

ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Find the *Copper Mountain Planar TR1300/1* at our website: *Click HERE*

COPPER MOUNTAIN
TECHNOLOGIES

TR Series

Planar TR1300/1,
TR5048, TR7530

Vector Network Analyzer

Programming Manual SCPI



Software version: 19.3.0
August 2019

TABLE OF CONTENTS

1 Introduction 11

1.1 Programming Manual Scope 11
1.2 Related Documents 11
1.3 References 11

2 Analyzer Interfaces 12

2.1 Ethernet Interface 12

3 SCPI Overview 13

3.1 Command Tree 13
3.2 Subsystems 14
3.3 Optional Subsystems 14
3.4 Long and Short Formats 15
3.5 Case Sensitivity 15
3.6 Parameters 15
3.6.1 Numeric Values 15
3.6.1.1 Multiplier Prefixes 16
3.6.1.2 Notations 16
3.6.2 Booleans 17
3.6.3 Character Data 17
3.6.4 String Parameters 17
3.6.5 Numeric Lists 17
3.7 Query Commands 18
3.8 Numeric Suffixes 18
3.9 Compound Commands 18
3.10 IEEE488.2 Common Commands Overview 19
3.11 String Terminator 19

4 Remote Control Commands 20

4.1 Conventions 20
4.1.1 Syntax 20
4.1.2 Identifiers 21
4.2 IEEE488.2 Common Commands 22
*CLS 22
*ESE 23
*ESR? 23
*IDN? 24
*OPC 24
*OPC? 25
*RST 25
*SRE 26
*STB? 26
*TRG 27

*WAI 27
4.3 Network Analyzer Commands 28
ABOR 28
CALC:CONV 28
CALC:CONV:FUNC 29
CALC:CORR:EDEL:TIME 30
CALC:CORR:OFFS:PHAS 31
CALC:DATA:FDAT? 32
CALC:DATA:FMEM? 33
CALC:DATA:SDAT? 34
CALC:DATA:SMEM? 35
CALC:FILT:TIME 36
CALC:FILT:TIME:CENT 37
CALC:FILT:TIME:SHAP 38
CALC:FILT:TIME:SPAN 39
CALC:FILT:TIME:STAR 40
CALC:FILT:TIME:STAT 41
CALC:FILT:TIME:STOP 42
CALC:FORM 43
CALC:FSIM:SEND:DEEM:PORT:STAT 44
CALC:FSIM:SEND:DEEM:PORT:USER:FIL 45
CALC:FSIM:SEND:PMC:PORT:STAT 46
CALC:FSIM:SEND:PMC:PORT:USER:FIL 47
CALC:FSIM:SEND:ZCON:PORT:Z0 48
CALC:FSIM:SEND:ZCON:STAT 49
CALC:FUNC:DATA? 50
CALC:FUNC:DOM 51
CALC:FUNC:DOM:COUP 52
CALC:FUNC:DOM:STAR 53
CALC:FUNC:DOM:STOP 54
CALC:FUNC:EXEC 55
CALC:FUNC:PEXC 56
CALC:FUNC:POIN? 57
CALC:FUNC:PPOL 58
CALC:FUNC:TARG 59
CALC:FUNC:TTR 60
CALC:FUNC:TYPE 61
CALC:LIM 62
CALC:LIM:DATA 63
CALC:LIM:DISP 64
CALC:LIM:FAIL? 64
CALC:LIM:OFFS:AMPL 65
CALC:LIM:OFFS:STIM 66
CALC:LIM:REP:ALL? 67

CALC:LIM:REP:POIN? 68
CALC:LIM:REP? 69
CALC:MARK 70
CALC:MARK:ACT 71
CALC:MARK:BWID 71
CALC:MARK:BWID:DATA? 72
CALC:MARK:BWID:REF 73
CALC:MARK:BWID:THR 74
CALC:MARK:BWID:TYPE 75
CALC:MARK:COUN 76
CALC:MARK:COUP 76
CALC:MARK:FUNC:DOM 77
CALC:MARK:FUNC:DOM:STAR 78
CALC:MARK:FUNC:DOM:STOP 79
CALC:MARK:FUNC:EXEC 80
CALC:MARK:FUNC:PEXC 81
CALC:MARK:FUNC:PPOL 82
CALC:MARK:FUNC:TARG 83
CALC:MARK:FUNC:TRAC 84
CALC:MARK:FUNC:TTR 85
CALC:MARK:FUNC:TYPE 86
CALC:MARK:REF 87
CALC:MARK:SET 88
CALC:MARK:X 89
CALC:MARK:Y? 90
CALC:MATH:DEL 90
CALC:MATH:FUNC 91
CALC:MATH:MEM 92
CALC:MST 92
CALC:MST:DATA? 93
CALC:MST:DOM 94
CALC:MST:DOM:STAR 95
CALC:MST:DOM:STOP 96
CALC:PAR:COUN 97
CALC:PAR:DEF 98
CALC:PAR:SEL 99
CALC:PAR:SPOR 100
CALC:RLIM 101
CALC:RLIM:DATA 102
CALC:RLIM:DISP:LINE 103
CALC:RLIM:FAIL? 103
CALC:RLIM:REP? 104
CALC:SMO 105
CALC:SMO:APER 106

CALC:TRAC:DATA:FDAT? 107
CALC:TRAC:DATA:FMEM? 108
CALC:TRAC:DATA:SDAT? 109
CALC:TRAC:DATA:SMEM? 110
CALC:TRAN:TIME 111
CALC:TRAN:TIME:CENT 112
CALC:TRAN:TIME:IMP:WIDT 113
CALC:TRAN:TIME:KBES 114
CALC:TRAN:TIME:LPFR 114
CALC:TRAN:TIME:REFL:TYPE 115
CALC:TRAN:TIME:SPAN 116
CALC:TRAN:TIME:STAR 117
CALC:TRAN:TIME:STAT 118
CALC:TRAN:TIME:STEP:RTIM 119
CALC:TRAN:TIME:STIM 120
CALC:TRAN:TIME:STOP 121
CALC:TRAN:TIME:UNIT 122
DISP:COL:BACK 123
DISP:COL:GRAT 124
DISP:COL:RES 124
DISP:COL:TRAC:DATA 125
DISP:COL:TRAC:MEM 126
DISP:ENAB 126
DISP:FSIG 127
DISP:IMAG 127
SYST:HIDE 128
SYSTem:SHOW 128
DISP:MAX 129
DISP:SPL 129
DISP:UPD 130
DISP:WIND:ACT 131
DISP:WIND:ANN:MARK:ALIG 132
DISP:WIND:ANN:MARK:SING 133
DISP:WIND:MAX 133
DISP:WIND:SPL 134
DISP:WIND:TITL 135
DISP:WIND:TITL:DATA 135
DISP:WIND:TRAC:ANN:MARK:MEM 136
DISP:WIND:TRAC:ANN:MARK:POS:X 137
DISP:WIND:TRAC:ANN:MARK:POS:Y 138
DISP:WIND:TRAC:Y:AUTO 138
DISP:WIND:TRAC:Y:PDIV 139
DISP:WIND:TRAC:Y:RLEV 140
DISP:WIND:TRAC:Y:RPOS 141

DISP:WIND:Y:DIV 142
FORM:BORD 143
FORM:DATA 144
HCOP 145
HCOP:ABOR 145
HCOP:DATE:STAM 145
HCOP:IMAG 146
HCOP:PAIN 146
INIT 147
INIT:CONT 148
MMEM:COPY 148
MMEM:DEL 149
MMEM:LOAD 149
MMEM:LOAD:CHAN 150
MMEM:LOAD:CKIT 150
MMEM:LOAD:LIM 151
MMEM:LOAD:RLIM 152
MMEM:LOAD:SEGM 152
MMEM:MDIR 153
MMEM:STOR 153
MMEM:STOR:CHAN 154
MMEM:STOR:CHAN:CLE 154
MMEM:STOR:CKIT 155
MMEM:STOR:FDAT 155
MMEM:STOR:IMAG 156
MMEM:STOR:LIM 156
MMEM:STOR:RLIM 157
MMEM:STOR:SEGM 157
MMEM:STOR:SNP 158
MMEM:STOR:SNP:FORM 159
MMEM:STOR:SNP:TYPE:S1P 159
MMEM:STOR:SNP:TYPE:S2P 160
MMEM:STOR:STYP 161
MMEM:TRAN? 162
OUTP 162
SENS:AVER 163
SENS:AVER:CLE 163
SENS:AVER:COUN 164
SENS:BAND 165
SENS:BWID 165
SENS:CORR:CLE 166
SENS:CORR:COEF 167
SENS:CORR:COEF:METH:ERES 168
SENS:CORR:COEF:METH:OPEN 169

SENS:CORR:COEF:METH:SHOR 170
SENS:CORR:COEF:METH:SOLT1 171
SENS:CORR:COEF:METH:THRU 172
SENS:CORR:COEF:SAVE 173
SENS:CORR:COLL:CKIT 174
SENS:CORR:COLL:CKIT:LAB 175
SENS:CORR:COLL:CKIT:RES 176
SENS:CORR:COLL:CKIT:STAN:C0 176
SENS:CORR:COLL:CKIT:STAN:C1 177
SENS:CORR:COLL:CKIT:STAN:C2 178
SENS:CORR:COLL:CKIT:STAN:C3 179
SENS:CORR:COLL:CKIT:STAN:DEL 180
SENS:CORR:COLL:CKIT:STAN:L0 181
SENS:CORR:COLL:CKIT:STAN:L1 182
SENS:CORR:COLL:CKIT:STAN:L2 183
SENS:CORR:COLL:CKIT:STAN:L3 184
SENS:CORR:COLL:CKIT:STAN:LAB 185
SENS:CORR:COLL:CKIT:STAN:LOSS 186
SENS:CORR:COLL:CKIT:STAN:TYPE 187
SENS:CORR:COLL:CKIT:STAN:Z0 188
SENS:CORR:COLL:CLE 188
SENS:CORR:COLL:ECAL:CHECK:EXEC 189
SENS:CORR:COLL:ECAL:ORI:EXEC 189
SENS:CORR:COLL:ECAL:ORI:STAT 190
SENS:CORR:COLL:ECAL:PATH 191
SENS:CORR:COLL:ECAL:SOLT1 192
SENS:CORR:COLL:ECAL:SOLT2 192
SENS:CORR:COLL:ECAL:UCH 193
SENS:CORR:COLL:LOAD 194
SENS:CORR:COLL:METH:ERES 195
SENS:CORR:COLL:METH:OPEN 196
SENS:CORR:COLL:METH: SHOR 197
SENS:CORR:COLL:METH:SOLT1 198
SENS:CORR:COLL:METH:THRU 199
SENS:CORR:COLL:METH:TYPE? 200
SENS:CORR:COLL:OPEN 201
SENS:CORR:COLL:SAVE 202
SENS:CORR:COLL:SHOR 203
SENS:CORR:COLL:THRU 204
SENS:CORR:EXT 205
SENS:CORR:EXT:PORT:FREQ 206
SENS:CORR:EXT:PORT:INCL 207
SENS:CORR:EXT:PORT:LDC 208
SENS:CORR:EXT:PORT:LOSS 209

SENS:CORR:EXT:PORT:TIME 210
SENS:CORR:IMP 211
SENS:CORR:STAT 212
SENS:CORR:TRAN:TIME:FREQ 213
SENS:CORR:TRAN:TIME:LOSS 214
SENS:CORR:TRAN:TIME:RVEL 215
SENS:CORR:TRAN:TIME:STAT 216
SENS:CORR:TYPE? 217
SENS:FREQ 218
SENS:FREQ:DATA? 219
SENS:FREQ:CENT 220
SENS:FREQ:SPAN 221
SENS:FREQ:STAR 222
SENS:FREQ: STOP 223
SENS:OFFS 224
SENS:OFFS:ADJ 224
SENS:OFFS:ADJ:CONT 225
SENS:OFFS:ADJ:CONT:PER 226
SENS:OFFS:ADJ:EXEC 227
SENS:OFFS:ADJ:PORT 228
SENS:OFFS:ADJ:VAL 229
SENS:OFFS:PORT:DATA? 230
SENS:OFFS:PORT:DIV 231
SENS:OFFS:PORT:MULT 232
SENS:OFFS:PORT:OFFS 233
SENS:OFFS:PORT:STAR 234
SENS:OFFS:PORT:STOP 235
SENS:ROSC:SOUR 236
SENS:SEGM:DATA 237
SENS:SWE:POIN 238
SENS:SWE:POIN:TIME 239
SENS:SWE:TYPE 240
SENS:VVM:TYPE 241
SENS:VVM:FORM 242
SENS:VVM:FREQ 243
SENS:VVM:DATA? 244
SENS:VVM:REF:CLE 244
SENS:VVM:REF:MEM 245
SENS:VVM:REF:DATA? 245
SENS:VVM:TABL:CLE 246
SENS:VVM:TABL:MEM 246
SENS:VVM:TABL:INS 247
SENS:VVM:TABL:REM 247
SENS:VVM:TABL:SAVE 248

SENS:VVM:TABL:DATA? 249
SERV:CHAN:ACT? 250
SERV:CHAN:COUN? 250
SERV:CHAN:TRAC:ACT? 250
SERV:CHAN:TRAC:COUN? 251
SERV:PORT:COUN? 251
SERV:SWE:FREQ:MAX? 251
SERV:SWE:FREQ:MIN? 252
SERV:SWE:POIN? 252
SOUR:POW 253
SOUR:POW:CENT 254
SOUR:POW:SLOP 255
SOUR:POW:SPAN 256
SOUR:POW:STAR 257
SOUR:POW:STOP 258
STAT:OPER? 259
STAT:OPER:COND? 259
STAT:OPER:ENAB 260
STAT:OPER:NTR 261
STAT:OPER:PTR 262
STAT:PRES 262
STAT:QUES:COND? 263
STAT:QUES:ENAB 263
STAT:QUES:LIM:CHAN:COND? 264
STAT:QUES:LIM:CHAN:ENAB 264
STAT:QUES:LIM:CHAN:NTR 265
STAT:QUES:LIM:CHAN:PTR 266
STAT:QUES:LIM:CHAN? 267
STAT:QUES:LIM:COND? 267
STAT:QUES:LIM:ENAB 268
STAT:QUES:LIM:NTR 269
STAT:QUES:LIM:PTR 270
STAT:QUES:LIM? 270
STAT:QUES:NTR 271
STAT:QUES:PTR 272
STAT:QUES:RLIM:CHAN:COND? 273
STAT:QUES:RLIM:CHAN:ENAB 273
STAT:QUES:RLIM:CHAN:NTR 274
STAT:QUES:RLIM:CHAN:PTR 275
STAT:QUES:RLIM:CHAN? 275
STAT:QUES:RLIM:COND? 276
STAT:QUES:RLIM:ENAB 276
STAT:QUES:RLIM:NTR 277
STAT:QUES:RLIM:PTR 278

STAT:QUES:RLIM? 278
STAT:QUES? 279
SYST:COMM:ECAL:TEMP:SENS? 279
SYST:CORR 280
SYST:DATE 280
SYST:ERR? 281
SYST:LOC 281
SYST:PRES 282
SYST: READy? 282
SYST:REM 283
SYST:RWL 283
SYST:TEMP:SENS? 284
SYST:TIME 285
SYST:TERM 285
TRIG 286
TRIG:SING 287
TRIG:SOUR 288
TRIG:STAT? 289
TRIG:WAIT 290
VVM:RES 291
VVM:FONT 291

5 Programming Tips 292

5.1 Program Sweep Initiation and Waiting 292
5.2 External Trigger 293
5.3 Waiting for Calibration Commands 294
5.4 VISA Timeout Considerations 294
5.5 Receiving Data Arrays in Text Form 294
5.6 Receiving Data Arrays in Binary Form 296

Appendix 1. IEE488.2 Status Reporting Syste 297

Appendix 2. Error Codes 303

Appendix 3. Programming Examples 304

1 Introduction

This Manual contains information about the CMT Network Analyzer remote control and its data communication, carried out by means of user program through a computer network.

This manual describes the analyzer command system and the network protocols

used to deliver commands. The commands sent to the analyzer and the responses from it are text messages conforming to the **SCPI** (Standard Commands for Programmable Instruments) specification. The text messages are delivered over computer networks using TCP/IP Socket network protocols.

TCP/IP Socket is a general-purpose network protocol. The user program can connect to the analyzer using the TCP/IP Socket protocol both directly and through the VISA library.

VISA (Virtual Instrument Software Architecture) is a widely used software inputoutput interface in the field of testing and measurement for controlling devices from a personal computer. It is a library of functions for C / C ++, C #, Visual Basic, MATLAB, LabVIEW and others. The VISA library unifies access to all measuring instruments, regardless of the protocol and equipment used. The VISA library is installed on the client side, that is, on the computer where the user program is executed. The VISA library is available on the websites of many companies for free.

1.1 Programming Manual Scope

This Manual directly applies to the Planar – TR1300/1, TR5048 and TR7530 models Network Analyzer.

1.2 Related Documents

Before reading this Manual, familiarize yourself with Analyzer Operating Manual.

1.3 References

IEEE Standard 488.2–1992, IEEE Standard Codes, Formats, Protocols and Common Commands for Use with ANSI/IEEE Std 488.1–1987. IEEE, New York, NY, 1992.

SCPI Standard–1999, Standard Commands for Programmable Instruments Volume 1: Syntax and Style. SCPI Consortium, San Diego, CA, 1999.

VISA specifications, *http://www.ivifoundation.org/specifications*.

2 Analyzer Interfaces

2.1 Ethernet Interface

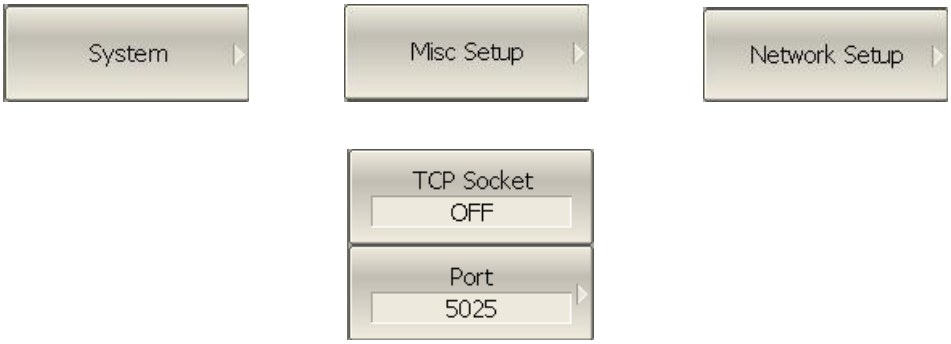
Ethernet interface is part of the equipment of a personal computer that connects to the device "Planar - TR1300/1», «TR5048» or «TR7530».

Data transfer between the PC user and the computer that is connected to the device, is performed via Socket protocol (TCP, port 5025).

Connect the device to a PC in the local Ethernet network of your company.

Activate the function of remote control via Socket protocol on your Analyzer as described below.

To enable/disable remote control via network Socket protocol, use the following softkeys: **System > Misc Setup > Network Setup > TCP Socket ON/OFF.**



---

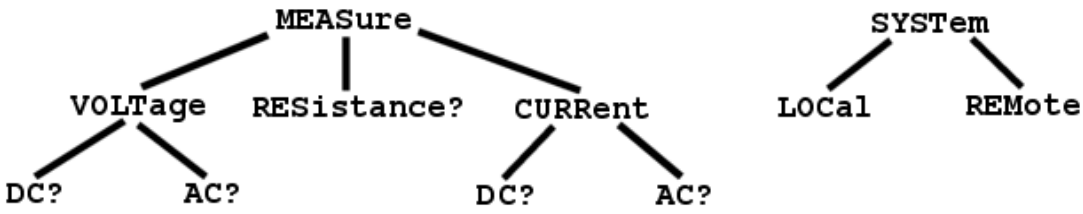
3 SCPI Overview

The Ethernet interface implement same set of commands based on 1999 SCPI standard (Standard Commands for Programmable Instruments). This command set allows the exchange of character messages.

SCPI is developed by SCPI Consortium (http:\\www.scpiconsortium.org).The main features of SCPI standard are described below. For a more detailed information on SCPI standard, see the website of SCPI Consortium.

3.1 Command Tree

The SCPI commands are organized in a tree structure. For example:



Each tree structure forms a functional system. The base of the tree is called root, e.g. MEASure and SYSTem. Each functional system can have subsystems of lower level. The final nodes are called leaves. The entire sequence from root to the leaf makes up the command. For example, part of SOURCe functional system looks as follows:

```
:SOURce
      :POWer
            :CENTer
            :STARt
            :SPAN
            :STOP
            [:LEVel]
                  :SLOPe
                        [:DATA]
                        :STATe
```

This SOURce branch has several levels, where CENTer, STARt, SPAN, STOP, DATA, STATe are the leaves, which represent the following six commands:

```
:SOURce:POWer:CENTer
:SOURce:POWer:STARt
:SOURce:POWer:SPAN
:SOURce:POWer:STOP
:SOURce:POWer[:LEVel]:SLOPe[:DATA]
:SOURce:POWer[:LEVel]:SLOPe:STATe
```

The tree can contain subsystems and leaves with the same names if they belong to different branches, e.g. CENTer leaf is on the tips of different branches:

```
:SOURce                    :SENSe
     :POWer                    :FREQuency
          :CENTer                   :CENTer
```

3.2 Subsystems

A colon (':') separates the subsystems. The subsystems, which follow the colon are of a lower level. For example, in command:

```
:SOURce:POWer:STARt
```

the start power STARt is a part of POWer subsystem, which is a part of SOURce subsystem. The stop power is also a part of :SOURce:POWer subsystem. It is specified by:

```
:SOURce:POWer:STOP
```

The first colon in the line can be omitted, for example:

```
SOURce:POWer:STOP
```

3.3 Optional Subsystems

Some subsystems can be specified as optional, if omission of such a subsystem will not lead to ambiguity. This means that the subsystem can be omitted in command line. The optional subsystems are bracketed ("[]"). For example, if full command specification is written as:

```
SOURce:POWer[:LEVel]:SLOPe[:DATA]
```

subsystems LEVel and DATA are optional. Therefore the both commands are valid:

```
SOURce:POWer:LEVel:SLOPe:DATA
SOURce:POWer:SLOPe
```

3.4 Long and Short Formats

Each keyword in a command specification has a long format and a short format. The short format of a command is indicated by capital letters. For example, a command specification:

```
SENSe:FREQuency:CENTer
```

can be written as:

```
SENS:FREQ:CENT
SENS:FREQ:CENTer
```

Only long or short form of each keyword is acceptable. For example, the following specification is incorrect:

```
:SENS:FREQuen:CEN
```

3.5 Case Sensitivity

The commands are not case sensitive. Upper case and lower case letters are only used to indicate the long and short formats of a command specification. For example, the following commands are equivalent:

```
SENS:FREQ:STAR
sens:freq:star
```

3.6 Parameters

The commands can have parameters. The parameters are separated from the command by a space. If a command has several parameters, they are separated by commas (',').

3.6.1 Numeric Values

The numeric values are integer or real numbers. These parameters can have measurement units. For example:

```
SENS:FREQ 1000000000
SENS:FREQ 1000 MHz
SENS:FREQ 1 GHz
SENS:FREQ 1E9
```

3.6.1.1 Multiplier Prefixes

The SCPI standard allows specification of the numeric values with multiplier prefix to the measurement units.

Prefix	Multiplier
A	1e-18
F	1e-15
P	1e-12
N	1e-9
U	18-6
M	1e-3
K	1e3
MA	1e6
G	1e9
T	1e12
PE	1e15
EX	1e18

There are two exceptions to the above designation: prefix M in combination with HZ or OHM means 1e6 (Mega), and not 1e−3 (milli), i.e. MHZ means Megahertz, same as MAHZ.

3.6.1.2 Notations

The SCPI standard allows numeric value specification in different notations. Decimal notation is used by default. To use other notations, specify the numeric values in the following way:

Notation	Prefix	Example
Binary	**#B**	#B11001010 = 202_{10}
Octal	**#Q**	#Q107 = 71_{10}
Hexadecimal	**#H**	#H10FF = 4351_{10}

3.6.2 Booleans

The booleans can assume two values: logical *yes* and logical *no* (ON and OFF), and specified in command as:

`ON` or `1` – logical yes
`OFF` or `0` – logical no

For example:

```
DISPlay:ENABle OFF
DISPlay:ENABle 0
```

3.6.3 Character Data

The SCPI standard allows specification of parameters as character data, as in the following command:

```
TRIGger:SOURce {BUS|IMMediate|EXTernal}
```

the possible values of the character data – "BUS", "IMMediate", "EXTernal".

The character data have long and short format, and the formats are specified in accordance with the same rules as described in Section 3.4.

Apart from that, the character data can be combined with numerical parameters. For example:

```
SENSe:FREQuency:STARt {MINimum|MAXimum|<value>}
```

The following specifications are acceptable:

```
SENSe:FREQuency:STARt MIN
SENSe:FREQuency:STARt maximum
SENSe:FREQuency:STARt 1000000
```

3.6.4 String Parameters

In some cases, the instrument can accept parameters made of character strings. Such strings are enclosed with single quotes (') or double quotes ("). For example, the file name in the state saving command:

```
MMEMory:STORe "state01.sta"
```

3.6.5 Numeric Lists

The numeric lists (<numeric list>) are used to specify a variable number of numerical parameters, for example:

```
CALC:LIMit:DATA 2,1,1E9,3E9,0,0,2,1E9,3E9,-3,-3
```

3.7 Query Commands

The query commands read out the parameter values from the instrument. After a query command has been sent, the response should return via remote control interface.

The query commands has a question mark ('?') in the end of the command. Many of the commands have two forms. The form with a question mark writes the parameter, the form without a question mark reads out the parameter. For expample:

```
SENSe:FREQuency:STARt 1 MHz
SENSe:FREQuency:STARt?
```

3.8 Numeric Suffixes

The instrument contains several items of the same type, such as 9 channels, each of which in turn contains 8 traces, etc. A numeric suffix is used to denote the item number in a command. The suffix is added to the keyword of the item (channel, trace, etc). For expample, in the following specification the channel number <Ch> and trace number <Tr> indicate the channel and trace, to which this command is addressed:

```
CALCulate<Ch>:PARameter<Tr>:DEFine
```

According to this specification, the command referred to the trace 2 of the channel 1 will be written as follows:

```
CALC1:PAR2:DEF
```

The numeric suffix can be omitted. In this case, it is 1 by default. For example, the following commands are equivalent:

```
CALC:PAR:DEF
CALC1:PAR1:DEF
```

3.9 Compound Commands

It is possible to enter more than one command in the same command line. The commands in the line are separated by a semicolon (';'). The specification of the first command is valid for the following command, except for the last leaf before the semicolon. For expample:

```
SENS:FREQ:STAR 1 MHZ;STOP 2MHZ
```

If you need to start the next command from the highest level of the structure, this command should start from a colon (':'):

```
SENS:FREQ:STAR 1 MHZ;:CALC:PAR:DEF S21
```

3.10 IEEE488.2 Common Commands Overview

A SCPI compatible instrument must support a set of common commands of IEEE488.2 standard. These commands start with an asterix ('*'). The list of such commands see below:

*CLS
*ESE
*ESE?
*ESR?
*IDN?
*OPC
*OPC?
*RST
*SRE
*SRE?
*STB?
*TRG
*WAI

These commands are used for resetting, state queries, etc.

3.11 String Terminator

The <new line> character (ASCII 10) in the last data byte of the command string is used as a command terminator. The string of instructions sent to the instrument is executed after the instruction terminator is received.

4 Remote Control Commands

4.1 Conventions

The following conventions are used throughout the Manual.

