



DAQ4000A Programming Manual (V 1.01)

Copyright Notice

Copyright

Qingdao Hantek Electronics Co., Ltd.

Statement

Qingdao Hantek Electronics Co., Ltd. reserves the right to modify this document without prior notice. Qingdao Hantek Electronics Co., Ltd. commits to providing accurate and reliable information, but does not guarantee that this document is free of errors. Please verify that the relevant technical specifications and documents you use are the latest and valid versions before using this product. If your company requires third-party products, patents, or copyrights to be used in conjunction with Qingdao Hantek Electronics Co., Ltd.'s documents or products, your company shall be responsible for obtaining the necessary third-party consent and authorization. Regarding the aforementioned consent and authorization, this is not the responsibility of our company to guarantee. Technical Support If you have any questions or uncertainties while using Qingdao Hantek Electronics Co., Ltd.'s products, you may obtain service and support through the following methods:

1. Please contact the local distributor of Qingdao Hantek Electronics Co., Ltd. ;
2. Please contact the local branch office of Qingdao Hantek Electronics Co., Ltd.
3. Please contact the headquarters of Qingdao Hantek Electronics Co., Ltd.

Company Contact Information:

Qingdao Hantek Electronics Co., Ltd.

<http://www.hantek.com>

Address: Building 35, No. 780 Baoyuan Road, High-Tech Zone, Shandong Province,

Postal Code: 266114

Phone: 0532-55678770/72/73

Fax: 0532-88705691

Technical Support:

Email: support@hantek.com

Contents

Copyright Notice	1
Copyright	1
Statement	1
SCPI Command Overview	1
Command format	1
Symbol explanation	1
Parameter type	2
Command abbreviations	3
SCPI command	4
R subcommand	4
R?	4
READ subcommand	4
READ?	4
UNIT subcommand	5
UNIT:TEMPerature	5
ABORt subcommand	5
ABORt	5
CALCulate:LIMit subcommand	5
CALCulate:LIMit:{LOWer UPPer}	5
CALCulate:LIMit:{LOWer UPPer}?	6
CALCulate:LIMit:{LOWer UPPer}:STATe	6
CALCulate:LIMit:{LOWer UPPer}:STATe?	6
CALCulate:SCALe subcommand	7
CALCulate:SCALe:DB:REFerence	7
CALCulate:SCALe:DB:REFerence?	7
CALCulate:SCALe:DBM:REFerence	7
CALCulate:SCALe:DBM:REFerence?	8
CALCulate:SCALe:FUNcTion	8
CALCulate:SCALe:FUNcTion?	8
CALCulate:SCALe:GAIN	9

CALCulate:SCALE:GAIN?	9
CALCulate:SCALE:OFFSet.....	9
CALCulate:SCALE:OFFSet?	10
CALCulate:SCALE:REFerence.....	10
CALCulate:SCALE:REFerence?.....	10
CALCulate:SCALE[:STATe].....	11
CALCulate:SCALE[:STATe]?.....	11
CALCulate:SCALE:UNIT	11
CALCulate:SCALE:UNIT?	12
CALCulate:SCALE:UNIT:STATe.....	12
CALCulate:SCALE:UNIT:STATe?.....	12
CONFigure command.....	13
CONFigure?.....	13
CONFigure:CAPacitance.....	13
CONFigure:CURREnt:{AC DC}.....	13
CONFigure:DIODE.....	14
CONFigure:{FREQuency PERiod}	14
CONFigure:{FREsistance RESistance}.....	15
CONFigure:STRain:{DIRect FDIRect}.....	15
CONFigure:STRain:{FULL HALF}:BENDing.....	16
CONFigure:STRain:{FULL HALF}: POISSon	16
CONFigure:STRain:QUARter	17
CONFigure:TEMPerature:<probe_type>	17
CONFigure[:VOLTage]:{AC DC}.....	18
DATA command.....	18
DATA:LAST?.....	18
DATA:POINts?.....	18
DATA:REMOve?	19
DISPlay command.....	19
DISPlay	19
DISPlay:TEXT.....	19
DISPlay:TEXT:CLEar	20

FETCh command.....	20
FETCh?.....	20
FORMat command.....	20
FORMat:READIng:ALARm.....	20
FORMat:READIng:ALARm?.....	21
FORMat:READIng:CHANnel.....	21
FORMat:READIng:CHANnel?.....	22
FORMat:READIng:TIME.....	22
FORMat:READIng:TIME?.....	22
FORMat:READIng:TIME:TYPE.....	23
FORMat:READIng:TIME:TYPE?.....	23
FORMat:READIng:UNIT.....	23
FORMat:READIng:UNIT?.....	24
FUNC command.....	24
FUNC.....	24
FUNC?.....	25
IEEE-488.2 command.....	25
*CLS.....	25
*IDN?.....	25
*RST.....	26
*SAV.....	26
*RCL.....	26
INITiate command.....	26
INITiate.....	26
MEASure command.....	27
MEASure:CAPacitance?.....	27
MEASure:CURRent:{AC DC}?.....	27
MEASure:{FRESistance RESistance}?.....	27
MEASure:{FREQuency PERiod}?.....	28
MEASure:DIODE?.....	28
MEASure:TEMPerature: <probe_type>?.....	28
MEASure:STRain:{DIRect FDIRect}?.....	29

MEASure:STRain:{FULL HALF}:BENDing?	29
MEASure:STRain:{FULL HALF}:POISson?	29
MEASure:STRain:FULL:BENDing:POISson?	30
MEASure:STRain:QUARter?	30
MEASure[:VOLTage]:{AC DC}?	30
MMEMory subcommand	31
MMEMory:FORMat:READing:CSEParator	31
MMEMory:FORMat:READing:RLIMit	31
MMEMory:FORMat:READing:RLIMit:COUNT	32
MMEMory subcommand	32
MMEMory:LOAD:STATe	32
MMEMory:STORE:STATe	32
MMEMory:STATe:RECall:AUTO	33
MMEMory:STATe:RECall:AUTO?	33
MMEMory:STATe:VALId?	33
MMEMory subcommand	34
MMEMory:CATalog[:ALL]?	34
MMEMory:COpy	34
MMEMory:DElete	34
MMEMory:LOG[:ENABle]	35
MMEMory:LOG[:ENABle]?	35
MMEMory:LOG:NEW	35
MMEMory:MOVE	35
OUTPut command	36
OUTPut:ALARm[{1 2 3 4}]:SOURce?	36
OUTPut:ALARm:MODE?	36
OUTPut:ALARm:SLOPe?	36
OUTPut:TRIGger:SLOPe?	37
ROUte command	37
ROUte:OPEN	37
ROUte:CLOSe	37
ROUte:CHANnel:DELay?	38

ROUTe:CHANnel:DELay:AUTO?.....	38
ROUTe:MONitor.....	38
ROUTe:MONitor?.....	39
ROUTe:MONitor:STATe?	39
ROUTe:SCAN.....	39
ROUTe:SCAN?.....	39
ROUTe:SCAN:SIZE?	40
[SENSE:]CAPacitance command.....	40
[SENSe:]CAPacitance:RANGe:AUTO	40
[SENSe:]CAPacitance:RANGe.....	40
[SENSE:]CURRent subcommand.....	41
[SENSe:]CURRent[:DC]:NPLCycles.....	41
[SENSe:]CURRent[:DC]:ZERO:AUTO	41
[SENSe:]CURRent:AC:BANDwidth	42
[SENSE:] {FREQuency PERiod} subcommand.....	42
[SENSe:]{FREQuency PERiod}:APERture	42
[SENSe:]{FREQuency PERiod}:RANGe:LOWer.....	43
[SENSe:]{FREQuency PERiod}:VOLTage:RANGe	43
[SENSe:]{FREQuency PERiod}:VOLTage:RANGe:AUTO	43
[SENSE:] {RESistance FRESistance} subcommand.....	44
[SENSe:]{RESistance FRESistance}:NPLCycles	44
[SENSe:]{RESistance FRESistance}:OCOMPensated.....	44
[SENSe:]{RESistance FRESistance}:POWER:LIMit[:STATe].....	45
[SENSe:]{RESistance FRESistance}:RANGe	45
[SENSe:]{RESistance FRESistance}:RANGe:AUTO.....	46
[SENSe:]RESistance:ZERO:AUTO.....	46
[SENSE:] STRain subcommand	46
[SENSe:]STRain:EXCitation	46
[SENSe:]STRain:EXCitation:TYPE	47
[SENSe:] STRain:GFACtor	47
[SENSe:] STRain:NPLCycles.....	48
[SENSe:] STRain: OCOMPensated.....	48

[SENSe:]STRain:POISson.....	48
[SENSe:]STRain:RESistance	49
[SENSe:]STRain:UNSTrained.....	49
[SENSe:]STRain:VOLTage:RANGe:AUTO	50
[SENSe:]STRain:VOLTage:RANGe.....	50
[SENSe:]STRain:ZERO:AUTO	50
[SENSE:]TEMPerature subcommand.....	51
[SENSe:]TEMPerature:NPLCycles.....	51
[SENSe:]TEMPerature:RJUNction?.....	51
[SENSe:]TEMPerature:TRANsducer:{RTD FRTD}:OCOMpensated.....	52
[SENSe:]TEMPerature:TRANsducer:{RTD FRTD}:POWER:LIMit[:STATe].....	52
[SENSe:]TEMPerature:TRANsducer:{RTD FRTD}:REFerence	53
[SENSe:]TEMPerature:TRANsducer:{RTD FRTD}:RESistance[:REFerence].....	53
[SENSe:]TEMPerature:TRANsducer:{FTHermistor THERmistor}:POWER:LIMit[:STATe]	54
[SENSe:]TEMPerature:TRANsducer:{FTHermistor THERmistor}:REFerence	54
[SENSe:]TEMPerature:TRANsducer:{FTHermistor THERmistor}:TYPE.....	55
[SENSe:]TEMPerature:TRANsducer:TCouple:TYPE.....	55
[SENSe:]TEMPerature:TRANsducer:TYPE	55
[SENSe:]TEMPerature:ZERO:AUTO	56
[SENSE:]VOLTage subcommand.....	56
[SENSe:]VOLTage:AC:BANDwidth.....	56
[SENSe:]VOLTage:{AC DC}:RANGe:AUTO	57
[SENSe:]VOLTage:{AC DC}:RANGe.....	57
[SENSe:]VOLTage[:DC]:IMPedance:AUTO	57
[SENSe:]VOLTage[:DC]:NPLCycles	58
[SENSe:]VOLTage[:DC]:ZERO:AUTO	58
SYSTem subcommand	59
SYSTem:BEEPer[:IMMEDIATE].....	59
SYSTem:BEEPer:STATe	59
SYSTem:CLICk:STATe.....	59
SYSTem:CTYPE?	60

