Item	Type	Range of values	Default value
`<state>`	Discrete	ON|OFF	ON

Examples

The following command enables the reference trace in the graph analysis mode.

```
DISP:GRAP:TRAC:REF:STAT ON
```

The following query returns the reference trace state.

```
DISP:GRAP:TRAC:REF:STAT?
```

Typical response:

```
ON
```

DISPlay[:WINDow]:GRAPh:TRACe:X[:SCALe]:AUTO

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe:X[:SCALe]:AUTO
```

Description

Performs an autoscale on the X-axis to automatically scale the graph display according to the signal each time this command is sent.

Example

The following command performs an autoscale on the X-axis for the graph view in the graph analysis mode.

```
DISP:GRAP:TRAC:X:AUTO
```

DISPlay[:WINDow]:GRAPh:TRACe:X[:SCALe]:LEFT

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe:X[:SCALe]:LEFT <minimum_limit>
```

```
DISPlay[:WINDow]:GRAPh:TRACe:X[:SCALe]:LEFT?
```

Description

Sets the value represented by the minimum (left) edge of the X-axis. The query returns the left X-axis minimum edge.

Parameter

Item	Type	Range of values	Default value
`<minimum_limit>`	Numeric	Frequency domain (standard bandwidth) – 1 to 96 k Frequency domain (wide bandwidth) – 1 to 768 k Time domain (standard bandwidth) – 0 to 10.922 (time domain) Time domain (wide bandwidth) – 0 to 1.3653 (time domain)	 1 1 0 0

Remark

If the X-axis minimum edge value is more than the X-axis maximum edge value or the X-axis minimum edge value is equals to the X-axis maximum edge value, the X-axis maximum edge value will be set to a default span value of 10.

Examples

The following command sets the left X-axis minimum edge to 100 for the graph view in the graph analysis mode.

```
DISP:GRAP:TRAC:X:LEFT 100
```

The following query returns the left X-axis minimum edge for the graph view in the graph analysis mode.

```
DISP:GRAP:TRAC:X:LEFT?
```

Typical response:

```
1.000000E+02
```

DISPlay[:WINDow]:GRAPh:TRACe:X[:SCALe]:RIGHt

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe:X[:SCALe]:RIGHt <maximum_limit>
DISPlay[:WINDow]:GRAPh:TRACe:X[:SCALe]:RIGHt?
```

Description

Sets the value represented by the maximum (right) edge of the X-axis. The query returns the right X-axis maximum edge.

Parameter

Item	Type	Range of values	Default value
<maximum_limit>	Numeric	Frequency domain (standard bandwidth) – 1 to 96 k Frequency domain (wide bandwidth) – 1 to 768 k Time domain (standard bandwidth) – 0 to 10.922 (time domain) Time domain (wide bandwidth) – 0 to 1.3653 (time domain)	 1 1 0 0

Remark

If the X-axis minimum edge value is more than the X-axis maximum edge value or the X-axis minimum edge value is equals to the X-axis maximum edge value, the X-axis maximum edge value will be set to a default span value of 10.

Examples

The following command sets the right X-axis maximum edge to 10000 for the graph view in the graph analysis mode.

```
DISP:GRAP:TRAC:X:RIGH 10000
```

The following query returns the right X-axis maximum edge for the graph view in the graph analysis mode.

```
DISP:GRAP:TRAC:X:RIGH?
```

Typical response:

```
1.000000E+04
```

DISPlay[:WINDow]:GRAPh:TRACe:X:SPACing

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe:X:SPACing <spacing_type>
```

```
DISPlay[:WINDow]:GRAPh:TRACe:X:SPACing?
```

Description

Sets the X-axis spacing as either linear or logarithmic scaling. The query returns the X-axis spacing type in the form of LIN or LOG.

The X-axis spacing can be set at any time to determine how the graph data will be displayed. Setting the X-axis spacing will not restart the graph or trace data or change the number of graph points. The markers will stay at the respective X-axis values and move accordingly on the display when the X-axis spacing is set.

Parameter

Item	Type	Range of values	Default value
<spacing_type>	Discrete	LINear|LOGarithmic	LOGarithmic

Remark

The saved trace data when the X-axis spacing is set to linear will be identical to the saved trace data when the X-axis spacing is set to logarithmic. When recalling the saved trace data, the current value of the logarithmic or linear scaling will be used to display the data.

Examples

The following command sets the X-axis spacing to logarithmic for the graph view in the graph analysis mode.

```
DISP:GRAP:TRAC:X:SPAC LOG
```

The following query returns the X-axis spacing type for the graph view in the graph analysis mode.

```
DISP:GRAP:TRAC:X:SPAC?
```

Typical response:

```
LOG
```

DISPlay[:WINDow]:GRAPh:TRACe:Y[:SCALe]:AUTO

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe:Y[:SCALe]:AUTO
```

Description

Performs an autoscale on the Y-axis to automatically scale the graph display according to the signal each time this command is sent.

Example

The following command performs an autoscale on the Y-axis for the graph view in the graph analysis mode.

```
DISP:GRAP:TRAC:Y:AUTO
```

DISPlay[:WINDow]:GRAPh:TRACe:Y[:SCALe]:BOTTom

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe:Y[:SCALe]:BOTTom <minimum_limit>
DISPlay[:WINDow]:GRAPh:TRACe:Y[:SCALe]:BOTTom?
```

Description

Sets the value represented by the minimum (bottom) edge of the Y-axis. The query returns the bottom Y-axis minimum edge.

Parameter

Item	Type	Range of values	Default value
<minimum_limit>	Numeric	–∞ to ∞	1

Remarks

– If the Y-axis minimum edge value is more than the Y-axis maximum edge value or the Y-axis minimum edge value is equals to the Y-axis maximum edge value, the Y-axis maximum edge value will be set to 10 to maintain a relative difference to the Y-axis minimum edge value.

– If the Y-axis spacing is changed from linear to logarithmic scaling and the current Y-axis minimum edge value is 0 or in the negative region, the Y-axis minimum edge value will be set to 0.1.

Examples

The following command sets the bottom Y-axis minimum edge to –200 for the graph view in the graph analysis mode.

```
DISP:GRAP:TRAC:Y:BOTT –200
```

The query returns the bottom Y-axis minimum edge for the graph view in the graph analysis mode.

```
DISP:GRAP:TRAC:Y:BOTT?
```

Typical response:

```
–2.000000E+02
```

DISPlay[:WINDow]:GRAPh:TRACe:Y[:SCALe]:TOP

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe:Y[:SCALe]:TOP <maximum_limit>
```

```
DISPlay[:WINDow]:GRAPh:TRACe:Y[:SCALe]:TOP?
```

Description

Sets the value represented by the maximum (top) edge of the Y-axis. The query returns the top Y-axis maximum edge.

Parameter

Item	Type	Range of values	Default value
<maximum_limit>	Numeric	–∞ to ∞	0

Remarks

– If the Y-axis minimum edge value is more than the Y-axis maximum edge value or the Y-axis minimum edge value is equals to the Y-axis maximum edge value, the Y-axis maximum edge value will be set to 10 to maintain a relative difference to the Y-axis minimum edge value.

– If the Y-axis spacing is changed from linear to logarithmic scaling and the current Y-axis minimum edge value is 0 or in the negative region, the Y-axis minimum edge value will be set to 0.1.

Examples

The following command sets the top Y-axis maximum edge to 200 for the graph view in the graph analysis mode.

```
DISP:GRAP:TRAC:Y:TOP 200
```

The following query returns the top Y-axis maximum edge for the graph view in the graph analysis mode.

```
DISP:GRAP:TRAC:Y:TOP?
```

Typical response:

```
2.000000E+02
```

DISPlay[:WINDow]:GRAPh:TRACe:Y:SPACing

Syntax

```
DISPlay[:WINDow]:GRAPh:TRACe:Y:SPACing <spacing_type>
DISPlay[:WINDow]:GRAPh:TRACe:Y:SPACing?
```

Description

Sets the Y-axis spacing as either linear or logarithmic scaling. The query returns the Y-axis spacing type in the form of LIN or LOG.

The Y-axis spacing can be set at any time to determine how the graph data will be displayed. Setting the Y-axis spacing will not restart the graph or trace data or change the number of graph points. The markers will stay at the respective Y-axis values and move accordingly on the display when the Y-axis spacing is set.

Parameter

Item	Type	Range of values	Default value
<spacing_type>	Discrete	LINear|LOGarithmic	LINear

Remark

The saved trace data when the Y-axis spacing is set to linear will be identical to the saved trace data when the Y-axis spacing is set to logarithmic. When recalling the saved trace data, the current value of the logarithmic or linear scaling will be used to display the data.

Examples

The following command sets the Y-axis spacing to logarithmic for the graph view in the graph analysis mode.

```
DISP:GRAP:TRAC:Y:SPAC LOG
```

The following query returns the Y-axis spacing type for the graph view in the graph analysis mode.

```
DISP:GRAP:TRAC:Y:SPAC?
```

Typical response:

```
LOG
```

DISPlay[:WINDow]:MODE

Syntax

```
DISPlay[:WINDow]:MODE <display_mode>
```

```
DISPlay[:WINDow]:MODE?
```

Description

Sets the display mode of the front panel display. The query returns the current display mode.

Parameter

Item	Type	Range of values	Default value
`<display_mode>`	Discrete	VIEW1|VIEW2|VIEW4|VIEW10	VIEW2

Remarks

– This command is only valid in the analyzer, generator, and graph analysis modes.

– For analyzer and generator modes, the <display_mode> parameter is restricted to `VIEW2`, `VIEW4`, and `VIEW10`.

– For graph analysis mode, the `<display_mode>` parameter is restricted to `VIEW1` and `VIEW2`.

Examples

The following command sets the front panel display mode to 2-view.

```
DISP:MODE VIEW2
```

The following query returns the current display mode.

```
DISP:MODE?
```

Typical response:

```
VIEW2
```

DISPlay[:WINDow]:STATe

Syntax

```
DISPlay[:WINDow]:STATe <state>
```

```
DISPlay[:WINDow]:STATe?
```

Description

Enables or disables the front panel LCD backlight. The query returns the LCD backlight state as `0` if the state is `OFF`, or `1` if the state is `ON`.

Parameter

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	ON

Remark

Sending the **SYSTem:PRESet** or ***RST** command, or cycling the U8903B power, will enable the LCD backlight.

Examples

The following command enables the front panel LCD backlight.

```
DISP:STAT ON
```

The following query returns the LCD backlight state.

```
DISP:STAT?
```

Typical response:

```
1
```

DISPlay[:WINDow]:SWEep:TRACe:AUTO

Syntax

```
DISPlay[:WINDow]:SWEep:TRACe:AUTO
```

Description

Performs an autoscale on the sweep plot to automatically scale the sweep plot according to the signal each time this command is sent.

Remark

Rescaling the sweep plot will affect the parameter values under the [:SCALe] node.

Example

The following command performs an autoscale on the sweep plot.

```
DISP:SWE:TRAC:AUTO
```

DISPlay[:WINDow]:SWEep:TRACe:HOLD

Syntax

```
DISPlay[:WINDow]:SWEep:TRACe:HOLD <hold_type>
```

```
DISPlay[:WINDow]:SWEep:TRACe:HOLD?
```

Description

Sets the hold configuration in sweep mode. The query returns the hold configuration type in the sweep mode. Multiple responses are separated by commas.

Parameter

Item	Type	Range of values	Default value
<hold_type>	Discrete	NONE|MAX|MIN	NONE

Examples

The following command sets the hold configuration to MAX in the sweep mode.

```
DISP[:WIND]:SWE:TRAC:HOLD MAX
```

The following query returns the hold configuration type in the sweep mode.

```
DISP[:WIND]:SWE:TRAC:HOLD?
```

Typical response:

```
MAX
```

DISPlay[:WINDow]:SWEep:TRACe:REFerence:SLOT

Syntax

```
DISPlay[:WINDow]:SWEep:TRACe:REFerence:SLOT <slot_number>
```

```
DISPlay[:WINDow]:SWEep:TRACe:REFerence:SLOT?
```

Description

Sets the reference trace slot in the sweep mode. The query returns the reference trace slot.

Parameter

Item	Type	Range of values	Default value
<slot_number>	Numeric	1|2|3	1

Examples

The following command sets the reference trace slot to 2 in the sweep mode.

```
DISP:SWE:TRAC:REF:SLOT 2
```

The following query returns the reference trace slot in the sweep mode.

```
DISP:SWE:TRAC:REF:SLOT?
```

Typical response:

```
2
```

DISPlay[:WINDow]:SWEep:TRACe:REFerence:SOURce

Syntax

```
DISPlay:[:WINDow]:SWEep:TRACe:REFerence:SOURce
<source_type>,<filename>
```

```
DISPlay:[:WINDow]:SWEep:TRACe:REFerence:SOURce?
```

Description

Sets the source for the current reference trace slot in the sweep mode. The query returns the source for the current reference trace slot in the sweep mode.

Parameters

Item	Type	Range of values	Default value
<source_type>	Discrete	NONE|FILE|CH1|CH2	NONE
<filename>	String	Ful file path name. Only used if source_type is FILE. For example, “\Storage 1\ file.csv”. Left blank if source type is channel 1 or channel 2.	Required parameter

Examples

The following command sets the source for the current reference slot to File and named ‘file.csv’ in the ‘\Storage 1’ directory in the sweep mode.

```
DISP:SWE:TRAC:REF:SOUR FILE,“\Storage 1\file.csv”
```

The following query returns the source for the current reference slot in the sweep mode.

```
DISP:SWE:TRAC:REF:SOUR?
```

Typical response:

```
FILE
```

DISPlay[:WINDow]:SWEep:TRACe:REFerence:STATe

Syntax

```
DISPlay[:WINDow]:SWEep:TRACe:REFerence:STATe <state>
```

```
DISPlay[:WINDow]:SWeep:TRACe:REFerence:STATe?
```

Description

Enables or disables the reference trace in the sweep mode. The query returns the reference trace state.

Parameter

Item	Type	Range of values	Default value
<state>	Discrete	ON|OFF	ON

Examples

The following command enables the reference trace in the sweep mode.

```
DISP:SWE:TRAC:REF:STAT ON
```

The following query returns the reference trace state in the sweep mode.

```
DISP:SWE:TRAC:REF:STAT?
```

Typical response:

```
ON
```

DISPlay[:WINDow]:SWEep:TRACe:X[:SCALe]:AUTO

Syntax

```
DISPlay[:WINDow]:SWEep:TRACe:X[:SCALe]:AUTO
```

Description

Performs an autoscale on the X-axis to automatically scale the graph display according to the signal each time this command is sent.

Examples

The following command performs an autoscale on the X-axis for the graph view in the sweep mode.

```
DISP:SWE:TRAC:X:AUTO
```

DISPlay[:WINDow]:SWEep:TRACe:X[:SCALe]:LEFT

Syntax

```
DISPlay[:WINDow]:SWEep:TRACe:X[:SCALe]:LEFT <minimum_limit>
```

```
DISPlay[:WINDow]:SWEep:TRACe:X[:SCALe]:LEFT?
```

Description

Sets the value represented by the minimum (left) edge of the X-axis. The query returns the left X-axis minimum edge.

Parameter

Item	Type	Range of values	Default value
`<minimum_limit>`	Numeric	–200000 to 200000	20

Examples

The following command sets the left X-axis minimum edge to 5 for the graph view in the sweep mode.

```
DISP:SWE:TRAC:X:LEFT 5
```

The following query returns the left X-axis minimum edge for the graph view in the sweep mode.

```
DISP:SWE:TRAC:X:LEFT?
```

Typical response:

```
5.000000E+00
```

DISPlay[:WINDow]:SWEep:TRACe:X[:SCALe]:RIGHt

Syntax

```
DISPlay[:WINDow]:SWEep:TRACe:X[:SCALe]:RIGHt <maximum_limit>
```

```
DISPlay[:WINDow]:SWEep:TRACe:X[:SCALe]:RIGHt?
```

Description

Sets the value represented by the maximum (right) edge of the X-axis. The query returns the right X-axis maximum edge.

Parameter

Item	Type	Range of values	Default value
<maximum_limit>	Numeric	–200000 to 200000	20000

Examples

The following command sets the right X-axis maximum edge to 10000 for the graph view in the sweep mode.

```
DISP:SWE:TRAC:X:RIGH 10000
```

The following query returns the right X-axis maximum edge for the graph view in the sweep mode.

```
DISP:SWE:TRAC:X:RIGH?
```

Typical response:

```
1.000000E+04
```

DISPlay[:WINDow]:SWEep:TRACe:X:SPACing

Syntax

```
DISPlay[:WINDow]:SWEep:TRACe:X:SPACing <spacing_type>
```

```
DISPlay[:WINDow]:SWEep:TRACe:X:SPACing?
```

Description

Sets the X-axis spacing as either linear or logarithmic scaling. The query returns the X-axis spacing type in the form of `LIN` or `LOG`.

Parameter

Item	Type	Range of values	Default value
<spacing_type>	Discrete	LINear|LOGarithmic	LINear

Examples

The following command sets the X-axis spacing to logarithmic for the graph view in the sweep mode.

```
DISP:SWE:TRAC:X:SPAC LOG
```

The following query returns the X-axis spacing type for the graph view in the sweep mode.

```
DISP:SWE:TRAC:X:SPAC?
```

Typical response:

```
LOG
```

DISPlay[:WINDow]:SWEep:TRACe:X:SOURce

Syntax

```
DISPlay[:WINDow]:SWEep:TRACe:X:SOURce <data_source>
```

```
DISPlay[:WINDow]:SWEep:TRACe:X:SOURce?
```

Description

Sets the X-axis data source to be displayed on the graph in the sweep view. The query returns the X-axis data source.

GEN	Generator
FUNC1	Analyzer function 1
FUNC2	Analyzer function 2
FUNC3	Analyzer function 3
FUNC4	Analyzer function 4

Parameter

Item	Type	Range of values	Default value
<data_source>	Discrete	GEN|FUNC1|FUNC2|FUNC3|FUNC4	FUNC1

Remark

The X-axis data source can be set to FUNC1, FUNC2, FUNC3, or FUNC4 if the analyzer function is not set to NONE.

Examples

The following command sets the X-axis data source to generator in the sweep mode.

```
DISP:SWE:TRAC:X:SOUR GEN
```

The following query returns the X-axis data source in the sweep mode.

```
DISP:SWE:TRAC:X:SOUR?
```

Typical response:

```
GEN
```

DISPlay[:WINDow]:SWEep:TRACe:Y[:SCALe]:AUTO

Syntax

```
DISPlay[:WINDow]:SWEep:TRACe:Y[:SCALe]:AUTO
```

Description

Performs an autoscale on the Y-axis to automatically scale the graph display according to the signal each time this command is sent.

Example

The following command performs an autoscale on the Y-axis for the graph view in the sweep mode.

```
DISP:SWE:TRAC:Y:AUTO
```

DISPlay[:WINDow]:SWEep:TRACe:Y[:SCALe]:BOTTom

Syntax

```
DISPlay[:WINDow]:SWEep:TRACe:Y[:SCALe]:BOTTom <minimum_limit>
```

```
DISPlay[:WINDow]:SWEep:TRACe:Y[:SCALe]:BOTTom?
```

Description

Sets the value represented by the minimum (bottom) edge of the Y-axis. The query returns the bottom Y-axis minimum edge.

Parameter

Item	Type	Range of values	Default value
<minimum_limit>	Numeric	–200000 to 200000	1

Examples

The following command sets the bottom Y-axis minimum edge to 50 for the graph view in the sweep mode.

```
DISP:SWE:TRAC:Y:BOTT 50
```

The query returns the bottom Y-axis minimum edge for the graph view in the sweep mode.

```
DISP:SWE:TRAC:Y:BOTT?
```

Typical response:

```
5.000000E+01
```

DISPlay[:WINDow]:SWEep:TRACe:Y[:SCALe]:TOP

Syntax

```
DISPlay[:WINDow]:SWEep:TRACe:Y[:SCALe]:TOP <maximum_limit>
DISPlay[:WINDow]:SWEep:TRACe:Y[:SCALe]:TOP?
```

Description

Sets the value represented by the maximum (top) edge of the Y-axis. The query returns the top Y-axis maximum edge.

Parameter

Item	Type	Range of values	Default value
`<maximum_limit>`	Numeric	–200000 to 200000	30

Examples

The following command sets the top Y-axis maximum edge to 100 for the graph view in the sweep mode.

```
DISP:SWE:TRAC:Y:TOP 100
```

The following query returns the top Y-axis maximum edge for the graph view in the sweep mode.

```
DISP:SWE:TRAC:Y:TOP?
```

Typical response:

```
1.000000E+02
```

DISPlay[:WINDow]:SWEep:TRACe:Y:SPACing

Syntax

```
DISPlay[:WINDow]:SWEep:TRACe:Y:SPACing <spacing_type>
```

```
DISPlay[:WINDow]:SWEep:TRACe:Y:SPACing?
```

Description

Sets the Y-axis spacing as either linear or logarithmic scaling. The query returns the Y-axis spacing type in the form of `LIN` or `LOG`.

Parameter

Item	Type	Range of values	Default value
`<spacing_type>`	Discrete	LINear|LOGarithmic	LINear

Examples

The following command sets the Y-axis spacing to logarithmic for the graph view in the sweep mode.

```
DISP:SWE:TRAC:Y:SPAC LOG
```

The following query returns the Y-axis spacing type for the graph view in the sweep mode.

```
DISP:SWE:TRAC:Y:SPAC?
```

Typical response:

```
LOG
```

DISPlay[:WINDow]:SWEep:TRACe:Y:SOURce

Syntax

```
DISPlay[:WINDow]:SWEep:TRACe:Y:SOURce <data_source>
```

```
DISPlay[:WINDow]:SWEep:TRACe:Y:SOURce?
```

Description

Sets the Y-axis data source to be displayed on the graph in the sweep view. The query returns the Y-axis data source.

GEN	Generator
FUNC1	Analyzer function 1
FUNC2	Analyzer function 2
FUNC3	Analyzer function 3
FUNC4	Analyzer function 4

Parameter

Item	Type	Range of values	Default value
<data_source>	Discrete	GEN|FUNC1|FUNC2|FUNC3|FUNC4	GEN

Remark

The Y-axis data source can be set to FUNC1, FUNC2, FUNC3, or FUNC4 if the analyzer function is not set to NONE.

Examples

The following command sets the Y-axis data source to analyzer function 1 in the sweep mode.

```
DISP:SWE:TRAC:Y:SOUR FUNC1
```

The following query returns the Y-axis data source in the sweep mode.

```
DISP:SWE:TRAC:Y:SOUR?
```

Typical response:

```
FUNC1
```

DISPlay[:WINDow]:VIEW

Syntax

```
DISPlay[:WINDow]:VIEW <view>,<panel>,[<channel>]
```

Description

Sets the front panel LCD display type for the specified panel.

Parameters

Item	Type	Range of values	Default value
<view>	String	Analog Analyzer|Analog Generator|Digital Analyzer|Digital Generator|Graph|Sweep|Sweep Graph|Sweep List|System|Digital Generator DSIDigital Generator AES|Digital Analyzer DSI|Digital Analyzer Statistic|Analog Analyzer Statistic|Digital Analyzer AES|Audio Data Bits| Audio Active Bits|Bit Error	Analog Generator
<panel>	Discrete	PANel1|PANel2|PANel3|PANel4|PANel5|PANel6|PANel7|PANel8|PANel9|PANel10	PANel1
<channel>	Discrete	CH1|CH2|CH3|CH4|CH5|CH6|CH7|CH8	CH1

Remarks

- The <channel> parameter is optional.
- This command is only applicable in the Standard view.
- The <panel> parameter is used to select the focused panel.
- The <channel> parameter is used to change the focused panel to the respective channel.
- When Graph is the selected display type, the U8903B will ignore the <channel> parameter.
- When Sweep, Sweep Graph, Sweep List, or System is the selected display type, the U8903B will ignore the <panel> and <channel> parameters.

Examples

The following command displays the analog analyzer channel 1 at panel 2.

```
DISP:VIEW "Analog Analyzer",PAN2,CH1
```

The following command displays the system page.

```
DISP:VIEW "System",PAN1
```

THIS PAGE HAS BEEN INTENTIONALLY LEFT BLANK.

7 FETCh Subsystem

This chapter describes the `FETCh` subsystem commands.

FETCh:ARRay?

Syntax

```
FETCh:ARRay? (@<channel>)[,<type>]
```

Description

Returns an array of measurement data of the selected channel. The returned data is the result of the last acquisition trigger, and in the IEEE-488.2 binary block format. The data is valid until the next **INITiate[:IMMediate]:GRAPh** command is sent.

Parameters

Item	Type	Range of values	Default value
`<channel>`	Numeric	1|2|3|4|5|6|7|8	Required parameter
`<type>`	Discrete	MAGNitude|PHASe|TIME|PSD	Optional parameter

Remarks

- The channel that you have selected to acquire the array of data is based on the channel(s) specified in the **INITiate[:IMMediate]:GRAPh** command.
- To plot a graph with the array of data, the X-axis points can be calculated using the following equations.
  - If the graph analysis mode is in the time domain, the X-axis point can be computed as follows.

    $$Point\ X = X \times \left(\frac{1}{Measurement\ bandwidth}\right)$$

    Where X = 0, 1, 2, ...
  - If the graph analysis mode is in the frequency domain, the X-axis point can be computed as follows.

    $$Point\ X = X \times \left(\frac{X \times Measurement\ bandwidth}{2 \times [Point\ count - 1]}\right)$$

    Where X = 0, 1, 2, ...

- The measurement data is returned in the unit dBV if the graph analysis mode is in the frequency domain (magnitude). The data is returned in the unit radian if the graph analysis mode is in the frequency domain (phase). The data is returned in the unit V if the graph analysis mode is in the time domain.
- If the `<data>` parameter is not specified, the query returns the data of the current active analysis mode.

Examples

The following command sequence is used to acquire an array of measurement data for channel 1 in the graph analysis mode.

```
TRIG:GRAP:SOUR IMM
INIT:GRAP (@1)
FETC:ARR? (@1),MAGN
```

FETCh[:SCALar]?

Syntax

```
FETCh[:SCALar]? <function>,(@<channel_list>)
```

Description

Retrieves the measurement result for the specified measurement function and channel(s). Multiple responses are separated by commas.

The description for each `<function>` parameter is listed as follows.

`FUNC1`	Measurement result of function 1
`FUNC2`	Measurement result of function 2
`FUNC3`	Measurement result of function 3
`FUNC4`	Measurement result of function 4
`ALL`	Measurement result of function 1 and function 2

Parameters

Item	Type	Range of values	Default value
`<function>`	Discrete	FUNC1|FUNC2|FUNC3|FUNC4|ALL	FUNC1
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- The data returned by the query is the result of the last acquisition trigger. The data is valid until the next I**INITiate[:IMMediate]:ANALyzer** command is sent.
- The `FETCh? ALL` query returns a sequential data format. For example, the `FETCh? ALL,(@1,2)` query returns the result of the measurement function 1 of channel 1, the result of the measurement function 2 of channel 1, the result of the measurement function 3 of channel 1, the result of the measurement function 4 of channel 1, the result of the measurement function 1 of channel 2, and so on.

- If no measurement has been taken or there is an error occurred when the measurement is being taken for a particular channel, `NaN (9.91E+37)` will be returned. If there is voltage overload, `INF (9.9E+37)` will be returned.
- The returned measurement result unit depends on the unit set using the **SENSe[:ANALog]:FUNCtion<j>:UNIT** command.

Examples

The following command sequence is used to measure VDC and VAC at channel 1.

```
SENS:FUNC1 VDC,(@1)
SENS:FUNC2 VAC,(@1)
TRIG:ANAL:SOUR IMM
INIT:ANAL (@1)
STAT:OPER:COND?          // poll this command until 0 is returned
FETC? FUNC1,(@1)
```

Typical response:

```
8.116441E-02
```

FETC? FUNC2,(@1)

Typical response:

```
9.807300E-01
```

FETCh:SWEep?

Syntax

```
FETCh:SWEep? <function>,(@<channel>)
```

Description

Returns the sweep result for the specified measurement function and channel. All active measurement functions will be measured together during `INIT:SWE`. Multiple responses are separated by commas.

FUNC1	Measurement result of function 1
FUNC2	Measurement result of function 2
FUNC3	Measurement result of function 3
FUNC4	Measurement result of function 4

Parameters

Item	Type	Range of values	Default value
<function>	Discrete	FUNC1|FUNC2|FUNC3|FUNC4	FUNC1
<channel>	Numeric	1|2|3|4|5|6|7|8	1

Remarks

– The **INITiate[:IMMediate]:SWEep** command must be used to initiate the sweep prior to sending the **FETCh:SWEep?** query.

– The specified channel must be one of the channels specified in the **SENSe:SWEep:CHANnel** command.

– Refer to "**Example 8: Perform Sweep**" on page 649 for the example on performing sweep.

– The measurement unit in the sweep mode can be configured using the **SENSe[:ANALog]:FUNCtion<j>:UNIT** command.

– The sweep result is returned in the current unit of the respective analyzer function.

Example

The following command sequence is used to obtain the sweep result function 1 for channel 1.

```
SOUR:SWE:CHAN 1
INIT:SWE
FETC:SWE? FUNC1,(@1)
```

Typical response:

```
7.800041E+04,7.800030E+04,7.377602E+04,6.919201E+04,6.850725E+04,
6.282951E+04,6.018090E+04,5.758000E+04,5.519361E+04,…
```

FETCh:DIGital[:SCALar]?

Syntax

```
FETCh:DIGital[:SCALar]? <function>,(@<channel list>)
```

Description

Retrieves the measurement result for the specified measurement function and channel(s). Multiple responses are separated by commas.

The description for each `<function>` parameter is listed as follows.

FUNC1	Measurement result of function 1
FUNC2	Measurement result of function 2
FUNC3	Measurement result of function 3
FUNC4	Measurement result of function 4
ALL	Measurement result of function 1, function 2, function 3, and function 4

Parameters

Item	Type	Range of values	Default value
<function>	Discrete	FUNC1|FUNC2||FUNC3|FUNC4|ALL	FUNC1
<channel_list>	Discrete	One or more analyzer channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- The data returned by the `FETCh:DIGital?` query is the result of the last acquisition trigger. The data is valid until the next `INITiate[:IMMediate]:DIGital:ANALyzer` command is sent.
- The `FETCh:DIGital? ALL` query returns a sequential data format. For example, the `FETCh:DIGital? ALL, (@1,2)` query returns the result of the first measurement function of channel 1, the result of the second measurement function of channel 1, the result of the third measurement function of channel 1, the result of the fourth measurement function of channel 1, the result of the first measurement function of channel 2, and so on.

- If no measurement has been taken or there is an error occurred when the measurement is being taken for a particular channel, `NAN (9.91E+37)` will be returned. If there is voltage overload, `INF (9.9E+37)` will be returned.

NOTE

When FETCh is queried, the measurement result will be returned in the unit as listed in "**Appendix B: Units of the Measurement Function Returned Values**" on page 661.

Examples

The following command sequence is used to measure VDC and VAC at digital channel 1.

```
SENS:DIG:FUNC1 VDC, (@D1)
SENS:DIG:FUNC2 VAC, (@D1)
INIT:DIG:ANAL (@D1)
FETC:DIG? FUNC1, (@D1)
```

Typical response:

```
8.116441E-02
```

```
FETC:DIG? FUNC2, (@D1)
```

Typical response:

```
9.807300E-01
```

FETCh:DIGital:AUDIO:BITS?

Syntax

```
FETCh:DIGital:AUDIO:BITS? <bit_type>,(@<channel_list>)
```

Description

Queries the data bits or active bits of the embedded data in the digital signal audio word for each subframe. The two subframes correspond to channel 1 and 2. The query returns the audio bits.

Parameters

Item	Type	Range of values	Default value
<bit_type>	Discrete	DATA|ACTive	DATA
<channel_list>	Discrete	(@D1)|(@D2)	Required parameter

Examples

The following query returns the data bits for channel 1.

```
FETC:DIG:AUD:BITS? DATA, (@D1)
```

Typical Response:

```
101000000000000000000000
```

FETCh:DIGital:ERRor:FLAG?

Syntax

```
FETCh:DIGital:ERRor:FLAG?
<error_flag>[,<error_flag>[,<error_flag>...]]
```

Description

Queries the error flag(s) of the embedded audio data. The returned data is in Boolean. Multiple error flags and responses are separated by commas.

`CONFidence`	Confidence bit. Updated on sub-frame boundaries. 0 - No error 1 - Confidence error. The input data stream may be near error condition due to jitter degradation.
`CODing`	Bi-phase coding error bit. Updated on sub-frame boundaries. 0 - No error 1 - Bi-phase error. This indicates an error in the received bi-phase coding.
`LOCK`	Receiver lock status when sourced by an incoming AES3-compatible data. Updated on CS block. boundaries. 0 - Receiver locked 1 - Receiver out of lock
`PARity`	Parity bit. Updated on sub- frame boundaries. 0 - No error 1 - Parity error
`VALidity`	Received AES3 Validity bit status. Updated on sub-frame boundaries. 0 - Data is valid and is normally linear coded PCM audio 1 - Data is invalid, or may be valid compressed audio
`CCRC`	Channel Status Block Cyclic Redundancy Check bit. Updated on CS block boundaries, valid only in Pro mode. 0 - No error 1 - Error
`QCRC`	Q-subcode data CRC error indicator. Updated on Q-subcode block boundaries. 0 - No error 1 - Error

Parameter

Item	Type	Range of values	Default value
`<error_flag>`	Discrete	CONFidence|CODing|LOCK|VALidity| PARity|CCRC|QCRC	VALidity

Examples

The following query returns the Confidence and Parity bit error status.

```
FETC:DIG:ERR:FLAG? CONF, PAR
```

Typical response:

```
0,1
```

FETCh:DIGital:DELay?

Syntax

```
FETCh:DIGital:DELay?
```

Description

Queries the time delay between the digital input and output in seconds.

Examples

The following query returns the time delay.

```
FETC:DIG:DEL?
```

FETCh:DIGital:BERT?

Syntax

```
FETCh:DIGital:BERT? <type>, (@<channel_list>)
```

Description

Queries the total bits, total errors, or bit error rate in the Bit Error Rate Test (BERT). The returned data is in string.

TBITs	Total bits
ERRors	Total errors
BER	Bit error rate
IERRor	Instantaneous Errors

Parameters

Item	Type	Range of values	Default value
<type>	Discrete	TBITs|ERRors|BER|IERRor	BER
<channel_list>	Discrete	One or more analyzer channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Examples

The following query returns the total errors in BERT.

```
FETC:DIG:BERT? ERR, (@D1)
```

Typical response:

```
“3.000000E+00”
```

FETCh:DIGital:ARRay?

Syntax

```
FETCh:DIGital:ARRay? (@<channel>) [,<type>]
```

Description

Returns an array of measurement data of the selected channel(s). The returned data is the result of the last acquisition trigger, and in the IEEE- 488.2 binary block format. The data is valid until the next **INITiate[:IMMediate]:DIGital:GRAPh** command is sent.

Parameter

Item	Type	Range of values	Default value
`<channel>`	Numeric	D1|D2	Required parameter
`<type>`	Discrete	MAGNitude|PHASe|TIME	Optional parameter

Remark

- The channel(s) that you have selected to acquire the array of data is based on the channel(s) specified in the **INITiate[:IMMediate]:DIGital:GRAPh** command.
- To plot a graph with the array of data, the X-axis points can be calculated using the following equations.
  - If the graph analysis mode is in the time domain, the X-axis point can be computed as follows.

    $$Point\ X = X \times \left(\frac{1}{Measurement\ bandwidth}\right)$$

    Where X = 0, 1, 2, ...
  - If the graph analysis mode is in the frequency domain, the X-axis point can be computed as follows.

    $$Point\ X = X \times \left(\frac{X \times Measurement\ bandwidth}{2 \times [Point\ count - 1]}\right)$$

    Where X = 0, 1, 2, ...

– The measurement data is returned in the unit dBV if the graph analysis mode is in the frequency domain (magnitude). The data is returned in the unit radian if the graph analysis mode is in the frequency domain (phase). The data is returned in the unit V if the graph analysis mode is in the time domain.

Examples

The following command sequence is used to acquire an array of measurement data for channel 1 in the graph mode.

```
TRIG:DIG:GRAP:SOUR IMM

INIT:DIG:GRAP (@D1)

FETC:DIG:ARR? (@D1)
```

8 INITiate Subsystem

This chapter describes the `INITiate` subsystem commands.

INITiate:CONTinue:ANALyzer

Syntax

```
INITiate:CONTinue:ANALyzer <state>,(@<channel_list>)
INITiate:CONTinue:ANALyzer? (@<channel_list>)
```

Description

Enables or disables the analog analyzer to make continuous measurement. The query returns the analog analyzer continuous measurement state as `0` if the state is `OFF`, or `1` if the state is `ON`.

Parameters

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	OFF
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- After you have enabled the continuous measurement, you can continuously obtain the measurement result by sending the **FETCh[:SCALar]?** command.
- If **INITiate[:IMMediate]:ANALyzer** command is sent, the continuous measurement will be stopped and changed to single measurement.
- This command cannot be used when the GUI is in the HP8903B or Test Sequence mode.
- When the state is ON and the current page is not the standard analysis page, this command switches the page to the standard analysis page. As a result, the graph's continuous states that are set using the **INITiate:CONTinue:GRAPh** command will be set to OFF.

Examples

The following command enables the analog analyzer channel 2 to make continuous measurement.

```
INIT:CONT:ANAL ON,(@2)
```

The following query returns the analog analyzer channel 2 continuous measurement state.

```
INIT:CONT:ANAL? (@2)
```

Typical response:

```
1
```

INITiate:CONTinue:GRAPh

Syntax

```
INITiate:CONTinue:GRAPh <state>,(@<channel_list>)[,<measurement>]
INITiate:CONTinue:GRAPh? (@<channel_list>)[,<measurement>]
```

Description

Enables or disables the graph analysis mode to make continuous measurement. If `DEFault` or no parameter is sent, only the graph trigger system will be initiated. If `MEASurement` is sent, the two active displayed measurement in the graph analysis page will be triggered and the measurement will be displayed on the screen if the graph analysis page is active. You should use the corresponding command to fetch the measurement. The query returns the graph analysis mode continuous measurement state as `0` if the state is `OFF`, or `1` if the state is `ON`.

Parameters

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	OFF
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter
`<measurement>`	Discrete	DEFault|MEASurement	Optional Parameter

Remarks

- You can obtain the measurement result in continuous measurement with the **FETCh:ARRay?** command.
- This command cannot be used when the GUI is in the HP8903B or Test Sequence mode.
- When the state is `ON` and the current page is not the graph analysis page, this command switches the page to the graph analysis page. As a result, the analyzer's continuous states that are set using the **INITiate:CONTinue:ANALyzer** command will be set to `OFF`.

- The continuous measurement will be stopped if you perform any of the following commands.
  - Measurement in analog analyzer or sweep mode (**INITiate[:IMMediate]:ANALyzer**, **INITiate[:IMMediate]:SWEep**, or **INITiate:CONTinue:ANALyzer**)
  - Self-test (***TST?**)
  - Reset (***RST**)

Examples

The following command sequence enables the graph analysis mode channel 2 in the frequency domain to make continuous measurement.

```
DISP:ANAL:MODE MAGN
INIT:CONT:GRAP ON,(@2)
FETC:ARR? (@2)
```

The following query returns the graph analysis mode channel 2 continuous measurement state.

```
INIT:CONT:GRAP? (@2)
```

Typical response:

```
1
```

INITiate:CONTinue:DIGital:ANALyzer

Syntax

```
INITiate:CONTinue:DIGital:ANALyzer <state>,(@<channel list>)
```

Description

Enables or disables the digital analyzer to make continuous measurement.

Parameter

Item	Type	Range of values	Default value
`<state>`	Bool	OFF(0)|ON(1)	OFF
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

– When you enable the continuous measurement, you can continuously obtain the measurement result by sending the **FETCh:DIGital[:SCALar]?** command.

– If the **INITiate[:IMMediate]:DIGital:ANALyzer** command is sent, the continuous measurement will be stopped and changed to a single measurement.

– This command cannot be used when the GUI is in the HP8903B or Test Sequence mode.

Example

```
The following command enables the digital analyzer channel 1 to
make continuous measurement.

INIT:CONT:DIG:ANAL ON, (@D1)
```

INITiate:CONTinue:DIGital:AUDio:BITS

Syntax

```
INITiate:CONTinue:DIGital:AUDio:BITS <state>,(@<channel list>)
```

Description

Enables or disables the digital analyzer to make continuous audio bits measurement.

Parameter

Item	Type	Range of values	Default value
`<state>`	Bool	OFF(0)|ON(1)	OFF
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- When you enable the continuous measurement for audio bits, you can continuously obtain the measurement result by sending the **FETCh:DIGital:AUDIO:BITS?** command.
- If the **INITiate:CONTinue:DIGital:AUDio:BITS** command is sent, the continuous bits measurement will be stopped and changed to a single measurement.
- This command cannot be used when the GUI is in the HP8903B or Test Sequence mode.

Example

The following command enables the digital analyzer channel 1 to make continuous audio bits measurement.

```
INIT:CONT:DIG:AUD:BITS ON, (@D1)
```

INITiate:CONTinue:DIGital:DELay

Syntax

```
INITiate:CONTinue:DIGital:DELay <state>
```

Description

Starts or Stops the digital analyzer to make time delay measurement. Time delay measurement is done by measuring the delay time from the digital input with reference to the digital output. This measurement measures the time delay introduced by the Device Under Test (DUT).

Parameter

Item	Type	Range of values	Default value
`<state>`	Bool	OFF(0)|ON(1)	OFF

Remark

- This command is only applicable when the analyzer trigger source is set to immediate.
- You can obtain the measurement result in continuous measurement with the **FETCh:DIGital:DELay?** command.
- In order to perform this test, the digital generator must be set to output the signal.
- The delay measurement is only applicable for AES3/SPDIF audio signal.

Examples

The following command starts a time delay measurement.

```
INIT:CONT:DIG:DEL ON
```

The time delay measurement result can be obtained continuously with the following command.

```
FETC:DIG:DEL?
```

INITiate:CONTinue:DIGital:GRAPh

Syntax

```
INITiate:CONTinue:DIGital:GRAPh <state>, (@<channel_list>)
[,<measurement>]
```

```
INITiate:CONTinue:DIGital:GRAPh?
(@<channel_list>)[,<measurement>]
```

Description

Enables or disables the graph analysis mode to make continuous measurement. If `DEFault` or no parameter is sent, only the graph trigger system will be initiated. If `MEASurement` is sent, the two active displayed measurement in the graph analysis page will be triggered and the measurement will be displayed on the screen if the graph analysis page is active. You should use the corresponding command to fetch the measurement. The query returns the graph analysis mode continuous measurement state as `0` if the state is `OFF`, or `1` if the state is `ON`.

Parameters

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	OFF
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter
`<measurement>`	Discrete	DEFault|MEASurement	DEFault

Remarks

– You can obtain the measurement result in continuous measurement with the **FETCh:DIGital:ARRay?** command.

– This command cannot be used when the GUI is in the HP8903B or Test Sequence mode.

– When the state is `ON` and the current page is not the graph analysis page, this command switches the page to the graph analysis page. As a result, the analyzer's continuous states that are set using the **INITiate:CONTinue:DIGital:ANALyzer** command will be set to `OFF`.

- The continuous measurement will be stopped if you perform any of the following commands.
  - Measurement in analog analyzer or sweep mode (**INITiate[:IMMediate]:DIGital:ANALyzer**, **INITiate[:IMMediate]:SWEep**, or **INITiate:CONTinue:DIGital:ANALyzer**)
  - Self-test (***TST?**)
  - Reset (***RST**)

Examples

The following command sequence enables the graph analysis mode channel 2 in the frequency domain to make continuous measurement.

```
DISP:ANAL:MODE MAGN
INIT:CONT:GRAP ON,(@2)
FETC:ARR? (@2)
```

The following query returns the graph analysis mode channel 2 continuous measurement state.

```
INIT:CONT:GRAP? (@2)
```

Typical response:

```
1
```

INITiate[:IMMediate]:ANALyzer

Syntax

```
INITiate[:IMMediate]:ANALyzer (@<channel_list>)
```

Description

Initiates the analog analyzer measurement trigger system for the specified channel(s). When a measurement trigger system is initiated, an event on a selected trigger source causes the specified triggering action to occur. If the trigger system is not initiated, all triggers are ignored.

Parameter

Item	Type	Range of values	Default value
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– If the analog analyzer trigger source is set to IMMediate, sending this command will cause the U8903B to take the measurement directly. If the trigger source is set to `EXTernal`, sending this command will cause the U8903B to start taking the measurement when the external signal is received. If the trigger source is set to `BUS`, sending this command will put the U8903B in the ‘waiting for trigger’ state until the ***TRG** command is sent. The U8903B will only start to take the measurement when the ***TRG** command is received.

– You can verify whether a measurement has completed by polling the status register value via the **STATus:OPERation:CONDition?** query. While a measurement is in progress, bit 4 of the condition register of the Standard Operation register group will be set. After the measurement has completed, bit 4 will be cleared to 0.

– Sending this command will stop the continuous measurement if it is running previously.

Example

The following command initiates the measurement trigger system on channel 1.

```
INIT:ANAL (@1)
```

INITiate[:IMMediate]:GRAPh

Syntax

```
INITiate[:IMMediate]:GRAPh (@<channel_list>)[,<measurement>]
```

Description

Initiates the graph trigger system for an array of data for the specified channel(s). When a graph trigger system is initiated, an event on a selected trigger source causes the specified triggering action to occur. If the trigger system is not initiated, all triggers are ignored.

Parameters

Item	Type	Range of values	Default value
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter
<measurement>	Discrete	DEFault|MEASurement	DEFault

Remarks

– It takes a few milliseconds for the U8903B to be ready to acquire a trigger signal after receiving this command.

– You can verify whether a measurement has completed by polling the status register value via the **STATus:OPERation:CONDition?** query. While a measurement is in progress, bit 4 of the condition register of the Standard Operation register group will be set. After the measurement has completed, bit 4 will be cleared to 0.

Example

The following command initiates the graph trigger system on channel 1 to acquire an array of graph points.

```
INIT:GRAP (@1)
```

INITiate[:IMMediate]:SWEep

Syntax

```
INITiate[:IMMediate]:SWEep
```

Description

Initiates the sweep for the channel specified in the **SENSe:SWEep:CHANnel** command.

NOTE Do not perform other operations while sweep is in progress as doing so might cause unexpected results.

Remark

The sweep bit at the condition register of the Standard Operation register group will be set if the sweep mode is Auto Sweep or Auto List to indicate the automatic sweep is in progress, when sweep is initiated using this command. The sweep bit is cleared when the automatic sweep has completed.

Example

The following command initiates the sweep.

```
INIT:SWE
```

INITiate[:IMMediate]:WAVFile:RECord

Syntax

```
INITiate[:IMMediate]:WAVFile:RECord
<filename>,(@<active_channel>)
```

Description

Initiates the recording of the input signal for the specified channel to a wave file.

Parameters

Item	Type	Range of values	Default value
`<filename>`	String	The filename of the wave file. For example, "\Storage 1\ file.wav".	Required parameter
`<active_channel>`	Numeric	1|2|3|4|5|6|7|8	Required parameter

Remarks

– The recording bit (bit 1) at the condition register of the Status Operation register will be set when recording is initiated with this command. After the recording is completed, the recording bit is cleared.

– When the duration is reached, the recording will automatically stop.

– This command can be stopped by sending the **ABORt:WAVFile:RECord** command.

– It is recommended to store the wave file in an external USB flash storage as there is limited storage space in the internal storage.

– Refer to "**Example 11: Record Input Signal to Wave File**" on page 656 for the example on recording to a wave file.

Example

The following command initiates the recording of the channel 1 input signal to a wave file named 'file.wav' in the '\Storage 1' directory.

```
INIT:WAVF:REC "\Storage 1\file.wav",(@1)
```

INITiate[:IMMediate]:DIGital:ANALyzer

Syntax

```
INITiate[:IMMediate]:DIGital:ANALyzer(@<channel_list>)
```

Description

Initiates the digital analyzer measurement trigger system for the specified channel(s). When a measurement trigger system is initiated, an event on a selected trigger source causes the specified triggering action to occur. If the trigger system is not initiated, all triggers are ignored.

Parameter

Item	Type	Range of values	Default value
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- If the digital analyzer trigger source is set to Immediate, sending this command will cause the U8903B to take the measurement directly. If the trigger source is set to External, sending this command will cause the U8903B to start taking the measurement when the external signal is received. If the trigger source is set to Bus, sending this command will put the U8903B in the 'waiting for trigger' state until the `*TRG` command is sent. The U8903B will only start to take the measurement when the `*TRG` command is received.
- You can verify whether a measurement has completed by polling the status register value via the **STATus:OPERation:CONDition?** query. While a measurement is in progress, bit 2 of the condition register of the Standard Operation register group will be set. After the measurement has completed, bit 2 will be cleared to 0.

Examples

The following command initiates the measurement trigger system on channel 1.

```
INIT:DIG:ANAL (@D1)
```

INITiate[:IMMediate]:DIGital:AUDio:BITS

Syntax

```
INITiate[:IMMediate]:DIGital:AUDio:BITS(@<channel_list>)
```

Description

Initiates the digital analyzer audio bits measurement for the specified channel(s).

Parameter

Item	Type	Range of values	Default value
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- This command is not applicable for the bus trigger source.
- You can send the **FETCh:DIGital:AUDIO:BITS?** command to obtain the measured bits value and send the **FETCh:DIGital:ERRor:FLAG?** command to obtain the error flags result of the bits measurement.

Examples

The following command initiates the audio bits measurement on channel 1.

INIT:DIG:AUD:BITS (@D1)

INITiate[:IMMediate]:DIGital:GRAPh

Syntax

```
INITiate[:IMMediate]:DIGital:GRAPh(@<channel_list>)
[,<measurement>]
```

Description

Initiates the graph trigger system for an array of data for the specified channel(s). When a graph trigger system is initiated, an event on a selected trigger source causes the specified triggering action to occur. If the trigger system is not initiated, all triggers are ignored.

Parameter

Item	Type	Range of values	Default value
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter
`<measurement>`	Discrete	DEFault|MEASurement	Optional Parameter

Remarks

– It takes a few milliseconds for the U8903B to be ready to acquire a trigger signal after receiving this command.

– You can verify whether a measurement has completed by polling the status register value via the **STATus:OPERation:CONDition?** query. While a measurement is in progress, bit 4 of the condition register of the Standard Operation register group will be set. After the measurement has completed, bit 4 will be cleared to 0.

Examples

The following command initiates the graph trigger system on channel 1 to acquire an array of graph points.

```
INIT:DIG:GRAP (@D1)
```

9 INPut Subsystem

This chapter describes the `INPut` subsystem commands.

INPut[:ANALog]:COUPling

Syntax

```
INPut[:ANALog]:COUPling <coupling>,(@<channel_list>)
INPut[:ANALog]:COUPling? (@<channel_list>)
```

Description

Sets the analog analyzer AC or DC coupling for the specified channel(s). The DC coupling allows both AC and DC input signals to pass through to the measurement circuitry. The AC coupling blocks the DC component of the input signal. The query returns the input coupling type of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<coupling>	Discrete	AC|DC	AC
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Examples

The following command sequence sets the input coupling types for channel 1 and channel 2 to AC and DC respectively.

```
INP:COUP AC,(@1)
INP:COUP DC,(@2)
```

The following query returns the input coupling types of channel 1 and channel 2.

```
INP:COUP? (@1,2)
```

Typical response:

```
AC,DC
```

INPut[:ANALog]:BANDwidth

Syntax

```
INPut[:ANALog]:BANDwidth <bandwidth>,(@<channel_list>)
INPut[:ANALog]:BANDwidth? (@<channel_list>)
```

Description

Sets the analog analyzer measurement bandwidth. The query returns the measurement bandwidth.

BW90	Bandwidth 90 kHz
BW1500	Bandwidth 1.5 MHz

Parameters

Item	Type	Range of values	Default value
<bandwidth>	Discrete	BW90|BW1500	BW90
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- By default, all the analog analyzer cards bandwidth is 90 kHz. The 1.5 MHz selection will only be available if the wide bandwidth option is purchased. The wide bandwidth option is only available for analog analyzer card 1 (channel 1 and channel 2).
- The bandwidth setting is applied to both channels in the same analog analyzer card. For example, the `INP:BAND BW90,(@1)` command is sent. The 90 kHz bandwidth is applied to both channel 1 and channel 2 in the analog analyzer even though the command refers to only channel 1.
- For compatibility with firmware 2.10.x.x and below, the HIGH and LOW selections for the `<bandwidth>` parameter are still applicable. the bandwidth will be set to 90 kHz.
- For compatibility with firmware 2.10.x.x and below, the `<channel_list>` parameter can be omitted and the U8903B will assume the `<channel_list>` parameter as `(@1,2)`. For the query, the U8903B will assume the `<channel_list>` parameter is channel 1.

Examples

The following command sets the analog analyzer measurement bandwidth to 90 kHz for the analog analyzer card 1.

```
INP:BAND BW90,(@1,2)
```

The following query returns the analog analyzer measurement bandwidth for the analog analyzer card 1.

```
INP:BAND? (@1)
```

Typical response:

```
BW90
```

INPut[:ANALog]:EXTernal:GAIN

Syntax

```
INPut[:ANALog]:EXTernal:GAIN
<external_gain><unit>,(@<channel_list>)
```

```
INPut[:ANALog]:EXTernal:GAIN? (@<channel_list>)
```

Description

Sets the external gain of the input signal. External gain is used to correct the effects of any external gain or loss that may be part of the measurement setup. For example, if an amplifier is part of the measurement setup, this external gain value is used to remove the effects of its gain. However, if an external attenuator is used in a measurement of a high voltage, this external gain value is used to correct the loss. The instrument will then report the levels being measured and not simply the levels being presented to the instrument. The external gain factor can be entered in dB or as a multiplication factor x. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<external_gain>`	Numeric	(–60 dB to 60 dB)|(0.00x to 1000x)	0 dB
`<unit>`	Discrete	dB|x	dB
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- The query will return the external gain factor in the relative unit that you set. For example, if you set the external gain to –6 dB, it will return the value as –6. If you set the external gain to 0.5x, it will return the value as 0.5.
- However, you can also change the external gain factor to be retuned in your desired unit by sending the **INPut[:ANALog]:EXTernal:GAIN:UNIT** command. The last unit set in the specified channel will always take effect and cause this query to return the value in the particular unit.

Examples

The following command sets the external gain for channel 1 to –6 dB.

```
INP:EXT:GAIN -6dB,(@1)
```

The following query returns the external gain for channel 1.

```
INP:EXT:GAIN? (@1)
```

Typical response:

```
-6
```

INPut[:ANALog]:EXTernal:GAIN:UNIT

Syntax

```
INPut[:ANALog]:EXTernal:GAIN:UNIT <unit>,(@<channel_list>)
INPut[:ANALog]:EXTernal:GAIN:UNIT? (@<channel_list>)
```

Description

Sets the unit of the external gain for the specified channel(s). The query returns the unit of the external gain for the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<unit>`	Discrete	dB|x	dB
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Examples

The following command sets the unit of the external gain for channel 1 to x.

```
INP:EXT:GAIN:UNIT x,(@1)
```

The following query returns the unit of the external gain for channel 1.

```
INP:EXT:GAIN:UNIT? (@1)
```

Typical response:

```
x
```

INPut[:ANALog]:IMPedance:BALanced

Syntax

```
INPut[:ANALog]:IMPedance:BALanced
<impedance_balanced>,(@<channe_list>)
```

```
INPut[:ANALog]:IMPedance:BALanced? (@<channel_list>)
```

Description

Sets the impedance of the analog analyzer balanced input. The query returns the impedance value of the balanced input. Multiple responses are separated by commas.

R200000	Impedance of 200 kΩ
R600	Impedance of 600 Ω
R300	Impedance of 300 Ω

Parameters

Item	Type	Range of values	Default value
<impedance_balanced>	Discrete	R200000|R600|R300	R200000
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Examples

The following command sets the impedance of the analog analyzer balanced input to 600 Ω for channel 1.

```
INP:IMP:BAL R600,(@1)
```

The following query returns the impedance value of the analog analyzer balanced input for channel 1.

```
INP:IMP:BAL? (@1)
```

Typical response:

```
R600
```

INPut[:ANALog]:IMPedance:UNBalanced

Syntax

```
INPut[:ANALog]:IMPedance:UNBalanced
<impedance_unbalanced>,(@<channel_list>)
```

```
INPut[:ANALog]:IMPedance:UNBalanced? (@<channel_list>)
```

Description

Sets the impedance of the analog analyzer unbalanced input. The query returns the impedance value of the unbalanced input. Multiple responses are separated by commas.

R100000	Impedance of 100 kΩ
R600	Impedance of 600 Ω
R300	Impedance of 300 Ω

Parameter

Item	Type	Range of values	Default value
`<impedance_unbalanced>`	Discrete	R100000|R600|R300	R100000
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Examples

The following command sets the impedance of the analog analyzer unbalanced input to 600 **Ω** for channel 1.