4.1.1 Syntax

The following symbols are used in command syntax:

<>	identifiers enclosed in angular brackets indicated that a particular type of data must be specified
[ ]	part enclosed in square brackets can be omitted
{ }	part enclosed in curly brackets indicates that you must select one of the items in this part. Individual items are separated by a vertical bar "|"
Space	space separates commands from parameters
,	comma separates adjacent parameters
…	ellipses indicates that parameters in that part are omitted

4.1.2 Identifiers

Identifier	Parameter	Description
<numeric>	Number	{<integer>|<real>}
<frequency>	Frequency	<numeric>{[HZ]|KHZ|MHZ|GHZ}
<power>	Power	<numeric>{[DBM]|DBMW|DBW|KW|W|MW|UW|NW}
<time>	Time	<numeric>{[S]|MS|US|NS|PS|FS}
<logmag>	Log Amplitude	<numeric>[DB]
<phase>	Phase	<numeric>{[DEG]|MADEG|KDEG|MDEG|UDEG}
<stimulus>	Stimulus	{<frequency>|<power>|<time>}
<response>	Response	{<logmag>|<phase>|<time>}
<numeric list>	Numeric List	<numeric 1>,<numeric 2>,...<numeric N>

4.2 IEEE488.2 Common Commands

*CLS

```
*CLS
```

Description	Clears the following (no query) • Error Queue • Status Byte Register • Standard Event Status Register • Operation Status Event Register • Questionable Status Event Register • Questionable Limit Status Event Register • Questionable Limit Channel Status Event Register
Target	Status Reporting System
Equivalent Softkeys	**None**

*ESE

```
*ESE <numeric>

*ESE?
```

Description	Sets or reads out the value of the Standard Event Status Enable Register (command/query).
Target	Status Reporting System
Parameter	<numeric> 0 to 255
Out of Range	Bitwise AND with 255 number
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**None**

*ESR?

```
*ESR?
```

Description	Reads out the value of the Standard Event Status Register. Executing this command clears the register value (query only).
Target	Status Reporting System
Query Response	<numeric>
Equivalent Softkeys	**None**

*IDN?

```
*IDN?
```

Description	Reads out the instrument information string. The string format: <manufacturer>, <model>, <serial number>, <software version/firmware version> (query only). For example: Copper Mountain Technologies, Planar – TR1300/1, 00000101, 0.8.12/1.0
Target	Instrument
Query Response	String up to 40 characters
Equivalent Softkeys	**None**

*OPC

```
*OPC
```

Description	Sets the OPC bit (bit 0) of the Standard Event Status Register at the completion of all pending operations (no query).
Target	Status Reporting System
Equivalent Softkeys	**None**

*OPC?

```
*OPC?
```

Description	1 is read out at the completion of all pending operations (query only).
Target	Status Reporting System
Query Response	`1`
Equivalent Softkeys	**None**

*RST

```
*RST
```

Description	Restores the default settings of the instrument (no query). There is difference from presetting the instrument with `:SYST:PRES` command – in this case the trigger mode is set to *Hold*.
Target	Instrument
Related Commands	`SYSTem:PRESet`
Equivalent Softkeys	**None**

*SRE

```
*SRE <numeric>

*SRE?
```

Description	Sets or reads out the value of the Service Request Enable Register (command/query)
Target	Status Reporting System
Parameter	<numeric> 0 to 255.
Out of Range	Bitwise AND with 255 number
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**None**

*STB?

```
*STB?
```

Description	Reads out the value of the Status Byte Register (query only).
Target	Status Reporting System
Query Response	<numeric>
Equivalent Softkeys	**None**

*TRG

```
*TRG
```

Description	If the trigger source is set to LAN (set to BUS with the `:TRIG:SOUR` command), triggers a sweep. If the trigger source is not set to the bus or the instrument is not waiting for a trigger, the command is ignored. The command is completed before the end of the sweep. (no query)
Target	Instrument
Related Commands	`TRIG:SOUR`
Equivalent Softkeys	**None**

*WAI

```
*WAI
```

Description	Waits for the execution of all commands sent before this command. (no query)
Target	Instrument
Equivalent Softkeys	**None**

4.3 Network Analyzer Commands

ABOR

```
ABORt
```

Description	Aborts the sweep and switches the triggers of all the channels to *Hold*. The channels in *Continuous* trigger mode switch to waiting for a trigger. If the trigger source is set to Internal, the channel from waiting for a trigger turns to a new sweep. (no query)
Equivalent Softkeys	**None**

CALC:CONV

```
CALCulate<Ch>[:SELected]:CONVersion[:STATe] {ON|OFF|1|0}

CALCulate<Ch>[:SELected]:CONVersion[:STATe]?
```

Description	Sets or reads out the ON/OFF state of S−parameter conversion function (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} S−parameter conversion function ON {OFF|0} S−parameter conversion function OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Analysis > Conversion > Conversion**

CALC:CONV:FUNC

```
CALCulate<Ch>[:SELected]:CONVersion:FUNCtion <char>

CALCulate<Ch>[:SELected]:CONVersion:FUNCtion?
```

Description	Sets or reads out the S–parameter conversion function type (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<char> specifies parameter: IMPedance : Equivalent transmission or reflection impedance, depending on the parameter S11 or S21 ADMittance : Equivalent transmission or reflection conductance, depending on the parameter S11 or S21 INVersion : Inverse S–parameter CONJugation : S–parameter conjugate
Out of Range	Error occurs. The command is ignored.
Query Response	{IMP|ADM|INV|CONJ}
Preset Value	IMP
Equivalent Softkeys	**Analysis > Conversion > Function { Impedance Z | Admittance Y | Inverse 1/S | Conjugation }**

CALC:CORR:EDEL:TIME

```
CALCulate<Ch>[:SELected]:CORRection:EDELay:TIME <time>

CALCulate<Ch>[:SELected]:CORRection:EDELay:TIME?
```

Description	Sets or reads out the value of the electrical delay (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<time> the electrical delay value from −10 to 10
Unit	s (second)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**Scale > Electrical Delay**

CALC:CORR:OFFS:PHAS

```
CALCulate<Ch>[:SELected]:CORRection:OFFSet:PHASe <phase>

CALCulate<Ch>[:SELected]:CORRection:OFFSet:PHASe?
```

Description	Sets or reads out the value of the phase offset (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<phase> the phase offset value from –360 to 360
Unit	° (degree)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**Scale > Phase Offset**

CALC:DATA:FDAT?

```
CALCulate<Ch>[:SELected]:DATA:FDATa?
```

Description	Reads out the formatted data array (query only). The formatted data array is the data, whose processing is completed including the formatting as the last step. Such data represent the data trace values as they are shown on the screen. The array size is 2N, where N is the number of measurement points. For the n–th point, where n from 1 to N: `<numeric 2n–1>` real number in rectangular format, real part in polar and Smith chart formats; `<numeric 2n>` 0 in rectangular format, imaginary part in polar and Smith chart formats.
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Query Response	`<numeric 1>, <numeric 2>, …<numeric 2N>` The data transfer format depends on the `FORM:DATA` command setting.
Related Commands	`CALC:FORM` `FORM:DATA`
Equivalent Softkeys	**None**

CALC:DATA:FMEM?

```
CALCulate<Ch>[:SELected]:DATA:FMEMory?
```

Description	Reads out the formatted memory array (query only). The formatted memory array is the data, whose processing is completed including the formatting as the last step. Such data represent the memory trace values as they are shown on the screen. The array size is 2N, where N is the number of measurement points. For the n–th point, where n from 1 to N: `<numeric 2n–1>` real number in rectangular format, real part in polar and Smith chart formats; `<numeric 2n>` 0 in rectangular format, imaginary part in polar and Smith chart formats.
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Query Response	`<numeric 1>, <numeric 2>, …<numeric 2N>` The data transfer format depends on the `FORM:DATA` command setting.
Notes	If the memory is empty, an error occurs and the command is ignored.

Related Commands	`CALC:MATH:MEM`
	`CALC:FORM`
	`FORM:DATA`

Equivalent Softkeys	**None**

CALC:DATA:SDAT?

```
CALCulate<Ch>[:SELected]:DATA:SDATa?
```

Description	Reads out the corrected data array (query only). The corrected data array is the data, whose processing is completed excluding the formatting as the last step. Such data represent S–parameter complex values. The array size is 2N, where N is the number of measurement points. For the n–th point, where n from 1 to N: `<numeric 2n–1>` the real part of corrected measurement; `<numeric 2n>` the imaginary part of corrected measurement.
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Query Response	`<numeric 1>, <numeric 2>, …<numeric 2N>` The data transfer format depends on the `FORM:DATA` command setting.
Related Commands	`FORM:DATA`
Equivalent Softkeys	**None**

CALC:DATA:SMEM?

```
CALCulate<Ch>[:SELected]:DATA:SMEMory?
```

Description	Reads out the corrected memory array (query only). The corrected memory array is the data, whose processing is completed excluding the formatting as the last step. Such data represent S–parameter complex values. The array size is 2N, where N is the number of measurement points. For the n–th point, where n from 1 to N: `<numeric 2n–1>` the real part of corrected measurement memory; `<numeric 2n>` the imaginary part of corrected measurement memory.
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Query Response	`<numeric 1>, <numeric 2>, …<numeric 2N>` The data transfer format depends on the `FORM:DATA` command setting.
Notes	If the memory is empty, an error occurs and the command is ignored.
Related Commands	`CALC:MATH:MEM` `FORM:DATA`
Equivalent Softkeys	**None**

CALC:FILT:TIME

```
CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME[:TYPE] <char>

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME[:TYPE]?
```

Description	Sets or reads out the gate type of the gating function (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<char> specifies the gate type: BPASs : Bandpass type NOTCh : Notch type
Out of Range	Error occurs. The command is ignored.
Query Response	{BPAS|NOTC}
Preset Value	BPAS
Equivalent Softkeys	**Analysis > Gating > Type**

CALC:FILT:TIME:CENT

```
CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:CENTer <time>

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:CENTer?
```

Description	Sets or reads out the gate center value of the gating function (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<time> the center value of the gate, the range varies depending on the frequency span and the number of points
Unit	s (second)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**Analysis > Gating > Center**

CALC:FILT:TIME:SHAP

```
CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:SHAPe <char>

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:SHAPe?
```

Description	Sets or reads out the gate shape of the gating function (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<char> specifies the gate shape: MAXimum : Maximum shape WIDE : Wide shape NORMal : Normal shape MINimum : Minimum shape
Out of Range	Error occurs. The command is ignored.
Query Response	{MAX|WIDE|NORM|MIN}
Preset Value	NORM
Equivalent Softkeys	**Analysis > Gating > Shape > { Maximum | Wide | Normal | Minimum }**

CALC:FILT:TIME:SPAN

```
CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:SPAN <time>

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:SPAN?
```

Description	Sets or reads out the gate span value of the gating function (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<time> the span value of the gate, the range varies depending on the frequency span and the number of points
Unit	s (second)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	2e-8
Equivalent Softkeys	**Analysis > Gating > Span**

CALC:FILT:TIME:STAR

```
CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:STARt <time>

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:STARt?
```

Description	Sets or reads out the gate start value of the gating function (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<time> the start value of the gate, the range varies depending on the frequency span and the number of points
Unit	s (second)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	-1e-8
Equivalent Softkeys	**Analysis > Gating > Start**

CALC:FILT:TIME:STAT

```
CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:STATe {ON|OFF|1|0}

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:STATe?
```

Description	Sets or reads out the ON/OFF state of the gating function (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Gating function ON {OFF|0} Gating function OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Analysis > Gating > Gating**

CALC:FILT:TIME:STOP

```
CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:STOP <time>

CALCulate<Ch>[:SELected]:FILTer[:GATE]:TIME:STOP?
```

Description	Sets or reads out the gate stop value of the gating function (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<time> the stop value of the gate, the range varies depending on the frequency span and the number of points
Unit	s (second)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	+1e-8
Equivalent Softkeys	**Analysis > Gating > Stop**

CALC:FORM

```
CALCulate<Ch>[:SELected]:FORMat{<char>

CALCulate<Ch>[:SELected]:FORMat?
```

Description	Sets or reads out the trace format (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<char> specifies the trace format: MLOGarithmic : Logarithmic magnitude PHASe : Phase GDELay : Group delay time SLINear : Smith chart format (Lin) SLOGarithmic : Smith chart format (Log) SCOMplex : Smith chart format (Real/Imag) SMITh : Smith chart format (R + jX) SADMittance : Smith chart format (G + jB) PLINear : Polar format (Lin) PLOGarithmic : Polar format (Log) POLar : Polar format (Real/Imag) MLINear : Linear magnitude SWR : Voltage standing wave ratio REAL : Real part IMAGinary : Imaginary part UPHase : Expanded phase
Out of Range	Error occurs. The command is ignored.
Query Response	{MLOG|PHAS|GDEL|SLIN|SLOG|SCOM|SMIT|SADM|PLIN|PLOG|POL|MLIN|SWR|REAL|IMAG|UPH}
Preset Value	MLOG
Equivalent Softkeys	**Response > Format > { Log Mag | Phase | Group Delay | Lin Mag | SWR | Real | Imag | Phase Exp }** **Response > Format > Smith > { Log/Phase | Lin/Phase | Real/Imag | R+jX | G+jB }** **Response > Format > Polar > { Log/Phase | Ling/Phase | Real/Imag }**

CALC:FSIM:SEND:DEEM:PORT:STAT

```
CALCulate<Ch>:FSIMulator:SENDed:DEEMbed:PORT<Pt>:STATe
{ON|OFF|1|0}

CALCulate<Ch>:FSIMulator:SENDed:DEEMbed:PORT<Pt>:STATe?
```

Description	Sets or reads out the ON/OFF state of the de–embedding function (command/query).
Target	Port <Pt> of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Pt>={[1]|2}
Parameter	{ON|1} De–embedding function ON {OFF|0} De–embedding function OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Analysis > Fixture Simulator > De–Embedding > Port n**

CALC:FSIM:SEND:DEEM:PORT:USER:FIL

```
CALCulate<Ch>:FSIMulator:SENDed:DEEMbed:PORT<Pt>:USER:FILename
<string>

CALCulate<Ch>:FSIMulator:SENDed:DEEMbed:PORT<Pt>:USER:FILename
?
```

Description	Sets or reads out the name of *.s2p file of the de−embedded circuit of the de−embedding function (command/query). The file contains the circuit S−parameters in Touchstone format.
Target	Port <Pt> of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Pt>={[1]|2}
Parameter	<srting>, up to 256 characters (quoted string)
Preset Value	""
Notes	If the full path of the file is not specified, the \FixtireSim subdirectory of the main directory will be searched for the file.
Equivalent Softkeys	**Analysis > Fixture Simulator > De−Embedding > S−parameters File**

CALC:FSIM:SEND:PMC:PORT:STAT

```
CALCulate<Ch>:FSIMulator:SENDed:DEEMbed:PORT<Pt>:STATe
{ON|OFF|1|0}

CALCulate<Ch>:FSIMulator:SENDed:PMCircuit:PORT<Pt>:STATe?
```

Description	Sets or reads out the ON/OFF state of the embedding function (command/query).
Target	Port <Pt> of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Pt>={[1]|2}
Parameter	{ON|1} Embedding function ON {OFF|0} Embedding function OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Analysis > Fixture Simulator > Embedding > Port n**

CALC:FSIM:SEND:PMC:PORT:USER:FIL

```
CALCulate<Ch>:FSIMulator:SENDed:PMCircuit:PORT<Pt>:USER:
FILename <string>

CALCulate<Ch>:FSIMulator:SENDed:PMCircuit:PORT<Pt>:USER:
FILename?
```

Description	Sets or reads out the name of *.s2p file of the embedded circuit of the embedding function (command/query). The file contains the circuit S–parameters in Touchstone format.
Target	Port <Pt> of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Pt>={[1]|2}
Parameter	<string>, up to 256 characters (quoted string)
Preset Value	""
Notes	If the full path of the file is not specified, the \FixtireSim subdirectory of the main directory will be searched for the file.
Equivalent Softkeys	**Analysis > Fixture Simulator > Embedding > User File**

CALC:FSIM:SEND:ZCON:PORT:Z0

```
CALCulate<Ch>:FSIMulator:SENDed:ZCONversion:PORT<Pt>:Z0[:R]
<impedance>

CALCulate<Ch>:FSIMulator:SENDed:ZCONversion:PORT<Pt>:Z0[:R]?
```

Description	Sets or reads out the value of the impedance for port impedance conversion function (command/query).
Target	Port <Pt> of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Pt>={[1]|2}
Parameter	<impedance> the impedance value from 1e−6 to 1e6
Unit	Ω (Ohm)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	50
Equivalent Softkeys	**Analysis > Fixture Simulator > Port Z Conversion > Port n Z0**

CALC:FSIM:SEND:ZCON:STAT

```
CALCulate<Ch>:FSIMulator:SENDed:ZCONversion:STATe {ON|OFF|1|0}

CALCulate<Ch>:FSIMulator:SENDed:ZCONversion:STATe?
```

Description	Sets or reads out the ON/OFF state of the port impedance conversion function (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Port Z conversion function ON {OFF|0} Port Z conversion function OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Analysis > Fixture Simulator > Port Z Conversion > Port Z Conversion**

CALC:FUNC:DATA?

`CALCulate<Ch>[:SELected]:FUNCtion:DATA?`

Description	Reads out the data array, which is the `CALC:FUNC:EXEC` command analysis result (query only). The array size is 2N, where N is the number of points set by `CALC:FUNC:POIN?` command. For the n–th point, where n from 1 to N: `<numeric 2n-1>` the response value in n–th measurement point; `<numeric 2n>` the stimulus value in n–th measurement point. Always set to 0 for the analysis of mean value, standard deviation, and peak–to–peak value.
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Query Response	`<numeric 1>, <numeric 2>, …<numeric 2N>` The data transfer format depends on the `FORM:DATA` command setting.

Related Commands	`CALC:FUNC:EXEC`
	`CALC:FUNC:POIN?`
	`FORM:DATA`

Equivalent Softkeys	**None**

CALC:FUNC:DOM

```
CALCulate<Ch>[:SELected]:FUNCtion:DOMain[:STATe] {ON|OFF|1|0}

CALCulate<Ch>[:SELected]:FUNCtion:DOMain[:STATe]?
```

Description	Sets or reads out the ON/OFF state of an arbitrary range use in `CALC:FUNC:EXEC` command (command/query).
Target	All traces of channel <Ch> (if the coupling is set to ON by the `CALC:FUNC:DOM:COUP` command), the active trace of channel <Ch> (if otherwise), <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Arbitrary range ON {OFF|0} Arbitrary range OFF (entire sweep range)
Query Response	{0|1}
Preset Value	0
Related Commands	CALC:FUNC:EXEC CALC:FUNC:DOM:COUP
Equivalent Softkeys	**None**

CALC:FUNC:DOM:COUP

```
CALCulate<Ch>[:SELected]:FUNCtion:DOMain:COUPle {ON|OFF|1|0}

CALCulate<Ch>[:SELected]:FUNCtion:DOMain:COUPle?
```

Description	Sets or reads out the ON/OFF state of trace coupling for analysis range of the `CALC:FUNC:EXEC` command (command/query).
Target	All traces of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`{ON|1}` coupling ON `{OFF|0}` coupling OFF
Query Response	`{0|1}`
Preset Value	`1`
Related Commands	`CALC:FUNC:EXEC`
Equivalent Softkeys	**None**

CALC:FUNC:DOM:STAR

```
CALCulate<Ch>[:SELected]:FUNCtion:DOMain:STARt <stimulus>

CALCulate<Ch>[:SELected]:FUNCtion:DOMain:STARt?
```

Description	Sets the start value of the analysis range of the `CALC:FUNC:EXEC` command (command/query).
Target	All traces of channel <Ch> (if the coupling is set to ON by the `CALC:FUNC:DOM:COUP` command), the active trace of channel <Ch> (if otherwise), `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<stimulus>` the start value of analysis range
Unit	{ Hz (Hertz) | s (second) | dBm (decibels above 1 milliwatt) }
Query Response	`<numeric>`
Preset Value	`0`
Related Commands	`CALC:FUNC:DOM`
Equivalent Softkeys	**None**

CALC:FUNC:DOM:STOP

```
CALCulate<Ch>[:SELected]:FUNCtion:DOMain:STOP <stimulus>

CALCulate<Ch>[:SELected]:FUNCtion:DOMain:STOP?
```

Description	Sets the stop value of the analysis range of the `CALC:FUNC:EXEC` command (command/query).
Target	All traces of channel <Ch> (if the coupling is set to ON by the `CALC:FUNC:DOM:COUP` command), the active trace of channel <Ch> (if otherwise), `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<stimulus>` the stop value of analysis range
Unit	{ Hz (Hertz) | s (second) | dBm (decibels above 1 milliwatt) }
Query Response	`<numeric>`
Preset Value	`0`
Related Commands	`CALC:FUNC:DOM`
Equivalent Softkeys	**None**

CALC:FUNC:EXEC

```
CALCulate<Ch>[:SELected]:FUNCtion:EXECute
```

Description	Executes the analysis specified by the `CALC:FUNC:TYPE` command. The analysis result can be read out by the `CALC:FUNC:DATA?` command. (no query)
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Related Commands	`CALC:FUNC:TYPE` `CALC:FUNC:DATA?`
Equivalent Softkeys	**None**

CALC:FUNC:PEXC

```
CALCulate<Ch>[:SELected]:FUNCtion:PEXCursion <response>

CALCulate<Ch>[:SELected]:FUNCtion:PEXCursion?
```

Description	Sets the lower limit for the peak excursion value when executing the peak search with the `CALC:FUNC:EXEC` command (command/query).
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<response>` the lower limit of the peak excursion value, varies depending on the data format
Unit	{ dB (decibel) | ° (degree) | s (second) }
Query Response	`<numeric>`
Preset Value	`3`
Related Commands	`CALC:FUNC:EXEC`
Equivalent Softkeys	**None**

CALC:FUNC:POIN?

```
CALCulate<Ch>[:SELected]:FUNCtion:POINts?
```

Description	Reads out the number of points (data pairs) of the analysis result by the `CALC:FUNC:EXEC` command (query only). Always reads out 1, when the search is executed for the maximum, minimum, mean, standard deviation, peak, and peak–to–peak values. The actual number of points is read out, when the search is executed for all peak or all targets.
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Query Response	`<numeric>`
Related Commands	`CALC:FUNC:EXEC`
Equivalent Softkeys	**None**

CALC:FUNC:PPOL

```
CALCulate<Ch>[:SELected]:FUNCtion:PPOLarity <char>

CALCulate<Ch>[:SELected]:FUNCtion:PPOLarity?
```

Description	Selects the polarity when performing the peak search with the `CALC:FUNC:EXEC` command (command/query).
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<char>` specifies the polarity: POSitive : Positive peaks NEGative : Negative peaks BOTH : Both positive peaks and negative peaks
Out of Range	The command is ignored.
Query Response	`{POS|NEG|BOTH}`
Preset Value	`POS`
Related Commands	`CALC:FUNC:EXEC`
Equivalent Softkeys	**None**

CALC:FUNC:TARG

```
CALCulate<Ch>[:SELected]:FUNCtion:TARGet <response>

CALCulate<Ch>[:SELected]:FUNCtion:TARGet?
```

Description	Selects the target level when performing the search for the trace and the target level crosspoints with the `CALC:FUNC:EXEC` command (command/query).
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<response>` the target value, varies depending on the data format
Unit	{ dB (decibel) | ° (degree) | s (second) }
Query Response	`<numeric>`
Preset Value	`0`
Related Commands	`CALC:FUNC:EXEC`
Equivalent Softkeys	**None**

CALC:FUNC:TTR

```
CALCulate<Ch>[:SELected]:FUNCtion:TTRansition <char>

CALCulate<Ch>[:SELected]:FUNCtion:TTRansition?
```

Description	Selects the transition type when performing the search for the trace and the target level crosspoints with the `CALC:FUNC:EXEC` command (command/query).
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<char>` specifies the type of transition: POSitive : Positive peaks NEGative : Negative peaks BOTH : Both positive peaks and negative peaks
Out of Range	The command is ignored.
Query Response	`{POS|NEG|BOTH}`
Preset Value	`POS`
Related Commands	`CALC:FUNC:EXEC`
Equivalent Softkeys	**None**

CALC:FUNC:TYPE

```
CALCulate<Ch>[:SELected]:FUNCtion:TYPE <char>

CALCulate<Ch>[:SELected]:FUNCtion:TYPE?
```

Description	Selects the type of analysis executed with the `CALC:FUNC:EXEC` command (command/query).
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<char>` specifies the transition: PTPeak : Peak–to–peak (difference between the maximum value and the minimum value) STDEV : Standard deviation MEAN : Mean value MAXimum : Maximum value MINimum : Minimum value PEAK : Search for peak APEak : Search for all the peaks ATARget : Search for all targets
Out of Range	The command is ignored.
Query Response	`{PTP|STDEV|MEAN|MAX|MIN|PEAK|APE|ATAR}`
Preset Value	`PTP`
Related Commands	`CALC:FUNC:EXEC`
Equivalent Softkeys	**None**

CALC:LIM

```
CALCulate<Ch>[:SELected]:LIMit[:STATe] {ON|OFF|1|0}

CALCulate<Ch>[:SELected]:LIMit[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the limit test function (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Limit test function ON {OFF|0} Limit test function OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Analysis > Limit Test > Limit Test**

CALC:LIM:DATA

```
CALCulate<Ch>[:SELected]:LIMit:DATA <numeric list>

CALCulate<Ch>[:SELected]:LIMit:DATA?
```

Description	Sets the data array, which is the limit line in the limit test function (command/query). The array size is 1 + 5N, where N is the number of limit line segments. For the n–th point, where n from 1 to N: `<numeric 1>` the number of limit line segments N is from 0 to 100. Setting 0 clears the limit line. `<numeric 5n –3>` type of the n–th limit line segment • 0: Off. • 1: Upper limit • 2: Lower limit • 3: Single Point limit `<numeric 5n–2>` the stimulus value in the start point of the n–th segment `<numeric 5n–1>` the stimulus value in the end point of the n–th segment `<numeric 5n–0>` the response value in the start point of the n–th segment `<numeric 5n+1>` the response value in the end point of the n–th segment
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Query Response	`<numeric 1>, <numeric 2>, …<numeric 5N+1>` The data transfer format depends on the `FORM:DATA` command setting.
Notes	If the array size is not 1 + 5N, where N is `<numeric 1>`, an error occurs. If `<numeric 5n–3>` is less than 0 or more than 2, an error occurs. When `<numeric 5n–2>`, `<numeric 5n–1>`, `<numeric 5n–0>`, and `<numeric 5n+1>` elements are out of allowable range, the value is set to the limit, which is closer to the specified value.
Related Commands	`FORM:DATA`
Equivalent Softkeys	**Analysis > Limit Test > Edit Limit Line**

CALC:LIM:DISP

```
CALCulate<Ch>[:SELected]:LIMit:DISPlay[:STATe] {ON|OFF|1|0}

CALCulate<Ch>[:SELected]:LIMit:DISPlay[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the limit line display of the limit test function (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Limit line display ON {OFF|0} Limit line display OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Analysis > Limit Test > Limit Line**

CALC:LIM:FAIL?

```
CALCulate<Ch>[:SELected]:LIMit:FAIL?
```

Description	Reads out the limit test result (query only).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Query Response	1 Fail 0 Pass
Equivalent Softkeys	**None**

CALC:LIM:OFFS:AMPL

```
CALCulate<Ch>[:SELected]:LIMit:OFFSet:AMPLitude <response>

CALCulate<Ch>[:SELected]:LIMit:OFFSet:AMPLitude?
```

Description	Sets the value of the limit line offset along Y–axis (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<response> the value of the limit line offset along Y–axis, varies depending on the data format
Unit	{ dB (decibel) | ° (degree) | s (second) }
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**Analysis > Limit Test > Limit Line Offsets > Response Offset**

CALC:LIM:OFFS:STIM

```
CALCulate<Ch>[:SELected]:LIMit:OFFSet:STIMulus <stimulus>

CALCulate<Ch>[:SELected]:LIMit:OFFSet:STIMulus?
```

Description	Sets the value of the limit line offset along X−axis (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<stimulus> the value of the limit line offset along X−axis
Unit	{ Hz (Hertz) | s (second) | dBm (decibels above 1 milliwatt) }
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**Analysis > Limit Test > Limit Lines Offsets > Stimulus Offset**

CALC:LIM:REP:ALL?

```
CALCulate<Ch>[:SELected]:LIMit:REPort:ALL?
```

Description	Reads out the data array, which is the limit test results (query only). The array size is 4N, where N is the number of measurement points. For the n–th point, where n from 1 to N: `<numeric 4n–3>` the stimulus value in the n–th point; `<numeric 4n–2>` the limit test result in the n–th point; • –1: No limit • 0: Fail • 1: Pass `<numeric 4n–1>` the upper limit value in the n–th point (0 – if there is no limit) `<numeric 4n–0>` the lower limit value in the n–th point (0 – if there is no limit)
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Query Response	`<numeric 1>, <numeric 2>, …<numeric 4N>` The data transfer format depends on the `FORM:DATA` command setting.
Related Commands	`FORM:DATA`
Equivalent Softkeys	**None**

CALC:LIM:REP:POIN?

```
CALCulate<Ch>[:SELected]:LIMit:REPort:POINts?
```

Description	Reads out the number of the measurement points that failed the limit test (query only). The stimulus data array of these points can be read out by the `CALC:LIM:REP?` command.
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Query Response	`<numeric>`
Related Commands	`CALC:LIM:REP?`
Equivalent Softkeys	**None**

CALC:LIM:REP?

```
CALCulate<Ch>[:SELected]:LIMit:REPort[:DATA]?
```

Description	Reads out the data array, which is the stimulus values of the measurement points that failed the limit test (query only). The array size is set by the `CALC:LIM:REP:POIN?` command.
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Query Response	<numeric 1>, <numeric 2>,…<numeric N> The data transfer format depends on the `FORM:DATA` command setting.
Related Commands	`CALC:LIM:REP:POIN?` `FORM:DATA`
Equivalent Softkeys	**None**

CALC:MARK

```
CALCulate<Ch>[:SELected]:MARKer<Mk>[:STATe] {ON|OFF|1|0}

CALCulate<Ch>[:SELected]:MARKer<Mk>[:STATe]?
```

Description	Sets or reads out the ON/OFF state of a marker (command/query). Turning ON a marker with the number from 1 to 15 will turn ON all the markers of smaller numbers. Turning OFF a marker with the number from 1 to 15 will turn OFF all the markers of greater numbers (except of the reference marker). Turning ON/OFF the reference marker with number 16 does not turn ON/OFF the markers with the numbers from 1 to 15, but switchs these markers to the relative measurement mode.
Target	Marker <Mk> of the active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Mk>={[1]|2|3|4|5|6|7|8|9|10|11|12|13|14|15|16}
Parameter	{ON|1} marker ON {OFF|0} marker OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Markers > Add Marker | Remove Marker** **Markers > Reference Marker**

CALC:MARK:ACT

```
CALCulate<Ch>:SELected]:MARKer<Mk>:ACTivate
```

Description	Sets the active marker (no query). If the marker is not ON, this function will turn the marker ON. Turning ON a marker with the number from 1 to 15 will turn ON all the markers of smaller numbers. Turning ON the reference marker with number 16 does not turn ON the markers with the numbers from 1 to 15, but switchs these markers to the relative measurement mode.