SYSTem:CPON.....	60
SYSTem:DATE	60
SYSTem:ERRor?	61
SYSTem:LOCal	61
SYSTem:PERSonA[:MANuFActurer]	61
SYSTem:PERSonA[:MANuFActurer]:DEFault	61
SYSTem:PERSonA:MODEl.....	62
SYSTem:PERSonA:MODEl:DEFault.....	62
SYSTem:TIME	62
SYSTem:TIME:SCAN?	63
SYSTem:VERSion?.....	63
SYSTem:WMESsage	63
SYSTem:REMote.....	64
SYSTem subcommand	64
SYSTem:COMMunicate:LAN:DHCP	64
SYSTem:COMMunicate:LAN:IPADdress	64
SYSTem:COMMunicate:LAN:SMASk	65
SYSTem:COMMunicate:LAN:UPD	65
TRIGger command.....	65
TRIGger:COUNt.....	65
TRIGger:COUNt?	66
TRIGger:SLOPe?	66
TRIGger:SOURe	66
TRIGger:SOURce?	67
TRIGger:TIMer	67
TRIGger:TIMer?.....	67

SCPI Command Overview

SCPI (Standard Commands for Programmable Instruments) is the standard command set for programmable instruments. SCPI is a standardized instrument programming language based on existing standards such as IEEE 488.1 and IEEE 488.2, and adheres to multiple standards including the floating-point arithmetic rules specified in IEEE 754, the 7-bit encoding symbols for information exchange defined in ISO 646 (equivalent to ASCII programming), and others. SCPI commands are organized in a tree-like hierarchical structure, comprising multiple subsystems, each of which consists of a root keyword and one or more hierarchical keywords.

Command format

Commands usually begin with a colon “:”. Keywords are separated by colons “:”, followed by optional parameter settings. Adding a question mark “?” after the command line indicates a query for this function. Command keywords and the first parameter are separated by spaces.

For example:

```
:ACquire:TYPE <type>
:ACquire:TYPE?
```

ACquire is the root keyword of the command, and TYPE is the second-level keyword. The command line begins with a colon (:), and colons (:) are used to separate keywords at each level. <type> represents the parameters that can be set. A question mark (?) indicates a query. The command keywords ACquire:TYPE and the parameter <type> are separated by spaces.

In some commands with parameters, multiple parameters are typically separated by commas “,” for example:

```
[:TRACe[<n>]]:DATA:VALue volatile,<points>,<data>
```

Symbol explanation

The following symbols are not sent with the command.

1. Curly brackets {}

The content within curly brackets is a parameter option. Parameter options are typically separated by a vertical bar “|”. When using the command, you must select one of the parameters.

2. Vertical bar |

The vertical bar is used to separate multiple parameter options. When using the command, you must select one of the parameters.

3. Square brackets []

The contents within square brackets are optional.

4. Angle brackets <>

The parameters within angle brackets must be replaced with a valid value.

Parameter type

1. Boolean (Bool) Parameter values are ON, OFF, 1, or 0. For example:

```
:MEASure:ADISplay <bool> :MEASure:ADISplay?
```

Where:

<bool> can be set to: `{1|ON}|{0|OFF}`. The query returns 1 or 0.

2. Discrete (Discrete)

Parameter values are the listed options. For example:

```
:ACQuire:TYPE <type> :ACQuire:TYPE? Where: <type> can be set to:  
NORMal|AVERages|PEAK|HRESolution. The query returns the abbreviated form: NORM,  
AVER, PEAK, or HRES.
```

3. Integer

Unless otherwise specified, the parameter can be any integer within the valid range (NR1 format). Note that the parameter should not be set to decimal format, otherwise an exception will occur. For example:

```
:DISPlay:GBrightness <brightness> :DISPlay:GBrightness? Where: <brightness>  
can be set to: an integer between 0 and 100. The query returns an integer  
between 0 and 100.
```

4. Real

Parameters can be any real number within the valid range. This command accepts decimal (NR2 format) and scientific notation (NR3 format) parameter inputs.

For example:

```
:TRIGger:TIMEout:TIME <NR3> :TRIGger:TIMEout:TIME?
```

Where: The parameter <NR3> can be set to any real number between 1.6e-08 (i.e., 16 ns) and 1e+01 (i.e., 10 s). The query returns a real number in scientific notation format.

5. ASCII String Parameters take values as combinations of ASCII characters.

For example:

```
:SYSTem:OPTion:INSTall <license> Where: <license> can be set to:  
PDUY9N9QTS9PQSWPLAETRD3UJHYA
```

Command abbreviations

All commands are case-insensitive, so you can use either uppercase or lowercase letters. However, if you want to abbreviate a command, you must enter all uppercase letters in the command format.

For example:

:MEASure:ADISplay? can be abbreviated to MEAS:ADIS?

SCPI command

R subcommand

R?

This command reads and deletes scan data in SCAN mode.

Command syntax

```
R? [<max_rdgs>]
```

Parameter range

```
<max_rdgs> = 1-100000
```

Return parameters

```
Scan data
```

Example

```
#247+8.11900000E-03,+5.15280000E-03,+3.11220000E-03
```

For example

```
R? 3
```

READ subcommand

READ?

This command enables and reads scan data in SCAN mode.

Command syntax

```
READ?
```

Return parameters

```
Scan data
```

Example

```
+4.27150000E-03,+1.32130000E-03
```

For example

```
READ?
```

UNIT subcommand

UNIT:TEMPerature

This command is used to set the unit for the channel's Temp mode.

Command syntax

```
UNIT:TEMPerature
```

Parameters

```
{C|F|K}, [(@<ch_list>)]
```

For example

```
UNIT:TEMP F, (@101)
```

Query command

```
UNIT:TEMPerature? (@101)
```

Return parameters

```
{C|F|K}
```

ABORt subcommand

ABORt

This command scans and stops the ongoing measurement, returning the instrument to the trigger idle state.

Command syntax

```
ABORt
```

For example

```
ABOR
```

CALCulate:LIMit subcommand

CALCulate:LIMit:{LOWer|UPPer}

This command is used to set the upper and lower limits for the alarm.

Command syntax

```
CALCulate:LIMit:{LOWer|UPPer}
```

Parameters

```
<limit>|MIN|MAX|DEF [, (@<ch_list>)]
```

For example

CALC:LIM:LOW -0.25, (@101, 102)

Query command

CALC:LIM:LOW? (@101, 102)

CALCulate:LIMit:{LOWer|UPPer}?

This command is used to read the upper and lower alarm limits of the Alarm.

Command syntax

CALCulate:LIMit:{LOWer|UPPer}?

Parameters

[{@<ch_list>}|MIN|MAX|DEF]

For example

CALC:LIM:LOW? (@101, 102)

Return parameters

-2.50000000E-01, -2.50000000E-01

CALCulate:LIMit:{LOWer|UPPer}:STATE

This command is used to enable the upper and lower limit values for the alarm.

Command syntax

CALCulate:LIMit:{LOWer|UPPer}:STATE

Parameters

{OFF|ON} [, (@<ch_list>)]

For example

CALC:LIM:LOW:STAT ON, (@101, 102)

Query command

CALC:LIM:LOW:STAT? (@101, 102)

CALCulate:LIMit:{LOWer|UPPer}:STATE?

This command is used to obtain the upper and lower limit values of the Alarm switch.

Command syntax

CALCulate:LIMit:{LOWer|UPPer}:STATE?

Parameters

[{@<ch_list>}]

For example

CALC:LIM:LOW:STAT? (@101, 102)

Return parameters

0(OFF) or 1(ON)

CALCulate:SCALE subcommand

CALCulate:SCALE:DB:REference

This command is used to set the Db Ref Value of the DB function in Math.

Command syntax

```
CALCulate:SCALE:DB:REference
```

Parameters

```
{ <reference>|MIN|MAX|DEF } [, (@<ch_list>)]
```

Parameter range

```
<reference> = -200.0 dBm - +200.0 dBm
```

For example

```
CALC:SCAL:DB:REF -10, (@101)
```

Query command

```
CALC:SCAL:DB:REF? (@101)
```

CALCulate:SCALE:DB:REference?

This command is used to obtain the Db Ref Value of the DB function in Math.

Command syntax

```
CALCulate:SCALE:DB:REference?
```

Parameters

```
[ { (@<ch_list>)} |MIN|MAX|DEF ]
```

For example

```
CALC:SCAL:DB:REF? (@101)
```

Return parameters

```
+3.00000000E+02
```

CALCulate:SCALE:DBM:REference

This command is used to set Ref R in the DB or DBm function in Math.

Command syntax

```
CALCulate:SCALE:DBM:REference
```

Parameters

```
{ <reference>|MIN|MAX|DEF } [, (@<ch_list>)]
```

Parameter range

<reference> = 50, 75, 93, 110, 124, 125, 135, 150, 250, 300, 500,
600, 800, 900, 1000, 1200 or 8000 Ω .

For example

CALC:SCAL:DBM:REF 300, (@101)

Query command

CALC:SCAL:DBM:REF? (@101)

CALCulate:SCALE:DBM:REFerence?

This command is used to obtain the Ref R of the DB or DBm function in Math.

Command syntax

CALCulate:SCALE:DB:REFerence?

Parameters

[{ (@<ch_list>)} | MIN | MAX | DEF]

For example

CALC:SCAL:DBM:REF? (@101)

Return parameters

+3.00000000E+02

CALCulate:SCALE:FUNcTion

This command is used to set Math functions.

Command syntax

CALCulate:SCALE:FUNcTion

Parameters

{DB|DBM|PCT|SCALE} [, (@<ch_list>)]

For example

CALC:SCAL:FUNC SCAL, (@101)

Query command

CALC:SCAL:FUNC? (@101)

CALCulate:SCALE:FUNcTion?

This command is used to obtain Math functions.

Command syntax

CALCulate:SCALE:FUNcTion?

Parameters

[(@<ch_list>)]

For example

CALC:SCAL:FUNC? (@101)

Return parameters

DB, DBM, PCT, or SCAL

CALCulate:SCALE:GAIN

This command is used to set the Gain of Math functions.

Command syntax

CALCulate:SCALE:GAIN

Parameters

{ <gain>|MIN|MAX|DEF } [, (@<ch_list>)]

Parameter range

-1.0E+15 - +1.0E15

For example

CALC:SCAL:GAIN 1.25, (@101)

Query command

CALC SCAL:GAIN? (@101)

CALCulate:SCALE:GAIN?

This command is used to obtain the Gain of the Math function.

Command syntax

CALCulate:SCALE:GAIN?

Parameters

[(@<ch_list>) |MIN|MAX|DEF]

For example

CALC:SCAL:GAIN? (@101)

Return parameters

+1.25000000E+00

CALCulate:SCALE:OFFSet

This command is used to set the offset of Math functions.