```
INP:IMP:UNB R600,(@1)
```

The following query returns the impedance value of the analog analyzer unbalanced input for channel 1.

```
INP:IMP:UNB? (@1)
```

Typical response:

```
R600
```

INPut[:ANALog]:TYPE

Syntax

```
INPut[:ANALog]:TYPE <type>, (@<channel_list>)
INPut[:ANALog]:TYPE? (@<channel_list>)
```

Description

Sets the input connection for the specified channel(s). The query returns the input connection type of the selected channel(s). Multiple responses are separated by commas.

`BALanced`	Balanced or differential input
`UNBalanced`	Unbalanced input with respect to the ground
`LOOPback`	Internal loopback

Parameters

Item	Type	Range of values	Default value
`<type>`	Discrete	BALanced|UNBalanced|LOOPback	UNBalanced
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Examples

The following command sequence sets the analog analyzer input connections for channel 1 and channel 2 to `UNBalanced` and `BALanced` respectively.

```
INP:TYPE UNB,(@1)
INP:TYPE BAL,(@2)
```

The following query returns the input connection type of channel 1.

```
INP:TYPE? (@1)
```

Typical response:

```
UNB
```

INPut:DIGital:TYPE

Syntax

```
INPut:DIGital:TYPE <type>
```

```
INPut:DIGital:TYPE?
```

Description

Sets the digital analyzer input connection. The query returns the digital analyzer input connection type.

BALanced	Balanced or differential input
UNBalanced	Unbalanced input with respect to the ground
OPTical	Optical connector (TOSLINK)
DSI	Digital Serial Interface

Parameter

Item	Type	Range of values	Default value
<type>	Discrete	BALanced|UNBalanced|OPTical|DSI	UNBalanced

Examples

The following command sets the digital analyzer input connection to Optical.

```
INP:DIG:TYPE OPT
```

The following query returns the digital analyzer input connection type.

```
INP:DIG:TYPE?
```

Typical response:

```
OPT
```

INPut:DIGital:SRATe?

Syntax

```
INPut:DIGital:SRATe?
```

Description

Queries the sampling rate of the digital analyzer input signals.

Examples

The following query returns the sampling rate of the digital analyzer input signal.

```
INP:DIG:SRAT?
```

Typical response:

```
4.800000E+01
```

INPut:DIGital:IMPedance:BALanced

Syntax

```
INPut:DIGital:IMPedance:BALanced <impedance>
```

```
INPut:DIGital:IMPedance:BALanced?
```

Description

Sets the impedance of the digital analyzer balanced input connection. The query returns the impedance value.

HIZ	High impedance
R110	Low impedance of 110 Ω for balanced input type

Parameter

Item	Type	Range of values	Default value
<impedance>	Discrete	HIZ|R110	HIZ

Examples

The following command sets the impedance of the digital analyzer balanced input connection to 110 Ω.

```
INP:DIG:IMP:BAL R110
```

The following query returns the impedance value.

```
INP:DIG:IMP:BAL?
```

Typical response:

```
R110
```

INPut:DIGital:IMPedance:UNBalanced

Syntax

```
INPut:DIGital:IMPedance:UNBalanced <impedance>
INPut:DIGital:IMPedance:UNBalanced?
```

Description

Sets the impedance of the digital analyzer unbalanced input connection. The query returns the impedance value.

HIZ	High impedance
R75	Low impedance of 75 Ω for unbalanced input type

Parameter

Item	Type	Range of values	Default value
<impedance>	Discrete	HIZ|R75	HIZ

Examples

The following command sets the impedance of the digital analyzer unbalanced input connection to 75 **Ω**.

```
INP:DIG:IMP:UNB R75
```

The following query returns the impedance value.

```
INP:DIG:IMP:UNB?
```

Typical response:

```
R75
```

INPut:DIGital:AUDio[:DECoding]:FORMat

Syntax

```
INPut:DIGital:AUDio[:DECoding]:FORMat <format>
INPut:DIGital:AUDio[:DECoding]:FORMat?
```

Description

Sets the audio decoding format of the embedded audio signals. The query returns the audio decoding format.

LPCM	Linear Pulse Code Modulation
ULAW	μ-Law decoding format
ALAW	A-Law decoding format

Parameter

Item	Type	Range of values	Default value
<format>	Discrete	LPCM|ULAW|ALAW	LPCM

Examples

The following command sets the decoding format to A-Law.

```
INP:DIG:AUD:FORM ALAW
```

The following query returns the decoding format.

```
INP:DIG:AUD:FORM?
```

Typical response:

```
ALAW
```

INPut:DIGital:AUDio:RESolution

Syntax

```
INPut:DIGital:AUDio:RESolution <resolution>
INPut:DIGital:AUDio:RESolution?
```

Description

Sets the audio resolution or bit depth for the audio data to be analyzed. The query returns the audio resolution.

Parameter

Item	Type	Range of values	Default value
<resolution>	Numeric	8 to 24	24

Remark

- If the audio resolution value matches or exceeds the resolution of the incoming digital signal, the signal is passed on.
- If the audio resolution value is lower than the resolution of the incoming digital signal, the signal is truncated at the least significant bit (LSB).

Examples

The following command sets the audio resolution to 20 bits.

```
INP:DIG:AUD:RES 20
```

The following query returns the audio resolution.

```
INP:DIG:AUD:RES?
```

Typical response:

```
20
```

INPut:DIGital:DSI:AUDio:WLENgth

Syntax

```
INPut:DIGital:DSI:AUDio:WLENgth <length>
```

```
INPut:DIGital:DSI:AUDio:WLENgth?
```

Description

Sets the word length for the DSI interface. The query returns the word length.

Parameter

Item	Type	Range of values	Default value
`<length>`	Numeric	8 to 32	32

Examples

The following command sets the word length to 20 bits.

```
INP:DIG:DSI:AUD:WLEN 20
```

The following query returns the word length.

```
INP:DIG:DSI:AUD:WLEN?
```

Typical response:

```
20
```

INPut:DIGital:DSI:BCLK:SYNC

Syntax

```
INPut:DIGital:DSI:BCLK:SYNC <polarity>
```

```
INPut:DIGital:DSI:BCLK:SYNC?
```

Description

Sets the leading edge of the data to be synchronized to the rising edge or falling edge of the bit clock. The query returns the bit clock sync polarity type.

Parameter

Item	Type	Range of values	Default value
`<polarity>`	Discrete	RISing or FALLing	FALLing

Examples

The following command sets the polarity to the falling edge.

```
INP:DIG:DSI:BCLK:SYNC FALL
```

The following query returns the bit clock sync polarity type.

```
INP:DIG:DSI:BCLK:SYNC?
```

Typical response:

```
FALL
```

INPut:DIGital:DSI:DATA:FORMat

Syntax

```
INPut:DIGital:DSI:DATA:FORMat <format>
```

```
INPut:DIGital:DSI:DATA:FORMat?
```

Description

Sets the format for the DSI interface audio data to be analyzed. The query returns the format for the DSI interface audio data.

LEFT	Left justified. The active data bits are filled to the left edge of the data word.
RIGHt	Right justified. The active data bits are filled to the right edge of the data word.
IIS	IIS format. The word clock is low for the first channel (left channel) and high for the second channel (right channel)
DSP	The period of the word clock is only 1 bit of the bit clock and the data is 1 bit clock delay from the beginning of the word clock.

Parameter

Item	Type	Range of values	Default value
<format>	Discrete	LEFT|RIGHt|IIS|DSP	LEFT

Examples

The following command sets the DSI data format to IIS.

```
INP:DIG:DSI:DATA:FORM IIS
```

The following query returns the DSI data format.

```
INP:DIG:DSI:DATA:FORM?
```

Typical response:

```
IIS
```

INPut:DIGital:DSI:DATA:SHIFt:COUNt

Syntax

```
INPut:DIGital:DSI:DATA:SHIFt:COUNt <number>
INPut:DIGital:DSI:DATA:SHIFt:COUNt?
```

Description

Sets the number for the data bits in relative to the frame clock to be shifted. The query returns the number for the data bits.

Parameter

Item	Type	Range of values	Default value
`<number>`	Numeric	0 to 8	0

Remark

Use the **INPut:DIGital:DSI:DATA:SHIFt:DIRection** command to shift the data bits right or left in relative to the frame clock.

Examples

The following command shifts the data to the right of the frame clock by 2 bits.

```
INP:DIG:DSI:DATA:SHIF:DIR RIGH
INP:DIG:DSI:DATA:SHIF:COUN 2
```

The following query returns the shifted number for the data bits.

```
INP:DIG:DSI:DATA:SHIF:COUN?
```

Typical response:

2

INPut:DIGital:DSI:DATA:SHIFt:DIRection

Syntax

```
INPut:DIGital:DSI:DATA:SHIFt:DIRection <direction>
```

```
INPut:DIGital:DSI:DATA:SHIFt:DIRection?
```

Description

Sets the direction for the data bits in relative to the frame clock to be shifted. The query returns the direction for the data bits.

Parameter

Item	Type	Range of values	Default value
`<direction>`	Discrete	LEFT|RIGHt	RIGHt

Remark

Use the **INPut:DIGital:DSI:DATA:SHIFt:DIRection** command to shift the data bits right or left in relative to the frame clock.

Examples

The following command shifts the data to the left of the frame clock by 1 bit.

```
INP:DIG:DSI:DATA:SHIF:DIR LEFT
```

```
INP:DIG:DSI:DATA:SHIF:COUN 1
```

The following query returns the shifted direction for the data bits.

```
INP:DIG:DSI:DATA:SHIF:DIR?
```

Typical response:

```
LEFT
```

INPut:DIGital:DSI:FSYNc:POLarity

Syntax

```
INPut:DIGital:DSI:FSYNc:POLarity <polarity>
INPut:DIGital:DSI:FSYNc:POLarity?
```

Description

Sets the polarity for the frame clock. The query returns the polarity for the frame clock.

RISing	Indicates the frame clock is high on the left channel of the data.
FALLing	Indicates the frame clock is low on the left channel of the data.

Parameter

Item	Type	Range of values	Default value
<polarity>	Numeric	RISing|FALLing	RISing

Examples

The following command sets the polarity to falling.

```
INP:DIG:DSI:FSYN:POL FALLING
```

The following query returns the polarity.

```
INP:DIG:DSI:FSYN:POL?
```

Typical response:

```
FALL
```

INPut:DIGital:DSI:FSYNc:WIDTh

Syntax

```
INPut:DIGital:DSI:FSYNc:WIDTh <width>
INPut:DIGital:DSI:FSYNc:WIDTh?
```

Description

Sets the width for the frame clock pulse. The query returns the width for the frame clock pulse.

One Bit Clock	Sets the width for the frame clock pulse to one bit clock.
One Subframe	Sets the width for the frame clock pulse to one subframe. The subframe is the width of one data word and is divided by the number of channels on that data line.
50% Duty Cycle	Sets the width for the frame clock pulse to 50% duty cycle. This makes the width of the frame clock pulse at 1/2 the width of the frame.

Parameter

Item	Type	Range of values	Default value
<width>	String	One Bit Clock|One Subframe|50% Duty Cycle	One Subframe

Examples

The following command sets the width for the frame clock pulse to 50% duty cycle.

```
INP:DIG:DSI:FSYN:WIDTH “50% Duty Cycle”
```

The following query returns the width for the frame clock pulse.

```
INP:DIG:DSI:FSYN:WIDTH?
```

Typical response:

```
50% Duty Cycle
```

INPut:DIGital:DSI:WBCLk:DIRection

Syntax

```
INPut:DIGital:DSI:WBCLk:DIRection <direction>
INPut:DIGital:DSI:WBCLk:DIRection?
```

Description

Sets the word clock and bit clock direction. The query returns the word clock and bit clock direction.

Parameter

Item	Type	Range of values	Default value
`<direction>`	Discrete	IN|OUT	IN

Examples

The following command sets the word clock and bit clock direction to IN.

```
INP:DIG:DSI:WBCL:DIR IN
```

The following query returns the word clock and bit clock direction.

```
INP:DIG:DSI:WBCL:DIR?
```

Typical response:

```
IN
```

INPut:DIGital:DSI:VOLTage

Syntax

```
INPut:DIGital:DSI:VOLTage <level>
```

```
INPut:DIGital:DSI:VOLTage?
```

Description

Sets the input logic level of the incoming signal for the DSI interface. The query returns the input logic level of the DSI interface.

Parameter

Item	Type	Range of values	Default value
<level>	Numeric	1.2 to 3.3 V	2.5 V

Examples

The following command sets the input logic level to 1.5 V.

```
INP:DIG:DSI:VOLT 1.5
```

The following query returns the input logic level.

```
INP:DIG:DSI:VOLT?
```

Typical response:

```
1.500000E+00
```

INPut:DIGital:FREQuency:SCALing

Syntax

```
INPut:DIGital:FREQuency:SCALing <scaling>,(@<channel_list>)
```

```
INPut:DIGital:FREQuency:SCALing?(@<channel_list>)
```

Description

Sets the reference sampling rate source to scale the frequency measurement for the specified channel(s). The query returns the frequency scaling source.

MISR	Measured input sampling rate
CUSTom	Custom reference sampling rate

Parameters

Item	Type	Range of values	Default value
<scaling>	Numeric	MISR|CUSTom	MISR
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remark

When the reference sampling rate source is set to custom, the value set at INPut:DIGital:REFerence:SRATe will be used as the reference sampling rate.

Examples

The following command sets the reference sampling rate source for channel 1 to measured input sampling rate.

```
INP:DIG:FREQ:SCAL MISR, (@D1)
```

The following query returns the frequency scaling source for channel 1.

```
INP:DIG:FREQ:SCAL? (@D1)
```

Typical response:

```
MISR
```

INPut:DIGital:REFerence:SRATe

Syntax

```
INPut:DIGital:REFerence:SRATe <sampling rate>[<unit>],
(@<channel_list>)
```

```
INPut:DIGital:REFerence:SRATe? (@<channel_list>)
```

Description

Sets the reference sampling rate for the specified channel(s). The query returns the reference sampling rate.

Parameters

Item	Type	Range of values	Default value
<sampling rate>	Numeric	6.75 kHz to 400 kHz	48 kHz
<channel list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remark

You can also include a multiplier for the unit, for example, kHz. The 'k' is the multiplier for the unit Hz.

Examples

The following command sets the reference sampling rate for channel 1 to 48 kHz.

```
INP:DIG:REF:SRAT 48000 (@D1)
```

The following query returns the reference sampling rate of channel 1.

```
INP:DIG:REF:SRAT? (@D1)
```

Typical response:

```
4.800000E+04
```

THIS PAGE HAS BEEN INTENTIONALLY LEFT BLANK.

10 Mass MEMory Subsystem

This chapter describes the `Mass MEMory` subsystem commands.

KEYSIGHT TECHNOLOGIES

MMEMory:CATalog?

Syntax

```
MMEMory:CATalog? <location>,<directory>
```

Description

Returns the memory usage information (total amount of storage currently used and free space available) in bytes and a list of files and directories in a specified parent directory. The specified parent directory can reside in the U8903B internal storage or an external USB flash storage. Multiple responses are separated by commas.

The response is in the following format:

<used_bytes_in_this_directory>,<free_bytes_on_this_disk>,
"<file_name>,<file_type>,<filesize_in_bytes>",
"<file_name>,<file_type>,<filesize_in_bytes>", ...

Parameters

Item	Type	Range of values	Default value
`<location>`	Discrete	INTernal|EXTernal	Required parameter
`<directory>`	String	The desired parent directory in quoted string.	Required parameter

Remarks

`INTernal` indicates the U8903B internal storage and `EXTernal` indicates an external USB flash storage.

Examples

The following query returns the existing files in the 'Filter' folder in the U8903B internal storage.

```
MMEM:CAT? INT,"\Filter"
```

The following query returns the existing files in an external USB flash storage.

```
MMEM:CAT? EXT,"\Storage 1"
```

MMEMory:DELete

Syntax

```
MMEMory:DELete <location>,<directory>,<filename>
```

Description

Deletes the specified file in the selected directory.

Parameters

Item	Type	Range of values	Default value
`<location>`	Discrete	INTernal|EXTernal	Required parameter
`<directory>`	String	The directory of the desired folder in quoted string.	Required parameter
`<filename>`	String	Can be any letters (A to Z or a to z), numbers (0 to 9) or underscore character ("_"). Blank spaces are not allowed.	Required parameter

Remarks

– `INTernal` indicates the U8903B internal storage and `EXTernal` indicates an external USB flash storage.

– The specified file must reside in the selected folder, otherwise an error will be generated. You can verify whether the file is available in the 'Filter' folder in the U8903B internal storage using the `MMEMory:CATalog? INTernal,"\Filter"` command.

Examples

The following command deletes a file named 'MyFilter.juf' in the '\Filter' directory of the U8903B internal storage.

```
MMEM:DEL INT,"\Filter","MyFilter.juf"
```

The following command deletes a file named 'MyFilter.juf' in the '\Storage 1' directory of an external USB flash storage.

```
MMEM:DEL EXT,"\Storage 1","MyFilter.juf"
```

MMEMory:LOAD

Syntax

```
MMEMory:LOAD <label>,<filename>
```

Description

Loads the 32-bit floating point data from a file into the U8903B. The `<filename>` parameter is a quoted string and the `<label>` parameter refers to an identifier for the data type to be recalled.

Parameters

Item	Type	Range of values	Default value
`<label>`	Discrete	FILTer|WAVeform|SWEep|DFILter	Required parameter
`<filename>`	String	Full file path in quoted string. For example, "\Storage 1\filter1.juf".	Required parameter

Remarks

– The file extension type for each <label> parameter is listed as follows.

FILTer	.juf
WAVeform	.arb
SWEep	.csv
DFilter	.juf

– The accessible U8903B internal storage folders are listed as follows.

Folders	Description
Filter	This folder is used for custom filter files storage.
Graph	This folder is used for graph data points files storage.
State	This folder is used for state files storage.
Sweep	This folder is used for sweep points files storage.
Temp	This folder is used for transferring data to and from the FTP site.
Waveform	This folder is used for arbitrary waveform files storage.

- For external USB flash storage, the file path must begin with '\Storage x' where x is the USB flash storage number that are plugged into the U8903B in sequence.
- The arbitrary waveform data file (.arb) is only applicable for the analog generator.
- For sweep mode, the **SOURce:SWEep:CHANnel** command must be sent prior to sending the **MMEMory:LOAD** command.
- When you send the **MMEMory:LOAD** command, the custom sweep points in the file will be loaded into the U8903B based on the sweep source channel set in the **SOURce:SWEep:CHANnel** command. The sweep mode will also be set to Automatic List or Manual list, depending on the previous sweep mode.

Examples

The following command loads the filter data into the U8903B from the 'filter1.juf' file in the U8903B internal storage.

```
MMEM:LOAD FILT,“\Filter\filter1.juf”
```

The following command loads the arbitrary waveform data into the U8903B from the 'waveform1.arb' file in the external USB flash storage.

```
MMEM:LOAD WAV,“\Storage 1\waveform1.arb”
```

The following command loads the sweep list values into the U8903B from the 'mySweep.csv' file in the U8903B internal storage.

```
MMEM:LOAD SWE,“\Sweep\mySweep.csv”
```

MMEMory:LOAD:STATe:CHANnel

Syntax

```
MMEMory:LOAD:STATe:CHANnel
<location>,<system_mode>,(@<channel_list>),<filename>
```

Description

Loads the specified single channel state file to the selected U8903B mode channel. The `<filename>` parameter is a quoted string and the `<location>` parameter refers to the storage location of the state file. The `<system_mode>` parameter refers to the U8903B mode of either analog analyzer or analog generator.

Parameters

Item	Type	Range of values	Default value
`<location>`	Discrete	INTernal|EXTernal	Required parameter
`<system_mode>`	Discrete	AANalyzer|AGENerator	Required parameter
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter
`<filename>`	String	Full file path in quoted string for the external USB flash storage. For example, "\Storage 1\GenCh1State.xml". For the internal storage, only the file name and extension are required. For example, "GenCh1State.xml".	Required parameter

Remarks

– `INTernal` indicates the U8903B internal storage and `EXTernal` indicates an external USB flash storage.

– If the specified state file is located in the internal storage, only the file name and extension are required. However, if the specified state file is located in an external USB flash storage, the file directory must be stated in full or an error will be generated.

– The file to be loaded must be a single channel state file and included in the `<system_mode>` parameter list.

– This command is not applicable for the sweep and graph analysis mode.

Examples

The following command loads a single channel state file named ‘GenCh1State.xml’ from the U8903B internal storage to the analog generator channel 1.

```
MMEM:LOAD:STAT:CHAN INT,AGEN,(@1),“GenCh1State.xml”
```

The following command loads a single channel state file named ‘AnaCh1State.xml’ in the root directory from the external USB flash storage to the analog analyzer channel 1 and channel 2.

```
MMEM:LOAD:STAT:CHAN EXT,AAN,(@1,2),“\Storage 1\AnaCh1State.xml”
```

MMEMory:LOAD:STATe[:MODE]

Syntax

```
MMEMory:LOAD:STATe[:MODE] <location>,<system_mode>,<filename>
```

Description

Loads the specified state file to the selected U8903B mode. The `<filename>` parameter is a quoted string and the `<location>` parameter refers to the storage location of the state file. The `<system_mode>` parameter refers to the U8903B mode of either analog analyzer, analog generator, sweep, or graph analysis.

Parameters

Item	Type	Range of values	Default value
`<location>`	Discrete	INTernal|EXTernal	Required parameter
`<system_mode>`	Discrete	AANalyzer|AGENerator|SWEep|GRAPh|DGENerator|DANalyzer	Required parameter
`<filename>`	String	Full file path in quoted string for the external USB flash storage. For example, "\Storage 1\GenCh1State.xml". For the internal storage, only the file name and extension are required. For example, "GenCh1State.xml".	Required parameter

Remarks

– `INTernal` indicates the U8903B internal storage and `EXTernal` indicates an external USB flash storage.

– If the specified state file is located in the internal storage, only the file name and extension are required. However, if the specified state file is located in the external USB flash storage, the file directory must be stated in full or an error will be generated.

– An error will be generated if the state file to be loaded is not a `<system_mode>` parameter.

Examples

The following command loads a generator mode state file named ‘GenState1.xml’ from the U8903B internal storage to the analog generator.

```
MMEM:LOAD:STAT INT,AGEN,“GenState1.xml”
```

The following command loads the analyzer mode state file named ‘AnaState2.xml’ in the root directory from the external USB flash storage to the analog analyzer.

```
MMEM:LOAD:STAT EXT,AAN,“\Storage 1\AnaState2.xml”
```

MMEMory:LOAD:STATe:PUP

Syntax

```
MMEMory:LOAD:STATe:PUP <power_up_state>
```

```
MMEMory:LOAD:STATe:PUP?
```

Description

Sets the power-up state of the U8903B. Select `LAST` to load the last settings of the U8903B, which are the settings before the U8903B was turned off. Select `DEFault` to load the default settings of the U8903B. The query returns the U8903B power-up state.

Parameter

Item	Type	Range of values	Default value
<power_up_state>	Discrete	DEFault|LAST	DEFault

Examples

The following command loads the U8903B last settings upon power up.

```
MMEM:LOAD:STAT:PUP LAST
```

The following query returns the power-up state of the U8903B.

```
MMEM:LOAD:STAT:PUP?
```

Typical response:

```
LAST
```

MMEMory:LOAD:TSEQuence:PROJect

Syntax

```
MMEMory:LOAD:TSEQuence:PROJect <filename>
```

Description

Loads a U8903B test sequence mode project setting from a file. The current project settings on the test sequence mode will be overwritten. The `<filename>` parameter is a quoted string.

Parameter

Item	Type	Range of values	Default value
<filename>	String	Full file path in quoted string. For example, – External USB flash storage "\Storage 1\project.xml" – Internal storage "project.xml"	Required parameter

Remarks

- The acceptable file type is restricted to XML file format (.xml).
- For external USB flash storage, the file path must begin with '\Storage x' where x is the USB flash storage number that are plugged into the U8903B in sequence.
- The test sequence must be in the STOP state before this command is allowed.
- When the specified file is located in the internal storage, only the file name and extension is needed. However, when the specified file is located in the external USB flash storage, the file name specified must be in full path, contained also the extension, otherwise an error is generated.

Example

The following command loads the test sequence mode project settings from "project.xml" in the internal storage.

```
MMEM:LOAD:TSEQ:PROJ "project.xml"
```

MMEMory:LOAD:WAVFile

Syntax

```
MMEMory:LOAD:WAVFile <filename>
```

Description

Loads a Microsoft compatible wave file (.wav) to the U8903B buffer. The `<filename>` parameter is a quoted string.

Parameter

Item	Type	Range of values	Default value
`<filename>`	String	Full file path in quoted string. For example, – External USB flash storage "\Storage 1\sine.wav" – Internal storage "\waveform\sine.wav"	Required parameter

Remarks

- The acceptable file type is restricted to wave file format (.wav).
- For the internal storage, the file path must begin with "\waveform\".
- For external USB flash storage, the file path must begin with '\Storage x' where x is the USB flash storage number that are plugged into the U8903B in sequence.
- The supported data resolution is 8, 16, and 24 bits per sample.
- According to the definition of the wave file format, the PCM data is two's-complement except for resolutions of 1 to 8 bits, which are represented as offset binary. Therefore, for wave file with 8 bits per sample resolution, the data will automatically be converted to two's-complement at the output.

Examples

The following command loads the ‘sine.wav’ file in the U8903B internal storage into the U8903B buffer.

```
MMEM:LOAD:WAVF “\waveform\sine.wav”
```

The following command loads the ‘sine.wav’ file in the external USB flash storage into the U8903B buffer.

```
MMEM:LOAD:WAVF “\Storage 1\sine.wav”
```

MMEMory:STORe

Syntax

```
MMEMory:STORe <label>,<filename>
```

Description

Stores the 32-bit floating point data to a file in either the U8903B internal storage or an external USB flash storage. The <filename> parameter is a quoted string and the <label> parameter refers to an identifier for the data type to be saved.

Parameters

Item	Type	Range of values	Default value
`<label>`	Discrete	FILTer|WAVeform|DFILter	Required parameter
`<filename>`	String	Full file path in quoted string. For example, "\Storage 1\filter1.juf".	Required parameter

Remarks

– The file extension type for each `<label>` parameter is listed as follows.

FILTer	.juf
WAVeform	.arb
DFILter	.juf

– The accessible U8903B internal storage folders are listed as follows.

Folders	Description
Filter	This folder is used for custom filter files storage.
Waveform	This folder is used for arbitrary waveform files storage.

– For external USB flash storage, the file path must begin with '\Storage x' where x is the USB flash storage number that are plugged into the U8903B in sequence.

Examples

The following command stores the filter data to a file named ‘filter1.juf’ in the U8903B internal storage.

```
MMEM:STOR FILT,“\Filter\filter1.juf”
```

The following command stores the arbitrary waveform data to a file names ‘waveform1.arb’ in the external USB flash storage.

```
MMEM:STOR WAV,“\Storage 1\waveform1.arb”
```

MMEMory:STORe:STATe:CHANnel

Syntax

```
MMEMory:STORe:STATe:CHANnel
<location>,<system_mode>,(@<channel>),<filename>
```

Description

Stores the current U8903B single channel state to a file in either the internal storage or an external USB flash storage. The `<filename>` parameter is a quoted string and the `<location>` parameter refers to the storage location of the state file. The `<system_mode>` parameter refers to the U8903B mode of either analog analyzer or analog generator.

Parameters

Item	Type	Range of values	Default value
`<location>`	Discrete	INTernal|EXTernal	Required parameter
`<system_mode>`	Discrete	AANalyzer|AGENerator	Required parameter
`<channel>`	Numeric	1|2|3|4|5|6|7|8	Required parameter
`<filename>`	String	Full file path in quoted string for the external USB flash storage. For example, "\Storage 1\GenCh1State.xml". For the internal storage, only the file name and extension are required. For example, "GenCh1State.xml".	Required parameter

Remarks

- `INTernal` indicates the U8903B internal storage and `EXTernal` indicates an external USB flash storage.
- All state files are saved in the XML file format (.xml).
- The stored channel state file may be loaded to any other channel but must be within the same system mode. For example, if you have stored the analyzer channel 1 state to a file named 'AnaCh1State.xml', then you may load the 'AnaCh1State.xml' file to channel 2 within the analyzer mode.
- This command is not applicable for the sweep and graph analysis mode.

Examples

The following command stores the analog generator channel 1 state to a file named 'GenCh1State.xml' in the U8903B internal storage.

```
MMEM:STOR:STAT:CHAN INT,AGEN,(@1),"GenCh1State.xml"
```

The following command stores the analog analyzer channel 2 state to a file named 'AnaCh2State.xml' in the '\Storage 1' directory of an external USB flash storage.

```
MMEM:STOR:STAT:CHAN EXT,AAN,(@2),"\Storage 1\AnaCh2State.xml"
```

MMEMory:STORe:STATe[:MODE]

Syntax

```
MMEMory:STORe:STATe[:MODE] <location>,<system_mode>,<filename>
```

Description

Stores the current U8903B state to a file in either the internal storage or an external USB flash storage. The `<filename>` parameter is a quoted string and the `<location>` parameter refers to the storage location of the state file. The `<system_mode>` parameter refers to the U8903B mode of either analog analyzer, analog generator, sweep, or graph analysis.

Parameters

Item	Type	Range of values	Default value
`<location>`	Discrete	INTernal|EXTernal	Required parameter
`<system_mode>`	Discrete	AANalyzer|AGENerator|SWEep|GRAPh|DGENerator|DANalyzer	Required parameter
`<filename>`	String	Full file path in quoted string for the external USB flash storage. For example, "\Storage 1\GenCh1State.xml". For the internal storage, only the file name and extension are required. For example, "GenCh1State.xml".	Required parameter

Remarks

– `INTernal` indicates the U8903B internal storage and `EXTernal` indicates an external USB flash storage.

– All state files are saved in the XML file format (.xml).

Examples

The following command stores the analog generator state to a file named ‘GenState1.xml’ in the U8903B internal storage.

```
MMEM:STOR:STAT INT,AGEN,“GenState1.xml”
```

The following command stores the analog analyzer state to a file named ‘AnaState2.xml’ in the root directory of an external USB flash storage.

```
MMEM:STOR:STAT EXT,AAN,“\Storage 1\AnaState2.xml”
```

MMEMory:STORe:SWEep

Syntax

```
MMEMory:STORe:SWEep <channel>,<function>,<filename>
```

Description

Stores the measured sweep data points from memory to a file. The `<channel>` parameter refers to the measured channel used in the sweep process. The `<function>` parameter refers to the particular measurement function used in the sweep process. The `<filename>` parameter is a quoted string.

Parameters

Item	Type	Range of values	Default value
`<channel>`	Numeric	1|2|3|4|5|6|7|8	1
`<function>`	Discrete	FUNC1|FUNC2|FUNC3|FUNC4	FUNC1
`<filename>`	String	Full file path in quoted string for the external USB flash storage. For example, “\Storage 1\SweepPoints.csv”. For the internal storage, only the file name and extension are required. For example, “SweepPoints.csv”.	Required parameter

Remark

For external USB flash storage, the file path must begin with ‘\Storage x’ where x is the USB flash storage number that are plugged into the U8903B in sequence.

Example

The following command stores the measured sweep data points of function 2 in channel 2 to a file named ‘Sweep_CH2_F2.csv’ in the external USB flash storage.

```
MMEM:STOR:SWE 2,FUNC2,“\Storage 1\Sweep_CH2_F2.csv”
```

MMEMory:STORe:TSEQuence:PROJect

Syntax

```
MMEMory:STORe:TSEQuence:PROJect <filename>
```

Description

Stores the U8903B test sequence mode project setting to a file. The `<filename>` parameter is a quoted string.

Parameter

Item	Type	Range of values	Default value
`<filename>`	String	Full file path in quoted string. For example, – External USB flash storage "\Storage 1\project.xml" – Internal storage "project.xml"	Required parameter

Remarks

- The stored file type is in the XML file format (.xml).
- For external USB flash storage, the file path must begin with '\Storage x' where x is the USB flash storage number that are plugged into the U8903B in sequence.
- The test sequence must be in the STOP state before this command is allowed.
- When the specified file is located in the internal storage, only the file name and extension is needed. However, when the specified file is located in the external USB flash storage, the file name specified must be in full path, contained also the extension. Otherwise, an error is generated.

Example

The following command stores the test sequence mode project settings to "project.xml" in the internal storage.

```
MMEM:STOR:TSEQ:PROJ "project.xml"
```

MMEMory:STORe:TSEQuence:REPort

Syntax

```
MMEMory:STORe:TSEQuence:REPort <filename>
```

Description

Stores the U8903B test sequence mode report to a file. The `<filename>` parameter is a quoted string.

Parameter

Item	Type	Range of values	Default value
`<filename>`	String	Full file path in quoted string. For example, – External USB flash storage "\Storage 1\report.docx" – Internal storage "report.docx"	Required parameter

Remarks

- The stored file type is in the DOCX file format (.docx).
- For external USB flash storage, the file path must begin with '\Storage x' where x is the USB flash storage number that are plugged into the U8903B in sequence.
- The test sequence must be in the STOP state before this command is allowed.
- When the specified file is located in the internal storage, only the file name and extension is needed. However, when the specified file is located in the external USB flash storage, the file name specified must be in full path, contained also the extension, otherwise an error is generated.
- If the report is not available, error -200, "Execution Error" will be generated.

Example

The following command stores the test sequence mode report to "report.docx" in the internal storage.

```
MMEM:STOR:TSEQ:REP "report.docx"
```

Keysight U8903B
Audio Analyzer

Programmer's Reference

11 MEASure Subsystem

This chapter describes the `MEASure` subsystem commands.

KEYSIGHT TECHNOLOGIES

MEASure:DIGital:CSTatus:DATA?

Syntax

```
MEASure:DIGital:CSTatus:DATA? (@<channel_list>)
```

Description

Retrieves the 24 bytes of the channel status bytes.

Parameter

Item	Type	Range of values	Default value
`<channel_list>`	Discrete	One or more analyzer channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Example

The following command queries the 24 bytes of the channel status bytes for digital analyzer channel 1.

```
MEAS:DIG:CST:DATA? (@D1)
```

Typical Response:

```
#H00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00
```

MEASure:DIGital:CSTatus:BYTE?

Syntax

```
MEASure:DIGital:CSTatus:BYTE? <byte_number>,(@<channel_list>)
```

Description

Retrieves a particular byte of the channel status bytes in hexadecimal characters.

Parameter

Item	Type	Range of values	Default value
`<byte_number>`	Numeric	0 to 23	0
`<channel_list>`	Discrete	One or more analyzer channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Example

The following command queries the byte 8 of the channel status bytes for digital analyzer channel 1.

```
MEAS:DIG:CST:BYTE? 8, (@D1)
```

Typical Response:

```
#H00
```

MEASure:DIGital:CSTatus:FIELd?

Syntax

```
MEASure:DIGital:CSTatus:FIELd? <field_name>,(@<channel_list>)
```

Description

Retrieves the channel status bytes field value of a particular field name for the specified channel.

Parameter

Item	Type	Range of values	Default value
`<field_name>`	String	Refer to “**Appendix J: AES3/SPDIF Interface Channel Status Bits Field Names**” on page 679	
`<channel_list>`	Discrete	One or more analyzer channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Example

The following command queries the field value of the Audio Mode for digital analyzer channel 1.

```
MEAS:DIG:CST:FIEL? “Audio Mode”, (@D1)
```

Typical Response:

```
Linear PCM
```

MEASure:DIGital:USTatus:DATA?

Syntax

```
MEASure:DIGital:USTatus:DATA? (@<channel_list>)
```

Description

Retrieves the 24 bytes of the user status bytes.

Parameter

Item	Type	Range of values	Default value
<channel_list>	Discrete	One or more analyzer channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Example

The following command queries the 24 bytes of the user status bytes for digital analyzer channel 1.

```
MEAS:DIG:UST:DATA? (@D1)
```

Typical Response:

```
#H00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00
```

MEASure:DIGital:USTatus:BYTE?

Syntax

```
MEASure:DIGital:USTatus:BYTE? <byte_number>,(@<channel_list>)
```

Description

Retrieves a particular byte of the user status bytes in hexadecimal characters.

Parameters

Item	Type	Range of values	Default value
<byte_number>	Numeric	0 to 23	0
<channel_list>	Discrete	One or more analyzer channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Examples

The following command queries the byte 8 of the user status bytes for digital analyzer channel 1.

```
MEAS:DIG:UST:BYTE? 8, (@D1)
```

Typical Response:

```
#H00
```

12 OUTPut Subsystem

KEYSIGHT
TECHNOLOGIES

This chapter describes the `OUTPut` subsystem commands.

OUTPut[:ANALog]:COMMiec60268

Syntax

```
OUTPut[:ANALog]:COMMiec60268 <common_config>,(@<channel_list>)
OUTPut[:ANALog]:COMMiec60268? (@<channel_list>)
```

Description

Sets the common mode connector configuration for the specified channel(s). The query returns the common mode connector configuration of the selected channel(s). Multiple responses are separated by commas.

The common mode connector configuration selection is described as follows.

10ohm Pin2	Add 10 ohm resistor to the pin 2 leg.
10ohm Pin3	Add 10 ohm resistor to the pin 3 leg.

Parameters

Item	Type	Range of values	Default value
<common_config>	Discrete	10ohm Pin2|10ohm Pin3	10ohm Pin2
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is only applicable for the common IEC60268 connector type.

Examples

The following command sets the common mode connector configuration for channel 1 to 10ohm Pin2.

```
OUTP:COMM60268 "10ohm Pin2",(@1)
```

The following query returns the common mode connector configuration for channel 1.

```
OUTP:COMM60268? (@1)
```

Typical response:

```
10ohm Pin2
```

OUTPut[:ANALog]:IMPedance

Syntax

```
OUTPut[:ANALog]:IMPedance <impedance>,(@<channel_list>)
OUTPut[:ANALog]:IMPedance? (@<channel_list>)
```

Description

Sets the analog generator output impedance for the specified channel(s). The query returns the output impedance of the selected channel(s). Multiple responses are separated by commas.

The output impedance selection is described as follows.

IMP20	Output impedance is 20 Ω for the unbalanced output connection.
IMP40	Output impedance is 40 Ω for the balanced, common, and common IEC60268 output connections.
IMP50	Output impedance is 50 Ω for the unbalanced output connection.
IMP100	Output impedance is 100 Ω for the balanced, common, and common IEC60268 output connections.
IMP600	Output impedance is 600 Ω for the unbalanced, balanced, common, and common IEC60268 output connections.

Parameters

Item	Type	Range of values	Default value
<impedance>	Discrete	– IMP40|IMP100|IMP600 (balanced, common, or common IEC60268 output connection) – IMP20|IMP50|IMP600 (unbalanced output connection)	IMP600
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

You must set the output connection type before configuring the output impedance.

Examples

The following command sequence sets the analog generator output impedance for channel 1 and channel 2 to 50 Ω and 100 Ω respectively. Assume that the output connection for channel 1 has been set to unbalanced, and channel 2 to balanced.

```
OUTP:IMP IMP50,(@1)
OUTP:IMP IMP100,(@2)
```

The following query returns the output impedance of channel 1 and channel 2.

```
OUTP:IMP? (@1,2)
```

Typical response:

```
IMP50,IMP100
```

OUTPut[:ANALog]:LOW

Syntax

```
OUTPut[:ANALog]:LOW <low>,(@<channel_list>)
OUTPut[:ANALog]:LOW? (@<channel_list>)
```

Description

Sets the analog generator low signal terminal to ground or allows it to float for the specified channel(s). The query returns the analog generator low signal terminal of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<low>`	Discrete	FLOat|GROund	FLOat
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Examples

The following command sets the analog generator low signal terminal for channel 3 to FLOat.

```
OUTP:LOW FLO,(@3)
```

The following query returns the analog generator low signal terminal for channel 3.

```
OUTP:LOW? (@3)
```

Typical response:

```
FLO
```

OUTPut[:ANALog]:STATe

Syntax

```
OUTPut[:ANALog]:STATe <state>,(@<channel_list>)
```

```
OUTPut[:ANALog]:STATe? (@<channel_list>)
```

Description

Enables or disables the analog generator output for the specified channel(s). The query returns the output state of the selected channel(s) as `0` if the output state is `OFF`, or `1` if the output state is `ON`. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	OFF
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

You must configure the output connection and impedance before setting the output state.

Examples

The following command sequence enables the analog generator channel 1 output but disables the channel 2 output.

```
OUTP:STAT ON,(@1)
OUTP:STAT OFF,(@2)
```

The following query returns the output states of channels 1 and 2.

```
OUTP:STAT? (@1,2)
```

Typical response:

```
1,0
```

OUTPut[:ANALog]:TYPE

Syntax

```
OUTPut:TYPE <type>,(@<channel_list>)
```

```
OUTPut:TYPE? (@<channel_list>)
```

Description

Sets the analog generator output connection type for the specified channel(s). The query returns the output connection type of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<type>`	Discrete	BALanced|UNBalanced|COMMon|COMMiec60268	UNBalanced
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- You are required to reconfigure the output impedance each time you change the output connection.
- When the output connection type is set to `COMMiec60268`, the output impedance for the output type is set to 40 Ω.
- When the output connection type is set from `COMMiec60268` to `BALanced`, `UNBalanced`, or `COMMon`, the output impedance for the output type is set to the assigned value.
- When the output connection type is set from `UNBalanced` to `BALanced` or `COMMon`, or vice versa, the output impedance for the output type is set according to the matching impedance value.
  - 20/40 Ω
  - 50/100 Ω
  - 600/600 Ω

Examples

The following command sequence sets the generator outputs for channel 1 and channel 2 to UNBalanced and BALanced respectively.

```
OUTP:TYPE UNB,(@1)
OUTP:TYPE BAL,(@2)
```

The following query returns the output connection types of channels 1 and 2.

```
OUTP:TYPE? (@1,2)
```

Typical response:

```
UNB,BAL
```

OUTPut[:ANALog]:VOLTage:MAXimum

Syntax

```
OUTPut[:ANALog]:VOLTage:MAXimum <voltage>,(@<channel_list>)
OUTPut[:ANALog]:VOLTage:MAXimum? (@<channel_list>)
```

Description

Sets the maximum output voltage for the specified channel(s). The query returns the maximum output voltage of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<voltage>`	Discrete	0 Vrms to 22.6 Vrms	22.6
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Examples

The following command sets the maximum output voltage for channel 1 to 10 Vrms.

```
OUTP:VOLT:MAX 10,(@1)
```

The following query returns the maximum output voltage for channel 1.

```
OUTP:VOLT:MAX? (@1)
```

Typical response:

```
10
```

OUTPut:DIGital:TYPE

Syntax

```
OUTPut:DIGital:TYPE <type>
```

```
OUTPut:DIGital:TYPE?
```

Description

Sets the digital generator AES3/SPDIF output connection. When a connection type is selected, the AES3/SPDIF output will be turned on. The query returns the AES3/SPDIF output connection type.

Parameter

Item	Type	Range of values	Default value
<type>	Discrete	BALanced|UNBalanced	UNBalanced

Examples

The following command sets the digital generator AES3/SPDIF output to Balanced.

```
OUTP:DIG:TYPE BAL
```

The following query returns the AES3/SPDIF output connection type.

```
OUTP:DIG:TYPE?
```

Typical response:

```
BAL
```

OUTPut:DIGital:SRATe

Syntax

```
OUTPut:DIGital:SRATe <sampling_rate> [<unit>]
```

```
OUTPut:DIGital:SRATe?
```

Description

Sets the sampling rate of the digital generator output signals. The query returns the sampling rate of the digital generator output signals.

Parameter

Item	Type	Range of values	Default value
`<sampling_rate>`	Numeric	6.75 kHz to 400 kHz	48 kHz
`<unit>`	Discrete	Hz	Hz

Remarks

- For AES3/SPDIF output connection, the range is limited to 28 kHz to 192 kHz.
- The AES3/SPDIF output will be turned off, if the sampling rate exceeds the range.

Examples

The following command sets the sampling rate of the digital generator output signal to 32 kHz.

```
OUTP:DIG:SRAT 32kHz
```

The following query returns the sampling rate of the digital generator output signal.

```
OUTP:DIG:SRAT?
```

Typical response:

```
3.200000E+04
```

OUTPut:DIGital:STATe

Syntax

```
OUTPut:DIGital:STATe <state>, (@<channel_list>)
OUTPut:DIGital:STATe? (@<channel_list>)
```

Description

Enables or disables the digital generator output for the specified channel(s). The query returns the output state of the selected channel(s) as 0 if the output state is OFF, or 1 if the output state is ON. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<state>	Boolean	OFF(0)|ON(1)	OFF
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Examples

The following commands enable the channel 1 digital generator output but disable the output for channel 2.

```
OUTP:DIG:STAT ON, (@D1)
OUTP:DIG:STAT OFF, (@D2)
```

The following query returns the output states of channel 1 and 2.

```
OUTP:DIG:STAT? (@D1,D2)
```

Typical response:

```
1,0
```

OUTPut:DIGital:AUDio[:ENCoding]:FORMat

Syntax

```
OUTPut:DIGital:AUDio[:ENCoding]:FORMat <encoding_format>
```

```
OUTPut:DIGital:AUDio[:ENCoding]:FORMat?
```

Description

Sets the audio encoding format of the embedded digital generator audio signals. The query returns the audio encoding format.

LPCM	Linear Pulse Code Modulation
ULAW	µ-Law encoding format
ALAW	A- Law encoding format

Parameter

Item	Type	Range of values	Default value
<encoding_format>	Discrete	LPCM|ULAW|ALAW	LPCM

Examples

The following command sets the encoding format of the digital generator to A- Law.

```
OUTP:DIG:AUD:FORM ALAW
```

The following query returns the encoding format.

```
OUTP:DIG:AUD:FORM?
```

Typical response:

```
ALAW
```

OUTPut:DIGital:AES:STATe

Syntax

```
OUTPut:DIGital:AES:STATe <state>
```

```
OUTPut:DIGital:AES:STATe?
```

Description

Enables or disables the AES3/SPDIF output for the digital generator. The query returns the AES3/SPDIF output state as 0 if the output state is OFF, or 1 if the output state is ON.

Parameter

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	ON

Remark

- The AES3/SPDIF output can also be enabled by setting the AES3/SPDIF output connection type using the **OUTPut:DIGital:TYPE** command.

Examples

The following command enables the AES3/SPDIF output.

```
OUTP:DIG:AES:STAT ON
```

The following query returns the output state of the AES3/SPDIF output.

```
OUTP:DIG:AES:STAT?
```

Typical response:

```
1
```

OUTPut:DIGital:AES:VOLTage

Syntax

```
OUTPut:DIGital:AES:VOLTage <level>
```

```
OUTPut:DIGital:AES:VOLTage?
```

Description

Sets the output logic level for the AES3/SPDIF interface in Peak-to-Peak Voltage (Vpp). The query returns the output logic level of the AES3/SPDIF interface.

Parameter

Item	Type	Range of values	Default value
`<level>`	Numeric	– 0.3 Vpp to 5.1 Vpp (Balanced output) – 0.3 Vpp to 2.5 Vpp (Unbalanced output)	2.5 Vpp

Examples

The following command sets the output logic level to 1.5 Vpp.

```
OUTP:DIG:AES:VOLT 1.5
```

The following query returns the output logic level.

```
OUTP:DIG:AES:VOLT?
```

Typical response:

```
1.500000E+00
```

OUTPut:DIGital:AES:AUDio:RESolution

Syntax

```
OUTPut:DIGital:AES:AUDio:RESolution <resolution>
OUTPut:DIGital:AES:AUDio:RESolution?
```

Description

Sets the audio resolution or bit depth for the AES3/SPDIF interface audio data to be generated. The query returns the audio resolution in integer.

Parameter

Item	Type	Range of values	Default value
<resolution>	Numeric	8 to 24	24

Examples

The following command sets the audio resolution to 20 bits.

```
OUTP:DIG:AES:AUD:RES 20
```

The following query returns the audio resolution.

```
OUTP:DIG:AES:AUD:RES?
```

Typical response:

```
20
```

OUTPut:DIGital:AES:AUDio:VALidity

Syntax

```
OUTPut:DIGital:AES:AUDio:VALidity <validity>
OUTPut:DIGital:AES:AUDio:VALidity?
```

Description

Sets the validity bit (bit 28) of the AES3/SPDIF interface output. When the validity bit is set to 0, the data is valid and is normally a linear coded PCM audio, and when it is set to 1, the data is invalid or may be a valid compressed audio. The query returns the validity bit value in integer.

Parameter

Item	Type	Range of values	Default value
<validity>	Numeric	0|1	0

Examples

The following command sets the AES3/SPDIF interface output validity bit to 1.

```
OUTP:DIG:AES:AUD:VAL 1
```

The following query returns the AES3/SPDIF interface output validity bit value.

```
OUTP:DIG:AES:AUD:VAL?
```

Typical response:

```
1
```

OUTPut:DIGital:AES[:PROTocol]:MODE

Syntax

```
OUTPut:DIGital:AES[:PROTocol]:MODE <mode>,(@<channel_list>)
OUTPut:DIGital:AES[:PROTocol]:MODE?(@<channel_list>)
```

Description

Sets the first bit in the channel status block of the AES3/SPDIF interface to indicate the mode. The first bit of the channel status block is `0` if the mode is `CONSumer`, or `1` if the mode is `PROFessional`. The query returns the channel status block mode.

Parameters

Item	Type	Range of values	Default value
`<mode>`	Discrete	PROFessional|CONSumer	CONSumer
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Examples

The following command sets the channel status block mode for channel 1 to Consumer.

```
OUTP:DIG:AES:MODE CONS, (@D1)
```

The following query returns the channel status block mode of channel 1.

```
OUTP:DIG:AES:MODE? (@D1)
```

Typical response:

```
CONS
```

OUTPut:DIGital:AES[:PROTocol]:CSTatus:DATA

Syntax

```
OUTPut:DIGital:AES[:PROTocol]:CSTatus:DATA <data>,
(@<channel_list>)
```

```
OUTPut:DIGital:AES[:PROTocol]:CSTatus:DATA? (@<channel_list>)
```

Description

Sets the 24 bytes array of the channel status bits in hexadecimal characters for the specified channel(s). The query returns the channel status bits 24 bytes array in hexadecimal characters.

Parameters

Item	Type	Range of values	Default value
`<data>`	Hex		
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- #H must be added in front of the bytes array to indicate that the array is in hexadecimal characters.
- In Professional mode, the byte-23 (CRCC) is only for query and is not modifiable. The byte-23 value will be updated with the auto-computed CRCC value.
- The channel status bits are arranged in the format of byte-0 to byte-23. For example, the value `#H2F2C6CFBD800553839304455431B0E704008E1553010000` is arranged as follows.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
2F	2C	6C	FB	D8	00	55	38	39	30	44	55	54	31	B0	E7	04	00	8E	15	53	01	00	00

Examples

The following command sets the 24 bytes array of the channel status bits for channel 1.

```
OUTP:DIG:AES:CST:DATA
#H2F2C6CFBD80055383930445554 31B0E704008E1553010000, (@D1)
```

The following query returns the 24 bytes array of the channel status bits of channel 1.

```
OUTP:DIG:AES:CST:DATA? (@D1)
```

Typical response:

```
#H2F2C6CFBD8005538393044555431B0E704008E1553010000
```

OUTPut:DIGital:AES[:PROTocol]:CSTatus:BYTE

Syntax

```
OUTPut:DIGital:AES[:PROTocol]:CSTatus:BYTE <number>, <value>,
(@<channel_list>)
```

```
OUTPut:DIGital:AES[:PROTocol]:CSTatus:BYTE? <number>,
(@<channel_list>)
```

Description

Sets a particular byte of the channel status bits for the specified channel(s). The query returns the byte(s) value in hexadecimal characters.

Parameters

Item	Type	Range of values	Default value
`<number>`	Numeric	0 to 23	0
`<value>`	Hex	0 to FF	0
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- #H must be added in front of the byte value to indicate that the value is in hexadecimal characters.
- In Professional mode, the byte-23 (CRCC) is only for query and is not modifiable. The byte-23 value will be updated with the auto-computed CRCC value.
- The channel status bits are arranged in the format of byte-0 to byte-23. For example, the value `#H2F2C6CFBD80055383930445554 31B0E704008E1553010000` is arranged as follows.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
2F	2C	6C	FB	D8	00	55	38	39	30	44	55	54	31	B0	E7	04	00	8E	15	53	01	00	00

Examples

The following command sets the byte 2 of the channel status bits for channel 1 to 6F.

```
OUTP:DIG:AES:CST:BYTE 2, #H6F, (@D1)
```

The following query returns the byte 2 of the channel status bits of channel 1.

```
OUTP:DIG:AES:CST:BYTE? 2, (@D1)
```

Typical response:

```
#H6F
```

OUTPut:DIGital:AES[:PROTocol]:CSTatus:FIELd

Syntax

```
OUTPut:DIGital:AES[:PROTocol]:CSTatus:FIELd <name>, <value>,
(@<channel_list>)
```

```
OUTPut:DIGital:AES[:PROTocol]:CSTatus:FIELd? <name>,
(@<channel_list>)
```

Description

Sets the channel status bits data of a specified field name for the specified channel(s). The query returns the data of the channel status bits data of a specified field name.

Parameters

Item	Type	Range of values	Default value
`<name>`	String	Refer to "**Appendix J: AES3/SPDIF Interface Channel Status Bits Field Names**" on page 679	
`<value>`	String	Refer to "**Appendix J: AES3/SPDIF Interface Channel Status Bits Field Names**" on page 679	
`<channel list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- The field name is not case sensitive.
- The field value is case sensitive.
- For Word Length, Channel Number, and Multichannel Mode, there is a condition before the field value can be set. Refer to "**Appendix J: AES3/SPDIF Interface Channel Status Bits Field Names**" on page 679 for more information.
- Before setting the respective fields or querying, ensure that the correct mode is set.

Examples

The following command sets the field value of “Category Code” for channel 1 to “Musical Instrument”.

```
OUTP:DIG:AES:CST:FIEL “Category Code”, “Musical Instrument”,
(@D1)
```

The following query returns the field value of “Category Code” for channel 1.

```
OUTP:DIG:AES:CST:FIEL? “Category Code”, (@D1)
```

Typical response:

```
Musical Instrument
```

OUTPut:DIGital:AES[:PROTocol]:USTatus:DATA

Syntax

```
OUTPut:DIGital:AES[:PROTocol]:USTatus:DATA <data>,
(@<channel_list>)
```

```
OUTPut:DIGital:AES[:PROTocol]:USTatus:DATA? (@<channel_list>)
```

Description

Sets the 24 bytes array of the user status bits in hexadecimal characters for the specified channel(s). The query returns the user status bits 24 bytes array in hexadecimal characters.

Parameters

Item	Type	Range of values	Default value
`<data>`	Hex		
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- #H must be added in front of the bytes array to indicate that the array is in hexadecimal characters.
- The channel status bits are arranged in the format of byte-0 to byte-23. For example, the value `#H2F2C6CFBD800553839304455431B0E704008E1553010000` is arranged as follows.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
2F	2C	6C	FB	D8	00	55	38	39	30	44	55	54	31	B0	E7	04	00	8E	15	53	01	00	00

Example

The following command sets the 24 bytes array of the user status bits for channel 1.

```
OUTP:DIG:AES:UST:DATA
#H2F2C6CFBD800553839304455543 1B0E704008E1553010000, (@D1)
```

The following query returns the 24 bytes array of the user status bits of channel 1.

```
OUTP:DIG:AES:UST:DATA? (@D1)
```

Typical response:

```
#H2F2C6CFBD8005538393044555431B0E704008E1553010000
```

OUTPut:DIGital:AES[:PROTocol]:USTatus:BYTE

Syntax

```
OUTPut:DIGital:AES[:PROTocol]:USTatus:BYTE <number>, <value>,
(@<channel_list>)
```

```
OUTPut:DIGital:AES[:PROTocol]:USTatus:BYTE? <number>,
(@<channel_list>)
```

Description

Sets a particular byte of the user status bits for the specified channel(s). The query returns the byte(s) value in hexadecimal characters.

Parameters

Item	Type	Range of values	Default value
`<number>`	Numeric	0 to 23	0
`<value>`	Hex	0 to FF	0
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

– #H must be added in front of the bytes value to indicate that the value is in hexadecimal characters.

– The channel status bits are arranged in the format of byte-0 to byte-23. For example, the value `#H2F2C6CFBD8005538393044555431B0E704008E1553010000` is arranged as follows.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
2F	2C	6C	FB	D8	00	55	38	39	30	44	55	54	31	B0	E7	04	00	8E	15	53	01	00	00

Examples

The following command sets the byte 2 of the user status bits for channel 1 to 6F.

```
OUTP:DIG:AES:UST:BYTE 2, #H6F, (@D1)
```

The following query returns the byte 2 of the user status bits of channel 1.