Target	Marker <Mk> of the active trace of channel <Ch>,
	`<Ch>={[1]|2|3|4|5|6|7|8|9}`
	`<Mk>={[1]|2|3|4|5|6|7|8|9|10|11|12|13|14|15|16}`

Equivalent Softkeys	**Markers > Active Marker > Marker n** **Markers > Reference Marker**

CALC:MARK:BWID

```
CALCulate<Ch>[:SELected]:MARKer:BWIDth[:STATe] {ON|OFF|1|0}

CALCulate<Ch>[:SELected]:MARKer:BWIDth[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the bandwidth search function (command/query).
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`{ON|1}` Bandwidth search function ON `{OFF|0}` Bandwidth search function OFF
Query Response	`{0|1}`
Preset Value	`0`
Equivalent Softkeys	**Markers > Marker Math > Bandwidth Search > Bandwidth Search**

CALC:MARK:BWID:DATA?

`CALCulate<Ch>[:SELected]:MARKer<Mk>:BWIDth:DATA?`

Description	Reads out the bandwidth search result (query only). The bandwidth search can performed relatively to the marker <Mk>, or relatively to the absolute maximum value of the trace (in this case the number of the marker is ignored), what is set by the `CALC:MARK:BWID:REF` command. The data include 4 elements: `<numeric 1>` Bandwidth; `<numeric 2>` Center frequency; `<numeric 3>` Q value; `<numeric 4>` Loss;

Target	Marker <Mk> of the active trace of channel <Ch>,
	`<Ch>={[1]|2|3|4|5|6|7|8|9}`
	`<Mk>={[1]|2|3|4|5|6|7|8|9|10|11|12|13|14|15|16}`

Query Response	`<numeric 1>, <numeric 2>, …<numeric 4>`
Related Commands	`CALC:MARK:BWID:REF`
Notes	If the bandwidth search is impossible, all the read out values are 0. If the search is performed relatively to a maker, which is OFF, an error occurs.
Equivalent Softkeys	**None**

CALC:MARK:BWID:REF

```
CALCulate<Ch>[:SELected]:MARKer:BWIDth:REFerence <char>

CALCulate<Ch>[:SELected]:MARKer:BWIDth:REFerence?
```

Description	Selects the reference point for the bandwidth search function: reference marker or absolute maximum value of the trace (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<char> choose from: MARKer : Bandwidth search relative to the reference marker MAXimum : Bandwidth search relative to the absolute maximum of the trace MINimum : Bandwidth search relative to the absolute minimum of the trace
Out of Range	The command is ignored.
Query Response	{MAX|MARK|MIN}
Preset Value	MAX
Equivalent Softkeys	**Markers > Marker Math > Bandwidth Search > Search Ref To**

CALC:MARK:BWID:THR

```
CALCulate<Ch>[:SELected]:MARKer<Mk>:BWIDth:THReshold
<response>

CALCulate<Ch>[:SELected]:MARKer<Mk>:BWIDth:THReshold?
```

Description	Sets the bandwidth definition value. (command/query).
Target	Marker <Mk> of the active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Mk>={[1]|2|3|4|5|6|7|8|9|10|11|12|13|14|15|16}
Parameter	<response> the bandwidth definition value, the range varies depending on the data format
Unit	{ dB (decibel) | ° (degree) | s (second) }
Query Response	<numeric>
Preset Value	-3
Equivalent Softkeys	**Markers > Marker Math > Bandwidth Search > Bandwidth Value**

CALC:MARK:BWID:TYPE

```
CALCulate<Ch>[:SELected]:MARKer:BWIDth:TYPE <char>

CALCulate<Ch>[:SELected]:MARKer:BWIDth:TYPE?
```

Description	Sets the type of the bandwidth search function (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<char> specifies the type of the bandwidth: BPASs : Bandpass NOTCh : Notch
Out of Range	The command is ignored.
Query Response	{BPAS|NOTC}
Preset Value	BPAS
Equivalent Softkeys	**Markers > Marker Math > Bandwidth Search > Type**

CALC:MARK:COUN

```
CALCulate<Ch>[:SELected]:MARKer:COUNt <numeric>

CALCulate<Ch>[:SELected]:MARKer:COUNt?
```

Description	Sets the number of the turned ON markers (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric>, range from 0 to 16
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**None**

CALC:MARK:COUP

```
CALCulate<Ch>[:SELected]:MARKer:COUPle {ON|OFF|1|0}

CALCulate<Ch>[:SELected]:MARKer:COUPle?
```

Description	Sets or reads out the ON/OFF state of the marker coupling function (command/query).
Target	All the traces of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Marker coupling ON {OFF|0} Marker coupling OFF
Query Response	{0|1}
Preset Value	1
Equivalent Softkeys	**Marker > Properties > Marker Couple**

CALC:MARK:FUNC:DOM

```
CALCulate<Ch>[:SELected]:MARKer:FUNCtion:DOMain[:STATe]
{ON|OFF|1|0}

CALCulate<Ch>[:SELected]:MARKer:FUNCtion:DOMain[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the marker search range (command/query).
Target	All traces of channel <Ch> (if the marker search range coupling is set to ON by the `CALC:MARK:FUNC:DOM:COUP` command), The active trace of channel <Ch> (if otherwise), `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`{ON|1}` Marker search range ON `{OFF|0}` Marker search range OFF
Query Response	`{0|1}`
Preset Value	`0`
Equivalent Softkeys	**Markers > Marker Search > Search Range**

CALC:MARK:FUNC:DOM:STAR

```
CALCulate<Ch>[:SELected]:MARKer:FUNCtion:DOMain:STARt
<stimulus>

CALCulate<Ch>[:SELected]:MARKer:FUNCtion:DOMain:STARt?
```

Description	Sets or reads out the start value of the marker search range (command/query).
Target	All traces of channel <Ch> (if the marker search range coupling is set to ON by the `CALC:MARK:FUNC:DOM:COUP` command), The active trace of channel <Ch> (if otherwise), `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<stimulus>` the start value of the marker search
Unit	{ Hz (Hertz) | s (second) | dBm (decibels above 1 milliwatt) }
Query Response	`<numeric>`
Preset Value	`3E5` (Planar – TR1300/1) `2E4` (TR5048) `2E4` (TR7530)
Equivalent Softkeys	**Markers > Marker Search > Search Start**

CALC:MARK:FUNC:DOM:STOP

```
CALCulate<Ch>[:SELected]:MARKer:FUNCtion:DOMain:STOP
<stimulus>

CALCulate<Ch>[:SELected]:MARKer:FUNCtion:DOMain:STOP?
```

Description	Sets or reads out the stop value of the marker search range (command/query).
Target	All traces of channel <Ch> (if the marker search range coupling is set to ON by the `CALC:MARK:FUNC:DOM:COUP` command), The active trace of channel <Ch> (if otherwise), `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<stimulus>` the stop value of the marker search
Unit	{ Hz (Hertz) | s (second) | dBm (decibels above 1 milliwatt) }
Query Response	`<numeric>`
Preset Value	`1.3E9` (Planar – TR1300/1) `4.8E9` (TR5048) `3.0E9` (TR7530)
Equivalent Softkeys	**Markers > Marker Search > Search Stop**

CALC:MARK:FUNC:EXEC

```
CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:EXECute
```

Description	Executes the marker search according to the specified criterion. The type of the marker search is set by the `CALC:MARK:FUNC:TYPE` command. (no query)

Target	Marker <Mk> of the active trace of channel <Ch>,
	`<Ch>={[1]|2|3|4|5|6|7|8|9}`
	`<Mk>={[1]|2|3|4|5|6|7|8|9|10|11|12|13|14|15|16}`

Related Commands	`CALC:MARK:FUNC:TYPE` `CALC:MARK:FUNC:DOM`
Equivalent Softkeys	**Markers > Marker Search > { Maximum | Minimum }** **Markers > Marker Search > Peak > { Search Peak | Search Max Peak | Search Peak Left | Search Peak Right }** **Markers > Marker Search > Target > { Search Target | Search Target Left | Search Target Right }**

CALC:MARK:FUNC:PEXC

```
CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:PEXCursion
<response>

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:PEXCursion?
```

Description	Sets or reads out the peak excursion value, when the marker search for peak is performed by the `CALC:MARK:FUNC:EXEC` command (command/query).

Target	Marker <Mk> of the active trace of channel <Ch>,
	`<Ch>={[1]|2|3|4|5|6|7|8|9}`
	`<Mk>={[1]|2|3|4|5|6|7|8|9|10|11|12|13|14|15|16}`

Parameter	`<response>` the peak excursion value, the range varies depending on the data format
Unit	{ dB (decibel) | ° (degree) | s (second) }
Query Response	`<numeric>`
Preset Value	`1`
Equivalent Softkeys	**Markers > Marker Search > Peak > Peak Excursion**

CALC:MARK:FUNC:PPOL

```
CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:PPOLarity <char>

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:PPOLarity?
```

Description	Selects the peak polarity, when the marker search for peak is performed by the `CALC:MARK:FUNC:EXEC` command (command/query).
Target	Marker <Mk> of the active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Mk>={[1]|2|3|4|5|6|7|8|9|10|11|12|13|14|15|16}
Parameter	<char> specifies the peak polarity: POSitive : Positive polarity NEGative : Negative polarity BOTH : Both positive polarity and negative polarity
Out of Range	The command is ignored.
Query Response	{POS|NEG|BOTH}
Preset Value	POS
Related Commands	CALC:MARK:FUNC:EXEC
Equivalent Softkeys	**Markers > Marker Search > Peak > Peak Polarity > { Positive | Negative | Both }**

CALC:MARK:FUNC:TARG

```
CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:TARGet <response>

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:TARGet?
```

Description	Sets or reads out the target value, when the marker search for target is performed by the `CALC:MARK:FUNC:EXEC` command (command/query).
Target	Marker <Mk> of the active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Mk>={[1]|2|3|4|5|6|7|8|9|10|11|12|13|14|15|16}
Parameter	`<response>` the target value, the range varies depending on the data format
Unit	{ dB (decibel) | ° (degree) | s (second) }
Query Response	`<numeric>`
Preset Value	`0`
Equivalent Softkeys	**Markers > Marker Search > Target > Target Value**

CALC:MARK:FUNC:TRAC

```
CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:TRACking
{ON|OFF|1|0}

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:TRACking?
```

Description	Sets or reads out the ON/OFF state of the marker search tracking function (command/query).
Target	Marker <Mk> of the active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Mk>={[1]|2|3|4|5|6|7|8|9|10|11|12|13|14|15|16}
Parameter	{ON|1} Marker search tracking ON {OFF|0} Marker search tracking OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Markers > Marker Search > Tracking**

CALC:MARK:FUNC:TTR

```
CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:TTRansition
<char>

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:TTRansition?
```

Description	Selects the type of the target transition, when the marker search for transition is performed by the `CALC:MARK:FUNC:EXEC` command (command/query).
Target	Marker <Mk> of the active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Mk>={[1]|2|3|4|5|6|7|8|9|10|11|12|13|14|15|16}
Parameter	<char> specifies the type of the target transition: POSitive : Positive target transition NEGative : Negative target transition BOTH : Both positive target transition and negative target transition
Out of Range	The command is ignored.
Query Response	{POS|NEG|BOTH}
Preset Value	POS
Related Commands	CALC:MARK:FUNC:EXEC
Equivalent Softkeys	**Marker > Marker Search > Target > Target Transition > { Positive | Negative | Both }**

CALC:MARK:FUNC:TYPE

```
CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:TYPE <char>

CALCulate<Ch>[:SELected]:MARKer<Mk>:FUNCtion:TYPE?
```

Description	Selects the type of the marker search, which is performed by the `CALC:MARK:FUNC:EXEC` command (command/query).
Target	Marker <Mk> the active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Mk>={[1]|2|3|4|5|6|7|8|9|10|11|12|13|14|15|16}
Parameter	`<char>` specifies the type of the marker search: MAXimum : Maximum value search MINimum : Minimum value search PEAK : Peak search LPEak : Peak search to the left from the marker RPEak : Peak search to the right from the marker TARGet : Target search LTARget : Target search to the left from the marker RTARget : Target search to the right from the marker
Out of Range	The command is ignored.
Query Response	{MAX|MIN|PEAK|LPE|RPE|TARG|LTAR|RTAR}
Preset Value	MAX
Related Commands	CALC:MARK:FUNC:EXEC
Equivalent Softkeys	**Markers > Marker Search > { Maximum | Minimum }** **Markers > Marker Search > Peak > { Search Peak | Search Max Peak | Search Peak Left | Search Peak Right }** **Markers > Marker Search > Target > { Search Target | Search Target Left | Search Target Right }**

CALC:MARK:REF

```
CALCulate<Ch>[:SELected]:MARKer:REFerence[:STATe] {ON|OFF|1|0}

CALCulate<Ch>[:SELected]:MARKer:REFerence[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the reference marker (command/query). When the reference marker is turned ON, all the values of the other markers turn to relative values.
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Reference marker ON {OFF|0} Reference marker OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Markers > Reference Marker**

CALC:MARK:SET

```
CALCulate<Ch>[:SELected]:MARKer<Mk>:SET <char>
```

Description	Sets the value of the specified item to the value of the position of the marker (no query).
Target	Marker <Mk> of the active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Mk>={[1]|2|3|4|5|6|7|8|9|10|11|12|13|14|15|16}
Parameter	<char> specifies the type of the marker search: STARt : Sweep start value set to the stimulus value of the marker position. STOP : Sweep stop value set to the stimulus value of the marker position. CENTer : Sweep center value set to the stimulus value of the marker position. RLEVel : Reference value set to the response value of the marker position.
Out of Range	The command is ignored.
Equivalent Softkeys	**Markers > Marker Functions > { Marker–>Start | Marker–>Stop | Marker –>Center | Marker–>Ref Value }**

CALC:MARK:X

```
CALCulate<Ch>[:SELected]:MARKer<Mk>:X <stimulus>

CALCulate<Ch>[:SELected]:MARKer<Mk>:X?
```

Description	Sets or reads out the stimulus value of the marker (command/query).
Target	Marker <Mk> of the active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Mk>={[1]|2|3|4|5|6|7|8|9|10|11|12|13|14|15|16}
Parameter	<stimulus> the stimulus value of the marker, the range is from the stimulus start value to the stimulus stop value currently set
Unit	{ Hz (Hertz) | s (second) | dBm (decibels above 1 milliwatt) }
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	Stimulus center value
Equivalent Softkeys	**Markers > Edit Stimulus**

CALC:MARK:Y?

```
CALCulate<Ch>[:SELected]:MARKer<Mk>:Y?
```

Description	Reads out the response value of the marker (query only). If the reference marker is turned ON, the values of the markers from 1 to 15 are read out as relative values to the reference marker. The data include 2 elements: `<numeric 1>` real number in rectangular format, real part in polar and Smith chart formats; `<numeric 2>` 0 in rectangular format, imaginary part in polar and Smith chart formats.

Target	Marker <Mk> of the active trace of channel <Ch>,
	`<Ch>={[1]|2|3|4|5|6|7|8|9}`
	`<Mk>={[1]|2|3|4|5|6|7|8|9|10|11|12|13|14|15|16}`

Query Response	`<numeric 1>, <numeric 2>`
Related Commands	`CALC:MARK:REF`
Equivalent Softkeys	**None**

CALC:MATH:DEL

```
CALCulate<Ch>[:SELected]:MATH:DELete
```

Description	Removes the data (All memory traces) from the memory trace.
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Equivalent Softkeys	**Trace > Delete All Memory**

CALC:MATH:FUNC

```
CALCulate<Ch>[:SELected]:MATH:FUNCtion <char>

CALCulate<Ch>[:SELected]:MATH:FUNCtion?
```

Description	Selects the math operation between the measurement data and the memory trace data. The math result replaces the data trace. If the data trace is not saved, the command is ignored (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<char> specifies the math operation: DIVide : Division Data / Mem. MULTiply : Multiplication Data x Mem. ADD : Addition Data + Mem. SUBTract : Subtraction Data – Mem. NORMal : No math
Out of Range	Error occurs. The command is ignored.
Query Response	{NORM|SUBT|DIV|ADD|MULT}
Preset Value	NORM
Related Commands	CALC:MATH:MEM
Equivalent Softkeys	**Trace > Data Math > { Data/Mem | Data*Mem | Data+Mem | Data–Mem | OFF }**

CALC:MATH:MEM

```
CALCulate<Ch>[:SELected]:MATH:MEMorize
```

Description	Saves the measurement data to the memory trace. Automatically turns on the display the memory trace (no query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Equivalent Softkeys	**Trace > Display > Data–>Memory**

CALC:MST

```
CALCulate<Ch>[:SELected]:MSTatistics[:STATe] {ON|OFF|1|0}

CALCulate<Ch>[:SELected]:MSTatistics[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the math statistics display (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Statistics display ON {OFF|0} Statistics display OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Markers > Marker Math > Statistics > Statistics**

CALC:MST:DATA?

```
CALCulate<Ch>[:SELected]:MSTatistics:DATA?
```

Description	Reads out the math statistics values (query only). The statistics function is applied either over the whole range (for all the trace), or within the range specified by `CALC:MST:DOM` command (the range limits are determined by two markers). The data include 3 elements: `<numeric 1>` Mean value; `<numeric 2>` Standard deviation; `<numeric 3>` Peak–to–peak (difference between the maximum value and the minimum value).
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Query Response	`<numeric 1>, <numeric 2>, <numeric 3>`
Related Commands	`CALC:MST:DOM`
Equivalent Softkeys	**None**

CALC:MST:DOM

```
CALCulate<Ch>[:SELected]:MSTatistics:DOMain[:STATe]
{ON|OFF|1|0}

CALCulate<Ch>[:SELected]:MSTatistics:DOMain[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the math statistics range (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Statistics range ON {OFF|0} Statistics range OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Markers > Marker Math > Statistics > Statistics Range**

CALC:MST:DOM:STAR

```
CALCulate<Ch>[:SELected]:MSTatistics:DOMain[:MARKer]:STARt
<numeric>

CALCulate<Ch>[:SELected]:MSTatistics:DOMain[:MARKer]:STARt?
```

Description	Sets or reads out the number of the marker, which specifies the start frequency of the math statistics range (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> marker number from 1 to 16
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	1
Equivalent Softkeys	**Markers > Marker Math > Statistics > Statistics Start**

CALC:MST:DOM:STOP

```
CALCulate<Ch>[:SELected]:MSTatistics:DOMain[:MARKer]:STOP
<numeric>

CALCulate<Ch>[:SELected]:MSTatistics:DOMain[:MARKer]:STOP?
```

Description	Sets or reads out the number of the marker, which specifies the stop frequency of the math statistics range (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> marker number from 1 to 16
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	2
Equivalent Softkeys	**Markers > Marker Math > Statistics > Statistics Stop**

CALC:PAR:COUN

```
CALCulate<Ch>:PARameter:COUNt <numeric>

CALCulate<Ch>:PARameter:COUNt?
```

Description	Sets or reads out the number of traces in the channel (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> the number of the trace in the channel from 1 to 16
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	1
Equivalent Softkeys	**None**

CALC:PAR:DEF

```
CALCulate<Ch>:PARameter<Tr>:DEFine <char>

CALCulate<Ch>:PARameter<Tr>:DEFine?
```

Description	Sets the measurement parameter of the trace (command/query).
Target	Trace <Tr> of channel <Ch>, <Tr>={[1]|2|3|4|5|6|7|8} <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<char> specifies the measurement parameter: S11 : 11 parameter S21 : S21 parameter A : Receiver A B : Receiver B R : Receiver R
Out of Range	Error occurs. The command is ignored..
Query Response	{S11|S21|A|B|R}
Preset Value	Depends on the trace number. Tr 1, Tr 5, Tr 9, Tr 13 "S11" Tr 2, Tr 6, Tr 10, Tr 14 "S21" Tr 3, Tr 7, Tr 11, Tr 15 "S11" Tr 4, Tr 8, Tr 12, Tr 16 "S21"
Equivalent Softkeys	**Response > Measurement > S11 | S21** **Response > Measurement > { Abs A | Abs B | Abs R }**

CALC:PAR:SEL

```
CALCulate<Ch>:PARameter<Tr>:SELect
```

Description	Sets the active trace in channel <Ch> (no query).
Target	Trace <Tr> of channel <Ch>, <Tr>={[1]|2|3|4|5|6|7|8} <Ch>={[1]|2|3|4|5|6|7|8|9}
Notes	If the trace number is greater than the number of the traces displayed in the channel, an error occurs and the command is ignored.
Related Commands	`CALC:PAR:COUN`
Equivalent Softkeys	**Trace > Active Trace**

CALC:PAR:SPOR

```
CALCulate<Ch>:PARameter<Tr>:SPORt <port>

CALCulate<Ch>:PARameter<Tr>:SPORt?
```

Description	Sets or reads out the number of the receiver port, when performing absolute measurements (command/query).
Target	Trace <Tr> of channel <Ch>, <Tr>={[1]|2|3|4|5|6|7|8} <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<port> the number of the receiver port from 1 to 2
Out of Range	Error occurs. The command is ignored..
Query Response	<numeric>
Preset Value	1
Equivalent Softkeys	**Response > Measurement > { Abs A | Abs B }**

CALC:RLIM

```
CALCulate<Ch>[:SELected]:RLIMit[:STATe] {ON|OFF|1|0}

CALCulate<Ch>[:SELected]:RLIMit[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the ripple limit test (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Ripple limit test ON {OFF|0} Ripple limit test OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Analysis > Ripple Limit > Ripple Test**

CALC:RLIM:DATA

```
CALCulate<Ch>[:SELected]:RLIMit:DATA <numeric list>

CALCulate<Ch>[:SELected]:RLIMit:DATA?
```

Description	Sets the data array, which is the limit line for the ripple limit function (command/query). The array size is 1 + 4N, where N is the number of limit line segments. For the n–th point, where n from 1 to N: `<numeric 1>` the number of limit line segments N is the integer from 0 to 12. Setting 0 clears the limit line. `<numeric 4n –2>` type of the n–th limit line segment • 0: Off. • 1: On `<numeric 4n–1>` the stimulus value in the beginning point of the n–th segment `<numeric 4n–0>` the stimulus value in the end point of the n–th segment `<numeric 4n+1>` the ripple limit value of the n–th segment
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Query Response	`<numeric 1>, <numeric 2>, …<numeric 4N+1>` The data transfer format depends on the `FORM:DATA` command setting.
Notes	If the array size is not 1 + 4N, where N is `<numeric 1>`, an error occurs. If `<numeric 4n–2>` is less than 0 or more than 1, an error occurs. When `<numeric 4n–1>`, `<numeric 4n–0>`, and `<numeric 4n+1>` elements are out of allowable range, the value is set to the limit, which is closer to the specified value.
Related Commands	`FORM:DATA`
Equivalent Softkeys	**Analysis > Ripple Limit > Edit Ripple Limit**

CALC:RLIM:DISP:LINE

```
CALCulate<Ch>[:SELected]:RLIMit:DISPlay:LINE {ON|OFF|1|0}

CALCulate<Ch>[:SELected]:RLIMit:DISPlay:LINE?
```

Description	Sets or reads out the ON/OFF state of the ripple limit line display (command/query).
Target	Active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Ripple limit line ON {OFF|0} Ripple limit line OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Analysis > Ripple Limit > Limit Line**

CALC:RLIM:FAIL?

```
CALCulate#[:SELected]:RLIMit:FAIL?
```

Description	Reads out the ripple limit test result (query only).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Query Response	1 Fail 0 Pass
Equivalent Softkeys	**None**

CALC:RLIM:REP?

```
CALCulate<Ch>[:SELected]:RLIMit:REPort[:DATA]?
```

Description	Reads out the data array, which is the ripple limit test results (query only). The array size is 1+3N, where N is the number of ripple limit bands. For the n–th point, where n from 1 to N: `<numeric 1>` N total number of the bands `<numeric 3n–1>` n number of the band `<numeric 3n–0>` Ripple value in the n–th band `<numeric 3n+1>` Ripple limit test result in the n–th band: • 0: Pass • 1: Fail
Target	The active trace of channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Query Response	`<numeric 1>, <numeric 2>, …<numeric 3N+1>` The data transfer format depends on the `FORM:DATA` command setting.
Related Commands	`FORM:DATA`
Equivalent Softkeys	**None**

CALC:SMO

```
CALCulate<Ch>[:SELected]:SMOothing[:STATe] {ON|OFF|1|0}

CALCulate<Ch>[:SELected]:SMOothing[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the trace smoothing function (command/query).
Target	Active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Trace smoothing ON {OFF|0} Trace smoothing OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Response > Smoothing**

CALC:SMO:APER

```
CALCulate<Ch>[:SELected]:SMOothing:APERture <numeric>

CALCulate<Ch>[:SELected]:SMOothing:APERture?
```

Description	Sets or reads out the smoothing aperture, when performing smoothing function (command/query).
Target	Active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> the smoothing aperture from 0.01 to 20
Unit	% (percent)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	1
Equivalent Softkeys	**Response > Smo Aperture**

CALC:TRAC:DATA:FDAT?

`CALCulate<Ch>:TRACe<Tr>:DATA:FDATa?`

Description	Reads out the formatted data array (query only). The formatted data array is the data, whose processing is completed including the formatting as the last step. Such data represent the data trace values as they are shown on the screen. The array size is 2N, where N is the number of measurement points. For the n–th point, where n from 1 to N: `<numeric 2n–1>` real number in rectangular format, real part in polar and Smith chart formats; `<numeric 2n>` 0 in rectangular format, imaginary part in polar and Smith chart formats.

Target	The specified trace <Tr> of channel <Ch>,
	`<Tr>={[1]|2|3|4|5|6|7|8}`
	`<Ch>={[1]|2|3|4|5|6|7|8|9}`

Query Response	`<numeric 1>, <numeric 2>, …<numeric 2N>` The data transfer format depends on the `FORM:DATA` command setting.
Related Commands	`CALC:DATA:FDAT`
Equivalent Softkeys	**None**

CALC:TRAC:DATA:FMEM?

`CALCulate<Ch>:TRACe<Tr>:DATA:FMEMory?`

Description	Reads out the formatted memory array (query only). The formatted memory array is the data, whose processing is completed including the formatting as the last step. Such data represent the memory trace values as they are shown on the screen. The array size is 2N, where N is the number of measurement points. For the n–th point, where n from 1 to N: `<numeric 2n–1>` real number in rectangular format, real part in polar and Smith chart formats; `<numeric 2n>` 0 in rectangular format, imaginary part in polar and Smith chart formats.

Target	The specified trace <Tr> of channel <Ch>,
	`<Tr>={[1]|2|3|4|5|6|7|8}`
	`<Ch>={[1]|2|3|4|5|6|7|8|9}`

Query Response	`<numeric 1>, <numeric 2>, …<numeric 2N>` The data transfer format depends on the `FORM:DATA` command setting.
Notes	If the memory is empty, an error occurs and the command is ignored.
Related Commands	`CALC:DATA:FMEM`
Equivalent Softkeys	**None**

CALC:TRAC:DATA:SDAT?