Command syntax

CALCulate:SCALE:OFFSet

Parameter range

-1.0E+15 - +1.0E15

Parameters

{ < offset>|MIN|MAX|DEF }[, (@<ch_list>)]

For example

CALC:SCAL:OFFS 1.25, (@101)

Query command

CALC:SCAL:OFFS? (@101)

CALCulate:SCALE:OFFSet?

This command is used to obtain the OFFSet of Math functions.

Command syntax

CALCulate:SCALE:OFFSet?

Parameters

[(@<ch_list>) |MIN|MAX|DEF]

For example

CALC:SCAL:OFFS? (@101)

Return parameters

+1.25000000E+00

CALCulate:SCALE:REFerence

This command is used to set the Ref Value of the % function in Math.

Command syntax

CALCulate:SCALE:REFerence

Parameter range

-1.0E+15 - +1.0E15

Parameters

{ < offset>|MIN|MAX|DEF }[, (@<ch_list>)]

For example

CALC:SCAL:REF 1.25, (@101)

Query command

CALC:SCAL:REF? (@101)

CALCulate:SCALE:REFerence?

This command is used to obtain the Ref Value of Math functions.

Command syntax

CALCulate:SCALE:REFerence?

Parameters

[(@<ch_list>) | MIN | MAX | DEF]

For example

CALC:SCAL:REF? (@101)

Return parameters

+1.25000000E+00

CALCulate:SCALE [:STATe]

This command is used to enable Math.

Command syntax

CALCulate:SCALE[:STATe]

Parameters

{OFF|ON} [, (@<ch_list>)]

For example

CALC:SCAL:STAT ON, (@101)

Query command

CALC:SCAL:STAT? (@101)

CALCulate:SCALE [:STATe]?

This command is used to obtain Math switches.

Command syntax

CALCulate:SCALE[:STATe]?

Parameters

[(@<ch_list>) | MIN | MAX | DEF]

For example

CALC:SCAL:STAT? (@101)

Return parameters

0(OFF) or 1(ON)

CALCulate:SCALE:UNIT

This command is used to set custom units for Math's $Mx+b$.

Command syntax

CALCulate:SCALE:UNIT

Parameters

```
"<string>" [, (@<ch_list>)]
```

For example

```
CALC:SCAL:UNIT "PSI", (@101)
```

Query command

```
CALC:SCAL:UNIT? (@101)
```

CALCulate:SCALE:UNIT?

This command is used to set custom units for Math's $Mx+b$.

Command syntax

```
CALCulate:SCALE:UNIT?
```

Parameters

```
[(@<ch_list>)]
```

For example

```
CALC:SCAL:UNIT? (@101)
```

Return parameters

```
"PSI"
```

CALCulate:SCALE:UNIT:STATE

This command is used to enable custom units for Math's $Mx+b$.

Command syntax

```
CALCulate:SCALE:UNIT:STATE
```

Parameters

```
{OFF|ON} [, (@<ch_list>)]
```

For example

```
CALC:SCAL:UNIT:STAT ON, (@101)
```

Query command

```
CALC:SCAL:UNIT:STAT? (@101)
```

CALCulate:SCALE:UNIT:STATE?

This command is used to enable custom units for Math's $Mx+b$.

Command syntax

```
CALCulate:SCALE:UNIT:STATE?
```

Parameters

```
[(@<ch_list>)]
```

For example

CALC:SCAL:UNIT:STAT? (@101)

Return parameters

0(OFF) or 1(ON)

CONFigure command

CONFigure?

This command reads the measurement phase of the channel in SCAN mode, range.

Command syntax

CONFigure? [(@<ch_list>)]

Return parameters

Measurement phase, range

Example

RES +1.000000E+03,+1.000000E-01

For example

CONF? (@101)

CONFigure:CAPacitance

This command is used to set the measurement phase and range of the channel in SCAN mode to CAP.

Command syntax

CONFigure:CAPacitance

Parameters

[{<range>|AUTO|MIN|MAX|DEF}, [{<resolution>|MIN|MAX|DEF},]]
(@<ch_list>)

Parameter range

<range> = {1nF|10nF|100nF|1 μ F|10 μ F|100 μ F}.

For example

CONF:CAP (@101)

Query command

CONF? (@101)

CONFigure:CURRent:{AC|DC}

This command is used to set the measurement phase of the channel in SCAN mode to DCI or ACI and the range.

Command syntax

```
CONFigure:CAPacitance: {AC|DC}
```

Parameters

```
[ {<range>|AUTO|MIN|MAX|DEF}, [ {<PLCs>|MIN|MAX|DEF}, ] ]  
(@<ch_list>)
```

Parameter range

```
<range>AC: {AUTO|100  $\mu$  A|1 mA|10 mA|100mA|1A}  
DC: {AUTO|1  $\mu$  A|10  $\mu$  A|100  $\mu$  A|1mA|10mA|100mA|1A}  
<PLCs>AC: {3Hz|20Hz|200Hz}  
DC: {0.02|0.2|1|10|100}
```

For example

```
CONF:CURR:AC 1, (@121)
```

Query command

```
CONF? (@121)
```

CONFigure:DIODE

This command is used to set the measurement phase of the channel in SCAN mode to DIODE and the range.

Command syntax

```
CONFigure:DIODE
```

Parameters

```
(@<ch_list>)
```

For example

```
CONF:DIOD (@101)
```

Query command

```
CONF? (@101)
```

CONFigure:{FREQuency|PERiod}

This command is used to set the measurement phase of the channel in SCAN mode to FREQ or PER and the range.

Command syntax

```
CONFigure: {FREQuency|PERiod}
```

Parameters

```
[ {<range>|AUTO|MIN|MAX|DEF}, [ {<PLCs>|MIN|MAX|DEF}, ] ]  
(@<ch_list>)
```

For example

```
CONF:FREQ (@101)
```

Query command

CONF? (@101)

CONFigure:{FRESistance|RESistance}

This command is used to set the measurement phase of the channel in SCAN mode to OHMS or OHMS4W and the range.

Command syntax

CONFigure:{FRESistance|RESistance}

Parameters

[{<range>|AUTO|MIN|MAX|DEF}, [{<PLCs>|MIN|MAX|DEF},]]
(@<ch_list>)

Parameter range

<range>{100 Ω | 1k Ω | 10k Ω | 100k Ω | 1M Ω | 10M Ω | 100 M Ω }
<PLCs>{0.02|0.2|1|10|100}

For example

CONF:RES (@101)

Query command

CONF? (@101)

CONFigure:STRain:{DIRect|FDIRect}

This command is used to set the measurement phase of the channel in SCAN mode to STRAIN, direct to 2 or 4 wires, and the range.

Command syntax

CONFigure:STRain:{DIRect|FDIRect}

Parameters

[{<gage_ohms>|MIN|MAX|DEF}, [{<gage_factor>|MIN|MAX|DEF},
[{<range>|MIN|MAX|DEF}, [{<PLCs>|MIN|MAX|DEF},]]]] (@<ch_list>)

Parameter range

<gage_ohms> 80 - 1100 ohms
<gage_factor> 0.5 - 5
<range>{100 Ω | 1k Ω | 10k Ω | 100k Ω | 1M Ω | 10M Ω | 100 M Ω | 1G Ω }.
<PLCs>{0.002|0.02|0.2|1|10|100}

For example

CONF:STR:DIR (@101)

Query command

CONF? (@101)

CONFigure:STRain:{FULL|HALF}:BENDing

This command is used to set the measurement phase of the channel in SCAN mode to STRAIN, whether the bridge is fully bent or half bent, and the range.

Command syntax

```
CONFigure:STRain: {FULL|HALF}:BENDing
```

Parameters

```
[{<gage_factor>|MIN|MAX|DEF}, [{<range>|MIN|MAX|DEF}, [{<PLCs>|MIN|MAX|DEF}, ]]] (@<ch_list>)
```

Parameter range

```
<gage_factor> 0.5 - 5
```

```
<range> {100 Ω | 1k Ω | 10k Ω | 100k Ω | 1M Ω | 10M Ω | 100 M Ω | 1G Ω}.
```

```
<PLCs> {0.002|0.02|0.2|1|10|100}
```

For example

```
CONF:STR:FULL:BEND (@101)
```

Query command

```
CONF? (@101)
```

CONFigure:STRain:{FULL|HALF}: POISSon

This command is used to set the measurement phase of the channel in SCAN mode to STRAIN, whether the bridge is fully Poisson or semi-Poisson, and the range.

Command syntax

```
CONFigure:STRain: {FULL|HALF}: POISSon
```

Parameters

```
[{<gage_factor>|MIN|MAX|DEF}, [{<range>|MIN|MAX|DEF}, [{<PLCs>|MIN|MAX|DEF}, ]]] (@<ch_list>)
```

Parameter range

```
<gage_factor> 0.5 - 5
```

```
<range> {100 Ω | 1k Ω | 10k Ω | 100k Ω | 1M Ω | 10M Ω | 100 M Ω | 1G Ω}.
```

```
<PLCs> {0.002|0.02|0.2|1|10|100}
```

For example

```
CONF:STR:FULL:POIS (@101)
```

Query command

```
CONF? (@101)
```

CONFigure:STRain:QUARter

This command is used to set the measurement phase of the channel in SCAN mode to STRAIN for a quarter bridge and range.

Command syntax

```
CONFigure:STRain:QUARter
```

Parameters

```
[{<gage_factor>|MIN|MAX|DEF}, [{<range>|MIN|MAX|DEF}, [{<PLCs>|MIN|MAX|DEF}, ]]] (@<ch_list>)
```

Parameter range

```
<gage_factor> 0.5 - 5
```

```
<range> {100 Ω | 1k Ω | 10k Ω | 100k Ω | 1M Ω | 10M Ω | 100 M Ω | 1G Ω}.
```

```
<PLCs> {0.002|0.02|0.2|1|10|100}
```

For example

```
CONF:STR:QUAR (@101)
```

Query command

```
CONF? (@101)
```

CONFigure:TEMPerature:<probe_type>

This command is used to set the measurement phase of the channel in SCAN mode to TEMP and the mode.

Command syntax

```
CONFigure:TEMPerature:<probe_type>
```

Parameters

```
<probe_type>: {FRTD|RTD|FThermistor|Thermistor|TCouple}  
[{{<type>|DEF}, [{<PLCs>|MIN|MAX|DEF}, ]]} (@<ch_list>)
```

Parameter range

```
<type>: FRTD、 RTD: 100ohms ± 1% or 1000ohms ± 1%
```

```
FTH、 THER: {2252|5000|10000}
```

```
TC: {B|E|J|K|N|R|S|T}
```

```
<PLCs> {0.02|0.2|1|10|100}
```

For example

```
CONF:TEMP:FRTD 100, (@101)
```

Query command

```
CONF? (@101)
```

Return parameters

```
[{<type>|DEF}, [{<PLCs>|MIN|MAX|DEF}, ]]
```

CONFigure[:VOLTage]:{AC|DC}

This command is used to set the measurement phase of the channel in SCAN mode to DCV or ACV and the range.