```
OUTP:DIG:AES:UST:BYTE? 2, (@D1)
```

Typical response:

```
#H6F
```

OUTPut:DIGital:DSI:VOLTage

Syntax

```
OUTPut:DIGital:DSI:VOLTage <level>
```

```
OUTPut:DIGital:DSI:VOLTage?
```

Description

Sets the output logic level for the digital serial interface (DSI) in Volts (V). The query returns the output logic level of the DSI.

Parameter

Item	Type	Range of values	Default value
`<level>`	Numeric	1.2 V to 3.3 V	2.5 V

Examples

The following command sets the output logic level to 1.5 V.

```
OUTP:DIG:DSI:VOLT 1.5
```

The following query returns the output logic level.

```
OUTP:DIG:DSI:VOLT?
```

Typical response:

```
1.500000E+00
```

OUTPut:DIGital:DSI:AUDio:RESolution

Syntax

```
OUTPut:DIGital:DSI:AUDio:RESolution <resolution>
```

```
OUTPut:DIGital:DSI:AUDio:RESolution?
```

Description

Sets the audio resolution or bit depth for the DSI interface audio data to be generated. The query returns the audio resolution in integer.

Parameter

Item	Type	Range of values	Default value
<resolution>	Numeric	8 to 24	24

Remark

The audio resolution must be less than or equal to the word length.

Examples

The following command sets the audio resolution to 20 bits.

```
OUTP:DIG:DSI:AUD:RES 20
```

The following query returns the audio resolution.

```
OUTP:DIG:DSI:AUD:RES?
```

Typical response:

```
20
```

OUTPut:DIGital:DSI:AUDio:WLENgth

Syntax

```
OUTPut:DIGital:DSI:AUDio:WLENgth <length>
```

```
OUTPut:DIGital:DSI:AUDio:WLENgth?
```

Description

Sets the word length for the DSI interface. The query returns the word length.

Parameter

Item	Type	Range of values	Default value
`<length>`	Numeric	8 to 32	32

Remarks

- The word length must be greater or equal to the audio resolution.
- Refer to "**Appendix K: Word Length, Sampling Rate, and Multiplier for DSI Interface**" on page 681 for the range of word length that can be set with different DSI multiplier and sampling rate.
- When setting the word length, the error message, –221,"Settings conflict..." may be generated. This error message can be ignored as this notifies that the word length or multiplier is auto adjusted to the nearest allowable value due to the settings conflict.

Examples

The following command sets the word length to 20 bits.

```
OUTP:DIG:DSI:AUD:WLEN 20
```

The following query returns the word length.

```
OUTP:DIG:DSI:AUD:WLEN?
```

Typical response:

```
20
```

OUTPut:DIGital:DSI:DATA:FORMat

Syntax

```
OUTPut:DIGital:DSI:DATA:FORMat <format>
```

```
OUTPut:DIGital:DSI:DATA:FORMat?
```

Description

Sets the format for the DSI interface audio data to be generated. The query returns the format for the DSI interface audio data.

LEFT	Left justified. The active data bits are filled to the left edge of the data word.
RIGHt	Right justified. The active data bits are filled to the right edge of the data word.
IIS	IIS format. The word clock is low for the first channel (left channel) and high for the second channel (right channel)
DSP	The period of the word clock is only 1 bit of the bit clock and the data is 1 bit clock delay from the beginning of the word clock.

Parameter

Item	Type	Range of values	Default value
<format>	Discrete	LEFT|RIGHt|IIS|DSP	LEFT

Examples

The following command sets the DSI data format to right.

```
OUTP:DIG:DSI:DATA:FORM RIGH
```

The following query returns the DSI data format.

```
OUTP:DIG:DSI:DATA:FORM?
```

Typical response:

```
RIGH
```

OUTPut:DIGital:DSI:DATA:SHIFt:COUNt

Syntax

```
OUTPut:DIGital:DSI:DATA:SHIFt:COUNt <number>
```

```
OUTPut:DIGital:DSI:DATA:SHIFt:COUNt?
```

Description

Sets the number for the data bits in relative to the frame clock to be shifted. The query returns the number for the data bits.

Parameter

Item	Type	Range of values	Default value
`<number>`	Numeric	0 to 8	0

Remark

Use the **OUTPut:DIGital:DSI:DATA:SHIFt:DIRection** command to shift the data bits right or left in relative to the frame clock.

Examples

The following command shifts the data to the right of the frame clock by 2 bits.

```
OUTP:DIG:DSI:DATA:SHIF:DIR RIGH
```

```
OUTP:DIG:DSI:DATA:SHIF:COUN 2
```

The following query returns the shifted number for the data bits.

```
OUTP:DIG:DSI:DATA:SHIF:COUN?
```

Typical response:

```
2
```

OUTPut:DIGital:DSI:DATA:SHIFt:DIRection

Syntax

```
OUTPut:DIGital:DSI:DATA:SHIFt:DIRection <direction>
```

```
OUTPut:DIGital:DSI:DATA:SHIFt:DIRection?
```

Description

Sets the direction for the data bits in relative to the frame clock to be shifted. The query returns the direction for the data bits.

Parameter

Item	Type	Range of values	Default value
`<direction>`	Discrete	LEFT|RIGHt	RIGHt

Remark

Use the **OUTPut:DIGital:DSI:DATA:SHIFt:DIRection** command to shift the data bits right or left in relative to the frame clock.

Examples

The following command shifts the data to the left of the frame clock by 1 bit.

```
OUTP:DIG:DSI:DATA:SHIF:DIR LEFT
```

```
OUTP:DIG:DSI:DATA:SHIF:COUN 1
```

The following query returns the shifted direction for the data bits.

```
OUTP:DIG:DSI:DATA:SHIF:DIR?
```

Typical response:

```
LEFT
```

OUTPut:DIGital:DSI:FSYNc:POLarity

Syntax

```
OUTPut:DIGital:DSI:FSYNc:POLarity <polarity>
OUTPut:DIGital:DSI:FSYNc:POLarity?
```

Description

Sets the polarity for the frame clock. The query returns the polarity for the frame clock.

RISing	Sets the frame clock on the left channel of the data to high.
FALLing	Sets the frame clock on the left channel of the data to low.

Parameter

Item	Type	Range of values	Default value
<polarity>	Numeric	RISing|FALLing	RISing

Examples

The following command sets the polarity to falling.

```
OUTP:DIG:DSI:FSYN:POL FALLING
```

The following query returns the polarity.

```
OUTP:DIG:DSI:FSYN:POL?
```

Typical response:

```
FALL
```

OUTPut:DIGital:DSI:FSYNc:WIDTh

Syntax

```
OUTPut:DIGital:DSI:FSYNc:WIDTh <width>
OUTPut:DIGital:DSI:FSYNc:WIDTh?
```

Description

Sets the width for the frame clock pulse. The query returns the width for the frame clock pulse.

`One Bit Clock`	Sets the width for the frame clock pulse to one bit clock.
`One Subframe`	Sets the width for the frame clock pulse to one subframe. The subframe is the width of one data word and is divided by the number of channels on that data line.
`50% Duty Cycle`	Sets the width for the frame clock pulse to 50% duty cycle. This makes the width of the frame clock pulse at 1/2 the width of the frame.

Parameter

Item	Type	Range of values	Default value
`<width>`	Discrete	One Bit Clock|One Subframe|50% Duty Cycle	One Subframe

Examples

The following command sets the width for the frame clock pulse to 50% duty cycle.

```
OUTP:DIG:DSI:FSYN:WIDTH 50% Duty Cycle
```

The following query returns the width for the frame clock pulse.

```
OUTP:DIG:DSI:FSYN:WIDTH?
```

Typical response:

```
50% Duty Cycle
```

OUTPut:DIGital:DSI:MCLK:STATe

Syntax

```
OUTPut:DIGital:DSI:MCLK:STATe <state>
```

```
OUTPut:DIGital:DSI:MCLK:STATe?
```

Description

Enables or disables the DSI interface master clock. The query returns the DSI interface master clock state as `0` if the master clock state is `OFF`, or `1` if the master clock state is `ON`.

Parameter

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	ON

Examples

The following command disables the DSI master clock.

```
OUTP:DIG:DSI:MCLK:STAT OFF
```

The following query returns the DSI master clock state.

```
OUTP:DIG:DSI:MCLK:STAT?
```

Typical response:

```
0
```

OUTPut:DIGital:DSI:MCLK:MULTiplier

Syntax

```
OUTPut:DIGital:DSI:MCLK:MULTiplier <multiplier>
```

```
OUTPut:DIGital:DSI:MCLK:MULTiplier?
```

Description

Sets the multiplier that is used to determine the master clock rate. The master clock rate is based on the output sampling frequency and multiplier values. The query returns the multiplier value.

Parameter

Item	Type	Range of values	Default value
`<multiplier>`	Numeric	Refer to "**Appendix K: Word Length, Sampling Rate, and Multiplier for DSI Interface**" on page 681	128

Remark

Refer to "**Appendix K: Word Length, Sampling Rate, and Multiplier for DSI Interface**" on page 681 for the range of multiplier that can be set with different DSI word length and sampling rate.

Examples

The following command sets the multiplier to 256.

```
OUTP:DIG:DSI:MCLK:MULT 256
```

The following query returns the multiplier value.

```
OUTP:DIG:DSI:MCLK:MULT?
```

Typical response:

```
256
```

OUTPut:DIGital:DSI:MCLK:RATE?

Syntax

```
OUTPut:DIGital:DSI:MCLK:RATE?
```

Description

Queries the master clock rate.

Examples

The following query returns the master clock rate.

```
OUTP:DIG:DSI:MCLK:RATE?
```

Typical response:

```
6.144000E+06
```

OUTPut:DIGital:DSI:BCLK:SYNC

Syntax

```
OUTPut:DIGital:DSI:BCLK:SYNC <polarity>
OUTPut:DIGital:DSI:BCLK:SYNC?
```

Description

Sets the leading edge of the data to be synchronized to the rising edge or falling edge of the bit clock for the digital generator. The query returns the bit clock sync polarity type.

Parameter

Item	Type	Range of values	Default value
<polarity>	Discrete	RISing|FALLing	FALLing

Examples

The following command sets the polarity to the falling edge.

```
OUTP:DIG:DSI:BCLK:SYNC FALL
```

The following query returns the bit clock sync polarity type.

```
OUTP:DIG:DSI:BCLK:SYNC?
```

Typical response:

```
FALL
```

OUTPut:DIGital:OPTical:STATe

Syntax

```
OUTPut:DIGital:OPTical:STATe <state>
OUTPut:DIGital:OPTical:STATe?
```

Description

Enables or disables the optical output for digital generator. The query returns the optical output state as `0` if the optical output state is `OFF`, or `1` if the optical output state is `ON`.

Parameter

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	OFF

Examples

The following command enables the optical output.

```
OUTP:DIG:OPT:STAT ON
```

The following query returns the optical output state.

```
OUTP:DIG:OPT:STAT?
```

Typical response:

```
1
```

OUTPut:DIGital:RCLK:SOURce

Syntax

```
OUTPut:DIGital:RCLK:SOURce <source>
OUTPut:DIGital:RCLK:SOURce?
```

Description

Sets the system clock reference source type. The query returns the system clock reference source type.

`INTernal`	Internal clock
`EXTernal`	External clock
`AESRclock`	Recovered clock from AES3/SPDIF interface input

Parameter

Item	Type	Range of values	Default value
`<source>`	Discrete	INTernal|EXTernal|AESRclock	INTernal

Examples

The following command sets the system clock reference source type to external clock.

```
OUTP:DIG:RCLK:SOUR EXT
```

The following query returns the system clock reference source type.

```
OUTP:DIG:RCLK:SOUR?
```

Typical response:

```
EXT
```

OUTPut:DIGital:RCLK:EXTernal[:TYPE]

Syntax

```
OUTPut:DIGital:RCLK:EXTernal[:TYPE] <type>
```

```
OUTPut:DIGital:RCLK:EXTernal[:TYPE]?
```

Description

Sets the external clock source type. The query returns the external clock source type.

MCLK	Master clock in
FSYNc	Frame sync in

Parameter

Item	Type	Range of values	Default value
<type>	Discrete	MCLK|FSYNc	MCLK

Examples

The following command sets the external clock source type to frame sync in.

```
OUTP:DIG:RCLK:EXT FSYN
```

The following query returns the external clock type.

```
OUTP:DIG:RCLK:EXT?
```

Typical response:

```
FSYN
```

OUTPut:DIGital:RCLK:EXTernal:MCLK:WLENgth

Syntax

```
OUTPut:DIGital:RCLK:EXTernal:MCLK:WLENgth <length>
```

```
OUTPut:DIGital:RCLK:EXTernal:MCLK:WLENgth?
```

Description

Sets the word length of the master clock for the external clock source. The query returns the word length.

Parameter

Item	Type	Range of values	Default value
`<length>`	Numeric	8 to 32	32

Remark

– Refer to “**Example L: Word Length, Sampling Rate, and Multiplier for Master Clock In**” on page 687 for the range of word length that can be set with different master clockin multiplier and sampling rate.

– When setting the word length, the error message, –221,”Settings conflict...” may be generated. This error message can be ignored as this is to notify that the word length or multiplier is auto adjusted to the nearest allowable value due to the settings conflict.

Examples

The following command sets the word length to 20 bits.

```
OUTP:DIG:RCLK:EXT:MCLK:WLEN 20
```

The following query returns the word length.

```
OUTP:DIG:RCLK:EXT:MCLK:WLEN?
```

Typical response:

```
20
```

OUTPut:DIGital:RCLK:EXTernal:MCLK:MULTiplier

Syntax

```
OUTPut:DIGital:RCLK:EXTernal:MCLK:MULTiplier <multiplier>
```

```
OUTPut:DIGital:RCLK:EXTernal:MCLK:MULTiplier?
```

Description

Sets the multiplier of the master clock for the external clock source. The query returns the multiplier value.

Parameter

Item	Type	Range of values	Default value
`<multiplier>`	Numeric	Refer to "**Example L: Word Length, Sampling Rate, and Multiplier for Master Clock In**" on page 687	128

Remark

Refer to "**Example L: Word Length, Sampling Rate, and Multiplier for Master Clock In**" on page 687 for the range of multiplier that can be set with different master clock in word length and sampling rate.

Examples

The following command sets the multiplier to 512.

```
OUTP:DIG:RCLK:EXT:MCLK:MULT 512
```

The following query returns the multiplier value.

```
OUTP:DIG:RCLK:EXT:MCLK:MULT?
```

Typical response:

```
512
```

OUTPut:DIGital:SCLK:OUT:STATe

Syntax

```
OUTPut:DIGital:SCLK:OUT:STATe <state>
```

```
OUTPut:DIGital:SCLK:OUT:STATe?
```

Description

Enables or disables the sync clock output for digital generator. The query returns the sync clock output state as `0` if the sync clock output state is `OFF`, or `1` if the sync clock output state is `ON`.

Parameter

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	OFF

Examples

The following command disables the sync clock output.

```
OUTP:DIG:SCLK:OUT:STAT OFF
```

The following query returns the sync clock output state.

```
OUTP:DIG:SCLK:OUT:STAT?
```

Typical response:

0

OUTPut:DIGital:SCLK:OUT:SOURce

Syntax

```
OUTPut:DIGital:SCLK:OUT:SOURce <source>
```

```
OUTPut:DIGital:SCLK:OUT:SOURce?
```

Description

Sets the sync clock source type. The query returns the sync clock source type.

INTernal	Internal clock
EXTernal	External clock
AESRclock	Recovered clock from AES3/SPDIF interface input

Parameter

Item	Type	Range of values	Default value
<source>	Discrete	INTernal|EXTernal|AESRclock	INTernal

Examples

The following command sets the sync clock source type to external clock.

```
OUTP:DIG:SCLK:OUT:SOUR EXT
```

The following query returns the sync clock source type.

```
OUTP:DIG:SCLK:OUT:SOUR?
```

Typical response:

```
EXT
```

OUTPut:DIGital:SCLK:OUT:DIVider

Syntax

```
OUTPut:DIGital:SCLK:OUT:DIVider <divider>
OUTPut:DIGital:SCLK:OUT:DIVider?
```

Description

Sets the sync clock divider. The query returns the sync clock divider.

D1	Divide by 1
D128	Divide by 128

Parameter

Item	Type	Range of values	Default value
<divider>	Discrete	D1|D128	D1

Examples

The following command sets the sync clock divider to 128.

```
OUTP:DIG:SCLK:OUT:DIV D128
```

The following query returns the sync clock divider.

```
OUTP:DIG:SCLK:OUT:DIV?
```

Typical response:

```
D128
```

13 SENSe Subsystem

KEYSIGHT
TECHNOLOGIES

This chapter describes the `SENSe` subsystem commands.

SENSe[:ANALog]:AVERaging:MOVing:POINts

Syntax

```
SENSe[:ANALog]:AVERaging:MOVing:POINts
<number_of_points>,(@<channel_list>)
```

```
SENSe[:ANALog]:AVERaging:MOVing:POINts? (@<channel_list>)
```

Description

Controls the number of points to be included in the moving average for the specified channel(s). In moving averaging, when a new measurement data is added, the oldest data is discarded. The query returns the number of points in the moving average of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<number_of_points>	Numeric	1 to 20	1
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is only applicable in the analog analyzer mode but not in the frequency domain and time domain modes.

Examples

The following command sets eight averaging points for channel 1 and channel 2.

```
SENS:AVER:MOV:POIN 8,(@1,2)
```

The following query returns the number of averaging points for channel 1.

```
SENS:AVER:MOV:POIN? (@1)
```

Typical response:

```
8
```

SENSe[:ANALog]:AVERaging:SYNC:POINts

Syntax

```
SENSe:AVERaging:SYNC:POINts
<number_of_points>[,(@<channel_list>)]
```

```
SENSe:AVERaging:SYNC:POINts? [(@<channel_list>)]
```

Description

Sets the number of points for the synchronous averaging. Synchronous averaging reduces noise levels by averaging the acquired data in the time domain. The query returns the number of averaging points.

Parameters

Item	Type	Range of values	Default value
`<number_of_points>`	Numeric	1 to 64	1
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Optional parameter

Remarks

- This setting is only applicable if the graph trigger source is channel 1 to channel 8.
- If the channel is not specified in this command, the setting will be applied to channel 1 and channel 2.

Examples

The following command sets eight averaging points.

```
SENS:AVER:SYNC:POIN 8
```

The following query returns the number of averaging points.

```
SENS:AVER:SYNC:POIN?
```

Typical response:

```
8
```

SENSe[:ANALog]:CALibrator:LEVel

Syntax

```
SENSe[:ANALog]:CALibrator:LEVel
<calibrator_level>,(@<channel_list>)
```

```
SENSe[:ANALog]:CALibrator:LEVel? (@<channel_list>)
```

Description

Sets the calibrator level of the specified channel(s) in unit dBSPL. The sound pressure level in Vrms that is selected will be referenced to this value. This is a standard reference threshold level for acoustic measurement. The query returns the calibrator level of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<calibrator_level>`	Numeric	(–160 < calibrator level < 160) dBSPL	94
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Examples

The following command sets the calibrator level for analog analyzer channel 1 and channel 2 to 94 dBSPL.

```
SENS:CAL:LEV 94,(@1,2)
```

The following query returns the calibrator level for analog analyzer channel 1.

```
SENS:CAL:LEV? (@1)
```

Typical response:

```
94
```

SENSe[:ANALog]:FFT:WINDow

Syntax

```
SENSe[:ANALog]:FFT:WINDow <type>[,(@<channel_list>)]
SENSe[:ANALog]:FFT:WINDow? [(@<channel_list>)]
```

Description

Sets the window function for the frequency domain graph analysis. The window functions are time-domain weighting functions applied to the input time records to condition the results of a time domain to frequency domain transformation. The query returns the window function.

The window functions with their corresponding <type> parameters are listed as follows.

BLACkman	Blackman-Harris window
FLATtop	Flattop window
HAMMing	Hamming window
HANN	Hanning window
KAISer	Kaiser window
RECTangular	Rectangular window
RIFe1	Rife-Vincent 1 window
RIFe3	Rife-Vincent 3 window

Parameters

Item	Type	Range of values	Default value
<type>	Discrete	BLACkman|FLATtop|HAMMing|HANN|KAISer|RECTangular|RIFe1|RIFe3	KAISer
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Optional parameter

Remark

If the channel is not specified in this command, the setting will be applied to channel 1 and channel 2.

Examples

The following command sets the window function to the Hanning window.

```
SENS:FFT:WIND HANN
```

The following query returns the window function.

```
SENS:FFT:WIND?
```

Typical response:

```
HANN
```

SENSe[:ANALog]:FILTer:CLEar

Syntax

```
SENSe[:ANALog]:FILTer:CLEar (@<channel_list>)
```

Description

Turns off all the filters including the low-pass, high-pass, weighting, and de-emphasis filters for the selected channel(s).

Parameter

Item	Type	Range of values	Default value
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Example

The following command turns off all filters for channel 1 and channel 2.

```
SENS:FILT:CLE (@1,2)
```

SENSe[:ANALog]:FILTer:DEEMphasis

Syntax

```
SENSe[:ANALog]:FILTer:DEEMphasis <de-emphasis>,(@channel_list)
SENSe[:ANALog]:FILTer:DEEMphasis? (@channel_list)
```

Description

Sets the de-emphasis condition of the input signal for the specified channel(s). If there is no pre-emphasis on the input signal, then the de-emphasis is not required. The query returns the de-emphasis condition of the selected channel(s). Multiple responses are separated by commas.

None	No de-emphasis
50us	50 µs de-emphasis
75us	75 µs de-emphasis
Custom	User-defined de-emphasis

Parameters

Item	Type	Range of values	Default value
<de-emphasis>	String	NONE|50us|75us|Custom	NONE
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– To use a custom de-emphasis condition, you need to load the custom filter data into the U8903B before you can use it. To load a custom filter, use either the **DATA:FILTer** or **MMEMory:LOAD** command.

– The **DATA:FILTer** or **MMEMory:LOAD** command must be send prior to sending the this command.

– If you switch from custom filter to either NONE, 50us, or 75us, your previously downloaded information for the custom filter will be lost. You will need to redownload the filter information into the U8903B.

– Refer to “**Example 9: Use the User-Defined Filter Data**” on page 652 for the example on how to use the custom filter.

Examples

The following command sets the de-emphasis condition to 50 μs for channel 1 and channel 2.

```
SENS:FILT:DEEM “50us”,(@1,2)
```

The following query returns the de-emphasis condition for channel 1.

```
SENS:FILT:DEEM? (@1)
```

Typical response:

```
50us
```

SENSe[:ANALog]:FILTer:HPASs

Syntax

```
SENSe[:ANALog]:FILTer:HPASs <high_pass_filter>,(@<channel_list>)
SENSe[:ANALog]:FILTer:HPASs? (@<channel_list>)
```

Description

Sets the high-pass filter for the specified channel(s). The query returns the high-pass filter type of the selected channel(s). Multiple responses are separated by commas.

The high-pass filter types with their corresponding `<high_pass_filter>` parameters are listed as follows.

NONE	No high-pass filter is applied
HP15	High-pass filter with 15 Hz cutoff frequency
HP20	High-pass filter with 20 Hz cutoff frequency
HP22	High-pass filter with 22 Hz cutoff frequency
HP30	High-pass filter with 30 Hz cutoff frequency
HP50	High-pass filter with 50 Hz cutoff frequency
HP70	High-pass filter with 70 Hz cutoff frequency
HP100	High-pass filter with 100 Hz cutoff frequency
HP200	High-pass filter with 200 Hz cutoff frequency
HP300	High-pass filter with 300 Hz cutoff frequency
HP400	High-pass filter with 400 Hz cutoff frequency
CUSTom	User-defined high-pass filter

Parameters

Item	Type	Range of values	Default value
`<high_pass_filter>`	Discrete	NONE|HP15|HP20|HP22|HP30|HP50|HP70|HP100|HP200|HP300|HP400|CUSTom	NONE
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- To use a custom high-pass filter, you need to load the custom filter data into the U8903B before you can use it. To load a custom filter, use either the **DATA:FILTer** or **MMEMory:LOAD** command.
- The **DATA:FILTer** or **MMEMory:LOAD** command must be send prior to sending the this command.
- If you switch from `CUSTom` to `NONE` or any predefined filter such as `HP22`, `HP100`, and so forth, your previously downloaded information for the custom filter will be lost. You will need to redownload the filter information into the U8903B.
- Refer to "**Example 9: Use the User-Defined Filter Data**" on page 652 for the example on how to use custom filter.

Examples

The following command sets the high-pass filter type to HP100 for channel 1 to channel 8.

```
SENS:FILT:HPAS HP100,(@1:8)
```

The following query returns the high-pass filter type for channel 3.

```
SENS:FILT:HPAS? (@3)
```

Typical response:

```
HP100
```

SENSe[:ANALog]:FILTer:LEFT

Syntax

```
SENSe[:ANALog]:FILTer:LEFT <left_filter>
SENSe[:ANALog]:FILTer:LEFT?
```

Description

Sets the HP8903B mode left filter. The query returns the HP8903B mode left filter.

The left filter types with their corresponding <left_filter> parameters are listed as follows.

`None`	No filter is applied
`AWEighting`	A-Weighting filter
`CCIR1k`	CCIR-1k weighted
`CCIR2k`	CCIR-2k weighted
`CCITt`	CCITT
`CMESsage`	C-Message
`HP400`	High-pass filter with 400 Hz cutoff frequency

Parameter

Item	Type	Range of values	Default value
`<left_filter>`	Discrete	NONE|AWEighting|CCIR1k|CCIR2k|CCITt|CMESsage|HP400	CCITt

Remark

This command can only be used when the GPIB is successfully initialized.

Examples

The following command sets the HP8903B left filter type to HP400.

```
SENS:FILT:LEFT HP400
```

The following query returns the HP8903B left filter type.

```
SENS:FILT:LEFT?
```

Typical response:

```
HP400
```

SENSe[:ANALog]:FILTer:LPASs

Syntax

```
SENSe[:ANALog]:FILTer:LPASs <low_pass_filter>,(@<channel_list>)
SENSe[:ANALog]:FILTer:LPASs? (@<channel_list>)
```

Description

Sets the low-pass filter for the specified channel(s). The query returns the low-pass filter type of the selected channel(s). Multiple responses are separated by commas.

The low-pass filter types with their corresponding <low_pass_filter> parameters are listed as follows.

NONE	No low-pass filter is applied
LP2	Low-pass filter with 2 kHz cutoff frequency
LP3	Low-pass filter with 3 kHz cutoff frequency
LP5	Low-pass filter with 5 kHz cutoff frequency
LP8	Low-pass filter with 8 kHz cutoff frequency
LP10	Low-pass filter with 10 kHz cutoff frequency
LP15	Low-pass filter with 15 kHz cutoff frequency
LP20	Low-pass filter with 20 kHz cutoff frequency
LP22	Low-pass filter with 22 kHz cutoff frequency
LP30	Low-pass filter with 30 kHz cutoff frequency
LP40	Low-pass filter with 40 kHz cutoff frequency
LP50	Low-pass filter with 50 kHz cutoff frequency
LP80	Low-pass filter with 80 kHz cutoff frequency
CUSTom	User-defined low-pass filter

Parameters

Item	Type	Range of values	Default value
<low_pass_filter>	Discrete	NONE|LP2|LP3|LP5|LP8|LP10|LP15|LP20|LP22|LP30|LP40|LP50|LP80|CUSTom	NONE
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- To use a custom low-pass filter, you need to load the custom filter data into the U8903B before you can use it. To load a custom filter, use either the **DATA:FILTer** or **MMEMory:LOAD** command.
- The **DATA:FILTer** or **MMEMory:LOAD** command must be send prior to sending the this command.
- If you switch from `CUSTom` to either `NONE` or any predefined filter such as `LP2`, `LP3`, and so forth, your previously downloaded information for the custom filter will be lost. You will need to redownload the filter information into the U8903B.
- Refer to "**Example 9: Use the User-Defined Filter Data**" on page 652 for the example on how to use custom filter.

Examples

The following command sets the low-pass filter type to LP15 for channel 1 to channel 8.

```
SENS:FILT:LPAS LP15,(@1:8)
```

The following query returns the low-pass filter type for channel 3.

```
SENS:FILT:LPAS? (@3)
```

Typical response:

```
LP15
```

SENSe[:ANALog]:FILTer:NOTCh:BANDwidth

Syntax

```
SENSe[:ANALog]:FILTer:NOTCh:BANDwidth
<bandwidth>[<unit>],(@<channel_list>)
```

```
SENSe[:ANALog]:FILTer:NOTCh:BANDwidth? (@<channel_list>)
```

Description

Sets the notch filter bandwidth value for the specified analog analyzer channel(s). The query returns the notch filter bandwidth value. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<bandwidth>	Numeric	78 Hz to 1000 Hz	500 Hz
<unit>	Discrete	Hz	Hz
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- This setting is only applicable for the SINAD, THD+N ratio, and THD+N level measurements.
- You can also include a multiplier for the unit, for example, kHz. The ‘k’ is the multiplier for the unit Hz.

Examples

The following command sets the notch filter bandwidth value to 80 Hz for channel 1 and channel 2 of the analog analyzer.

```
SENS:FILT:NOTC:BAND 80,(@1,2)
```

The following query returns the notch filter bandwidth value for channel 1.

```
SENS:FILT:NOTC:BAND? (@1)
```

Typical response:

```
8.000000E+00
```

SENSe[:ANALog]:FILTer:NOTCh:FREQuency:CENTer

Syntax

```
SENSe[:ANALog]:FILTer:NOTCh:FREQuency:CENTer
<center_frequency>[<unit>],(@<channel_list>)
```

```
SENSe[:ANALog]:FILTer:NOTCh:FREQuency:CENTer? (@<channel_list>)
```

Description

Sets the notch filter center frequency value for the specified analog analyzer channel(s). The query returns the notch filter center frequency value. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<center_frequency>`	Numeric	10 Hz to 90 kHz	1000 Hz
`<unit>`	Discrete	Hz	Hz
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– This setting is only applicable for the SINAD, THD+N ratio, and THD+N level measurements.

– You can also include a multiplier for the unit, for example, kHz. The ‘k’ is the multiplier for the unit Hz.

Examples

The following command sets the notch filter center frequency value to 2 kHz for channel 1 and channel 2 of the analog analyzer.

```
SENS:FILT:NOTC:FREQ:CENT 2kHz,(@1,2)
```

The following query returns the notch filter center frequency value for channel 1.

```
SENS:FILT:NOTC:FREQ:CENT? (@1)
```

Typical response:

```
2.000000E+03
```

SENSe[:ANALog]:FILTer:NOTCh:STATe

Syntax

```
SENSe[:ANALog]:FILTer:NOTCh:STATe <state>,(@<channel_list>)
```

```
SENSe[:ANALog]:FILTer:NOTCh:STATe? (@<channel_list>)
```

Description

Enables or disabled the notch filter of the analog analyzer for the specified channel(s). The query returns the notch filter state. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<state>	Discrete	ENABled|DISabled	DISabled
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is only applicable for the SINAD, THD+N ratio, and THD+N level measurements.

Examples

The following command enables the notch filter for channel 1 and channel 2.

```
SENS:FILT:NOTC:STAT ENAB,(@1,2)
```

The following query returns the notch filter state for channel 1.

```
SENS:FILT:NOTC:STAT? (@1)
```

Typical response:

```
ENAB
```

SENSe[:ANALog]:FILTer:RIGHt

Syntax

```
SENSe[:ANALog]:FILTer:RIGHt <right_filter>
SENSe[:ANALog]:FILTer:RIGHt?
```

Description

Sets the HP8903B mode right filter.

The right filter types with their corresponding <right_filter> parameters are listed as follows.

NONE	No filter is applied
AWEighting	A-Weighting filter
CCIR1k	CCIR-1k weighted
CCIR2k	CCIR-2k weighted
CCITt	CCITT
CMESsage	C-Message
HP400	High-pass filter with 400 Hz cutoff frequency

Parameter

Item	Type	Range of values	Default value
<left_filter>	Discrete	NONE|AWEighting|CCIR1k|CCIR2k|CCITt|CMESsage|HP400	HP400

Remark

This command can only be used when the GPIB is successfully initialized.

Examples

The following command sets the HP8903B right filter type to HP400.

```
SENS:FILT:RIGH HP400
```

The following query returns the HP8903B right filter type.

```
SENS:FILT:RIGH?
```

Typical response:

```
HP400
```

SENSe[:ANALog]:FILTer:WEIGhting

Syntax

```
SENSe[:ANALog]:FILTer:WEIGhting
<weighting_filter>,(@<channel_list>)
```

```
SENSe[:ANALog]:FILTer:WEIGhting? (@<channel_list>)
```

Description

Sets the weighting filter for the specified channel(s). The query returns the weighting filter type of the selected channel(s). Multiple responses are separated by commas.

The weighting filter types with their corresponding <weighting_filter> parameters are listed as follows.

NONE	No filter is applied
AWEighting	A-Weighting filter
CCIR1k	CCIR-1k weighted
CCIR2k	CCIR-2k weighted
CCITt	CCITT
CMESsage	C-Message
CUSTom	User-defined arbitrary filter type including bandpass and bandstop filters

Parameters

Item	Type	Range of values	Default value
<weighting_filter>	Discrete	NONE|AWEighting|CCIR1k|CCIR2k|CCITt|CMESsage|CUSTom	NONE
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- The custom filter type includes the bandpass and bandstop arbitrary filters.
- To use a custom weighting filter, you need to load the custom filter data into the U8903B before you can use it. To load a custom filter, use either the **DATA:FILTer** or **MMEMory:LOAD** command.
- The **DATA:FILTer** or **MMEMory:LOAD** command must be send prior to sending the this command.
- If you switch from `CUSTom` to either `NONE` or any predefined filter such as `AWEighting`, `CCIR1k`, and so forth, your previously downloaded information for the custom filter will be lost. You will need to redownload the filter information into the U8903B.
- Refer to "**Example 9: Use the User-Defined Filter Data**" on page 652 for the example on how to use the custom filter.

Examples

The following command sets the weighting filter type to C-Message for channel 1 to channel 8.

```
SENS:FILT:WEIG CMES,(@1:8)
```

The following query returns the weighting filter type for channel 3.

```
SENS:FILT:WEIG? (@3)
```

Typical response:

```
CMES
```

SENSe[:ANALog]:FUNCtion<j>

Syntax

```
SENSe:FUNCtion<j> <function>,(@<channel_list>)
SENSe:FUNCtion<j>? (@<channel_list>)
```

Description

Sets the analog analyzer measurement function for the specified channel(s). The query returns the measurement function of the selected channel(s). Multiple responses are separated by commas.

The measurement functions with their corresponding `<function>` parameters are listed as follows.

NONE	Disable the measurement
FREQuency	Frequency measurement
VAC	AC voltage measurement
VDC	DC voltage measurement
THDRatio	THD+N ratio measurement
THDLevel	THD+N level measurement
DRATio	THD ratio measurement
DLEVel	THD level measurement
SINad	SINAD measurement
SNRatio	SNR measurement
FSNRatio	SNR (fast) measurement
IMD	SMPTE IMD measuremet
SDFDiec118	DFD IEC 60118 2nd order measurement
TDFDiec118	DFD IEC 60118 3rd order measurement
SDFDiec268	DFD IEC 60268 2nd order measurement
TDFDiec268	DFD IEC 60268 3rd order measurement
JTESt	Jitter Test

Parameters

Item	Type	Range of values	Default value
`<j>`	Numeric	1|2|3|4 – SENS:FUNC1 indicates measurement function 1 – SENS:FUNC2 indicates measurement function 2	1
`<function>`	Discrete	NONE|FREQuency|VAC|VDC|THDRatio|THDLevel|DRATio|DLEVel|SINad|SNRatio|FSNRatio|IMD|JTESt|SDFDiec118|TDFDiec118|SDFDiec268|TDFDiec268	– VAC (function 1) – Frequency (function 2)
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Example

To measure the DC voltage on channel 1, you can set VDC as the measurement function 1. Assume that the DC voltage is measured immediately without waiting for any bus or external trigger. The following command sequence is configured.

```
SENS:FUNC1 VDC,(@1)
TRIG:ANAL:SOUR IMM
INIT:ANAL (@1)
FETC? FUNC1,(@1)
```

NOTE

When FETCh is queried, the measurement result will be returned in the unit as listed in "**Appendix B: Units of the Measurement Function Returned Values**" on page 661.

SENSe[:ANALog]:FUNCtion:MCHannel

Syntax

```
SENSe[:ANALog]:FUNCtion:MCHannel <multi-channel>
SENSe[:ANALog]:FUNCtion:MCHannel? <multi-channel>
```

Description

Sets the multi-channel function mode. This setting applies to all channels as phase and crosstalk measurement involves all analog analyzer channels. The query returns the multi-channel function mode.

Parameter

Item	Type	Range of values	Default value
`<multi-channel>`	Numeric	OFF|PHASe|XTD	OFF

Remarks

– When the multi-channel function mode is set to `PHASe` or `XTD`, Function 1 will automatically switch to VAC, Function 2 to frequency, and Function 3 to phase or crosstalk respectively.

– You can set the crosstalk driven channel and the phase reference channel with the **SENSe[:ANALog]:REFerence:CHANnel** command.

– Refer to "**Example 5: Measure the Crosstalk**" on page 646 for the example on measuring crosstalk.

Examples

The following command sequence measures the analog analyzer phase with channel 1 as the reference channel.

```
SENS:REF:CHAN 1
SENS:FUNC:MCH PHAS
```

The following query returns the multi-channel function mode.

```
SENS:FUNC:MCH?
```

Typical response:

```
PHAS
```

SENSe[:ANALog]:FUNCtion<j>:UNIT

Syntax

```
SENSe:FUNCtion<j>:UNIT <unit>,(@<channel_list>)
SENSe:FUNCtion<j>:UNIT? (@<channel_list>)
```

Description

Specifies the unit for the measurement result (which is obtained using the `FETCh` command) of the corresponding function for the selected channel(s). The query returns the unit of the corresponding function for the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<j>`	Numeric	1|2|3|4 – SENS:FUNC1 indicates measurement function 1 – SENS:FUNC2 indicates measurement function 2	1
`<unit>`	Discrete	Refer to "**Appendix B: Units of the Measurement Function Returned Values**" on page 661.	Required parameter
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

The unit specified using this command will cause the measurement result to be returned in that unit. For example, changing the unit to dBV for the VAC function will return the measurement result obtained by the `FETCh` command in dBV.

Examples

The following command sequence sets the AC voltage as the measurement function 2 in the unit dBV for channel 1 and channel 2.

```
SENS:FUNC2 VAC,(@1,2)
SENS:FUNC2:UNIT dBV,(@1,2)
```

The following query returns the unit of the measurement function 2 for channel 1 and channel 2.

```
SENS:FUNC2:UNIT? (@1,2)
```

Typical response:

```
dBV,dBV
```

SENSe[:ANALog]:FUNDamental:FREQuency

Syntax

```
SENSe[:ANALog]:FUNDamental:FREQuency
<fundamental_frequency>[<unit>],(@<channel_list>)

SENSe[:ANALog]:FUNDamental:FREQuency? (@<channel_list>)
```

Description

Sets the custom fundamental frequency value for the SINAD, THD+N level and THD+N ratio, THD level and THD ratio, SNR (fast mode), phase, and crosstalk for analog analyzer. The query returns the custom fundamental frequency value. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<fundamental_frequency>`	Numeric	– 10 Hz to 90 kHz (normal bandwidth) – 10 Hz to 1.5 MHz (wide bandwidth)	1000 Hz
`<unit>`	Discrete	Hz	Hz
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This command will only take effect if the measurement function of the specified channel(s) is set to SINAD, THD+N level, THD+N ratio, THD level, THD ratio, SNR (fast mode), phase, or crosstalk, and the fundamental frequency lock type is set to `CUSTom`.

Examples

The following command sets the custom fundamental frequency value to 2 kHz for channel 1 of the analog analyzer.

```
SENS:FUND:FREQ 2kHz,(@1)
```

The following query returns the custom fundamental frequency value for channel 1.

```
SENS:FUND:FREQ? (@1)
```

Typical response:

```
2.000000E+03
```

SENSe[:ANALog]:FUNDamental:FREQuency:LOCK

Syntax

```
SENSe[:ANALog]:FUNDamental:FREQuency:LOCK
<type>,(@<channel_list>)
```

```
SENSe[:ANALog]:FUNDamental:FREQuency:LOCK? (@<channel_list>)
```

Description

Sets the fundamental frequency lock type for THD level, THD ratio, THD+N level, THD+N ratio, SINAD, SNR (fast mode), phase, and crosstalk measurements in analog analyzer. The query returns the fundamental frequency lock type. Multiple responses are separated by commas.

AUTO	Automatically determines the fundamental frequency by selecting the signal with the highest power from the incoming signal at analog analyzer
GLOCk	The fundamental frequency is determined by the frequency value set at the corresponding channel of the analog generator
CUSTom	The fundamental frequency value is set by the **SENSe[:ANALog]:FUNDamental:FREQuency** command

Parameters

Item	Type	Range of values	Default value
`<type>`	Discrete	AUTO|GLOCk|CUSTom	AUTO
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- This setting will only take effect if any measurement function of the specified channel has been set to THD level, THD ratio, THD+N level, THD+N ratio, SINAD, SNR (fast mode), phase, or crosstalk.
- When you select `CUSTom`, you will need to define the fundamental frequency using the **SENSe[:ANALog]:FUNDamental:FREQuency** command.

Examples

The following command sets the fundamental frequency lock type to `AUTO` for channel 1.

```
SENS:FUND:FREQ:LOCK AUTO,(@1)
```

The following query returns the SINAD fundamental frequency lock type for channel 1.

```
SENS:FUND:FREQ:LOCK? (@1)
```

Typical response:

```
AUTO
```

SENSe[:ANALog]:IMD:FREQuency:LOCK

Syntax

```
SENSe[:ANALog]:IMD:FREQuency:LOCK <lock type>,(@<channel_list>)
SENSe[:ANALog]:IMD:FREQuency:LOCK? (@<channel_list>)
```

Description

Sets the upper and lower frequencies detection method of the SMPTE IMD measurement for the analog analyzer. The query returns the upper and lower frequencies detection method. Multiple responses are separated by commas.

GLOCk	Lock to the upper and lower frequencies of the generator signal
CUSTom	User-defined upper and lower frequencies

Parameters

Item	Type	Range of values	Default value
<type>	Discrete	GLOCk|CUSTom	GLOCk
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- This setting will only take effect if any measurement function of the specified channel has been set to SMPTE IMD.
- For `GLOCk`, the U8903B assumes that the generating signal is of SMPTE IMD. If you are using other types of signal source, it may cause inaccurate measurement due to the wrong frequencies locking.
- When you select `CUSTom`, you will need to define the upper and lower frequencies using the **SENSe[:ANALog]:IMD:FREQuency:LOWer** and **SENSe[:ANALog]:IMD:FREQuency:UPPer** commands respectively.

Examples

The following command sets the frequency lock type of the SMPTE IMD measurement to `CUSTom` for channel 1 and channel 2.

```
SENS:IMD:FREQ:LOCK CUST,(@1,2)
SENS:IMD:FREQ:UPP 10kHz,(@1,2)
SENS:IMD:FREQ:LOW 100Hz,(@1,2)
```

The following query returns the frequency lock type of the SMPTE IMD measurement for channel 1.

```
SENS:IMD:FREQ:LOCK? (@1)
```

Typical response:

```
CUST
```

SENSe[:ANALog]:IMD:FREQuency:LOWer

Syntax

```
SENSe[:ANALog]:IMD:FREQuency:LOWer
<lower_frequency>[<unit>],(@<channel_list>)
```

```
SENSe[:ANALog]:IMD:FREQuency:LOWer? (@<channel_list>)
```

Description

Sets the lower frequency value of the SMPTE IMD measurement for analog analyzer. The query returns the lower frequency value. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<lower_frequency>`	Numeric	40 Hz to 500 Hz	60 Hz
`<unit>`	Discrete	Hz	Hz
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

This setting will only take effect if any measurement function of the specified channel has been set to SMPTE IMD and the IMD frequency lock type is set to `CUSTom`.

Examples

The following command sets the lower frequency value to 100 Hz for channel 1 and channel 2 of the analog analyzer.

```
SENS:IMD:FREQ:LOW 100Hz,(@1,2)
```

The following query returns the lower frequency value for channel 1.

```
SENS:IMD:FREQ:LOW? (@1)
```

Typical response:

```
1.000000E+02
```

SENSe[:ANALog]:IMD:FREQuency:UPPer

Syntax

```
SENSe[:ANALog]:IMD:FREQuency:UPPer
<upper_frequency>[<unit>],(@<channel_list>)
```

```
SENSe[:ANALog]:IMD:FREQuency:UPPer? (@<channel_list>)
```

Description

Sets the upper frequency value of the SMPTE IMD measurement for analog analyzer. The query returns the upper frequency value. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<upper_frequency>`	Numeric	2 kHz to 60 kHz	7 kHz
`<unit>`	Discrete	Hz	Hz
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting will only take effect if any measurement function of the specified channel has been set to SMPTE IMD and the IMD frequency lock type is set to `CUSTom`.

Examples

The following command sets the upper frequency value to 4 kHz for channel 1 and channel 2 of the analog analyzer.

```
SENS:IMD:FREQ:UPP 4kHz,(@1,2)
```

The following query returns the upper frequency value for channel 1.

```
SENS:IMD:FREQ:UPP? (@1)
```

Typical response:

```
4.000000E+03
```

SENSe[:ANALog]:REFerence:CHANnel

Syntax

```
SENSe[:ANALog]:REFerence:CHANnel <reference_channel>
SENSe[:ANALog]:REFerence:CHANnel?
```

Description

Sets the reference channel for the phase or crosstalk measurement functions. The query returns the reference channel.

Parameter

Item	Type	Range of values	Default value
`<reference_channel>`	Numeric	1|2|3|4|5|6|7|8	1

Examples

The following command sequence measures the crosstalk from channel 2 to channel 1.

```
SENS:REF:CHAN 2
SENS:FUNC:MCHAN XTD
INIT:ANAL (@1)
STAT:OPER:COND?
FETC? FUNC3,(@1)
```

The following query returns the reference channel.

```
SENS:REF:CHAN?
```

Typical response:

```
2
```

SENSe[:ANALog]:REFerence:FREQuency

Syntax

```
SENSe[:ANALog]:REFerence:FREQuency <frequency>,(@<channel_list>)
SENSe[:ANALog]:REFerence:FREQuency? (@<channel_list>)
```

Description

Sets the reference frequency for the specified channel(s) in Hz (Hertz). The reference frequency is used for conversion of the measurement result in unit ΔHz (delta Hertz). The query returns the reference frequency of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<frequency>`	Numeric	$0 \leq$ frequency $\leq 1.0E+9$	0
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

The reference frequency setting is only applicable for the frequency measurement function to specify the measurement results in ΔHz.

Examples

The following command sequence sets the reference frequencies to 100 Hz and 300 Hz for channel 1 and 2 respectively.

```
SENS:REF:FREQ 100,(@1)
SENS:REF:FREQ 300,(@2)
```

The following query returns the reference frequency for channel 1 and channel 2.

```
SENS:REF:FREQ? (@1,2)
```

Typical response:

```
1.000000E+02,3.000000E+02
```

SENSe[:ANALog]:REFerence:

Syntax

```
SENSe[:ANALog]:REFerence:IMPedance <impedance>,(@<channel_list>)
SENSe[:ANALog]:REFerence:IMPedance? (@<channel_list>)
```

Description

Sets the reference impedance for the specified channel(s) in ohms (Ω). The reference impedance is used for conversion of the measurement result in unit W or dBm. The query returns the reference impedance of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<impedance>	Numeric	1.0E-12 ≤ impedance ≤ 1.0E+9	600 Ω
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

The reference impedance setting is only applicable for the VAC and THD+N level measurement functions to specify the measurement results in W or dBm.

Examples

The following command sequence sets the reference impedances to 600 Ω and 50 Ω for channel 1 and channel 2 respectively.

```
SENS:REF:IMP 600,(@1)
SENS:REF:IMP 50,(@2)
```

The following query returns the reference impedances for channel 1 and channel 2.

```
SENS:REF:IMP? (@1,2)
```

Typical response:

```
6.000000E+02,5.000000E+01
```

SENSe[:ANALog]:REFerence:LEVel

Syntax

```
SENSe[:ANALog]:REFerence:LEVel <level>,(@<channel_list>)
SENSe[:ANALog]:REFerence:LEVel? (@<channel_list>)
```

Description

Sets the reference level for the specified channel(s) in V. The reference level is used for conversion of the measurement result in unit dBr, dBSPL, ΔV, or x. The query returns the reference level of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<level>`	Numeric	10^{-6} V < level < 140 V	387.3 mV
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

The reference level setting is only applicable for the VAC, VDC, and THD+N level measurement functions to specify the measurement results in dBr, dBSPL, ΔV, or x.

Examples

The following command sequence sets the reference levels to 200 mV and 500 mV for channel 1 and channel 2 respectively.

```
SENS:REF:LEV 0.2,(@1)
SENS:REF:LEV 0.5,(@2)
```

The following query returns the reference levels for channel 1 and channel 2.

```
SENS:REF:LEV? (@1,2)
```

Typical response:

```
2.000000E-01,5.000000E-01
```

SENSe[:ANALog]:REFerence:RATio

Syntax

```
SENSe[:ANALog]:REFerence:RATio <ratio>,(@<channel_list>)
SENSe[:ANALog]:REFerence:RATio? (@<channel_list>)
```

Description

Sets the reference ratio for the specified channel(s) in dB. The reference ratio is used for conversion of the measurement result in unit ΔdB (delta decibel). The query returns the reference ratio of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<ratio>`	Numeric	–180 ≤ level ≤ 180	0
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

The reference ratio setting is only applicable for the SINAD, THD ratio, DFD, SMPTE IMD, crosstalk, and SNR measurement functions to specify the measurement results in ΔdB.

Examples

The following command sequence sets the reference ratio to 10 dB and 30 dB for channel 1 and channel 2 respectively.

```
SENS:REF:RAT 10,(@1)
SENS:REF:RAT 30,(@2)
```

The following query returns the reference ratio for channel 1 and channel 2.

```
SENS:REF:RAT? (@1,2)
```

Typical response:

```
1.000000E+01,3.000000E+01
```

SENSe[:ANALog]:REFerence:RESult:SET

Syntax

```
SENSe[:ANALog]:REFerence:RESult:SET
<function_number>,<source_channel>,(@<target_channel_list>)
```

Description

Sets the last measurement result obtained from the specified function of the selected source as the reference value for the corresponding target analog analyzer channels. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<function_number>	Discrete	Indicates which source channel's function for the measurement reading to be recorded as the reference value of specified target channel(s). FUNCtion1|FUNCtion2|FUNCtion3|FUNCtion4	FUNCtion1
<source_channel>	Discrete	CH1|CH2|CH3|CH4|CH5|CH6|CH7|CH8	Required parameter
<target_channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- If there is no data in the last measurement or invalid data such as not a number (NaN), this command will not have any effect; the previous reference value will be remain.
- If the selected function (based on the function number) is measuring frequency, sending this command will set the reference frequency to the measured frequency reading.
- If the selected function (based on the function number) is measuring VAC, VDC, THD level, THD+N level, or any level-based measurement type, sending this command will set the reference level to the measured reading.
- If the selected function (based on the function number) is measuring THD+N ratio, SINAD, or any ratio-based measurement type, sending this command will set the reference ratio to the measured reading.

– After sending this command, you can always check the recorded reference values by sending either the **SENSe[:ANALog]:REFerence:LEVel**, **SENSe[:ANALog]:REFerence:FREQuency**, or **SENSe[:ANALog]:REFerence:RATio** query to query for the recorded values, depending on the selected measurement function type.

Example

The following command sets the function 1 VAC result of the channel 1 as the reference value to channel 1 and channel 2.

```
SENS:REF:RES:SET FUNC1,CH1,(@1,2)
```

SENSe[:ANALog]:SAMPle:SIZE

Syntax

```
SENSe[:ANALog]:SAMPle:SIZE <sample_size>,(@<channel_list>)
SENSe[:ANALog]:SAMPle:SIZE? (@<channel_list>)
```

Description

Sets the analog analyzer acquisition sample size for each measurement. The query returns the acquisition sample size. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<sample_size>	Numeric	2048|4096|8192|16384|32768|65536|131072|262144|524288|1048576|2097152	32768
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

The sample size for the channels in the same analyzer card must be the same.

Examples

The following command sets the acquisition sample size to 4096 for channel 1 and channel 2 (analog analyzer card 1).

```
SENS:SAMP:SIZE 4096,(@1,2)
```

The following query returns the acquisition sample size for channel 1.

```
SENS:SAMP:SIZE? (@1)
```

Typical response:

```
4096
```

SENSe[:ANALog]:SNR:DELay

Syntax

```
SENSe[:ANALog]:SNR:DELay <delay>,(@<channel_list>)
SENSe[:ANALog]:SNR:DELay? (@<channel_list>)
```

Description

Sets the SNR (signal-to-noise ratio) measurement delay for the specified channel(s) in milliseconds. The measurement delay is the time delay between the changing of the generator states and the start of making the measurement. The query returns the SNR measurement delay of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<delay>`	Numeric	0 to 2000 ms	0
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is applicable for all analog analyzer cards.

Examples

The following command sets the SNR measurement delay to 200 ms for channel 1.

```
SENS:SNR:DEL 200,(@1)
```

The following query returns the SNR delay for channel 1.

```
SENS:SNR:DEL? (@1)
```

Typical response:

```
200
```

SENSe[:ANALog]:SNR:HARMonic:COUNt

Syntax

```
SENSe[:ANALog]:SNR:HARMonic:COUNt
<number_of_harmonics>,(@<channel_list>)
```

```
SENSe[:ANALog]:SNR:HARMonic:COUNt? (@<channel_list>)
```

Description

Sets the number of harmonic orders to be removed in the calculation used for the SNR (fast mode) measurement for the specified channel(s). The query returns the SNR (fast mode) measurement harmonic count of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<number_of_harmonics>	Numeric	2 to 10	5
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is applicable if the measurement function of the specified channel has been set to SNR (fast mode).

Examples

The following command sets the number of harmonics to 6 for channel 1 and channel 2.

```
SENS:SNR:HARM:COUN 6,(@1,2)
```

The following query returns the number of harmonics.

```
SENS:SNR:HARM:COUN? (@1)
```

Typical response:

```
6
```

SENSe[:ANALog]:SOURce:CHANnel

Syntax

```
SENSe[:ANALog]:SOURce:CHANnel <source_channel>,(@channel_list)
SENSe[:ANALog]:SOURce:CHANnel? (@channel_list)
```

Description

Sets the source channel of the SNR measurement for the analog analyzer specified channel(s). Source channel is the generator output channel used for the SNR measurement. This setting is also used in the calculation for the result returned in unit dBg. The source channel setting is also used to determine which generator channel's frequency to lock when you choose `GLOCK` as the frequency lock type. The query returns the source channel of the analog analyzer selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<source_channel>`	Numeric	1|2	1
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is only applicable when the function 2 measurement of the specified analog analyzer channel is set to SNR.

Examples

The following command sets the source channel of the SNR measurement to channel 2 for analog analyzer channel 1.

```
SENS:SOUR:CHAN 2,(@1)
```

The following query returns the source channel of the SNR measurement for analog analyzer channel 1.

```
SENS:SOUR:CHAN? (@1)
```

Typical response:

```
2
```

SENSe[:ANALog]:THD:HARMonic:COMPonent

Syntax

```
SENSe[:ANALog]:THD:HARMonic:COMPonent
(<harmonics_components>),(@<channel_list>)
```

```
SENSe[:ANALog]:THD:HARMonic:COMPonent? (@<channel_list>)
```

Description

Sets the THD measurement harmonics components for the analog analyzer specified channel(s). The query returns the THD measurement harmonics components of the analog analyzer selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<harmonics_components>`	Numeric	One or more harmonics can be selected. 1 to 9 – (2) for 2nd harmonic – (2,5) for 2nd and 5th harmonics – (2:6) for 2nd to 6th harmonics	(2:9)
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Examples

The following command sets the THD measurement harmonic components 2 and 3 for channel 1 and channel 2.

```
SENSe:THD:HARM:COMP (2,3),(@1,2)
```

The following query returns the THD measurement harmonic components for channel 1 and channel 2.

```
SENSe:THD:HARM:COMP? (@1,2)
```

Typical response:

```
2,3
```

SENSe[:ANALog]:VOLTage:DETector

Syntax

```
SENSe[:ANALog]:VOLTage:DETector <detector_type>,@<channel_list>)
SENSe[:ANALog]:VOLTage:DETector? (@<channel_list>)
```

Description

Sets the analog analyzer AC level voltage detector for the specified channel(s). The query returns the voltage detector type of the selected channel(s). Multiple responses are separated by commas.

`RMS`	RMS
`VPP`	Peak to peak

Parameters

Item	Type	Range of values	Default value
`<detector_type>`	Discrete	RMS|VPP	RMS
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- This command is only applicable if the selected measurement function is VAC.
- If `VPP` is selected, sending the query will return the AC voltage result in Vpp. If `RMS` is selected, the returned AC voltage result is in Vrms.

Examples

The following command sequence sets the voltage detector types to RMS and Vpp for channel 1 and channel 2 respectively.