```
CALCulate<Ch>:TRACe<Tr>:DATA:SDATa?
```

Description	Reads out the corrected data array (query only). The corrected data array is the data, whose processing is completed excluding the formatting as the last step. Such data represent S–parameter complex values. The array size is 2N, where N is the number of measurement points. For the n–th point, where n from 1 to N: `<numeric 2n–1>` the real part of corrected measurement; `<numeric 2n>` the imaginary part of corrected measurement.

Target	The specified trace <Tr> of channel <Ch>,
	`<Tr>={[1]|2|3|4|5|6|7|8}`
	`<Ch>={[1]|2|3|4|5|6|7|8|9}`

Query Response	`<numeric 1>, <numeric 2>, …<numeric 2N>` The data transfer format depends on the `FORM:DATA` command setting.
Related Commands	`CALC:DATA:SDAT`
Equivalent Softkeys	**None**

CALC:TRAC:DATA:SMEM?

`CALCulate<Ch>:TRACe<Tr>:DATA:SMEMory?`

Description	Reads out the corrected memory array (query only). The corrected memory array is the data, whose processing is completed excluding the formatting as the last step. Such data represent S–parameter complex values. The array size is 2N, where N is the number of measurement points. For the n–th point, where n from 1 to N: `<numeric 2n–1>` the real part of corrected measurement memory; `<numeric 2n>` the imaginary part of corrected measurement memory.

Target	The specified trace <Tr> of channel <Ch>,
	`<Tr>={[1]|2|3|4|5|6|7|8}`
	`<Ch>={[1]|2|3|4|5|6|7|8|9}`

Query Response	`<numeric 1>, <numeric 2>, …<numeric 2N>` The data transfer format depends on the `FORM:DATA` command setting.
Notes	If the memory is empty, an error occurs and the command is ignored.
Related Commands	`CALC:DATA:SMEM`
Equivalent Softkeys	**None**

CALC:TRAN:TIME

```
CALCulate<Ch>[:SELected]:TRANsform:TIME[:TYPE] <char>

CALCulate<Ch>[:SELected]:TRANsform:TIME[:TYPE]?
```

Description	Selects the transformation type for the time domain transformation function: bandpass response or direct current circuit (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<char>specifies the transformation type: BPASs : Bandpass LPASs : Lowpass
Out of Range	The command is ignored.
Query Response	{BPAS|LPAS}
Preset Value	BPAS
Equivalent Softkeys	**Analysis > Time Domain > Response Type > { Bandpass | Lowpass Step | Lowpass Impulse }**

CALC:TRAN:TIME:CENT

```
CALCulate<Ch>[:SELected]:TRANsform:TIME:CENTer <time>

CALCulate<Ch>[:SELected]:TRANsform:TIME:CENTer?
```

Description	Sets or reads out the time domain center value, when the time domain transformation function is turned ON (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<time> the time domain center value, the range varies depending on the specified frequency range and the number of points
Unit	s (second)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	1
Equivalent Softkeys	**Analysis > Time Domain > Center**

CALC:TRAN:TIME:IMP:WIDT

```
CALCulate<Ch>[:SELected]:TRANsform:TIME:IMPulse:WIDTh <time>

CALCulate<Ch>[:SELected]:TRANsform:TIME:IMPulse:WIDTh?
```

Description	Sets or reads out the impulse width (time domain transformation resolution), coupled with the Kaiser–Bessel window shape β parameter. The impulse width setting changes the β parameter, and setting of β parameter changes the impulse width (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<time> the impulse width, the range varies depending on the specified frequency range and the number of points
Unit	s (second)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Equivalent Softkeys	**Analysis > Time Domain > Window > Impulse Width (when the transformation type is set to Bandpass or Lowpass Impulse)**

CALC:TRAN:TIME:KBES

```
CALCulate<Ch>[:SELected]:TRANsform:TIME:KBESsel <numeric>

CALCulate<Ch>[:SELected]:TRANsform:TIME:KBESsel?
```

Description	Sets or reads out the β parameter, which controls the Kaiser–Bessel window shape, when performing time domain transformation (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> β parameter from 0 to 13
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	6
Equivalent Softkeys	**Analysis > Time Domain > Window > Kaiser Beta**

CALC:TRAN:TIME:LPFR

```
CALCulate<Ch>[:SELected]:TRANsform:TIME:LPFRequency
```

Description	Changes the frequency range to match with the lowpass type of the time domain transformation function (no query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Equivalent Softkeys	**Analysis > Time Domain > Set Frequency Low Pass**

CALC:TRAN:TIME:REFL:TYPE

```
CALCulate<Ch>[:SELected]:TRANsform:TIME:REFLection:TYPE <char>

CALCulate<Ch>[:SELected]:TRANsform:TIME:REFLection:TYPE?
```

Description	Selects the reflection distance either one way or round trip for the time domain transformation function. (command/query)
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<char> Choose from: RTRip : Round Trip OWAY : One Way
Query Response	{RTR|OWAY}
Preset Value	RTR
Equivalent Softkeys	**Analysis > Time Domain > Reflection Type > Round Trip | One Way**

CALC:TRAN:TIME:SPAN

```
CALCulate<Ch>[:SELected]:TRANsform:TIME:SPAN <time>

CALCulate<Ch>[:SELected]:TRANsform:TIME:SPAN?
```

Description	Sets or reads out the time domain span value, when the time domain transformation function is turned ON (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<time> the time domain span value, the range varies depending on the specified frequency range and the number of points
Unit	s (second)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	2e-8
Equivalent Softkeys	**Analysis > Time Domain > Span**

CALC:TRAN:TIME:STAR

```
CALCulate<Ch>[:SELected]:TRANsform:TIME:STARt <time>

CALCulate<Ch>[:SELected]:TRANsform:TIME:STARt?
```

Description	Sets or reads out the time domain start value, when the time domain transformation function is turned ON (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<time> the time domain start value, the range varies depending on the specified frequency range and the number of points
Unit	s (second)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	-1e-8
Equivalent Softkeys	**Analysis > Time Domain > Start**

CALC:TRAN:TIME:STAT

```
CALCulate<Ch>[:SELected]:TRANsform:TIME:STATe {ON|OFF|1|0}

CALCulate<Ch>[:SELected]:TRANsform:TIME:STATe?
```

Description	Sets or reads out the ON/OFF state of the time domain transformation function (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Time domain transformation ON {OFF|0} Time domain transformation OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Analysis > Time Domain > Time Domain**

CALC:TRAN:TIME:STEP:RTIM

```
CALCulate<Ch>[:SELected]:TRANsform:TIME:STEP:RTIMe <time>

CALCulate<Ch>[:SELected]:TRANsform:TIME:STEP:RTIMe?
```

Description	Sets or reads out the rise time of the step signal (time domain transformation resolution), coupled with the Kaiser–Bessel window shape β parameter. The impulse width setting changes the β parameter, and setting of β parameter changes the impulse width (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<time> the impulse width, the range varies depending on the specified frequency range and the number of points
Unit	s (second)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Equivalent Softkeys	**Analysis > Time Domain > Window > Impulse Width (when the transformation type is set to Lowpass Step)**

CALC:TRAN:TIME:STIM

```
CALCulate<Ch>[:SELected]:TRANsform:TIME:STIMulus <char>

CALCulate<Ch>[:SELected]:TRANsform:TIME:STIMulus?
```

Description	Selects the stimulus type for the time domain transformation function: impulse or step (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<char> specifies the stimulus type: IMPulse : Impulse STEP : Step
Out of Range	The command is ignored.
Query Response	{IMP|STEP}
Preset Value	IMP
Equivalent Softkeys	**Analysis > Time Domain > Response Type > { Bandpass | Lowpass Step | Lowpass Impulse }**

CALC:TRAN:TIME:STOP

```
CALCulate<Ch>[:SELected]:TRANsform:TIME:STOP <time>

CALCulate<Ch>[:SELected]:TRANsform:TIME:STOP?
```

Description	Sets or reads out the time domain stop value, when the time domain transformation function is turned ON (command/query).
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<time> the time domain stop value, the range varies depending on the specified frequency range and the number of points
Unit	s (second)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	+1e-8
Equivalent Softkeys	**Analysis > Time Domain > Stop**

CALC:TRAN:TIME:UNIT

```
CALCulate<Ch> [:SELected]:TRANsform:TIME:UNIT <char>

CALCulate<Ch> [:SELected]:TRANsform:TIME:UNIT?
```

Description	Selects the the transformation unit for the time domain transformation function: seconds, meters, feet. (command/query)
Target	The active trace of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<char> choose from: SEC : Seconds MET : Meters FEET : Feet
Query Response	{SEC|MET|FEET}
Preset Value	SEC
Equivalent Softkeys	**Analysis > Time Domain > Unit > Time, ns | Metric, m | Imperial, ft**

DISP:COL:BACK

```
DISPlay:COLor:BACK <numeric 1>,<numeric 2>,<numeric 3>

DISPlay:COLor:BACK?
```

Description	Sets or reads out the background color for trace display (command/query).
Parameter	`<numeric 1>` Red value R from 0 to 255; `<numeric 2>` Green value G from 0 to 255; `<numeric 3>` Blue value B from 0 to 255.
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	`<numeric 1>, <numeric 2>, <numeric 3>`
Preset Value	`0, 0, 0`
Equivalent Softkeys	**Display > Color > Background**

DISP:COL:GRAT

```
DISPlay:COLor:GRATicule <numeric 1>,<numeric 2>,<numeric 3>

DISPlay:COLor:GRATicule?
```

Description	Sets or reads out the grid and the graticule label color for trace display (command/query).
Parameter	`<numeric 1>` Red value R from 0 to 255; `<numeric 2>` Green value G from 0 to 255; `<numeric 3>` Blue value B from 0 to 255.
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	`<numeric 1>, <numeric 2>, <numeric 3>`
Preset Value	`160, 160, 164`
Equivalent Softkeys	**Display > Color > Grid**

DISP:COL:RES

```
DISPlay:COLor:RESet
```

Description	Restores the display settings to the default values (no query).
Equivalent Softkeys	**Display > Set Defaults**

DISP:COL:TRAC:DATA

```
DISPlay:COLor:TRACe<Tr>:DATA <numeric 1>,<numeric 2>,
<numeric 3>

DISPlay:COLor:TRACe<Tr>:DATA?
```

Description	Sets or reads out the data trace color (command/query).
Target	Trace <Tr>, <Tr>={[1]|2|3|4|5|6|7|8}
Parameter	<numeric 1> Red value R from 0 to 255; <numeric 2> Green value G from 0 to 255; <numeric 3> Blue value B from 0 to 255.
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric 1>, <numeric 2>, <numeric 3>
Preset Value	Varies depending on the trace number.
Equivalent Softkeys	**Display > Color > Data Trace**

DISP:COL:TRAC:MEM

```
DISPlay:COLor:TRACe<Tr>:MEMory <numeric 1>,<numeric 2>,
<numeric 3>

DISPlay:COLor:TRACe<Tr>:MEMory?
```

Description	Sets or reads out the memory trace color (command/query).
Target	Trace <Tr>, <Tr>={[1]|2|3|4|5|6|7|8}
Parameter	<numeric 1> Red value R from 0 to 255; <numeric 2> Green value G from 0 to 255; <numeric 3> Blue value B from 0 to 255.
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric 1>, <numeric 2>, <numeric 3>
Preset Value	Varies depending on the trace number.
Equivalent Softkeys	**Display > Color > Memory Trace**

DISP:ENAB

```
DISPlay:ENABle {OFF|ON|0|1}

DISPlay:ENABle?
```

Description	Turns ON/OFF the display update (command/query).
Parameter	{ON|1} ON {OFF|0} OFF
Query Response	{0|1}
Preset Value	1
Equivalent Softkeys	**Display > Update**

DISP:FSIG

```
DISPlay:FSIGn {ON|OFF|1|0}

DISPlay:FSIGn?
```

Description	Sets or reads out the ON/OFF state of the Fail sign display, when performing limit test or ripple limit test (command/query).
Parameter	{ON|1} Fail sign display ON {OFF|0} Fail sign display OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Analysis > Limit Test > Fail Sign** **Analysis > Ripple Limit > Fail Sign**

DISP:IMAG

```
DISPlay:IMAGe <char>

DISPlay:IMAGe?
```

Description	Sets or reads out the inverted color display of the data traces (command/query).
Parameter	<char> choose from: NORMal : Normal display INVert : Inverted color display
Out of Range	The command is ignored.
Query Response	{NORM|INV}
Preset Value	NORM
Equivalent Softkeys	**Display > Invert Color**

SYST:HIDE

```
SYSTem:HIDE
```

Description	Hides the instrument GUI (no query).
Related Commands	`SYST:SHOW`
Equivalent Softkeys	**None**

SYSTem:SHOW

```
DISPlay:SHOW
```

Description	Shows the instrument GUI hidden by the `SYST:HIDE` command (no query).
Related Commands	`SYST:HIDE`
Equivalent Softkeys	**None**

DISP:MAX

```
DISPlay:MAXimize {ON|OFF|1|0}

DISPlay:MAXimize?
```

Description	Sets or reads out the ON/OFF state of the window maximization of the active channel (command/query).
Parameter	`{ON|1}` Maximization ON `{OFF|0}` Maximization OFF
Query Response	`{0|1}`
Preset Value	`0`
Equivalent Softkeys	**Channel > Maximize Channel**

DISP:SPL

```
DISPlay:SPLit <numeric>

DISPlay:SPLit?
```

Description	Sets or reads out the layout of the channel windows on the screen (command/query). The channel window layout on the screen see below.
Parameter	`<numeric>` the number of the channel window layout from 1 to 10
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	`<numeric>`
Preset Value	`1`
Equivalent Softkeys	**Channel > Allocate Channels**

Channel window layout on the screen

1: 	2: 	3: 	4: 
5: 	6: 	7: 	8: 
9: 	10: 		

DISP:UPD

```
DISPlay:UPDate[:IMMediate]
```

Description	Updates the display once, when the display update is set to OFF by the `DISP:ENAB` command. (no query).
Related Commands	`DISP:ENAB`
Equivalent Softkeys	**None**

DISP:WIND:ACT

```
DISPlay:WINDow<Ch>:ACTivate
```

Description	Sets the active channel (no query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Notes	At attempt to set to the active channel the channel, which is not displayed by the `DISP:SPL` command, an error occurs.
Related Commands	`DISP:SPL`
Equivalent Softkeys	**Channel > Active Channel**

DISP:WIND:ANN:MARK:ALIG

```
DISPlay:WINDow<Ch>:ANNotation:MARKer:ALIGn[:TYPE] <char>

DISPlay:WINDow<Ch>:ANNotation:MARKer:ALIGn[:TYPE]?
```

Description	Sets or reads out the alignment mode of the marker display position of each trace, when the only active trace display feature is turned OFF by the `DISP:WIND:ANN:MARK:SING` command (command/query).
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<char>` choose from: VERTical : Vertical alignment HORizontal : Horizontal alignment NONE : No alignment
Out of Range	The command is ignored.
Query Response	`{NONE|VERT|HOR}`
Preset Value	`NONE`
Related Commands	`DISP:WIND:ANN:MARK:SING`
Equivalent Softkeys	**Markers > Properties > Align > { Vertical | Horizontal | OFF }**

DISP:WIND:ANN:MARK:SING

```
DISPlay:WINDow<Ch>:ANNotation:MARKer:SINGle[:STATe]
{ON|OFF|1|0}

DISPlay:WINDow<Ch>:ANNotation:MARKer:SINGle[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the marker display for the active trace only (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Only active trace markers display ON {OFF|0} Only active trace markers display OFF
Query Response	{0|1}
Preset Value	1
Equivalent Softkeys	**Markers > Properties > Active Only**

DISP:WIND:MAX

```
DISPlay:WINDow<Ch>:MAXimize {ON|OFF|1|0}

DISPlay:WINDow<Ch>:MAXimize?
```

Description	Sets or reads out the ON/OFF state of the active trace maximization of the specified channel (command/query).
Parameter	{ON|1} Maximization ON {OFF|0} Maximization OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Trace > Maximize Trace**

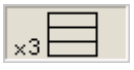
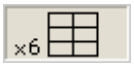
DISP:WIND:SPL

```
DISPlay:WINDow<Ch>:SPLit <numeric>

DISPlay:WINDow<Ch>:SPLit?
```

Description	Sets or reads out the layout of the graph in the channel window (command/query). The graph layout in the channel window see below.
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> the number of the graph layout from 1 to 9
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	1
Equivalent Softkeys	**Trace > Allocate Traces**

Graph layout in the channel window

1: 	2: 	3: 	4: 
5: 	6: 	7: 	8: 
9: 			

DISP:WIND:TITL

```
DISPlay:WINDow<Ch>:TITLe[:STATe] {ON|OFF|1|0}

DISPlay:WINDow<Ch>:TITLe[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the channel title display (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Channel title display ON {OFF|0} Channel title display OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Display > Title Label**

DISP:WIND:TITL:DATA

```
DISPlay:WINDow<Ch>:TITLe:DATA <string>

DISPlay:WINDow<Ch>:TITLe:DATA?
```

Description	Sets or reads out the channel title label (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<string>, up to 256 characters (quoted string)
Query Response	<string>
Preset Value	""
Equivalent Softkeys	**None**

DISP:WIND:TRAC:ANN:MARK:MEM

```
DISPlay:WINDow<Ch>:TRACe<Tr>:ANNotation:MARKer:MEMory <bool>

DISPlay:WINDow<Ch>:TRACe<Tr>:ANNotation:MARKer:MEMory?
```

Description	Turns ON/OFF the state of the memory value display on the marker.
Target	Trace <Tr> of channel <Ch>, <Tr>={[1]|2|3|4|5|6|7|8} <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<bool> Specifies the memory value display: {ON|1} ON {OFF|0} OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Markers > Properties > Memory Value**

DISP:WIND:TRAC:ANN:MARK:POS:X

```
DISPlay:WINDow<Ch>:TRACe<Tr>:ANNotation:MARKer:POSition:X
<numeric>

DISPlay:WINDow<Ch>:TRACe<Tr>:ANNotation:MARKer:POSition:X?
```

Description	Sets or reads out the display position of the marker value on the X–axis by a percentage of the display width (command/query).
Target	Trace <Tr> of channel <Ch>, <Tr>={[1]|2|3|4|5|6|7|8} <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> the display position of the marker value on the X–axis from 0 to 100
Unit	% (percent)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**Markers > Properties > Data X Position**

DISP:WIND:TRAC:ANN:MARK:POS:Y

```
DISPlay:WINDow<Ch>:TRACe<Tr>:ANNotation:MARKer:POSition:Y
<numeric>

DISPlay:WINDow<Ch>:TRACe<Tr>:ANNotation:MARKer:POSition:Y?
```

Description	Sets or reads out the display position of the marker value on the Y–axis by a percentage of the display height (command/query).
Target	Trace <Tr> of channel <Ch>, <Tr>={[1]|2|3|4|5|6|7|8} <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> the display position of the marker value on the Y–axis from 0 to 100
Unit	% (percent)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**Markers > Properties > Data Y Position**

DISP:WIND:TRAC:Y:AUTO

```
DISPlay:WINDow<Ch>:TRACe<Tr>:Y[:SCALe]:AUTO
```

Description	Executes the auto scale function for the trace (no query).
Target	Trace <Tr> of channel <Ch>, <Tr>={[1]|2|3|4|5|6|7|8} <Ch>={[1]|2|3|4|5|6|7|8|9}
Equivalent Softkeys	**Scale > Auto Scale**

DISP:WIND:TRAC:Y:PDIV

```
DISPlay:WINDow<Ch>:TRACe<Tr>:Y[:SCALe]:PDIVision <response>

DISPlay:WINDow<Ch>:TRACe<Tr>:Y[:SCALe]:PDIVision?
```

Description	Sets or reads out the trace scale. Sets the scale per division, when the data format is the rectangular format. Sets the full scale value, when the data format is the Smith chart format or the polar format.
Target	Trace <Tr> of channel <Ch>, <Tr>={[1]|2|3|4|5|6|7|8} <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<response> the scale value from 10E−18 to 1E18
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	Varies depending on the format. Logarithmic Magnitude: 10 dB/Div Phase: 40 °/Div Expand Phase: 100 °/Div Group Delay: 10e−9 s/Div Smith Chart, Polar, SWR: 1 /Div Linear Magnitude: 0.1 /Div Real part, Imaginary part: 0.2 /Div
Equivalent Softkeys	**Scale > Scale**

DISP:WIND:TRAC:Y:RLEV

```
DISPlay:WINDow<Ch>:TRACe<Tr>:Y[:SCALe]:RLEVel <response>

DISPlay:WINDow<Ch>:TRACe<Tr>:Y[:SCALe]:RLEVel?
```

Description	Sets the value of the reference line (response value on the reference line). For the rectangular format only (command/query).
Target	Trace <Tr> of channel <Ch>, <Tr>={[1]|2|3|4|5|6|7|8} <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<response> the reference value from 10E−18 to 1E18
Unit	{ dB (decibel) | ° (degree) | s (second) }
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	0 (except for SWR: 1)
Equivalent Softkeys	**Scale > Ref Value**

DISP:WIND:TRAC:Y:RPOS

```
DISPlay:WINDow<Ch>:TRACe<Tr>:Y[:SCALe]:RPOSition <numeric>

DISPlay:WINDow<Ch>:TRACe<Tr>:Y[:SCALe]:RPOSition?
```

Description	Sets the position of the reference line. For the rectangular format only (command/query).
Target	Trace <Tr> of channel <Ch>, <Tr>={[1]|2|3|4|5|6|7|8} <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> the reference line position from 0 to the number of the scale divisions (set by the DISP:WIND:Y:DIV command, 10 by default)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	5 (except for SWR: 0)
Equivalent Softkeys	**Scale > Ref Position**

DISP:WIND:Y:DIV

```
DISPlay:WINDow<Ch>:Y[:SCALe]:DIVisions <numeric>

DISPlay:WINDow<Ch>:Y[:SCALe]:DIVisions?
```

Description	Sets the number of the vertical scale divisions. For the rectangular format only (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> the number of the vertical scale divisions from 4 to 30
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	10
Resolution	2
Equivalent Softkeys	**Scale > Divisions**

FORM:BORD

```
FORMat:BORDer <char>

FORMat:BORDer?
```

Description	Sets or reads out the transfer order of each byte in data, when the binary data transfer format is set by the `FORM:DATA` command (command/query).
Parameter	`<char>` choose from: NORMal : Normal (from the most significant byte) SWAPped : Swapped (from the least significant byte)
Out of Range	The command is ignored.
Query Response	`{NORM|SWAP}`
Preset Value	`NORM`
Related Commands	`FORM:DATA`
Equivalent Softkeys	**None**

FORM:DATA

```
FORMat:DATA <char>

FORMat:DATA?
```

Description	Sets or reads out the data transfer format, when responding to the following queries: CALC:DATA:FDAT? CALC:DATA:FMEM? CALC:DATA:SDAT? CALC:DATA:SMEM? CALC:FUNC:DATA? CALC:LIM:DATA? CALC:LIM:REP? CALC:LIM:REP:ALL? CALC:RLIM:DATA? CALC:RLIM:REP? SENS:FREQ:DATA? SENS:SEGM:DATA? (command/query)
Parameter	<char> choose from: ASCii : Character format REAL : Binary format (IEEE–64 floating point) REAL32 : Binary format (IEEE–32 floating point)
Out of Range	The command is ignored.
Query Response	{ASC|REAL|REAL32}
Preset Value	ASC
Related Commands	FORM:BORD
Equivalent Softkeys	**None**

HCOP

```
HCOPy[:IMMediate]
```

Description	Prints out the image displayed on the screen without previewing (no query).
Equivalent Softkeys	**System > Print > Print Embedded**

HCOP:ABOR

```
HCOPy:ABORt
```

Description	Aborts the printout (no query).
Equivalent Softkeys	**None**

HCOP:DATE:STAM

```
HCOPy:DATE:STAMp {ON|OFF|1|0}

HCOPy:DATE:STAMp?
```

Description	Sets or reads out the ON/OFF state of the current date and time printout in the upper right corner (command/query).
Parameter	`{ON|1}` Date & time printout ON `{OFF|0}` Date & time printout OFF
Query Response	`{0|1}`
Preset Value	`1`
Equivalent Softkeys	**System > Print > Print Date & Time**

HCOP:IMAG

```
HCOPy:IMAGe <char>

HCOPy:IMAGe?
```

Description	Sets or reads out the inverted color image printout (command/query).
Parameter	<char> choose from: NORMal : Normal printout INVert : Inverted color printout
Out of Range	The command is ignored.
Query Response	{NORM|INV}
Preset Value	NORM
Equivalent Softkeys	**System > Print > Invert Image**

HCOP:PAIN

```
HCOPy:PAINt <char>

HCOPy:PAINt?
```

Description	Sets or reads out the color chart for the image printout (command/query).
Parameter	<char> choose from: COLor : Color printout GRAY : Grayscale printout BW : Black&white printout
Out of Range	The command is ignored.
Query Response	{COL|GRAY|BW}
Preset Value	BW
Equivalent Softkeys	**System > Print > Print Color**

INIT

```
INITiate<Ch>[:IMMediate]
```

Description	Sets the channel to the single trigger mode. Before this command is sent, the channel must be in hold state, otherwise an error occurs and the command is ignored. On completion of the sweep, the channel goes back into the hold state. The command is completed before the end of the sweep. (no query)
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Notes	The sweep start in single trigger mode depends on the trigger source. If the trigger is set to internal, the sweep will start immediately upon the single mode activation. If the trigger is set otherwise, the sweep will start when the trigger signal is received.
Related Commands	TRIG:SOUR
Equivalent Softkeys	**Stimulus > Trigger > Single**

INIT:CONT

```
INITiate<Ch>:CONTinuous {ON|OFF|1|0}

INITiate<Ch>:CONTinuous?
```

Description	Sets or reads out the ON/OFF state of the continuous trigger initiation mode. If the continuous trigger initiation mode is set to OFF, the channel tuns to the hold state (command /query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Continuous trigger initiation mode ON {OFF|0} Continuous trigger initiation mode OFF
Query Response	{0|1}
Preset Value	1
Notes	The sweep start in continuous trigger initiation mode depends on the trigger source. If the trigger is set to internal, the sweeps will go immediately one after another. If the trigger is set otherwise, the sweep will start when the trigger signal is received.
Related Commands	TRIG:SOUR
Equivalent Softkeys	**Stimulus > Trigger > Continuous** **Stimulus > Trigger > Hold**

MMEM:COPY