Command syntax

```
CONFigure[:VOLTage]:{AC|DC}
```

Parameters

```
[{<range>|AUTO|MIN|MAX|DEF}, [{<PLCs>|MIN|MAX|DEF}, ]]  
(@<ch_list>)
```

Parameter range

```
<range>: {100mV|1V|10V|100V|300V}  
<PLCs>AC: {3Hz|20Hz|200Hz}  
DC: {0.002|0.02|0.2|1|10|100}
```

For example

```
CONF:VOLT:AC 100, (@101)
```

Query command

```
CONF? (@101)
```

DATA command

DATA:LAST?

This command reads the latest scan data from a single channel.

Command syntax

```
DATA:LAST?
```

Parameters

```
<num_rdgs>(@<channel>)
```

Example

```
-1.93147367E-04
```

For example

```
DATA:LAST? (@101)
```

DATA:POINTs?

This command reads the number of buffer data items.

Command syntax

```
DATA: POINTs?
```

Parameters

(@<ch_list>)

Example

+1

For example

DATA:POIN? (@101)

DATA:REMove?

This command deletes the scanned storage data and returns.

Command syntax

DATA:REMove?

Parameters

<num_rdgs>[, WAIT]

Return parameters

+8.54530000E+01

For example

DATA:REM? 3

DISPlay command

DISPlay

This command is used to set the screen saver switch for user setting.

Command syntax

DISPlay

Parameters

{OFF|ON}

For example

DISPlay OFF

Query command

DISPlay?

Return parameters

{0|1}

DISPlay:TEXT

This command is used to set the screen saver subtitles for user setting.

Command syntax

DISPlay:TEXT

Parameters

“<string>”

For example

DISPlay:TEXT “Test in progress”

Query command

DISPlay:TEXT?

Return parameters

“<string>”

DISPlay:TEXT:CLEAr

This command is used to clear the letters in the screen saver of user setting.

Command syntax

DISPlay:TEXT:CLEAr

For example

DISP:TEXT:CLEAr

FEtCh command

FEtCh?

This command reads the scanned storage data.

Command syntax

FEtCh?

Parameters

None

Return parameters

+4.27150000E-03, +1.32130000E-03

For example

FEtC?

FORMat command

FORMat:READIng:ALARm

This command sets whether the scan results display includes alerts.

Command syntax

FORMat:READing:ALARm

Parameters

{OFF|ON}

For example

FORM:READ:ALAR ON

Query command

FORM:READ:ALAR?

Return parameters

0(OFF) or 1(ON)

FORMat:READing:ALARm?

This command retrieves the scan results and displays whether they contain any alerts.

Command syntax

FORMat:READing:ALARm

For example

FORM:READ:ALAR?

Return parameters

0(OFF) or 1(ON)

FORMat:READing:CHANnel

This command sets whether the scan results display channel numbers.

Command syntax

FORMat:READing:CHANnel

Parameters

{OFF|ON}

For example

FORM:READ:CHAN ON

Query command

FORM:READ:CHAN?

Return parameters

0(OFF) or 1(ON)

FORMat:READIng:CHANnel?

This command retrieves the scan results to show whether they contain channel numbers.

Command syntax

```
FORMat:READIng:CHANnel?
```

For example

```
FORM:READ:CHAN?
```

Return parameters

```
0(OFF) or 1(ON)
```

FORMat:READIng:TIME

This command sets whether the scan results display include the time.

Command syntax

```
FORMat:READIng:TIME
```

Parameters

```
{OFF|ON}
```

For example

```
FORM:READ:TIME ON
```

Query command

```
FORM:READ:TIME?
```

Return parameters

```
0(OFF) or 1(ON)
```

FORMat:READIng:TIME?

This command retrieves the scan results and displays whether they contain time information.

Command syntax

```
FORMat:READIng:TIME?
```

For example

```
FORM:READ:TIME?
```

Return parameters

```
0(OFF) or 1(ON)
```

FORMat:READIng:TIME:TYPE

This command sets the type of time displayed in the scan results.

Command syntax

```
FORMat:READIng:TIME
```

Parameters

```
{ABSolute|RELative}
```

For example

```
FORM:READ:TIME:TYPE ABS
```

Query command

```
FORM:READ:TIME:TYPE?
```

Return parameters

```
ABS or REL
```

FORMat:READIng:TIME:TYPE?

This command sets the type of time displayed in the scan results.

Command syntax

```
FORMat:READIng:TIME?
```

For example

```
FORM:READ:TIME:TYPE?
```

Return parameters

```
ABS or REL
```

FORMat:READIng:UNIT

This command sets whether the scan results display include units.

Command syntax

```
FORMat:READIng:UNIT
```

Parameters

```
{OFF|ON}
```

For example

```
FORM:READ:UNIT ON
```

Query command

```
FORM:READ:UNIT?
```

Return parameters

```
0 (OFF) or 1 (ON)
```

FORMat:READing:UNIT?

This command retrieves the scan results to show whether they contain units.

Command syntax

```
FORMat:READing:UNIT
```

For example

```
FORM:READ:UNIT?
```

Return parameters

```
0 (OFF) or 1 (ON)
```

FUNC command

FUNC

This command is used to configure different measurement settings, enabling the scanning of various measurement items.

Command syntax

```
FUNC
```

Parameters

```
{<string>}, (@<ch_list>)
```

Parameter Range

```
< string > =
```

DC voltage	AC Voltage	Period	Frequency	2-wire resistance	4-wire resistance	Thermocouple
VOLT	VOLT:AC	PER	FREQ	RES	FRES	TEMP:TC

Diode	Capacitance	Direct strain	4-wire direct strain	Quarter (bending)	Half Bridge (Bending)
DIOD	CAP	STR:DIR	STR:FDIR	STR:QUAR	STR:HALF:BEND

Full Bridge (Poisson)	Full Bridge (Bending Poisson)	Half Bridge (Poisson)	Full Bridge (Bending)	Thermistor	4-wire thermistor
STR:FULL:POIS	STR:FULL:BEND:POIS	STR:HALF:POIS	STR:FULL:BE ND	TEMP:THER	TEMP:FTH

2-wire RTD	4-wire RTD	Direct current	Alternating current
TEMP:RTD	TEMP:FRTD	CURR	CURR:AC

For example

```
FUNC "VOLT", (@101)
```

```
ROUT:SCAN (@101)
```

Command syntax

```
FUNC? (@101)
```

FUNC?

This command retrieves the channel scan type.

Command Syntax

```
FUNC?
```

Example

```
FUNC? (@101)
```

Returns

```
"VOLT"
```

IEEE-488.2 command

*CLS

This command clears the alarm.

Command syntax

```
*CLS
```

*IDN?

This command reads the machine manufacturer, model, serial number, and version number.

Command syntax

```
*IDN?
```

Return parameters

```
Hantek, DAQ4070A, CN1231231231231, 1.7.9
```

***RST**

This command clears the channel scan configuration.

Command syntax

*RST

***SAV**

This command quickly stores the channel configuration.

Command syntax

*SAV

Parameters

{0|1|2|3|4|5}

Example

*SAV 1

***RCL**

This command quickly uses the storage channel configuration.

Command syntax

*RCL

Parameters

{0|1|2|3|4|5}

Example

*RCL 1

INITiate command

INITiate

This command starts scanning.

Command syntax

INITiate

MEASure command

MEASure:CAPacitance?

This command scans specific channels in Cap mode and returns data.

Command syntax

```
MEASure:CAPacitance?
```

Parameters

```
[{|AUTO|MIN|MAX|DEF}, [{|MIN|MAX|DEF}, ]]
```

For example

```
MEAS:CAP? (@101)
```

MEASure:CURRent:{AC|DC}?

This command reads measurement data in DCI or ACI mode.

Command syntax

```
MEASure:CURRent:{AC|DC}?
```

Parameters

```
[{<range>|AUTO|MIN|MAX|DEF}, [{<resolution>|MIN|MAX|DEF}, ]]  
(@<ch_list>)
```

Return parameters

```
+8.54530000E+01
```

For example

```
MEAS:CURR:AC? 1, (@121)
```

MEASure:{FRESistance|RESistance}?

This command reads measurement data in OHMS or OHMS 4W mode.

Command syntax

```
MEASure:{ FRESistance| RESistance}?
```

Parameters

```
[{<range>|AUTO|MIN|MAX|DEF}, [{<resolution>|MIN|MAX|DEF}, ]]  
(@<ch_list>)
```

Return parameters

```
+8.54530000E+01
```

For example

```
MEAS:FRES? 100, (@101)
```

MEASure:{FREQuency|PERiod}?

This command reads measurement data in FREQ or PERIOD mode.

Command syntax

```
MEASure:{FREQuency|PERiod}?
```

Parameters

```
[{<range>|AUTO|MIN|MAX|DEF}, [{<resolution>|MIN|MAX|DEF}, ]]  
(@<ch_list>)
```

Return parameters

```
+8.54530000E+01
```

For example

```
MEAS:FREQ? (@101)
```

MEASure:DIODE?

This command reads measurement data in DIODE mode.

Command syntax

```
MEASure:DIODE?
```

Parameters

```
(@<ch_list>)
```

Return parameters

```
+1.32130000E-01
```

For example

```
MEAS:DIOD? (@101)
```

MEASure:TEMPerature: <probe_type>?

This command reads measurement data in TEMP mode.

Command syntax

```
MEASure:TEMPerature:<probe_type>?
```

Command syntaxParameters

```
<probe_type>:{FRTD|RTD|FThermistor|THERmistor|TCouple|DEF}.
```

Parameters

```
[{<type>|DEF}, [{<resolution>|MIN|MAX|DEF}, ]] (@<ch_list>)
```

Return parameters

```
+2.64110326E+02
```

For example

```
MEAS:TEMP:RTD? (@101)
```

MEASure:STRain:{DIRect|FDIRect}?

This command reads the measurement data of Direct 2W and 4W in Strain mode.

Command syntax

```
MEASure:STRain:{DIRect|FDIRect}?
```

Parameters

```
[{<gage_ohms>|MIN|MAX|DEF}, [{<gage_factor>|MIN|MAX|DEF},  
[{<range>|MIN|MAX|DEF}, [{<resolution>|MIN|MAX|DEF}, ]]]] (@<ch_list>)
```

Return parameters

```
+2.64110326E+02
```

For example

```
MEAS:STR:DIR? (@101)
```

MEASure:STRain:{FULL|HALF}:BENDING?

This command reads the measurement data of the full bridge and half bridge banding types in Strain mode.