```
SENS:VOLT:DET RMS,(@1)
SENS:VOLT:DET VPP,(@2)
```

The following query returns the voltage detector types for channel 1 and channel 2.

```
SENS:VOLT:DET? (@1,2)
```

Typical response:

```
RMS,VPP
```

SENSe[:ANALog]:VOLTage:RANGe:AUTO

Syntax

```
SENSe[:ANALog]:VOLTage:RANGe:AUTO <state>,(@<channel_list>)
SENSe[:ANALog]:VOLTage:RANGe:AUTO? (@<channel_list>)
```

Description

Disables or enables the voltage measurements autoranging for the specified channel(s). Autoranging allows the U8903B to automatically select the range for each measurement based on the input signal detected. The query returns the autoranging state of the selected channel(s) as `0` if the state is `OFF`, or `1` if the state is `ON`. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	OFF
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– Selecting a discrete range using the **SENSe[:ANALog]:VOLTage:RANGe[:UPPer]** command will disable the autoranging.

– Autoranging is enabled after a factory reset (***RST**) command or instrument preset (**SYSTem:PRESet**) command.

Examples

The following command sequence disables the autoranging for channel 1 but enables the autoranging for channel 2.

```
SENS:VOLT:RANG:AUTO OFF,(@1)
SENS:VOLT:RANG:AUTO ON,(@2)
```

The following query returns the autoranging states for channel 1 and channel 2.

```
SENS:VOLT:RANG:AUTO? (@1,2)
```

Typical response:

```
0,1
```

SENSe[:ANALog]:VOLTage:RANGe[:UPPer]

Syntax

```
SENSe[:ANALog]:VOLTage:RANGe[:UPPer]
<range>[<unit>],(@<channel_list>)
```

```
SENSe[:ANALog]:VOLTage:RANGe[:UPPer]? (@<channel_list>)
```

Description

Sets the input peak voltage range for the specified channel(s) in V. The query returns the input peak voltage range of the selected channel(s) in V. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<range>	Numeric	0.32|1|3.2|10|32|100|140 (unbalanced)| 300 (balanced)	1
<unit>	Discrete	V	V
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- Selecting a discrete range using this command will disable the autoranging (**SENSe[:ANALog]:VOLTage:RANGe:AUTO**) on the specified channel(s).
- If the input signal is greater than the selected measurement range, an overload indication of 9.9E+37 will be generated.
- You can also include a multiplier for the unit, for example, mV. The ‘m’ is the multiplier for the unit V.
- If you set the voltage value in between the desired value range as listed below, it will clip the value to the upper range. For example, 1.5 V will clip the voltage to 3.2 V.
- The voltage ranges for firmware 2.10.x.x and below will be mapped to the new voltage ranges (firmware 3.0.0.0 and above) as listed in “**Appendix N: Migrating from U8903A to U8903B**” on page 695.

Examples

The following commands set the measurement voltage range values to 320 mV for channel 1 and channel 2.

```
SENS:VOLT:RANG 320mV,(@1,2)
```

The following query returns the measurement voltage range values for channel 1.

```
SENS:VOLT:RANG? (@1)
```

Typical response:

```
3.200000E+00
```

SENSe[:ANALog]:WAVFile:BPS

Syntax

```
SENSe[:ANALog]:WAVFile:BPS <bits_per_sample>,(@<channel_list>)
SENSe[:ANALog]:WAVFile:BPS? (@<channel_list>)
```

Description

Sets the bits per sample for the wave file to be saved. This setting is applicable for all analog analyzer channels. The query returns the wave file bits per sample.

Parameters

Item	Type	Range of values	Default value
`<bits_per_sample>`	Numeric	8|16|24	8
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- Each analyzer card has two channels. This setting must be the same for both channels in the the same analyzer card. For example, if you only set channel 1 to 8 bits, channel 2 will also be set to 8 bits automatically.
- Refer to "**Example 11: Record Input Signal to Wave File**" on page 656 for the example on how to record to a wave file.

Examples

The following command sets the wave file bits per sample to 8 bits for analyzer card 1.

```
SENS:WAVF:BPS 8,(@1,2)
```

The following query returns the wave file bits per sample for channel 1.

```
SENS:WAVF:BPS? (@1)
```

Typical response:

```
8
```

SENSe[:ANALog]:WAVFile:CHANnel

Syntax

```
SENSe[:ANALog]:WAVFile:CHANnel
<channel_to_save>,(@<channel_list>)
```

```
SENSe[:ANALog]:WAVFile:CHANnel? (@<channel_list>)
```

Description

Sets the channel(s) to be saved in the wave file for the specified channel(s). This settings is applicable for all analog analyzer channels. The query returns the channel(s) to be saved in the wave file of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<channel_to_save>`	Discrete	LEFT|RIGHt|STEReo	LEFT
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- Each analyzer card has two channels. This setting must be the same for both channels in the the same analyzer card.
- `LEFT` indicates the first channel of the selected analyzer card and `RIGHt` indicates the second channel of the selected analyzer card and `STEReo` indicates both channels of the selected analyzer card. For example, the left channel for analyzer card number 1 means channel 1 of the analyzer card and the right channel means the channel 2 of the analyzer card, and stereo means channel 1 and 2 of the analyzer card.
- For example, at analyzer channel 3 (channel 3 is the first channel and channel 4 is the second channel of the analyzer card 2), you select channel `RIGHt` as the recording channel. This means that the signal from analyzer channel 4 will be saved in the wave file.
- Refer to “**Example 11: Record Input Signal to Wave File**” on page 656 for the example on how to record to a wave file.

Examples

The following command sets channel 1 to be saved in the wave file for analyzer card 1.

```
SENS:WAVF:CHAN LEFT,(@1,2)
```

The following query returns the channel(s) to be saved in the wave file for analyzer card 1.

```
SENS:WAVF:CHAN? (@1)
```

Typical response:

```
LEFT
```

SENSe[:ANALog]:WAVFile:DURation

Syntax

```
SENSe[:ANALog]:WAVFile:DURation <duration>,(@<channel_list>)
SENSe[:ANALog]:WAVFile:DURation? (@<channel_list>)
```

Description

Sets the duration for the wave file to be saved. This setting is applicable for all analog analyzer channels. The query returns the wave file duration.

Parameters

Item	Type	Range of values	Default value
`<duration>`	Numeric	1 s to 600 s	1 s
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– Each analyzer card has two channels. This setting must be the same for both channels in the the same analyzer card.

– Refer to “**Example 11: Record Input Signal to Wave File**” on page 656 for the example on how to record to a wave file.

Examples

The following command sets the wave file duration to 10 s for analyzer card 1 (channel 1 and channel 2).

```
SENS:WAVF:DUR 10,(@1,2)
```

The following query returns the wave file duration for channel 1.

```
SENS:WAVF:DUR? (@1)
```

Typical response:

```
10
```

SENSe:DIGital:COUPling

Syntax

```
SENSe:DIGital:COUPling <coupling>,(@<channel_list>)
SENSe:DIGital:COUPling? (@<channel_list>)
```

Description

Sets the coupling mode of the embedded digital analyzer audio signal for the selected channel(s). The query returns the coupling mode for the selected channel(s). Multiple responses are separated by commas.

AC	AC coupling blocks the DC component of the audio signal.
DC	DC coupling allows both AC and DC input signals to pass through to the digital analyzer and to be measured down to 0 Hz. This setting should be selected when making DC voltage measurements.

Parameters

Item	Type	Range of values	Default value
<coupling>	Discrete	AC or DC	DC
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Examples

The following command sets the coupling mode for channel 1 to AC.

```
SENS:DIG:COUP AC, (@D1)
```

The following query returns the coupling mode for channel 1.

```
SENS:DIG:COUP? (@D1)
```

Typical response:

```
AC
```

SENSe:DIGital:SAMPle:SIZE

Syntax

```
SENSe:DIGital:SAMPle:SIZE <sample_size>
```

```
SENSe:DIGital:SAMPle:SIZE?
```

Description

Sets the acquisition data size of the digital analyzer audio signal to be analyzed. The query returns the sample size.

Parameter

Item	Type	Range of values	Default value
`<sample_size>`	Numeric	2048|4096|8192|16384|32768|65536|131072|262144|524288|1048576|2097152	32768

Examples

The following command sets the sample size to 4096.

```
SENS:DIG:SAMP:SIZE 4096
```

The following query returns the sample size.

```
SENS:DIG:SAMP:SIZE?
```

Typical response:

```
4096
```

SENSe:DIGital:VOLTage:DETector

Syntax

```
SENSe:DIGital:VOLTage:DETector <detector_type>,(@<channel_list>)
SENSe:DIGital:VOLTage:DETector?(@<channel_list>)
```

Description

Sets the digital analyzer AC level detector for the specified channel(s). The query returns the detector type of the selected channel(s). Multiple responses are separated by commas.

RMS	RMS detector
VPP	Peak-to-peak detector

Parameter

Item	Type	Range of values	Default value
<detector_type>	Discrete	RMS|VPP	RMS
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remark

This command is only applicable if the selected measurement function is VAC.

Examples

The following commands set the detector types to RMS and Vpp for channel 1 and 2 respectively.

```
SENS:DIG:VOLT:DET RMS, (@D1)
SENS:DIG:VOLT:DET VPP, (@D2)
```

The following query returns the detector types for channel 1 and 2.

```
SENS:DIG:VOLT:DET? (@D1,D2)
```

Typical response:

```
RMS,VPP
```

SENSe:DIGital:FILTer:CLEar

Syntax

```
SENSe:DIGital:FILTer:CLEar    (@channel_list)
```

Description

Disables all filters including low pass, high pass, weighting, and de-emphasis for the selected channels.

Parameter

Item	Type	Range of values	Default value
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Examples

The following command disables all filters for channel 1 and 2.

```
SENS:DIG:FILT:CLE (@D1,@D2)
```

SENSe:DIGital:FILTer:LPASs

Syntax

```
SENSe:DIGital:FILTer:LPASs <low_pass_filter>,(@<channel_list>)
SENSe:DIGital:FILTer:LPASs? (@<channel_list>)
```

Description

Sets the low-pass filter for the specified channel(s). The query returns the low-pass filter type of the selected channel(s). Multiple responses are separated by commas.

The low-pass filter types with their corresponding <low_pass_filter> parameters are listed as follows.

NONE	No low-pass filter is applied
LP15	Low-pass filter with 15 kHz cutoff frequency
LP20	Low-pass filter with 20kHz cutoff frequency
LP22	Low-pass filter with 22 kHz cutoff frequency
LP30	Low-pass filter with 30 kHz cutoff frequency
CUSTom	User-defined low-pass filter

Parameters

Item	Type	Range of values	Default value
<low_pass_filter>	Discrete	NONE|LP15|LP20|LP22|LP30|CUSTom	NONE
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

– To use a custom low-pass filter, you need to load the custom filter data into the U8903B before you can use it. To load a custom filter data, use the **DATA:DIGital:FILTer** command.

– The **DATA:DIGital:FILTer** command must be sent prior to sending the **SENSe:DIGital:FILTer:LPASs** command.

– If you switch from CUSTom to either NONE, LP15, LP20, LP22, or LP30, your previously downloaded information for the custom filter will be lost. You will need to redownload the filter information into the system. Refer to "**Use the User-Defined Filter Data**" on page 652 for the information on how to use the custom filter.

Examples

The following commands set the low-pass filter types to LP15 and LP30 for channel 1 and 2 respectively.

```
SENS:DIG:FILT:LPAS LP15, (@D1)
```

```
SENS:DIG:FILT:LPAS LP30, (@D2)
```

The following query returns the low-pass filter types for channel 1 and 2.

```
SENS:DIG:FILT:LPAS? (@D1,D2)
```

Typical response:

```
LP15,LP30
```

SENSe:DIGital:FILTer:HPASs

Syntax

```
SENSe:DIGital:FILTer:HPASs <high_pass_filter>,(@<channel_list>)
SENSe:DIGital:FILTer:HPASs? (@<channel_list>)
```

Description

Sets the high-pass filter for the specified channel(s). The query returns the high-pass filter type of the selected channel(s). Multiple responses are separated by commas.

The high-pass filter types with their corresponding `<high_pass_filter>` parameters are listed as follows.

NONE	No high-pass filter is applied
HP20	High-pass filter with 20 Hz cutoff frequency
HP100	High-pass filter with 100 Hz cutoff frequency
HP400	High-pass filter with 400 Hz cutoff frequency
CUSTom	User- defined high-pass filter

Parameters

Item	Type	Range of values	Default value
<high_pass_filter>	Discrete	NONE|HP20|HP100|HP400|CUSTom	NONE
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- To use a custom high-pass filter, you need to load the custom filter data into the U8903B before you can use it. To load a custom filter data, use either the **DATA:DIGital:FILTer** or **MMEMory:LOAD** command.
- The command **DATA:DIGital:FILTer** or **MMEMory:LOAD** must be sent prior to sending this command.
- If you switch from CUSTom to either NONE, HP20, HP100, or HP400, your previously downloaded information for the custom filter will be lost. You will need to redownload

the filter information into the system. Refer to "**Use the User-Defined Filter Data**" on page 652 for the information on how to use the custom filter.

Examples

The following commands set the high-pass filter types to HP20 and HP100 for channel 1 and 2 respectively.

```
SENS:DIG:FILT:HPAS HP20, (@D1)
```

```
SENS:DIG:FILT:HPAS HP100, (@D2)
```

The following query returns the high-pass filter types for channel 1 and 2.

```
SENS:DIG:FILT:HPAS? (@D1,D2)
```

Typical response:

```
HP20,HP100
```

SENSe:DIGital:FILTer:WEIGhting

Syntax

```
SENSe:DIGital:FILTer:WEIGhting <weighting_filter>,
(@<channel_list>)
```

```
SENSe:DIGital:FILTer:WEIGhting?(@<channel_list>)
```

Description

Sets the weighting filter for the specified channel(s). The query returns the weighting filter type of the selected channel(s). Multiple responses are separated by commas.

The weighting filter types with their corresponding `<weighting filter>` parameters are listed as follows.

NONE	No weighting filter is applied
AWEighting	A-Weighting filter
CCIR1k	CCIR-1k weighted
CCIR2k	CCIR-2k weighted
CMESsage	C-Message
CCITt	CCITT
CUSTom	User- defined arbitrary filter type including Bandpass, Bandstop and notch filters

Parameters

Item	Type	Range of values	Default value
<weighting_filter>	Discrete	NONE|AWEighting|CCIR1k|CCIR2k|CMESsage|CCITt|CUSTom	NONE
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- The custom filter type includes the bandpass, bandstop, notch, and arbitrary filters.
- To use a custom weighting filter, you need to load the custom filter data into the U8903B before you can use it. To load a custom filter data, use either the **DATA:DIGital:FILTer** or **MMEMory:LOAD** command.

– The command **DATA:DIGital:FILTer** or **MMEMory:LOAD** must be sent prior to sending this command.

– If you switch from CUSTom to either NONE, AWE, CCIR1k, CCIR2k, CMES, or CCIT, your previously downloaded information for the custom filter will be lost. You will need to redownload the filter information into the system. Refer to "**Use the User-Defined Filter Data**" on page 652 for the information on how to use the custom filter.

Examples

The following commands set the weighting filter types to A-Weighting and C- Message for channel 1 and 2 respectively.

```
SENS:DIG:FILT:WEIG AWE, (@D1)
```

```
SENS:DIG:FILT:WEIG CMES, (@D2)
```

The following query returns the weighting filter types for channel 1 and 2.

```
SENS:DIG:FILT:WEIG? (@D1,D2)
```

Typical response:

```
AWE,CMES
```

SENSe:DIGital:FILTer:DEEMphasis

Syntax

```
SENSe:DIGital:FILTer:DEEMphasis <de-emphasis>,(@<channel_list>)
SENSe:DIGital:FILTer:DEEMphasis?(@<channel_list>)
```

Description

Sets the de-emphasis condition for the specified channel(s). The query returns the de-emphasis condition of the selected channel(s). Multiple responses are separated by commas.

The de-emphasis conditions with their corresponding `<de-emphasis>` parameters are listed as follows.

NONE	No de-emphasis
50us	50 µs de-emphasis
75us	75 µs de-emphasis
CUSTom	User-defined de-emphasis filter

Parameters

Item	Type	Range of values	Default value
<de-emphasis>	String	NONE|50us|75us|CUSTom	NONE
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- To use a custom deemphasis filter, you need to load the custom filter data into the U8903B before you can use it. To load a custom filter data, use either the **DATA:DIGital:FILTer** or **MMEMory:LOAD** command.
- The command **DATA:DIGital:FILTer** or **MMEMory:LOAD** must be sent prior to sending this command.
- If you switch from `CUSTom` to either NONE, 50us, or 75us, your previously downloaded information for the custom filter will be lost. You will need to redownload the filter

information into the system. Refer to “**Use the User-Defined Filter Data**” on page 652 for the information on how to use the custom filter.

Examples

The following commands set the de- emphasis condition to 50 μs for channel 1.

```
SENS:DIG:FILT:DEEM “50us”, (@D1)
```

The following query returns the de- emphasis condition for channel 1.

```
SENS:DIG:FILT:DEEM? (@D1)
```

Typical response:

```
50us
```

SENSe:DIGital:FILTer:SRATe

Syntax

```
SENSe:DIGital:FILTer:SRATe <sampling_rate>,(@<channel_list>)
SENSe:DIGital:FILTer:SRATe? (@<channel_list>)
```

Description

Sets the filter sampling rate for the specified channel(s). The query returns the filter sampling rate of the selected channel(s). Multiple responses are separated by commas.

S32000	32 kHz
S44100	44.1 kHz
S48000	48 kHz
S88200	88.2 kHz
S96000	96 kHz
S17640	176.4 kHz
S19200	192 kHz

Parameters

Item	Type	Range of values	Default value
<sampling rate>	Discrete	S32000|S44100|S48000|S88200|S96000|S17640|S19200	S48000
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Examples

The following command sets the sampling rate for channel 1.

```
SENS:DIG:FILT:SRAT S32000, (@D1)
```

The following query returns the sampling rate for channel 1.

```
SENS:DIG:FILT:SRAT? (@D1)
```

Typical response:

```
S32000
```

SENSe:DIGital:FUNCtion:MCHannel

Syntax

```
SENSe:DIGital:FUNCtion:MCHannel <multi-channel>
SENSe:DIGital:FUNCtion:MCHannel?
```

Description

Sets the multi-channel function mode to phase, crosstalk, or disable. This setting is applied to all channels of the digital analyzer.

OFF	Disables the multi-channel function mode
PHASe	Phase measurement
XTD	Crosstalk measurement driven by one channel

Parameter

Item	Type	Range of values	Default value
<multi-channel>	Numeric	OFF|PHASe|XTD	OFF

Remarks

- When multi-channel function mode is set to phase or XTD, function 1 will switch to VAC, function 2 to frequency, and Function 3 to phase or crosstalk.
- Use the **SENSe:DIGital:REFerence:CHANnel** command to set the crosstalk driven channel and the phase reference channel.

Examples

The following command measures the phase where the reference channel is channel 1.

```
SENS:DIG:REF:CHANNEL 1
```

```
SENS:DIG:FUNCtion:MCHANNEL  PHASe
```

The following command measures the crosstalk where the driven channel is channel 1.

```
SENS:DIG:REF:CHANNEL 1
```

```
SENS:DIG:FUNCtion:MCHANNEL  XTD
```

The following command queries for the multi-channel function mode where crosstalk is selected.

```
SENS:DIG:FUNCtion:MCHannel?
```

Typical Response:

```
XTD
```

SENSe:DIGital:FUNCtion<j>

Syntax

```
SENSe:DIGital:FUNCtion<j> <function>,(@<channel_list>)
SENSe:DIGital:FUNCtion<j>? (@<channel_list>)
```

Description

Sets the digital analyzer measurement function for the specified channel(s). The query returns the measurement function of the selected channel(s). Multiple responses are separated by commas.

The measurement functions with their corresponding `<function>` parameters are listed as follows.

FREQuency	Frequency measurement
VAC	AC voltage measurement
VDC	DC voltage measurement
THDRatio	THD+N ratio measurement
THDLevel	THD+N level measurement
SINad	SINAD measurement
IMD	SMPTE IMD measuremet
SDFDiec118	DFD IEC 60118 2nd order measurement
TDFDiec118	DFD IEC 60118 3rd order measurement
SDFDiec268	DFD IEC 60268 2nd order measurement
TDFDiec268	DFD IEC 60268 3rd order measurement
PPEak	Maximum peak measurement
NPEak	Minimum peak measurement
DRATio	THD ratio measurement
DLEVel	THD level measurement

Parameters

Item	Type	Range of values	Default value
<j>	Numeric	1 to 2 – SENS:DIG:FUNC1 indicates measurement function 1 – SENS:DIG:FUNC2 indicates measurement function 2	1
<function>	Discrete	FREQuency|VAC|VDC|THDRatio|THDLevel|SINad|IMD|SDFDiec118|TDFDiec118|SDFDiec268|TDFDiec268|PPEak|NPEak|DRATio|DLEVel	– Frequency (function 1) – VAC (function 2)
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- For the first measurement function, there are only three types of selectable functions comprising frequency, VAC, and VDC. For the second measurement function, you can select any of the measurement functions listed above.
- You need to set the DC coupling (**SENSe:DIGital:COUPling** command) prior to setting the VDC measurement function.
- If either phase or crosstalk is selected, you must also specify the reference channel using the **SENSe:DIGital:REFerence:CHANnel** command. You must set the reference channel prior to sending the **SENSe:DIGital:FUNCtion<j>** command.

Examples

The following commands set Frequency as the first measurement function and SINAD as the second measurement function for channel 1. Make the measurement and fetch the SINAD measurement result.

```
SENS:DIG:FUNC1 FREQ, (@D1)
SENS:DIG:FUNC2 SIN, (@D1)
INIT:DIG:ANAL (@D1)
FETC:DIG:SCAL? FUNC2, (@D1)
```

NOTE

- When FETCh is queried, the measurement result will be returned in the unit as listed in "**Appendix B: Units of the Measurement Function Returned Values**" on page 661.
- For crosstalk measurements, a value of 0 dB or 100% will always be returned when FETCh is used to acquire the result of the reference channel.

The following query returns the measurement function of channel 1.

```
SENS:DIG:FUNC2? (@D1)
```

Typical response:

```
SINAD
```

SENSe:DIGital:FUNCtion<j>:UNIT

Syntax

```
SENSe:DIGital:FUNCtion<j>:UNIT <unit>,(@<channel_list>)
SENSe:DIGital:FUNCtion<j>:UNIT?(@<channel_list>)
```

Description

Specifies the unit for the measurement result (which is obtained using the `FETCh:DIGital` command) of the corresponding function for the selected channel(s). The query returns the unit of the corresponding function for the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<j>`	Numeric	1 to 2 – SENS:DIG:FUNC1 indicates measurement function 1 – SENS:DIG:FUNC2 indicates measurement function 2	1
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

For the `<unit>` range of values and formulas, refer to "**Appendix B: Units of the Measurement Function Returned Values**" on page 661

Remark

The unit specified using this command will cause the measurement result to be returned in that unit. For example, changing the unit to dBV for the Vac function will return the measurement result obtained by the `FETCh:DIGital` command in dBV.

Examples

The following commands set the AC voltage as the second measurement function in the unit dBV for both channels.

```
SENS:DIG:FUNC2 VAC, (@D1,D2)
```

```
SENS:DIG:FUNC2:UNIT dBV, (@D1,D2)
```

The following query returns the unit of the second measurement function for both channels.

```
SENS:DIG:FUNC2:UNIT? (@D1,D2)
```

Typical response:

```
dBV,dBV
```

SENSe:DIGital:REFerence:LEVel

Syntax

```
SENSe:DIGital:REFerence:LEVel <level>,(@<channel_list>)
SENSe:DIGital:REFerence:LEVel? (@<channel_list>)
```

Description

Sets the reference level for the specified channel(s) in FFS. The reference level is used for conversion of the measurement result in unit dBr or x. The query returns the reference level of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<level>`	Numeric	0 < level < 1 FFS	0.1 FFS
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Examples

The following commands set the reference levels to 200 mFFS and 500 mFFS for channel 1 and 2 respectively.

```
SENS:DIG:REF:LEV 0.2, (@D1)
SENS:DIG:REF:LEV 0.5, (@D2)
```

The following query returns the reference levels for channel 1 and 2.

```
SENS:DIG:REF:LEV? (@D1,D2)
```

Typical response:

```
2.000000E-01,5.000000E-01
```

SENSe:DIGital:REFerence:FREQuency

Syntax

```
SENSe:DIGital:REFerence:FREQuency <frequency>,(@<channel_list>)
```

```
SENSe:DIGital:REFerence:FREQuency?(@<channel_list>)
```

Description

Sets the reference frequency for the specified channel(s) in Hz (Hertz). The reference frequency is used for conversion of the measurement result in unit ΔHz (delta Hertz). The query returns the reference frequency of the selected channel(s). Multiple responses are separated by commas.

Parameters

<table>
<tr><th>Item</th><th>Type</th><th>Range of values</th><th>Default value</th></tr>
<tr><td><frequency></td><td>Numeric</td><td>0 ≤ frequency ≤ 1.0E+9</td><td>0</td></tr>
<tr><td><channel_list></td><td>Discrete</td><td>One or more channels.
– (@D1)|(@D2) for single channel
– (@D1,D2) for channel 1 and 2</td><td>Required parameter</td></tr>
</table>

Remark

The reference frequency setting is only applicable for the frequency measurement function to specify the measurement results in ΔHz.

Examples

The following commands set the reference frequencies to 100 Hz and 300 Hz for channel 1 and 2 respectively.

```
SENS:DIG:REF:FREQ 100, (@D1)
SENS:DIG:REF:FREQ 300, (@D2)
```

The following query returns the reference frequency for channel 1 and 2.

```
SENS:DIG:REF:FREQ? (@D1,D2)
```

Typical response:

```
1.000000E+02,3.000000E+02
```

SENSe:DIGital:REFerence:RATio

Syntax

```
SENSe:DIGital:REFerence:RATio <ratio>,(@<channel_list>)
SENSe:DIGital:REFerence:RATio? (@<channel_list>)
```

Description

Sets the reference ratio for the specified channel(s) in dB. The reference ratio is used for conversion of the measurement result in unit ΔdB (delta decibel). The query returns the reference ratio of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<ratio>`	Numeric	–180 < ratio < 180	0
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Examples

The following commands set the reference ratio to 10 dB and 30 dB for channel 1 and 2 respectively.

```
SENS:DIG:REF:RAT 10, (@D1)
SENS:DIG:REF:RAT 30, (@D2)
```

The following query returns the reference ratio for channel 1 and 2.

```
SENS:DIG:REF:RAT? (@D1,D2)
```

Typical response:

```
1.000000E+01,3.000000E+01
```

SENSe:DIGital:REFerence:VOLTage

Syntax

```
SENSe:DIGital:REFerence:VOLTage <reference_voltage>,
(@<channel_list>)
```

```
SENSe:DIGital:REFerence:VOLTage?(@<channel_list>)
```

Description

Sets the full scale (FFS) voltage for the specified channel(s) in V. The query returns the FFS voltage of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<reference_voltage>	Numeric	$0 \leq$ voltage $\leq$ 1.0E+9	1
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Examples

The following command sets the FFS voltage to 2.5 V for channel 1.

```
SENS:DIG:REF:VOLT 2.5, (@D1)
```

The following query returns the FFS voltage for channel 1.

```
SENS:DIG:REF:VOLT? (@D1)
```

Typical response:

```
2.500000E+01
```

SENSe:DIGital:REFerence:CHANnel

Syntax

```
SENSe:DIGital:REFerence:CHANnel <reference_channel>
SENSe:DIGital:REFerence:CHANnel?
```

Description

Sets the reference channel for the phase or crosstalk measurement functions. The query returns the reference channel.

Parameter

Item	Type	Range of values	Default value
`<reference_channel>`	Numeric	1|2	1

Examples

The following commands measures phase with channel 2 as the reference channel.

```
SENS:DIG:REF:CHAN 2
SENS:DIG:FUNC:MCH PHAS
INIT:DIG:ANAL (@D1)
FETC:DIG:SCAL? FUNC3, (@D1)
```

The following query returns the reference channel.

```
SENS:DIG:REF:CHAN?
```

Typical response:

```
2
```

SENSe:DIGital:REFerence:RESult:SET

Syntax

```
SENSe:DIGital:REFerence:RESult:SET <function_number>,
<source_channel>, <@target_channel_list>
```

Description

Sets the last measurement result obtained from the specified function of the selected source as the reference value for the corresponding target digital analyzer channels. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<function_number>	Discrete	Indicates which source channel's function for the measurement reading to be recorded as the reference value of specified target channel(s). FUNCtion1|FUNCtion2|FUNCtion3|FUNCtion4	FUNCtion1
<source_channel>	Discrete	CH1|CH2	Required parameter
<target_channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- If there is no data in the last measurement or invalid data such as not a number (NaN), this command will not have any effect; the previous reference value will be remain.
- If the selected function (based on the function number) is measuring frequency, sending this command will set the reference frequency to the measured frequency reading.
- If the selected function (based on the function number) is measuring VAC, VDC, THD level, THD+N level, or any level-based measurement type, sending this command will set the reference level to the measured reading.
- If the selected function (based on the function number) is measuring THD+N ratio, SINAD, or any ratio-based measurement type, sending this command will set the reference ratio to the measured reading.
- After sending this command, you can always check the recorded reference values by sending either the **SENSe:DIGital:REFerence:LEVel**,

SENSe:DIGital:REFerence:FREQuency, or **SENSe:DIGital:REFerence:RATio** query to query for the recorded values, depending on the selected measurement function type.

Example

The following command sets the function 1 VAC result of the channel 1 as the reference value to channel 1 and channel 2.

```
SENS:DIG:REF:RES:SET FUNC1,CH1,(@D1:D2)
```

SENSe:DIGital:CALibrator:LEVel

Syntax

```
SENSe:DIGital:CALibrator:LEVel
<calibrator_level>,(@<channel_list>)
```

```
SENSe:DIGital:CALibrator:LEVel? (@<channel_list>)
```

Description

Sets the calibrator level of the specified channel(s) in unit dBSPL. The sound pressure level in dBFS that is selected will be referenced to this value. This is a standard reference threshold level for acoustic measurement. The query returns the calibrator level of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<calibrator_level>	Numeric	(–160 ≤ calibrator level < 160) dBSPL	94
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Examples

The following command sets the calibrator level for digital analyzer channel 1 and channel 2 to 94 dBSPL.

```
SENS:DIG:CAL:LEV 94,(@D1,D2)
```

The following query returns the calibrator level for digital analyzer channel 1.

```
SENS:DIG:CAL:LEV? (@D1)
```

Typical response:

```
94
```

SENSe:DIGital:AVERaging:MOVing:POINts

Syntax

```
SENSe:DIGital:AVERaging:MOVing:POINts <number_of_points>
SENSe:DIGital:AVERaging:MOVing:POINts?
```

Description

Controls the number of points to be included in the moving average. In moving averaging, when a new measurement data is added, the oldest data is discarded. The query returns the number of the moving average points.

Parameter

Item	Type	Range of values	Default value
`<number_of_points>`	Numeric	1 to 20	1

Remark

This setting is only applicable in the analyzer mode but not in the frequency domain and time domain mode.

Examples

The following command sets eight averaging points.

```
SENS:DIG:AVER:MOV:POIN 8
```

The following query returns the number of averaging points.

```
SENS:DIG:AVER:MOV:POIN?
```

Typical response:

```
8
```

SENSe:DIGital:THDN:MODE

Syntax

```
SENSe:DIGital:THDN:MODE <mode>,(<@<channel_list>)
SENSe:DIGital:THDN:MODE? (<@<channel_list>)
```

Description

Sets the measurement mode for THD+N and SINAD measurements in the digital analyzer. The query returns the measurement mode. This is the same as the Precision mode in the GUI.

Parameters

Item	Type	Range of values	Default value
<mode>	Discrete	NORMal|PRECision	NORMal
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Examples

The following command set the measurement mode to Precision for channel 1.

```
SENS:DIG:THDN:MODE PREC, (@D1)
```

The following query returns the measurement mode for channel 1.

```
SENS:DIG:THDN:MODE? (@D1)
```

Typical response:

```
PREC
```

SENSe:DIGital:FUNDamental:FREQuency:LOCK

Syntax

```
SENSe:DIGital:FUNDamental:FREQuency:LOCK <type>,(@<channel_list>)
SENSe:DIGital:FUNDamental:FREQuency:LOCK? (@<channel_list>)
```

Description

Sets the fundamental frequency lock type for THD+N level, THD+N ratio, and SINAD measurements in the digital analyzer. The query returns the fundamental frequency lock type. Multiple responses are separated by commas.

`AUTO`	Automatically determines the fundamental frequency by selecting the signal with the highest power from the incoming signal at the analyzer
`GLOCk`	The fundamental frequency is determined by the frequency value set at the corresponding channel of the generator
`CUSTom`	The fundamental frequency value is user-defined.

Parameters

Item	Type	Range of values	Default value
`<type>`	Discrete	AUTO|GLOCk|CUSTom	AUTO
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- This setting will only take effect if any measurement function of the specified channel has been set to THD+N level, THD+N ratio, or SINAD.
- When you select `CUSTom`, you will need to define the fundamental frequency using the **SENSe:DIGital:FUNDamental:FREQuency** command.

Examples

The following command sets the fundamental frequency lock type to `AUTO` for channel 1.

```
SENS:DIG:FUND:FREQ:LOCK AUTO,(@D1)
```

The following query returns the SINAD fundamental frequency lock type for channel 1.

```
SENS:DIG:FUND:FREQ:LOCK? (@D1)
```

Typical response:

```
AUTO
```

SENSe:DIGital:FUNDamental:FREQuency

Syntax

```
SENSe:DIGital:FUNDamental:FREQuency
<fundamental_frequency>[<unit>], (@<channel_list>)
```

```
SENSe:DIGital:FUNDamental:FREQuency? (@<channel_list>)
```

Description

Sets the custom fundamental frequency value for SINAD, THD+N Ratio, and THD+N Level measurement. The query returns the fundamental frequency value.

Parameters

Item	Type	Range of values	Default value
`<fundamental_frequency>`	Numeric	(10 – 0.45 fs) Hz, where fs is the input sampling frequency	1000 Hz
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- This setting is only applicable when the Function 2 measurement of the specified channel is set to SINAD, THD+N Ratio, or THD+N Level.
- This setting is only valid when the fundamental frequency lock type is set to CUSTom.

Examples

The following command sets the fundamental frequency value to 2 kHz for digital analyzer channel 1.

```
SENS:DIG:FUND:FREQ 2kHz, (@D1)
```

The following query returns the fundamental frequency value for digital analyzer channel 1.

```
SENS:DIG:FUND:FREQ? (@D1)
```

Typical response:

```
2.000000E+03
```

SENSe:DIGital:THD:HARMonic:COMPonent

Syntax

```
SENSe:DIGital:THD:HARMonic:COMPonent
(<harmonics_components>),(@<channel_list>)
```

```
SENSe:DIGital:THD:HARMonic:COMPonent? (@<channel_list>)
```

Description

Sets the THD measurement harmonics components for the digital analyzer specified channel(s). The query returns the THD measurement harmonics components of the digital analyzer selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<harmonics_components>`	Numeric	One or more harmonics can be selected. – (2) for 2nd harmonic – (2,5) for 2nd and 5th harmonics – (2:6) for 2nd to 6th harmonics	(2:6)
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Examples

The following command sets the THD measurement harmonic components 2 and 3 for channel 1 and channel 2.

```
SENS:DIG:THD:HARM:COMP (2,3),(@D1,D2)
```

The following query returns the THD measurement harmonic components for channel 1 and channel 2.

```
SENSe:DIG:THD:HARM:COMP? (@D1,D2)
```

Typical response:

```
2,3
```

SENSe:DIGital:FFT:WINDow

Syntax

```
SENSe:DIGital:FFT:WINDow <type>
```

```
SENSe:DIGital:FFT:WINDow?
```

Description

Sets the window function for frequency domain analysis. The query returns the window function.

The window functions with their corresponding `<type>` parameters are listed as follows.

HANN	Hanning window
RECTangular	Rectangular window
BLACkman	Blackman-Harris window
RIFe1	Rife-Vincent 1 window
RIFe3	Rife-Vincent 3 window
FLATtop	Flattop window
HAMMing	Hamming window

Parameter

Item	Type	Range of values	Default value
<type>	Discrete	RECTangular|HANN|BLACkman|RIFe1|RIFe3|HAMMing|FLATtop	RECTangular

Examples

The following command sets the Hanning window function.

```
SENS:DIG:FFT:WIND HANN
```

The following query returns the window function.

```
SENS:DIG:FFT:WIND?
```

Typical response:

```
HANN
```

SENSe:DIGital:AVERaging:SYNC:POINts

Syntax

```
SENSe:DIGital:AVERaging:SYNC:POINts <number_of_points>
```

```
SENSe:DIGital:AVERaging:SYNC:POINts?
```

Description

Sets the number of points for the synchronous averaging. Synchronous averaging reduces noise levels by averaging the acquired data in the time domain. The query returns the number of averaging points.

Parameter

Item	Type	Range of values	Default value
`<number_of_points>`	Numeric	1 to 64	1

Remark

This setting is only applicable if you trigger from the channel 1 or channel 2 input.

Examples

The following command sets eight averaging points.

```
SENS:DIG:AVER:SYNC:POIN 8
```

The following query returns the number of averaging points.

```
SENS:DIG:AVER:SYNC:POIN?
```

Typical response:

```
8
```

SENSe:DIGital:BERT:UNIT

Syntax

```
SENSe:DIGital:BERT:UNIT <unit>
SENSe:DIGital:BERT:UNIT?
```

Description

Sets the unit of the Total Error and Total Bits for the Bit Error Rate Test (BERT). The query returns the unit of the Total Error and Total Bits.

HEX	Hexadecimal
DEC	Decimal

Parameter

Item	Type	Range of values	Default value
<unit>	Discrete	HEX|DEC	DEC

Examples

The following command sets the unit of the Total Error and Total Bits to HEX

```
SENS:DIG:BERT:UNIT HEX
```

The following query returns the unit of the Total Error and Total Bits.

```
SENS:DIG:BERT:UNIT?
```

Typical response:

```
HEX
```

SENSe:DIGital:BERT:RESet

Syntax

```
SENSe:DIGital:BERT:RESet
```

Description

Resets the Total Samples, Total Errors, and Instantaneous Error of the Bit Error Rate Test (BERT) to zero.

Remark

This setting takes effect only in BERT mode. You can switch to BERT mode using the SENSe:DIGital:MEASurement:MODE command.

Examples

The following command resets the Total Samples, Total Errors, and Instantaneous Error to zero

```
SENS:DIG:BERT:RES
```

SENSe:DIGital:BITS:REFResh:RATE

Syntax

```
SENSe:DIGital:BITS:REFResh:RATE <rate>
SENSe:DIGital:BITS:REFResh:RATE?
```

Description

Sets the refresh rate of the active bits for the digital bits analysis. The query returns the refresh rate of the active bits .

Parameter

Item	Type	Range of values	Default value
`<rate>`	Discrete	1 to 65536	1

Examples

The following command sets the the refresh rate to 50

```
SENSe:DIGital:BITS:REFResh:RATE  50
```

The following query returns the refresh rate unit.

```
SENSe:DIGital:BITS:REFResh:RATE?
```

Typical response:

```
50
```

14 SOURce Subsystem

This chapter describes the `SOURce` subsystem commands.

SOURce[:ANALog]:DTMF:MODE

Syntax

```
SOURce[:ANALog]:DTMF:MODE <value>,(@<channel_list>)
SOURce[:ANALog]:DTMF:MODE? (@<channel_list>)
```

Description

Sets the DTMF mode for the specified channel(s). The query returns the DTMF mode of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<value>`	Discrete	SINGle|SEQuence	SINGle
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is only applicable for the DTMF waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

Examples

The following command sets the DTMF mode for channel 1 to sequence.

```
SOUR:DTMF:MODE SEQ,(@1)
```

The following query returns the DTMF mode of channel 1.

```
SOUR:DTMF:MODE? (@1)
```

Typical response:

```
SEQ
```

SOURce[:ANALog]:DTMF:PAUSe :TIME

Syntax

```
SOURce[:ANALog]:DTMF:PAUSe:TIME <value>,(@<channel_list>)
SOURce[:ANALog]:DTMF:PAUSe:TIME? (@<channel_list>)
```

Description

Sets the pause time of the DTMF waveform for the specified channel(s) in ms. The query returns the pause time of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<value>	Numeric	20 ms to 1000 ms	90 ms
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is only applicable for the DTMF waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

Examples

The following command sets the pause time to 90 ms for channel 1.

```
SOUR:DTMF:PAUS:TIME 90,(@1)
```

The following query returns the pause time for channel 1.

```
SOUR:DTMF:PAUS:TIME? (@1)
```

Typical response:

```
90
```

SOURce[:ANALog]:DTMF:REPeat

Syntax

```
SOURce[:ANALog]:DTMF:REPeat <state>,(@<channel_list>)
SOURce[:ANALog]:DTMF:REPeat? (@<channel_list>)
```

Description

Enables or disables the DTMF waveform repeating function. The query returns the output state. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	ON
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is only applicable for the DTMF waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

Examples

The following command repeats the DTMF waveform for channel 1.

```
SOUR:DTMF:REP ON,(@1)
```

The following query returns the DTMF waveform output state for channel 1.

```
SOUR:DTMF:REP? (@1)
```

Typical response:

```
1
```

SOURce[:ANALog]:DTMF:SEQuence

Syntax

```
SOURce[:ANALog]:DTMF:SEQuence <value>,(@<channel_list>)
SOURce[:ANALog]:DTMF:SEQuence? (@<channel_list>)
```

Description

Sets the character sequence of the DTMF waveform for the specified channel(s). The query returns the character sequence of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<value>`	Discrete	0|1|2|3|4|5|6|7|8|9|A|B|C|D|#|* Character sequence in quoted string. For example, ‘1234567890#*ABCD’.	Required parameter
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is only applicable for the DTMF waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

Examples

The following command sets the DTMF character sequence to ‘1234’ for channel 1.

```
SOUR:DTMF:SEQ ‘1234’,(@1)
```

The following query returns the DTMF character sequence for channel 1.

```
SOUR:DTMF:SEQ? (@1)
```

Typical response:

```
1234
```

SOURce[:ANALog]:DTMF:TONE :DELAY

Syntax

```
SOURce[:ANALog]:DTMF:TONE:DELAY <value>,(@<channel_list>)
SOURce[:ANALog]:DTMF:TONE:DELAY? (@<channel_list>)
```

Description

Sets the tone delay of the DTMF waveform for the specified channel(s) in ms. The query returns the tone delay of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<value>`	Numeric	20 ms to 1000 ms	90 ms
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is only applicable for the DTMF waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

Examples

The following command sets the tone delay to 90 ms for channel 1.

```
SOUR:DTMF:TONE:DEL 90,(@1)
```

The following query returns the tone delay for channel 1.

```
SOUR:DTMF:TONE:DEL? (@1)
```

Typical response:

```
90
```

SOURce[:ANALog]:DTMF:TONE :DURation

Syntax

```
SOURce[:ANALog]:DTMF:TONE:DURation <value>,(@<channel_list>)
SOURce[:ANALog]:DTMF:TONE:DURation? (@<channel_list>)
```

Description

Sets the tone duration of the DTMF waveform for the specified channel(s) in ms. The query returns the tone duration of the selected channel(s). Multiple responses are separated by commas.

Parameter

Item	Type	Range of values	Default value
`<value>`	Numeric	20 ms to 1000 ms	90 ms
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is only applicable for the DTMF waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

Examples

The following command sets the tone duration to 90 ms for channel 1.

```
SOUR:DTMF:TONE:DUR 90,(@1)
```

The following query returns the tone duration for channel 1.

```
SOUR:DTMF:TONE:DUR? (@1)
```

Typical response:

```
90
```

SOURce[:ANALog]:DTMF:VOLTage

Syntax

```
SOURce[:ANALog]:DTMF:VOLTage <value>,(@<channel_list>)
```

```
SOURce[:ANALog]:DTMF:VOLTage? (@<channel_list>)
```

Description

Sets the sum level of the DTMF waveform for the specified channel(s) in dBu. The query returns the sum level of the selected channel(s). Multiple responses are separated by commas.

Parameter

Item	Type	Range of values	Default value
`<value>`	Numeric	–9 dBu to 0 dBu	–4.5 dBu
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is only applicable for the DTMF waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

Examples

The following command sets the sum level to –4.5 dBu for channel 1.

```
SOUR:DTMF:VOLT -4.5,(@1)
```

The following query returns the sum level for channel 1.

```
SOUR:DTMF:VOLT? (@1)
```

Typical response:

```
-4.5
```

SOURce[:ANALog]:DTMF:VOLTage :RATio

Syntax

```
SOURce[:ANALog]:DTMF:VOLTage:RATio <value>,(@<channel_list>)
SOURce[:ANALog]:DTMF:VOLTage:RATio? (@<channel_list>)
```

Description

Sets the level difference of the DTMF waveform for the specified channel(s) in dB. Level difference is the level of the high-frequency tone minus the level of the low-frequency tone. The query returns the level difference of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<value>`	Numeric	0 dB to 4 dB	2 dB
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is only applicable for the DTMF waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

Examples

The following command sets the level difference to 2 dB for channel 1.

```
SOUR:DTMF:VOLT:RAT 2,(@1)
```

The following query returns the level difference for channel 1.

```
SOUR:DTMF:VOLT:RAT? (@1)
```

Typical response:

```
2
```

SOURce[:ANALog]:DTMF:VOLTage:SUMMation

Syntax

```
SOURce[:ANALog]:DTMF:VOLTage:SUMMation <value>, (@<channel_list>)
SOURce[:ANALog]:DTMF:VOLTage:SUMMation? (@<channel_list>)
```

Description

Sets the DTMF signal's level summation method. The query returns the DTMF signal's level summation method. Multiple responses are separated by commas.

RSS	Amplitude shown is the Root Sum Square summation result of the 2 levels. The analyzer will read the same amplitude set at the generator regardless of the tone ratio.
LINear	Amplitude shown is the linear summation result of the 2 levels. The level of each tone can be calculated from the tone ratio.

Parameter

Item	Type	Range of values	Default value
<channel_list>	Numeric	One or more channels. – (@2) for channel 2 – (@1,4) for channel 1 and 4 – (@1:3) for channel 1 to 3	Required parameter
<value>	Discrete	RSS|LINear	RSS

Remarks

- Switch to the DTMF waveform type before sending this command.
- Use the **SOURce[:ANALog]:VOLTage:SUMMation** command for other dual tones signal.

Examples

The following command sets the method for the level summation of the signal at channel 1 to RSS.

```
SOUR:DTMF:VOLT:SUMM RSS, (@1)
```

The following query returns the method for the level summation of the signal for channel 1.

```
SOUR:DTMF:VOLT:SUMM?(@1)
```

Typical response:

```
RSS
```

SOURce[:ANALog]:FREQuency[<j>][:CW]

Syntax

```
SOURce[:ANALog]:FREQuency[<j>][:CW]
<frequency>[<unit>],(@<channel_list>)
```

```
SOURce[:ANALog]:FREQuency[<j>][:CW]? (@<channel_list>)
```

Description

Sets the signal frequency for the specified channel(s) in Hz. The query returns the frequency of the selected channel(s) in Hz. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<j>`	Numeric	1|2	1
`<frequency>`	Numeric	Refer to "**Appendix A: Waveform Frequency Range and Default Values**" on page 660.	1000 Hz
`<unit>`	Discrete	Hz	Hz
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– You can also include a multiplier for the unit, for example, kHz. The 'k' is the multiplier for the unit Hz.

– For sine, variable phase, and square waveforms, the `SOURce:FREQuency1` command sets the frequency. If the `SOURce:FREQuency2` command is sent when the waveform is sine, variable phase, or square, the second frequency value will be ignored.

– For dual waveform, `SOURce:FREQuency1` represents the frequency of the first sine component while `SOURce:FREQuency2` represents the frequency of the second sine component.

– For the SMPTE IMD 1:1, 4:1, and 10:1 waveforms, use the **SOURce[:ANALog]:FREQuency:LOWer** command to set the lower frequency while **SOURce[:ANALog]:FREQuency:UPPer** represents the upper frequency.

- For the DFD IEC 60118 waveform, use the **SOURce[:ANALog]:FREQuency:UPPer** command to set the upper frequency and the **SOURce[:ANALog]:FREQuency:DIFFerence** command to set the frequency difference.
- For the DFD IEC 60268 waveform, use the **SOURce[:ANALog]:FREQuency:DIFFerence** command to set the frequency difference and the **SOURce[:ANALog]:FREQuency:CENTer** command to set the center frequency.
- This setting is not applicable for the DC, Gaussian noise, rectangular noise, multitone, pink noise, DTMF, and arbitrary waveforms.
- For backward compatibility with U8903A, `SOURce:FREQuency1` can also be used to set the lower frequency and `SOURce:FREQuency2` can also be used to set the upper frequency.

Examples

The following command sequence sets the sine waveform frequency for channel 1 and square waveform frequency for channel 2 to 1 kHz and 5 kHz respectively.

```
SOUR:FREQ 1000,(@1)
SOUR:FREQ 5000,(@2)
```

The following query returns the frequency values of channel 1 and channel 2 in Hz.

```
SOUR:FREQ? (@1,2)
```

Typical response:

```
1.000000E+03,5.000000E+03
```

SOURce[:ANALog]:FREQuency:CENTer

Syntax

```
SOURce[:ANALog]:FREQuency:CENTer
<frequency>[<unit>],(@<channel_list>)
```

```
SOURce[:ANALog]:FREQuency:CENTer? (@<channel_list>)
```

Description

Sets the center frequency of the DFD IEC 60268 waveform for the specified channel(s) in Hz. The center frequency determines the frequency around which the two tones are spaced in equal increments above and below. The query returns the center frequency value of the selected channel(s) in Hz. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<frequency>`	Numeric	3 kHz to 79 kHz	10 kHz
`<unit>`	Discrete	Hz	Hz
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– You can also include a multiplier for the unit, for example, kHz. The ‘k’ is the multiplier for the unit Hz.

– This setting is only applicable for the DFD IEC 60268 waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

Examples

The following command sequence sets the center frequencies for channel 1 and channel 2 to 1 kHz and 5 kHz respectively.

```
SOUR:FREQ:CENT 1kHz,(@1)
SOUR:FREQ:CENT 5kHz,(@2)
```

The following query returns the center frequency values of channel 1 and channel 2 in Hz.

```
SOUR:FREQ:CENT? (@1,2)
```

Typical response:

```
1.000000E+03,5.000000E+03
```

SOURce[:ANALog]:FREQuency:DIFFerence

Syntax

```
SOURce[:ANALog]:FREQuency:DIFFerence
<frequency>[<unit>],(@<channel_list>)
```

```
SOURce[:ANALog]:FREQuency:DIFFerence? (@<channel_list>)
```

Description

Sets the frequency difference of the DFD IEC 60268 and DFD IEC 60118 waveforms for the specified channel(s) in Hz. The frequency difference determines the difference frequency (spacing) between the two tones of the DFD IEC 60268 and DFD IEC 60118 signals. The query returns the frequency difference of the selected channel(s) in Hz. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<frequency>`	Numeric	80 Hz to 2 kHz	80 Hz
`<unit>`	Discrete	Hz	Hz
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– You can also include a multiplier for the unit, for example, kHz. The ‘k’ is the multiplier for the unit Hz.

– This setting is only applicable for the DFD IEC 60118 waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

Examples

The following command sequence sets the frequency difference values for channel 1 and channel 2 to 100 Hz and 80 Hz respectively.

```
SOUR:FREQ:DIFF 100Hz,(@1)
SOUR:FREQ:DIFF 80Hz,(@2)
```

The following query returns the frequency difference values of channel 1 and channel 2 in Hz.

```
SOUR:FREQ:DIFF? (@1,2)
```

Typical response:

```
1.000000E+02,8.000000E+01
```

SOURce[:ANALog]:FREQuency:LOWer

Syntax

```
SOURce[:ANALog]:FREQuency:LOWer
<frequency>[<unit>],(@<channel_list>)
```

```
SOURce[:ANALog]:FREQuency:LOWer? (@<channel_list>)
```

Description

Sets the lower frequency of the SMPTE 1:1, SMPTE 4:1, and SMPTE 10:1 waveforms for the specified channel(s) in Hz. The lower frequency determines the frequency of the lower frequency tone in the two-tone waveform. The query returns the lower frequency of the selected channel(s) in Hz. Multiple responses are separated by commas.

Parameter

Item	Type	Range of values	Default value
`<frequency>`	Numeric	40 Hz to 500 Hz	60 Hz
`<unit>`	Discrete	Hz	Hz
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- You can also include a multiplier for the unit, for example, kHz. The ‘k’ is the multiplier for the unit Hz.
- This setting is only applicable for the SMPTE 1:1, SMPTE 4:1, and SMPTE 10:1 waveforms. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

Examples

The following command sequence sets the lower frequency values of the SMPTE 1:1 waveform for channel 1 and channel 2 to 100 Hz and 80 Hz respectively.

```
SOUR:FREQ:LOW 100Hz,(@1)
SOUR:FREQ:LOW 80Hz,(@2)
```

The following query returns the lower frequency values of channel 1 and channel 2 in Hz.

```
SOUR:FREQ:LOW? (@1,2)
```

Typical response:

```
1.000000E+02,8.000000E+01
```

SOURce[:ANALog]:FREQuency:UPPer

Syntax

```
SOURce[:ANALog]:FREQuency:UPPer
<frequency>[<unit>],(@<channel_list>)
```

```
SOURce[:ANALog]:FREQuency:UPPer? (@<channel_list>)
```

Description

Sets the upper frequency of the DFD IEC 60118, SMPTE 1:1, SMPTE 4:1, and SMPTE 10:1 waveforms for the specified channel(s) in Hz. The upper frequency determines the frequency of the higher frequency tone in the two-tone waveform. The query returns the upper frequency of the selected channel(s) in Hz. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<frequency>`	Numeric	3 kHz to 80 kHz (DFD IEC 60118) 2 kHz to 60 kHz (SMPTE 1:1, SMPTE 4:1, and SMPTE 10:1)	10 Hz 7 kHz
`<unit>`	Discrete	Hz	Hz
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– You can also include a multiplier for the unit, for example, kHz. The ‘k’ is the multiplier for the unit Hz.

– This setting is only applicable for the DFD IEC 60118, SMPTE 1:1, SMPTE 4:1, and SMPTE 10:1 waveforms. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

Examples

The following command sequence sets the upper frequency values for channel 1 and channel 2 to 5 kHz and 10 kHz respectively.

```
SOUR:FREQ:UPP 5kHz,(@1)
SOUR:FREQ:UPP 10kHz,(@2)
```

The following query returns the upper frequency values of channel 1 and channel 2 in Hz.

```
SOUR:FREQ:UPP? (@1,2)
```

Typical response:

```
5.000000E+03,1.000000E+04
```

SOURce[:ANALog]:FUNCtion

Syntax

```
SOURce[:ANALog]:FUNCtion <waveform_type>,(@<channel_list>)
SOURce[:ANALog]:FUNCtion? (@<channel_list>)
```

Description

Sets the analog generator waveform type for the specified channel(s). The query returns the waveform type of the selected channel(s). Multiple responses are separated by commas.

The waveform types with their corresponding <waveform_type> parameters are listed as follows.

SINE	Sine waveform
DUAL	Dual waveform
VPHase	Variable phase waveform
SMPTe11	SMPTE IMD 1 to 1 waveform
SMPTe41	SMPTE IMD 4 to 1 waveform
SMPTe101	SMPTE IMD 10 to 1 waveform
DFDiec118	DFD IEC 60118 waveform
DFDiec268	DFD IEC 60268 waveform
WGAussian	Gaussian PDF white noise signal
WRECtangular	Rectangular PDF white noise signal
PNOise	Pink noise signal
DC	DC signal
MULTitone	Multitone waveform
SQUare	Square waveform
ARBitrary	Arbitrary waveform
DTMF	DTMF signal

Parameters

Item	Type	Range of values	Default value
<waveform_type>	Discrete	SINE|DUAL|VPHase|SMPTe11|SMPTe41|SMPTe101|DFDiec118|DFDiec268|WGAussian|WRECtagular|PNOise|DC|MULTitone|SQUare|ARBitrary|DTMF	SINE
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- If you change the waveform type, the particular parameter values of the previous waveform will be set to the default values of the current waveform.
- Selecting the variable phase waveform on a selected channel will change the waveform type for all channels to variable phase.
- Refer to "**Appendix C: Waveform Parameters**" on page 664 for the configurable parameters of the corresponding waveform types.

Examples

The following command sequence sets the waveform types for channel 1 and channel 2 to sine and square waveforms respectively.

```
SOUR:FUNC SINE,(@1)
SOUR:FUNC SQU,(@2)
```

The following query returns the waveform types of channel 1 and channel 2.