```
MMEMory:COPY <string1>,<string2>
```

Description	Copies a file (no query).
Parameter	<string1> Source file name (quoted string) <string2> Destination file name (quoted string)
Equivalent Softkeys	**None**

MMEM:DEL

```
MMEMory:DELete <string>
```

Description	Deletes a file (no query).
Parameter	`<string>` File name (quoted string)
Equivalent Softkeys	**None**

MMEM:LOAD

```
MMEMory:LOAD[:STATe] <string>
```

Description	Recalls the specified instrument state file. The file must be saved by the `MMEM:STOR` command (no query).
Parameter	`<string>` File name (quoted string)
Notes	If the full path of the file is not specified, the \State subdirectory of the main directory will be searched for the file. The instrument state file has *.sta extension by default.
Equivalent Softkeys	**System > Save/Recall > State**

MMEM:LOAD:CHAN

```
MMEMory:LOAD:CHANnel[:STATe] <char>
```

Description	Recalls the instrument state for the active channel. The file must be saved in one of the four memory registers by the `MMEM:STOR:CHAN` command (no query).
Target	Active channel set by the `DISP:WIND:ACT` command
Parameter	`<char>` choose from: A : Recall from register A B : Recall from register B C : Recall from register C D : Recall from register D
Out of Range	The command is ignored.
Equivalent Softkeys	**System > Save/Recall > Channel { State A | B | C | D }**

MMEM:LOAD:CKIT

```
MMEMory:LOAD:CKIT<Ck> <string>
```

Description	Recalls the definition file for the calibration kit. The file must be saved by the MMEM:STOR:CKIT command (no query).
Target	Calibration kit <Ck>, `<Ck>={[1]|2|3|4|5|6|7|8|9|10|11}`
Parameter	`<string>` Fine name (quoted string)
Notes	If the full path of the file is not specified, the \CalKit subdirectory of the main directory will be searched for the file. The calibration kit definition file has *.ckd extension by default.
Equivalent Softkeys	**None**

MMEM:LOAD:LIM

```
MMEMory:LOAD:LIMit <string>
```

Description	Recalls the limit table file. The file must be saved by the `MMEM:STOR:LIM` command (no query).
Target	Active trace of the active channel, set by the `CALC:PAR:SEL` command
Parameter	`<string>` File name (quoted string)
Notes	If the full path of the file is not specified, the \Limit subdirectory of the main directory will be searched for the file. The limit table file has *.lim extension by default.
Equivalent Softkeys	**Analysis > Limit Test > Edit Limit Line > Restore Limit Table**

MMEM:LOAD:RLIM

```
MMEMory:LOAD:RLIMit <string>
```

Description	Recalls the ripple limit table file. The file must be saved by `MMEM:STOR:RLIM` the command (no query).
Target	Active trace of the active channel, set by the `CALC:PAR:SEL` command
Parameter	`<string>` File name (quoted string)
Notes	If the full path of the file is not specified, the \Limit subdirectory of the main directory will be searched for the file. The ripple limit file has *.rlm extension by default.
Equivalent Softkeys	**Analysis > Ripple Limit > Edit Ripple Limit > Restore Ripple Limit Table**

MMEM:LOAD:SEGM

```
MMEMory:LOAD:SEGMent <string>
```

Description	Recalls the segment table file. The file must be saved by the `MMEM:STOR:SEGM` command (no query).
Target	Active channel, set by the `DISP:WIND:ACT` command
Parameter	`<string>` File name (quoted string)
Notes	If the full path of the file is not specified, the \Segment subdirectory of the main directory will be searched for the file. The segment file has *.seg extension by default.
Equivalent Softkeys	**Stimulus > Segment Table > Recall**

MMEM:MDIR

```
MMEMory:MDIRectory <string>
```

Description	Creates a new directory (no query).
Parameter	`<string>` Directory full name (quoted string)
Equivalent Softkeys	**None**

MMEM:STOR

```
MMEMory:STORe[:STATe] <string>
```

Description	Saves the instrument state into a file (no query).
Parameter	`<string>` File name (quoted string)
Notes	If the full path of the file is not specified, the \State subdirectory of the main directory will be searched for the file. The state file has *.sta extension by default.
Equivalent Softkeys	**System > Save > State**

MMEM:STOR:CHAN

```
MMEMory:STORe:CHANnel[:STATe] <char>
```

Description	Saves the instrument state of the items set for the active channel into one of the four memory registers (no query).
Target	Active channel set by the `DISP:WIND:ACT` command
Parameter	`<char>` choose from: A : Save to register A B : Save to register B C : Save to register C D : Save to register D
Out of Range	The command is ignored.
Equivalent Softkeys	**System > Save/Recall > Channel > { State A | B | C | D }**

MMEM:STOR:CHAN:CLE

```
MMEMory:STORe:CHANnel:CLEar
```

Description	Clears the memory of the channel state saved by the `MMEM:STOR:CHAN` command (no query).
Equivalent Softkeys	**System > Save/Recall > Channel > Clear States**

MMEM:STOR:CKIT

```
MMEMory:STORe:CKIT<Ck> <string>
```

Description	Saves the definition file for the calibration kit (no query).
Target	Calibration kit <Ck>, <Ck>={[1]|2|3|4|5|6|7|8|9|10|11}
Parameter	<string> File name (quoted string)
Notes	If the full path of the file is not specified, the \CalKit subdirectory of the main directory will be searched for the file. The calibration kit definition file has *.ckd extension by default.
Equivalent Softkeys	**None**

MMEM:STOR:FDAT

```
MMEMory:STORe:FDATa <string>
```

Description	Saves the CSV formatted data into a file (no query).
Target	Active trace of the active channel, set by the CALC:PAR:SEL command
Parameter	<string> File name (quoted string)
Notes	If the full path of the file is not specified, the \CSV subdirectory of the main directory will be searched for the file. The file has *.csv extension by default.
Equivalent Softkeys	**System > Save > Save Trace Data**

MMEM:STOR:IMAG

```
MMEMory:STORe:IMAGe <string>
```

Description	Saves the display image in BMP or PNG format into a file (no query).
Parameter	`<string>` File name (quoted string)
Notes	If the full path of the file is not specified, the \Image subdirectory of the main directory will be searched for the file. If the file has *.png extension, the file had PNG format, in all the other cases the file has BMP format.
Equivalent Softkeys	**System > Print > Print Windows > Save as...**

MMEM:STOR:LIM

```
MMEMory:STORe:LIMit <string>
```

Description	Saves the limit table into a file (no query).
Target	Active trace of the active channel, set by the `CALC:PAR:SEL` command
Parameter	`<string>` File name (quoted string)
Notes	If the full path of the file is not specified, the \Limit subdirectory of the main directory will be searched for the file. The file has *.lim extension by default.
Equivalent Softkeys	**Analysis > Limit Test > Edit Limit Line > Save Limit Table**

MMEM:STOR:RLIM

```
MMEMory:STORe:RLIMit <string>
```

Description	Saves the ripple limit table into a file (no query).
Target	Active trace of the active channel, set by the `CALC:PAR:SEL` command
Parameter	`<string>` File name (quoted string)
Notes	If the full path of the file is not specified, the \Limit subdirectory of the main directory will be searched for the file. The ripple limit file has *.rlm extension by default.
Equivalent Softkeys	**Analysis > Ripple Limit > Edit Ripple Limit > Save Ripple Limit Table**

MMEM:STOR:SEGM

```
MMEMory:STORe:SEGMent <string>
```

Description	Save the segment table in a file (no query).
Target	Active channel, set by the `DISP:WIND:ACT` command
Parameter	`<string>` File name (quoted string)
Notes	If the full path of the file is not specified, the \Segment subdirectory of the main directory will be searched for the file. The segment file has *.seg extension by default.
Equivalent Softkeys	**Stimulus > Segment Table > Save...**

MMEM:STOR:SNP

```
MMEMory:STORe:SNP[:DATA] <string>
```

Description	Saves the measured S–parameters of the active channel into a Touchstone file. The file type (1–port or 2–port) is set by the `MMEM:STOR:SNP:TYPE:S1P` and `MMEM:STOR:SNP:TYPE:S2P` commands. 1–port type file saves one reflection parameter: S11 or S22. 2–port type file saves all the four parameters: S11, S21, S12, S22. (no query)
Target	Active channel, set by the `DISP:WIND:ACT` command
Parameter	`<string>` File name (quoted string)
Notes	If the full path of the file is not specified, the \FixtureSim subdirectory of the main directory will be searched for the file. The 1–port measurement file has *.slp extension; the 2–port measurement file has *.s2p extension.
Equivalent Softkeys	**System > Save > Touchstone File > Save**

MMEM:STOR:SNP:FORM

```
MMEMory:STORe:SNP:FORMat <char>

MMEMory:STORe:SNP:FORMat?
```

Description	Sets the data format for the S–parameter saving by the MMEM:STOR:SNP command (command/query).
Target	Active channel, set by the `DISP:WIND:ACT` command
Parameter	`<char>` choose from: MA : Logarithmic Magnitude / Angle format DB : Linear Magnitude / Angle format RI : Real part /Imaginary part format
Out of Range	The command is ignored.
Query Response	`{RI|DB|MA}`
Preset Value	`RI`
Equivalent Softkeys	**System > Save > Touchstone File > Format**

MMEM:STOR:SNP:TYPE:S1P

```
MMEMory:STORe:SNP: TYPE:S1P <port>

MMEMory:STORe:SNP:TYPE:S1P?
```

Description	Sets the 1-port Touchstone file type (*.s1p) and the port number, when saving S–parameters by the `MMEM:STOR:SNP` command. (command/query)
Parameter	`<port>` port number from 1 to 2
Query Response	`<numeric>`
Preset Value	`1`
Equivalent Softkeys	**System > Save > Touchstone File > Type > 1-Port (s1p)**

MMEM:STOR:SNP:TYPE:S2P

```
MMEMory:STORe:SNP:TYPE:S2P <rcvport>, <srcport>

MMEMory:STORe:SNP:TYPE:S2P?
```

Description	Sets the 2-port Touchstone file type (*.s2p) and the port numbers, when saving S–parameters by the `MMEM:STOR:SNP` command. (command/query)
Parameter	`<rcvport>` the number of the receiver port from 1 to 2 `<srcport>` the number of the source port 1
Query Response	`<numeric1>, <numeric2>`
Equivalent Softkeys	**System > Save > Touchstone File > Type > 2-Port (s2p)**

MMEM:STOR:STYP

```
MMEMory:STORe:STYPe <char>

MMEMory:STORe:STYPe?
```

Description	Selects the type of the instrument or channel state saving by the `MMEM:STOR:CHAN` command (command/query).
Parameter	`<char>` choose from: STATe : Measurement conditions CSTate : Measurement conditions and calibration tables DSTate : Measurement conditions and data traces CDSTate : Measurement conditions, calibration tables and data traces
Out of Range	An error occurs.
Query Response	`{STAT|CST|DST|CDST}`
Preset Value	`CST`
Equivalent Softkeys	**System > Save > Save Type**

MMEM:TRAN?

```
MMEMory:TRANsfer? <string>
```

Description	Transfers the contents of a specified file from the instrument to the external computer (command/query).
Parameter	`<string>` the file name with the full path (quoted string)
Query Response	Block data transfer format. For example: #6001000 <binary block with length of 1000 bytes> #6 Symbol # introduces the data block. The next number indicates how many of the following digits describe the length of the data block; 001000 Length of the data block;
Notes	The file must be 20 Mbytes or less.
Equivalent Softkeys	**None**

OUTP

```
OUTPut[:STATe] {ON|OFF|1|0}

OUTPut[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the stimulus signal output. Measurements cannot be performed when the stimulus signal output is set to OFF (command/query).
Parameter	`{ON|1}` Stimulus signal output ON `{OFF|0}` Stimulus signal output OFF
Query Response	`{0|1}`
Preset Value	`1`
Equivalent Softkeys	**Stimulus > Power > RF Out Off**

SENS:AVER

```
SENSe<Ch>:AVERage[:STATe] {ON|OFF|1|0}

SENSe<Ch>:AVERage[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the averaging function (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Averaging ON {OFF|0} Averaging OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Response > Averaging**

SENS:AVER:CLE

```
SENSe<Ch>:AVERage:CLEar
```

Description	Restarts the averaging process, when averaging function is set to ON (no query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Related Commands	SENS:AVER
Equivalent Softkeys	**None**

SENS:AVER:COUN

```
SENSe<Ch>:AVERage:COUNt <numeric>

SENSe<Ch>:AVERage:COUNt?
```

Description	Sets or reads out the averaging factor, when the averaging function is set to ON (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> the averaging factor from 1 to 999
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	10
Related Commands	SENS:AVER
Equivalent Softkeys	**Response > Avg Factor**

SENS:BAND

```
SENSe<Ch>:BANDwidth[:RESolution] <frequency>

SENSe<Ch>:BANDwidth[:RESolution]?
```

SENS:BWID

```
SENSe<Ch>:BWIDth[:RESolution] <frequency>

SENSe<Ch>:BWIDth[:RESolution]?
```

Description	Sets or reads out the IF bandwidth (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<frequency> the IF bandwidth value from 1 to 30000
Unit	Hz (Hertz)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	10000
Resolution	In steps of 10, 30, 100, 300, 1000, 3000, 10000, 30000
Equivalent Softkeys	**Response > IF Bandwidth**

SENS:CORR:CLE

```
SENSe<Ch>:CORRection:CLEar
```

Description	Clears the calibration coefficient table (no query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Equivalent Softkeys	**None**

SENS:CORR:COEF

```
SENSe<Ch>:CORRection:COEFficient[:DATA]? <char>,
<rcvport>,<srcport>,<numeric list>

SENSe<Ch>:CORRection:COEFficient[:DATA]? <char>,
<rcvport>,<srcport>
```

Description	Writes or reads out the calibration coefficient data array (command/query). The array size is 2N, where N is the number of measurement points. For the n–th point, where n from 1 to N: `<numeric 2n–1>` real part of the calibration coefficients `<numeric 2n>` imaginary part of the calibration coefficients
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<char>` specifies the error term selection: ER : Reflection tracking ED : Directivity ES : Source match ET : Transmission tracking EX : Isolation EL : Load match `<rcvport>,` the number of the receiver port from 1 to 2 `<srcport>` the number of the source port = 1 `<numeric list>` the calibration coefficient array When ES, ER, or ED is used, the numbers of the ports `<rcvport>` and `<srcport>` must be the same. When EL, ET, or EX is used, the numbers of the ports `<rcvport>` and `<srcport>` must be different.
Query Response	`<numeric 1>, <numeric 2>, …<numeric 2N>`
Notes	The written calibration coefficients become effective only after the `SENS:CORR:COEF:SAVE` command is executed.
Equivalent Softkeys	**None**

SENS:CORR:COEF:METH:ERES

```
SENSe<Ch>:CORRection:COEFficient:METHod:ERESponse
<rcvport>,<srcport>
```

Description	Selects the ports and sets the 1–path 2–port calibration type, when the written calibration coefficients are made effective by the `SENS:CORR:COEF:SAVE` command (no query).
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<rcvport>`, the number of the receiver port from 1 to 2 `<srcport>`, the number of the source port = 1
Out of Range	If an incorrect port number is specified, an error occurs. If the same port numbers are specified, an error occurs.
Related Commands	`SENS:CORR:COEF:SAVE`
Equivalent Softkeys	**None**

SENS:CORR:COEF:METH:OPEN

`SENSe<Ch>:CORRection:COEFficient:METHod[:RESPonse]:OPEN <port>`

Description	Selects the port and sets the response calibration (Open) type, when the written calibration coefficients are made effective by the `SENS:CORR:COEF:SAVE` command (no query).
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<port>` the number of the port from 1 to 2
Out of Range	An error occurs..
Related Commands	`SENS:CORR:COEF:SAVE`
Equivalent Softkeys	**None**

SENS:CORR:COEF:METH:SHOR

```
SENSe<Ch>:CORRection:COEFficient:METHod[:RESPonse]:SHORt
<port>
```

Description	Selects the port and sets the response calibration (Short) type, when the written calibration coefficients are made effective by the `SENS:CORR:COEF:SAVE` command (no query).
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<port>` the number of the port from 1 to 2
Out of Range	An error occurs.
Related Commands	`SENS:CORR:COEF:SAVE`
Equivalent Softkeys	**None**

SENS:CORR:COEF:METH:SOLT1

```
SENSe<Ch>:CORRection:COEFficient:METHod:SOLT1 <port>
```

Description	Selects the port and sets the full 1-port calibration type, when the written calibration coefficients are made effective by the `SENS:CORR:COEF:SAVE` command (no query).
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<port>` the number of the port from 1 to 2
Out of Range	An error occurs.
Related Commands	`SENS:CORR:COEF:SAVE`
Equivalent Softkeys	**None**

SENS:CORR:COEF:METH:THRU

```
SENSe<Ch>:CORRection:COEFficient:METHod[:RESPonse]:THRU
<rcvport>, <srcport>
```

Description	Selects the ports and sets the response calibration (Thru) type, when the written calibration coefficients are made effective by the `SENS:CORR:COEF:SAVE` command (no query).
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<rcvport>`, the number of the receiver port from 1 to 2 `<srcport>`, the number of the source port = 1
Out of Range	If an incorrect port number is specified, an error occurs. If the same port numbers are specified, an error occurs.
Related Commands	`SENS:CORR:COEF:SAVE`
Equivalent Softkeys	**None**

SENS:CORR:COEF:SAVE

```
SENSe<Ch>:CORRection:COEFficient:SAVE
```

Description	Enables the written calibration coefficients depending on the selected calibration type. On completion of the command the error correction automatically turns ON. At the attempt to execute this command before all the needed calibration coefficients are written, an error occurs and the command is ignored (no query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Related Commands	Calibration type selection: SENS:CORR:COEF:METH:ERES SENS:CORR:COEF:METH:OPEN SENS:CORR:COEF:METH:SHOR SENS:CORR:COEF:METH:THRU SENS:CORR:COEF:METH:SOLT1 Calibration coefficient writing: SENS:CORR:COEF
Equivalent Softkeys	**None**

SENS:CORR:COLL:CKIT

```
SENSe<Ch>:CORRection:COLLect:CKIT[:SELect] <numeric>

SENSe<Ch>:CORRection:COLLect:CKIT[:SELect]?
```

Description	Sets or reads out the number of the selected calibration kit (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> the number of the calibration kit from 1 to 11
Out of Range	Error occurs. The command is ignored..
Query Response	<numeric>
Preset Value	1
Equivalent Softkeys	**Calibration > Cal Kit > Cal Kit n**

SENS:CORR:COLL:CKIT:LAB

```
SENSe<Ch>:CORRection:COLLect:CKIT:LABel <string>

SENSe<Ch>:CORRection:COLLect:CKIT:LABel?
```

Description	Sets or reads out the calibration kit label (command/query).
Target	Calibration kit, selected for channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<string>, up to 254 characters (quoted string)
Query Response	<string>
Preset Value	Varies depending on the number of the calibration kit.
Equivalent Softkeys	**Calibration > Edit Cal Kit > Label**

SENS:CORR:COLL:CKIT:RES

```
SENSe<Ch>:CORRection:COLLect:CKIT:RESet
```

Description	Resets the calibration kit to the factory settings (no query).
Target	Calibration kit, selected for channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Equivalent Softkeys	**Calibration > Restore Cal Kit**

SENS:CORR:COLL:CKIT:STAN:C0

```
SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:C0 <numeric>

SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:C0?
```

Description	Sets or reads out the C0 value for the open calibration standard (command/query).
Target	Standard <Std> of the calibration kit specified for channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Std>={[1]|2|...N}, where N – the number of the standards in the calibration kit
Parameter	<numeric> the C0 value from –1E18 to 1E18
Unit	1E–15 F (Farad)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Equivalent Softkeys	**Calibration > Edit Cal Kit > Capacitance > C0 10^{-15} F**

SENS:CORR:COLL:CKIT:STAN:C1

```
SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:C1 <numeric>

SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:C1?
```

Description	Sets or reads out the C1 value for the open calibration standard (command/query).
Target	Standard <Std> of the calibration kit specified for channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Std>={[1]|2|...N}, where N – the number of the standards in the calibration kit
Parameter	<numeric> the C1 value from –1E18 to 1E18
Unit	1E–27 F/Hz (Farad/Hertz)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Equivalent Softkeys	**Calibration > Edit Cal Kit > Capacitance > C1 10^{-27} F/Hz**

SENS:CORR:COLL:CKIT:STAN:C2

```
SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:C2 <numeric>

SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:C2?
```

Description	Sets or reads out the C2 value for the open calibration standard (command/query).
Target	Standard <Std> of the calibration kit specified for channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Std>={[1]|2|...N}, where N – the number of the standards in the calibration kit
Parameter	<numeric> the C2 value from –1E18 to 1E18
Unit	1E–36 F/Hz2 (Farad/Hertz2)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Equivalent Softkeys	**Calibration > Edit Cal Kit > Capacitance > C2 10^{-36} F/Hz^2**

SENS:CORR:COLL:CKIT:STAN:C3

```
SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:C3 <numeric>

SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:C3?
```

Description	Sets or reads out the C3 value for the open calibration standard (command/query).
Target	Standard <Std> of the calibration kit specified for channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Std>={[1]|2|...N}, where N – the number of the standards in the calibration kit
Parameter	<numeric> the C3 value from –1E18 to 1E18
Unit	1E–45 F/Hz3 (Farad/Hertz3)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Equivalent Softkeys	**Calibration > Edit Cal Kit > Capacitance > C3 10^{-45} F/Hz^3**

SENS:CORR:COLL:CKIT:STAN:DEL

```
SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:DELay <numeric>

SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:Delay?
```

Description	Sets or reads out the offset delay value for the calibration standard (command/query).
Target	Standard <Std> of the calibration kit specified for channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Std>={[1]|2|...N}, where N – the number of the standards in the calibration kit
Parameter	<numeric> the offset delay value form –1E18 to 1E18
Unit	s (second)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Equivalent Softkeys	**Calibration > Edit Cal Kit > Offset Delay**

SENS:CORR:COLL:CKIT:STAN:L0

```
SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:L0 <numeric>

SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:L0?
```

Description	Sets or reads out the L0 value for the short calibration standard (command/query).
Target	Standard <Std> of the calibration kit specified for channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Std>={[1]|2|...N}, where N – the number of the standards in the calibration kit
Parameter	<numeric> the L0 value from –1E18 to 1E18
Unit	1E–12 H (Henry)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Equivalent Softkeys	**Calibration > Edit Cal Kit > Inductance > L0 10^{-12} H**

SENS:CORR:COLL:CKIT:STAN:L1

```
SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:L1 <numeric>

SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:L1?
```

Description	Sets or reads out the L1 value for the short calibration standard (command/query).
Target	Standard <Std> of the calibration kit specified for channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Std>={[1]|2|...N}, where N – the number of the standards in the calibration kit
Parameter	<numeric> the L1 value from –1E18 to 1E18
Unit	1E–24 H/Hz (Henry/Hertz)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Equivalent Softkeys	**Calibration > Edit Cal Kit > Inductance > L1 10^{-24} H/Hz**

SENS:CORR:COLL:CKIT:STAN:L2

```
SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:L2 <numeric>

SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:L2?
```

Description	Sets or reads out the L2 value for the short calibration standard (command/query).
Target	Standard <Std> of the calibration kit specified for channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Std>={[1]|2|...N}, where N – the number of the standards in the calibration kit
Parameter	<numeric> the L2 value from –1E18 to 1E18
Unit	1E–33 H/Hz2 (Henry/Hertz2)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Equivalent Softkeys	**Calibration > Edit Cal Kit > Inductance > L2 10^{-33} H/Hz^2**

SENS:CORR:COLL:CKIT:STAN:L3

```
SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:L3 <numeric>

SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:L3?
```

Description	Sets or reads out the L3 value for the short calibration standard (command/query).
Target	Standard <Std> of the calibration kit specified for channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Std>={[1]|2|...N}, where N – the number of the standards in the calibration kit
Parameter	<numeric> the L3 value from –1E18 to 1E18
Unit	1E–42 H/Hz3 (Henry/Hertz3)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Equivalent Softkeys	**Calibration > Edit Cal Kit > Inductance > L3 10^{-42} H/Hz^3**

SENS:CORR:COLL:CKIT:STAN:LAB

```
SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:LABel <string>

SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:LABel?
```

Description	Sets or reads out the label for the calibration standard (command/query).
Target	Standard <Std> of the calibration kit specified for channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Std>={[1]|2|...N}, where N – the number of the standards in the calibration kit
Parameter	<string>, up to 254 characters (quoted string)
Query Response	<string>
Equivalent Softkeys	**Calibration > Edit Cal Kit > Label**

SENS:CORR:COLL:CKIT:STAN:LOSS

```
SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:LOSS <loss>

SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:LOSS?
```

Description	Sets or reads out the offset loss value for the calibration standard (command/query).
Target	Standard <Std> of the calibration kit specified for channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Std>={[1]|2|...N}, where N – the number of the standards in the calibration kit
Parameter	<loss> the offset loss value from –1E18 to 1E18
Unit	Ω/s (Ohm/second)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Equivalent Softkeys	**Calibration > Edit Cal Kit > Offset Loss**

SENS:CORR:COLL:CKIT:STAN:TYPE

```
SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:TYPE <char>

SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:TYPE?
```

Description	Sets or reads out the type of calibration standard (command/query).
Target	Standard <Std> of the calibration kit specified for channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Std>={[1]|2|...N}, where N – the number of the standards in the calibration kit
Parameter	<char> choose from: OPEN : Open SHORt : Short LOAD : Load THRU : Thru
Out of Range	Error occurs. The command is ignored..
Query Response	{OPEN|SHOR|LOAD|THRU}
Equivalent Softkeys	**Calibration > Edit Cal Kit > STD Type**

SENS:CORR:COLL:CKIT:STAN:Z0

```
SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:Z0 <impedance>

SENSe<Ch>:CORRection:COLLect:CKIT:STAN<Std>:Z0?
```

Description	Sets or reads out the offset Z0 value for the calibration standard (command/query).
Target	Standard <Std> of the calibration kit specified for channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Std>={[1]|2|...N}, where N – the number of the standards in the calibration kit
Parameter	<impedance> the offset Z0 value from –1E18 to 1E18
Unit	Ω (Ohm)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	50 or 75, depending on the selected calibration kit
Equivalent Softkeys	**Calibration > Edit Cal Kit > Offset Z0**

SENS:CORR:COLL:CLE

```
SENSe<Ch>:CORRection:COLLect:CLEar
```

Description	Clears the measurement values of the calibration standards (no query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Equivalent Softkeys	**Calibration > Calibrate > { Response (Open) | Response (Short) | Response (Thru) | One Path 2–Port Cal | Full 1–Port Cal } > Cancel**

SENS:CORR:COLL:ECAL:CHECK:EXEC

```
SENSe<Ch>:CORRection:COLLect:ECAL:CHECK:EXECute <numeric1>,
<numeric2>
```

Description	Executing confidence check
Target	AutoCal module (*Ch* – arbitrary number 1 to 9)
Equivalent Softkeys	**Calibration > Autocalibration > Confidence Check**

SENS:CORR:COLL:ECAL:ORI:EXEC

```
SENSe<Ch>:CORRection:COLLect:ECAL:ORIentation:EXECute
<numeric1>, <numeric2>
```

Description	Executing AutoCal orientation to the ports Rcv, Scr
Target	AutoCal module
Equivalent Softkeys	**Calibration > Autocalibration > Perform Auto-Orientation**

SENS:CORR:COLL:ECAL:ORI:STAT