Command syntax

```
MEASure:STRain:{FULL|HALF}:BENDING?
```

Parameters

```
[{<gage_factor>|MIN|MAX|DEF}, [{<range>|MIN|MAX|DEF},  
[{<resolution>|MIN|MAX|DEF}, ]]]] (@<ch_list>)
```

Return parameters

```
+2.64110326E+02
```

For example

```
MEAS:STR:FULL:BEND? (@101)
```

MEASure:STRain:{FULL|HALF}:POISSon?

This command reads the Poisson-type measurement data for the full bridge and half bridge in Strain mode.

Command syntax

```
MEASure:STRain:{FULL|HALF}:POISSon?
```

Parameters

```
[{<gage_factor>|MIN|MAX|DEF}, [{<poisson_ratio>|MIN|MAX|DEF},  
[{<range>|MIN|MAX|DEF}, [{<resolution>|MIN|MAX|DEF}, ]]]]  
(@<ch_list>)
```

Return parameters

+2.64110326E+02

For example

```
MEAS:STR:FULL:POIS? (@101)
```

MEASure:STRain:FULL:BENDING:POISSon?

This command reads the measurement data of the full bridge and half bridge BendPois types in Strain mode.

Command syntax

```
MEASure:STRain:FULL:BENDING:POISSon?
```

Parameters

```
[{<gage_factor>|MIN|MAX|DEF}, [{<poisson_ratio>|MIN|MAX|DEF},  
[{<range>|MIN|MAX|DEF}, [{<resolution>|MIN|MAX|DEF}, ]]]]  
(@<ch_list>)
```

Return parameters

+2.64110326E+02

For example

```
MEAS:STR:FULL:BEND:POIS? (@101)
```

MEASure:STRain:QUARter?

This command reads the measurement data of the quarter bridge in Strain mode.

Command syntax

```
MEASure:STRain:QUAR?
```

Parameters

```
[{<gage_factor>|MIN|MAX|DEF}, [{<range>|MIN|MAX|DEF},  
[{<resolution>|MIN|MAX|DEF}, ]]] (@<ch_list>)
```

Return parameters

+2.64110326E+02

For example

```
MEAS:STR:QUAR? (@101)
```

MEASure[:VOLTage]:{AC|DC}?

This command scans specific channels in DCV or ACV mode and returns data.

Command syntax

```
MEASure[:VOLTage]:{AC|DC}?
```

Parameters

```
[{<range>|AUTO|MIN|MAX|DEF},  
[{<resolution>|MIN|MAX|DEF},]] (@<ch_list>)
```

For example

```
MEAS:VOLT:DC? 10,0.001,(@101)
```

MMEMory subcommand

MMEMory:FORMat:READing:CSEParator

This command is used to set the interval type for logging to USB in Save Recall.

Command syntax

```
MMEMory:FORMat:READing:CSEParator
```

Parameters

```
{COMMa|SEMicolon|TAB}
```

For example

```
MMEM:FORM:READ:CSEP TAB
```

Query command

```
MMEM:FORM:READ:CSEP?
```

Return parameters

```
{COMMa|SEMicolon|TAB}
```

MMEMory:FORMat:READing:RLIMit

This command is used to set the line limit switch for logging to USB in Save Recall.

Command syntax

```
MMEMory:FORMat:READing:RLIMit
```

Parameters

```
{OFF|ON}
```

For example

```
MMEM:FORM:READ:RLIM OFF
```

Query command

```
MMEM:FORM:READ:RLIM?
```

Return parameters

```
{OFF|ON}
```

MMEMory:FORMat:READing:RLIMit:COUNT

This command is used to set the line limit for logging to USB in Save Recall.

Command syntax

```
MMEMory:FORMat:READing:RLIMit:COUNT
```

Parameters

```
{65536|1048576|MIN|MAX|DEF}
```

For example

```
MMEM:FORM:READ:RLIM:COUNT 65536
```

Query command

```
MMEM:FORM:READ:RLIM:COUNT?
```

Return parameters

```
{65536|1048576|MIN|MAX|DEF}
```

MMEMory subcommand

MMEMory:LOAD:STATe

This command is used to read the instrument status inside the machine (INT) or on a USB flash drive (USB).

Command syntax

```
MMEMory:LOAD:STATe
```

Parameters

```
<file>
```

For example

```
MMEM:LOAD:STAT "INT:\MySetup.sta"
```

MMEMory:STORe:STATe

This command is used to save the current instrument status to the machine's internal memory (INT) or a USB flash drive (USB).

Command syntax

```
MMEMory:STOR:STATe
```

Parameters

```
<file>
```

For example

```
MMEM:STOR:STAT "INT:\MySetup"
```

MMEMory:STATe:RECall:AUTO

This command is used to set whether “power on” in User Settings is Last.

Command syntax

```
MMEMory:STATe:RECall:AUTO
```

Parameters

```
{OFF|ON}
```

For example

```
MMEM:STAT:REC:AUTO ON
```

Query command

```
MMEM:STAT:REC:AUTO?
```

Return parameters

```
{OFF|ON}
```

MMEMory:STATe:RECall:AUTO?

This command is used to read whether “power on” in User Settings is set to “Last.”

Command syntax

```
MMEMory:STATe:RECall:AUTO?
```

For example

```
MMEM:STAT:REC:AUTO?
```

Return parameters

```
{OFF|ON}
```

MMEMory:STATe:VALid?

This command is used to check whether the file exists inside the machine.

Command syntax

```
MMEMory:STATe:VALid?
```

For example

```
MMEM:STAT:VAL? "INT:\MyState.sta"
```

Return parameters

```
0 or 1
```

MMEMory subcommand

MMEMory:CATalog[:ALL]?

This command is used to obtain file information from the machine's internal memory (INT) or USB flash drive (USB).

Command syntax

```
MMEMory:CATalog[:ALL]?
```

Parameters

```
[<folder>]
```

Return parameters

```
+1000000000, +327168572, "command. exe, , 375808",  
"MyDCVMeas. sta, STAT, 8192", MyData. csv, ASC, 11265"
```

For example

```
MMEM:CAT? "INT:\*.sta"
```

MMEMory:COPY

This command is used to copy files within the machine (INT) or USB flash drive (USB).

Command syntax

```
MMEMory:COPY
```

Parameters

```
<file1><file2>
```

For example

```
MMEM:COPY "INT:\MyDAQ.sta", "INT:\Backup"
```

MMEMory:DElete

This command is used to delete files from the machine's internal memory (INT) or USB flash drive (USB).

Command syntax

```
MMEMory:DElete
```

Parameters

```
<file1>
```

For example

```
MMEM:DEL "INT:\MySetup.sta"
```

MMEMory:LOG[:ENABLE]

This command is used to set whether scan data is stored on an external USB flash drive.

Command syntax

```
MMEMory:LOG[:ENABLE]
```

Parameters

```
{OFF|ON}
```

For example

```
MMEM:LOG ON
```

Query command

```
MMEM:LOG?
```

MMEMory:LOG[:ENABLE]?

This command is used to read whether the scan data is stored on an external USB flash drive.

Command syntax

```
MMEMory:LOG[:ENABLE]?
```

For example

```
MMEM:LOG?
```

Return parameters

```
0(OFF) or 1(ON)
```

MMEMory:LOG:NEW

This command is used to create a new file storage scan data on an external USB flash drive.

Command syntax

```
MMEMory:LOG:NEW
```

For example

```
MMEM:LOG:NEW
```

MMEMory:MOVE

This command is used to move or rename files inside the machine (INT).

Command syntax

```
MMEMory:MOVE
```

Parameters

<file1><file2>

For example

MMEM:MOVE "MyDAQ.sta", "INT:\Backup"

OUTPut command

OUTPut:ALARm[{1|2|3|4}]:SOURce?

This command reads the alarm status of the Alarm.

Command syntax

OUTPut:ALARm[{1|2|3|4}]:SOURce?

Return parameters

Alarm number, channel

For example

OUTP:ALAR1:SOUR?

OUTPut:ALARm:MODE?

This command reads the Mode of the alarm in home.

Command syntax

OUTPut:ALARm:MODE?

Return parameters

status

Example

LATC or TRAC

For example

OUTP:ALAR:MODE?

OUTPut:ALARm:SLOPe?

This command reads the Slop of alarm in home.

Command syntax

OUTPut:ALARm:SLOPe?

Return parameters

status

Example

NEG or POS

For example

OUTP:ALAR:SLOP?

OUTPut:TRIGger:SLOPe?

This command reads the external trigger polarity in the interval.

Command syntax

OUTPut:TRIGger:SLOPe?

Return parameters

status

Example

NEG or POS

ROUTe command

ROUTe:OPEN

This command disconnects the relay for a specific channel.

Command syntax

ROUTe:OPEN

Parameters

(@<ch_list>)

For example

ROUT:OPEN (@103,113)

ROUTe:CLOSe

This command closes the relay for a specific channel.

Command syntax

ROUTe:CLOSe

Parameters

(@<ch_list>)

For example

ROUT:CLOS (@103,113)

ROUTe:CHANnel:DElay?

This command is used to read the delay time of the channel.

Command syntax

```
ROUTe:CHANnel:DElay?
```

Parameters

```
[(@<ch_list>)]
```

For example

```
ROUT:CHAN:DEL? (@101)
```

Return parameters

```
+2.00000000E+00
```

ROUTe:CHANnel:DElay:AUTO?

This command is used to read the switch status of the channel's delay Auto mode.

Command syntax

```
ROUTe:CHANnel:DElay:AUTO?
```

Parameters

```
[(@<ch_list>)]
```

For example

```
ROUT:CHAN:DEL:AUTO? (@101,102)
```

Return parameters

```
1, 1
```

ROUTe:MONitor

This command sets the display to the current channel.

Command syntax

```
ROUTe:MONitor
```

Parameters

```
(@<channel>)
```

For example

```
ROUT:MON (@103)
```

Query command

```
ROUT:MON?
```

ROUTe:MONitor?

This command retrieves and displays the current channel.

Command syntax

```
ROUTe:MONitor?
```

For example

```
ROUT:MON?
```

Return parameters

```
#16(@103)
```

ROUTe:MONitor:STATe?

This command reads whether the remote mode display is Monitor.

Command syntax

```
ROUTe:MONitor:STATe?
```

Return parameters

```
status
```

Example

```
0(OFF) or 1(ON)
```

ROUTe:SCAN

This command sets the scan list.

Command syntax

```
ROUTe:SCAN
```

Parameters

```
(@<ch_list>)
```

Example

```
ROUTe:SCAN (@101,102)
```

ROUTe:SCAN?

This command reads the scan list.

Command syntax

```
ROUTe:SCAN?
```

For example

```
ROUTe:SCAN?
```

Return parameters

#210(@101, 102)

ROUTe:SCAN:SIZE?