```
SOUR:FUNC? (@1,2)
```

Typical response:

```
SINE,SQU
```

SOURce[:ANALog]:MULTitone:COUNt

Syntax

```
SOURce[:ANALog]:MULTitone:COUNt <tone_count>,(@<channel_list>)
SOURce[:ANALog]:MULTitone:COUNt? (@<channel_list>)
```

Description

Sets the tones of the multitone waveform for the specified channel(s). Tones refer to the number of signal frequency components. The query returns the number of tones of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<tone_count>	Numeric	2 to 60	2
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- This setting is only applicable for the multitone waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.
- If there are tones with the same frequency, error -221,“Settings Conflict” will be generated and the duplicated tone with the same frequency will be removed.

Examples

The following command sequence sets the tones for channel 1 and channel 2.

```
SOUR:MULT:COUN 3,((@1,2))
```

The following query returns the number of tones of channel 1 and channel 2.

```
SOUR:MULT:COUN? (@1,2)
```

Typical response:

```
3,3
```

SOURce[:ANALog]:MULTitone:CRESt?

Syntax

```
SOURce[:ANALog]:MULTitone:CRESt? (@<channel_list>)
```

Description

Queries the crest factor of the multitone waveform for the selected channel(s). Multiple responses are separated by commas.

Parameter

Item	Type	Range of values	Default value
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is only applicable for the multitone waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

Examples

The following query returns the crest factor of the multitone waveform for channel 1.

```
SOUR:MULT:CRESt? (@1)
```

Typical response:

```
1.4142000E+0
```

SOURce[:ANALog]:MULTitone:FREQuency:SPACing

Syntax

```
SOURce[:ANALog]:MULTitone:FREQuency:SPACing
<spacing>,(@<channel_list>)
```

```
SOURce[:ANALog]:MULTitone:FREQuency:SPACing? (@<channel_list>)
```

Description

Sets the frequency spacing type between the start and stop frequency of the multitone waveform for the specified channel(s). The query returns the frequency spacing type of the selected channel(s) in LIN or LOG. Multiple responses are separated by commas.

LINear	Linear frequency spacing
LOG	Logarithmic frequency spacing

Parameters

Item	Type	Range of values	Default value
<spacing>	Discrete	LINear|LOG	LINear
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- This setting is only applicable for the multitone waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.
- If there are tones with the same frequency, error -221,"Settings Conflict" will be generated and the duplicated tone with the same frequency will be removed.

Examples

The following command sets the frequency spacing for channel 1 to logarithmic.

```
SOUR:MULT:FREQ:SPAC LOG,(@1)
```

The following query returns the frequency spacing type of channel 1.

```
SOUR:MULT:FREQ:SPAC? (@1)
```

Typical response:

```
LOG
```

SOURce[:ANALog]:MULTitone:FREQuency:STARt

Syntax

```
SOURce[:ANALog]:MULTitone:FREQuency:STARt
<frequency>[<unit>],(@<channel_list>)
```

```
SOURce[:ANALog]:MULTitone:FREQuency:STARt? (@<channel_list>)
```

Description

Sets the start frequency of the multitone waveform for the specified channel(s). The start frequency defines the lowest tone frequency in the multitone waveform. The query returns the start frequency of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<frequency>`	Numeric	5 Hz to 80 kHz	1000 Hz
`<unit>`	Discrete	Hz	Hz
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– You can also include a multiplier for the unit, for example, kHz. The ‘k’ is the multiplier for the unit Hz.

– This setting is only applicable for the multitone waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

– If there are tones with the same frequency, error -221,“Settings Conflict” will be generated and the duplicated tone with the same frequency will be removed.

Examples

The following command sets the start frequency value for channel 1 to 5 kHz.

```
SOUR:MULT:FREQ:STAR 5000,(@1)
```

The following query returns the start frequency value of channels 1 in Hz.

```
SOUR:MULT:FREQ:STAR? (@1)
```

Typical response:

```
5.000000E+03
```

SOURce[:ANALog]:MULTitone:FREQuency:STOP

Syntax

```
SOURce[:ANALog]:MULTitone:FREQuency:STOP
<frequency>[<unit>],(@<channel_list>)
```

```
SOURce[:ANALog]:MULTitone:FREQuency:STOP? (@<channel_list>)
```

Description

Sets the stop frequency of the multitone waveform for the specified channel(s). The stop frequency defines the highest tone frequency in the multitone waveform. The query returns the stop frequency of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<frequency>`	Numeric	5 Hz to 80 kHz	5000 Hz
`<unit>`	Discrete	Hz	Hz
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– You can also include a multiplier for the unit, for example, kHz. The ‘k’ is the multiplier for the unit Hz.

– This setting is only applicable for the multitone waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

– If there are tones with the same frequency, error -221,“Settings Conflict” will be generated and the duplicated tone with the same frequency will be removed.

Examples

The following command sets the stop frequency value for channel 1 to 10 kHz.

```
SOUR:MULT:FREQ:STOP 10kHz,(@1)
```

The following query returns the stop frequency value of channels 1 in Hz.

```
SOUR:MULT:FREQ:STOP? (@1)
```

Typical response:

```
1.000000E+04
```

SOURce[:ANALog]:MULTitone:OPTimize

Syntax

```
SOURce[:ANALog]:MULTitone:OPTimize (@<channel_list>)
```

Description

Optimizes all the tones of the multitone waveform for the selected channel(s).

Parameter

Item	Type	Range of values	Default value
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- This setting is only applicable for the multitone waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.
- After this command is sent, all existing tones phase, crest factor, rms, and peak will be optimized.
- For the optimization of the maximum number of tones (60) and maximum waveform length (32768), the measured time for completion is 8 s to 10 s.

Example

The following command optimizes all the tones of the multitone waveform for channel 1.

```
SOUR:MULT:OPT (@1)
```

SOURce[:ANALog]:MULTitone:PEAK?

Syntax

```
SOURce[:ANALog]:MULTitone:PEAK? (@<channel_list>)
```

Description

Queries the peak of the multitone waveform for the selected channel(s). Multiple responses are separated by commas.

Parameter

Item	Type	Range of values	Default value
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is only applicable for the multitone waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

Example

The following query returns the peak of the multitone waveform for channel 1.

```
SOUR:MULT:PEAK? (@1)
```

Typical response:

```
1.4142000E+0
```

SOURce[:ANALog]:MULTitone:RMS?

Syntax

```
SOURce[:ANALog]:MULTitone:RMS? (@<channel_list>)
```

Description

Queries the rms value of the multitone waveform for the selected channel(s). Multiple responses are separated by commas.

Parameter

Item	Type	Range of values	Default value
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is only applicable for the multitone waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

Example

The following query returns the rms value of the multitone waveform for channel 1.

```
SOUR:MULT:RMS? (@1)
```

Typical response:

```
1.4142000E+0
```

SOURce[:ANALog]:MULTitone:TONE:ADD

Syntax

```
SOURce[:ANALog]:MULTitone:TONE:ADD
<index>,<frequency>[<unit1>],<voltage>[<unit2>],<phase>,(@<channe
l_list>)
```

Description

Adds a customized tone into the multitone waveform for the specified channel(s).

Parameters

Item	Type	Range of values	Default value
`<index>`	Numeric	0 to 59	0
`<frequency>`	Numeric	5 Hz to 80 kHz	1 kHz
`<unit1>`	Discrete	Hz	Hz
`<voltage>`	Numeric	0 to 8 Vrms (0 to 11.3 Vp) for unbalanced 0 to 16 Vrms (0 to 22.6 Vp) for balanced	1 Vrms
`<unit2>`	Discrete	Vrms|Vpp|Vp|dBV|dBu	Vrms
`<phase>`	Numeric	–180 to ~179.99	0
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– This setting is only applicable for the multitone waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

– If there are tones with the same frequency, error -221,“Settings Conflict” will be generated and the duplicated tone with the same frequency will be removed.

– This command will add a customized tone into the existing multitone waveform. The position of the added tone will be determined by the index. The index must be less than the total number of tones.

– The `<voltage>` entered is the ratio voltage relative to the total amplitude.

Example

The following command adds a 5 kHz frequency, 0.5 Vrms voltage, and 80° phase tone to the current multitone waveform for channel 1 at position 5.

```
SOUR:MULT:TONE:ADD 4,5000,0.5,80,(@1)
```

SOURce[:ANALog]:MULTitone:TONE:CLEar

Syntax

```
SOURce[:ANALog]:MULTitone:TONE:CLEar (@<channel_list>)
```

Description

Clears all the tones of the multitone waveform for the specified channel(s).

Parameter

Item	Type	Range of values	Default value
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- This setting is only applicable for the multitone waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.
- After this command is sent, all existing tones will be cleared and two tones will be created. The first tone will have the minimum frequency, half of the total amplitude, and 0° phase. The second tone will have the maximum frequency, half of the total amplitude, and 0° phase.

Example

The following command clears all the tones of the multitone waveform for channel 1.

```
SOUR:MULT:TONE:CLE (@1)
```

SOURce[:ANALog]:MULTitone:TONE:DELete

Syntax

```
SOURce[:ANALog]:MULTitone:TONE:DELete <index>,(@<channel_list>)
```

Description

Deletes a specific tone from the multitone waveform for the specified channel(s).

Parameters

Item	Type	Range of values	Default value
<index>	Numeric	0 to 59	0
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– This setting is only applicable for the multitone waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

– This command will delete a tone from the existing multitone waveform. The deleted tone will be determined by the index. The index must be less than the total number of tones.

Example

The following command deletes the tone 5 of the multitone waveform for channel 1.

```
SOUR:MULT:TONE:DEL 4,(@1)
```

SOURce[:ANALog]:MULTitone:TONE:FREQuency

Syntax

```
SOURce[:ANALog]:MULTitone:TONE:FREQuency
<frequency>[<unit>],(<tone_list>),(@<channel_list>)
```

```
SOURce[:ANALog]:MULTitone:TONE:FREQuency?
(<tone_list>),(@<channel_list>)
```

Description

Sets the frequency of the tone(s) in the multitone waveform for the specified channel(s). The query returns the frequency of the tone(s) of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<frequency>`	Numeric	5 Hz to 80 kHz	1 kHz
`<unit>`	Discrete	Hz	Hz
`<tone_list>`	Discrete	One or more tones. – (1) for tone 1 – (1,2) for tone 1 and 2 – (1:10) for tone 1 through 10	Required parameter
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– You can also include a multiplier for the unit, for example, kHz. The ‘k’ is the multiplier for the unit Hz.

– This setting is only applicable for the multitone waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

– If there are tones with the same frequency, error -221,“Settings Conflict” will be generated and the duplicated tone with the same frequency will be removed.

Examples

The following command sets the frequency of tone 5 for channel 1 to 2 kHz.

```
SOUR:MULT:TONE:FREQ 2kHz,(5),(@1)
```

The following query returns the frequency of tone 5 and tone 6 for channel 1.

```
SOUR:MULT:TONE:FREQ? (5,6),(@1)
```

Typical response:

```
2.000000E+03,3.000000E+03
```

SOURce[:ANALog]:MULTitone:TONE:PHASe

Syntax

```
SOURce[:ANALog]:MULTitone:TONE:PHASe
<phase>,(<tone_list>),(@<channel_list>)
```

```
SOURce[:ANALog]:MULTitone:TONE:PHASe?
(<tone_list>),(@<channel_list>)
```

Description

Sets the phase of the tone(s) in the multitone waveform for the specified channel(s). The query returns the phase of the tone(s) of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<phase>`	Numeric	–180 to ~179.99	0
`<tone_list>`	Discrete	One or more tones. – (1) for tone 1 – (1,2) for tone 1 and 2 – (1:10) for tone 1 through 10	Required parameter
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is only applicable for the multitone waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

Examples

The following command sets the phase of the tone 5 and tone 6 for channel 1 to 90°.

```
SOUR:MULT:TONE:PHAS 90,(5,6),(@1)
```

The following query returns the phase of tone 5 and tone 6 for channel 1.

```
SOUR:MULT:TONE:PHAS? (5,6),(@1)
```

Typical response:

```
9.000000E+01,9.000000E+01
```

SOURce[:ANALog]:MULTitone:TONE:VOLTage

Syntax

```
SOURce[:ANALog]:MULTitone:TONE:VOLTage
<voltage>[<unit>],(<tone_list>),(@<channel_list>)
```

```
SOURce[:ANALog]:MULTitone:TONE:VOLTage?
(<tone_list>),(@<channel_list>)
```

Description

Sets the voltage of the tone(s) in the multitone waveform for the specified channel(s). The query returns the voltage of the tone(s) of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<voltage>`	Numeric	0 to 8 Vrms (0 to 11.3 Vp) for unbalanced 0 to 16 Vrms (0 to 22.6 Vp) for balanced	1 Vrms
`<unit>`	Discrete	Vrms|Vpp|Vp|dBV|dBu|dBSPL	Vrms
`<tone_list>`	Discrete	One or more tones. – (1) for tone 1 – (1,2) for tone 1 and 2 – (1:10) for tone 1 through 10	Required parameter
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- You can also include a multiplier for the unit, for example, mVrms. The ‘m’ is the multiplier for the unit Vrms.
- This setting is only applicable for the multitone waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.
- The `<voltage>` entered is the ratio voltage relative to the total amplitude.

Examples

The following command sets the voltage of tone 5 and tone 6 for channel 1 to 0.5 Vrms.

```
SOUR:MULT:TONE:VOLT 0.5,(5,6),(@1)
```

The following query returns the voltage of tone 5 and tone 6 for channel 1.

```
SOUR:MULT:TONE:VOLT? (5,6),(@1)
```

Typical response:

```
5.000000E-01,5.000000E-01
```

SOURce[:ANALog]:MULTitone:WLEN

Syntax

```
SOURce[:ANALog]:MULTitone:WLEN <length>,(@<channel_list>)
SOURce[:ANALog]:MULTitone:WLEN? (@<channel_list>)
```

Description

Sets the waveform length of the multitone waveform for the specified channel(s). The waveform length determines the number of samples used to create one iteration of the multitone waveform. Longer waveform length provides higher frequency resolution but takes more time in generation and processing. The query returns the waveform length of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<length>	Discrete	L1024|L2048|L4096|L8192|L16384|L32768	L1024
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– This setting is only applicable for the multitone waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

– If there are tones with the same frequency, error -221,“Settings Conflict” will be generated and the duplicated tone with the same frequency will be removed.

– The record length value must be greater than or equal to the record length.

Examples

The following command sets the waveform length for channel 1 to 2048 points.

```
SOUR:MULT:WLEN L2048,(@1)
```

The following query returns the waveform length of channel 1.

```
SOUR:MULT:WLEN? (@1)
```

Typical response:

```
L2048
```

SOURce[:ANALog]:PHASe[:ADJust]

Syntax

```
SOURce:PHASe[:ADJust] <phase>,(@<channel_list>)
SOURce:PHASe[:ADJust]? (@<channel_list>)
```

Description

Sets the phase of the selected channel with reference to channel 1 in degree. The query returns the phase of the selected channel(s) in degree. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<phase>`	Numeric	–180° to 179.99°	0
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

- This setting is only applicable for the variable phase waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.
- Channel 1 is used as the reference channel and will always have the value of 0°. Therefore, this command is not applicable for channel 1.

Examples

The following command sets the phase for channel 2 to 100° with reference to channel 1.

```
SOUR:PHAS 100,(@2)
```

The following query returns the phase of channel 2 with reference to channel 1.

```
SOUR:PHAS? (@2)
```

Typical response:

```
1.000000E+02
```

SOURce[:ANALog]:REFerence:IMPedance

Syntax

```
SOURce[:ANALog]:REFerence:IMPedance <impedance>,(@<channel_list>)
SOURce[:ANALog]:REFerence:IMPedance? (@<channel_list>)
```

Description

Sets the generator reference impedance for the specified channel(s) in ohms (Ω). The reference impedance is used to set the amplitude value in unit dBm. The query returns the generator reference impedance of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<impedance>	Numeric	1.0E–12 < impedance ≤ 1.0E+09	600 Ω
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Examples

The following command sets the reference impedance to 600 Ω for channel 1 and channel 2.

```
SOUR:REF:IMP 600,(@1,2)
```

The following query returns the reference impedances for channel 1 and channel 2.

```
SOUR:REF:IMP? (@1,2)
```

Typical response:

```
6.000000E+02,6.000000E+02
```

SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate][:AMPLitude]

Syntax

```
SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate][:AMPLitude]
<voltage>[<unit>],(@<channel_list>)
```

```
SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate][:AMPLitude]?
(@<channel_list>)
```

Description

Sets the signal amplitude level for the specified channel(s). The query returns the amplitude of the selected channel(s) in Vrms. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<voltage>`	Numeric	Refer to "**Appendix D: Analog Waveform Amplitude Range**" on page 666.	0 Vrms
`<unit>`	Discrete	– V (DC signal) – Vrms|Vpp|Vp|dBV|dBm|dBu (other waveforms)	Vrms
`<tone_list>`	Discrete	One or more tones. – (1) for tone 1 – (1,2) for tone 1 and 2 – (1:10) for tone 1 through 10	Required parameter
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– The amplitude is dependent on the DC offset of the signal to be generated for a particular channel. When the DC offset and amplitude are added together, it must not exceed the maximum voltage. The relationship between the amplitude in Vp and DC offset is as follows.

  $Vp \leq Vmax - |Voffset|$

  where Vmax is the maximum voltage of the output connector. For the balanced output connection, the maximum voltage is 22.6 Vp, while for unbalanced and common, the maximum voltage is 11.3 Vp.

- This command is used to set the amplitude of the composite signal if the dual waveform is selected using the **SOURce[:ANALog]:FUNCtion** command. Use the **SOURce[:ANALog]:VOLTage:RATio** command to set the amplitude ratio of the second component over the first component.
- The allowable unit for the DC signal is only V. Error -131,"Invalid suffix" will be generated if other units have been selected for the DC signal.
- For all waveform types except DC, you can select either Vrms, Vpp, Vp, dBV, dBm, or dBu. Error -131,"Invalid suffix" will be generated if you have selected an invalid unit.
- You can also include a multiplier for the unit, for example, mVrms. The "em" is the multiplier for the unit Vrms.
- If the amplitude setting is invalid, the analog generator will automatically adjust the amplitude to the maximum value allowed with the specified DC offset. Error -222,"Data out of range" will be generated and the amplitude value will be clipped to the maximum value allowed.

Examples

The following command sequence sets the amplitude levels for channel 1 and channel 2 to 1 Vrms and 5 Vrms respectively.

```
SOUR:VOLT 1Vrms,(@1)
SOUR:VOLT 5Vrms,(@2)
```

The following query returns the amplitude levels of channel 1 and channel 2 in Vrms.

```
SOUR:VOLT? (@1,2)
```

Typical response:

```
1.000000E+00,5.000000E+00
```

SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate]:OFFSet

Syntax

```
SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate]:OFFSet
<voltage>[<unit>],(@<channel_list>)
```

```
SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate]:OFFSet?
(@<channel_list>)
```

Description

Sets the signal DC offset level in V for the specified channel(s). The query returns the DC offset of the selected channel(s) in V. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<voltage>`	Numeric	–11.3 V to 11.3 V	0 V
`<unit>`	Discrete	V	V
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– The DC offset is not applicable for the square, DC, and variable phase waveform types.

– The DC offset is dependent on the amplitude of the signal to be generated for a particular channel. When the DC offset and amplitude are added together, it must not exceed the maximum voltage. The relationship between the amplitude in Vp and DC offset is as follows.

  $Vp \leq Vmax - |Voffset|$

  where Vmax is the maximum voltage of the output connector. For the balanced output connection, the maximum voltage is 22.6 Vp, while for unbalanced and common, the maximum voltage is 11.3 Vp.

– If the specified DC offset is invalid, the analog generator will automatically adjust it to the maximum DC offset allowed with the specified amplitude. Error -222,"Data out of range" will be generated and the DC offset will be adjusted as described.

– You can also include a multiplier for the unit, for example, mV. The “em” is the multiplier for the unit V.

Examples

The following command sequence sets the DC offset for channel 1 and channel 2 to 1 V and 3.1 V respectively.

```
SOUR:VOLT:OFFS 1,(@1)
SOUR:VOLT:OFFS 3.1,(@2)
```

The following query returns the DC offset values of channel 1 and channel 2 in V.

```
SOUR:VOLT:OFFS? (@1,2)
```

Typical response:

```
1.000000E+00,3.100000E+00
```

SOURce[:ANALog]:VOLTage:RATio

Syntax

```
SOURce[:ANALog]:VOLTage:RATio <ratio>,(@<channel_list>)
SOURce[:ANALog]:VOLTage:RATio? (@<channel_list>)
```

Description

Sets the voltage ratio of the second component over the first component of the dual waveform for the specified channel(s) in percentage or dB. The query returns the amplitude ratio of the selected channel(s) in percentage. Multiple responses are separated by commas.

Parameter

Item	Type	Range of values	Default value
`<ratio>`	Discrete	0 to 100	100
`<channel_list>`	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is only applicable for the dual waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

Examples

The following command sequence sets the voltage ratio values for channel 1 and channel 2 to 1% and 10% respectively.

```
SOUR:VOLT:RAT 1,(@1)
SOUR:VOLT:RAT 10,(@2)
```

The following query returns the voltage ratio values of channel 1 and channel 2 in percentage.

```
SOUR:VOLT:RAT? (@1,2)
```

Typical response:

```
1.000E+00,1.000E+01
```

SOURce[:ANALog]:VOLTage:SUMMation

Syntax

```
SOURce[:ANALog]:VOLTage:SUMMation <value>, (@<channel_list>)
SOURce[:ANALog]:VOLTage:SUMMation? (@<channel_list>)
```

Description

Obtain or set the method for the dual tones waveform's level summation. The query command returns the method for the dual tones waveform's level summation. Multiple responses are separated by commas.

RSS	Amplitude shown is the Root Sum Square summation result of the 2 levels. The analyzer will read the same amplitude set at the generator regardless of the tone ratio.
LINear	Amplitude shown is the linear summation result of the 2 levels. The level of each tone can be calculated from the tone ratio.

Parameter

Item	Type	Range of values	Default value
<channel_list>	Numeric	One or more channels. – (@2) for channel 2 – (@1,4) for channel 1 and 4 – (@1:3) for channel 1 to 3	Required parameter
<value>	Discrete	RSS|LINear	RSS

Remarks

– Switch to the Dual, DFD, or SMPTE waveform types before sending this command.

– Use the **SOURce[:ANALog]:DTMF:VOLTage:SUMMation** command for the DTMF signal.

Examples

The following command sets the method for the level summation of the signal at channel 2 to RSS.

```
SOUR:VOLT:SUMM RSS, (@2)
```

The following query returns the method for the level summation of the signal for channel 2.

```
SOUR:VOLT:SUMM?(@2)
```

Typical response:

```
RSS
```

SOURce:DIGital:FUNCtion

Syntax

```
SOURce:DIGital:FUNCtion <waveform type>
SOURce:DIGital:FUNCtion?
```

Description

Sets the digital generator waveform type. The query returns the waveform type.

The waveform types with their corresponding `<waveform type>` parameters are listed as follows.

SINE	Sine waveform
SBURst	Sine burst waveform
STEReo	Stereo waveform
DUAL	Dual waveform
VPHase	Variable phase waveform
SMPTe11	SMPTE IMD 1 to 1 waveform
SMPTe41	SMPTE IMD 4 to 1 waveform
SMPTe101	SMPTE IMD 10 to 1 waveform
DFDiec118	DFD IEC 60118 waveform
DFDiec268	DFD IEC 60268 waveform
SQUare	Square waveform
WGAussian	Gaussian PDF white noise signal
WRECtangular	Rectangular PDF white noise signal
TNOise	Triangular statistic distribution noise signal
PNOise	Pink noise signal
CONStant	Constant value
MONotonicity	Monotonicity
WZERo	Walking zero
WONE	Walking one
MULTitone	Multitone waveform
ARBitrary	Arbitrary waveform

Parameter

Item	Type	Range of values	Default value
`<waveform type>`	Discrete	SINE|DUAL|STEReo|SBURst|VPHase|SMPTe11|SMPTe41|SMPTe101|DFDiec118|DFDiec268|SQUare|WGAussian|WRECtangular|TNOise|PNOise|CONStant|MONotonicity|WZERo|WONE|MULTitone|ARBitrary	SINE

Remarks

- If you change the waveform type, the particular parameter values of the previous waveform will be set to the default values of the current waveform.
- Refer to "**Appendix C: Waveform Parameters**" on page 664 for the configurable parameters of the corresponding waveform types.

Examples

The following command sets the waveform type to Square.

```
SOUR:DIG:FUNC SQU
```

The following query returns the waveform type.

```
SOUR:DIG:FUNC?
```

Typical response:

```
SQU
```

SOURce:DIGital:DITHer:TYPE

Syntax

```
SOURce:DIGital:DITHer:TYPE <dither>
```

```
SOURce:DIGital:DITHer:TYPE?
```

Description

Sets the dither type of the digital generator signal. The query returns the dither type of the digital generator signal.

Parameter

Item	Type	Range of values	Default value
<dither>	Discrete	OFF|TRIangular|RECTangular	OFF

Examples

The following commands set the dither type of the digital generator signal to triangular.

```
SOUR:DIG:DITH:TYPE TRI
```

The following query returns the dither type of the digital generator signal.

```
SOUR:DIG:DITH:TYPE?
```

Typical response:

```
TRI
```

SOURce:DIGital:VOLTage[:LEVel][:IMMediate][:AMPLitude]

Syntax

```
SOURce:DIGital:VOLTage[:LEVel][:IMMediate][:AMPLitude]
<voltage>[<unit>], (@<channel_list>)
```

```
SOURce:DIGital:VOLTage[:LEVel][:IMMediate][:AMPLitude]?
(@<channel_list>)
```

Description

Sets the signal amplitude level for the specified digital channel(s). The query returns the amplitude of the selected digital channel(s) in FFS. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<voltage>`	Numeric	0 to 1 FFS (–1 FFS to 1 FFS for constant value)	0.1 FFS
`<unit>`	Discrete	Vrms|Vpp|Vp|dBV|dBu|FFS|dBFS|pctFS	FFS
`<channel list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

– This command is not applicable for Monotonicity, Walking One, and Walking Zero waveform types.

– The amplitude is dependent on the DC offset of the signal. When the DC offset and amplitude are added together, it must not exceed the maximum voltage of 1 FFS. The relationship between the amplitude and DC offset is as follows.
Amplitude ≤1 FFS – |DC Offset|

– For Constant Value waveform, both channel 1 and 2 are sharing the same amplitude parameter. This means that when you change the amplitude value for channel 1, the amplitude for channel 2 will change accordingly to the same value and vice versa. Refer to “**Appendix E: Relationship between Digital Waveform Parameters and Channels**” on page 667 for more information.

– For Sine, Stereo, Square, Noise, and Arbitrary waveforms, the amplitude values for channel 1 and 2 can be different. These waveforms are sharing the same DC offset parameter for both its channels. So, when setting the amplitude, make sure that the added value for the DC offset and amplitude do not exceed the maximum voltage of 1 FFS at both channels.

- This command is used to set the amplitude of the composite signal if the dual waveform is selected. Use the **SOURce:DIGital:VOLTage:RATio** command to set the amplitude ratio of the second component over the first component.
- If the amplitude setting is invalid, the -222, "Data out of range" error will be generated and the amplitude value will be clipped to the maximum value allowed.
- If Constant Value waveform is selected, the unit allowed is V, FFS, dBFS, and pctFS. The -131,"Invalid suffix" error will be generated if you select other units.
- You can also include a multiplier for the unit. For example, mFFS. The 'm' is the multiplier for the unit FFS.

Examples

The following command sets the amplitude level for channel 1 to 0.5 FFS.

```
SOUR:DIG:VOLT 0.5FFS, (@D1)
```

The following query returns the amplitude level of channel 1 in FFS.

```
SOUR:DIG:VOLT? (@D1)
```

Typical response:

```
5.000000E-01
```

SOURce:DIGital:VOLTage[:LEVel][:IMMediate]:OFFSet

Syntax

```
SOURce:DIGital:VOLTage[:LEVel][:IMMediate]:OFFSet
<offset>[<unit>], (@<channel_list>)
```

```
SOURce:DIGital:VOLTage[:LEVel][:IMMediate]:OFFSet?
(@<channel_list>)
```

Description

Sets the signal DC offset level for the specified digital channel(s). The query returns the DC offset of the selected channel(s) in FFS. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<offset>`	Numeric	-1 to 1 FFS	
`<unit>`	Discrete	V|FFS|dBFS|pctFS	FFS
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remark

– This command is not applicable for Sine Burst, Variable Phase, Constant Value, Walking One, Walking Zero, and Multitone waveform.

– The DC offset parameter is channelless. This means both channel 1 and 2 will always have the same DC offset value. When a single channel is selected in the command, the setting is applied to both channel 1 and 2. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for more information.

– The DC offset is dependent on the amplitude of the signal to be generated for a particular channel. When the DC offset and amplitude are added together, it must not exceed the maximum voltage of 1 FFS. The relationship between the amplitude and DC offset is as follows.
Amplitude ≤1 FFS – |DC Offset|

– For Sine, Stereo, Square, Noise, and Arbitrary waveform, the amplitude values for channel 1 and 2 can be different. Both the channels at these waveforms are sharing the same DC offset parameter. When setting the DC offset, make sure that the added value for the DC offset and amplitude do not exceed the maximum voltage of 1 FFS at both channels.

– If the DC offset setting is invalid, the -222, "Data out of range" error will be generated and the DC offset value will be clipped to the maximum value allowed.
– You can also include a multiplier for the unit. For example, mFFS. The 'm' is the multiplier for the unit FFS.

Examples

The following command sets the DC offset for channel 1 to 0.1 FFS.

```
SOUR:DIG:VOLT:OFFS 0.1, (@D1)
```

The following query returns the DC offset values of channel 1 in FFS.

```
SOUR:DIG:VOLT:OFFS? (@D1)
```

Typical response:

```
1.000000E-01
```

SOURce:DIGital:VOLTage:RATio

Syntax

```
SOURce:DIGital:VOLTage:RATio <ratio>,(@<channel_list>)
SOURce:DIGital:VOLTage:RATio? (@<channel_list>)
```

Description

Sets the voltage ratio of the second component over the first component of the dual waveform for the specified digital channel(s) in percentage. The query returns the voltage ratio of the selected channel(s) in percentage. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<ratio>	Numeric	0 to 100%	100
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- This setting is only applicable for the dual waveform. Use the **SOURce:DIGital:FUNCtion** command to select the dual waveform type.
- The ratio parameter is not dependent on the channels. This means that both channel 1 and 2 will always have the same ratio value. If you only select a single channel in the command, the setting is applied to both channel 1 and 2. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between the ratio parameter and channels.

Examples

The following command sets the voltage ratio value for channel 1 to 50%.

```
SOUR:DIG:VOLT:RAT 50, (@D1)
```

The following query returns the voltage ratio values of channel 1 in percentage.

```
SOUR:DIG:VOLT:RAT? (@D1)
```

Typical response:

```
5.000E+01
```

SOURce:DIGital:VOLTage:SUMMation

Syntax

```
SOURce:DIGital:VOLTage:SUMMation <value>, (@<channel_list>)
SOURce:DIGital:VOLTage:SUMMation? (@<channel_list>)
```

Description

Obtain or set the method for the dual tones waveform's level summation. The query command returns the method for the dual tones waveform's level summation. Multiple responses are separated by commas.

RSS	Amplitude shown is the Root Sum Square summation result of the 2 levels. The analyzer will read the same amplitude set at the generator regardless of the tone ratio.
LINear	Amplitude shown is the linear summation result of the 2 levels. The level of each tone can be calculated from the tone ratio.

Parameter

Item	Type	Range of values	Default value
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter
<value>	Discrete	RSS|LINear	RSS

Remarks

– Switch to the Dual, DFD, or SMPTE waveform types before sending this command.

Examples

The following command sets the method for the level summation of the signal at channel 2 to RSS.

```
SOUR:DIG:VOLT:SUMM RSS, (@D2)
```

The following query returns the method for the level summation of the signal for channel 2.

```
SOUR:DIG:VOLT:SUMM?(@D2)
```

Typical response:

```
RSS
```

SOURce:DIGital:FREQuency[<j>]

Syntax

```
SOURce:DIGital:FREQuency[<j>]<frequency>[<unit>],
(@<channel_list>)
```

```
SOURce:DIGital:FREQuency[<j>]? (@<channel_list>)
```

Description

Sets the signal frequency for the specified channel(s) in Hz. The query returns the frequency of the selected channel(s) in Hz. Multiple responses are separated by commas.

Parameter

Item	Type	Range of values	Default value
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Refer to "**Appendix A: Waveform Frequency Range and Default Values**" on page 660 for the `<frequency>` parameter.

Remarks

- The <j> parameter represents 1 or 2.
  - For the SMPTE IMD 1:1, 4:1, and 10:1 waveforms, **SOURce:DIGital:FREQuency:LOWer** represents the lower frequency while **SOURce:DIGital:FREQuency:UPPer** represents the upper frequency.
  - For the dual waveform, `SOURce:DIGital:FREQuency1` represents the frequency of the first sine component while `SOURce:DIGital:FREQuency2` represents the frequency of the second sine component.
  - For the DFD IEC 60118 waveform, use **SOURce:DIGital:FREQuency:UPPer** to set the upper frequency and the **SOURce:DIGital:FREQuency:DIFFerence** command to set the frequency difference.
  - For the DFD IEC 60268 waveform, use the **SOURce:DIGital:FREQuency:DIFFerence** command to set the frequency difference and the **SOURce:DIGital:FREQuency:CENTer** command to set the center frequency.

- The frequency setting is not applicable for the constant value, noise, monotonicity, walking one, walking zero, multitone, and arbitrary waveforms.
- You can also include a multiplier for the unit, for example, kHz. The 'k' is the multiplier for the unit Hz.
- Refer to ““**Appendix E: Relationship between Digital Waveform Parameters and Channels**” on page 667 for the relationship between the frequency parameter and channels.

Examples

The following command sets the sine waveform frequency for channel 1 to 5 kHz.

```
SOUR:DIG:FREQ 5000, (@D1)
```

The following query returns the frequency value of channel 1 in Hz.

```
SOUR:DIG:FREQ? (@D1)
```

Typical response:

```
5.000000E+03
```

SOURce:DIGital:FREQuency:CENTer

Syntax

```
SOURce:DIGital:FREQuency:CENTer<frequency>[<unit>],
(@<channel_list>)
```

```
SOURce:DIGital:FREQuency:CENTer?(@<channel_list>)
```

Description

Sets the center frequency of the DFD IEC 60268 waveform for the specified channel(s) in Hz. The center frequency determines the frequency for the two tones of the DFD IEC 60268 signal are spaced in equal increments above and below. The query returns the center frequency value of the selected channel(s) in Hz. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<frequency>`	Numeric	3 kHz to 79 kHz	10 kHz
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remark

– This setting is only applicable for the DFD IEC 60268 waveform. Use the SOURce:DIGital:FUNCtion command to select the DFD IEC 60268 waveform type.

– You can also include a multiplier for the unit, for example, kHz. The 'k' is the multiplier for the unit Hz.

– The center frequency parameter is not dependent on the channels. This means that both channel 1 and 2 will always have the same center frequency value. If you only select a single channel in the command, the setting is applied to both channel 1 and 2. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between the center frequency parameter and channels.

Examples

The following command sets the center frequency value for channel 1 and 2 to 5 kHz.

```
SOUR:DIG:FREQ:CENT 5kHz, (@D1,D2)
```

The following query returns the center frequency values of channel 1 and 2 in Hz.

```
SOUR:DIG:FREQ:CENT? (@D1,D2)
```

Typical response:

```
5.000000E+03,5.000000E+03
```

SOURce:DIGital:FREQuency:DIFFerence

Syntax

```
SOURce:DIGital:FREQuency:DIFFerence<frequency>[<unit>],
(@<channel_list>)
```

```
SOURce:DIGital:FREQuency:DIFFerence?(@<channel_list>)
```

Description

Sets the frequency difference of the DFD IEC 60268 and DFD IEC 60118 waveforms for the specified channel(s) in Hz. The frequency difference determines the difference frequency (spacing) between the two tones of the DFD IEC 60268 and DFD IEC 60118 signals. The query returns the frequency difference of the selected channel(s) in Hz. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<frequency>`	Numeric	80 Hz to 2 kHz	80 Hz
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

– This setting is only applicable for the DFD IEC 60118 and DFD IEC 60268 waveforms. Use the **SOURce:DIGital:FUNCtion** command to select either one of these two waveform types.

– You can also include a multiplier for the unit, for example, kHz. The 'k' is the multiplier for the unit Hz.

– The difference frequency parameter is not dependent on the channels. This means that both channel 1 and 2 will always have the same difference frequency value. If you only select a single channel in the command, the setting is applied to both channel 1 and 2. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between the difference frequency parameter and channels.

Examples

The following command sets the frequency difference value for channel 1 and 2 to 100 Hz.

```
SOUR:DIG:FREQ:DIFF 100Hz, (@D1,D2)
```

The following query returns the frequency difference values of channel 1 and 2 in Hz.

```
SOUR:DIG:FREQ:DIFF? (@D1,D2)
```

Typical response:

```
1.000000E+02,1.000000E+02
```

SOURce:DIGital:FREQuency:UPPer

Syntax

```
SOURce:DIGital:FREQuency:UPPer<frequency>[<unit>],
(@<channel_list>)
```

```
SOURce:DIGital:FREQuency:UPPer?(@<channel_list>)
```

Description

Sets the upper frequency of the DFD IEC 60118, SMPTE 1:1, SMPTE 4:1, and SMPTE 10:1 waveforms for the specified channel(s) in Hz. The upper frequency determines the frequency of the higher frequency tone in the two-tone waveform. The query returns the upper frequency of the selected channel(s) in Hz. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<frequency>`	Numeric	Refer to "**Appendix A: Waveform Frequency Range and Default Values**" on page 660	3 kHz
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

– This setting is only applicable for the DFD IEC 60118, SMPTE 1:1, SMPTE 4:1, and SMPTE 10:1 waveforms. Use the **SOURce:DIGital:FUNCtion** command to select either one of these four waveform types.

– You can also include a multiplier for the unit, for example, kHz. The 'k' is the multiplier for the unit Hz.

– The upper frequency parameter is not dependent on the channels. This means that both channel 1 and 2 will always have the same upper frequency value. If you only select a single channel in the command, the setting is applied to both channel 1 and 2. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between the upper frequency parameter and channels.

Examples

The following command sets the upper frequency values for channel 1 and 2 to 3 kHz.

```
SOUR:DIG:FREQ:UPP 3kHz, (@D1,D2)
```

The following query returns the upper frequency values of channel 1 and 2 in Hz.

```
SOUR:DIG:FREQ:UPP? (@D1,D2)
```

Typical response:

```
3.000000E+03,3.000000E+03
```

SOURce:DIGital:FREQuency:LOWer

Syntax

```
SOURce:DIGital:FREQuency:LOWer<frequency>[<unit>],
(@<channel_list>)
```

```
SOURce:DIGital:FREQuency:LOWer?(@<channel_list>)
```

Description

Sets the lower frequency of the SMPTE 1:1, SMPTE 4:1, and SMPTE 10:1 waveforms for the specified channel(s) in Hz. The lower frequency determines the frequency of the lower frequency tone in the two-tone waveform. The query returns the lower frequency of the selected channel(s) in Hz. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<frequency>`	Numeric	40 Hz to 500 Hz	60 Hz
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

– This setting is only applicable for the SMPTE 1:1, SMPTE 4:1, and SMPTE 10:1 waveforms. Use the **SOURce:DIGital:FUNCtion** command to select either one of these three waveform types.

– You can also include a multiplier for the unit, for example, kHz. The 'k' is the multiplier for the unit Hz.

– The lower frequency parameter is not dependent on the channels. This means that both channel 1 and 2 will always have the same lower frequency value. If you only select a single channel in the command, the setting is applied to both channel 1 and 2. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between the lower frequency parameter and channels.

Examples

The following command sets the lower frequency value for channel 1 and 2 to 80 Hz.

```
SOUR:DIG:FREQ:LOW 80Hz, (@D1,D2)
```

The following query returns the lower frequency values of channel 1 and 2 in Hz.

```
SOUR:DIG:FREQ:LOW? (@D1,D2)
```

Typical response:

```
8.000000E+01,8.000000E+01
```

SOURce:DIGital:SBURst:ONTime

Syntax

```
SOURce:DIGital:SBURst:ONTime <on_time>, (@<channel_list>)
SOURce:DIGital:SBURst:ONTime? (@<channel_list>)
```

Description

Sets the burst on time of the sine burst waveform for the specified channel(s) in number of cycles. The burst on time determines the number of cycles at which the amplitude is at the highest level. The query returns the burst on time of the selected channel(s) in number of cycles. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<on_time>`	Numeric	1 to 65534	1
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

– This setting is only applicable for the sine burst waveform. Use the **SOURce:DIGital:FUNCtion** command to select the waveform type.

– The burst on time must be smaller than the burst period for the sine burst waveform.

– The burst on time parameter is not dependent on the channels. This means that both channel 1 and 2 will always have the same burst on time value. If you only select a single channel in the command, the setting is applied to both channel 1 and 2. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between the burst on time parameter and channels.

Examples

The following command sets the sine burst on time for channel 1 to 1000 cycles.

```
SOUR:DIG:SBUR:ONT 1000, (@D1)
```

The following query returns the sine burst on time of channel 1 in number of cycles.

```
SOUR:DIG:SBUR:ONT? (@D1)
```

Typical response:

```
1000
```

SOURce:DIGital:SBURst:PERiod

Syntax

```
SOURce:DIGital:SBURst:PERiod <period>,(@<channel_list>)
SOURce:DIGital:SBURst:PERiod? (@<channel_list>)
```

Description

Sets the burst period of the sine burst waveform for the specified channel(s) in number of cycles. The burst period determines the number of cycles from the beginning of one burst to the beginning of the next burst. The query returns the burst period of the selected channel(s) in number of cycles. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<period>	Numeric	2 to 65535	3
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

– This setting is only applicable for the sine burst waveform. Use the **SOURce:DIGital:FUNCtion** command to select the waveform type.

– The burst period must be greater than the burst on time for the sine burst waveform.

– The burst period parameter is not dependent on the channels. This means that both channel 1 and 2 will always have the same burst period value. If you only select a single channel in the command, the setting is applied to both channel 1 and 2. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between the burst period parameter and channels.

Examples

The following command sets the sine burst period for channel 1 to 5000 cycles.

```
SOUR:DIG:SBUR:PER 5000, (@D1)
```

The following query returns the sine burst period of channel 1 in number of cycles.

```
SOUR:DIG:SBUR:PER? (@D1)
```

Typical response:

```
5000
```

SOURce:DIGital:SBURst:LOWLevel

Syntax

```
SOURce:DIGital:SBURst:LOWLevel <low_level>,(@<channel_list>)
SOURce:DIGital:SBURst:LOWLevel?(@<channel list>)
```

Description

Sets the amplitude ratio of burst on to burst off for the sine burst waveform for the specified channel(s) in percentage. The query returns the low level of the selected channel(s) in percentage. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<low_level>`	Numeric	0 to 100 %	50 %
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- This setting is only applicable for the sine burst waveform. Use the **SOURce:DIGital:FUNCtion** command to select the waveform type.
- The low level parameter is not dependent on the channels. This means that both channel 1 and 2 will always have the same low level value. If you only select a single channel in the command, the setting is applied to both channel 1 and 2. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between the low level parameter and channels.

Examples

The following command sets the sine burst low level for channel 1 to 50%.

```
SOUR:DIG:SBUR:LOWL 50, (@D1)
```

The following query returns the sine burst low level of channel 1 in percentage.

```
SOUR:DIG:SBUR:LOWL? (@D1)
```

Typical response:

```
5.000000E+01
```

SOURce:DIGital:SAMPle

Syntax

```
SOURce:DIGital:SAMPle <sample>,(@<channel_list>)
SOURce:DIGital:SAMPle? (@<channel_list>)
```

Description

Sets the samples per step of the Monotonicity, Walking Zero, and Walking One waveform for the specified channel(s). The query returns the samples per step of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<sample>`	Numeric	1 to 32768 (Monotonicity) 1 to 65535 (Walking Zero and Walking One)	1 (Monotonicity) 48000 (Walking Zero and Walking One)
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

– This setting is only applicable for the Monotonicty, Walking Zero, and Walking One waveform. Use the **SOURce:DIGital:FUNCtion** command to select either one of these three waveform types.

– The samples per step parameter is not dependent on the channels. This means that both channel 1 and 2 will always have the same samples per step value. If you only select a single channel in the command, the setting is applied to both channel 1 and 2. Refer to “**Appendix E: Relationship between Digital Waveform Parameters and Channels**” on page 667 for the relationship between the samples per step parameter and channels.

Examples

The following command sets the samples per step for channel 1 to 1000 steps.

```
SOUR:DIG:SAMP 1000, (@D1)
```

The following query returns the samples per step of channel 1.

```
SOUR:DIG:SAMP? (@D1)
```

Typical response:

```
1000
```

SOURce:DIGitaL:PHASe[:ADJust]

Syntax

```
SOURce:DIGital:PHASe[:ADJust] <phase>,(@<channel_list>)
SOURce:DIGital:PHASe[:ADJust]? (@<channel_list>)
```

Description

Sets the phase offset value of the selected channel with reference to the digital generator channel 1 in degree. The query returns the phase of the selected channel(s) in degree. Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<phase>`	Numeric	–180 ° to 179.99 °	0°
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- This setting is only applicable for the variable phase waveform. Use the **SOURce:DIGital:FUNCtion** command to select the variable phase waveform type.
- Channel 1 is used as reference channel and will always have the value of 0°. Therefore, this command is not applicable for channel 1.

Examples

The following command sets the phase for channel 2 to 100 degrees with reference to channel 1.

```
SOUR:DIG:PHAS 100, (@D2)
```

The following query returns the phase of channel 2 with reference to channel 1.

```
SOUR:DIG:PHAS? (@D2)
```

Typical response:

```
1.000000E+02
```

SOURce:DIGital:MULTitone:FREQuency:STARt

Syntax

```
SOURce:DIGital:MULTitone:FREQuency:STARt <frequency>[<unit>],
(@<channel_list>)
```

```
SOURce:DIGital:MULTitone:FREQuency:STARt? (@<channel_list>)
```

Description

Sets the start frequency of the multitone waveform for the specified channel(s). The start frequency defines the lowest tone frequency in the multitone waveform. The query returns the start frequency of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<frequency>`	Numeric	2 Hz to (0.45 x sampling rate) Hz	1 kHz
`<channel list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- This setting is only applicable for the multitone waveform. Use the **SOURce:DIGital:FUNCtion** command to select the multitone waveform type.
- You can also include a multiplier for the unit, for example, kHz. The 'k' is the multiplier for the unit Hz.
- The start frequency parameter is channel based. This means that channel 1 and 2 can have different start frequency values. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between start frequency parameter and channels.
- If there are tones with the same frequency, the –221,"Settings Conflict" error will be generated and the duplicated tone with the same frequency will be removed.
- You can verify whether a generator setup has completed by polling the status register value via the **STATus:OPERation:CONDition?** query. While the generator setup is in progress, bit 8 of the condition register of the Standard Operation register group will be set. After the generator setup has completed, bit 8 will be cleared to 0.

Examples

The following command sets the start frequency value for channel 1 to 5 kHz.

```
SOUR:DIG:MULT:FREQ:STAR 5000, (@D1)
```

The following query returns the start frequency value of channels 1 in Hz.

```
SOUR:DIG:MULT:FREQ:STAR? (@D1)
```

Typical response:

```
5.000000E+03
```

SOURce:DIGital:MULTitone:FREQuency:STOP

Syntax

```
SOURce:DIGital:MULTitone:FREQuency:STOP<frequency>[<unit>],
(@<channel_list>)

SOURce:DIGital:MULTitone:FREQuency:STOP?(@<channel_list>)
```

Description

Sets the stop frequency of the multitone waveform for the specified channel(s). The stop frequency defines the highest tone frequency in the multitone waveform. The query returns the stop frequency of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<frequency>`	Numeric	2 Hz to (0.45 x sampling rate) Hz	5 kHz
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- This setting is only applicable for the multitone waveform. Use the **SOURce:DIGital:FUNCtion** command to select the multitone waveform type.
- You can also include a multiplier for the unit, for example, kHz. The 'k' is the multiplier for the unit Hz.
- The stop frequency parameter is channel based. This means that channel 1 and 2 can have different stop frequency values. Refer to “**Appendix E: Relationship between Digital Waveform Parameters and Channels**” on page 667 for the relationship between stop frequency parameter and channels.
- If there are tones with the same frequency, the –221,"Settings Conflict" error will be generated and the duplicated tone with the same frequency will be removed.
- You can verify whether a generator setup has completed by polling the status register value via the **STATus:OPERation:CONDition?** query. While the generator setup is in progress, bit 8 of the condition register of the Standard Operation register group will be set. After the generator setup has completed, bit 8 will be cleared to 0.

Examples

The following command sets the stop frequency value for channel 1 to 10 kHz.

```
SOUR:DIG:MULT:FREQ:STOP 10kHz, (@D1)
```

The following query returns the stop frequency value of channels 1 in Hz.

```
SOUR:DIG:MULT:FREQ:STOP? (@D1)
```

Typical response:

```
1.000000E+04
```

SOURce:DIGital:MULTitone:FREQuency:SPACing

Syntax

```
SOURce:DIGital:MULTitone:FREQuency:SPACing<spacing>,
(@<channel_list>)
```

```
SOURce:DIGital:MULTitone:FREQuency:SPACing?(@<channel_list>)
```

Description

Sets the frequency spacing type between the start and stop frequency of the multitone waveform for the specified channel(s). The query returns the frequency spacing type of the selected channel(s). Multiple responses are separated by commas.

LINear	Linear frequency spacing
LOG	Logarithmic frequency spacing

Parameters

Item	Type	Range of values	Default value
<spacing>	Discrete	LINear|LOG	LINear
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- This setting is only applicable for the multitone waveform. Use the **SOURce:DIGital:FUNCtion** command to select the multitone waveform type.
- The frequency spacing parameter is channel based. This means that channel 1 and 2 can have different frequency spacing values. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between frequency spacing parameter and channels.
- If there are tones with the same frequency, the –221,"Settings Conflict" error will be generated and the duplicated tone with the same frequency will be removed.
- You can verify whether a generator setup has completed by polling the status register value via the **STATus:OPERation:CONDition?** query. While the generator setup is in progress, bit 8 of the condition register of the Standard Operation register group will be set. After the generator setup has completed, bit 8 will be cleared to 0.

Examples

The following command sets the frequency spacing for channel 1 to Log.

```
SOUR:DIG:MULT:FREQ:SPAC LOG, (@D1)
```

The following query returns the frequency spacing type of channel 1.

```
SOUR:DIG:MULT:FREQ:SPAC? (@D1)
```

Typical response:

```
LOG
```

SOURce:DIGital:MULTitone:COUNt

Syntax

```
SOURce:DIGital:MULTitone:COUNt <tone_count>,(@<channel_list>)
SOURce:DIGital:MULTitone:COUNt?(@<channel_list>)
```

Description

Sets the tones of the multitone waveform for the specified channel(s). Tones refer to the number of signal frequency components. The query returns the number of tones of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
<tone_count>	Numeric	2 to 64	2
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- This setting is only applicable for the multitone waveform. Use the **SOURce:DIGital:FUNCtion** command to select the multitone waveform type.
- The tone count parameter is channel based. This means that channel 1 and 2 can have different tone count values. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between tone count parameter and channels.
- If there are tones with the same frequency, the –221,"Settings Conflict" error will be generated and the duplicated tone with the same frequency will be removed.
- You can verify whether a generator setup has completed by polling the status register value via the **STATus:OPERation:CONDition?** query. While the generator setup is in progress, bit 8 of the condition register of the Standard Operation register group will be set. After the generator setup has completed, bit 8 will be cleared to 0.

Examples

The following commands set the tones for channel 1 and 2 to 3 and 15 respectively.

```
SOUR:DIG:MULT:COUN 3, (@D1)
```

```
SOUR:DIG:MULT:COUN 15, (@D2)
```

The following query returns the number of tones of channel 1 and 2.

```
SOUR:DIG:MULT:COUN? (@D1,D2)
```

Typical response:

```
3,15
```

SOURce:DIGital:MULTitone:WLEN

Syntax

```
SOURce:DIGital:MULTitone:WLEN <length>,(@<channel_list>)
SOURce:DIGital:MULTitone:WLEN? (@<channel_list>)
```

Description

Sets the waveform length of the multitone waveform for the specified channel(s). The waveform length determines the number of samples used to create one iteration of the multitone waveform. Longer waveform length provides higher frequency resolution but take more time in generation and processing. The query returns the waveform length of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<length>`	Discrete	L1024|L2048|L4096|L8192|L16384|L32768	L1024
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- This setting is only applicable for the multitone waveform. Use the **SOURce:DIGital:FUNCtion** command to select the multitone waveform type.
- The waveform length value must be less than or equal to the record length.
- If there are tones with the same frequency, the –221,"Settings Conflict" error will be generated and the duplicated tone with the same frequency will be removed.
- You can verify whether a generator setup has completed by polling the status register value via the **STATus:OPERation:CONDition?** query. While the generator setup is in progress, bit 8 of the condition register of the Standard Operation register group will be set. After the generator setup has completed, bit 8 will be cleared to 0.

Examples

The following command sets the waveform length for channel 1 to 2048 points.

```
SOUR:DIG:MULT:WLEN L2048, (@D1)
```

The following query returns the waveform length of channel 1.

```
SOUR:DIG:MULT:WLEN? (@D1)
```

Typical response:

```
L2048
```

SOURce:DIGital:MULTitone:CRESt?

Syntax

```
SOURce:DIGital:MULTitone:CRESt?(@<channel_list>)
```

Description

Queries the crest factor of the multitone waveform for the selected channel(s). Multiple responses are separated by commas.

Parameter

Item	Type	Range of values	Default value
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- This setting is only applicable for the multitone waveform. Use the **SOURce:DIGital:FUNCtion** command to select the multitone waveform type.
- The crest factor is channel based. This means that channel 1 and 2 can have different crest factor values. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between crest factor and channels.
- You can verify whether a generator setup has completed by polling the status register value via the **STATus:OPERation:CONDition?** query. While the generator setup is in progress, bit 8 of the condition register of the Standard Operation register group will be set. After the generator setup has completed, bit 8 will be cleared to 0.

Examples

The following query returns the crest factor of multitone waveform for channel 1.

```
SOUR:DIG:MULT:CRESt? (@D1)
```

Typical response:

```
1.4142000E+0
```

SOURce:DIGital:MULTitone:OPTimize

Syntax

```
SOURce:DIGital:MULTitone:OPTimize (@<channel_list>)
```

Description

Optimizes all the tones of the multitone signal for the embedded audio signal.

Parameter

Item	Type	Range of values	Default value
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- This setting is only applicable for the multitone waveform. Use the **SOURce:DIGital:FUNCtion** command to select the multitone waveform type.
- The crest factor is channel based. This means that channel 1 and 2 can have different crest factor values. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between crest factor and channels.
- All the existing tones phase and crest factor will be optimized.
- When optimizating the maximum number of tones (60) and waveform length (32768), the measured time for the completion of the optimization is approximately 8 to 10 seconds.
- You can verify whether a generator setup has completed by polling the status register value via the **STATus:OPERation:CONDition?** query. While the generator setup is in progress, bit 8 of the condition register of the Standard Operation register group will be set. After the generator setup has completed, bit 8 will be cleared to 0.

Examples

The following command optimizes all the tones of multitone signal for channel 1.

```
SOURce:DIGital:MULTitone:OPTimize (@D1)
```

SOURce:DIGital:MULTitone:TONE:CLEar

Syntax

```
SOURce:DIGital:MULTitone:TONE:CLEar (@<channel_list>)
```

Description

Clears all the tones of the multitone waveform for the specified channel(s).

Parameter

Item	Type	Range of values	Default value
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- This setting is only applicable for the multitone waveform. Use the **SOURce:DIGital:FUNCtion** command to select the multitone waveform type.
- After all the tones are cleared, an initial tone with 1 kHz frequency, 1 FFS amplitude, and 0° will be the default.
- The command is channel based. This means that the command can be applied differently to channel 1 and 2. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between this command and channels.
- You can verify whether a generator setup has completed by polling the status register value via the **STATus:OPERation:CONDition?** query. While the generator setup is in progress, bit 8 of the condition register of the Standard Operation register group will be set. After the generator setup has completed, bit 8 will be cleared to 0.

Example

The following command clears all the tones of the multitone waveform for channel 1.