```
SENSe:CORRection:COLLect:ECAL:ORIentation:STATe {ON|OFF|1|0}

SENSe:CORRection:COLLect:ECAL:ORIentation:STATe?
```

Description	Sets or reads out the ON/OFF state of the Auto-Orientation function used when executing AutoCal (command/query).
Target	AutoCal
Parameter	`{ON|1}` Auto-Orientation function ON `{OFF|0}` Auto-Orientation function OFF
Query Response	`{0|1}`
Preset Value	`0`
Equivalent Softkeys	**Calibration > AutoCal > Orientation > Auto-Orientation**

SENS:CORR:COLL:ECAL:PATH

```
SENSe:CORRection:COLLect:ECAL:PATH <numeric1>,<numeric2>

SENSe:CORRection:COLLect:ECAL:PATH? <numeric1>
```

Description	Sets or reads out the AutoCal module port number which is connected to a specified port of Network Analyzer (command/query).
Target	AutoCal
Parameter	<numeric1> Network Analyzer Port Number, <numeric2> AutoCal Module Port Number, 1 : Port A of AutoCal Module 2 : Port B of AutoCal Module 3 : Port C of AutoCal Module 4 : Port D of AutoCal Module
Query Response	<numeric>
Equivalent Softkeys	**Calibration > AutoCal > Orientation > Port 1 – Port A | Port 1 – Port B | Port 1 – Port C| Port 1 – Port D**

SENS:CORR:COLL:ECAL:SOLT1

```
SENSe<Ch>:CORRection:COLLect:ECAL:SOLT1 <port>
```

Description	Executes Full 1-port calibration of the specified port of specified channel (Ch) using the AutoCal module (command only).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<port> : Port Number 1
Equivalent Softkeys	**Calibration > AutoCal > Full 1-Port Calibration**

SENS:CORR:COLL:ECAL:SOLT2

```
SENSe<Ch>:CORRection:COLLect:ECAL:SOLT1 <port2>,<port1>
```

Description	Executes One Path 2-Port calibration of the specified port of specified channel (Ch) using the AutoCal module (command only).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<port1> the number of the receiver port (always 1) <port2> the number of the source port (always 2)
Equivalent Softkeys	**Calibration > AutoCal > One Path 2-Port Calibration**

SENS:CORR:COLL:ECAL:UCH

```
SENSe:CORRection:COLLect:ECAL:UCHar <char>

SENSe:CORRection:COLLect:ECAL:UCHar?
```

Description	Sets or reads out the Characteristic used when executing AutoCal (factory or user characterization). (command/query).
Target	AutoCal
Parameter	<char> specifies the stimulus type: CHAR0 : factory characterization CHAR1 : user characterization 1 CHAR2 : user characterization 2 CHAR3 : user characterization 3
Query Response	{CHAR0|CHAR1|CHAR2|CHAR3}
Preset Value	CHAR0
Equivalent Softkeys	**Calibration > AutoCal > Characterization**

SENS:CORR:COLL:LOAD

```
SENSe<Ch>:CORRection:COLLect[:ACQuire]:LOAD <port>
```

Description	Measures the calibration data of the load standard for the specified port (no query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<port> the number of the port from 1 to 2
Out of Range	Error occurs. The command is ignored.
Notes	The command start the measurement for the channel independently of the trigger and trigger source settings. The command waits for the completion of the measurement.
Equivalent Softkeys	**Calibration > Calibrate > Response (Open) > Load (Optional)** **Calibration > Calibrate > Response (Short) > Load (Optional)** **Calibration > Calibrate > Full 1–Port Cal > Load** **Calibration > Calibrate > One Path 2–Port Cal > Load**

SENS:CORR:COLL:METH:ERES

```
SENSe<Ch>:CORRection:COLLect:METHod:ERESponse <rcvport>,
<srcport>
```

Description	Selects the ports and sets the one path 2-port calibration type for the calculation of the calibration coefficients on completion of the calibration executed by the `SENS:CORR:COLL:SAVE` command (no query).
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<rcvport>`, the number of the receiver port from 1 to 2 `<srcport>`, the number of the source port = 1
Out of Range	If an incorrect port number is specified, an error occurs. If the same port numbers are specified, an error occurs. The command is ignored.
Related Commands	`SENS:CORR:COLL:SAVE`
Equivalent Softkeys	**None**

SENS:CORR:COLL:METH:OPEN

```
SENSe<Ch>:CORRection:COLLect:METHod[:RESPonse]:OPEN <port>
```

Description	Selects the port and sets the response calibration (Open) type for the calculation of the calibration coefficients on completion of the calibration executed by the `SENS:CORR:COLL:SAVE` command (no query).
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<port>` the number of the port from 1 to 2
Out of Range	Error occurs. The command is ignored.
Related Commands	`SENS:CORR:COLL:SAVE`
Equivalent Softkeys	**None**

SENS:CORR:COLL:METH: SHOR

```
SENSe<Ch>:CORRection:COLLect:METHod[:RESPonse]:SHORt <port>
```

Description	Selects the port and sets the response calibration (Short) type for the calculation of the calibration coefficients on completion of the calibration executed by the `SENS:CORR:COLL:SAVE` command (no query).
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<port>` the number of the port from 1 to 2
Out of Range	Error occurs. The command is ignored.
Related Commands	`SENS:CORR:COLL:SAVE`
Equivalent Softkeys	**None**

SENS:CORR:COLL:METH:SOLT1

```
SENSe<Ch>:CORRection:COLLect:METHod:SOLT1 <port>
```

Description	Selects the port and sets the full 1-port calibration type for the calculation of the calibration coefficients on completion of the calibration executed by the `SENS:CORR:COLL:SAVE` command (no query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<port> the number of the port from 1 to 2
Out of Range	Error occurs. The command is ignored.
Related Commands	`SENS:CORR:COLL:SAVE`
Equivalent Softkeys	**None**

SENS:CORR:COLL:METH:THRU

```
SENSe<Ch>:CORRection:COLLect:METHod[:RESPonse]:THRU <rcvport>,
<srcport>
```

Description	Selects the ports and sets the response calibration (Thru) type for the calculation of the calibration coefficients on completion of the calibration executed by the `SENS:CORR:COLL:SAVE` command (no query).
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Parameter	`<rcvport>`, the number of the receiver port from 1 to 2 `<srcport>`, the number of the source port = 1
Out of Range	If an incorrect port number is specified, an error occurs. If the same port numbers are specified, an error occurs. The command is ignored.
Related Commands	`SENS:CORR:COLL:SAVE`
Equivalent Softkeys	**None**

SENS:CORR:COLL:METH:TYPE?

`SENSe<Ch>:CORRection:COLLect:METHod:TYPE?`

Description	Reads out the calibration type selected for the calculation of the calibration coefficients on completion of the calibration executed by the `SENS:CORR:COLL:SAVE` command (query only).
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Query Response	RESPO" : Response (Open) RESPS : Response (Short) RESPT : Response (Thru) SOLT1 : Full 1–port calibration 1PATH : One path 2–port calibration NONE : Not defined
Equivalent Softkeys	**None**

SENS:CORR:COLL:OPEN

```
SENSe<Ch>:CORRection:COLLect[:ACQuire]:OPEN <port>
```

Description	Measures the calibration data of the open standard for the specified port (no query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<port> the number of the port from 1 to 2
Out of Range	Error occurs. The command is ignored.
Notes	The command start the measurement for the channel independently of the trigger and trigger source settings. The command waits for the completion of the measurement.
Equivalent Softkeys	**Calibration > Calibrate > Response (Open) > Open** **Calibration > Calibrate > Full 1–Port Cal > Open** **Calibration > Calibrate > One Path 2–Port Cal > Open**

SENS:CORR:COLL:SAVE

`SENSe<Ch>:CORRection:COLLect:SAVE`

Description	Calculates the calibration coefficients from the calibration standards measurements depending on the selected calibration type. On completion of the command, all the calibration standards measurements are cleared and the error correction automatically turns ON. Before executing this command it is neccessery to select calibration type by one of commands `SENS:CORR:COLL:METH:XXXX`. At the attempt to execute this command before all the needed standards are measured, an error occurs and the command is ignored (no query).
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`

Related Commands	Calibration type selection:
	`SENS:CORR:COLL:METH:OPEN`
	`SENS:CORR:COLL:METH:SHOR`
	`SENS:CORR:COLL:METH:THRU`
	`SENS:CORR:COLL:METH:ERES`
	`SENS:CORR:COLL:METH:SOLT1`
	
	Calibration standards measurement:
	`SENS:CORR:COLL:LOAD`
	`SENS:CORR:COLL:OPEN`
	`SENS:CORR:COLL:SHOR`
	`SENS:CORR:COLL:THRU`

Equivalent Softkeys	**Calibration > Calibrate > { Response (Open) | Response (Short) | Response (Thru) | One Path 2–Port Cal | Full 1–Port Cal } > Apply**

SENS:CORR:COLL:SHOR

```
SENSe<Ch>:CORRection:COLLect[:ACQuire]:SHORt <port>
```

Description	Measures the calibration data of the short standard for the specified port (no query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<port> the number of the port from 1 to 2
Out of Range	Error occurs. The command is ignored.
Notes	The command start the measurement for the channel independently of the trigger and trigger source settings. The command waits for the completion of the measurement.
Equivalent Softkeys	**Calibration > Calibrate > Response (Short) > Short** **Calibration > Calibrate > Full 1–Port Cal > Short** **Calibration > Calibrate > One Path 2–Port Cal > Short**

SENS:CORR:COLL:THRU

```
SENSe<Ch>:CORRection:COLLect[:ACQuire]:THRU <rcvport>,
<srcport>
```

Description	Measures the calibration data of the thru standard between the receiver port <rcvport> and the source port <srcport> (no query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<rcvport>, the number of the receiver port from 1 to 2 <srcport>, the number of the source port = 1
Out of Range	If an incorrect port number is specified, an error occurs. If the same port numbers are specified, an error occurs. The command is ignored.
Notes	The command start the measurement for the channel independently of the trigger and trigger source settings. The command waits for the completion of the measurement.
Equivalent Softkeys	**Calibration > Calibrate > Response (Thru) > Thru** **Calibration > Calibrate > One Path 2–Port Cal > Thru**

SENS:CORR:EXT

```
SENSe<Ch>:CORRection:EXTension[:STATe] {ON|OFF|1|0}

SENSe<Ch>:CORRection:EXTension[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the port extension function (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Port extension function ON {OFF|0} Port extension function OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Calibration > Port Extensions > Extension**

SENS:CORR:EXT:PORT:FREQ

```
SENSe<Ch>:CORRection:EXTension:PORT<Pt>:FREQuency{[1]|2}
<frequency>

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:FREQuency{[1]|2}?
```

Description	Sets or reads out the values of the frequency 1 and frequency 2 to calculate the loss for the port extension function (command/query).
Target	Port <Pt> of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Pt>={[1]|2}
Parameter	<frequency> the frequency value from 3E5 to 1.3E9 (Planar – TR1300/1), from 2E4 to 4.8E9 (TR5048), from 2E4 to 3.0E9 (TR7530)
Unit	Hz (Hertz)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	1E9
Equivalent Softkeys	**Calibration > Port Extensions > Loss > { Freq1 | Freq2 }**

SENS:CORR:EXT:PORT:INCL

```
SENSe<Ch>:CORRection:EXTension:PORT<Pt>:INCLude{[1]|2}[:STATe]
{ON|OFF|1|0}

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:INCLude{[1]|2}
[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the compensation of the loss 1 and loss 2 for the port extension function (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Loss compensation ON {OFF|0} Loss compensation OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Calibration > Port Extensions > Loss > { Loss1 | Loss2 }**

SENS:CORR:EXT:PORT:LDC

```
SENSe<Ch>:CORRection:EXTension:PORT<Pt>:LDC <loss>

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:LDC?
```

Description	Sets or reads out the loss value at DC for the port extension function (command/query).
Target	Port <Pt> of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Pt>={[1]|2}
Parameter	<loss> the loss value from –200 to 200
Unit	dB (decibel)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**Calibration > Port Extensions > Loss > Loss at DC**

SENS:CORR:EXT:PORT:LOSS

```
SENSe<Ch>:CORRection:EXTension:PORT<Pt>:LOSS{[1]|2} <loss>

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:LOSS{[1]|2}?
```

Description	Sets or reads out the values of the loss 1 and loss 2 for the port extension function (command/query).
Target	Port <Pt> of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Pt>={[1]|2}
Parameter	<loss> the loss value from –200 to 200
Unit	dB (decibel)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**Calibration > Port Extensions > Loss > { Loss1 | Loss2 }**

SENS:CORR:EXT:PORT:TIME

```
SENSe<Ch>:CORRection:EXTension:PORT<Pt>:TIME <time>

SENSe<Ch>:CORRection:EXTension:PORT<Pt>:TIME?
```

Description	Sets or reads out the electrical delay value for the port extension function (command/query).
Target	Port <Pt> of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Pt>={[1]|2}
Parameter	<time> the electrical delay value from –10 to 10
Unit	s (second)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**Calibration > Port Extensions > { Extension Port1 | Extension Port2 }**

SENS:CORR:IMP

```
SENSe:CORRection:IMPedance[:INPut][:MAGNitude] <impedance>

SENSe:CORRection:IMPedance[:INPut][:MAGNitude]?
```

Description	Sets or reads out the system impedance Z0 (command/query)
Parameter	`<impedance>` the Z0 value from 0.001 to 1000
Unit	Ω (Ohm)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	`<numeric>`
Preset Value	`50`
Equivalent Softkeys	**Calibration > System Z0**

SENS:CORR:STAT

```
SENSe<Ch>:CORRection:STATe {ON|OFF|1|0}

SENSe<Ch>:CORRection:STATe?
```

Description	Sets or reads out the ON/OFF state of the error correction (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} Error correction ON {OFF|0} Error correction OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Calibration > Correction**

SENS:CORR:TRAN:TIME:FREQ

```
SENSe<Ch>:CORRection:TRANsform:TIME:FREQuency <frequency>

SENSe<Ch>:CORRection:TRANsform:TIME:FREQuency?
```

Description	Sets or reads out the frequency value at which the cable loss specified for the cable correction function, when the time domain transformation function is turned ON. (command/query)
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<frequency> the frequency value.
Unit	Hz (Hertz)
Query Response	<numeric>
Preset Value	1 GHz
Equivalent Softkeys	**Analysis > Time Domain > Cable Correction > Frequency**

SENS:CORR:TRAN:TIME:LOSS

```
SENSe<Ch>:CORRection:TRANsform:TIME:LOSS <numeric>

SENSe<Ch>:CORRection:TRANsform:TIME:LOSS?
```

Description	Sets or reads out the cable loss value for the cable correction function, when the time domain transformation function is turned ON. (command/query)
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> the cable loss value.
Unit	dB/m (decibell / meter)
Query Response	<numeric>
Preset Value	0 dB/m
Equivalent Softkeys	**Analysis > Time Domain > Cable Correction > Cable Loss**

SENS:CORR:TRAN:TIME:RVEL

```
SENSe<Ch>:CORRection:TRANsform:TIME:RVELocity <numeric>

SENSe<Ch>:CORRection:TRANsform:TIME:RVELocity?
```

Description	Sets or reads out the cable relative wave speed velocity for the cable correction function, when the time domain transformation function is turned ON. (command/query)
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> the cable velocity factor.
Query Response	<numeric>
Preset Value	1.0
Equivalent Softkeys	**Analysis > Time Domain > Cable Correction > Velocity Factor**

SENS:CORR:TRAN:TIME:STAT

```
SENSe<Ch>:CORRection:TRANsform:TIME:STATe <bool>

SENSe<Ch>:CORRection:TRANsform:TIME:STATe?
```

Description	Turns ON/OFF the cable correction when the time domain transformation function is turned ON. (command/query)
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	Specifies state of the cable correction: {ON|1} : ON {OFF|0} : OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Analysis > Time Domain > Cable Correction > Cable Correction**

SENS:CORR:TYPE?

```
SENSe<Ch>:CORRection:TYPE<Tr>?
```

Description	Reads out the applied calibration type and the port numbers for the specified trace (query only).
Target	Trace <Tr> of channel <Ch>, <Tr>={[1]|2|3|4|5|6|7|8} <Ch>={[1]|2|3|4|5|6|7|8|9}
Query Response	{RESPO|RESPS|RESPT|SOLT1|SOLT2|1PATH|NONE}, <srcport>,<rcvport> Where: RESPO : Response (Open) RESPS : Response (Short) RESPT : Response (Thru) SOLT1 : Full 1–port calibration 1PATH : One path 2–port calibration NONE : Not defined <rcvport>, the number of the receiver port from 1 to 2 <srcport>, the number of the source port = 1
Equivalent Softkeys	**None**

SENS:FREQ

```
SENSe<Ch>:FREQuency[:CW] <frequency>

SENSe<Ch>:FREQuency[:FIXed] <frequency>

SENSe<Ch>:FREQuency[:CW]?

SENSe<Ch>:FREQuency[:FIXed]?
```

Description	Sets or reads out he fixed frequency for the power sweep (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<frequency> the CW frequency value from 3E5 to 1.3E9 (Planar – TR1300/1), from 2E4 to 4.8E9 (TR5048), from 2E4 to 3.0E9 (TR7530).
Unit	Hz (Hertz)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	6.5015E8 (Planar – TR1300/1), 2.40001E9 (TR5048), 1.50001E9 (TR7530).
Equivalent Softkeys	**Stimulus > Power > CW Freq**

SENS:FREQ:DATA?

```
SENSe<Ch>:FREQuency:DATA?
```

Description	Reads out the array of the measurement points frequency (query only). The array size is N, where N is the number of measurement points. For the n–th point, where n from 1 to N: `<numeric n>` the frequency value at the n–th measurement point
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Query Response	`<numeric 1>, <numeric 2>, …<numeric N>` The data transfer format depends on the `FORM:DATA` command setting.
Related Commands	`FORM:DATA`
Equivalent Softkeys	**None**

SENS:FREQ:CENT

```
SENSe<Ch>:FREQuency:CENTer <frequency>

SENSe<Ch>:FREQuency:CENTer?
```

Description	Sets or reads out the stimulus center value of the sweep range for linear or logarithmic sweep type (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<frequency> the stimulus center value from 3E5 to 1.3E9 (Planar – TR1300/1), from 2E4 to 4.8E9 (TR5048), from 2E4 to 3.0E9 (TR7530).
Unit	Hz (Hertz)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	6.5015E8 (Planar - TR1300/1), 2.40001E9 (TR5048), 1.50001E9 (TR7530)
Equivalent Softkeys	**Stimulus > Center**

SENS:FREQ:SPAN

```
SENSe<Ch>:FREQuency:SPAN <frequency>

SENSe<Ch>:FREQuency:SPAN?
```

Description	Sets or reads out the stimulus span value of the sweep range for linear or logarithmic sweep type (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<frequency> the stimulus span value from 0 to 1.2997E9 (Planar – TR1300/1), from 0 to 4.79998E9 (TR5048), from 0 to 2.99998E9 (TR7530).
Unit	Hz (Hertz)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	1.2997E9 (Planar – TR1300/1), 4.79998E9 (TR5048), 2.99998E9 (TR7530).
Equivalent Softkeys	**Stimulus > Span**

SENS:FREQ:STAR

```
SENSe<Ch>:FREQuency:STARt <frequency>

SENSe<Ch>:FREQuency:STARt?
```

Description	Sets or reads out the stimulus start value of the sweep range for linear or logarithmic sweep type (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<frequency> the stimulus start value from 3E5 to 1.3E9 (Planar – TR1300/1), from 2E4 to 4.8E9 (TR5048), from 2E4 to 3.0E9 (TR7530).
Unit	Hz (Hertz)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	3E5 (Planar – TR1300/1), 2E4 (TR5048), 2E4 (TR7530).
Equivalent Softkeys	**Stimulus > Start**

SENS:FREQ: STOP

```
SENSe<Ch>:FREQuency:STOP <frequency>

SENSe<Ch>:FREQuency:STOP?
```

Description	Sets or reads out the stimulus stop value of the sweep range for linear or logarithmic sweep type (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<frequency> the stimulus stop value form 3E5 to 1.3E9 (Planar – TR1300/1), from 2E4 to 4.8E9 (TR5048), from 2E4 to 3.0E9 (TR7530).
Unit	Hz (Hertz)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	1.3E9 (Planar – TR1300/1), 4.8E9 (TR5048), 3.0E9 (TR7530).
Equivalent Softkeys	**Stimulus > Stop**

SENS:OFFS

```
SENSe<Ch>:OFFSet[:STATe] {ON|OFF|1|0}

SENSe<Ch>:OFFSet[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the frequency offset feature (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} frequency offset ON {OFF|0} frequency offset OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Stimulus > Freq. Offset > Freq. Offset**

SENS:OFFS:ADJ

```
SENSe<Ch>:OFFSet:ADJust[:STATe] {ON|OFF|1|0}

SENSe<Ch>:OFFSet:ADJust[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the frequency offset adjust feature (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} frequency offset adjust ON {OFF|0} frequency offset adjust OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Stimulus > Freq. Offset > Offset Adjust > Offset Adjust**

SENS:OFFS:ADJ:CONT

```
SENSe<Ch>:OFFSet:ADJust:CONTinuous[:STATe] {ON|OFF|1|0}

SENSe<Ch>:OFFSet:ADJust:CONTinuous[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the continuous frequency offset adjust feature (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	{ON|1} continuous frequency offset adjust ON {OFF|0} continuous frequency offset adjust OFF
Query Response	{0|1}
Preset Value	0
Equivalent Softkeys	**Stimulus > Freq. Offset > Offset Adjust > Contin. Adjust**

SENS:OFFS:ADJ:CONT:PER

```
SENSe<Ch>:OFFSet:ADJust:CONTinuous:PERiod <numeric>

SENSe<Ch>:OFFSet:ADJust:CONTinuous:PERiod?
```

Description	Sets or reads out the period for continuous offset adjust feature (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> the period for continuous offset adjust feature: 1 – 3 sec, 2 – 10 sec, 3 – 30 sec, 4 – 100 sec, 5 – 300 sec,
Unit	n/a
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	3
Equivalent Softkeys	**Stimulus > Freq. Offset > Offset Adjust > Adjust Period**

SENS:OFFS:ADJ:EXEC

```
SENSe<Ch>:OFFSet:ADJust:EXECute
```

Description	Executes the offset adjust when the frequency offset feature is ON (no query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Equivalent Softkeys	**Stimulus > Freq. Offset > Offset Adjust > Adjust Immediate**

SENS:OFFS:ADJ:PORT

```
SENSe<Ch>:OFFSet:ADJust:PORT <numeric>

SENSe<Ch>:OFFSet:ADJust:PORT?
```

Description	Sets or reads out the port number for frequency offset adjust feature (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> port number 1 to 2;
Unit	n/a
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	1
Equivalent Softkeys	**Stimulus > Freq. Offset > Offset Adjust > Select Port**

SENS:OFFS:ADJ:VAL

```
SENSe<Ch>:OFFSet:ADJust:VALue <hertz>

SENSe<Ch>:OFFSet:ADJust:VALue?
```

Description	Sets or reads out the frequency offset adjust value of the offset adjust feature (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<hertz> adjust value of the offset adjust feature from –5E5 to 5E5;
Unit	Hz
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**Stimulus > Freq. Offset > Offset Adjust > Adjust Value**

SENS:OFFS:PORT:DATA?

```
SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:DATA?
```

Description	Reads out the array of the frequency points of port <Pt> when the frequency offset feature is ON and offset type is "PORT" (query only). The array size is N, where N is the number of measurement points. For the n−th point, where n from 1 to N: `<numeric n>` the frequency value at the n−th point

Target	Port <Pt> of channel <Ch>,
	`<Ch>={[1]|2|3|4|5|6|7|8|9}`
	`<Pt>={[1]|2}`

Query Response	`<numeric 1>, <numeric 2>, …<numeric N>` The data transfer format depends on the `FORM:DATA` command setting.

Related Commands	`FORM:DATA`
	`SENS:OFFS:STAT`
	`SENS:OFFS:TYPE`

Equivalent Softkeys	**None**

SENS:OFFS:PORT:DIV

```
SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:DIVisor <numeric>

SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:DIVisor?
```

Description	Sets or reads out the basic frequency range divisor of port <Pt> when the frequency offset feature is ON and offset type is "PORT" (command/query).
Target	Port <Pt> of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Pt>={[1]|2}
Parameter	<numeric> divisor from 1 to 1000;
Unit	n/a
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	1
Related Commands	SENS:OFFS:STAT SENS:OFFS:TYPE
Equivalent Softkeys	**Stimulus > Freq. Offset > Port n > Divider**

SENS:OFFS:PORT:MULT

```
SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:MULTiplier <numeric>

SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:MULTiplier?
```

Description	Sets or reads out the basic frequency range multiplier of port <Pt> when the frequency offset feature is ON and offset type is "PORT" (command/query).
Target	Port <Pt> of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Pt>={[1]|2}
Parameter	<numeric> multiplier from –1000 to 1000;
Unit	n/a
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	1
Related Commands	SENS:OFFS:STAT SENS:OFFS:TYPE
Equivalent Softkeys	**Stimulus > Freq. Offset > Port n > Multiplier**

SENS:OFFS:PORT:OFFS

```
SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:OFFSet <hertz>

SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:OFFSet?
```

Description	Sets or reads out the basic frequency range offset of port <Pt> when the frequency offset feature is ON and offset type is "PORT" (command/query).
Target	Port <Pt> of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Pt>={[1]|2}
Parameter	<hertz> offset from –1e12 to 1e12;
Unit	Hz
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	0
Related Commands	SENS:OFFS:STAT SENS:OFFS:TYPE
Equivalent Softkeys	**Stimulus > Freq.Offset > Port n > Offset**

SENS:OFFS:PORT:STAR

```
SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:STARt <hertz>

SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:STARt?
```

Description	Sets or reads out the frequency sweep start of port <Pt> when the frequency offset feature is ON and offset type is "PORT" (command/query).
Target	Port <Pt> of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Pt>={[1]|2}
Unit	Hz
Query Response	<numeric>
Related Commands	SENS:OFFS:STAT SENS:OFFS:TYPE
Equivalent Softkeys	**Stimulus > Freq. Offset > Port n > Start**

SENS:OFFS:PORT:STOP

```
SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:STOP <hertz>

SENSe<Ch>:OFFSet:PORT<Pt>[:FREQuency]:STOP?
```

Description	Sets or reads out the frequency sweep stop of port <Pt> when the frequency offset feature is ON and offset type is "PORT" (command/query).
Target	Port <Pt> of channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9} <Pt>={[1]|2}
Unit	Hz
Query Response	<numeric>
Related Commands	SENS:OFFS:STAT SENS:OFFS:TYPE
Equivalent Softkeys	**Stimulus > Freq.Offset > Port n > Stop**

SENS:ROSC:SOUR

```
SENSe<Ch>:ROSCillator:SOURce <char>

SENSe<Ch>:ROSCillator:SOURce?
```

Description	Sets or reads out the internal or external source of the reference frequency of 10 MHz (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<char> choose from: INTernal : Internal source of the reference frequency EXTernal : External source of the reference frequency
Out of Range	Error occurs. The command is ignored.
Query Response	{INT|EXT}
Preset Value	INT
Equivalent Softkeys	**System > Misc Setup > Ref Source**

SENS:SEGM:DATA

```
SENSe<Ch>:SEGMent:DATA <numeric list>

SENSe<Ch>:SEGMent:DATA?
```