This command reads the number of scan list channels.

Command syntax

```
ROUTe:SCAN:SIZE?
```

For example

```
ROUTe:SCAN:SIZE?
```

Return parameters

```
+2
```

[SENSE:]CAPacitance command

[SENSe:]CAPacitance:RANGe:AUTO

This command is used to turn on or off the Auto switch of the range configured with Cap measurement mode.

Command syntax

```
[SENSe:]CAPacitance:RANGe:AUTO
```

Parameters

```
{OFF|ON} [, (@<ch_list>)]
```

For example

```
CAP:RANG:AUTO ON, (@101)
```

Query command

```
CAP:RANG:AUTO? (@101)
```

Return parameters

```
{0|1}
```

[SENSe:]CAPacitance:RANGe

This command is used to set the range of the configured Cap measurement mode.

Command syntax

```
[SENSe:]CAPacitance:RANGe
```

Parameters

```
<range>|MIN|MAX|DEF [, (@<ch_list>)]
```

Parameter range

```
<range> = {1nF|10nF|100nF|1 μF|10 μF|100 μF}.
```

For example

```
CAP:RANG MAX, (@101)
```

Query command

```
CAP:RANG? (@101)
```

Return parameters

```
+1.00000000E-09
```

[SENSE:]CURRent subcommand

[SENSe:]CURRent[:DC]:NPLCycles

This command is used to set the PLC to the configured DCI measurement mode.

Command syntax

```
[SENSe:]CURRent[:DC]:NPLCycles
```

Parameters

```
{<PLCs>|MIN|MAX|DEF} [, (@<ch_list>)]
```

Parameter range

```
<PLCs>{0.02|0.2|1|10|100}
```

For example

```
CURR:DC:NPLC 10, (@121)
```

Query command

```
CURR:NPLC? (@121)
```

Return parameters

```
<PLCs>
```

[SENSe:]CURRent[:DC]:ZERO:AUTO

This command is used to turn on or off the Autozero switch for the configured DCI measurement mode.

Command syntax

```
[SENSe:]CURRent[:DC]:ZERO:AUTO
```

Parameters

```
{OFF|ON} [, (@<ch_list>)]
```

For example

```
CURR:DC:ZERO:AUTO ON, (@121)
```

Query command

```
CURR:ZERO:AUTO? (@121)
```

Return parameters

{0|1}

[SENSe:]CURRent:AC:BANDwidth

This command is used to set the ACI measurement mode of AC Filtar.

Command syntax

```
[SENSe:]CURRent:AC:BANDwidth
```

Parameters

```
{<freq>|MIN|MAX|DEF} [, (@<ch_list>)]
```

Parameter range

```
<freq> {3Hz|20Hz|200Hz}
```

For example

```
CURR:AC:BAND 3, (@121)
```

Query command

```
CURRent:AC:BANDwidth? (@121)
```

Return parameters

```
{<freq>|MIN|MAX|DEF}
```

[SENSE:] {FREQuency | PERiod} subcommand

[SENSe:]{FREQuency | PERiod}:APERture

This command is used to set the Gate Time for the FREQ or PERIOD measurement mode.

Command syntax

```
[SENSe:] {FREQuency|PERiod}:APERture
```

Parameters

```
{<seconds>|MIN|MAX|DEF} [, (@<ch_list>)]
```

Parameter range

```
<seconds> {1ms|10ms|100ms|1s}
```

For example

```
FREQ:APER 1, (@101)
```

Query command

```
FREQ:APER? (@101) or PER:APER? (@101)
```

Return parameters

```
{<seconds>|MIN|MAX|DEF}
```

[SENSe:]{FREQuency|PERiod}:RANGe:LOWer

This command is used to set the AC Filter for the FREQ or PERIOD measurement mode.

Command syntax

```
[SENSe:] {FREQuency|PERiod}:RANGe:LOWer
```

Parameters

```
{<freq>|MIN|MAX|DEF} [, (@<ch_list>)]
```

Parameter range

```
<freq> {3Hz|20Hz|200Hz}
```

For example

```
FREQ:RANG:LOW 3, (@101)
```

Query command

```
FFREQ:RANG:LOW? (@101) or PER:RANG:LOW? (@101)
```

Return parameters

```
{<freq>|MIN|MAX|DEF}
```

[SENSe:]{FREQuency|PERiod}:VOLTage:RANGe

This command is used to set the Range for the FREQ or PERIOD measurement mode.

Command syntax

```
[SENSe:] {FREQuency|PERiod}:VOLTage:RANGe
```

Parameters

```
{<range>|MIN|MAX|DEF} [, (@<ch_list>)]
```

Parameter range

```
<range>: {100mV|1V|10V|100V|300V}
```

For example

```
FREQ:VOLT:RANG 10, (@101)
```

Query command

```
FREQ:VOLT:RANG? (@101)
```

Return parameters

```
{<range>|MIN|MAX|DEF}
```

[SENSe:]{FREQuency|PERiod}:VOLTage:RANGe:AUTO

This command is used to set the Auto switch for the Range of the FREQ or PERIOD measurement mode.

Command syntax

[SENSe:] {FREQuency|PERiod}:VOLTage:RANGe:AUTO

Parameters

{OFF|ON} [, (@<ch_list>)]

For example

FREQ:VOLT:RANG:AUTO ON, (@101)

Query command

FREQ:VOLT:RANG:AUTO? (@101)

Return parameters

{OFF(0)|ON(1)}

[SENSE:] {RESistance|FRESistance} subcommand

[SENSe:]{RESistance|FRESistance}:NPLCycles

This command is used to set the PLC for the configured OHMS (RES) or OHMS 4W (FRES).

Command syntax

[SENSe:] {RESistance|FRESistance}:NPLCycles

Parameters

<PLCs>|MIN|MAX|DEF [, (@<ch_list>)]

Parameter range

<PLCs>{0.02|0.2|1|10|100}

For example

RES:NPLC 10, (@101) or FRES:NPLC 10, (@101)

Query command

RES:NPLC? (@101) or FRES:NPLC? (@101)

Return parameters

{<PLCs>|MIN|MAX|DEF}

[SENSe:]{RESistance|FRESistance}:OCOMpensated

This command is used to set the Offset Comp switch for the configured OHMS (RES) or OHMS 4W (FRES).

Command syntax

[SENSe:] {RESistance|FRESistance}:OCOMpensated

Parameters

{OFF|ON} [, (@<ch_list>)]

For example

RES:OCOM ON, (@101) or FRES:OCOM ON, (@101)

Query command

RES:OCOM? (@101) or FRES:OCOM? (@101)

Return parameters

{OFF(0) | ON(1)}

[SENSe:]{RESistance | FRESistance}:POWer:LIMit[:STATe]

This command is used to set the Low Power switch for the configured OHMS (RES) or OHMS 4W (FRES).

Command syntax

[SENSe:] {RESistance | FRESistance} :POWer:LIMit[:STATe]

Parameters

{OFF | ON} [, (@<ch_list>)]

For example

FRES:POW:LIM ON, (@101)

Query command

FRES:POWer:LIMit? (@101)

Return parameters

{OFF(0) | ON(1)}

[SENSe:]{RESistance | FRESistance}:RANGe

This command is used to set the range of the configured OHMS (RES) or OHMS 4W (FRES).

Command syntax

[SENSe:] {RESistance | FRESistance} :RANGe

Parameters

{<range> | MIN | MAX | DEF} [, (@<ch_list>)]

Parameter range

<range> {100 Ω | 1k Ω | 10k Ω | 100k Ω | 1M Ω | 10M Ω | 100 M Ω }

For example

RES:RANG MAX, (@101)

Query command

RES:RANGe? (@101)

Return parameters

{<range> | MIN | MAX | DEF}

[SENSe:]{RESistance|FRESistance}:RANGe:AUTO

This command is used to set the Auto switch for the Range of the configured OHMS (RES) or OHMS 4W (FRES).

Command syntax

```
[SENSe:] {RESistance|FRESistance}:RANGe:AUTO
```

Parameters

```
{OFF|ON} [, (@<ch_list>)]
```

For example

```
RES:RANG:AUTO ON, (@101) or FRES:RANG:AUTO ON, (@101)
```

Query command

```
RES:RANG:AUTO? (@101) or FRES:RANG:AUTO? (@101)
```

Return parameters

```
{OFF(0)|ON(1)}
```

[SENSe:]RESistance:ZERO:AUTO

This command is used to set the Autozero switch for the configured OHMS (RES) or OHMS 4W (FRES).

Command syntax

```
[SENSe:]RESistance:ZERO:AUTO
```

Parameters

```
{OFF|ON} [, (@<ch_list>)]
```

For example

```
RES:ZERO:AUTO ON, (@101)
```

Query command

```
RES:ZERO:AUTO? (@101)
```

Return parameters

```
{OFF(0)|ON(1)}
```

[SENSe:]STRain subcommand

[SENSe:]STRain:EXCitation

This command is used to set the Excitation Volt of the configured Strain.

Command syntax

```
[SENSe:] STRain:EXCitation
```

Parameters

{<voltage>|MIN|MAX|DEF} [, (@<ch_list>)]

Parameter range

<voltage> 1 - 12 V.

For example

STR:EXC 5, (@101)

Query command

STR:EXC? (@101)

Return parameters

voltage

[SENSe:]STRain:EXCitation:TYPE

This command is used to set the Excitation Type of the configured Strain.

Command syntax

[SENSe:] STRain:EXCitation:TYPE

Parameters

{EXTernal|FIXed} [, (@<ch_list>)]

For example

STR:EXC:TYPE EXT, (@101)

Query command

STR:EXC:TYPE? (@101)

Return parameters

EXT or FIX

[SENSe:] STRain:GFACtor

This command is used to set the Gage Factor for the configured Strain.

Command syntax

[SENSe:]STRain:GFACtor

Parameters

{<gage_factor>|MIN|MAX|DEF} [, (@<ch_list>)]

Parameter range

<gage_factor> 0.5 - 5

For example

STR:GFAC 2, (@101)

Query command

STR:GFAC? (@101)

Return parameters

gage_factor

[SENSe:] STRain:NPLCycles

This command is used to set the Gage NPLCycles for the configured Strain.

Command syntax

```
[SENSe:]STRain:NPLCycles
```

Parameters

```
{<PLCs>|MIN|MAX|DEF} [, (@<ch_list>)]
```

Parameter range

```
<PLCs> {0.002|0.02|0.2|1|10|100}
```

For example

```
STR:NPLC 10, (@101)
```

Query command

```
STR:NPLC? (@101)
```

Return parameters

```
PLCs
```

[SENSe:] STRain: OCOMpensated

This command is used to set the Gage OCOMpensated switch for the configured Strain.