```
SOUR:DIG:MULT:TONE:CLE (@D1)
```

SOURce:DIGital:MULTitone:TONE:ADD

Syntax

```
SOURce:DIGital:MULTitone:TONE:ADD <index>, <frequency>[<unit>],
<voltage>[<unit>], <phase>, (@<channel_list>)
```

Description

Adds a customized tone into the multitone waveform for the specified channel(s).

Parameters

Item	Type	Range of values	Default value
`<index>`	Numeric	0 to 63	0
`<frequency>`	Numeric	2 Hz to (0.45 x sampling rate) Hz	1 kHz
`<voltage>`	Numeric	0 to 1 FFS	1 FFS
`<unit>`	Discrete	Vrms|Vpp|Vp|dBV|dBu|FFS|dBFS|pctFS	FFS
`<phase>`	Numeric	–180 to ~179.99	0
`<channel list>`	Discrete	One or more channels. – (@D1) or (@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- This setting is only applicable for the multitone waveform. Use the **SOURce:DIGital:FUNCtion** command to select the multitone waveform type.
- This command will add a customized tone into the existing multitone waveform. The position of the added tone will be determined by the index. The index must be less than the total number of tones.
- The command is channel based. This means that the command can be applied differently to channel 1 and 2. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between this command and channels.
- If there are tones with the same frequency, the –221,"Settings Conflict" error will be generated and the duplicated tone with the same frequency will be removed.
- You can verify whether a generator setup has completed by polling the status register value via the **STATus:OPERation:CONDition?** query. While the generator setup is in

progress, bit 8 of the condition register of the Standard Operation register group will be set. After the generator setup has completed, bit 8 will be cleared to 0.

Examples

The following command adds a 5 kHz frequency, 0.5 FFS voltage, and 80° phase tone to the current multitone waveform for channel 1 at position 5.

```
SOUR:DIG:MULT:TONE:ADD 4, 5000, 0.5, 80, (@D1)
```

SOURce:DIGital:MULTitone:TONE:DELete

Syntax

```
SOURce:DIGital:MULTitone:TONE:DELete <index>,(@<channel_list>)
```

Description

Deletes a specific tone from the multitone waveform for the specified channel(s).

Parameters

Item	Type	Range of values	Default value
`<index>`	Numeric	0 to 63	0
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- This setting is only applicable for the multitone waveform. Use the **SOURce:DIGital:FUNCtion** command to select the multitone waveform type.
- This command will delete a tone from the existing multitone waveform. The deleted tone will be determined by the index. The index must be less than the total number of tones.
- The command is channel based. This means that the command can be applied differently to channel 1 and 2. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between this command and channels.
- You can verify whether a generator setup has completed by polling the status register value via the **STATus:OPERation:CONDition?** query. While the generator setup is in progress, bit 8 of the condition register of the Standard Operation register group will be set. After the generator setup has completed, bit 8 will be cleared to 0.

Examples

The following command deletes the tone 5 of the multitone waveform for channel 1.

```
SOUR:DIG:MULT:TONE:DEL 4, (@D1)
```

SOURce:DIGital:MULTitone:TONE:FREQuency

Syntax

```
SOURce:DIGital:MULTitone:TONE:FREQuency<frequency>[<unit>],
(<tone_list>),(@<channel_list>)
```

```
SOURce:DIGital:MULTitone:TONE:FREQuency? (<tone_list>),
(@<channel_list>)
```

Description

Sets the frequency of the tone(s) in the multitone waveform for the specified channel(s). The query returns the frequency of the tone(s) of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<frequency>`	Numeric	2 Hz to (0.45 x sampling rate) Hz	1 kHz
`<tone list>`	Discrete	One or more tones. – (1) for tone 1 – (1,2) for tone 1 and 2 – (1:10) for tone 1 through 10	Required parameter
`<channel list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

– This setting is only applicable for the multitone waveform. Use the **SOURce:DIGital:FUNCtion** command to select the multitone waveform type.

– You can also include a multiplier for the unit, for example, kHz. The 'k' is the multiplier for the unit Hz.

– The command is channel based. This means that the command can be applied differently to channel 1 and 2. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between this command and channels.

– If there are tones with the same frequency, the –221,"Settings Conflict" error will be generated and the duplicated tone with the same frequency will be removed.

– You can verify whether a generator setup has completed by polling the status register value via the **STATus:OPERation:CONDition?** query. While the generator setup is in progress, bit 8 of the condition register of the Standard Operation register group will be set. After the generator setup has completed, bit 8 will be cleared to 0.

Examples

The following command sets the frequency of the tone 5 for channel 1 to 2 kHz.

```
SOUR:DIG:MULT:TONE:FREQ 2kHz, (5), (@D1)
```

The following query returns the frequency of tone 5 and tone 6 for channel 1.

```
SOUR:DIG:MULT:TONE:FREQ? (5,6), (@D1)
```

Typical response:

```
2.000000E+03,3.000000E+03
```

SOURce:DIGital:MULTitone:TONE:VOLTage

Syntax

```
SOURce:DIGital:MULTitone:TONE:VOLTage<voltage>[<unit>],
(<tone_list>),(@<channel_list>)
```

```
SOURce:DIGital:MULTitone:TONE:VOLTage?
(<tone_list>),(@<channel_list>)
```

Description

Sets the voltage of the tone(s) in the multitone waveform for the specified channel(s). The query returns the voltage of the tone(s) of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<voltage>`	Numeric	0 to 1 FFS	1 FFS
`<unit>`	Discrete	Vrms|Vpp|Vp|dBV|dBu|FFS|dBFS|pctFS	FFS
`<tone_list>`	Discrete	One or more tones. – (1) for tone 1 – (1,2) for tone 1 and 2 – (1:10) for tone 1 through 10	Required parameter
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

– This setting is only applicable for the multitone waveform. Use the **SOURce:DIGital:FUNCtion** command to select the multitone waveform type.

– You can also include a multiplier for the unit, for example, mFFS. The 'm' is the multiplier for the unit FFS.

– The command is channel based. This means that the command can be applied differently to channel 1 and 2. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between this command and channels.

– You can verify whether a generator setup has completed by polling the status register value via the **STATus:OPERation:CONDition?** query. While the generator setup is in progress, bit 8 of the condition register of the Standard Operation register group will be set. After the generator setup has completed, bit 8 will be cleared to 0.

Examples

The following command sets the voltage of the tone 5 and tone 6 for channel 1 to 0.5 FFS.

```
SOUR:DIG:MULT:TONE:VOLT 0.5, (5,6), (@D1)
```

The following query returns the voltage of tone 5 and tone 6 for channel 1.

```
SOUR:DIG:MULT:TONE:VOLT? (5,6), (@D1)
```

Typical response:

```
5.000000E-01,5.000000E-01
```

SOURce:DIGital:MULTitone:TONE:PHASe

Syntax

```
SOURce:DIGital:MULTitone:TONE:PHASe <phase>,(<tone_list>),
(@<channel_list>)
```

```
SOURce:DIGital:MULTitone:TONE:PHASe? (<tone_list>),
(@<channel_list>)
```

Description

Sets the phase of the tone(s) in the multitone waveform for the specified channel(s). The query returns the phase of the tone(s) of the selected channel(s). Multiple responses are separated by commas.

Parameters

Item	Type	Range of values	Default value
`<phase>`	Numeric	180 to ~ 179,99	0
`<tone_list>`	Discrete	One or more tones. – (1) for tone 1 – (1,2) for tone 1 and 2 – (1:10) for tone 1 through 10	Required parameter
`<channel_list>`	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- This setting is only applicable for the multitone waveform. Use the **SOURce:DIGital:FUNCtion** command to select the multitone waveform type.
- The command is channel based. This means that the command can be applied differently to channel 1 and 2. Refer to "**Appendix E: Relationship between Digital Waveform Parameters and Channels**" on page 667 for the relationship between this command and channels.
- You can verify whether a generator setup has completed by polling the status register value via the **STATus:OPERation:CONDition?** query. While the generator setup is in progress, bit 8 of the condition register of the Standard Operation register group will be set. After the generator setup has completed, bit 8 will be cleared to 0.

Examples

The following command sets the phase of the tone 5 and tone 6 for channel 1 to 90°.

```
SOUR:DIG:MULT:TONE:PHAS 90, (5,6), (@D1)
```

The following query returns the phase of tone 5 and tone 6 for channel 1.

```
SOUR:DIG:MULT:TONE:PHAS? (5,6), (@D1)
```

Typical response:

```
9.000000E+01,9.000000E+01
```

SOURce:DIGital:MULTitone:TONE:PHASe:RANDomize

Syntax

```
SOURce:DIGital:MULTitone:TONE:PHASe:RANDomize (@<channel_list>)
```

Description

Randomizes the phase of all the tones in the multitone waveform for the specified channel(s).

Parameter

Item	Type	Range of values	Default value
<channel_list>	Discrete	One or more channels. – (@D1)|(@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- This setting is only applicable for the multitone waveform. Use the **SOURce:DIGital:FUNCtion** command to select the multitone waveform type.
- The command is channel based. This means that the command can be applied differently to channel 1 and 2. Refer to “**Appendix E: Relationship between Digital Waveform Parameters and Channels**” on page 667 for the relationship between this command and channels.
- You can verify whether a generator setup has completed by polling the status register value via the **STATus:OPERation:CONDition?** query. While the generator setup is in progress, bit 8 of the condition register of the Standard Operation register group will be set. After the generator setup has completed, bit 8 will be cleared to 0.

Examples

The following command randomized the phase of all the tones for channel 1.

```
SOUR:DIG:MULT:TONE:PHAS:RAND (@D1)
```

SOURce:DIGital:REFerence:VOLTage

Syntax

```
SOURce:DIGital:REFerence:VOLTage <voltage>
```

```
SOURce:DIGital:REFerence:VOLTage?
```

Description

Sets the voltage reference of the digital generator in V. The query returns the voltage reference of the digital generator in V.

Parameter

Item	Type	Range of values	Default value
`<voltage>`	Numeric	1.0E-9 to 1.0E+12	1

Examples

The following command sets the voltage reference of the digital generator to 5 V.

```
SOUR:DIG:REF:VOLT 5
```

The following query returns the voltage reference of the digital generator in V.

```
SOUR:DIG:REF:VOLT?
```

Typical response:

```
5.000000E+00
```

Keysight U8903B
Audio Analyzer

Programmer's Reference

15 STATus Subsystem

This chapter describes the `STATus` subsystem commands.

STATus:OPERation:CONDition?

Syntax

```
STATus:OPERation:CONDition?
```

Description

Queries the condition register for the Standard Operation register group and returns the binary-weighted sum of all bits set in the register. This is a read-only register and the bits are not cleared when you read the register.

Remark

For more information on the Operation condition register, refer to "**Status system diagram**" on page 32. The bit definitions for the Standard Operation register are listed in "**Standard Operation register**" on page 29.

Example

The following query reads the condition register (bit 3 is set).

```
STAT:OPER:COND?
```

Typical response:

8

STATus:OPERation:ENABle

Syntax

```
STATus:OPERation[:EVENt]?
```

Description

Queries the event register for the Standard Operation register group and returns the binary-weighted sum of all bits set in the register. This is a read-only register and the bits are cleared when you read the register.

Remarks

- For more information on the Operation event register, refer to “**Status system diagram**” on page 32. The bit definitions for the Standard Operation register are listed in “**Standard Operation register**” on page 29.
- Once a bit is set, it remains set until cleared by reading the event register or the clear status (***CLS**) command.
- The ***RST**, instrument preset (**SYSTem:PRESet**), and **STATus:PRESet** commands have no effect on this register.

Example

The following query reads the event register (bit 3 is set).

```
STAT:OPER?
```

Typical response:

```
8
```

STATus:OPERation[:EVENt]?

Syntax

```
STATus:OPERation[:EVENt]?
```

Description

Queries the event register for the Standard Operation register group and returns the binary-weighted sum of all bits set in the register. This is a read-only register and the bits are cleared when you read the register.

Remarks

– For more information on the Operation event register, refer to "**Status system diagram**" on page 32. The bit definitions for the Standard Operation register are listed in "**Standard Operation register**" on page 29.

– Once a bit is set, it remains set until cleared by reading the event register or the clear status (***CLS**) command.

– The ***RST**, instrument preset (**SYSTem:PRESet**), and **STATus:PRESet** commands have no effect on this register.

Example

The following query reads the event register (bit 3 is set).

```
STAT:OPER?
```

Typical response:

```
8
```

STATus:OPERation:NTRansition

Syntax

```
STATus:OPERation:NTRansition <value>
STATus:OPERation:NTRansition?
```

Description

Sets and reads the value of the Operation Negative-Transition (NTR) register. This register serves as a polarity filter between the Operation condition and Operation event registers. When a bit in the Operation NTR register is set to 1, then a 1-to-0 transition of the corresponding bit in the Operation condition register causes that bit in the Operation event register to be set. The query returns the binary-weighted sum of all bits set in the register.

Parameter

Item	Type	Range of values	Default value
`<value>`	Numeric	A decimal value which corresponds to the binary-weighted sum of the bits in the register	0

Remarks

- The bit definitions for the Standard Operation register are listed in "**Standard Operation register**" on page 29.
- If the same bits in both NTR and Positive-Transition (PTR) registers are set to 1, then any transition of that bit at the Operation condition register sets the corresponding bit in the Operation event register.
- If the same bits in both NTR and PTR registers are set to 0, then no transition of that bit at the Operation condition register can set the corresponding bit in the Operation event register.
- The **STATus:PRESet** command will set all bits in the NTR register to 0.
- The ***RST** and instrument preset (**SYSTem:PRESet**) commands have no effect on this register.

Examples

The following command enables bits 3 and 4 (decimal value = 24) in the NTR register.

```
STAT:OPER:NTR 24
```

The following query returns the bits enabled in the register.

```
STAT:OPER:NTR?
```

Typical response:

```
24
```

STATus:OPERation:PTRansition

Syntax

```
STATus:OPERation:PTRansition <value>
```

```
STATus:OPERation:PTRansition?
```

Description

Sets and reads the value of the Operation Positive-Transition (PTR) register. This register serves as a polarity filter between the Operation condition and Operation event registers. When a bit in the Operation PTR register is set to 1, then a 0-to-1 transition of the corresponding bit in the Operation condition register causes that bit in the Operation event register to be set. The query returns the binary-weighted sum of all bits set in the register.

Parameter

Item	Type	Range of values	Default value
`<value>`	Numeric	A decimal value which corresponds to the binary-weighted sum of the bits in the register	32767

Remarks

- The bit definitions for the Standard Operation register are listed in "**Standard Operation register**" on page 29.
- If the same bits in both NTR and PTR registers are set to 1, then any transition of that bit at the Operation condition register sets the corresponding bit in the Operation event register.
- If the same bits in both NTR and PTR registers are set to 0, then no transition of that bit at the Operation condition register can set the corresponding bit in the Operation event register.
- The **STATus:PRESet** command will set all bits in the PTR register to 1.
- The ***RST** and instrument preset (**SYSTem:PRESet**) commands have no effect on this register.

Examples

The following command enables bits 3 and 4 (decimal value = 24) in the PTR register.

```
STAT:OPER:PTR 24
```

The following query returns the bits enabled in the register.

```
STAT:OPER:PTR?
```

Typical response:

```
24
```

STATus:PRESet

Syntax

```
STATus:PRESet
```

Description

Sets all defined bits in the status system PTR registers and clears all bits in the NTR and enable registers.

Operation register	Preset setting
STATus:OPERation:ENABle	0 (all bits disabled)
STATus:OPERation:NTRansition	0 (all bits disabled)
STATus:OPERation:PTRansition	32767 (all defined bits enabled)
STATus:QUEStionable:ENABle	0 (all bits disabled)
STATus:QUEStionable:NTRansition	0 (all bits disabled)
STATus:QUEStionable:PTRansition	32767 (all defined bits enabled)

Example

The following command presets the Operation enable register.

```
STAT:PRES
```

STATus:QUEStionable:CONDition?

Syntax

```
STATus:QUEStionable:CONDition?
```

Description

Queries the condition register for the Questionable Data register group and returns the binary-weighted sum of all bits set in the register. This is a read-only register and the bits are not cleared when you read the register.

Remark

For more information on the Questionable condition register, refer to “**Status system diagram**” on page 32. The bit definitions for the Questionable Data register are listed in “**Questionable Status registers**” on page 29.

Example

The following query reads the condition register (bit 0 is set).

```
STAT:QUES:COND?
```

Typical response:

```
1
```

STATus:QUEStionable:ENABle

Syntax

```
STATus:QUEStionable:ENABle <enable_value>
STATus:QUEStionable:ENABle?
```

Description

Enables the bits in the enable register for the Questionable Data register group. The selected bits are then reported to the Status Byte register. The query returns the binary-weighted sum of all bits set in the register.

Parameter

Item	Type	Range of values	Default value
`<enable_value>`	Numeric	A decimal value which corresponds to the binary-weighted sum of the bits in the register	0

Remarks

– For more information on the Questionable enable register, refer to “**Status system diagram**” on page 32. The bit definitions for the Questionable Data register are listed in “**Questionable Status registers**” on page 29.

– Use the `<enable_value>` parameter to specify which bits will be reported to the Status Byte register. The specified decimal value corresponds to the binary-weighted sum of the bits you wish to enable in the register.

– The clear status (***CLS**) command will not clear the enable register but it clears all bits in the event register.

– The **STATus:PRESet** command will clear all bits in the enable register.

– The ***RST** and instrument preset (**STATus:PRESet**) commands have no effect on this register.

Examples

The following command enables bit 0 (decimal value = 1) in the enable register.

```
STAT:QUES:ENAB 1
```

The following query returns the bit enabled in the register.

```
STAT:QUES:ENAB?
```

Typical response:

```
1
```

STATus:QUEStionable[:EVENt]?

Syntax

```
STATus:QUEStionable[:EVENt]?
```

Description

Queries the event register for the Questionable Data register group and returns the binary-weighted sum of all bits set in the register. This is a read-only register and the bits are cleared when you read the register.

Remarks

- For more information on the Questionable event register, refer to "**Status system diagram**" on page 32. The bit definitions for the Questionable Data register are listed in "**Questionable Status registers**" on page 29.
- Once a bit is set, it remains set until cleared by reading the event register or the clear status (***CLS**) command.
- The ***RST**, instrument preset (**SYSTem:PRESet**), and **STATus:PRESet** commands have no effect on this register.

Example

The following query reads the event register (bit 0 is set).

```
STAT:QUES?
```

Typical response:

```
1
```

STATus:QUEStionable:NTRansition

Syntax

```
STATus:QUEStionable:NTRansition <value>
STATus:QUEStionable:NTRansition?
```

Description

Sets and reads the value of the Questionable Negative-Transition (NTR) register. This register serves as a polarity filter between the Questionable condition and Questionable event registers. When a bit in the Questionable NTR register is set to 1, then a 1-to-0 transition of the corresponding bit in the Questionable condition register causes that bit in the Questionable event register to be set. The query returns the binary-weighted sum of all bits set in the register.

Parameter

Item	Type	Range of values	Default value
`<value>`	Numeric	A decimal value which corresponds to the binary-weighted sum of the bits in the register	0

Remarks

- The bit definitions for the Questionable Data register are listed in "**Questionable Status registers**" on page 29.
- If the same bits in both NTR and PTR registers are set to 1, then any transition of that bit at the Questionable condition register sets the corresponding bit in the Questionable event register.
- If the same bits in both NTR and PTR registers are set to 0, then no transition of that bit at the Questionable condition register can set the corresponding bit in the Questionable event register.
- The **STATus:PRESet** command will set all bits in the NTR register to 0.
- The ***RST** and instrument preset (**SYSTem:PRESet**) commands have no effect on this register.

Examples

The following command enables bit 0 (decimal value = 1) in the NTR register.

```
STAT:QUES:NTR 1
```

The following query returns the bit enabled in the register.

```
STAT:QUES:NTR?
```

Typical response:

```
1
```

STATus:QUEStionable:PTRansition

Syntax

```
STATus:QUEStionable:PTRansition <value>
STATus:QUEStionable:PTRansition?
```

Description

Sets and reads the value of the Questionable Positive-Transition (PTR) register. This register serves as a polarity filter between the Questionable condition and Questionable event registers. When a bit in the Questionable PTR register is set to 1, then a 0-to-1 transition of the corresponding bit in the Questionable condition register causes that bit in the Questionable event register to be set. The query returns the binary-weighted sum of all bits set in the register.

Parameter

Item	Type	Range of values	Default value
`<value>`	Numeric	A decimal value which corresponds to the binary-weighted sum of the bits in the register	32767

Remarks

– The bit definitions for the Questionable Data register are listed in "**Questionable Status registers**" on page 29.

– If the same bits in both NTR and PTR registers are set to 1, then any transition of that bit at the Questionable condition register sets the corresponding bit in the Questionable event register.

– If the same bits in both NTR and PTR registers are set to 0, then no transition of that bit at the Questionable condition register can set the corresponding bit in the Questionable event register.

– The **STATus:PRESet** command will set all bits in the PTR register to 1.

– The ***RST** and instrument preset (**SYSTem:PRESet**) commands have no effect on this register.

Examples

The following command enables bit 0 (decimal value = 1) in the PTR register.

```
STAT:QUES:PTR 1
```

The following query returns the bit enabled in the register.

```
STAT:QUES:PTR?
```

Typical response:

```
1
```

STATus:QUEStionable:VOLTage:CONDition?

Syntax

```
STATus:QUEStionable:VOLTage:CONDition?
```

Description

Queries the condition register for the Questionable Voltage register group and returns the binary-weighted sum of all bits set in the register. This is a read-only register and the bits are not cleared when you read the register.

Remark

For more information on the Questionable condition register, refer to "**Status system diagram**" on page 32. The bit definitions for the Questionable Voltage register are listed in "**Questionable Status registers**" on page 29.

Example

The following query reads the condition register (bit 1 is set).

```
STAT:QUES:VOLT:COND?
```

Typical response:

```
2
```

STATus:QUEStionable:VOLTage:ENABle

Syntax

```
STATus:QUEStionable:VOLTage:ENABle <enable_value>
STATus:QUEStionable:VOLTage:ENABle?
```

Description

Enables the bits in the enable register for the Questionable Voltage register group. The selected bits are then reported to the Questionable Data register. The query returns the binary-weighted sum of all bits set in the register.

Parameter

Item	Type	Range of values	Default value
`<enable_value>`	Numeric	A decimal value which corresponds to the binary-weighted sum of the bits in the register	0

Remarks

– For more information on the Questionable enable register, refer to "**Status system diagram**" on page 32. The bit definitions for the Questionable Voltage register are listed in "**Questionable Status registers**" on page 29.

– Use the `<enable value>` parameter to specify which bits will be reported to the Questionable Data register. The specified decimal value corresponds to the binary-weighted sum of the bits you wish to enable in the register. For example, to enable bit 0 (decimal value = 1) and bit 1 (decimal value = 2), the corresponding decimal value would be 3 (1 + 2).

– The clear status (***CLS**) command will not clear the enable register but it clears all bits in the event register.

– The ***RST** and instrument preset (**SYSTem:PRESet**) commands have no effect on this register.

Examples

The following command enables bit 9 (decimal value = 512) in the enable register.

```
STAT:QUES:VOLT:ENAB 512
```

The following query returns the bit enabled in the register.

```
STAT:QUES:VOLT:ENAB?
```

Typical response:

```
512
```

STATus:QUEStionable:VOLTage[:EVENt]?

Syntax

```
STATus:QUEStionable:VOLTage[:EVENt]?
```

Description

Queries the event register for the Questionable Voltage register group and returns the binary-weighted sum of all bits set in the register. This is a read-only register and the bits are cleared when you read the register.

Remarks

- For more information on the Questionable event register, refer to "**Status system diagram**" on page 32. The bit definitions for the Questionable Voltage register are listed in "**Questionable Status registers**" on page 29.
- Once a bit is set, it remains set until cleared by reading the event register or the clear status (***CLS**) command.
- The ***RST**, instrument preset (**SYSTem:PRESet**), and **STATus:PRESet** commands have no effect on this register.

Example

The following query reads the event register (bit 1 is set).

```
STAT:QUES:VOLT?
```

Typical response:

```
2
```

STATus:QUEStionable:VOLTage:NTRansition

Syntax

```
STATus:QUEStionable:VOLTage:NTRansition <value>
STATus:QUEStionable:VOLTage:NTRansition?
```

Description

Sets and reads the value of the Questionable Voltage Negative-Transition (NTR) register. This register serves as a polarity filter between the Questionable Voltage condition and Questionable Voltage event registers. When a bit in the Questionable Voltage NTR register is set to 1, then a 1-to-0 transition of the corresponding bit in the Questionable Voltage condition register causes that bit in the Questionable Voltage event register to be set. The query returns the binary-weighted sum of all bits set in the register.

Parameter

Item	Type	Range of values	Default value
`<value>`	Numeric	A decimal value which corresponds to the binary-weighted sum of the bits in the register	0

Remarks

- The bit definitions for the Questionable Voltage register are listed in "**Questionable Status registers**" on page 29.
- If the same bits in both NTR and PTR registers are set to 1, then any transition of that bit at the Questionable Voltage condition register sets the corresponding bit in the Questionable Voltage event register.
- If the same bits in both NTR and PTR registers are set to 0, then no transition of that bit at the Questionable Voltage condition register can set the corresponding bit in the Questionable Voltage event register.
- The **STATus:PRESet** command will set all bits in the NTR register to 0.
- The ***RST** and instrument preset (**SYSTem:PRESet**) commands have no effect on this register.

Examples

The following command enables bits 0 and 1 (decimal value = 3) in the NTR register.

```
STAT:QUES:VOLT:NTR 3
```

The following query returns the bits enabled in the register.

```
STAT:QUES:VOLT:NTR?
```

Typical response:

```
3
```

STATus:QUEStionable:VOLTage:PTRansition

Syntax

```
STATus:QUEStionable:VOLTage:PTRansition <value>
STATus:QUEStionable:VOLTage:PTRansition?
```

Description

Sets and reads the value of the Questionable Voltage Positive-Transition (PTR) register. This register serves as a polarity filter between the Questionable Voltage condition and Questionable Voltage event registers. When a bit in the Questionable Voltage PTR register is set to 1, then a 0-to-transition of the corresponding bit in the Questionable Voltage condition register causes that bit in the Questionable Voltage event register to be set. The query returns the binary-weighted sum of all bits set in the register.

Parameter

Item	Type	Range of values	Default value
`<value>`	Numeric	A decimal value which corresponds to the binary-weighted sum of the bits in the register	32767

Remarks

- The bit definitions for the Questionable Voltage register are listed in "**Questionable Status registers**" on page 29.
- If the same bits in both NTR and PTR registers are set to 1, then any transition of that bit at the Questionable Voltage condition register sets the corresponding bit in the Questionable Voltage event register.
- If the same bits in both NTR and PTR registers are set to 0, then no transition of that bit at the Questionable Voltage condition register can set the corresponding bit in the Questionable Voltage event register.
- The **STATus:PRESet** command will set all bits in the PTR register to 1.
- The ***RST** and instrument preset (**SYSTem:PRESet**) commands have no effect on this register.

Examples

The following command enables bits 0 and 1 (decimal value = 3) in the PTR register.

```
STAT:QUES:VOLT:PTR 3
```

The following query returns the bits enabled in the register.

```
STAT:QUES:VOLT:PTR?
```

Typical response:

```
3
```

THIS PAGE HAS BEEN INTENTIONALLY LEFT BLANK.

16 SWEep Subsystem

This chapter describes the `SWEep` subsystem commands.

SENSe:APPLication:TYPE

Syntax

```
SENSe:APPLication:TYPE <application_type>
```

```
SENSe: APPLication:TYPE?
```

Description

Selects the predefined application type: SWEep or GDELay.

Parameter

Item	Type	Range of values	Default value
<application_type>	Discrete	SWEep|GDELay	SWEep

Remarks

- SWEep functions as a standard sweep operation.
- GDELay's sweep operates on a digital interface by sweeping the frequency on the sine wave and the analyzer's function 3 is set to measure the group delay.
- GDELay option can only be selected when the digital card option is purchased and used.
- When GDELay is active the following cannot be changed: waveform of the reference generator, function 3 of the reference analyzer, and multi-channel status.

Examples

The following command sets sweep application type to GDELay.

```
SENSe:APPLication:TYPE   GDELay
```

The following query returns the application type.

```
SENS:APPL:TYPE   ?
```

Typical response:

```
GDELay
```

SENSe:SWEep:CHANnel

Syntax

```
SENSe:SWEep:CHANnel <channel>
```

```
SENSe:SWEep:CHANnel?
```

Description

Sets the analyzer channel to perform sweep. The query returns the selected sweep channel.

Parameter

Item	Type	Range of values	Default value
`<channel>`	Numeric	1|2	1

Remark

This command is invalid when legacy sweep is enabled. Refer to “**Appendix M: Legacy Sweep**” on page 693 for more information.

Examples

The following command sets channel 1 to perform sweep.

```
SENS:SWE:CHAN 1
```

The following query returns the sweep channel.

```
SENS:SWE:CHAN?
```

Typical response:

```
1
```

SENSe:SWEep:REFerence:CHANnel

Syntax

```
SENSe:SWEep:REFerence:CHANnel <channel>
SENSe:SWEep:REFerence:CHANnel?
```

Description

Sets the analyzer reference channel to perform sweep. The query returns the selected sweep reference channel.

Parameter

Item	Type	Range of values	Default value
`<channel>`	Numeric	1|2|3|4|5|6|7|8	1

Remark

This command is invalid when legacy sweep is enabled. Refer to “**Appendix M: Legacy Sweep**” on page 693 for more information.

Examples

The following command sets the analog reference channel to 1.

```
SENS:SWE:REF:CHAN 1
```

The following query returns the reference sweep channel.

```
SENS:SWE:REF:CHAN?
```

Typical response:

```
1
```

SOURce:SWEep:CHANnel

Syntax

```
SOURce:SWEep:CHANnel <channel>[,<channel>]
```

```
SOURce:SWEep:CHANnel?
```

Description

Sets the channel to perform sweep. The query returns the selected sweep channel. Multiple responses are separated by commas.

Parameter

Item	Type	Range of values	Default value
`<channel>`	Numeric	1|2	1

Remarks

– Multiple channels can be swept at a time.

– The sweep channel refers to the generator channel to perform sweep.

– This command is invalid when legacy sweep is enabled. Refer to "**Appendix M: Legacy Sweep**" on page 693 for more information.

– Refer to "**Example 8: Perform Sweep**" on page 649 for the examples on performing sweep.

Examples

The following command sets channel 1 and channel 2 to perform sweep.

```
SOUR:SWE:CHAN 1,2
```

The following query returns the sweep channels.

```
SOUR:SWE:CHAN?
```

Typical response:

```
1,2
```

SOURce:SWEep:DWELl

Syntax

```
SOURce:SWEep:DWELl <delay>
SOURce:SWEep:DWELl?
```

Description

Sets the sweep dwell time (ms) for the channel specified in the **SOURce:SWEep:CHANnel** command. The dwell time is the delay between the start of the signal generation and the start of making the measurement. The query returns the dwell time of the selected channel in ms.

Parameter

Item	Type	Range of values	Default value
`<delay>`	Numeric	0 to 5000 ms	0

Remarks

– The setting will be the same for all channels selected by the **SOURce:SWEep:CHANnel** command.

– Refer to “**Example 8: Perform Sweep**” on page 649 for the examples on performing sweep.

Examples

The following command sets the dwell time to 1 s.

```
SOUR:SWE:DWEL1 1000
```

The following query returns the dwell time.

```
SOUR:SWE:DWEL1?
```

Typical response:

```
1000
```

SOURce:SWEep:MODE

Syntax

```
SOURce:SWEep:MODE <mode>
SOURce:SWEep:MODE?
```

Description

Sets the sweep or list mode for the channel specified in the **SOURce:SWEep:CHANnel** command. The query returns the sweep mode of the specified channel.

The description for each <mode> parameter is shown as follows.

`ASWeep`	Auto sweep – Sweep is performed automatically. – Sweep points are based on the Start, Stop, and Step Size sweep parameter settings.
`ALISt`	Auto list – Sweep is performed automatically. – Sweep points are predefined and downloaded, or loaded from a file into the U8903B.
`MSWeep`	Manual sweep – Sweep is performed manually. – Sweep points are based on the Start, Stop, and Step Size sweep parameter settings.
`MLISt`	Manual list – Sweep is performed manually. – Sweep points are predefined and downloaded, or loaded from a file into the U8903B.

Parameter

Item	Type	Range of values	Default value
`<mode>`	Discrete	ASWeep|ALISt|MSWeep|MLISt	ASWeep

Remarks

– The setting will be the same for all channels selected by the **SOURce:SWEep:CHANnel** command.

– Refer to “**Example 8: Perform Sweep**” on page 649 for the examples on performing sweep.

Examples

The following command sets the sweep mode to auto sweep.

```
SOUR:SWE:MODE ASW
```

The following query returns the sweep mode.

```
SOUR:SWE:MODE?
```

Typical response:

```
ASW
```

SOURce:SWEep:NEXT

Syntax

```
SOURce:SWEep:NEXT
```

Description

Jumps to the next sweep point in the manual sweep or manual list sweep mode.

SOURce:SWEep:PARameter

Syntax

```
SOURce:SWEep:PARameter <sweep_parameter>
SOURce:SWEep:PARameter?
```

Description

Sets the parameter to sweep for the channel specified in the **SOURce:SWEep:CHANnel** command. The query returns the sweep parameter of the selected channel.

The description for each <sweep_parameter> is shown as follows.

FREQuency	Frequency values of the sine, variable phase, and square waveforms.
FREQuency1	Frequency values of the sine, variable phase, and square waveforms. Frequency 1 value of the dual waveform.
FREQuency2	Frequency 2 value of the dual waveform.
AMPLitude	Amplitude values of all waveform types.
PHASe	Phase value of the variable phase waveform.
CENTer	Center frequency value of the DFD IEC 60268 waveform.
DIFFerence	Difference frequency value of the DFD IEC 60118 or DFD IEC 60268 waveform.
UPPer	Upper frequency value of the DFD IEC 60118, SMPTE 1:1, SMPTE 4:1, or SMPTE 10:1 waveform.
LOWer	Lower frequency value of the SMPTE 1:1, SMPTE 4:1, or SMPTE 10:1 waveform.

Parameter

Item	Type	Range of values	Default value
<sweep_parameter>	Discrete	FREQuency1|FREQuency2|AMPLitude|PHASe|CENTer|DIFFerence|UPPer|LOWer	FREQuency1

Remarks

- For sine, variable phase, and square waveforms, it is recommended to use `FREQuency`. For backward compatibility with U8903A, `FREQuency1` can also be used.
- The setting will be the same for all channels selected by the **SOURce:SWEep:CHANnel** command.
- Refer to "**Example 8: Perform Sweep**" on page 649 for the examples on performing sweep.

Examples

The following command sets the sweep parameter to the frequency of the sine waveform.

```
SOUR:SWE:PAR FREQ1
```

The following query returns the sweep parameter.

```
SOUR:SWE:PAR?
```

Typical response:

```
FREQ1
```

SOURce:SWEep:POINts

Syntax

```
SOURce:SWEep:POINts <points>
SOURce:SWEep:POINts?
```

Description

Sets the number of sweep points for the channel specified in the **SOURce:SWEep:CHANnel** command. The query returns the number of sweep points of the selected channel.

The relationship between the number of points and the stop, start, and step size for linear sweep is computed as follows.

$STEP = (STOP - STARt) / (POINts - 1)$

The following equation shows the relationship between the number of points and the stop, start, and step size for logarithmic sweep.

$STOP = (STARt)(STEP)^{POINts - 1}$

If the number of points changes, the step size will also change, but span will not be affected.

$SPAN = STOP - STARt$

Parameter

Item	Type	Range of values	Default value
`<points>`	Numeric	Minimum: 2 Maximum: 1024	30

Remarks

– The number of sweep points configuration is not applicable for the auto list or manual list sweep mode.

– The setting will be the same for all channels selected by the **SOURce:SWEep:CHANnel** command.

– Refer to “**Example 8: Perform Sweep**” on page 649 for the examples on performing sweep.

Examples

The following command sets the number of sweep points to 20.

```
SOUR:SWE:POIN 20
```

The following query returns the number of sweep points.

```
SOUR:SWE:POIN?
```

Typical response:

```
20
```

SOURce:SWEep:REFerence:CHANnel

Syntax

```
SOURce:SWEep:REFerence:CHANnel <channel>
```

```
SOURce:SWEep:REFerence:CHANnel?
```

Description

Sets the reference channel to perform sweep. The query returns the selected sweep reference channel.

Parameter

Item	Type	Range of values	Default value
`<channel>`	Numeric	1|2	1

Remark

This command is invalid when legacy sweep is enabled. Refer to “**Appendix M: Legacy Sweep**” on page 693 for more information.

Examples

The following command sets the analog reference channel to 1.

```
SOUR:SWE:REF:CHAN 1
```

The following query returns the reference sweep channel.

```
SOUR:SWE:REF:CHAN?
```

Typical response:

```
1
```

SOURce:SWEep:SPACing

Syntax

```
SOURce:SWEep:SPACing <spacing>
SOURce:SWEep:SPACing?
```

Description

Sets either linear or log interval for the sweep of the channel specified in the **SOURce:SWEep:CHANnel** command. The query returns the sweep spacing of the selected channel in the form of `LIN` or `LOG`.

The description for each `<spacing>` parameter is shown as follows.

`LINear`	The sweep step size will increment or decrement the sweep point until the sweep limit is reached.
`LOGarithmic`	For non-linear sweeps, the step size is determined by a logarithmic curve fitted between the start and stop frequency. Stepping is determined by the number of sweep points (**SOURce:SWEep:POINts**).

Parameter

Item	Type	Range of values	Default value
`<spacing>`	Discrete	LINear|LOGarithmic	LOGarithmic

Remarks

- The setting will be the same for all channels selected by the **SOURce:SWEep:CHANnel** command.
- Refer to “**Example 8: Perform Sweep**” on page 649 for the examples on performing sweep.

Examples

The following command sets the log sweep interval.

```
SOUR:SWE:SPAC LOG
```

The following query returns the sweep spacing.

```
SOUR:SWE:SPAC?
```

Typical response:

```
LOG
```

SOURce:SWEep:STARt

Syntax

```
SOURce:SWEep:STARt <start>
SOURce:SWEep:STARt?
```

Description

Sets the sweep start point for the channel specified in the **SOURce:SWEep:CHANnel** command. The query returns the sweep start point of the specified channel.

Parameter

Refer to "**Appendix F: Sweep Start and Stop Range**" on page 669 for the range of the start values for each waveform type and sweep parameter.

Remarks

– The unit for the start value of each corresponding sweep parameter is listed as follows. The returned value is also in the unit as listed.

Sweep parameter	Unit for the start value
`FREQuency1`	Hz
`FREQuency2`	Hz
`AMPLitude`	– Vrms – V (for the DC signal)
`PHASe`	°
`CENTer`	Hz
`DIFFerence`	Hz
`UPPer`	Hz
`LOWer`	Hz

– Refer "**SOURce:SWEep:POINts**" on page 578 for the relation between the step size and points, and between the start value and stop value.

– The setting will be the same for all channels selected by the **SOURce:SWEep:CHANnel** command.

– Refer to "**Example 8: Perform Sweep**" on page 649 for the examples on performing sweep.

Examples

The following command sets the sweep start point to 1 kHz. (Assume that frequency is the sweep parameter.)

```
SOUR:SWE:STAR 1000
```

The following query returns the start point.

```
SOUR:SWE:STAR?
```

Typical response:

```
1.000000E+03
```

SOURce:SWEep:STEP

Syntax

```
SOURce:SWEep:STEP <step>
SOURce:SWEep:STEP?
```

Description

Sets the step size of the linear sweep interval, or multiplier factor of the log sweep interval for the channel specified in the **SOURce:SWEep:CHANnel** command. The query returns the step size of the selected channel.

Parameter

Refer to the **SOURce:SWEep:POINts** command for the relationship between the step size and the stop, start, and number of points for linear or logarithmic sweep. The start and stop range of values for each waveform type and sweep parameter are listed in “**Appendix F: Sweep Start and Stop Range**” on page 669.

Remarks

– The multiplier factor for the log interval must not be more than 0 or equal to 1.

– This command will set the step size if the sweep spacing is linear. If the sweep spacing is logarithmic, this command will set the multiply factor.

– Refer “**SOURce:SWEep:POINts**” on page 578 for the relation between the step size and points, and between the start value and stop value.

– The multiply factor for logarithmic spacing must not be less than 0 or equal to 1.

– The unit for the step size of each corresponding sweep parameter is listed as follows. The returned value is also in the unit as listed.

Sweep parameter	Unit for the start value
`FREQuency1`	Hz
`FREQuency2`	Hz
`AMPLitude`	– Vrms – V (for the DC signal)
`PHASe`	°
`CENTer`	Hz
`DIFFerence`	Hz

Sweep parameter	Unit for the start value
`UPPer`	Hz
`LOWer`	Hz

- The setting will be the same for all channels selected by the **SOURce:SWEep:CHANnel** command.
- Refer to "**Example 8: Perform Sweep**" on page 649 for the examples on performing sweep.

Examples

The following command sets the step size to 100 Hz. (Assume that frequency is the sweep parameter.)

```
SOUR:SWE:STEP 100
```

The following query returns the step size.

```
SOUR:SWE:STEP?
```

Typical response:

```
1.000000E+02
```

SOURce:SWEep:STOP

Syntax

```
SOURce:SWEep:STOP <stop>
SOURce:SWEep:STOP?
```

Description

Sets the sweep stop point for the channel specified in the **SOURce:SWEep:CHANnel** command. The query returns the sweep stop point of the specified channel.

Parameter

Refer to "**Appendix F: Sweep Start and Stop Range**" on page 669 for the range of the stop values for each waveform type and sweep parameter.

Remarks

– The unit for the stop value of each corresponding sweep parameter is listed as follows. The returned value is also in the unit as listed.

Sweep parameter	Unit for the start value
FREQuency1	Hz
FREQuency2	Hz
AMPLitude	– Vrms – V (for the DC signal)
PHASe	°
CENTer	Hz
DIFFerence	Hz
UPPer	Hz
LOWer	Hz

– Refer "**SOURce:SWEep:POINts**" on page 578 for the relation between the step size and points, and between the start value and stop value.

– The setting will be the same for all channels selected by the **SOURce:SWEep:CHANnel** command.

– Refer to "**Perform Sweep**" on page 649 for the examples on performing sweep.

Examples

The following command sets the sweep stop point to 3 kHz. (Assume that frequency is the sweep parameter.)

```
SOUR:SWE:STOP 3000
```

The following query returns the stop point.

```
SOUR:SWE:STOP?
```

Typical response:

```
3.000000E+03
```

SOURce:SWEep:VALues?

Syntax

```
SOURce:SWEep:VALues? (@<channel>)
```

Description

Returns the values of the sweep points for the specified channel. Multiplecresponses are separated by commas.

Parameter

Item	Type	Range of values	Default value
`<channel>`	Numeric	1|2|3|4|5|6|7|8	Required parameter

Remarks

- You may query the values of the sweep points after sending the **INITiate[:IMMediate]:SWEep** command to trigger the sweep.
- This command is useful if you are plotting the sweep results in a graph. You can use the value returned with this command to plot the X-axis of the sweep chart and the result obtained by sending the **FETCh:SWEep?** command to plot the Y-axis of the sweep chart.
- Refer to “**Example 8: Perform Sweep**” on page 649 for the examples on performing sweep.

Example

The following query returns the values of the sweep points. (Assume that the sweep start point is 100 Hz, stop point is 1000 Hz, and step size is 100 Hz.)

```
SOUR:SWE:VAL? (@2)
```

Typical response:

```
1.000000E+02,2.000000E+02,3.000000E+02,4.000000E+02,5.000000E+02,
6.000000E+02,7.000000E+02,8.000000E+02,9.000000E+02,1.000000E+03
```

THIS PAGE HAS BEEN INTENTIONALLY LEFT BLANK.

17 SYSTem Subsystem

KEYSIGHT TECHNOLOGIES

This chapter describes the `SYSTem` subsystem commands.

SYSTem[:ANALog]:CHANnel?

Syntax

```
SYSTem[:ANALog]:CHANnel? <card_type>
```

Description

Queries the available channels in the U8903B to determine if the channel hardware card is available or in good condition. The query returns comma-separated channel numbers of the available channels in the U8903B.

Parameter

Item	Type	Range of values	Default value
`<card_type>`	Discrete	ANALyzer|GENerator	ANALyzer

Remark

If a hardware card is available but in bad condition, this query will not return the channel number for that particular channel.

Example

The following query returns the channel numbers of the available analyzer channels which are in good condition.

```
SYST:CHAN? ANAL
```

Typical response:

```
1,2
```

SYSTem:AUXiliary:DCOutput

Syntax

```
SYSTem:AUXiliary:DCOutput <voltage>
```

```
SYSTem:AUXiliary:DCOutput
```

Description

Sets the auxiliary output voltage value. The query returns the auxiliary output voltage value.

Parameter

Item	Type	Range of values	Default value
`<voltage>`	Numeric	0.5 V to 5.2 V	0.5 V

Remark

The input voltage value will be rounded to 1 decimal place.

Examples

The following command sets the auxiliary output voltage to 1.5 V.

```
SYST:AUX:DCO 1.5
```

The following query returns the auxiliary output signal voltage.

```
SYST:AUX:DCO?
```

Typical response:

```
1.5E+00
```

SYSTem:AUXiliary:MODE

Syntax

```
SYSTem:AUXiliary:MODE <mode>
SYSTem:AUXiliary:MODE?
```

Description

Sets the auxiliary output signal mode. The query returns the auxiliary output signal mode.

Parameter

Item	Type	Range of values	Default value
`<mode>`	Discrete	STEReo|MONO	MONO

Examples

The following command sets the auxiliary output signal mode to mono.

```
SYST:AUX:MODE MONO
```

The following query returns the auxiliary output signal mode.

```
SYST:AUX:MODE?
```

Typical response:

```
MONO
```

SYSTem:AUXiliary[:MONitor]

Syntax

```
SYSTem:AUXiliary[:MONitor] <state>
```

```
SYSTem:AUXiliary[:MONitor]?
```

Description

Enables or disables the U8903B auxiliary output monitoring. The query returns the U8903B auxiliary output state.

Parameter

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	OFF

Examples

The following command enables the U8903B auxiliary output.

```
SYST:AUX ON
```

The following query returns the U8903B auxiliary output state.

```
SYST:AUX?
```

Typical response:

```
1
```

SYSTem:AUXiliary:OUTPut

Syntax

```
SYSTem:AUXiliary:OUTPut <type>
SYSTem:AUXiliary:OUTPut?
```

Description

Sets the auxiliary output type. The query returns the auxiliary output type.

Parameter

Item	Type	Range of values	Default value
`<type>`	Discrete	SPEaker|PHONe	SPEaker

Examples

The following command sets the auxiliary output to headphone.

```
SYST:AUX:OUTP PHON
```

The following query returns the auxiliary output type.

```
SYST:AUX:OUTP?
```

Typical response:

```
PHON
```

SYSTem:AUXiliary:VOLume

Syntax

```
SYSTem:AUXiliary:VOLume <volume>
```

```
SYSTem:AUXiliary:VOLume
```

Description

Sets the auxiliary output signal volume. The query returns the auxiliary output signal volume.

Parameter

Item	Type	Range of values	Default value
`<volume>`	Numeric	0 to 100	0

Examples

The following command sets the auxiliary output signal volume to 30.

```
SYST:AUX:VOL 30
```

The following query returns the auxiliary output signal volume.

```
SYST:AUX:VOL?
```

Typical response:

```
30
```

SYSTem:COMMunicate:FTP[:STATe]

Syntax

```
SYSTem:COMMunicate:FTP[:STATe] <state>
```

Description

Enables or disables the FTP server control on the U8903B.

Parameter

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	ON

Example

The following command disables the FTP control.

```
SYST:COMM:FTP:STAT OFF
```

SYSTem:COMMunicate:GPIB[:SELF]:ADDRess

Syntax

```
SYSTem:COMMunicate:GPIB[:SELF]:ADDRess <address>
```

```
SYSTem:COMMunicate:GPIB[:SELF]:ADDRess?
```

Description

Assigns the U8903B GPIB (IEEE-488) address. Each device on the GPIB interface must have a unique address. The query returns the GPIB address.

Parameter

Item	Type	Range of values	Default value
`<address>`	Numeric	0 to 30	28

Remarks

- Your PC GPIB interface card has its own address. Avoid using the PC address for any instrument on the interface bus.
- The GPIB address is stored in nonvolatile memory, and does not change when power has been turned off, after a factory reset (***RST**) command, or after an instrument preset (**SYSTem:PRESet**) command.

Examples

The following command sets the GPIB address to 28.

```
SYST:COMM:GPIB:ADDR 28
```

The following query returns the GPIB address.

```
SYST:COMM:GPIB:ADDR?
```

Typical response:

```
28
```

SYSTem:COMMunicate:LAN:ADDRess

Syntax

```
SYSTem:COMMunicate:LAN:ADDRess <address>
```

```
SYSTem:COMMunicate:LAN:ADDRess?
```

Description

Assigns a static Internet Protocol (IP) address for the U8903B. The query returns the IP address in the form of “A.B.C.D”.

Parameter

Item	Type	Range of values	Default value
`<address>`	String	Up to 15 characters formatted as A.B.C.D where A, B, C, and D is within the range of 0 to 255 each (no embedded spaces)	Required parameter

Remarks

- Sending this command will automatically disable the DHCP and switch to static IP.
- The IP address is stored in nonvolatile memory, and does not change when power has been turned off, after a factory reset (***RST**) command, or after an instrument preset (**SYSTem:PRESet**) command.

Examples

The following command sets the IP address.

```
SYST:COMM:LAN:ADDR “169.254.149.35”
```

The following query returns the IP address in double quotes.

```
SYST:COMM:LAN:ADDR?
```

Typical response:

```
“169.254.149.35”
```

SYSTem:COMMunicate:LAN:DGATeway

Syntax

```
SYSTem:COMMunicate:LAN:DGATeway <gateway>
```

```
SYSTem:COMMunicate:LAN:DGATeway?
```

Description

Assigns the static default gateway address. The query returns the default gateway address in the form of “A.B.C.D”.

Parameter

Item	Type	Range of values	Default value
`<gateway>`	String	Up to 15 characters formatted as A.B.C.D where A, B, C, and D is within the range of 0 to 255 each (no embedded spaces)	Required parameter

Remarks

- Sending this command will automatically disable the DHCP and switch to static default gateway.
- The default gateway address is stored in nonvolatile memory, and does not change when power has been turned off, after a factory reset (***RST**) command, or after an instrument preset (**SYSTem:PRESet**) command.

Examples

The following command sets the default gateway.

```
SYST:COMM:LAN:DGAT “255.255.20.11”
```

The following query returns the default gateway address in double quotes.

```
SYST:COMM:LAN:DGAT?
```

Typical response:

```
“255.255.20.11”
```

SYSTem:COMMunicate:LAN:DHCP:ENABled

Syntax

```
SYSTem:COMMunicate:LAN:DHCP:ENABled
```

Description

Enables the Dynamic Host Configuration Protocol (DHCP) for the U8903B. When the DHCP is enabled (factory setting), the U8903B will try to obtain an IP address from a DHCP server. If a DHCP server is found, it will assign a dynamic IP address, subnet mask, and default gateway to the U8903B.

Example

The following command enables the DHCP.

```
SYST:COMM:LAN:DHCP:ENAB
```

SYSTem:COMMunicate:LAN:HNAMe?

Syntax

```
SYSTem:COMMunicate:LAN:HNAMe?
```

Description

Queries the LAN hostname, and returns an ASCII string enclosed in double quotes.

Example

The following query returns the hostname of the U8903B in double quotes.

```
SYST:COMM:LAN:HNAM?
```

Typical response:

```
“U8903B”
```

SYSTem:COMMunicate:LAN:MAC?

Syntax

```
SYSTem:COMMunicate:LAN:MAC?
```

Description

Reads the U8903B Media Access Control (MAC) address, also known as either the link-layer address, Ethernet (station) address, LANIC ID, or hardware address. This is an unchangeable 48-bit address assigned by the manufacturer to each unique Internet device. The query returns an ASCII string enclosed in double quotes. The MAC address is represented as 12 hexadecimal characters.

NOTE Your network administrator may need the MAC address if they are assigning a static IP address for this device.

Remarks

- The U8903B MAC address is set at the factory and cannot be changed.
- The MAC address is stored in nonvolatile memory, and does not change when power has been turned off, after a factory reset (***RST**) command, or after an instrument preset (**SYSTem:PRESet**) command.

Examples

The following query returns the MAC address in double quotes.

```
SYST:COMM:LAN:MAC?
```

Typical response:

```
“0003D3041075”
```

SYSTem:COMMunicate:LAN:SMASk

Syntax

```
SYSTem:COMMunicate:LAN:SMASk <subnet_mask>
SYSTem:COMMunicate:LAN:SMASk?
```

Description

Sets the static subnet mask address. The query returns the subnet mask address in the form of “A.B.C.D”.

Parameter

Item	Type	Range of values	Default value
`<subnet_mask>`	String	Up to 15 characters formatted as A.B.C.D where A, B, C, and D is within the range of 0 to 255 each (no embedded spaces)	Required parameter

Remarks

- Sending this command will automatically disable the DHCP and switch to static subnet mask.
- The subnet mask address is stored in nonvolatile memory, and does not change when power has been turned off, after a factory reset (***RST**) command, or after an instrument preset (**SYSTem:PRESet**) command.

Examples

The following command sets the subnet mask.

```
SYST:COMM:LAN:SMAS “255.255.20.11”
```

The following query returns the subnet mask address in double quotes.

```
SYST:COMM:LAN:SMAS?
```

Typical response:

```
“255.255.20.11”
```

SYSTem:CTYPe?

Syntax

```
SYSTem:CTYPe? <slot_number>
```

Description

The query returns the card type of the specified card slot number in the U8903B. There are six card slots (maximum) in the U8903B.

Parameter

Item	Type	Range of values	Default value
`<slot_number>`	Numeric	1|2|3|4|5|6	1

Example

The following query returns the card type of slot 1.

```
SYST:CTYP? 1
```

Typical response:

```
AGEN
```

SYSTem:DATE

Syntax

```
SYSTem:DATE <yyyy>,<mm>,<dd>
SYSTem:DATE?
```

Description

Sets the date of the real time clock in year (yyyy), month (mm), and day (dd) format. The query returns comma-separated values that correspond to the year, month, and day.

Parameters

Item	Type	Range of values	Default value
<yyyy>	Numeric	A 4-digit integer representing the year. The value is within the range of 2000 to 2099.	Required parameter
<mm>	Numeric	An integer from 1 to 12	Required parameter
<dd>	Numeric	An integer from 1 to 31	Required parameter

Examples

The following command sets the date (January 1, 2014).

```
SYST:DATE 2014,1,1
```

The following query returns the date.

```
SYST:DATE?
```

Typical response:

```
2014,1,1
```

SYSTem:DISPlay:IMAGe?

Syntax

```
SYSTem:DISPlay:IMAGe? [<invert_color>]
```

Description

Prints the screen and retrieves the print screen image data. The `<invert_color>` parameter is an optional parameter to invert the background color of the graph view. Setting the `<invert_color>` parameter to 1 will invert the background color from black to white.

Parameter

Item	Type	Range of values	Default value
`<invert_color>`	Boolean	0|1	0

Remarks

– The color inversion is only applicable to black background color image in the graph view.

– Inverting the background color will take quite a significant amount of time.

– The print screen image returned is in the JPEG format.

Examples

The following command retrieves the print screen image data without any background color inversion.

```
SYST:DISP:IMAG? 0
```

The following command retrieves the print screen image data with the background color inverted to white.

```
SYST:DISP:IMAG? 1
```

SYSTem:ERRor[:NEXT]?

Syntax

```
SYSTem:ERRor[:NEXT]?
```

Description

Returns the error number and its corresponding message string from the U8903B error queue. A record of up to 30 errors can be stored in the U8903B error queue. The format of the response is: `<error_number>,"<error_string>"` where the error number is defined in "**Error messages**" on page 611.

Remarks

- Errors are retrieved in the first-in, first-out (FIFO) order where the first error returned is the first error that has been stored.
- If more than 30 errors have occurred, the last error stored in the queue (the most recent error) is replaced with error -350,"Error Queue overflow". No additional errors are stored until you remove errors from the queue. If no error occur when you read the error queue, the U8903B responds with error 0,"No error".
- The error queues are cleared by the clear status (***CLS**) command and when power is cycled. The errors are also cleared when you read the error queue. The error queue is not cleared by a factory reset (***RST**) command or an instrument preset (**SYSTem:PRESet**) command.
- The command reads and clears one error string from the error queue. The error string may contain up to 255 characters and consists of an error number and an error string enclosed in double quotes. For example, error -113,"Undefined header".

Example

The following query reads and clears one error.