Description	Sets or reads out the array of the segment sweep table (command/query). The array has the following format: {<Buf>, <Flag1>, <Flag2>, <Flag3>, <Flag4>, <Flag5>, <N>, <Start 1>, <Stop 1>, <NOP 1> [,<IFBW 1>] [,<Pow 1>] [,<Del 1>] [,<Time 1>], <Start 2>, <Stop 2>, <NOP 2> [,<IFBW 2>] [,<Pow 2>] [,<Del 2>] [,<Time 2>], … <StartN>, <StopN>, <NOP N> [,<IFBW N>] [,<Pow N>] [,<Del N>] [,<TimeN>]} <Buf> Always 5, <Flag1> Stimulus start setting (0 – start/stop, 1 – center/span), <Flag2> Setting of the <IFBW> field (0 – disabled, 1 – enabled), <Flag3> Setting of the <Pow> field (0 – disabled, 1 – enabled), <Flag4> Setting of the <Del> field (0 – disabled, 1 – enabled), <Flag5> Setting of the <Time> field (0 – disabled, 1 – enabled), <N> Number of segments, <Start n> Start value of the n–th segment, <Stop n> Stop value of the n–th segment, <NOP n> Number of points of the n–th segment, <IFBW n> IF bandwidth of the n–th segment (if enabled), <Pow n> Power of the n–th segment (if enabled), <Del n> Measurement delay of the n–th segment (if enabled), <Time n> Reserved for future use (if enabled)
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Query Response	<numeric 1>,<numeric 2>,…<numeric 7+M×N> Where, N – the number of the segments, M – depends on the values of the flags M = 3 + <Flag2> + <Flag3> + <Flag4> + <Flag5>
Equivalent Softkeys	**Stimulus > Segment Table**

SENS:SWE:POIN

```
SENSe<Ch>:SWEep:POINts <numeric>

SENSe<Ch>:SWEep:POINts?
```

Description	Sets or reads out the number of measurement points (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> the number of measurement points from 2 to 16001 (Planar – TR1300/1), from 2 to 100001 (TR5048), from 2 to 100001 (TR7530).
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	201
Equivalent Softkeys	**Stimulus > Points**

SENS:SWE:POIN:TIME

```
SENSe<Ch>:SWEep:POINt:TIME <time>

SENSe<Ch>:SWEep:POINt:TIME?
```

Description	Sets or reads out the delay before measurement in each measurement point (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<time> the measurement delay value from 0 to 0.3
Resolution	5E−6
Unit	s (second)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**Stimulus > Meas Delay**

SENS:SWE:TYPE

```
SENSe<Ch>:SWEep:TYPE <char>

SENSe<Ch>:SWEep:TYPE?
```

Description	Sets or reads out the sweep type (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<char> specifies the sweep type: LINear : Linear frequency sweep LOGarithmic : Logarithmic frequency sweep SEGMent : Segment frequency sweep POWer : Power sweep VVM : Vector Voltmeter mode
Out of Range	Error occurs. The command is ignored.
Query Response	{LIN|LOG|SEGM|POW}
Preset Value	LIN
Equivalent Softkeys	**Stimulus > Sweep Type**

SENS:VVM:TYPE

```
SENSe<Ch>:VVM:TYPE <char>

SENSe<Ch>:VVM:TYPE?
```

Description	Sets or reads out the smeasured parameter in Vector Voltmeter mode.
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<char> choose from: S11 : Reflection parameter S21 : Transmission parameter A/B : The ratio of the receivers signals A to B, an external frequency generator is used. B/A : The ratio of the receivers signals B to A, an external frequency generator is used.
Out of Range	Error occurs. The command is ignored..
Query Response	{S11|S21|A/B|B/A}
Preset Value	S11
Equivalent Softkeys	**Stimulus > Vector Voltmeter > Measurement > {S11|S21|A/B|B/A}**

SENS:VVM:FORM

```
SENSe<Ch>:VVM:FORMat <char>

SENSe<Ch>:VVM:FORMat?
```

Description	Sets or reads out the data format in Vector Voltmeter mode.
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<char> choose from: LOGP : Log magnitude and phase LINP : Linear magnitude and phase SWR : Voltage standing wave ratio IMP : Impedance value
Out of Range	Error occurs. The command is ignored..
Query Response	{LOGP|LINP|SWR|IMP}
Preset Value	LOGP
Equivalent Softkeys	**Stimulus > Vector Voltmeter > Format > {LOGP | LINP | SWR | IMP}**

SENS:VVM:FREQ

```
SENSe<Ch>:VVM:FREQuency[:CW] <numeric>

SENSe<Ch>:VVM:FREQuency[:CW]?
```

Description	Sets or reads out the operating frequency in the Vector Voltmeter mode.
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> the frequency value.
Unit	Hz (Hertz)
Query Response	<numeric>
Equivalent Softkeys	**Stimulus > Vector Voltmeter > Frequency**

SENS:VVM:DATA?

```
SENSe<Ch>:VVM:DATA?
```

Description	Reads out the value of the vector voltmeter measurement (query only). If a reference value is specified, the return value is read relatively to it. The data include 2 elements: `<numeric 1>` Magnitude value, depending on the data format or SWR value; `<numeric 2>` Phase value or zero in SWR format.
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Query Response	`<numeric 1>, <numeric 2>`

Related Commands	`SENSe<Ch>:VVM:FORMat`
	`SENSe<Ch>:VVM:REFerence:MEMorize`
	`SENSe<Ch>:VVM:REFerence:CLEar`

Equivalent Softkeys	**None**

SENS:VVM:REF:CLE

```
SENSe<Ch>:VVM:REFerence:CLEar
```

Description	Clears the reference value of the vector voltmeter .
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Equivalent Softkeys	**Stimulus > Vector Voltmeter > Clear Reference**

SENS:VVM:REF:MEM

```
SENSe<Ch>:VVM:REFerence:MEMorize
```

Description	Memorizes the reference value of the vector voltmeter .
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Equivalent Softkeys	**Stimulus > Vector Voltmeter > Save Reference**

SENS:VVM:REF:DATA?

```
SENSe<Ch>:VVM:REFerence:DATA?
```

Description	Reads out the reference value of the vector voltmeter measurement (query only). The data include 2 elements: <numeric 1> Magnitude value, depending on the data format or SWR value; <numeric 2> Phase value or zero in SWR format.
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Query Response	<numeric 1>, <numeric 2>
Related Commands	SENSe<Ch>:VVM:FORMat SENSe<Ch>:VVM:REFerence:MEMorize SENSe<Ch>:VVM:REFerence:CLEar
Equivalent Softkeys	**None**

SENS:VVM:TABL:CLE

```
SENSe<Ch>:VVM:TABLe:CLEar
```

Description	Clears all data from the vector voltmeter measurement table.
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Related Commands	SENSe<Ch>:VVM:TABLe:MEMorize SENSe<Ch>:VVM:TABLe:INSert SENSe<Ch>:VVM:TABLe:REMove
Equivalent Softkeys	**Stimulus > Vector Voltmeter > Data Table > Clear Table**

SENS:VVM:TABL:MEM

```
SENSe<Ch>:VVM:TABLe:MEMorize
```

Description	Saves measurements of the vector voltmeter to the table.
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Related Commands	SENSe<Ch>:VVM:TABLe:MEMorize SENSe<Ch>:VVM:TABLe:INSert SENSe<Ch>:VVM:TABLe:REMove
Equivalent Softkeys	**Stimulus > Vector Voltmeter > Data Table > Add Data**

SENS:VVM:TABL:INS

```
SENSe<Ch>:VVM:TABLe:MEMorize
```

Description	Inserts a row of data into the table with the current measurement of a vector voltmeter `<numeric>` The row number in the table where new data will be inserted.
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`

Related Commands	`SENSe<Ch>:VVM:TABLe:MEMorize`
	`SENSe<Ch>:VVM:TABLe:INSert`
	`SENSe<Ch>:VVM:TABLe:REMove`

Equivalent Softkeys	**Stimulus > Vector Voltmeter > Data Table > Insert Data**

SENS:VVM:TABL:REM

```
SENSe<Ch>:VVM:TABLe:REMove <numeric>
```

Description	Removes a table row with a vector voltmeter measurement index `<numeric>` The row number in the table which will be removed.
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`

Related Commands	`SENSe<Ch>:VVM:TABLe:CLEar`
	`SENSe<Ch>:VVM:TABLe:MEMorize`
	`SENSe<Ch>:VVM:TABLe:INSert`

Equivalent Softkeys	**Stimulus > Vector Voltmeter > Data Table > Remove Data**

SENS:VVM:TABL:SAVE

```
SENSe<Ch>:VVM:TABLe:SAVE <string>
```

Description	Saves the table into CSV file.
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	`<string>` File name (quoted string)
Notes	If the full path of the file is not specified, the \CSV subdirectory of the main directory will be searched for the file. The file has *.csv extension by default.
Equivalent Softkeys	**Stimulus > Vector Voltmeter > Data Table > Save Table to CSV**

SENS:VVM:TABL:DATA?

`SENSe<Ch>:VVM:TABLe:DATA`

Description	Returns data table of the vector voltmeter (query only). The array has the following format: {<Flag1>, <N>, <Abs.Mag 1>, <Abs.Phase 1> [,<Rel.Mag 1>] [,<Rel.Phase 1>], … <Abs.Mag N>, <Abs.Phase N> [,<Rel.Mag N>] [,<Rel.Phase N>]}, Where: <Flag1> reference value, 0 – is not specified, 1 – is specified, N – rows number in the data table; <Abs.Mag i>, <Abs.Phase i> - measurement data of a vector voltmeter; [,<Rel.Mag i>] [,<Rel.Phase i>] - measurement data relative to the reference value (if specified).
Target	Channel <Ch>, `<Ch>={[1]|2|3|4|5|6|7|8|9}`
Query Response	`<numeric 1>,<numeric 2>,…<numeric 2+N×(2+M×2)>` Where: N – number of table rows, M – flag value Flag1

Related Commands	`SENSe<Ch>:VVM:TABLe:CLEar`
	`SENSe<Ch>:VVM:TABLe:MEMorize`
	`SENSe<Ch>:VVM:TABLe:REMove`
	`SENSe<Ch>:VVM:TABLe:INSert`

Equivalent Softkeys	**None**

SERV:CHAN:ACT?

```
SERVice:CHANnel:ACTive?
```

Description	Reads out the active channel number (query only).
Query Response	<numeric> from 1 to 16
Equivalent Softkeys	**None**

SERV:CHAN:COUN?

```
SERVice:CHANnel:COUNt?
```

Description	Reads out the maximum number of the channels (query only).
Query Response	<numeric>
Equivalent Softkeys	**None**

SERV:CHAN:TRAC:ACT?

```
SERVice:CHANnel<Ch>:TRACe:ACTive?
```

Description	Read out the active trace number of the channel (query only).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Query Response	<numeric> from 1 to 16
Equivalent Softkeys	**None**

SERV:CHAN:TRAC:COUN?

```
SERVice:CHANnel:TRACe:COUNt?
```

Description	Reads out the maximum number of the traces in the channel (query only).
Query Response	<numeric>
Equivalent Softkeys	**None**

SERV:PORT:COUN?

```
SERVice:PORT:COUNt?
```

Description	Reads out the number of the ports (query only).
Query Response	<numeric>
Equivalent Softkeys	**None**

SERV:SWE:FREQ:MAX?

```
SERVice:SWEep:FREQuency:MAXimum?
```

Description	Reads out the upper limit of the measurement frequency (query only).
Query Response	<numeric>
Unit	Hz (Hertz)
Equivalent Softkeys	**None**

SERV:SWE:FREQ:MIN?

```
SERVice:SWEep:FREQuency:MINimum?
```

Description	Reads out the lower frequency of the measurement frequency (query only).
Query Response	<numeric>
Unit	Hz (Hertz)
Equivalent Softkeys	**None**

SERV:SWE:POIN?

```
SERVice:SWEep:POINts?
```

Description	Reads out the maximum number of the measurement points (query only).
Query Response	<numeric>
Equivalent Softkeys	**None**

SOUR:POW

```
SOURce<Ch>:POWer[:LEVel][:IMMediate][:AMPLitude] <power>

SOURce<Ch>:POWer[:LEVel][:IMMediate][:AMPLitude]?
```

Description	Sets or reads out the power level for the frequency sweep (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<power> the power level from −55 to +3
Resolution	0.05
Unit	dBm (decibels above 1 milliwatt)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**Stimulus > Power > Output Power**

SOUR:POW:CENT

```
SOURce<Ch>:POWer:CENTer <power>

SOURce<Ch>:POWer:CENTer?
```

Description	Sets or reads out the center value of the power sweep range (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<power> the power sweep center value from −55 to +3
Resolution	0.025
Unit	dBm (decibels above 1 milliwatt)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	−22.5
Equivalent Softkeys	**Stimulus > Center**

SOUR:POW:SLOP

```
SOURce<Ch>:POWer[:LEVel]:SLOPe[:DATA] <power>

SOURce<Ch>:POWer[:LEVel]:SLOPe[:DATA]?
```

Description	Sets or reads out the power slope value for the frequency sweep (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<power> the power slope value from –2 to +2
Resolution	0.1
Unit	dB/GHz (decibel/gigahertz)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**Stimulus > Power > Power Slope [dB/GHz]**

SOUR:POW:SPAN

```
SOURce<Ch>:POWer:SPAN <power>

SOURce<Ch>:POWer:SPAN?
```

Description	Sets or reads out the power span for the power sweep (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<power> the power sweep span value from 0 to 58
Resolution	0.05
Unit	dBm (decibels above 1 milliwatt)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	58
Equivalent Softkeys	**Stimulus > Span**

SOUR:POW:STAR

```
SOURce<Ch>:POWer:STARt <power>

SOURce<Ch>:POWer:STARt?
```

Description	Sets or reads out the power sweep start for the power sweep (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<power> the power sweep start from −55 to +3
Resolution	0.05
Unit	dBm (decibels above 1 milliwatt)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	−55
Equivalent Softkeys	**Stimulus > Start**

SOUR:POW:STOP

```
SOURce<Ch>:POWer:STOP <power>

SOURce<Ch>:POWer:STOP?
```

Description	Sets or reads out the power sweep stop for the power sweep (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<power> the power sweep stop from –55 to +3
Resolution	0.05
Unit	dBm (decibels above 1 milliwatt)
Out of Range	Sets the value of the limit, which is closer to the specified value.
Query Response	<numeric>
Preset Value	3
Equivalent Softkeys	**Stimulus > Stop**

STAT:OPER?

```
STATus:OPERation[:EVENt]?
```

Description	Reads out the value of the Operation Status Event Register (query only).
Target	Status Reporting System
Query Response	`<numeric>`
Equivalent Softkeys	**None**

STAT:OPER:COND?

```
STATus:OPERation:CONDition?
```

Description	Reads out the value of the Operation Status Condition Register (query only).
Target	Status Reporting System
Query Response	`<numeric>`
Equivalent Softkeys	**None**

STAT:OPER:ENAB

```
STATus:OPERation:ENABle <numeric>

STATus:OPERation:ENABle?
```

Description	Sets or reads out the value of the Operation Status Enable Register (command/query).
Target	Status Reporting System
Parameter	`<numeric>` from 0 to 65535
Out of Range	Bit–to–bit AND with numeric 65535
Query Response	`<numeric>`
Preset Value	`0`
Equivalent Softkeys	**None**

STAT:OPER:NTR

```
STATus:OPERation:NTRansition <numeric>

STATus:OPERation:NTRansition?
```

Description	Sets or reads out the value of the Negative transition filter of the Operation Status Register (command/query).
Target	Status Reporting System
Parameter	`<numeric>` from 0 to 65535
Out of Range	Bit–to–bit AND with numeric 65535
Query Response	`<numeric>`
Preset Value	`0`
Equivalent Softkeys	**None**

STAT:OPER:PTR

```
STATus:OPERation:PTRansition <numeric>

STATus:OPERation:PTRansition?
```

Description	Sets or reads out the value of the Positive transition filter of the Operation Status Register (command/query).
Target	Status Reporting System
Parameter	`<numeric>` from 0 to 65535
Out of Range	Bit–to–bit AND with numeric 65535
Query Response	`<numeric>`
Preset Value	`65535`
Equivalent Softkeys	**None**

STAT:PRES

```
STATus:PRESet
```

Description	Resets all the status registers to the factory settings (no query).
Target	Status Reporting System
Query Response	`<numeric>`
Equivalent Softkeys	**None**

STAT:QUES:COND?

```
STATus:QUEStionable:CONDition?
```

Description	Reads out the value of the Questionable Status Condition Register (query only).
Target	Status Reporting System
Query Response	<numeric>
Equivalent Softkeys	**None**

STAT:QUES:ENAB

```
STATus:QUEStionable:ENABle <numeric>

STATus:QUEStionable:ENABle?
```

Description	Sets or reads out the value of the Questionable Status Enable Register (command/query).
Target	Status Reporting System
Parameter	<numeric> from 0 to 65535
Out of Range	Bit–to–bit AND with numeric 65535
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**None**

STAT:QUES:LIM:CHAN:COND?

```
STATus:QUEStionable:LIMit:CHANnel<Ch>:CONDition?
```

Description	Reads out the value of the Questionable Limit Channel Status Condition Register (query only).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Query Response	<numeric>
Equivalent Softkeys	**None**

STAT:QUES:LIM:CHAN:ENAB

```
STATus:QUEStionable:LIMit:CHANnel<Ch>:ENABle <numeric>

STATus:QUEStionable:LIMit:CHANnel<Ch>:ENABle?
```

Description	Sets or reads out the value of the Questionable Limit Channel Status Enable Register (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> from 0 to 65535
Out of Range	Bit–to–bit AND with numeric 65535
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**None**

STAT:QUES:LIM:CHAN:NTR

```
STATus:QUEStionable:LIMit:CHANnel<Ch>:NTRansition <numeric>

STATus:QUEStionable:LIMit:CHANnel<Ch>:NTRansition?
```

Description	Sets or reads out the value of the Negative transition filter of the Questionable Limit Channel Status Register (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> from 0 to 65535
Out of Range	Bit–to–bit AND with numeric 65535
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**None**

STAT:QUES:LIM:CHAN:PTR

```
STATus:QUEStionable:LIMit:CHANnel<Ch>:PTRansition <numeric>

STATus:QUEStionable:LIMit:CHANnel<Ch>:PTRansition?
```

Description	Sets or reads out the value of the Positive transition filter of the Questionable Limit Channel Status Register (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> from 0 to 65535
Out of Range	Bit–to–bit AND with numeric 65535
Query Response	<numeric>
Preset Value	65535
Equivalent Softkeys	**None**

STAT:QUES:LIM:CHAN?

```
STATus:QUEStionable:LIMit:CHANnel<Ch>[:EVENt]?
```

Description	Reads out the value of the Questionable Limit Channel Status Event Register (query only).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Query Response	<numeric>
Equivalent Softkeys	**None**

STAT:QUES:LIM:COND?

```
STATus:QUEStionable:LIMit:CONDition?
```

Description	Reads out the value of the Questionable Limit Status Condition Register (query only).
Target	Status Reporting System
Query Response	<numeric>
Equivalent Softkeys	**None**

STAT:QUES:LIM:ENAB

```
STATus:QUEStionable:LIMit:ENABle <numeric>

STATus:QUEStionable:LIMit:ENABle?
```

Description	Sets or reads out the value of the Questionable Limit Status Enable Register (command/query).
Target	Status Reporting System
Parameter	`<numeric>` from 0 to 65535
Out of Range	Bit–to–bit AND with numeric 65535
Query Response	`<numeric>`
Preset Value	`0`
Equivalent Softkeys	**None**

STAT:QUES:LIM:NTR

```
STATus:QUEStionable:LIMit:NTRansition <numeric>

STATus:QUEStionable:LIMit:NTRansition?
```

Description	Sets or reads out the value of the Negative transition filter of the Questionable Limit Status Register (command/query).
Target	Status Reporting System
Parameter	`<numeric>` from 0 to 65535
Out of Range	Bit–to–bit AND with numeric 65535
Query Response	`<numeric>`
Preset Value	`0`
Equivalent Softkeys	**None**

STAT:QUES:LIM:PTR

```
STATus:QUEStionable:LIMit:PTRansition <numeric>

STATus:QUEStionable:LIMit:PTRansition?
```

Description	Sets or reads out the value of the Positive transition filter of the Questionable Limit Status Register (command/query).
Target	Status Reporting System
Parameter	`<numeric>` from 0 to 65535
Out of Range	Bit–to–bit AND with numeric 65535
Query Response	`<numeric>`
Preset Value	`65535`
Equivalent Softkeys	**None**

STAT:QUES:LIM?

```
STATus:QUEStionable:LIMit[:EVENt]?
```

Description	Reads out the value of the Questionable Limit Status Event Register (query only).
Target	Status Reporting System
Query Response	`<numeric>`
Equivalent Softkeys	**None**

STAT:QUES:NTR

```
STATus:QUEStionable:NTRansition <numeric>

STATus:QUEStionable:NTRansition?
```

Description	Sets or reads out the value of the Negative transition filter of the Questionable Status Register (command/query).
Target	Status Reporting System
Parameter	`<numeric>` from 0 to 65535
Out of Range	Bit–to–bit AND with numeric 65535
Query Response	`<numeric>`
Preset Value	`0`
Equivalent Softkeys	**None**

STAT:QUES:PTR

```
STATus:QUEStionable:PTRansition <numeric>

STATus:QUEStionable:PTRansition?
```

Description	Sets or reads out the value of the Positive transition filter of the Questionable Status Register (command/query).
Target	Status Reporting System
Parameter	`<numeric>` from 0 to 65535
Out of Range	Bit–to–bit AND with numeric 65535
Query Response	`<numeric>`
Preset Value	`65535`
Equivalent Softkeys	**None**

STAT:QUES:RLIM:CHAN:COND?

```
STATus:QUEStionable:RLIMit:CHANnel<Ch>:CONDition?
```

Description	Reads out the value of the Questionable Ripple Limit Channel Status Condition Register (query only).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Query Response	<numeric>
Equivalent Softkeys	**None**

STAT:QUES:RLIM:CHAN:ENAB

```
STATus:QUEStionable:RLIMit:CHANnel<Ch>:ENABle <numeric>

STATus:QUEStionable:RLIMit:CHANnel<Ch>:ENABle?
```

Description	Sets or reads out the value of the Questionable Ripple Limit Channel Status Enable Register (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> from 0 to 65535
Out of Range	Bit–to–bit AND with numeric 65535
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**None**

STAT:QUES:RLIM:CHAN:NTR

```
STATus:QUEStionable:RLIMit:CHANnel<Ch>:NTRansition <numeric>

STATus:QUEStionable:RLIMit:CHANnel<Ch>:NTRansition?
```

Description	Sets or reads out the value of the Negative transition filter of the Questionable Ripple Limit Channel Status Register (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> from 0 to 65535
Out of Range	Bit–to–bit AND with numeric 65535
Query Response	<numeric>
Preset Value	0
Equivalent Softkeys	**None**

STAT:QUES:RLIM:CHAN:PTR

```
STATus:QUEStionable:RLIMit:CHANnel<Ch>:PTRansition <numeric>

STATus:QUEStionable:RLIMit:CHANnel<Ch>:PTRansition?
```

Description	Sets or reads out the value of the Positive transition filter of the Questionable Ripple Limit Channel Status Register (command/query).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Parameter	<numeric> from 0 to 65535
Out of Range	Bit–to–bit AND with numeric 65535
Query Response	<numeric>
Preset Value	65535
Equivalent Softkeys	**None**

STAT:QUES:RLIM:CHAN?

```
STATus:QUEStionable:RLIMit:CHANnel<Ch>[:EVENt]?
```

Description	Reads out the value of the Questionable Ripple Limit Channel Status Event Register (query only).
Target	Channel <Ch>, <Ch>={[1]|2|3|4|5|6|7|8|9}
Query Response	<numeric>
Equivalent Softkeys	**None**

STAT:QUES:RLIM:COND?

```
STATus:QUEStionable:RLIMit:CONDition?
```

Description	Reads out the value of the Questionable Ripple Limit Status Condition Register (query only).
Target	Status Reporting System
Query Response	`<numeric>`
Equivalent Softkeys	**None**

STAT:QUES:RLIM:ENAB

```
STATus:QUEStionable:RLIMit:ENABle <numeric>

STATus:QUEStionable:RLIMit:ENABle?
```

Description	Sets or reads out the value of the Questionable Ripple Limit Status Enable Register (command/query).
Target	Status Reporting System
Parameter	`<numeric>` from 0 to 65535
Out of Range	Bit–to–bit AND with numeric 65535
Query Response	`<numeric>`
Preset Value	`0`
Equivalent Softkeys	**None**

STAT:QUES:RLIM:NTR

```
STATus:QUEStionable:RLIMit:NTRansition <numeric>

STATus:QUEStionable:RLIMit:NTRansition?
```

Description	Sets or reads out the value of the Negative transition filter of the Questionable Ripple Limit Status Register (command/query).
Target	Status Reporting System
Parameter	`<numeric>` from 0 to 65535
Out of Range	Bit–to–bit AND with numeric 65535
Query Response	`<numeric>`
Preset Value	`0`
Equivalent Softkeys	**None**

STAT:QUES:RLIM:PTR

```
STATus:QUEStionable:RLIMit:PTRansition <numeric>

STATus:QUEStionable:RLIMit:PTRansition?
```

Description	Sets or reads out the value of the Positive transition filter of the Questionable Ripple Limit Status Register (command/query).
Target	Status Reporting System
Parameter	`<numeric>` from 0 to 65535
Out of Range	Bit–to–bit AND with numeric 65535
Query Response	`<numeric>`
Preset Value	`65535`
Equivalent Softkeys	**None**

STAT:QUES:RLIM?

```
STATus:QUEStionable:RLIMit[:EVENt]?
```

Description	Reads out the value of the Questionable Ripple Limit Status Event Register (query only).
Target	Status Reporting System
Query Response	`<numeric>`
Equivalent Softkeys	**None**

STAT:QUES?

```
STATus:QUEStionable[:EVENt]?
```

Description	Reads out the value of the Questionable Status Event Register (query only).
Target	Status Reporting System
Query Response	<numeric>
Equivalent Softkeys	**None**

SYST:COMM:ECAL:TEMP:SENS?

```
SYSTem:COMMunicate:ECAL:TEMPerature:SENSor?
```

Description	Reads out the temperature of the AutoCal module connected to the Analyzer. (query only)
Target	AutoCal module
Unit	°C
Query Response	<numeric>
Equivalent Softkeys	**None**

SYST:CORR

```
SYSTem:CORRection[:STATe] {ON|OFF|1|0}

SYSTem:CORRection[:STATe]?
```

Description	Sets or reads out the ON/OFF state of the system error correction (command/query).
Parameter	`{ON|1}` System error correction ON `{OFF|0}` System error correction OFF
Query Response	`{0|1}`
Preset Value	`1`
Equivalent Softkeys	**System > Misc Setup > System Correction**

SYST:DATE

```
SYSTem:DATE <numeric 1>,<numeric 2>,<numeric 3>

SYSTem:DATE?
```

Description	Sets or reads out the current date (command/query).
Parameter	`<numeric 1>` year from 1900 to 2100; `<numeric 2>` month from 1 to 12; `<numeric 3>` day from 1 to 31.
Query Response	`<numeric 1>, <numeric 2>, <numeric 3>`
Equivalent Softkeys	**None**

SYST:ERR?

```
SYSTem:ERRor[:NEXT]?
```

Description	Reads out the error message of the error occurred when executing the SCPI commands, from the FIFO (First In First Out) error queue stored in the instrument. The read out error is deleted from the error queue. The *CLS command clears the error queue. The maximum size of the queue is 100 messages (query only).
Query Response	`<numeric>, <string>` Where: `<numeric>` error code `<string>` error message If there is no error in the queue, "0, No error" is read out.
Equivalent Softkeys	**None**

SYST:LOC

```
SYSTem:LOCal
```

Description	Sets the instrument to the local operation mode, when all the keys on the front panel, mouse and the touch screen are active (no query).
Related Commands	`SYST:REM` `SYST:RWL`
Equivalent Softkeys	**None**

SYST:PRES

```
SYSTem:PRESet
```

Description	Resets the instrument to the factory settings (no query).
Notes	The difference from the*RST: command is that the trigger is set to the Continuous trigger mode.
Related Commands	`*RST`
Equivalent Softkeys	**System > Preset > Apply**

SYST: READy?

```
SYSTem:READy[:STATe]?
```

Description	Reads out the ready state of the Analyzer. The state is *True* when analyzer hardware is connected, powered and the boot process is completed (about 10 sec). (query only)
Query Response	`{0|1}`
Equivalent Softkeys	**None**