Command syntax

```
[SENSe:]STRain:OCOMpensated
```

Parameters

```
{OFF|ON} [, (@<ch_list>)]
```

For example

```
STR:OCOM ON, (@101)
```

Query command

```
STR:OCOM? (@101)
```

Return parameters

```
0(OFF) or 1(ON)
```

[SENSe:]STRain:POISson

This command is used to set the Poisson's ratio of the configured strain.

Command syntax

```
[SENSe:]STRain:POISson
```

Parameters

```
{<poisson_ratio>|MIN|MAX|DEF} [, (@<ch_list>)]
```

Parameter range

<poisson_ratio> -0.9999 - +0.5

For example

STR:POIS 0.3, (@101)

Query command

STR:POIS? (@101)

Return parameters

poisson_ratio

[SENSe:]STRain:RESistance

This command is used to set the Gage Ohms of the configured Strain.

Command syntax

[SENSe:]STRain:RESistance

Parameters

{<gage_ohms>|MIN|MAX|DEF} [, (@<ch_list>)]

Parameter range

<gage_ohms> 80 - 1100 ohms

For example

STR:RES 120, (@101)

Query command

STR:RES? (@101)

Return parameters

gage_ohms

[SENSe:]STRain:UNSTrained

This command is used to set the offset of the configured strain.

Command syntax

[SENSe:]STRain:RESistance

Parameters

{<offset>|MIN|MAX|DEF} [, (@<ch_list>)]

Parameter range

<offset> -90 - +90

For example

STR:UNST 1, (@101)

Query command

STR:UNST? (@101)

Return parameters

offset

[SENSe:]STRain:VOLTage:RANGe:AUTO

This command is used to set whether the Rang of the configured Strain is Auto.

Command syntax

```
[SENSe:]STRain:VOLTage:RANGe:AUTO
```

Parameters

```
{OFF|ON} [, (@<ch_list>)]
```

For example

```
STR:VOLT:RANG:AUTO ON, (@101)
```

Query command

```
STR:VOLT:RANG:AUTO? (@101)
```

Return parameters

```
0(OFF) or 1(ON)
```

[SENSe:]STRain:VOLTage:RANGe

This command is used to set the Rang of the configured Strain.

Command syntax

```
[SENSe:]STRain:VOLTage:RANGe
```

Parameters

```
<range>|MIN|MAX|DEF [, (@<ch_list>)]
```

Parameter range

```
<range>: {100mV|1V|10V|100V|300V}
```

For example

```
STR:VOLT:RANG 0.1, (@101)
```

Query command

```
STR:VOLT:RANG? (@101)
```

Return parameters

```
range
```

[SENSe:]STRain:ZERO:AUTO

This command is used to set the Auto Zero switch for the configured Strain.

Command syntax

```
[SENSe:]STRain:ZERO:AUTO
```

Parameters

{OFF|ON} [, (@<ch_list>)]

For example

STR:ZERO:AUTO ON, (@101)

Query command

STR:ZERO:AUTO? (@101)

Return parameters

0(OFF) or 1(ON)

[SENSE:]TEMPerature subcommand

[SENSe:]TEMPerature:NPLCycles

This command is used to set the configured Temp of the PLC.

Command syntax

[SENSe:]TEMPerature:NPLCycles

Parameters

<PLCs>|MIN|MAX|DEF [, (@<ch_list>)]

Parameter range

<PLCs>{0.02|0.2|1|10|100}

For example

TEMP:NPLC 10, (@101)

Query command

TEMP:NPLC? (@101)

Return parameters

{<PLCs>|MIN|MAX|DEF}

[SENSe:]TEMPerature:RJUNction?

This command is used to read the internal temperature standard.

Command syntax

[SENSe:]TEMPerature:RJUNction?

Parameters

[, (@<ch_list>)]

For example

TEMP:RJUN?

[SENSe:]TEMPerature:TRANsducer:{RTD|FRTD}:OCOMpensated

This command is used to set the switch for the Offset Comp of the configured Temp RTD (FRTD(4w)RTD(2W)) mode.

Command syntax

```
[SENSe:]TEMPerature:TRANsducer: {RTD|FRTD}:OCOMpensated
```

Parameters

```
{OFF|ON} [, (@<ch_list>)]
```

For example

```
TEMP:TRAN:FRTD:OCOM ON, (@101)
```

Query command

```
TEMP:TRAN:RTD:OCOM? (@101) or TEMP:TRAN:FRTD:OCOM? (@101)
```

Return parameters

```
{OFF(0) | ON(1)}
```

[SENSe:]TEMPerature:TRANsducer:{RTD|FRTD}:POWER:LIMit[:STATE]

This command is used to set the switch for the Low Power mode of the configured Temp RTD (FRTD(4w)RTD(2W)).

Command syntax

```
[SENSe:]TEMPerature:TRANsducer: {RTD|FRTD}:POWER:LIMit[:STATE]
```

Parameters

```
{OFF|ON} [, (@<ch_list>)]
```

For example

```
TEMP:TRAN:FRTD:POW:LIM ON, (@101)
```

Query command

```
TEMP:TRAN:RTD:POW:LIM? (@101) or TEMP:TRAN:FRTD:POW:LIM? (@101)
```

Return parameters

```
{OFF(0) | ON(1)}
```

[SENSe:]TEMPerature:TRANsducer:{RTD|FRTD}:REFeren ce

This command is used to set the switch for the configured Temp RTD (FRTD(4w)RTD(2W)) mode Use as TCouple ref.

Command syntax

```
[SENSe:]TEMPerature:TRANsducer: {RTD|FRTD}:REFerence
```

Parameters

```
{OFF|ON} [, (@<ch_list>)]
```

For example

```
TEMP:TRAN:RTD:REF ON, (@101)
```

Query command

```
TEMP:TRANsducer:RTD:REF? (@101)orTEMP:TRANsducer:FRTD:REF? (@101)
```

Return parameters

```
{OFF(0)|ON(1)}
```

[SENSe:]TEMPerature:TRANsducer:{RTD|FRTD}:RESistan ce[:REFerence]

This command is used to set the RTD (FRTD (4w) RTD (2W)) mode R0 of the configured

Temp.

Command syntax

```
[SENSe:]TEMPerature:TRANsducer: {RTD|FRTD}:RESistance[:REFerence]
```

Parameters

```
{<resistance>|MIN|MAX|DEF} [, (@<ch_list>)]
```

Parameter range

```
<resistance> 100ohms±1%or1000ohms±1%
```

For example

```
TEMP:TRAN:RTD:RES:REF 100, (@101)
```

Query command

```
TEMP:TRAN:RTD:RES? (@101)orTEMP:TRAN:FRTD:RES? (@101)
```

Return parameters

```
{<resistance>|MIN|MAX|DEF}
```

[SENSe:]TEMPerature:TRANsducer:{FTHermistor|THERmistor}:POWer:LIMit[:STATe]

This command is used to set the switch for the Low Power mode of the configured Temp THER (FTH(2W)THER(4W)).

Command syntax

```
[SENSe:]TEMPerature:TRANsducer:{FTHermistor|THERmistor}:POWer:LIMit
[:STATe]
```

Parameters

```
{OFF|ON} [, (@<ch_list>)]
```

For example

```
TEMP:TRAN:FTH:POW:LIM ON, (@101)
```

Query command

```
TEMP:TRAN:FTH:POW:LIM? (@101) or TEMP:TRAN:THER:POW:LIM? (@101)
```

Return parameters

```
{OFF(0)|ON(1)}
```

[SENSe:]TEMPerature:TRANsducer:{FTHermistor|THERmistor}:REFerence

This command is used to set the switch for using the configured Temp's THER (FTH(2W)THER(4W)) Low Power as TCouple ref.

Command syntax

```
[SENSe:]TEMPerature:TRANsducer:{FTHermistor|THERmistor}:REFerence
```

Parameters

```
{OFF|ON} [, (@<ch_list>)]
```

For example

```
TEMP:TRAN:THER:REF ON, (@101)
```

Query command

```
TEMP:TRAN:FTH:REF? (@101) or TEMP:TRAN:THER:REF? (@101)
```

Return parameters

```
{OFF(0)|ON(1)}
```

[SENSe:]TEMPerature:TRANsducer:{FTHermistor|THERmistor}:TYPE

This command is used to set the TYPE of the configured Temp's THER (FTH(2W)THER(4W)).

Command syntax

```
[SENSe:]TEMPerature:TRANsducer:{FTHermistor|THERmistor}:TYPE
```

Parameters

```
{2252|5000|10000|MIN|MAX|DEF} [, (@<ch_list>)]
```

For example

```
TEMP:TRAN:THER:TYPE 5000, (@101)
```

Query command

```
TEMP:TRAN:FTH:TYPE? (@101) 或 TEMP:TRAN:THER:TYPE? (@101)
```

[SENSe:]TEMPerature:TRANsducer:TCouple:TYPE

This command is used to set the type of Temp TC that has been configured.

Command syntax

```
[SENSe:]TEMPerature:TRANsducer:TCouple:TYPE
```

Parameters

```
{B|E|J|K|N|R|S|T|DEF} [, (@<ch_list>)]
```

For example

```
TEMP:TRAN:TC:TYPE J, (@101)
```

Query command

```
TEMP:TRAN:TC:TYPE? (@101)
```

[SENSe:]TEMPerature:TRANsducer:TYPE

This command is used to set the mode of the configured Temp Probe.

Command syntax

```
[SENSe:]TEMPerature:TRANsducer:TYPE
```

Parameters

```
{<probe_type>} [, (@<ch_list>)]
```

For example

```
TEMP:TRAN:TYPE FRTD, (@101)
```

Query command

```
TEMP:TRAN:TYPE? (@101)
```

Return parameters

<probe_type>

[SENSe:]TEMPerature:ZERO:AUTO

This command is used to set the Autozero switch for the configured Temp.

Command syntax

```
[SENSe:]TEMPerature:ZERO:AUTO
```

Parameters

```
{OFF|ON} [, (@<ch_list>)]
```

For example

```
TEMP:ZERO:AUTO ON, (@101)
```

Query command

```
TEMP:ZERO:AUTO? (@101)
```

Return parameters

```
{OFF(0) | ON(1)}
```

[SENSe:]VOLTage subcommand

[SENSe:]VOLTage:AC:BANDwidth

This command is used to set the AC Filter for the configured ACV.

Command syntax

```
[SENSe:]VOLTage:AC:BANDwidth
```

Parameters

```
<freq>|MIN|MAX|DEF [, (@<ch_list>)]
```

Parameter range

```
<freq>AC: {3Hz|20Hz|200Hz}
```

For example

```
VOLT:AC:BAND 3, (@101)
```

Query command

```
VOLT:AC:BAND? (@101)
```

Return parameters

```
{<freq>|MIN|MAX|DEF}
```

[SENSe:]VOLTage:{AC|DC}:RANGe:AUTO

This command is used to set the Auto switch for the Range of the configured ACV or DCV.