```
SYST:ERR?
```

Typical response:

```
-101,"Invalid character"
```

Error messages

Table 17-1 SCPI error messages: No error

<error_number>	<error_string>
0	No error The queue is completely empty. Every error or event in the queue has been read or the queue was purposely cleared by power on (***CLS**).

Table 17-2 SCPI error messages: Command errors

<error_number>	<error_string>
-100	Command error Generic syntax error.
-101	Invalid character An invalid character was found in the command string.
-102	Syntax error Invalid syntax was found in the command string. Check for blank spaces.
-103	Invalid separator An invalid separator was found in the command string. Check for proper use of comma (,), semicolon (;), and colon (:).
-104	Data type error A different data type than the one allowed was found in the command string.
-105	GET not allowed A group execute trigger is not allowed in a command string.
-108	Parameter not allowed More parameters were received than were expected.
-109	Missing parameter Fewer parameters were received than were expected.
-110	Command header error An error was detected in the header.
-111	Header separator error A character that was not a valid header separator was found in the command string.
-112	Program mnemonic too long The header contains more than 12 characters.
-113	Undefined header A command was received that was not valid for this instrument.
-114	Header suffix out of range The value of the numeric suffix is not valid.
-120	Numeric data error Generic numeric data error.

Table 17-2 SCPI error messages: Command errors (continued)

<error_number>	<error_string>
-121	Invalid character in number An invalid character for the data type was found in the command string.
-123	Exponent too large The magnitude of the exponent was larger than 32000.
-124	Too many digits The mantissa of a numeric parameter contained more than 255 digits, excluding leading zeros.
-128	Numeric data not allowed A numeric parameter was received but a character string was expected.
-130	Suffix error Generic suffix error.
-131	Invalid suffix A suffix was incorrectly specified for a numeric parameter.
-134	Suffix too long The suffix contains more than 12 characters.
-138	Suffix not allowed A suffix is not supported for this command.
-140	Character data error Generic character data error.
-141	Invalid character data Either the character data element contains an invalid character, or the element is not valid.
-144	Character data too long The character data element contains more than 12 characters.
-148	Character data not allowed A discrete parameter was received, but a string or numeric parameter was expected.
-150	String data error Generic string data error.
-151	Invalid string data An invalid character string was received. Check that the string is enclosed in quotation marks.
-158	String data not allowed A character string was received, but is not allowed for this command.
-160	Block data error Generic block data error.
-161	Invalid block data The number of data bytes sent does not match the number of bytes specified in the header.
-168	Block data not allowed Data was sent in arbitrary block format but is not allowed for this command.
-170	Expression error Generic expression error.

Table 17-2 SCPI error messages: Command errors (continued)

<error_number>	<error_string>
-171	Invalid expression data The expression data element was invalid.
-178	Expression data not allowed Expression data element was sent but is not allowed for this command.

Table 17-3 SCPI error messages: Execution errors

<error_number>	<error_string>
-200	Execution error Generic syntax error.
-210	Trigger error Error occured during triggering.
-211	Trigger Ignored Indicates that a ***TRG** command, or triggering signal was received and recognized by the device but was ignored because of device timing considerations.
-220	Parameter error A data element related error occurred.
-221	Settings conflict A data element could not be executed because of the present instrument state.
-222	Data out of range A data element could not be executed because the value was outside the valid range.
-223	Too much data A data element was received that contains more data than the instrument can handle.
-224	Illegal parameter value An exact value was expected but not received.
-225	Out of memory The device has insufficient memory to perform the requested operation.
-231	Data questionable The measurement accuracy is questionable.
-232	Invalid format The data format or structure is inappropriate.
-233	Invalid version The version of the data format is incorrect to the instrument.
-240	Hardware error The command could not be executed because of a hardware problem with the instrument.
-241	Hardware missing The command indicates that a legal program command or query could not be executed because of missing device hardware. For example, an option was not installed. Definition of what constitutes missing hardware is completely device-specific.

Table 17-3 SCPI error messages: Execution errors (continued)

<error_number>	<error_string>
-250	Mass Storage Error Generic error relating with mass storage.
-251	Missing Mass Storage The mass storage is not available.
-255	Directory Full The specified directory is full.
-256	File name not found The selected file is not found.
-257	File name error The file name is invalid.
-260	Expression execution error An expression program data element related error occurred.
-291	Out of Memory Error The memory is not sufficient to implement the command.

Table 17-4 SCPI error messages: Device-specific errors

<error_number>	<error_string>
-300	Device Specific Error This is the generic device dependent error for devices that cannot detect more specific errors. This code indicates only that a Device-Dependent Error as defined in IEEE 488.2, 11.5.1.1.6 has occurred.
-310	System error Indicates that some error, termed “system error” by the device, has occurred. This code is device-dependent.
-311	Memory error Indicates that an error was detected in the device's memory. The scope of this error is device-dependent.
-330	Self-test failed Indicates that the self-test of the instrument has failed. Refer to **Table 17-6** for the specified error that causes the self-test to fail.
-340	Calibration failed Indicates that the calibration of the instrument has failed.
-350	Error Queue Overflow The error queue is full because more than 20 errors have occurred. No additional errors are stored until you remove the errors from the queue. The error queue is cleared by the clear status (***CLS**) command or when the power is cycled. The errors are also cleared when you read the queue. The error queue is not cleared by an instrument reset (***RST**) command.

Table 17-5 SCPI error messages: Query errors

<error_number>	<error_string>
-400	Query Error Generic error query.
-410	Query INTERRUPTED A condition causing an interrupted query error occurred.
-420	Query UNTERMINATED A condition causing an unterminated query error occurred.
-430	Query DEADLOCKED A condition causing a deadlocked query error occurred.
-440	Query UNTERMINATED after indefinite response A query was received in the same program message after a query indicating an indefinite response was executed.

Table 17-6 SCPI error messages: Self-test errors

<error_number>	<error_string>
601	Front panel not found Possible cause: Unable to communicate with the front panel. The front panel firmware may be corrupted or front panel hardware malfunction.
602	Front panel faulty. Possible cause: The front panel malfunctioned.
603	LAN test failed Possible cause: The LAN adapter is not found or spoiled.
604	LAN device faulty Possible cause: The LAN adapter is found but malfunction.
605	USB device interface test failed Possible cause: The USB (device interface) device is not found or spoiled.
606	Analog analyzer card self-test in Slot <n> failed. <n> refers to slot 1 to slot 6.
607	Analog generator card self-test in Slot <n> failed. <n> refers to slot 1 to slot 6.
698	Self Test Time Out Possible cause: No response within the estimated period.
699	Unknown Error Possible cause: Unknown error.

SYSTem:HELP:HEADers?

Syntax

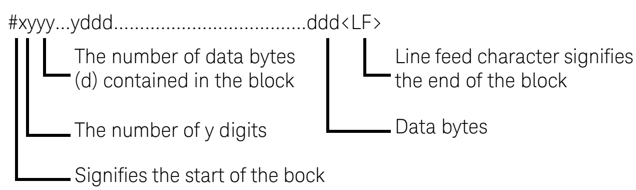
```
SYSTem:HELP:HEADers?
```

Description

Returns all SCPI commands and queries, and IEEE 488.2 common commands and queries implemented by the U8903B.

A SCPI header is defined as containing all the nodes from the root. The SCPI program mnemonic contains the node in a standard SCPI format. The short form will use uppercase characters while the additional characters for the long form will be in lowercase characters. Default nodes will be surrounded by square brackets “[]”.

Data is returned in IEEE 488.2 arbitrary block program data format as shown below.



Example: If there are 12345 data bytes, y = 12345, 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.

Figure 17-1 IEEE 488.2 arbitrary block program data format

SYSTem:LEGacy:CHANnel

Syntax

```
SYSTem:LEGacy:CHANnel <channel>
```

```
SYSTem:LEGacy:CHANnel?
```

Description

Sets the channel for the U8903B to emulate the HP8903B in the legacy mode. The query returns the legacy mode channel.

Parameter

Item	Type	Range of values	Default value
`<channel>`	Numeric	1|2	1

Remark

This command can only be used when the GPIB is successfully initialized.

Examples

The following command sets channel 1 as the channel to emulate the HP8903B in the legacy mode.

```
SYST:LEG:CHAN 1
```

The following query returns the legacy mode channel.

```
SYST:LEG:CHAN?
```

Typical response:

```
1
```

SYSTem:LEGacy:MODE

Syntax

```
SYSTem:LEGacy:MODE <state>
```

```
SYSTem:LEGacy:MODE?
```

Description

Enables or disables the legacy mode. The legacy mode will enable the U8903B to emulate the HP8903B and accept HP8903B commands. The query returns the legacy mode state.

Parameter

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	OFF

Remarks

- Some of the HP8903B commands are not supported. Refer to "**Appendix I: Sending HP8903B Commands to the U8903B**" on page 672 on the supported HP8903B commands.
- This command can only be used when the GPIB is successfully initialized.

Examples

The following command enables the legacy mode.

```
SYST:LEG:MODE ON
```

The following query returns the legacy mode state.

```
SYST:LEG:MODE?
```

Typical response:

```
1
```

SYSTem:LOCal

Syntax

```
SYSTem:LOCal
```

Description

Unlocks all the U8903B front panel keypad including the Local key and enables the U8903B to be controlled from the front panel.

Example

The following command returns the U8903B from the remote mode or the remote with lock mode to the local mode.

```
SYST:LOC
```

SYSTem:PRESet

Syntax

```
SYSTem:PRESet
```

Description

Presets the U8903B to the values depending on the preset type selected as follows.

`Factory`	Resets the instrument to the factory default settings and deletes all user-defined files.
`User`	Restores the instrument to the user-defined state that has been saved using the `SYST:PRESet:SAVE` command.

Remark

If you preset the U8903B with the User preset type without an existing User preset, or if there are any errors when loading the User preset, an error will be displayed.

Examples

The following command saves the current user settings and presets to the user-defined state.

```
SYST:PRES:SAVE
SYST:PRES:TYPE USER
SYST:PRESet
```

The following command presets the U8903B to the factory default settings.

```
SYST:PRES:TYPE FACTory
SYST:PRESet
```

SYSTem:PRESet:SAVE

Syntax

```
SYSTem:PRESet:SAVE
```

Description

Saves the current user settings that will be used in the preset.

Example

The following command saves the current user settings.

```
SYST:PRES:SAVE
```

SYSTem:PRESet:TYPE

Syntax

```
SYSTem:PRESet:TYPE <type>
```

Description

Sets the U8903B to the values depending on the preset type selected.

Parameters

Item	Type	Range of values	Default value
`<type>`	Discrete	FACTory|USER	FACTory

Remark

The ***RST** command will not reset this setting.

Example

The following command sets the preset type to User.

```
SYST:PRES:TYPE USER
```

SYSTem:REMote

Syntax

```
SYSTem:REMote
```

Description

Activates the remote mode. Locks the U8903B front panel keypad excluding the Local key, and displays the remote icon on the LCD display. Local front panel operation can be enabled by pressing the Local key.

Remark

Remote mode can also be activated by sending any SCPI commands.

Example

The following command returns the U8903B from the local mode to the remote mode.

```
SYST:REM
```

SYSTem:RESet[:MODE]

Syntax

```
SYSTem:RESet[:MODE] <system_mode>
```

Description

Resets the customized settings of the specified U8903B system mode to the default settings.

Parameter

Item	Type	Range of values	Default value
<system_mode>	Discrete	AANalyzer|AGENerator|SWEep|GRAPh|DANalyzer|DGENerator	Required parameter

Remarks

– This command resets the customized settings of the selected system mode excluding the stored files, I/O configuration, and common system settings.

– For the analog analyzer mode, the measurement bandwidth, measurement time, and trigger source will also be reset to the default settings.

– Sending the `SYST:RES SWE` command will perform the same action as the ***RST** command for firmware v 3.x.x.x and above. Do not use the `SYST:RES SWE` command for new development.

Example

The following command resets the analog analyzer mode.

```
SYST:RES AAN
```

SYSTem:RWLock

Syntax

```
SYSTem:RWLock
```

Description

Activates the remote with lock mode. Locks all the U8903B front panel keypad including the Local key, and displays the remote icon and the keypad lock icon on the LCD display.

The U8903B cannot return to the manual control from the front panel. This state can be cleared by sending the **SYSTem:LOCal** command.

Example

The following command locks all the U8903B front panel keys.

```
SYST:RWL
```

SYSTem:TIME

Syntax

```
SYSTem:TIME <hh>,<mm>,<ss>
```

```
SYSTem:TIME?
```

Description

Sets the real time clock in hours (hh), minutes (mm), and seconds (ss). The query returns comma-separated values that correspond to the hour, minute, and seconds.

Parameters

Item	Type	Range of values	Default value
`<hh>`	Numeric	An integer from 0 to 23	Required parameter
`<mm>`	Numeric	An integer from 0 to 59	Required parameter
`<ss>`	Numeric	An integer from 0 to 59	Required parameter

Examples

The following command sets the time.

```
SYST:TIME 13,30,10
```

The following query returns the time.

```
SYST:TIME?
```

Typical response:

```
13,30,10
```

SYSTem:UPDate:FIRMware?

Syntax

```
SYSTem:UPDate:FIRMware? <filename>
```

Description

The query performs the U8903B firmware update process based on the specified file name and returns the firmware update process state as 0 if the update process is not completed, or 1 if the update process is completed successfully without error.

Parameter

Item	Type	Range of values	Default value
`<filename>`	String	Full file path in quoted string. For example, “\Storage 1\firmware.bin”	Required parameter

Remarks

- Before starting the update, ensure that the firmware file is accessible.
- For external USB flash storage, the file path must begin with ‘\Storage x’ where x is the USB flash storage number that are plugged into the U8903B in sequence.
- The time required to complete this query varies according to the firmware file and may take 20 to 40 minutes.
- All measurements and waveform generation will be stopped automatically before the update process.
- After the firmware update process is completed, the U8903B will automatically restart.

Example

The following query updates the firmware with the firmware file (firmware.bin) stored in the external USB flash storage.

```
SYST:UPD:FIRM? “\Storage 1\firmware.bin”
```

Typical response:

```
1
```

SYSTem:VERSion?

Syntax

```
SYSTem:VERSion?
```

Description

Returns the SCPI standard version with which the U8903B is in compliant. The U8903B complies with the rules and conventions of the indicated SCPI standard version. The response format is in the form of XXXX.Y, where XXXX represents the year of the version and Y represents the version number for that year.

Example

The following query returns the SCPI version.

```
SYST:VERS?
```

Typical response:

```
1999.0
```

18 TRIGger Subsystem

This chapter describes the `TRIGger` subsystem commands.

TRIGger:CHANnel

Syntax

```
TRIGger:CHANnel <channel>
TRIGger:CHANnel?
```

Description

Sets the interface type of the channel to be triggered. The query returns the channel number.

Parameter

Item	Type	Range of values	Default value
`<channel>`	Numeric	1 to 8 (analog) 1 to 2 (digital)	1

Remark

- This setting is only applicable for trigger source set to `channel`.
- When changing the trigger source to channel, if the `<channel>` is invalid, the value will be set to the lowest available channel.

Examples

The following command sets the trigger channel to channel 3.

```
TRIGGER:CHANNEL 3
```

The following query returns the trigger channel.

```
TRIGGER:CHANNEL?
```

Typical response:

```
3
```

TRIGger:INTerface

Syntax

```
TRIGger:INTerface <interface>
```

```
TRIGger:INTerface?
```

Description

Sets the the interface type of the channel to be triggered. The query returns the trigger interface value.

Parameter

Item	Type	Range of values	Default value
`<interface>`	Discrete	ANALog|DIGital	ANALog

Remark

- This setting is only applicable for trigger source set to `channel`.
- When switching interface, if the current channel value for the **TRIGger:CHANnel** command is not valid, the value will change to the lowest available channel.

Examples

The following command sets the trigger signal to analog .

```
TRIG:INT ANAL
```

The following query returns the trigger interface of the signal.

```
TRIG:INT?
```

Typical response:

```
ANAL
```

TRIGger:LEVel[:ANALog]

Syntax

```
TRIGger:LEVel[:ANALog] <level>
TRIGger:LEVel[:ANALog]?
```

Description

Sets the threshold level of the trigger for the analog analyzer. The U8903B analysis begins when the incoming signal of the specified trigger channel exceeds the threshold level. The query returns the analog trigger level value.

Parameter

Item	Type	Range of values	Default value
`<level>`	Numeric	0 to 140	0

Remark

This setting is only applicable for trigger source set to `channel` and trigger interface set to analog.

Examples

The following command sets trigger level for the analog analyzer to 1.

```
TRIGger:LEVel 1
```

The following query returns the trigger level.

```
TRIGger:LEVel?
```

Typical response:

```
1
```

TRIGger:LEVel:DIGital

Syntax

```
TRIGger:LEVel:DIGital <level>
```

```
TRIGger:LEVel:DIGital?
```

Description

Sets the threshold level of the trigger for the digital analyzer. The U8903B analysis begins when the incoming signal of the specified trigger channel exceeds the threshold level. The query returns the digital trigger level value.

Parameter

Item	Type	Range of values	Default value
`<level>`	Hex	00000000 to FFFFFFFF	0

Remarks

This setting is only applicable for trigger source set to `channel` and trigger interface set to digital.

The value is constrained by the DSI word length and must be prefixed with #H to indicate that the value is in hexadecimal.

Examples

The following command sets trigger level for the digital analyzer to ABCD.

```
TRIGGER:LEVEL:DIGITAL #HABCD
```

The following query returns the trigger level.

```
TRIGGER:LEVEL:DIGITAL?
```

Typical response:

```
#HABCD
```

TRIGger:SLOPe

Syntax

```
TRIGger:SLOPe <edge>
```

```
TRIGger:SLOPe?
```

Description

Sets the trigger edge of the input signal. The query returns the trigger edge of the input signal.

Parameter

Item	Type	Range of values	Default value
`<edge>`	Discrete	POSitive|NEGative	POSitive

Remark

This setting is only applicable for trigger source set to `EXTernal`.

Examples

The following command sets the trigger edge of the signal to the rising edge.

```
TRIG:SLOP POS
```

The following query returns the trigger edge of the signal.

```
TRIG:SLOP?
```

Typical response:

```
POS
```

TRIGger:SOURce

Syntax

```
TRIGger:SOURce <trigger_source>
```

```
TRIGger:SOURce?
```

Description

Sets the input trigger source. The query returns the input trigger source in the form of either `IMM`, `EXT`, or `BUS`.

The description for each <trigger_source> parameter is listed as follows.

`IMMediate`	Free run
`EXTernal`	Triggers from an external source
`BUS`	Triggers from the internal bus

Parameter

Item	Type	Range of values	Default value
`<trigger_source>`	Discrete	IMMediate|EXTernal|BUS	IMMediate

Examples

The following command sets the input trigger source to external.

```
TRIG:SOUR EXT
```

The following query returns the input trigger source.

```
TRIG:SOUR?
```

Typical response:

```
EXT
```

THIS PAGE HAS BEEN INTENTIONALLY LEFT BLANK.

19 TSEQuence Subsystem

This chapter describes the `TSEQuence` subsystem commands.

TSEQuence:PROJect:DUT:ID

Syntax

```
TSEQuence:PROJect:DUT:ID <id>
TSEQuence:PROJect:DUT:ID?
```

Description

Sets the DUT ID for the test sequence mode. The query returns the DUT ID in quoted string.

Parameter

Item	Type	Range of values	Default value
`<id>`	String	Can be any letters (A to Z or a to z), numbers (0 to 9) or special characters (“@|%|*”). Blank spaces are not allowed.	Required parameter

Remark

The test sequence must be in the STOP state before this command is allowed.

Examples

The following command sets the DUT ID for the test sequence mode.

```
TSEQ:PROJ:DUT:ID “DUT1”
```

The following query returns the DUT ID for the test sequence mode.

```
TSEQ:PROJ:DUT:ID?
```

Typical response:

```
“DUT1”
```

TSEQuence:STATe

Syntax

```
TSEQuence:STATe <state>
TSEQuence:STATe?
```

Description

Sets the state of the test sequence mode. The query returns the test sequence mode state.

NOTE Do not perform other operations when the test sequence is being performed as this may cause unexpected results.

Parameter

Item	Type	Range of values	Default value
<state>	Discrete	RUN|PAUSe|STOP|CONTinue	STOP

Remarks

- If the ***RST** command is received from the remote controller, via the device interface, the test sequence mode will be stopped.
- The following matrix defines the effect of setting the state to the desired value from each of the possible current states. In certain cases, error -221,“Settings Conflict” will be generated.
  - Running: The test sequence mode is currently executing.
  - Paused: The test sequence mode has reached a break in execution but can be continued.
  - Stopped: Execution has been terminated.

State sent	Current state		
	Running	Paused	Stopped
RUN	Error -221	Running	Running
CONTinue	Error -221	Running	Error -221
PAUSe	PAUSed	Paused	Stopped

– Setting the test sequence to the RUN state will switch the front panel display to the test sequence mode automatically.

Examples

The following command sets the test sequence mode state to RUN.

```
TSEQ:STAT RUN
```

The following query returns the test sequence mode state.

```
TSEQ:STAT?
```

Typical response:

```
RUN
```

20 Programming Examples

This chapter provides programming sequence examples to remotely control the U8903B using SCPI commands.

Example 1: Generate Normal Sine Waveform

The following command sequence provides an example on how to generate a normal 3 kHz, 2 Vrms sine waveform on the analog generator channel 1.

`SOUR:FUNC SINE,(@1)`	// Sets the analog generator waveform type to sine on channel 1.
`SOUR:VOLT 2Vrms,(@1)`	// Sets the amplitude to 2 Vrms.
`SOUR:FREQ1 3kHz,(@1)`	// Sets the frequency to 3 kHz.
`OUTP:STAT ON,(@1)`	// Turns on the analog generator channel 1 output.

Example 2: Generate Multitone Waveform

The following command sequence gives an example on how to generate a multitone waveform on the analog generator channel 1.

`SOUR:FUNC MULT,(@1)`	// Sets the analog generator waveform type to multitone on channel 1.
`SOUR:MULT:WLEN L32768,(@1)`	// Sets the multitone waveform length to 32768 points.
`SOUR:MULT:FREQ:STAR 500Hz,(@1)`	// Sets the multitone waveform start frequency to 500 Hz.
`SOUR:MULT:FREQ:STOP 5kHz,(@1)`	// Sets the multitone waveform stop frequency to 5 kHz.
`SOUR:MULT:COUN 20,(@1)`	// Sets the multitone waveform number of tones to 20.
`SOUR:MULT:FREQ:SPAC LOG,(@1)`	// Sets the multitone waveform frequency spacing type to logarithmic.
`SOUR:VOLT 2Vrms,(@1)`	// Sets the amplitude to 2 Vrms.
`OUTP:STAT ON,(@1)`	// Turns on the analog generator channel 1 output.

Example 3: Generate Arbitrary Waveform

The following examples show how to generate a 3 Vp, 0 V offset arbitrary waveform on the generator channel 1.

NOTE The arbitrary data must be uploaded before setting the waveform type to arbitrary.

Example A

The following command sequence provides an example to generate an arbitrary waveform by uploading the data stream from the PC to the U8903B.

`DATA:WAV 3,0,<data>`	// Uploads the arbitrary data into the U8903B. The `<data>` parameter is in the IEEE-488.2 binary block program data format.
`SOUR:FUNC ARB,(@1)`	// Sets the analog generator waveform type to arbitrary on channel 1.
`OUTP:STAT ON,(@1)`	// Turns on the analog generator channel 1 output.

Example B

The following command sequence provides an example to generate an arbitrary waveform by loading the arbitrary waveform data into the U8903B from the “waveform1.arb” file in the external USB flash storage.

`MMEM:LOAD WAV, “\Storage 1\waveform1.arb”`	// Loads the arbitrary waveform data into the U8903B from the “waveform1.arb” file in the external USB flash storage.
`SOUR:FUNC ARB,(@1)`	// Sets the analog generator waveform type to arbitrary on channel 1.
`OUTP:STAT ON,(@1)`	// Turns on the analog generator channel 1 output.

If you wish to change the amplitude of the waveform from 3 Vp to 5 Vp, send the `SOUR:VOLT 5Vp,(@1)` command to change the amplitude to 5 Vp without the need to reupload the arbitrary data. This also applies to changing the value of the DC offset.

Example 4: Basic Measurements

The following command sequence provides an example on how to measure the frequency and amplitude using the analog analyzer.

`SENS:FUNC1 FREQ,(@1)`	// Sets the measurement function 1 to frequency on the analog analyzer channel 1.
`SENS:FUNC2 VAC,(@1)`	// Sets the measurement function 2 to amplitude on the analog analyzer channel 1.
`INIT:ANAL (@1)`	// Initiates the frequency and amplitude measurements on the analog analyzer channel 1.
`STAT:OPER:COND?`	// Queries the measurement status. The query returns `0` if the measurement is completed. To ensure that you get a valid result, you must keep sending this command until you get a `0`.
`FETC? FUNC1,(@1)`	// Acquires the function 1 frequency measurement result for channel 1.
`FETC? FUNC2,(@1)`	// Acquires the function 2 amplitude measurement result channel 1.

Example 5: Measure the Crosstalk

The following command sequence provides an example on how to measure crosstalk. The designated reference channel will be injected with the stimulus and the presence of this signal in the other channel will be measured.

`SENS:REF:CHAN 1`	// Sets the driven channel to channel 1 for the crosstalk measurement.
`SENS:FUNC:MCH XTD`	// Sets the multi-channel function mode to crosstalk.
`INIT:ANAL (@1,2)`	// Initiates the crosstalk measurement on the analog analyzer channel 1 and channel 2
`STAT:OPER:COND?`	// Queries the measurement status. The query returns `0` if the measurement is completed. To ensure that you get a valid result, you must keep sending this command until you get a `0`.
`FETC? FUNC3,(@2)`	// Acquires the function 3 crosstalk measurement result for channel 2.

Example 6: Continuous Graph Measurement

Example A

The following command sequence gives you an example on how to perform a continuous graph measurement.

`DISP:ANAL:TYPE MAGN`	// Sets the graph analysis mode to magnitude.
`INIT:CONT:GRAP ON,(@1)`	// Initiates the graph analysis mode continuous measurement on channel 1.
`INIT CONT:GRAP? (@1)`	// Queries the graph analysis mode channel 1 continuous measurement status.
`FETC:ARR? (@1)`	// Acquires the array of measurement data for channel 1.

Example B

The following command sequence gives you an example on how to perform a continuous graph measurement with the measurement result.

`DISP:ANAL:TYPE MAGN`	// Sets the graph analysis mode to magnitude.
`INIT:CONT:GRAP (@1), MEAS`	// Initiates the graph analysis mode continuous measurement on channel 1.
`INIT CONT:GRAP? (@1)`	// Queries the graph analysis mode channel 1 continuous measurement status.
`FETC:ARR? (@1)`	// Acquires the array of measurement data for channel 1.
`FETC? FUNC1,(@1)`	// Acquires the measurement result for channel 1 function 1.

Example 7: Single Graph Measurement

Example A

The following command sequence gives you an example on how to perform a single graph measurement.

`DISP:ANAL:TYPE MAGN`	// Sets the graph analysis mode to magnitude.
`INIT:GRAP (@1)`	// Initiates the graph analysis mode measurement on channel 1.
`STAT:OPER:COND?`	// Queries the condition register. Keep sending this command until you get a `0`.
`FETC:ARR? (@1)`	// Acquires the array of measurement data for channel 1.

Example B

The following command sequence gives you an example on how to perform a single graph measurement with the measurement result.

`DISP:ANAL:TYPE MAGN`	// Sets the graph analysis mode to magnitude.
`INIT:GRAP (@1), MEAS`	// Initiates the graph analysis mode measurement on channel 1.
`STAT:OPER:COND?`	// Queries the condition register. Keep sending this command until you get a `0`.
`FETC:ARR? (@1)`	// Acquires the array of measurement data for channel 1.
`FETC? FUNC1,(@1)`	// Acquires the measurement result for channel 1 function 1.

Example 8: Perform Sweep

Example A

Frequency response is a very common test. The sweep feature of U8903B can be utilized to perform this analysis. To perform a frequency response analysis of your DUT, you can connect your DUT to any generator channel and the corresponding analyzer channel. In this example, the DUT must be connected to the analog generator channel 1 and analog analyzer channel 2.

The following command sequence provides an example to perform an automatic linear sweep on a 5 Vp sine waveform on analog generator channel 1, from 100 Hz to 1000 Hz with a step size of 200 Hz and 1 s dwell time to stabilize the DUT. The DUT signal amplitude is measured.

`SOUR:FUNC SINE,(@1)`	// Sets the analog generator waveform type to sine on channel 1.
`SOUR:VOLT 5Vp,(@1)`	// Sets the amplitude of the sine waveform to 5 Vp.
`SOUR:SWE:REF:CHAN 1`	// Sets the sweep reference channel of the analog generator to channel 1.
`SOUR:SWE:CHAN 1`	// Sets analog generator channel 1 to perform sweep.
`SOUR:SWE:MODE ASW`	// Sets the sweep mode to auto sweep.
`SOUR:SWE:PAR FREQ1`	// Sets the sweep parameter to frequency.
`SOUR:SWE:SPAC LIN,(@1)`	// Sets the sweep spacing type to linear.
`SOUR:SWE:DWEL1 1000`	// Sets the sweep dwell time to 1 s (1000 ms).
`SOUR:SWE:STAR 100`	// Sets the sweep start value to 100 Hz.
`SOUR:SWE:STOP 1000`	// Sets the sweep stop value to 1000 Hz.
`SOUR:SWE:STEP 200`	// Sets the sweep step size to 200 Hz.
`SENS:SWE:REF:CHAN 1`	// Sets the sweep reference channel for measurement to channel 1.
`SENS:SWE:CHAN 2`	// Sets the analog analyzer channel to perform sweep to channel 2.
`SENS:FUNC1 FREQ,(@2)`	// Sets the analog analyzer measurement function 1 to frequency on channel 1.
`SENS:FUNC2 VAC,(@2)`	// Sets the analog analyzer measurement function 2 to VAC on channel 1.
`INIT:SWE`	// Initiates the sweep.

`STAT:OPER:COND?`	// Polls the status register to check if the measuring operation has completed. The condition register will return `0` if the operation has completed.
`SOUR:SWE:VAL? (@2)`	// Acquires the X-axis sweep points values.
`FETC:SWE? FUNC1,(@2)`	// Acquires the function 1 sweep result for channel 2.
`FETC:SWE? FUNC2,(@2)`	// Acquires the function 2 sweep result for channel 2.

Example B

The following command sequence provides an example to perform a manual log sweep on a 5 Vrms sine waveform on channel 2, from 100 Hz to 10 kHz with a 10 ms dwell time. The number of points to sweep is 20 points. The signal amplitude is measured at the analog analyzer channel 2.

`SOUR:FUNC SINE,(@2)`	// Sets the analog generator waveform type to sine on channel 2.
`SOUR:VOLT 5Vrms,(@2)`	// Sets the amplitude of the sine waveform to 5 Vrms.
`SOUR:SWE:REF:CHAN 1`	// Sets the sweep reference channel of the analog generator to channel 1.
`SOUR:SWE:CHAN 2`	// Sets channel 2 to perform sweep.
`SOUR:SWE:MODE MSW`	// Sets the sweep mode to manual sweep.
`SOUR:SWE:PAR FREQ1`	// Sets the sweep parameter to frequency.
`SOUR:SWE:SPAC LOG`	// Sets the sweep spacing type to logarithmic.
`SOUR:SWE:DWEL 10`	// Sets the sweep dwell time to 10 ms.
`SENSe:SAMPle:SIZE GTR,(@1)`	// Sets the analog analyzer channel 1 sample size to generator track.
`SOUR:SWE:STAR 100`	// Sets the sweep start value to 100 Hz.
`SOUR:SWE:STOP 10000`	// Sets the sweep stop value to 10 kHz.
`SOUR:SWE:POIN 20`	// Sets the sweep points to 20.
`SENS:SWE:INT ANAL`	// Sets the sweep analyzer interface to analog.
`SENS:SWE:REF:CHAN 1`	// Sets the sweep reference channel for measurement to channel 1.
`SENS:SWE:CHAN 2`	// Sets the analog analyzer channel to perform sweep to channel 2.
`SENS:FUNC1 FREQ,(@2)`	// Sets the analog analyzer measurement function 1 to frequency.

`SENS:FUNC2 VAC,(@2)`	// Sets the analog analyzer measurement function 2 to VAC.
`INIT:SWE`	// Initiates the sweep.
`STAT:OPER:COND?`	// Polls the status register to check if the measuring operation has completed. The condition register will return `0` if the operation has completed.
`FETC:SWE? FUNC1,(@2)`	// Acquires the sweep result. For manual sweep, only a single result will be returned each time this query is sent.
`FETC:SWE? FUNC2,(@2)`	// Acquires the sweep result.
`SOUR:SWE:NEXT`	// Jumps to the next sweep point.
`FETC:SWE? FUNC1, (@2)`	// Acquires the sweep result for the current point.

Use the **SOURce:SWEep:NEXT** command and **FETCh:SWEep?** query to obtain the sweep results for the rest of the 20 sweep points.

Example 9: Use the User-Defined Filter Data

This section describes the methods to load a custom filter into the U8903B by downloading the filter coefficients through SCPI or by loading the filter coefficients from a custom filter file stored in the U8903B. To create your own custom filter, you need to configure your custom filter data using an external software.

Example A

The following command sequence provides an example on how to download a custom filter coefficients through SCPI. Assume that you wish to load an IIR low-pass filter (two sections and three group delays) to the U8903B at channel 1, as well as a FIR low-pass filter (one section and ten group delays) at channel 2.

`DATA:FILT IIR,2,3,<data>`	// Sends the IIR custom filter data to the U8903B volatile memory allocated for the user-defined filter data. Refer to "**Appendix H: Using the IEEE-488.2 Binary Block Format**" on page 671 for the `<data>` parameter.
`SENS:FILT:LPAS CUST,(@1)`	// Sets the low-pass filter to user-defined at channel 1.
`DATA:FILT FIR,1,10,<data>`	// Downloads the FIR custom filter data into the U8903B.
`SENS:FILT:LPAS CUST,(@2)`	// Sets the low-pass filter to user-defined at channel 2.

The custom filters for channel 1 and channel 2 are now ready to be used in the analyzer mode.

Example B

The following command sequence provides an example on how to load a custom filter coefficients from a custom filter file stored in the U8903B. Assume that you wish to use a custom low-pass filter stored in the U8903B at both the analyzer channel 1 and 2.

`MMEM:LOAD FILT, "\Filter\LPF80kHz_MF_HIGH.juf"`	// Loads the filter coefficients from the file "LPF80kHz_MF_HIGH.juf".
`SENS:FILT:LPAS CUST,(@1,2)`	// Sets the low-pass filter to user-defined at channel 1 and channel 2.

The custom filters for channel 1 and channel 2 are now ready to be used in the analyzer mode.

Example C

The following command sequence provides an example on how to load two different custom filters for the analyzer channel. Assume that you wish to use a custom low-pass filter at analyzer channel 1 where the file “LPF80kHz_MF_HIGH.juf” is stored in the U8903B and a custom high-pass filter at analyzer channel 1 where the file “myHighPass.juf” is stored in the external USB flash storage.

`MMEM:LOAD FILT, “\Filter\LPF80kHz_MF_HIGH.juf”`	// Loads the filter coefficients from the file “LPF80kHz_MF_HIGH.juf”.
`SENS:FILT:LPAS CUST,(@1)`	// Sets the low-pass filter to user-defined at channel 1.
`MMEM:LOAD FILT, “\Storage 1\myHighPass.juf”`	// Loads the filter coefficients from the file “myHighPass.juf”.
`SENS:FILT:HPAS CUST,(@1)`	// Sets the high-pass filter to user-defined at channel 1.

The custom filters for channel 1 are now ready to be used in the analyzer mode.

Example 10: Make a Relative Measurement

This section describes the methods to obtain a relative measurement based on the previous measurement result of the same channel or other channels by changing the data format rather than using an absolute measurement data.

The following command sequence provides an example to perform a relative measurement for VAC function in analyzer mode. The data measured by channel 1 will be set as a reference level for the subsequent measurement data of channel 2.

`*RST`	// Resets the U8903B.
`SENS:FUNC2 VAC,(@1)`	// Sets the analog analyzer measurement function 2 to VAC on channel 1.
`INIT:ANAL (@1)`	// Initiates the analog analyzer measurement trigger system to take the measurement directly on channel 1.
`STAT:OPER:COND?`	// Queries the measurement status. The query returns `16` when the analog analyzer measurement is in progress and bit 4 of the Standard Operation register is set. When the measurement data is ready, bit 4 of the Standard Operation register is cleared. If there are no other operation in progress, the query returns `0`.
`SENS:REF:RES:SET FUNC2,CH1,(@2)`	// Sets the last measurement result obtained from function 1 of the channel 1 as the reference level for channel 2 as the function 2 of the channel 1 is measuring VAC which is a level-based measurement.
`CALC:READ:FORM LOG,FUNC2,(@2)`	// Sets the function 2 channel 2 (VAC) measurement format to logarithmic.
`INIT:ANAL (@2)`	// Initiates the analog analyzer measurement trigger system to take the measurement directly on channel 2.
`STAT:OPER:COND?`	// Queries the measurement status. The query returns `16` when the analog analyzer measurement is in progress and bit 4 of the Standard Operation register is set. When the measurement data is ready, bit 4 of the Standard Operation register is cleared. If there are no other operation in progress, the query returns `0`.
`SENS:FUNC2:UNIT? (@2)`	// Acquires the unit of the measurement function 2 for channel 2.
`FETCH? FUNC2,(@2)`	// Acquires the measurement function 2 data for channel 2 that has been set relative to the reference level.

NOTE The acquired data is in dBr due to the **CALCulate[:ANALog]:READing:FORMat** command, which sets the level measurement format to logarithmic.

Example 11: Record Input Signal to Wave File

The following command sequence provides an example on how to record the input signal of the analog analyzer channel 1 to a wave file using SCPI commands. Assume that you wish to record a 20 second signal with stereo 24 bits into a wave file named “MyWavFile.wav” and store the file in the external USB flash storage.

`SENS:WAVF:BPS 24`	// Sets the bits per sample of the input channel to be recorded into the wave file to 24 bits.
`SENS:WAVF:CHAN STER`	// Sets the number of input channel to be recorded into the wave file to stereo.
`SENS:WAVF:DUR 20`	// Sets the duration of the recording to 20 s.
`INIT:WAVF:REC “\Storage 1\MyWavFile.wav”,(@1)`	// Initiates the recording of the input signal of analog analyzer channel 1 to a file named “MyWavFile.wav” in the external USB flash storage. The recording will stop when the duration is reached.
`STAT:OPER:COND?`	// Queries the recording status. The query returns `0` if the recording is completed. To ensure that the recording is completed, you must keep sending this command until you get a `0`.

Example 12: Obtain Statistics Data from the Analyzer

The following command sequence provides an example on how to obtain the calculated statistics data. Assume that you wish to collect five analog analyzer measurement readings to calculate for the minimum, maximum, and standard deviation for the function 2 of the channel 1 (for example, VAC).

```
CALC:STAT:STAT ON,(@1)          // Turns on the statistics on analog analyzer channel 1.
CALC:STAT:RES (@1)              // Resets the statistics data on analog analyzer channel 1.
SENS:FUNC2 VAC,(@1)             // Sets the analog analyzer measurement function 2 to AC
                                   voltage on channel 1.
CALC:STAT:COUN 5,(@1)           // Sets the number of analog analyzer measurement readings
                                   to 5 for statistics function calculation.
CALC:STAT:TYPE1 MIN,(@1)        // Sets the analog analyzer channel 1 first statistics type
                                   to minimum.
CALC:STAT:TYPE2 MAX,(@1)        // Sets the analog analyzer channel 1 second statistics type
                                   to maximum.
CALC:STAT:TYPE3 SD,(@1)         // Sets the analog analyzer channel 1 third statistics type to
                                   standard deviation.
INIT:CONT:ANAL ON,(@1)          // Starts the continuous measurements on channel 1.
CALC:STAT:DATA1? FUNC2,(@1)     // Obtains the results of the first statistics type (minimum) for
                                   channel 1 of the analog analyzer function 2 (VAC).
CALC:STAT:DATA2? FUNC2,(@1)     // Obtains the results of the second statistics type (maximum)
                                   for channel 1 of the analog analyzer function 2 (VAC).
CALC:STAT:DATA3? FUNC2,(@1)     // Obtains the results of the third statistics type for channel 1
                                   of the analog analyzer function 2 (VAC).
```

THIS PAGE HAS BEEN INTENTIONALLY LEFT BLANK.

Keysight U8903B
Audio Analyzer

Programmer's Reference

Appendixes

KEYSIGHT
TECHNOLOGIES

Appendix A: Waveform Frequency Range and Default Values

Table A-1 Analog waveform frequency range and default values

Waveform	Frequency range	Default
Sine	Frequency 1: 5 Hz to 80 kHz Frequency 2: -	1 kHz 0
Dual sine	Frequency 1: 5 Hz to 80 kHz Frequency 2: 5 Hz to 80 kHz	1 kHz 2 kHz
Variable phase	Frequency 1: 5 Hz to 80 kHz Frequency 2: -	1 kHz 0
SMPTE IMD 1:1 SMPTE IMD 4:1 SMPTE IMD 10:1	Frequency 1 (lower frequency): 40 Hz to 500 Hz Frequency 2 (upper frequency): 2 kHz to 60 kHz	60 Hz 7 kHz
DFD IEC 60118	Difference frequency: 80 Hz to 2 kHz Frequency 2 (upper frequency): 3 kHz to 80 kHz	80 Hz 10 kHz
DFD IEC 60268	Difference frequency: 80 Hz to 2 kHz Center frequency: 3 kHz to 79 kHz	80 Hz 10 kHz
Gaussian	-	-
Rectangular	-	-
DC	-	-
Multitone	Start frequency: 5 Hz to 80 kHz Stop frequency: 5 Hz to 80 kHz Tone’s frequency: 5 Hz to 80 kHz	1 kHz 5 kHz Depends on the spacing
Square	Frequency 1: 5 Hz to 30 kHz Frequency 2: –	1 kHz 0
Arbitrary	-	-

Appendix B: Units of the Measurement Function Returned Values

Analog analyzer

Table A-2 Analog analyzer units of the measurement function returned values

Measurement function	Unit	Default
Frequency	Hz ΔHz	Hz
AC voltage THD+N level THD level	dBg dBm dBr dBu dBV W V ΔV dBSPL x	V (AC voltage) dBV (THD+N level and THD level)
DC voltage	V ΔV x	V
THD+N ratio SINAD THD ratio SMPTE IMD DFD IEC 60118 (2nd order) DFD IEC 60118 (3rd order) DFD IEC 60268 (2nd order) DFD IEC 60268 (3rd order) SNR SNR (Fast) Crosstalk	dB ΔdB % x	dB
Phase	°	°

Digital analyzer

Table A-3 Digital analyzer units of the meaurement function returned values

Measurement function	Unit	Default
Frequency	– Hz – ΔHz	Hz
AC voltage Max peak value Min peak value	– V – dBFS – dBr – dBu – dBV – FFS – x – pctFS – LSB – Hex – Dec	FFS
DC voltage	– FFS – V – Hex – x	FFS
THD+N level THD level	– V – dBFS – dBr – dBu – dBV – FFS – x – pctFS – LSB – Hex – Dec	dBFS
THD+N ratio SINAD SMPTE IMD DFD IEC 60118 (2nd order) DFD IEC 60118 (3rd order) DFD IEC 60268 (2nd order) DFD IEC 60268 (3rd order) Crosstalk (channel driven)	– dB – ΔdB – % – x	dB
Phase	°	°

The units can be computed using the following formulas.

Table A-4 Unit conversion formula

Unit	Formula	Description
dHz	$f - f_{ref}$	f_{ref} = Reference frequency
dB	$20 \times \log_{10}(\text{ratio})$	-
ddB	$(\text{ratio}) - R_{ref}$	R_{ref} = Reference ratio
dBu	$20 \times \log_{10}\left(\frac{V_{rms}}{\sqrt{0.6}}\right)$	-
dBV	$20 \times \log_{10}(V_{rms})$	-
dBm	$10 \times \log_{10}\left(\frac{1000\, V_{rms}{}^2}{Z_{ref}}\right)$	Z_{ref} = Reference impedance
dBr	$20 \times \log_{10}\left(\frac{V_{rms}}{V_{ref}}\right)^2$	V_{ref} = Reference level
dBg	$20 \times \log_{10}\left(\frac{V_{rms}}{V_{gen}}\right)^2$	V_{gen} = Amplitude of the generator signal for a corresponding channel
W	$\frac{V}{Z_{ref}}$	Z_{ref} = Reference impedance
x	$\frac{V}{V_{ref}}$ or $\frac{\text{Ratio (in \%)}}{R_{ref}\text{ (in \%)}}$	V_{ref} = Reference level R_{ref} = Reference ratio
PCT (%)	$100 \times (\text{ratio})$	-

Appendix C: Waveform Parameters

Table A-5 Analog generator waveform parameters

Waveform	Parameter	SCPI command
Sine	Frequency	**SOURce[:ANALog]:DTMF:VOLTage:SUMMation**
	Amplitude	**SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate][:AMPLitude]**
	DC offset	**SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate]:OFFSet**
Dual sine	Frequency 1	**SOURce[:ANALog]:DTMF:VOLTage:SUMMation**
	Frequency 2	**SOURce[:ANALog]:DTMF:VOLTage:SUMMation**
	Amplitude	**SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate][:AMPLitude]**
	Ratio	**SOURce[:ANALog]:VOLTage:RATio**
	DC offset	**SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate]:OFFSet**
Variable phase	Frequency	**SOURce[:ANALog]:DTMF:VOLTage:SUMMation**
	Amplitude	**SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate][:AMPLitude]**
	Phase	**SOURce[:ANALog]:PHASe[:ADJust]**
SMPTE IMD 1:1	Lower frequency	**SOURce[:ANALog]:FREQuency:LOWer**
SMPTE IMD 4:1	Upper frequency	**SOURce[:ANALog]:FREQuency:UPPer**
SMPTE IMD 10:1	Amplitude	**SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate][:AMPLitude]**
	DC offset	**SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate]:OFFSet**
DFD IEC 60118	Difference frequency	**SOURce[:ANALog]:FREQuency:DIFFerence**
	Upper frequency	**SOURce[:ANALog]:FREQuency:UPPer**
	Amplitude	**SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate][:AMPLitude]**
	DC offset	**SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate]:OFFSet**
DFD IEC 60268	Difference frequency	**SOURce[:ANALog]:FREQuency:DIFFerence**
	Center frequency	**SOURce[:ANALog]:FREQuency:CENTer**
	Amplitude	**SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate][:AMPLitude]**
	DC offset	**SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate]:OFFSet**
Gaussian	Amplitude	**SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate][:AMPLitude]**
Rectangular	DC offset	**SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate]:OFFSet**
DC	Amplitude	**SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate][:AMPLitude]**
Multitone	DC offset	**SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate]:OFFSet**
	Amplitude	**SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate][:AMPLitude]**
	Start frequency	**SOURce[:ANALog]:MULTitone:FREQuency:STARt**
	Stop frequency	**SOURce[:ANALog]:MULTitone:FREQuency:STOP**
	Frequency spacing	**SOURce[:ANALog]:MULTitone:FREQuency:SPACing**
	Count	**SOURce[:ANALog]:MULTitone:COUNt**
	Waveform length	**SOURce[:ANALog]:MULTitone:WLEN**
	Crest factor	**SOURce[:ANALog]:MULTitone:CRESt?**
	Clear all tones	**SOURce[:ANALog]:MULTitone:TONE:CLEar**
	Add tone	**SOURce[:ANALog]:MULTitone:TONE:ADD**
	Delete tone	**SOURce[:ANALog]:MULTitone:TONE:DELete**
	Tone frequency	**SOURce[:ANALog]:MULTitone:TONE:FREQuency**
	Tone amplitude	**SOURce[:ANALog]:MULTitone:TONE:VOLTage**
	Tone phase	**SOURce[:ANALog]:MULTitone:TONE:PHASe**

Table A-5 Analog generator waveform parameters (continued)

Waveform	Parameter	SCPI command
Square	Frequency Amplitude	**SOURce[:ANALog]:DTMF:VOLTage:SUMMation** **SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate][:AMPLitude]**
Arbitrary	Amplitude DC offset	**SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate][:AMPLitude]** **SOURce[:ANALog]:VOLTage[:LEVel][:IMMediate]:OFFSet**

Appendix D: Analog Waveform Amplitude Range

Table A-6 Analog waveform amplitude range

Waveform	Amplitude range	
	Unbalanced/Common output	Balanced output
Sine Dual sine Variable phase SMPTE IMD 1:1 SMPTE IMD 4:1 SMPTE IMD 10:1 DFD IEC 60118 DFD IEC 60268	0 Vrms to 8 Vrms (0 to 11.3 Vp)	0 Vrms to 16 Vrms (0 to 22.6 Vp)
Gaussian	0 Vrms to 3.6 Vrms (0 to 11.3 Vp)	0 Vrms to 7.2 Vrms (0 to 22.6 Vp)
Rectangular	0 Vrms to 5.09 Vrms (0 to 11.3 Vp)	0 Vrms to 10.19 Vrms (0 to 22.6 Vp)
DC	–11.3 V to 11.3 V	–22.6 V to 22.6 V
Multitone	0 Vp to 11.3 Vp	0 Vp to 22.6 Vp
Square	0 Vrms to 11.3 Vrms (0 to 11.3 Vp)	0 Vrms to 22.6 Vrms (0 to 22.6 Vp)
Arbitrary	0 Vp to 11.3 Vp	0 Vp to 22.6 Vp

Appendix E: Relationship between Digital Waveform Parameters and Channels

Table A-7 Relationship between Digital Waveform Parameters and Channels

Waveform	Parameter	Channel
Sine	Frequency Amplitude DC offset	Channel 1 and Channel 2 are the same Channel 1 and Channel 2 can be different Channel 1 and Channel 2 are the same
Stereo	Frequency Amplitude DC offset	Channel 1 and Channel 2 can be different Channel 1 and Channel 2 can be different Channel 1 and Channel 2 are the same
Square	Frequency Amplitude DC offset	Channel 1 and Channel 2 are the same Channel 1 and Channel 2 can be different Channel 1 and Channel 2 are the same
Sine burst	Frequency Amplitude Burst on Period Low level	Channel 1 and Channel 2 are the same Channel 1 and Channel 2 can be different Channel 1 and Channel 2 are the same Channel 1 and Channel 2 are the same Channel 1 and Channel 2 are the same
Variable phase	Frequency Amplitude Phase	Channel 1 and Channel 2 are the same Channel 1 and Channel 2 can be different Channel 1 and Channel 2 are the same
Dual	Frequency 1 Frequency 2 Amplitude Ratio DC offset	Channel 1 and Channel 2 are the same Channel 1 and Channel 2 are the same Channel 1 and Channel 2 are the same Channel 1 and Channel 2 are the same Channel 1 and Channel 2 are the same
SMPTE IMD 1:1 SMPTE IMD 4:1 SMPTE IMD 10:1	Lower frequency Upper frequency Amplitude DC offset	Channel 1 and Channel 2 are the same Channel 1 and Channel 2 are the same Channel 1 and Channel 2 are the same Channel 1 and Channel 2 are the same
DFD IEC 60118	Difference frequency Upper frequency Amplitude DC offset	Channel 1 and Channel 2 are the same Channel 1 and Channel 2 are the same Channel 1 and Channel 2 are the same Channel 1 and Channel 2 are the same
DFD IEC 60268	Difference frequency Center frequency Amplitude DC offset	Channel 1 and Channel 2 are the same Channel 1 and Channel 2 are the same Channel 1 and Channel 2 are the same Channel 1 and Channel 2 are the same

Table A-7 Relationship between Digital Waveform Parameters and Channels

Waveform	Parameter	Channel
Gaussian/ Rectangular/ Triangular Pink	Amplitude DC offset	Channel 1 and Channel 2 can be different Channel 1 and Channel 2 are the same
Constant	Amplitude	Channel 1 and Channel 2 are the same
	Amplitude	Channel 1 and Channel 2 can be different
	Start frequency	Channel 1 and Channel 2 can be different
	Stop frequency	Channel 1 and Channel 2 can be different
	Frequency spacing	Channel 1 and Channel 2 can be different
	Count	Channel 1 and Channel 2 can be different
	Crest factor	Channel 1 and Channel 2 can be different
	Clear all tones	Channel 1 and Channel 2 can be different
	Add tone	Channel 1 and Channel 2 can be different
	Delete tone	Channel 1 and Channel 2 can be different
	Tone frequency	Channel 1 and Channel 2 can be different
	Tone amplitude	Channel 1 and Channel 2 can be different
	Tone phase	Channel 1 and Channel 2 can be different
	Randomize tone phase	Channel 1 and Channel 2 can be different
Arbitrary	Amplitude DC offset	Channel 1 and Channel 2 can be different Channel 1 and Channel 2 are the same

Appendix F: Sweep Start and Stop Range

Table A-8 Sweep start and stop range

<table>
<tr><th><sweep_parameter></th><th>Waveform</th><th>Sweep start range</th><th>Sweep stop range</th></tr>
<tr><td rowspan="3">FREQuency1</td><td>Sine
Dual</td><td>5 Hz to 80 kHz</td><td>5 Hz to 80 kHz</td></tr>
<tr><td>Square</td><td>5 Hz to 30 kHz</td><td>5 Hz to 30 kHz</td></tr>
<tr><td>SMPTE IMD 1:1/4:1/ 10:1</td><td>40 Hz to 500 Hz</td><td>40 Hz to 500 Hz</td></tr>
<tr><td rowspan="3">FREQuency2</td><td>Dual</td><td>5 Hz to 80 kHz</td><td>5 Hz to 80 kHz</td></tr>
<tr><td>SMPTE IMD 1:1/4:1/ 10:1</td><td>2 kHz to 60 kHz</td><td>2 kHz to 60 kHz</td></tr>
<tr><td>DFD IEC 60118</td><td>3 kHz to 80 kHz</td><td>3 kHz to 80 kHz</td></tr>
<tr><td rowspan="2">AMPLitude</td><td>Sine
Dual
Square
Gaussian
Rectangular
SMPTE IMD 1:1/4:1/ 10:1
DFD IEC 60118/60268</td><td>– 0 Vp to 22.6 Vp (balanced output)
– 0 Vp to 11.3 Vp (unbalanced or common output)</td><td>– 0 Vp to 22.6 Vp (balanced output)
– 0 Vp to 11.3 Vp (unbalanced or common output)</td></tr>
<tr><td>DC</td><td>– –22.6 V to 22.6 V (balanced output)
– –11.3 V to 11.3 V (unbalanced or common output)</td><td>– –22.6 V to 22.6 V (balanced output)
– –11.3 V to 11.3 V (unbalanced or common output)</td></tr>
<tr><td>PHASe</td><td>Variable phase</td><td>–180° to 179.99°</td><td>–180° to 179.99°</td></tr>
<tr><td>CENTer</td><td>DFD IEC 60268</td><td>3 kHz to 79 kHz</td><td>3 kHz to 79 kHz</td></tr>
</table>

Appendix G: Sweep-Capable Parameters

Table A-9 Sweep-capable parameters

Waveform	Amplitude	Frequency 1	Frequency 2	Center frequency	Phase
Sine	Yes	Yes	-	-	-
Dual sine	Yes (total voltage)	Yes	Yes	-	-
Variable phase	Yes	Yes	-	-	Yes
SMPTE IMD 1:1 SMPTE IMD 4:1 SMPTE IMD 10:1	Yes (total voltage)	Yes	Yes	-	-
DFD IEC 60118	Yes (total voltage)	-	Yes	-	-
DFD IEC 60268	Yes (total voltage)	-	-	Yes	-
Gaussian	Yes (total voltage)	-	-	-	-
Rectangular	Yes (total voltage)	-	-	-	-
Pink	Yes (total voltage)	-	-	-	-
DC	Yes	-	-	-	-
Square	Yes	Yes	-	-	-

Appendix H: Using the IEEE-488.2 Binary Block Format

In the binary block format, a block header precedes the user data.

The block header has the following format.

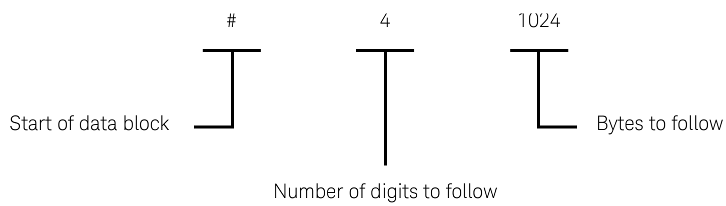


Figure A-1 IEEE-488.2 binary block format

The U8903B represents binary data as 32-bit floating points, which are sent as four bytes. Therefore, the total number of bytes is always four times the number of data points in the user data (and must always be an even number).

Appendix I: Sending HP8903B Commands to the U8903B

Before sending the HP8903B commands to the U8903B, enable the legacy mode. Refer to "**SYSTem:LEGacy:MODE**" on page 618 for more information on the legacy mode.

The default channel for the legacy mode is channel 1 as the HP8903B supports only one channel. However, the U8903B provides the flexibility to switch to other channels. Refer to "**SYSTem:LEGacy:CHANnel**" on page 617 for more information on the legacy mode channel.

NOTE

- You can still send SCPI commands when the legacy mode is enabled.
- When the legacy mode is disabled or enabled, the U8903B will execute 'device clear'. In order to comply with the legacy mode, the U8903B will set the following when the legacy mode is enabled.
  - 80 kHz low-pass filter (U8903B allows you to customize the default low-pass filter when the legacy mode is enabled)
  - T0 (auto query enabled)
- You can also send concatenated HP8903B commands. For example, sending the `AP1.2VLFR2KZ47.1SP` command is equivalent to sending the `AP1.2VL, FR2KZ`, and `47.1SP` commands separately.
- If there is an invalid HP8903B command in a concatenated HP8903B command, the whole command will be ignored. For example, the `AP1.2VLFFXXFR2KZ47.1SP` command will not be executed as `FFXX` is not a valid HP8903B command.

Supported HP8903B commands list

The supported versions of the HP8903B commands are versions R2.0.0.0 and above. The list of supported HP8903B commands are grouped into five different groups: generator, measurement, analyzer, sweep, and system.

Generator commands

Table A-10 Supported HP8903B generator commands

Command syntax	Description
AP<voltage><unit>	Sets the amplitude level of the source signal. The supported units are as follows. – VL – Vrms – MV – mVrms – DV – dBV (= voltage – 2.218 dB) Examples: AP1.5VL → sets the amplitude to 1.5 Vrms AP0DV → sets the voltage to 2.218 dB
FR<frequency><unit>	Sets the frequency of the source signal. The supported units are as follows. – HZ – KZ where KZ = kHz Example: FR3.2KZ → sets the frequency to 3.2 kHz
AN<amplitude><unit>	Sets the step value to increase or decrease the amplitude. The supported units are as follows. – VL – Vrms – MV – mVrms – DV – dBV (= voltage – 2.218 dB)
FN<frequency><unit>	Sets the step value to increase or decrease the frequency. The supported units are as follows. – HZ – KZ where KZ = kHz
UP DN	UP increases the amplitude or frequency to the step value set by the AN or FN commands. DN increases the amplitude or frequency to the step value set by the AN or FN commands. The parameter to increase or decrease depends on the last AN or FN command. Examples: Current amplitude = 0 Vrms, Frequency = 1 kHz AN0.5VL UP → increases the amplitude to 0.5 Vrms UP → increases the amplitude to 1 Vrms DN → decreases the amplitude to 0.5 Vrms FN1KZ UP → increases the frequency to 2 kHz

Table A-10 Supported HP8903B generator commands (continued)

Command syntax	Description
`47.0SP` `47.1SP`	Sets the impedance to 50 Ω and 600 Ω respectively.
`10.0SP`	Active generator menu of the HP8903B mode.

Measurement commands

Table A-11 Supported HP8903B measurement commands

Command syntax	Description
`T0` `T1`	`T0` turns on the auto query mode and `T1` turns off the auto query mode. `T0` and `T1` cannot be used in a concatenated command.
`T2` `T3`	Initiates the analyzer and returns the measurement readings. `T3` returns the measurement readings with a delay of 300 ms. The `T2` and `T3` commands will turn off the auto query. These commands return the measurement readings of only one function which is determined by the `RR` and `RL` commands.
`RR[?]` or `20.0SP` `RL[?]` or `20.1SP`	Determines the function type for the measurement readings return by the `T2` and `T3` commands. `RR[?]` or `20.0SP` will set the `T2` or `T3` command to return the measurement readings of function 2. `RL[?]` or `20.1SP` will set the `T2` or `T3` command to return the measurement readings of function 1.
`16.0SP` `16.1SP`	Determines the resolution of SINAD and SNR measurement in dB. `16.0SP` (default) – Resolution is set to 0.01 dB for values more than 25 dB – Resolution is set to 0.5 dB for values less than 25 dB `16.1SP` – Resolution is set to 0.01 dB for all range of values

Analyzer commands

Table A-12 Supported HP8903B analyzer commands

Command syntax	Description
M1 M2 M3 S1 S2 S3	Sets the measurement function. M1 → VAC M2 → SINAD M3 → THD ratio S1 → DC S1 → SNR S1 → THD level
1.0SP 1.3SP 1.4SP 1.5SP 1.6SP 1.7SP 1.8SP 1.9SP 1.10SP 1.11SP 1.12SP 1.13SP 1.14SP 1.15SP 1.16SP 1.17SP 1.18SP 1.19SP 2.0SP 2.2SP 2.3SP 2.4SP	Sets the input range of the analyzer. 1.0SP, 2.0SP → enable auto range 9.0SP → disable auto range 1.3SP → 140 V 1.4SP, 1.5SP, 2.2SP → 100 V 1.6SP, 1.7SP, 1.8SP, 2.3SP → 32 V 1.9SP, 1.10SP, 2.4SP → 10 V 1.11SP, 1.12SP, 1.13SP → 3.2 V 1.14SP, 1.15SP → 1V 1.6SP, 1.17SP, 1.18SP, 1.19SP → 0.32 V
A0 5.0SP 5.1SP	Sets the input detector. A0, 5.0SP, 5.1SP → RMS

Table A-12 Supported HP8903B analyzer commands (continued)

Command syntax	Description
H0 H1 H2 L0 L1 L2	Sets the filters of the U8903B. H0 → turns off the left or right plug-in filter H1, H2 → sets the left and right plug-in filters to high-pass filter or weighting filter respectively L0 → turns off the low-pass filter L1 → sets the low-pass filter to 30 kHz L2 → sets the low-pass filter to 80 kHz Configure the left and right filters before sending the H1 and H2 commands. Only one of the filters can be turned on at a time. Refer to “**SENSe[:ANALog]:FILTer:LEFT**” on page 343and “**SENSe[:ANALog]:FILTer:RIGHt**” on page 352 for more information on the filters. Example: Configure the left and right filters. SENS:FILT:LEFT HP400 → sets the left filter to high-pass filter with 400 Hz cutoff frequency SENS:FILT:RIGH CMES → sets the right filter to C-Message filter H1 → enables the HP400 filter H2 → enables the HP400 filter and disables the HP400 filter
LN	Changes the measurement result unit to linear unit. Refer to **Table A-15** for more information.
LG	Changes the measurement result unit to logarithmic unit. Refer to **Table A-15** for more information.
R0	Disables the ratio mode. Refer to **Table A-15** for more information.
<value>R1	Compares the measurement result to a reference value. If the <value> is not specified, the previous measurement will be used. Refer to **Table A-15** for more information.
11.0SP	Enables the ratio mode with the previous reference value. Equivalent to R1 but without taking the measurement before switching the measurement unit. Refer to **Table A-15** for more information.
11.1SP	Reads the reference type. This command will also activate the analyzer menu of the HP8903B mode.
19.0SP 19.NNNSP	Sets the reference impedance and changes the value to W (supported functions only) 19.0SP → sets the reference impedance to 8 Ω 19.NNNSP → sets the reference impedance to a specific value, NNN:1 to ~999 Ω Example: 19.8SP → equivalent to the 19.0SP command

Sweep commands

Table A-13 Supported HP8903B sweep commands

Command syntax	Description
`FA<start_frequency><unit>` `FA<stop_frequency><unit>`	Sets the start stop frequency. The supported units are as follows. – HZ – KZ where KZ = kHz Example: `FA100Hz` → sets the start frequency to 100 Hz

`17.0SP`
`17.1SP`
`17.2SP`
`17.3SP`
`17.4SP`
`17.5SP`
`17.6SP`
`17.7SP`
`17.8SP`
`17.9SP`
	Sets the sweep resolution. Each command determines the number of points over decade.
	`17.1SP` → 1/dec
	`17.2SP` → 2/dec
	`17.3SP` → 5/dec
	`17.0SP`, `17.4SP` → 10/dec
	`17.5SP` → 20/dec
	`17.6SP` → 50/dec
	`17.7SP` → 100/dec
	`17.8SP` → 200/dec
	`17.9SP` → 500/dec
	
	The total number of points is computed using the following formula.
	Point count = Sweep range × (Points/Decade)
	where Sweep range (in decades) = $\log_{10}$ (Stop frequency/Start frequency)
	
	Example:
	Start frequency = 30 Hz, Stop frequency = 30 kHz
	`17.3SP` → 15 points

`W1`	Initiates the frequency sweep.
`W0`	Aborts the frequency sweep.

System commands

Table A-14 Supported HP8903B system commands

Command syntax	Description
`21.1SP`	Reads the GPIB address.
`22.1SP`	Equivalent to the `*SRE 16` command. Refer to "***SRE**" on page 42 for more information.
`22.2SP`	Equivalent to the `*ESE 32; SRE 32` command. Refer to "***ESE**" on page 35 and "***SRE**" on page 42 for more information.
`22.4SP`	Equivalent to the `*SRE 4` command. Refer to "***SRE**" on page 42 for more information.

Unit charts

Table A-15 HP8903B unit charts

Measurement	Ratio On		Ratio Off	
	LIN	LOG	LIN	LOG
AC level	%	dB	V	dBm into 600 Ω
DC level	%	dB	V	dBm into 600 Ω
SINAD	%	dB	%	dB
Sig/Noise	%	dB	%	dB
DSTN	%	dB	%	dB
DSTN level	%	dB	V	dBm into 600 Ω

Example of usage

Amplitude = 5 Vrms
Frequency = 500 Hz
Measurement function = SINAD

In order to set to the settings above, the following command sequence is sent.

```
SYST:LEG:MODE ON
AP5VLFR0.5KZM2
```

In order to increment the amplitude by three times with step value of 100 mVrms, the following command is sent.

```
AN100MVUPUPUP
```

Appendix J: AES3/SPDIF Interface Channel Status Bits Field Names

Table A-16 AES3/SPDIF interface channel status bits field names

Field Name	Field Values
Mode	Non Linear PCM or Linear PCM
Audio Mode	Non Linear PCM or Linear PCM
Consumer	
Copyright	Non Copyright or Copyright
Emphasis	No pre-emphasis, 50/15ìs,Reserved 1, or Reserved 2
Channel Mode	0 to 3
Category Code	General, Laser Optical, D/D Converter, Magnetic, Digital Broadcast 1, Digital Broadcast 2, Musical Instrument, ADC Non Copyright, Solid State Memory, ADC Copyright, Experimental, Reserved 1, or Reserved 2
Source Number	0 to 15
Channel Number	0 to 15
Sample Freq	Not indicated, 22.05 kHz, 24 kHz, 32 kHz, 44.1kHz, 48 kHz, 88.2 kHz, 96 kHz, 176.4 kHz, 192 kHz, or 768 kHz
Clock Accuracy	Level 1, Level 2, Level 3, or Reserved
Max Word Length	20 bits or 24 bits
Word Length	For Max Word Length = 20 bits, Not indicated, 16 bits, 17 bits, 18 bits, 19 bits, or 20 bits For Max Word Length = 24 bits, Not indicated, 20 bits, 21 bits, 22 bits, 23 bits, or 24 bits
Original Sample Freq	Not indicated, 8 kHz, 11.025 kHz, 12 kHz, 16 kHz, 22.05 kHz, 24 kHz, 32 kHz, 44.1 kHz, 48 kHz, 88.2 kHz, 96 kHz, 176.4 kHz, 192 kHz, Reserved 1, or Reserved 2
CGMS-A	Copying Permitted, Condition Not Used, One Generation Copy, or Copying Denied
Professional	
Emphasis	Not indicated, No pre-emphasis, 50/15us, or CCITT J.17
Sample Freq	Not indicated, 22.05 kHz, 24 kHz, 32 kHz, 44.1 kHz, 48 kHz, 88.2 kHz, 96 kHz, 176.4 kHz, or 192 kHz
Sample Freq Scaling	Disable or Enable
Channel Mode	Not indicated, 2-channel, Single Channel, Primary-Secondary, Stereo, Reserved 1, Reserved 2, Mono Double Rate, Left Double Rate, Right Double Rate, or Multichannel
User Bits	Not indicated, 192-bit block, Reserved for AES18, User defined, Reserved for Metadata, or As in IEC60958-3
Auxiliary Bits	20-bit not defined, 24-bit main audio, 20-bit single, or Reserved

Table A-16 AES3/SPDIF interface channel status bits field names

Field Name	Field Values
Word Length	If Auxiliary Bits = 24-bit main audio, Not indicated, 20 bits, 21 bits, 22 bits, 23 bits, 24 bits If Auxiliary Bits = 20-bit not defined, 20-bit single, or Reserved, Not indicated, 16 bits, 17 bits, 18 bits, 19 bits, or 20 bits
Alignment Level	Not indicated, -18.06 dBFS, -20 dBFS, or Reserved
Multichannel Status	Undefined or Defined
Multichannel Mode	Mode 0, Mode 1, Mode 2, Mode 3, or User Defined Note: The Multichannel Mode is only applicable when the Multichannel Status is set to Defined.
Channel Number	If Multichannel Status = Defined, 1 to 16 If Multichannel Status = Undefined, 1 to 128
Reference Signal	Not a ref. signal, Grade 1, Grade 2, or Reserved
Channel Origin	0 to 4 (alphanumeric digit)
Channel Destination	0 to 4 (alphanumeric digit)
Local Address	0 to 2^{32}-1
Time-of-day	0 to 2^{32}-1
0-5 Reliable	False or True
6-13 Reliable	False or True
14-17 Reliable	False or True
18-21 Reliable	False or True
CRCC	Value between 0 to 255 (query only)

Appendix K: Word Length, Sampling Rate, and Multiplier for DSI Interface

Table A-17 Word length, sampling rate, and multiplier for DSI interface

≤ Sampling rate (kHz)[a]	Word length	Multiplier
400	8	128
400	9	72, 144
400	10	80, 160
400	11	88, 176
400	12	96, 192
400	13	104, 208
400	14	112, 224
400	15	120, 240
400	16	64, 128
400	17	68, 136
400	18	72, 144
400	19	76, 152
400	20	80, 160
400	21	84, 168
400	22	88, 176
400	23	92, 184
400	24	96, 192
400	25	100, 200
400	26	104, 208
400	27	108, 216
400	28	112, 224
400	29	116, 232
400	30	120, 240
400	31	124, 248
400	32	128
200	8	128, 256
200	9	72, 144, 288
200	10	80, 160, 320
200	11	88, 176, 352
200	12	96, 192, 384

Table A-17 Word length, sampling rate, and multiplier for DSI interface

≤ Sampling rate (kHz)[a]	Word length	Multiplier
200	13	104, 208, 416
200	14	112, 224, 448
200	15	120, 240, 480
200	16	64, 128, 256
200	17	68, 136, 272
200	18	72, 144, 288
200	19	76, 152, 304
200	20	80, 160, 320
200	21	84, 168, 336
200	22	88, 176, 352
200	23	92, 184, 368
200	24	96, 192, 384
200	25	100, 200, 400
200	26	104, 208, 416
200	27	108, 216, 432
200	28	112, 224, 448
200	29	116, 232, 464
200	30	120, 240, 480
200	31	124, 248, 496
200	32	128, 256
100	8	128, 256, 512
100	9	72, 144, 288, 576
100	10	80, 160, 320, 640
100	11	88, 176, 352, 704
100	12	96, 192, 384, 768
100	13	104, 208, 416, 832
100	14	112, 224, 448, 896
100	15	120, 240, 480, 960
100	16	64, 128, 256, 512
100	17	68, 136, 272, 544
100	18	72, 144, 288, 576
100	19	76, 152, 304, 608
100	20	80, 160, 320, 640

Table A-17 Word length, sampling rate, and multiplier for DSI interface

≤ Sampling rate (kHz)[a]	Word length	Multiplier
100	21	84, 168, 336, 672
100	22	88, 176, 352, 704
100	23	92, 184, 368, 736
100	24	96, 192, 384, 768
100	25	100, 200, 400, 800
100	26	104, 208, 416, 832
100	27	108, 216, 432, 864
100	28	112, 224, 448, 896
100	29	116, 232, 464, 928
100	30	120, 240, 480, 960
100	31	124, 248, 496, 992
100	32	128, 256, 512
50	8	128, 256, 512, 1024
50	9	72, 144, 288, 576
50	10	80, 160, 320, 640
50	11	88, 176, 352, 704
50	12	96, 192, 384, 768
50	13	104, 208, 416, 832
50	14	112, 224, 448, 896
50	15	120, 240, 480, 960
50	16	64, 128, 256, 512
50	17	68, 136, 272, 544
50	18	72, 144, 288, 576
50	19	76, 152, 304, 608
50	20	80, 160, 320, 640
50	21	84, 168, 336, 672
50	22	88, 176, 352, 704
50	23	92, 184, 368, 736
50	24	96, 192, 384, 768
50	25	100, 200, 400, 800
50	26	104, 208, 416, 832
50	27	108, 216, 432, 864
50	28	112, 224, 448, 896

Table A-17 Word length, sampling rate, and multiplier for DSI interface

≤ Sampling rate (kHz)[a]	Word length	Multiplier
50	29	116, 232, 464, 928
50	30	120, 240, 480, 960
50	31	124, 248, 496, 992
50	32	128, 256, 512, 1024
25	8	128, 256, 512, 1024
25	9	72, 144, 288, 576
25	10	80, 160, 320, 640
25	11	88, 176, 352, 704
25	12	96, 192, 384, 768
25	13	104, 208, 416, 832
25	14	112, 224, 448, 896
25	15	120, 240, 480, 960
25	16	64, 128, 256, 512
25	17	68, 136, 272, 544
25	18	72, 144, 288, 576
25	19	76, 152, 304, 608
25	20	80, 160, 320, 640
25	21	84, 168, 336, 672
25	22	88, 176, 352, 704
25	23	92, 184, 368, 736
25	24	96, 192, 384, 768
25	25	100, 200, 400, 800
25	26	104, 208, 416, 832
25	27	108, 216, 432, 864
25	28	112, 224, 448, 896
25	29	116, 232, 464, 928
25	30	120, 240, 480, 960
25	31	124, 248, 496, 992
25	32	128, 256, 512, 1024
12.5	8	128, 256, 512, 1024
12.5	9	72, 144, 288, 576
12.5	10	80, 160, 320, 640
12.5	11	88, 176, 352, 704

Table A-17 Word length, sampling rate, and multiplier for DSI interface

≤ Sampling rate (kHz)[a]	Word length	Multiplier
12.5	12	96, 192, 384, 768
12.5	13	104, 208, 416, 832
12.5	14	112, 224, 448, 896
12.5	15	120, 240, 480, 960
12.5	16	64, 128, 256, 512
12.5	17	68, 136, 272, 544
12.5	18	72, 144, 288, 576
12.5	19	76, 152, 304, 608
12.5	20	80, 160, 320, 640
12.5	21	84, 168, 336, 672
12.5	22	88, 176, 352, 704
12.5	23	92, 184, 368, 736
12.5	24	96, 192, 384, 768
12.5	25	100, 200, 400, 800
12.5	26	104, 208, 416, 832
12.5	27	108, 216, 432, 864
12.5	28	112, 224, 448, 896
12.5	29	116, 232, 464, 928
12.5	30	120, 240, 480, 960
12.5	31	124, 248, 496, 992
12.5	32	128, 256, 512, 1024
6.75	8	128, 256, 512, 1024
6.75	9	72, 144, 288, 576
6.75	10	80, 160, 320, 640
6.75	11	88, 176, 352, 704
6.75	12	96, 192, 384, 768
6.75	13	104, 208, 416, 832
6.75	14	112, 224, 448, 896
6.75	15	120, 240, 480, 960
6.75	16	128, 256, 512
6.75	17	136, 272, 544
6.75	18	144, 288, 576
6.75	19	152, 304, 608

Table A-17 Word length, sampling rate, and multiplier for DSI interface

≤ Sampling rate (kHz)[a]	Word length	Multiplier
6.75	20	160, 320, 640
6.75	21	168, 336, 672
6.75	22	176, 352, 704
6.75	23	184, 368, 736
6.75	24	192, 384, 768
6.75	25	200, 400, 800
6.75	26	208, 416, 832
6.75	27	216, 432, 864
6.75	28	224, 448, 896
6.75	29	116, 232, 464, 928
6.75	30	120, 240, 480, 960
6.75	31	124, 248, 496, 992
6.75	32	128, 256, 512, 1024

[a] For sampling rate less than or equal to.

Appendix L: Word Length, Sampling Rate, and Multiplier for Master Clock In

Table A-18 Word length, sampling rate, and multiplier for Master Clock In

≤ Sampling rate (kHz)[a]	Word length	Multiplier
400	8	64, 128
400	9	72, 144
400	10	80, 160
400	11	88
400	12	96
400	13	104
400	14	112
400	15	120
400	16	64, 128
400	17	68, 136
400	18	72, 144
400	19	76
400	20	80
400	21	84
400	22	88
400	23	92
400	24	96, 192
400	25	100
400	26	104
400	27	108
400	28	112
400	29	116,
400	30	120
400	31	124
400	32	128
200	8	64, 128, 256
200	9	72, 144, 288
200	10	80, 160
200	11	88, 176
200	12	96, 192

Table A-18 Word length, sampling rate, and multiplier for Master Clock In

≤ Sampling rate (kHz)[a]	Word length	Multiplier
200	13	104, 208
200	14	112, 224
200	15	120, 240
200	16	64, 128, 256
200	17	68, 136, 272
200	18	72, 144, 288
200	19	76, 152
200	20	80, 160
200	21	84, 168
200	22	88, 176
200	23	92, 184
200	24	96, 192
200	25	100, 200
200	26	104, 208
200	27	108, 216
200	28	112, 224
200	29	116, 232
200	30	120, 240
200	31	124, 248
200	32	128, 256
100	8	64. 128, 256, 512
100	9	72, 144, 288, 576
100	10	80, 160, 320
100	11	88, 176, 352
100	12	96, 192, 384
100	13	104, 208, 416
100	14	112, 224, 448
100	15	120, 240, 480
100	16	64, 128, 256, 512
100	17	68, 136, 272, 544
100	18	72, 144, 288, 576
100	19	76, 152, 304
100	20	80, 160, 320

Table A-18 Word length, sampling rate, and multiplier for Master Clock In

≤ Sampling rate (kHz)[a]	Word length	Multiplier
100	21	84, 168, 336
100	22	88, 176, 352
100	23	92, 184, 368
100	24	96, 192, 384
100	25	100, 200, 400
100	26	104, 208, 416
100	27	108, 216, 432
100	28	112, 224, 448
100	29	116, 232, 464
100	30	120, 240, 480
100	31	124, 248, 496
100	32	128, 256, 512
50	8	64, 128, 256, 512
50	9	72, 144, 288, 576
50	10	80, 160, 320, 640
50	11	88, 176, 352, 704
50	12	96, 192, 384, 768
50	13	104, 208, 416, 832
50	14	112, 224, 448, 896
50	15	120, 240, 480, 960
50	16	64, 128, 256, 512
50	17	68, 136, 272, 544
50	18	72, 144, 288, 576
50	19	76, 152, 304, 608
50	20	80, 160, 320, 640
50	21	84, 168, 336, 672
50	22	88, 176, 352, 704
50	23	92, 184, 368, 736
50	24	96, 192, 384, 768
50	25	100, 200, 400, 800
50	26	104, 208, 416, 832
50	27	108, 216, 432, 864
50	28	112, 224, 448, 896

Table A-18 Word length, sampling rate, and multiplier for Master Clock In

≤ Sampling rate (kHz)[a]	Word length	Multiplier
50	29	116, 232, 464, 928
50	30	120, 240, 480, 960
50	31	124, 248, 496, 992
50	32	128, 256, 512
25	8	64, 128, 256, 512
25	9	72, 144, 288, 576
25	10	80, 160, 320, 640
25	11	88, 176, 352, 704
25	12	96, 192, 384, 768
25	13	104, 208, 416, 832
25	14	112, 224, 448, 896
25	15	120, 240, 480, 960
25	16	64, 128, 256, 512
25	17	136, 272, 544
25	18	72, 144, 288, 576
25	19	152, 304, 608
25	20	80, 160, 320, 640
25	21	168, 336, 672
25	22	88, 176, 352, 704
25	23	184, 368, 736
25	24	96, 192, 384, 768
25	25	200, 400, 800
25	26	104, 208, 416, 832
25	27	216, 432, 864
25	28	112, 224, 448, 896
25	29	232, 464, 928
25	30	120, 240, 480, 960
25	31	248, 496, 992
25	32	128, 256, 512
12.5	8	64, 128, 256, 512
12.5	9	144, 288, 576
12.5	10	80, 160, 320, 640
12.5	11	176, 352, 704

Table A-18 Word length, sampling rate, and multiplier for Master Clock In

≤ Sampling rate (kHz)[a]	Word length	Multiplier
12.5	12	96, 192, 384, 768
12.5	13	208, 416, 832
12.5	14	112, 224, 448, 896
12.5	15	240, 480, 960
12.5	16	64, 128, 256, 512
12.5	17	272, 544
12.5	18	144, 288, 576
12.5	19	304, 608
12.5	20	160, 320, 640
12.5	21	336, 672
12.5	22	176, 352, 704
12.5	23	368, 736
12.5	24	192, 384, 768
12.5	25	400, 800
12.5	26	208, 416, 832
12.5	27	432, 864
12.5	28	224, 448, 896
12.5	29	464, 928
12.5	30	240, 480, 960
12.5	31	496, 992
12.5	32	128, 256, 512
6.75	8	64, 128, 256, 512
6.75	9	288, 576
6.75	10	160, 320, 640
6.75	11	352, 704
6.75	12	192, 384, 768
6.75	13	416, 832
6.75	14	224, 448, 896
6.75	15	480, 960
6.75	16	64, 128, 256, 512
6.75	17	544
6.75	18	288, 576
6.75	19	608

Table A-18 Word length, sampling rate, and multiplier for Master Clock In

≤ Sampling rate (kHz)[a]	Word length	Multiplier
6.75	20	320, 640
6.75	21	672
6.75	22	352, 704
6.75	23	736
6.75	24	384, 768
6.75	25	800
6.75	26	416, 832
6.75	27	864
6.75	28	448, 896
6.75	29	928
6.75	30	480, 960
6.75	31	992
6.75	32	128, 256, 512

[a] For sampling rate less than or equal to.

Appendix M: Legacy Sweep

Legacy sweep is a sweep mode that supports U8903A sweep SCPI commands. If you wish to use the sweep implementation of the U8903A with firmware v 2.10.1.0 and lower, enable the legacy sweep mode via the U8903B GUI. **Table A-19** and **Table A-20** describe the different behaviors and the affected SCPI commands when legacy sweep mode is enabled or disabled.

Table A-19 Comparison between legacy sweep enabled and disabled behaviors

Legacy sweep enabled (firmware v 2.10.1.0 and below)	Legacy sweep disabled (firmware v 2.10.1.0 and above)
Allow only analog sweep.	Allow cross interface sweep. (For example, digital sweep to analog sweep; determined by `SOURce:SWEep:INTerface` and `SENSe:SWEep:INTerface` commands)
Allow only on the same analyzer channel number and generator channel number to perform single channel sweep. (For example, analyzer channel 1 and generator channel 1; determined by `SOURce:SWEep:CHANnel` command)	Allow cross-channel sweep and multiple channels sweep. (For example, analyzer channel 1 and generator channel 2; determined by `SOURce:SWEep:CHANnel` and `SENSe:SWEep:CHANnel` commands)
Allow only one sweep function (determined by the analyzer's function 2) to be measured.	Allow multiple sweep functions (determined by the reference analyzer's measurement function settings) to be measured.
Each channel has its own individual sets of sweep configuration. The sweep configuration consists of start, stop, step, point count, parameter, spacing, and dwell time.	There is only one set of sweep configuration. The sweep configuration consists of start, stop, step, point count, parameter, spacing, and dwell time.

Table A-20 SCPI command comparison

Description	Legacy sweep enabled (firmware v 2.10.1.0 and below)	Legacy sweep disabled (firmware v 2.10.1.0 and above)
Obtaining sweep result	FETCh:SWEep? – This command only returns the sweep result of function 2	FETCh:SWEep? <function>,(@<channel>) – This command returns the sweep result of the <function> for the given channels
Managing sweep channel	SOURce:SWEep:CHANnel <channel> – This command determines which analyzer and generator channel to sweep	SOURce:SWEep:CHANnel <channel list> – This command determines which generator channels to sweep SOURce:SWEep:REFerence:CHANnel – This command sets the reference generator channel to sweep SOURce:SWEep:INTerface – This command sets the interface of the generator to sweep SENSe:SWEep:CHANnel – This command determines which analyzer channel to sweep SENSe:SWEep:REFerence:CHANnel – This command sets the reference analyzer channel to sweep SENSe:SWEep:INTerface – This command sets the interface of the analyzer to sweep
Setting sweep paramenters	– SOURce:SWEep:MODE <mode>,<channel_list> – SOURce:SWEep:PAR <paramenter>,<channel_list> – SOURce:SWEep:STARt <start_value>,<channel_list> – SOURce:SWEep:STOP <stop_value>,<channel_list> – SOURce:SWEep:SPACing <spacing_type>,<channel_list> – SOURce:SWEep:STEP <step_size>,<channel_list> – SOURce:SWEep:POINts <point_count>,<channel_list> – SOURce:SWEep:DWELl <dwell_time>,<channel_list>	– SOURce:SWEep:MODE <mode> – SOURce:SWEep:PAR <paramenter> – SOURce:SWEep:STARt <start_value> – SOURce:SWEep:STOP <stop_value> – SOURce:SWEep:SPACing <spacing_type> – SOURce:SWEep:STEP <step_size> – SOURce:SWEep:POINts <point_count> – SOURce:SWEep:DWELl <dwell_time>
Store sweep points and result	MMEMory:STORe SWEep, <filename>	MMEMory:STORe:SWEep <channel>,<function>,<filename> Note that the <function> field is not used but NOT optional
Performing group delay analysis		SENSe:APPLication:TYPE GDELay – Digital sweep to analyze group delay

Appendix N: Migrating from U8903A to U8903B

Table A-21 Migrating from U8903A to U8903B

Voltage range (firmware 2.10.x.x and below)	New voltage range (firmware 3.0.0.0 and above)	
BNC and XLR	BNC	XLR
400 mV	1 V	1 V
800 mV	1 V	1 V
1.6 V	3.2 V	3.2 V
3.2 V	3.2 V	3.2 V
6.4 V	10 V	10 V
12.8 V	32 V	32 V
25 V	32 V	32 V
50 V	100 V	100 V
100 V	100 V	100 V
140 V	140 V	300 V

Appendix O: Parameters to Reset

Table A-22 Parameters to reset

GUI	SCPI command	Parameters to reset
To factory settings	**SYSTem:RESet[:MODE]**	
Reset all	***RST**	All parameters for all generator and analyzer channels
Reset analog generator	**SYSTem:RESet[:MODE]** AGENerator	
Reset analog analyzer	**SYSTem:RESet[:MODE]** AANalyzer	Functions config – Function number, measurement function (1 to 4), unit (1 to 4), and reading format Function specific settings – Voltage detector type, analog notch state, frequency lock mode, notch bandwidth, analog notch state, odd harmonics, even harmonics, SNR delay, and reference channel References – Reference level, reference frequency, reference ratio, reference impedance, reference channel, and calibrator level Filter configs – LPF, HPF, weighting filter, and de-emphasis filter Meas configs – Auto range state, input voltage range, sample size, source channel, average, and trigger source Input configs – Connector type, input impedance, coupling type, bandwidth, and external gain Statistics – Number of readings, statistic function (1 to 3) Wave file – Number of channel, bits/sample, and duration

Appendix P: Deprecated SCPI Commands

CALCulate[:ANALog]:FORMat:FREQuency

Syntax

```
CALCulate[:ANALog]:FORMat:FREQuency <format>,(@<channel>)
CALCulate[:ANALog]:FORMat:FREQuency? (@<channel>)
```

Description

Sets the format of the measurement data for the specified analog analyzer channel(s) for frequency measurement function. The query returns the format of the measurement data of the selected channel(s).

The description for each `<format>` parameter is shown as follows.

OFF	Returns the measurement data in Hz.
DELTa	Returns the measurement data in ΔHz.

Parameters

Item	Type	Range of values	Default value
<format>	Discrete	OFF|DELTa	OFF
<channel>	Numeric	1|2	1

Remark

After this command is sent, the calculated measurement data can be acquired using the command for analyzer mode or the **FETCh:SWEep?** command for sweep mode.

Examples

The following command sets the frequency measurement format to delta for channel 1.

```
CALC:FORM:FREQ DELT,(@1)
```

The following query returns the format of the measurement data for channel 1.

```
CALC:FORM:FREQ? (@1)
```

Typical response:

```
DELT
```

CALCulate[:ANALog]:FORMat:LEVel

Syntax

```
CALCulate[:ANALog]:FORMat:LEVel <format>,(@<channel>)
```

```
CALCulate[:ANALog]:FORMat:LEVel? (@<channel>)
```

Description

Sets the format of the measurement data for the specified analog analyzer channel(s) for level measurement function such as VAC and VDC. The query returns the format of the measurement data of the selected channel(s).

The description for each `<format>` parameter is shown as follows.

OFF	Returns the measurement data in V.
LOGarithmic	Returns the measurement data in dBr.
LINear	Returns the measurement data in x.

Parameters

Item	Type	Range of values	Default value
<format>	Discrete	OFF|LOGarithmic|LINear	OFF
<channel>	Numeric	1|2	1

Remark

After this command is sent, the calculated measurement data can be acquired using the command for analyzer mode or the **FETCh:SWEep?** command for sweep mode.

Examples

The following command sets the level measurement format to linear for channel 1.

```
CALC:FORM:LEV LIN,(@1)
```

The following query returns the format of the measurement data for channel 1.

```
CALC:FORM:LEV? (@1)
```

Typical response:

```
LIN
```

CALCulate[:ANALog]:FORMat:RATio

Syntax

```
CALCulate[:ANALog]:FORMat:RATio <format>,(@<channel>)
```

```
CALCulate[:ANALog]:FORMat:RATio? (@<channel>)
```

Description

Sets the format of the measurement data for the specified analog analyzer channel(s) for ratio measurement function such as SINAD, THD Ratio, DFD, IMD, crosstalk, and SNR. The query returns the format of the measurement data of the selected channel(s).

The description for each `<format>` parameter is shown as follows.

OFF	Returns the measurement data in dB.
DELTa	Returns the measurement data in ΔdB.
LINear	Returns the measurement data in x.

Parameters

Item	Type	Range of values	Default value
<format>	Discrete	OFF|DELTa|LINear	OFF
<channel>	Numeric	1|2	1

Remark

After this command is sent, the calculated measurement data can be acquired using the command.

Examples

The following command sets the ratio measurement format to delta for channel 1.

```
CALC:FORM:RAT DELT,(@1)
```

The following query returns the format of the measurement data for channel 1.

```
CALC:FORM:RAT? (@1)
```

Typical response:

```
DELT
```

DISPlay:ANALysis:MODE

Syntax

`DISPlay:ANALysis:MODE <mode>`

`DISPlay:ANALysis:MODE?`

Description

Sets the graph display as either time domain, frequency domain (magnitude), or frequency domain (phase). The query returns the graph display mode in the form of `MAGN`, `PHAS`, or `TIME`.

Parameter

Item	Type	Range of values	Default value
`<mode>`	Discrete	MAGNitude|PHASe|TIME	Required parameter

Examples

The following command sets the graph display as frequency domain (magnitude).

```
DISP:ANAL:MODE MAGN
```

The following query returns the graph display mode.

```
DISP:ANAL:MODE?
```

Typical response:

```
MAGN
```

SENSe[:ANALog]:NOTCh:EMULation[:STATe]

Syntax

```
SENSe[:ANALog]:NOTCh:EMULation[:STATe] <state>
```

```
SENSe[:ANALog]:NOTCh:EMULation[:STATe]?
```

Description

Enables or disables the analog analyzer notch emulation. The query returns the analog analyzer notch emulation state as 0 if the state is OFF, or 1 if the state is ON. This command is for backward compatibility with U8903A (firmware 2.10.x.x and below).

Parameter

Item	Type	Range of values	Default value
`<state>`	Boolean	OFF(0)|ON(1)	– OFF (HP8903B mode is off) – ON (HP8903B mode is on)

Remarks

- This command can only be used when the GPIB is successfully initialized.
- The `SENSe[:ANALog]:NOTCh:EMULation[:STATe]` command will not bring any effect. The mode is now controlled by the HP8903B mode.

Example

The following query returns the analog analyzer notch emulation state when HP8903B mode is on.

```
SYS:LEG:MODE 1
SENS:NOTC:EMUL?
```

Typical response:

```
1
```

SENSe[:ANALog]:REFerence:MEASured

Syntax

```
SENSe[:ANALog]:REFerence:MEASured
<measurement_type>,<source_channel>,<@target_channel_list>
```

Description

Sets the last measurement result obtained from the specified measurement type of the selected source as the reference value for the corresponding target channels.

Parameters

Item	Type	Range of values	Default value
<measurement_type>	Discrete	LEVel|FREQuency|RATio	LEVel
<source_channel>	Discrete	CH1|CH2	Required parameter
<target_channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remarks

– If there is no data in the last measurement or invalid data such as INF, this command will not have any effect; the previous reference value will be remain.

– If the measurement type selection is level, sending this command will affect the reference level which is set using the **SENSe[:ANALog]:REFerence:LEVel** command. You can use the command `SENSe[:ANALog]:REFerence:LEVel?` to query for the reference level.

– If the measurement type selection is frequency, sending this command will affect the reference frequency, which is set using the **SENSe[:ANALog]:REFerence:FREQuency** command. You can use the command `SENSe[:ANALog]:REFerence:FREQuency?` to query for the reference frequency.

– If the measurement type selection is ratio, sending this command will affect the reference ratio which is set using the **SENSe[:ANALog]:REFerence:RATio** command. You can use the command `SENSe[:ANALog]:REFerence:RATio?` to query for the reference ratio.

Examples

The following command set the measured VAC result of channel 1 as reference value for both channels.

```
SENS:REF:MEAS LEV, CH1,(@1:2)
```

The following query returns the reference level of the measured value. Assume that the measured VAC result for channel 1 is 1 Vrms.

```
SENS:REF:LEV? (@1:2)
```

Typical response:

```
1.000000E+00
```

SENSe[:ANALog]:WAVeform:POINts

Syntax

```
SENSe[:ANALog]:WAVeform:POINts <number_of_points>
```

```
SENSe[:ANALog]:WAVeform:POINts?
```

Description

Sets the number of data points to acquire with the **FETCh:ARRay?** command. The query returns the selected acquisition length.

If you select time domain using the **DISPlay:ANALysis:TYPE** command, the waveform points set are the number of the data points acquired. If you select the frequency domain, either in the magnitude analysis mode or phase analysis mode using the **DISPlay:ANALysis:TYPE** command, the waveform points set are the FFT size instead of the data points acquired. The acquisition length of the frequency domain doubles the data points that you select using this command.

Parameter

Item	Type	Range of values	Default value
`<number_of_points>`	Numeric	2048|4096|8192|16384|32768	2048

Remark

If the number of points that you enter is not the exact value of the acquisition length, the value is always clipped to its lower value. For instance, if the number of points that you enter is 500, it will be clipped to 256 which is the number lower than 500.

Examples

The following command sets the data points to 512.

```
SENS:WAV:POIN 512
```

The following query returns the data points value.

```
SENS:WAV:POIN?
```

Typical response:

```
512
```

SOURce[:ANALog]:MULTitone:TONE:PHASe:RANDomize

Syntax

```
SOURce[:ANALog]:MULTitone:TONE:PHASe:RANDomize (@<channel_list>)
```

Description

Randomizes the phase of all the tones in the multitone waveform for the specified channel(s).

Parameter

Item	Type	Range of values	Default value
<channel_list>	Numeric	One or more channels. – (@1) for channel 1 – (@1,4) for channel 1 and 4 – (@1:6) for channel 1 to 6	Required parameter

Remark

This setting is only applicable for the multitone waveform. Use the **SOURce[:ANALog]:FUNCtion** command to select the waveform type before using this command.

Example

The following command randomizes the phase of all the tones for channel 1.

```
SOUR:MULT:TONE:PHAS:RAND (@1)
```

SYSTem:RESet:CHANnel

Syntax

```
SYSTem:RESet:CHANnel <system_mode>,(@<channel>)
```

Description

Resets the customized settings of the U8903B system mode for the specified channel to the default settings.

Parameters

Item	Type	Range of values	Default value
<system_mode>	Discrete	AANalyzer|AGENerator	Required parameter
<channel>	Numeric	1 to 8	Required parameter

Remarks

- This command resets the customized settings of the system mode for the selected channel excluding the stored files, I/O configuration, and common system settings.
- For the analog analyzer mode, the measurement bandwidth, measurement time, and trigger source will not be reset to the default settings.

Example

The following command resets channel 1 of the analog analyzer mode.

```
SYST:RES:CHAN AAN,(@1)
```

SENSe[:ANALog]:MTIMe

Syntax

```
SENSe:MTIMe <measurement_time>
```

```
SENSe:MTIMe?
```

Description

Sets the analyzer measurement time. The query returns the measurement time.

The measurement time values with their corresponding <measurement_time> parameters are listed as follows.

`GTRack`	Generator track
`SP128`	1/128 s
`SP64`	1/64 s
`SP32`	1/32 s
`SP16`	1/16 s
`SP8`	1/8 s
`SP4`	1/4 s
`SP2`	1/2 s
`S1`	1 s

Parameter

Item	Type	Range of values	Default value
<measurement_time>	Discrete	GTRack|SP128|SP64|SP32|SP16|SP8|SP4|SP2|S1	SP4

Remarks

- This command is only applicable to the analog analyzer card 1.
- This command can only be supported with bandwidth 90 k and below.
- This command is not supported in bandwidth 500 k mode. Error will be generated if you try to send this command in the 1.5 MHz bandwidth mode.
- The relationship between sample size and measurement time is
  Sample size = sampling rate × measurement time

– The measurement time selection will be converted to the corresponding sample size as shown below.

Measurement time	Sample size setting for band width 90k and below
Generator track	Auto select the optimum sample size
1/128 s	2048
1/64 s	2048
1/32 s	4096
1/16 s	8192
1/8 s	16384
1/4 s	32768
1/2 s	65536
1 s	131072

– For the generator track, the U8903B will select the optimum sample size for the analog analyzer based on the frequency of the corresponding generator. Channels 1, 3, 5, and 7 of the analog analyzer will refer to the generator channel 1 frequency while channels 2, 4, 6, and 8 of the analog analyzer will refer to generator channel 2 frequency.

Examples

The following command sets the measurement time to 1/128 s.

```
SENS:MTIM SP128
```

The following query returns the measurement time.

```
SENS:MTIM?
```

Typical response:

```
SP128
```

TRIGger:ANALyzer:SOURce

Syntax

`TRIGger:ANALyzer:SOURce <trigger_source>`

`TRIGger:ANALyzer:SOURce?`

Description

Sets the analyzer trigger source for the input signals. The query returns the trigger source in the form of `IMM`, `BUS`, or `EXT`.

The description for each `<trigger_source>` parameter is listed as follows.

`IMMediate`	Triggers a measurement automatically without waiting for any event to occur.
`BUS`	Triggers a measurement when the ***TRG** command is received.
`EXTernal`	Triggers a measurement when the external signal source connected to the Trigger In connector provides a low-true signal to the U8903B.

Parameter

Item	Type	Range of values	Default value
`<trigger_source>`	Discrete	IMMediate|BUS|EXTernal	IMMediate

Examples

The following command sets the analyzer trigger source to external.

```
TRIG:ANAL:SOUR EXT
```

The following query returns the trigger source.

```
TRIG:ANAL:SOUR?
```

Typical response:

```
EXT
```

TRIGger:GRAPh:SLOPe

Syntax

```
TRIGger:GRAPh:SLOPe <edge>
```

```
TRIGger:GRAPh:SLOPe?
```

Description

Sets the rising or falling edge of the signal to be triggered. The query returns the trigger edge in the form of `POS` or `NEG`.

Parameter

Item	Type	Range of values	Default value
`<edge>`	Discrete	POSitive|NEGative	POSitive

Remark

The trigger edge is only applicable for the graph trigger source of channel 1 to channel 8, else this setting will be ignored.

Examples

The following command sets the rising edge of the signal.

```
TRIG:GRAP:SLOP POS
```

The following query returns the trigger edge.

```
TRIG:GRAP:SLOP?
```

Typical response:

```
POS
```

TRIGger:GRAPh:SOURce

Syntax

```
TRIGger:GRAPh:SOURce <trigger_source>
```

```
TRIGger:GRAPh:SOURce?
```

Description

Sets the graph trigger source for the input signals. The query returns the trigger source in the form of either `IMM`, `EXT`, `BUS`, `CH1`, `CH2`. `CH3`, `CH4`, `CH5`, `CH6`, `CH7`, or `CH8`.

The description for each `<trigger_source>` parameter is listed as follows.

`IMMediate`	Free run
`EXTernal`	Triggers from an external source
`BUS`	Triggers from the internal bus
`CH1`	Triggers from the channel 1 input
`CH2`	Triggers from the channel 2input
`CH3`	Triggers from the channel 3 input
`CH4`	Triggers from the channel 4 input
`CH5`	Triggers from the channel 5 input
`CH6`	Triggers from the channel 6 input
`CH7`	Triggers from the channel 7 input
`CH8`	Triggers from the channel 8 input

Parameter

Item	Type	Range of values	Default value
`<trigger_source>`	Discrete	IMMediate|EXTernal|BUS|CH1|CH2|CH3|CH4|CH5|CH6|CH7|CH8	IMMediate

Example

The following command sets the graph trigger source to external.

```
TRIG:GRAP:SOUR EXT
```

The following query returns the trigger source.

```
TRIG:GRAP:SOUR?
```

Typical response:

```
EXT
```

INPut:DIGital:AES:AUDio[:DECoding]:FORMat

Syntax

```
INPut:DIGital:AES:AUDio[:DECoding]:FORMat<format>
INPut:DIGital:AES:AUDio[:DECoding]:FORMat?
```

Description

Sets the audio decoding format of the embedded AES3/SPDIF interface audio signals. The query returns the audio decoding format.

LPCM	Linear Pulse Code Modulation
ULAW	µ-Law decoding format
ALAW	A-Law decoding format

Parameter

Item	Type	Range of values	Default value
<format>	Discrete	LPCM|ULAW|ALAW	LPCM

Examples

The following command sets the decoding format of the AES3/SPDIF interface to A-Law.

```
INP:DIG:AES:AUD:FORM ALAW
```

The following query returns the decoding format.

```
INP:DIG:AES:AUD:FORM?
```

Typical response:

```
ALAW
```

INPut:DIGital:AES:AUDio:RESolution

Syntax

INPut:DIGital:AES:AUDio:RESolution <resolution>

INPut:DIGital:AES:AUDio:RESolution?

Description

Sets the audio resolution or bit depth for the AES3/SPDIF interface audio data to be analyzed. The query returns the audio resolution.

Parameter

Item	Type	Range of values	Default value
`<resolution>`	Numeric	8 to 24	24

Remarks

- If the audio resolution value matches or exceeds the resolution of the incoming digital signal, the signal is passed on.
- If the audio resolution value is lower than the resolution of the incoming digital signal, the signal is truncated at the least significant bit (LSB).

Examples

The following command sets the audio resolution to 20 bits.

```
INP:DIG:AES:AUD:RES 20
```

The following query returns the audio resolution.

```
INP:DIG:AES:AUD:RES?
```

Typical response:

```
20
```

INPut:DIGital:DSI:AUDio[:DECoding]:FORMat

Syntax

```
INPut:DIGital:DSI:AUDio[:DECoding]:FORMat<format>
```

```
INPut:DIGital:DSI:AUDio[:DECoding]:FORMat?
```

Description

Sets the audio decoding format of the embedded DSI interface audio signals. The query returns the audio decoding format.

LPCM	Linear Pulse Code Modulation
ULAW	µ-Law decoding format
ALAW	A-Law decoding format

Parameter

Item	Type	Range of values	Default value
<format>	Discrete	LPCM|ULAW|ALAW	LPCM

Examples

The following command sets the decoding format of the DSI interface to A-Law.

```
INP:DIG:DSI:AUD:FORM ALAW
```

The following query returns the decoding format.

```
INP:DIG:DSI:AUD:FORM?
```

Typical response:

```
ALAW
```

INPut:DIGital:DSI:AUDio:RESolution

Syntax

```
INPut:DIGital:DSI:AUDio:RESolution <resolution>
```

```
INPut:DIGital:DSI:AUDio:RESolution?
```

Description

Sets the audio resolution or bit depth for the DSI interface audio data to be analyzed. The query returns the audio resolution.

Parameter

Item	Type	Range of values	Default value
`<resolution>`	Numeric	8 to 24	24

Remarks

- If the audio resolution value matches or exceeds the resolution of the incoming digital signal, the signal is passed on.
- If the audio resolution value is lower than the resolution of the incoming digital signal, the signal is truncated at the least significant bit (LSB).

Examples

The following command sets the audio resolution to 20 bits.

```
INP:DIG:DSI:AUD:RES 20
```

The following query returns the audio resolution.

```
INP:DIG:DSI:AUD:RES?
```

Typical response:

```
20
```

SENSe:DIGital:REFerence:MEASured

Syntax

```
SENSe:DIGital:REFerence:MEASured <measurement_type>,
<source_channel>, <@target_channel_list>
```

Description

Sets the last measurement result obtained from the specified measurement type of the selected source as the reference value for the corresponding target channels.

Parameters

Item	Type	Range of values	Default value
<measurement_type>	Discrete	LEVel|FREQuency|RATio	LEVel
<source_channel>	Discrete	CH1|CH2	Required parameter
<target_channel_list>	Discrete	One or more channels. – (@D1) or (@D2) for single channel – (@D1,D2) for channel 1 and 2	Required parameter

Remarks

- If there is no data in the last measurement or invalid data such as INF, this command will not have any effect; the previous reference value will be remain.
- If the measurement type selection is level, sending this command will affect the reference level which is set using `SENSEe:DIGital:REFerence:LEVel`. You can use the command `SENSe:DIGital:REFerence:LEVel?` to query for the reference level.
- If the measurement type selection is frequency, sending this command will affect the reference frequency, which is set using `SENSEe:DIGital:REFerence:FREQuency`. You can use the command `SENSe:DIGital:REFerence:FREQuency?` to query for the reference frequency.
- If the measurement type selection is ratio, sending this command will affect the reference ratio which is set using `SENSEe:DIGital:REFerence:RATio`. You can use the command `SENSe:DIGital:REFerence:RATio?` to query for the reference ratio.

Examples

The following command sets the measured Vac result of channel 1 as reference value for both channels.

```
SENS:DIG:REF:MEAS LEV, CH1, (@D1:D2)
```

The following query returns the reference level of the measured value. Assume that the measured Vac result for channel 1 is 1 FFS.

```
SENS:DIG:REF:LEV? (@D1:D2)
```

Typical response:

```
1.000000E+00
```

SENSe:DIGital:WAVeform:POINts

Syntax

```
SENSe:DIGital:WAVeform:POINts <number_of_points>
```

```
SENSe:DIGital:WAVeform:POINts?
```

Description

Sets the number of data points to acquire with the `FETCh:ARRAy?` command. The query returns the selected acquisition length.

If you select the frequency domain analysis, the acquisition length represents the FFT size. The acquisition length of the frequency domain analysis doubles the acquisition length that you select using this command.

Parameter

Item	Type	Range of values	Default value
`<number_of_points>`	Numeric	256|512|1024|2048|4096|8192|16384|32768	1024

Remark

If the number of points that you enter is not the exact value of the acquisition length, the value is always clipped to its lower value. For instance, if the number of points that you enter is 500, it will be clipped to 256 which is the number lower than 500.

Examples

The following command sets the acquisition length to 512.

```
SENS:DIG:WAV:POIN 512
```

The following query returns the acquisition length.

```
SENS:DIG:WAV:POIN?
```

Typical response:

```
512
```

CALCulate:DIGital:FORMat:LEVel

Syntax

```
CALCulate:DIGital:FORMat:LEVel <format>,(@<channel>)
CALCulate:DIGital:FORMat:LEVel? (@<channel>)
```

Description

Sets the format of the measurement data for the specified channel(s) for level measurement function such as VAC, VDC, Noise Level, THD+N Level, Positive Peak, and Negative Peak. The query returns the format of the measurement data of the selected channel(s).

The description for each `<format>` parameter is shown as follows.

OFF	Returns the measurement data in V.
LOGarithmic	Returns the measurement data in dBr.
LINear	Returns the measurement data in x.

Parameters

Item	Type	Range of values	Default value
<format>	Discrete	OFF|LOGarithmic|LINear	OFF
<channel>	Numeric	1 or 2	1

Remark

After this command is sent, the calculated measurement data can be acquired using the `FETCh:DIGital:SCALar?` command for analyzer mode or the `FETCh:SWEep?` command for sweep mode.

Examples

The following command sets the level measurement format to linear for channel 1.

```
CALC:DIG:FORM:LEV LIN, (@D1)
```

The following query returns the format of the measurement data for channel 1.

```
CALC:DIG:FORM:LEV? (@D1)
```

Typical response:

```
LIN
```

CALCulate:DIGital:FORMat:FREQuency

Syntax

`CALCulate:DIGital:FORMat:FREQuency <format>,(@<channel>)`

`CALCulate:DIGital:FORMat:FREQuency? (@<channel>)`

Description

Sets the format of the measurement data for the specified channel(s) for frequency measurement function. The query returns the format of the measurement data of the selected channel(s).

The description for each <format> parameter is shown as follows.

`OFF`	Returns the measurement data in Hz.
`DELTa`	Returns the measurement data in ΔHz.

Parameters

Item	Type	Range of values	Default value
`<format>`	Discrete	OFF|DELTa	OFF
`<channel>`	Numeric	1 or 2	1

Remark

After this command is sent, the calculated measurement data can be acquired using the `FETCh:DIGital:SCALar?` command for analyzer mode or the `FETCh:SWEep?` command for sweep mode.

Examples

The following command sets the frequency measurement format to delta for channel 1.

`CALC:DIG:FORM:FREQ DELT, (@D1)`

The following query returns the format of the measurement data for channel 1.

`CALC:DIG:FORM:FREQ? (@D1)`

Typical response:

`DELT`

CALCulate:DIGital:FORMat:RATio

Syntax

```
CALCulate:DIGital:FORMat:RATio <format>,(@<channel>)
CALCulate:DIGital:FORMat:RATio? (@<channel>)
```

Description

Sets the format of the measurement data for the specified channel(s) for ratio measurement function such as SINAD and THD Ratio, DFD, IMD, crosstalk, and SNR. The query returns the format of the measurement data of the selected channel(s).

The description for each <format> parameter is shown as follows.

OFF	Returns the measurement data in dB.
DELTa	Returns the measurement data in ΔdB.
LINear	Returns the measurement data in x.

Parameters

Item	Type	Range of values	Default value
<format>	Discrete	OFF|DELTa|LINear	OFF
<channel>	Numeric	1|2	1

Remarks

After this command is sent, the calculated measurement data can be acquired using the `FETCh:DIGital:SCALar?` command for analyzer mode or the `FETCh:SWEep?` command for sweep mode.

Examples

The following command sets the level measurement format to linear for channel 1.

```
CALC:DIG:FORM:LEV LIN, (@D1)
```

The following query returns the format of the measurement data for channel 1.

```
CALC:DIG:FORM:LEV? (@D1)
```

Typical response:

```
LIN
```

Appendix Q: Obsoleted SCPI Commands

- DATA:FILE?
- SENSe[:ANALog]:THDN:NOTCh:BANDwidth
- SOURce[:ANALog]:MULtitone:RLEN
- SYSTem:UPDate:HELP?
- SENSe[:ANALog]:NOTCh:EMULation[:STATe]
- SYSTem:DIGital:CTYPe?
- INPut:DIGital:DSI:DATA:MSB:PADDing
- INPut:DIGital:DSI:MCLK:SOURce
- INPut:DIGital:DSI:MCLK:MULTiplier
- SOURce:DIGital:BERT[:MODE]
- SOURce:DIGital:BERT:PATTern:CATegory
- SOURce:DIGital:BERT:WCONstant:TYPE
- SOURce:DIGital:BERT:PSEudorandom:SEED
- SOURce:DIGital:BERT:WCONstant[:VALue]
- SOURce:DIGital:BERT:DURation
- SOURce:DIGital:BERT:BWIDth
- SOURce:DIGital:BERT[:MODE]
- SOURce:DIGital:BERT:PATTern:CATegory
- SOURce:DIGital:BERT:WCONstant:TYPE
- SOURce:DIGital:BERT:PSEudorandom:SEED
- SOURce:DIGital:BERT:WCONstant[:VALue]
- SOURce:DIGital:BERT:DURation
- SOURce:DIGital:BERT:BWIDth
- SENSe:DIGital:FUNDamental:FREQuency:LOCK:THD
- SENSe:DIGital:FUNDamental:FREQuency:LOCK[:SINad]
- SENSe:DIGital:THD:HARMonic:COUNt
- SENSe:DIGital:BERT:INTerval
- SENSe:DIGital:SNR:DELay
- SENSe:DIGital:SOURce:INTerface
- SENSe:DIGital:SOURce:CHANnel
- CALCulate:DIGital:HARMonic:COUNt

- `INITiate[:IMMediate]:DIGital:BERT`

KEYSIGHT TECHNOLOGIES

This information is subject to change without notice. Always refer to the Keysight Web site for the latest revision.

© Keysight Technologies 2014

Edition 1, November 2014

U8903-90052
www.keysight.com