Command syntax

```
[SENSe:]VOLTage: {AC|DC}:RANGe:AUTO
```

Parameters

```
{OFF|ON} [, (@<ch_list>)]
```

For example

```
VOLT:AC:RANG:AUTO ON, (@101)
```

Query command

```
VOLT:AC:RANG:AUTO? (@101) or VOLT:DC:RANG:AUTO? (@101)
```

Return parameters

```
{OFF(0) | ON(1)}
```

[SENSe:]VOLTage:{AC|DC}:RANGe

This command is used to set the range of the configured ACV or DCV.

Command syntax

```
[SENSe:]VOLTage: {AC|DC}:RANGe
```

Parameters

```
<range>|MIN|MAX|DEF [, (@<ch_list>)]
```

Parameter range

```
<range> {100mV | 1V | 10V | 100V | 300V}
```

For example

```
VOLT:AC:RANG 10, (@101)
```

Query command

```
VOLT:AC:RANG? (@101) or VOLT:DC:RANG? (@101)
```

Return parameters

```
<range>
```

[SENSe:]VOLTage[:DC]:IMPedance:AUTO

This command is used to set the switch for Input Z of the configured DCV.

Command syntax

```
[SENSe:]VOLTage[:DC]:IMPedance:AUTO
```

Parameters

```
{OFF|ON} [, (@<ch_list>)]
```

For example

```
VOLT:IMP:AUTO OFF, (@101)
```

Query command

```
VOLT:IMP:AUTO? (@101)
```

Return parameters

```
{OFF(0) | ON(1)}
```

[SENSe:]VOLTage[:DC]:NPLCycles

This command is used to set the PLC of the configured DCV.

Command syntax

```
[SENSe:]VOLTage[:DC]:NPLCycles
```

Parameters

```
<PLCs>|MIN|MAX|DEF [, (@<ch_list>)]
```

Parameter range

```
<PLCs> {0.002|0.02|0.2|1|10|100}
```

For example

```
VOLT:DC:NPLC 10, (@101)
```

Query command

```
VOLT:NPLC? (@101)
```

Return parameters

```
<PLCs>|MIN|MAX|DEF
```

[SENSe:]VOLTage[:DC]:ZERO:AUTO

This command is used to set the Autozero switch of the configured DCV.

Command syntax

```
[SENSe:]VOLTage[:DC]:ZERO:AUTO
```

Parameters

```
{OFF|ON} [, (@<ch_list>)]
```

For example

```
VOLT:DC:ZERO:AUTO ON, (@101)
```

Query command

```
VOLT:ZERO:AUTO? (@101)
```

Return parameters

```
{OFF(0) | ON(1)}
```

SYSTEM subcommand

SYSTEM:BEEPer[:IMMediate]

This command tests whether the buzzer is working properly.

Command syntax

```
SYSTEM:BEEPer[:IMMediate]
```

For example

```
SYST:BEEP
```

SYSTEM:BEEPer:STATe

This command sets the buzzer switch.

Command syntax

```
SYSTEM:BEEPer:STATe
```

Parameters

```
{OFF|ON}
```

For example

```
SYST:BEEP:STAT OFF
```

Query command

```
SYST:BEEP:STAT?
```

Return parameters

```
{OFF(0)|ON(1)}
```

SYSTEM:CLICk:STATe

This command sets the key tone switch.

Command syntax

```
SYSTEM:CLICk:STATe
```

Parameters

```
{OFF|ON}
```

For example

```
SYST:CLIC:STATE OFF
```

Query command

```
SYST:CLIC:STAT?
```

Return parameters

```
{OFF(0)|ON(1)}
```

SYSTEM:CTYPE?

This command reads the machine board information.

Command syntax

```
SYSTEM:CTYPE?
```

Return parameters

```
<slot>
```

For example

```
SYST:CTYP? 1
```

Example

```
DAQM4070A, 0, 00.76
```

SYSTEM:CPON

This command resets the board.

Command syntax

```
SYSTEM:CPON
```

Parameters

```
{<slot>|ALL}
```

For example

```
SYST:CPON 2
```

SYSTEM:DATE

This command sets the machine's date and time.

Command syntax

```
SYSTEM:DATE
```

Parameters

```
<year>, <month>, <day>
```

For example

```
SYST:DATE 2024, 1, 1
```

Query command

```
SYST:DATE?
```

Return parameters

```
<year>, <month>, <day>
```

SYSTem:ERRor?

This command reads the error list of the machine SCPI.

Command syntax

```
SYSTem:ERRor?
```

Return parameters

```
<error code>,<error string>
```

For example

```
SYST:ERR?
```

Example

```
<error code>,<error string>
```

SYSTem:LOCal

This command is used to cancel remote mode.

Command syntax

```
SYSTem:LOCal
```

For example

```
SYST:LOC
```

SYSTem:PERSONa[:MANUFACTURer]

This command sets the machine compatibility.

Command syntax

```
SYSTem:PERSONa[:MANUFACTURer]
```

Parameters

```
"Hantek" , "Keysight Technologies"
```

For example

```
SYST:PERS "Hantek"
```

Query command

```
SYST:PERSON?
```

Return parameters

```
{"<string>"}
```

SYSTem:PERSONa[:MANUFACTURer]:DEFault

This command sets the machine's default manufacturer.

Command syntax

SYSTem:PERSonA[:MANuFacterer]:DEFault

For example

SYST:PERS:DEF

Query command

SYST:PERS:DEF?

Return parameters

{"<string>"}

SYSTem:PERSonA:MODEl

This command sets the compatible machine model.

Command syntax

SYSTem:PERSonA:MODEl

Parameters

{"DAQ4070A" , "34970A" , "34972A"}

For example

SYST:PERS:MOD "DAQ4070A"

Query command

SYST:PERS:MOD?

Return parameters

{"DAQ4070A"}

SYSTem:PERSonA:MODEl:DEFault

This command sets the compatible machine model.

Command syntax

SYSTem:PERSonA:MODEl:DEFault

For example

SYST:PERS:MOD:DEF

Query command

SYST:PERS:MOD:DEF?

Return parameters

{"<string>"}

SYSTem:TIME

This command sets the machine's hours, minutes, and seconds.

Command syntax

SYSTem:TIME

Parameters

<hour>, <minute>, <second>

For example

SYST:TIME 15, 30, 23.000

Query command

SYSTem:TIME?

Return parameters

<hour>, <minute>, <second>

SYSTem:TIME:SCAN?

This command reads the measurement time of the machine.

Command syntax

SYSTem:TIME:SCAN?

For example

SYST:TIME:SCAN?

SYSTem:VERSion?

This command reads the machine's date of manufacture.

Command syntax

SYSTem:VERSion?

For example

SYST:VERS?

SYSTem:WMESsage

Command syntax

SYSTem:WMESsage

Parameters

"<string>"

For example

SYST:WMES "Hantek DAQ4070A Data Acquisition System"

Query command

SYSTem:WMESsage?

Return parameters

"<string>"

SYSTEM:REMOte

This command is used to enter remote mode.

Command syntax

```
SYSTEM:REMOte
```

For example

```
SYST:REM
```

SYSTEM subcommand

SYSTEM:COMMunicate:LAN:DHCP

This command is used to set whether the LAN is DHCP or Manual.

Command syntax

```
SYSTEM:COMMunicate:LAN: DHCP
```

Parameters

```
{OFF|ON}
```

For example

```
SYST:COMM:LAN:DHCP OFF
```

Query command

```
SYST:COMM:LAN:DHCP?
```

Return parameters

```
{OFF|ON}
```

SYSTEM:COMMunicate:LAN:IPADdress

This command is used to set the IP address.

Command syntax

```
SYSTEM:COMMunicate:LAN:IPADdress
```

Parameters

```
"<address>"
```

For example

```
SYST:COMM:LAN:IPADdress "192.168.1.126"
```

```
SYST:COMM:LAN:UPD
```

Query command

```
SYST:COMM:LAN:IPAD?
```

Return parameters

"<address>"

SYSTem:COMMunicate:LAN:SMASK

This command is used to set the subnet mask.

Command syntax

```
SYSTem:COMMunicate:LAN:SMASK
```

Parameters

```
"<mask>"
```

For example

```
SYST:COMM:LAN:SMAS "255.255.254.0"
```

```
SYST:COMM:LAN:UPD
```

Query command

```
SYST:COMM:LAN:SMAS?
```

Return parameters

```
"<mask>"
```

SYSTem:COMMunicate:LAN:UPD

This command is used to execute network settings.

Command syntax

```
SYSTem:COMMunicate:LAN:UPDate
```

For example

```
SYST:COMM:LAN:UPD
```

TRIGger command

TRIGger:COUNT

This command is used to set the number of samples in SCAN mode.

Command syntax

```
TRIGger:COUNT
```

Parameters

```
{<count>|MIN|MAX|DEF|INF}
```

For example

```
TRIG:COUN 10
```

Query command

```
TRIG:COUN?
```

Return parameters

{<count>|MIN|MAX|DEF|INF}

TRIGger:COUNT?

This command is used to set the number of samples in SCAN mode.

Command syntax

TRIGger:COUNT?

Return parameters

Sample size

Example

+1.00000000E+00

TRIGger:SLOPe?

This command reads the polarity of the external trigger signal of the interval.

Command syntax

TRIGger:SLOPe?

Return parameters

Polarity of external trigger signal

Example

POS or NEG

TRIGger:SOURe

This command is used to set the interval mode.

Command syntax

TRIGger:SOURce

Parameters

{IMMEDIATE|EXTERNAL|INTERNAL|BUS|TIMER|ALARM[{1|2|3|4}]}

For example

TRIG:SOUR BUS

Query command

TRIG:SOUR?

Return parameters

{IMMEDIATE|EXTERNAL|INTERNAL|BUS|TIMER|ALARM[{1|2|3|4}]}

TRIGger:SOURce?

This command reads the pattern of the interval.

Command syntax

```
TRIGger:SOURce?
```

Return parameters

Mode

Example

```
"IMM", "EXT", "INT", "BUS", "TIM", "ALAR1", "ALAR2", "ALAR3",  
or"ALAR4".
```

TRIGger:TIMer

This command is used to set the TIME mode time for the interval.

Command syntax

```
TRIGger:TIMer
```

Parameters

```
{<seconds>|MIN|MAX|DEF}
```

For example

```
TRIG:TIM 30E-03
```

Query command

```
TRIG:TIM?
```

Return parameters

```
{<seconds>|MIN|MAX|DEF}
```

TRIGger:TIMer?

This command reads the scan delay time in Time mode for the Scan mode interval.

Command syntax

```
TRIGger:TIMer?
```

Return parameters

Time

Example

```
+1.00000000E+01
```