SYST:REM

```
SYSTem:REMote
```

Description	Sets the instrument to the remote operation mode, when all the keys on the front panel, mouse and the touch screen are not active, except for one key labeled Return to Local. Pushing this button will reset the instrument to the local operation mode (no query).
Related Commands	`SYST:LOC` `SYST:RWL`
Equivalent Softkeys	**None**

SYST:RWL

```
SYSTem:RWLock
```

Description	Sets the instrument to the remote operation mode, when all the keys on the front panel, mouse and the touch screen are not active. Only `SYST:LOC` or `SYST:REM` command can release this remote operation mode (no query).
Related Commands	`SYST:LOC` `SYST:REM`
Equivalent Softkeys	**None**

SYST:TEMP:SENS?

```
SYSTem:TEMPerature:SENSor? <numeric>
```

Description	Reads out the specified sensor temperature inside the Analyzer. (query only)
Target	Analyzer
Parameter	`<numeric>` : Sensor number
Unit	°C
Query Response	`<numeric>`
Equivalent Softkeys	**None**

SYST:TIME

```
SYSTem:TIME <numeric 1>,<numeric 2>,<numeric 3>

SYSTem:TIME?
```

Description	Sets or reads out the current time (command/query).
Parameter	`<numeric 1>` hours from 0 to 23; `<numeric 2>` minutes from 0 to 59; `<numeric 3>` seconds from 0 to 59.
Query Response	`<numeric 1>, <numeric 2>, <numeric 3>`
Equivalent Softkeys	**None**

SYST:TERM

```
SYSTem:TERMinate
```

Description	Terminates the analyzer software. (no query)
Equivalent Softkeys	**System > Exit > Ok**

TRIG

```
TRIGger[:SEQuence][:IMMediate]
```

Description	Generates a trigger, independently of the trigger source setting (except for the *External*). If the trigger source is set to *External*, an error occurs and the command is ignored. If the instrument is not in the waiting for a trigger state (sweep is in progress or all the channels are set to *Hold*), an error occurs and the command is ignored. The command is completed before the end of the sweep. (no query)

Related Commands	`TRIG:SOUR`
	`INIT:CONT`
	`INIT:IMM`

Equivalent Softkeys	**None**

TRIG:SING

```
TRIGger[:SEQuence]:SINGle
```

Description	Generates a trigger, independently of the trigger source setting (except for the External). If the trigger source is set to *External*, an error occurs and the command is ignored. If the instrument is not in the waiting for a trigger state (sweep is in progress or all the channels are set to *Hold*), an error occurs and the command is ignored. The command is not completed before the end of the sweep (waiting for the completion of the sweep of all the channels). (no query)

Related Commands	`TRIG:SOUR`
	`INIT:CONT`
	`INIT:IMM`

Equivalent Softkeys	**None**

TRIG:SOUR

```
TRIGger[:SEQuence]:SOURce <char>

TRIGger[:SEQuence]:SOURce?
```

Description	Sets or reads out the sweep trigger source (command/query).
Parameter	<char> choose from: INTernal : Internal EXTernal : External MANual : Manual BUS : Bus
Out of Range	Error occurs. The command is ignored.
Query Response	{INT|EXT|MAN|BUS}
Preset Value	INT
Related Commands	TRIG TRIG:SING *TRG
Equivalent Softkeys	**Stimulus > Trigger > Trigger Source > { Internal | External | Bus }**

TRIG:STAT?

```
TRIGger[:SEQuence]:STATus?
```

Description	Reads out the the current state of the analyzer. (query only)
Parameter	HOLD scanning is stopped MEASure is scanning WAIT is scanning
Equivalent Softkeys	**None**

TRIG:WAIT

```
TRIGger[:SEQuence]:WAIT <char>
```

Description	Delays the execution by the analyzer of the next command till the specified state of the analyzer has been reached (see options below). A query that follows the `TRIG:WAIT` command blocks the execution of the user program till the specified state of the analyzer has been reached. The command can be used to wait for the end of the sweep initianed by the commands `TRIG, *TRG` or initiated by the external trigger signal. If the *continuous initiation mode* is turned ON by the command `INIT:CONT ON`, then the parameter of the command must be WAIT, otherwise HOLD. (no query)
Parameter	`<char>` choose from: HOLD : Waits for the *Hold* state MEASure : Waits for the *Measure* state WAIT : Waits for the *Trigger Waiting* state
Related Commands	`TRIG, *TRG` `TRIG:SOUR EXT`
Equivalent Softkeys	**None**

VVM:RES

```
VVM[:DISPlay]:RESolution <numeric>

VVM[:DISPlay]:RESolution?
```

Description	Sets or reads out number of indication symbols in the vector voltmeter mode
Parameter	Number of significant symbols from 3 to 12
Query Response	`<numeric>`
Preset value	`5`
Equivalent Softkeys	**Stimulus > Vector Voltmeter > Properties > Response Digits**

VVM:FONT

```
VVM[:DISPlay]:FONT[:SIZE] <numeric>

VVM[:DISPlay]:FONT[:SIZE]?
```

Description	Sets or reads out the font size for indication in the vector voltmeter mode
Parameter	Font size from 3 to 12
Query Response	`<numeric>`
Preset value	`36`
Equivalent Softkeys	**Stimulus > Vector Voltmeter > Properties > Font Size**

5 Programming Tips

This section gives recommendations for programming in certain specific situations.

5.1 Program Sweep Initiation and Waiting

The program sweep initiation and waiting for its completion the most simple method can be implemented by using commands `TRIG:SING` and `*OPC?`. The command `TRIG:SING` gerates a trigger signal and starts sweeping under the following conditions:

- The program trigger source is selected by command `TRIG:SOUR BUS`;
- The analyzer should be in the *trigger waiting* state, otherwise (the analyzer is sweeping or analyzer is in the *Hold* state) an error occurs and the command is ignored.

The transition of the analyzer to the *trigger waiting* state depends on the state of the *continuous initiation mode*, which is set by command `INIT:CONT`. Provided that the *continuous initiation mode* is ON, the analyzer automatically transits to the *trigger waiting* state when the program trigger source have been selected, and then each time at the end of a sweep. Provided that the *continuous initiation mode* is OFF, the analyzer transits to the *trigger waiting* state for single time upon receiving the command INIT.

The command `TRIG:SING` remains pending until the end of sweep. This allows use the `*OPC?` query for the waiting the end of sweep.

Example 1. Start sweeping in all channels and waits for its completion. The channels sweep one by one in turn. The *continuous initiation mode* is enabled by default.

`TRIG:SOUR BUS`	Selects the program trigger source and puts the analyzer to the *trigger waiting* state
<loop>:	
`TRIG:SING`	Starts sweep
`*OPC?`	Waits for the end of the sweep
`...`	

After the sweep completion the analyser returns to the *trigger waiting* state, and then the next sweep initiation command can be sended.

Example 2. Start sweeping in one channel and waits for its completion, then start sweeping in another channel and waits for its completion. The channel number must be set to 2.

`TRIG:SOUR BUS`	Selects the program trigger source
`INIT1:CONT OFF`	Puts channel 1 to the hold state
`INIT2:CONT OFF`	Puts channel 2 to the hold state
<loop>:	
`INIT1`	Puts channel 1 to the trigger waiting state
`TRIG:SING`	Starts sweep in channel 1
`*OPC?`	Waits for the end of the sweep
`...`	
`INIT2`	Puts channel 2 to the trigger waiting state
`TRIG:SING`	Starts sweep in channel 2
`*OPC?`	Waits for the end of the sweep
`...`	

After completion of the sweep on one channel the analyser returns to the *hold* state, then the sweep initiation for another channel is available.

5.2 External Trigger

If trigger source is set to the *External* by the command `TRIG:SOUR EXT`, the sweep starts at signal arrival on the external trigger input.

At the moment of a trigger signal arrival the Analyzer must be in the trigger waiting state, otherwise the signal is ignored but no error is detected.

As a rule the hardware trigger output is used to determine the end of the sweep. If there is a need to determine programmatically the end of the sweep, the TRIG:WAIT command can be used. Using program way to determine the end of the sweep requires the *continuous initiation mode* turned OFF.

Example 3. Starts sweep at the signal on the external trigger input. Programmatically waits for its completion. The *continuous initiation mode* must be turned OFF.

`TRIG:SOUR EXT`	Selects the external trigger source
`INIT:CONT OFF`	Puts channel 1 to the hold state
<loop>:	

`INIT`	Puts channel 1 to the *trigger waiting* state.
	The sweep starts at external trigger signal
`TRIG:WAIT HOLD`	Waits for the end of the sweep
`*OPC?`	Any query is required
`...`	

After completion of the sweep the analyser returns to the *hold* state, then the next sweep initiation is available.

5.3 Waiting for Calibration Commands

Depending on the sweep settings the calibration commands may have long execution time, as they start sweep and wait it completion. These commands are:

`SENS:CORR:COLL:XXXX`

`SENS:CORR:OFFS:COLL:XXXX`

`SENS:CORR:REC:COLL:XXXX`

`SENS:CORR:COLL:ECAL:XXXX`

`SENS:CORR:COLL:ECAL:ORI:EXEC`

The user program can determine the end of these command using any query, the `*OPC?` for example.

5.4 VISA Timeout Considerations

When using the query `*OPC?` for waiting an operation the VISA timeout value must be set to the value no less than the expected sweep time. For example in C/C++:

```
viSetAttribute(instr, VI_ATTR_TMO_VALUE, 5000);
```

5.5 Receiving Data Arrays in Text Form

By default the data from the analyzer is transmitted in a text form. The VISA library has built-in facilities for receiving an array of data from the analyzer. The example assumes that the size of the array is sufficient to receive a number of elements equal to twice the number of points.

An example of receiving a data array in C/C++:

```
double DataArray[1000];
ViUInt32 retCount;
...
```

```
retCount = sizeof(DataArray) / sizeof(double);
viQueryf(instr, "CALC:DATA:SDAT?\n", "%,#lf", &retCount, Data);
// retCount now contains the actual number of elements read
```

5.6 Receiving Data Arrays in Binary Form

The transfer of data from the analyzer in binary form reduces the amount of data transferred and the transmission time. The transfer of binary data is supported only by the HiSLIP protocol. To enable binary data transfer, use the `FORM: DATA` command. The list of commands that support the transfer of binary data is given in the description of the `FORM: DATA` command.

Binary data is transmitted as a block having a header followed by data. Block format:

#	8	<Data Size>	<Binary Data>

Where:

– the character '#';

8 – the character '8';

<Data Size> – 8 bytes, the symbolic representation of the number of bytes of binary data.

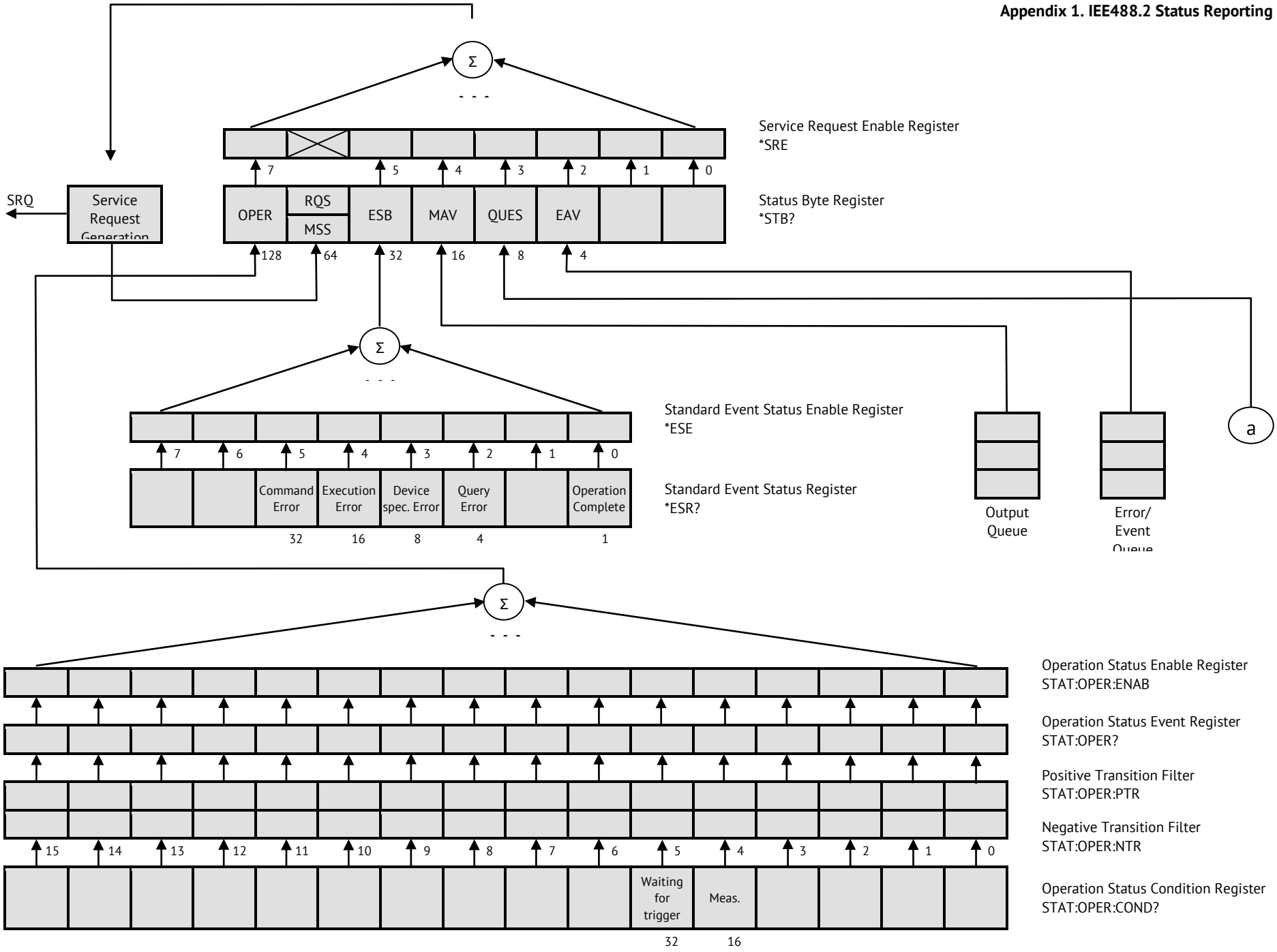
For example:

#800003216<Binary Data>

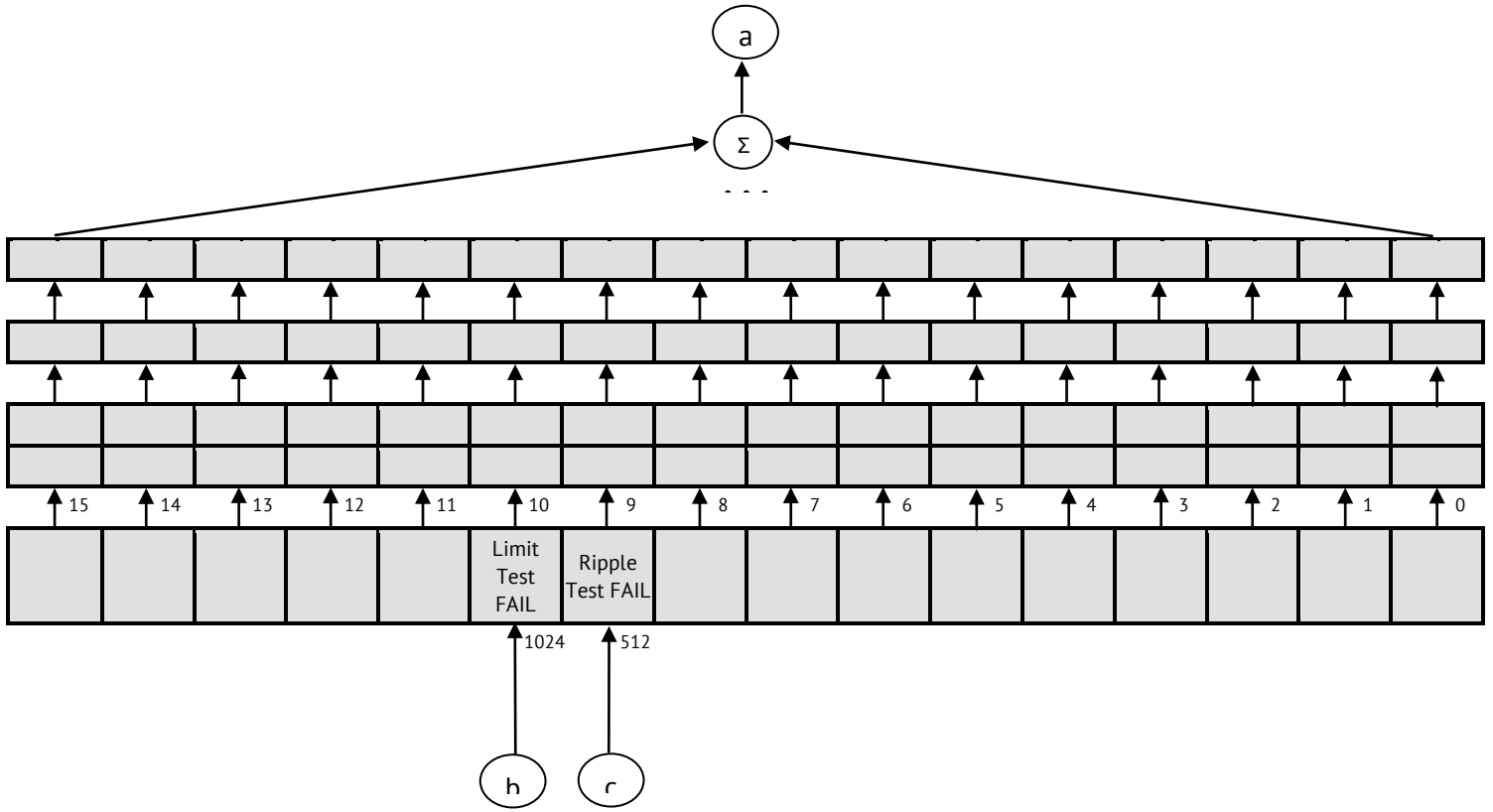
The VISA library has built-in tools for receiving binary data from the analyzer. The example assumes that the size of the array is sufficient to receive a number of elements equal to twice the number of points.

Example of receiving binary data array in C/C ++:

```
double DataArray[1000];
ViUInt32 retCount;
...
viPrintf(instr, "FORM:DATA REAL\n");
retCount = sizeof(DataArray) / sizeof(double);
viQueryf(instr, "CALC:DATA:FDAT?\n", "%#Zb", &retCount, DataArray);
// retCount now contains the actual number of elements read
```


Σ
Service Request Enable Register
*SRE
7 5 4 3 2 1 0
SRQ
Service Request Generation
OPER
RQS
MSS
ESB
MAV
QUES
EAV
Status Byte Register
*STB?
128 64 32 16 8 4
Σ
Standard Event Status Enable Register
*ESE
7 6 5 4 3 2 1 0
Command Error
Execution Error
Device spec. Error
Query Error
Operation Complete
Standard Event Status Register
*ESR?
32 16 8 4 1
Output Queue
Error/ Event Queue
a
Σ
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Waiting for trigger
Meas.
32 16
Operation Status Enable Register
STAT:OPER:ENAB
Operation Status Event Register
STAT:OPER?
Positive Transition Filter
STAT:OPER:PTR
Negative Transition Filter
STAT:OPER:NTR
Operation Status Condition Register
STAT:OPER:COND?

Appendix 1. IEE488.2 Status Reporting System



Questionable Status Enable Register
STAT:QUES:ENAB

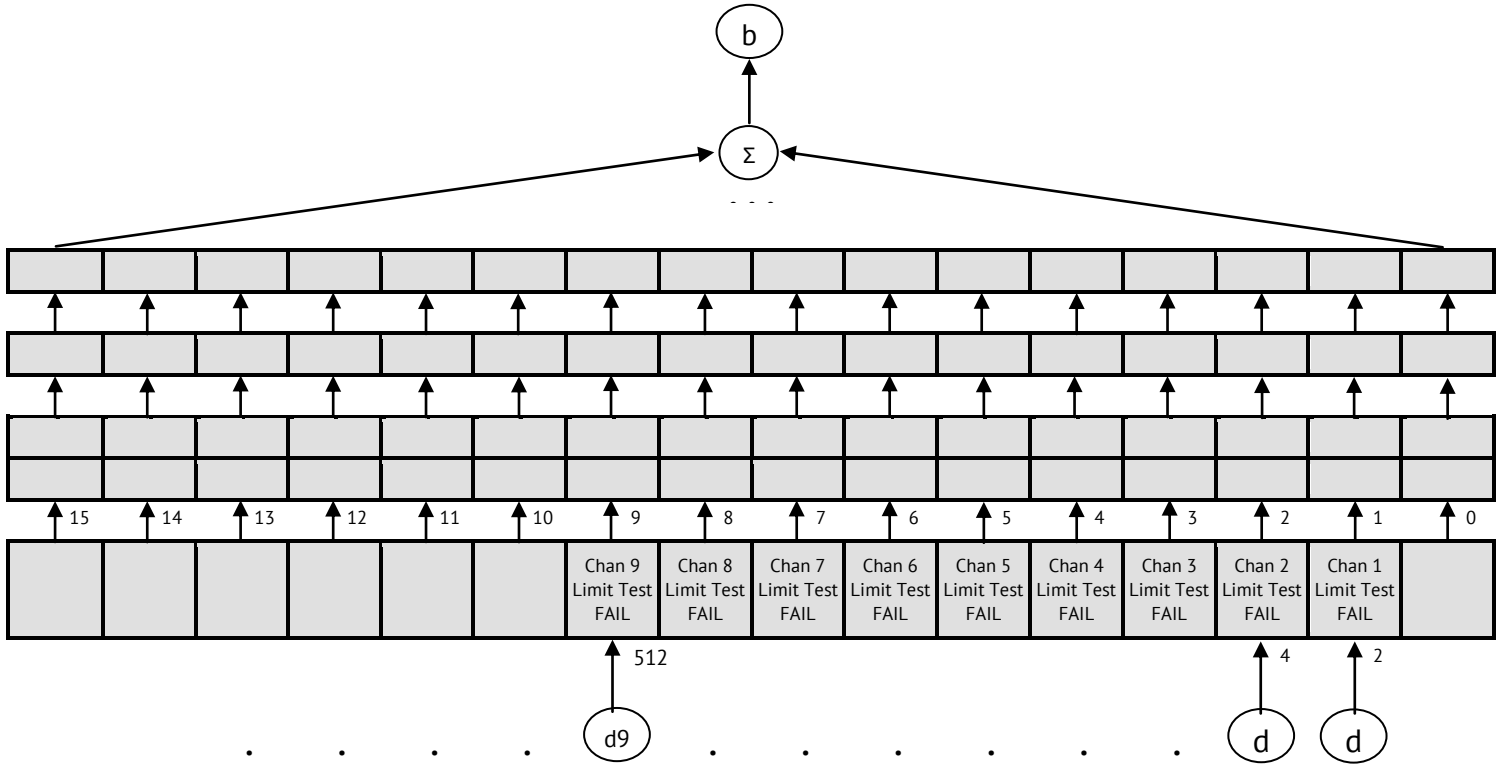
Questionable Status Event Register
STAT:QUES?

Positive Transition Filter
STAT:QUES:PTR

Negative Transition Filter
STAT:QUES:NTR

Questionable Status Condition Register
STAT:QUES:COND?

Appendix 1. IEE488.2 Status Reporting System



Questionable Limit Status Enable Register
STAT:QUES:LIM:ENAB

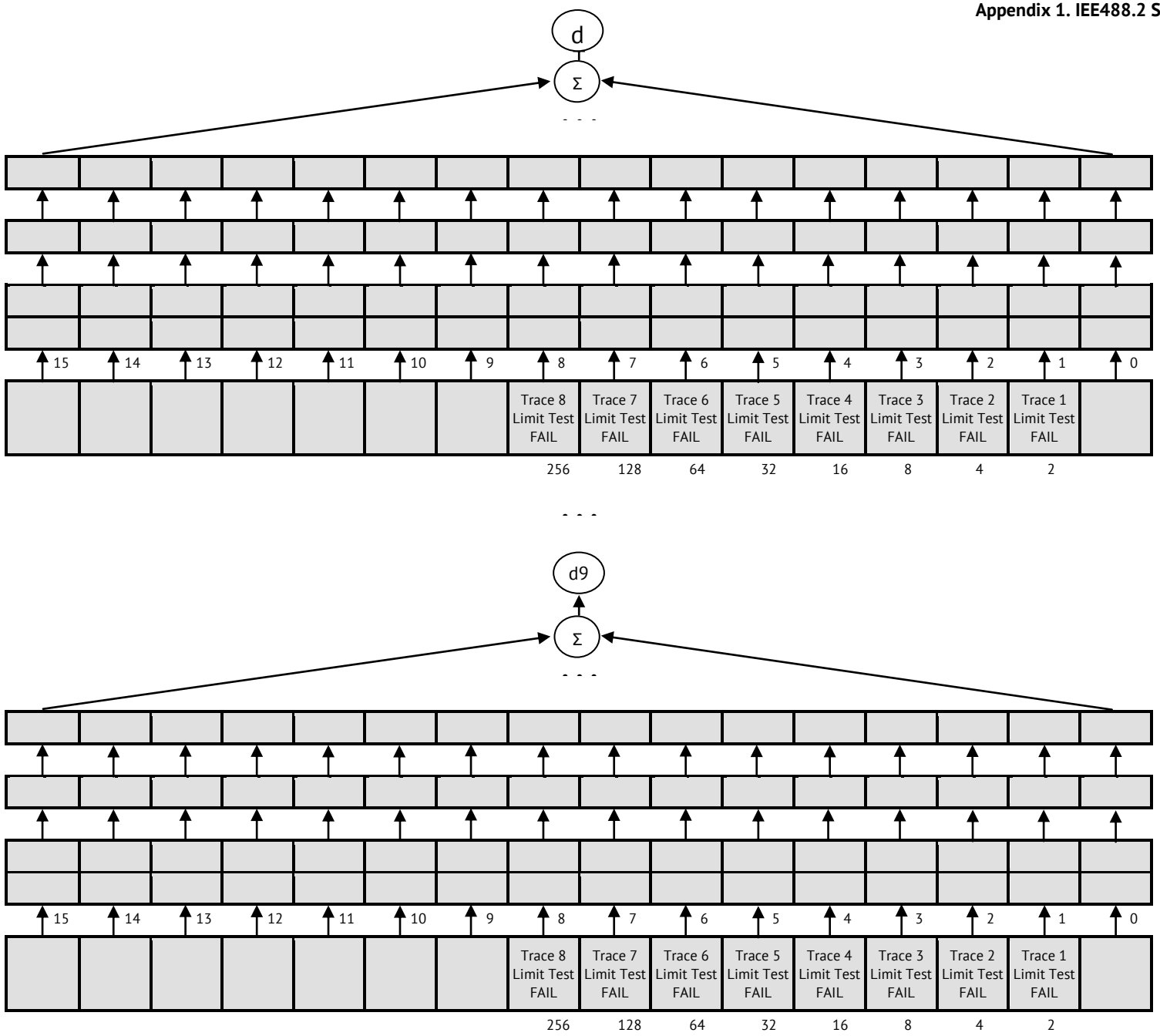
Questionable Limit Status Event Register
STAT:QUES:LIM?

Positive Transition Filter
STAT:QUES:LIM:PTR

Negative Transition Filter
STAT:QUES:LIM:NTR

Questionable Limit Status Condition Register
STAT:QUES:LIM:COND?

Appendix 1. IEE488.2 Status Reporting System



Questionable Limit Channel 1 Status Enable Register
STAT:QUES:LIM:CHAN1:ENAB

Questionable Limit Channel 1 Status Event Register
STAT:QUES:CHAN1:LIM2

Positive Transition Filter
STAT:QUES:LIM:CHAN1:PTR

Negative Transition Filter
STAT:QUES:LIM:CHAN1:NTR

Questionable Limit Channel 1 Status Condition Register
STAT:QUES:LIM:CHAN1:COND?

Questionable Limit Channel 14 Status Enable Register
STAT:QUES:LIM:CHAN14:ENAB

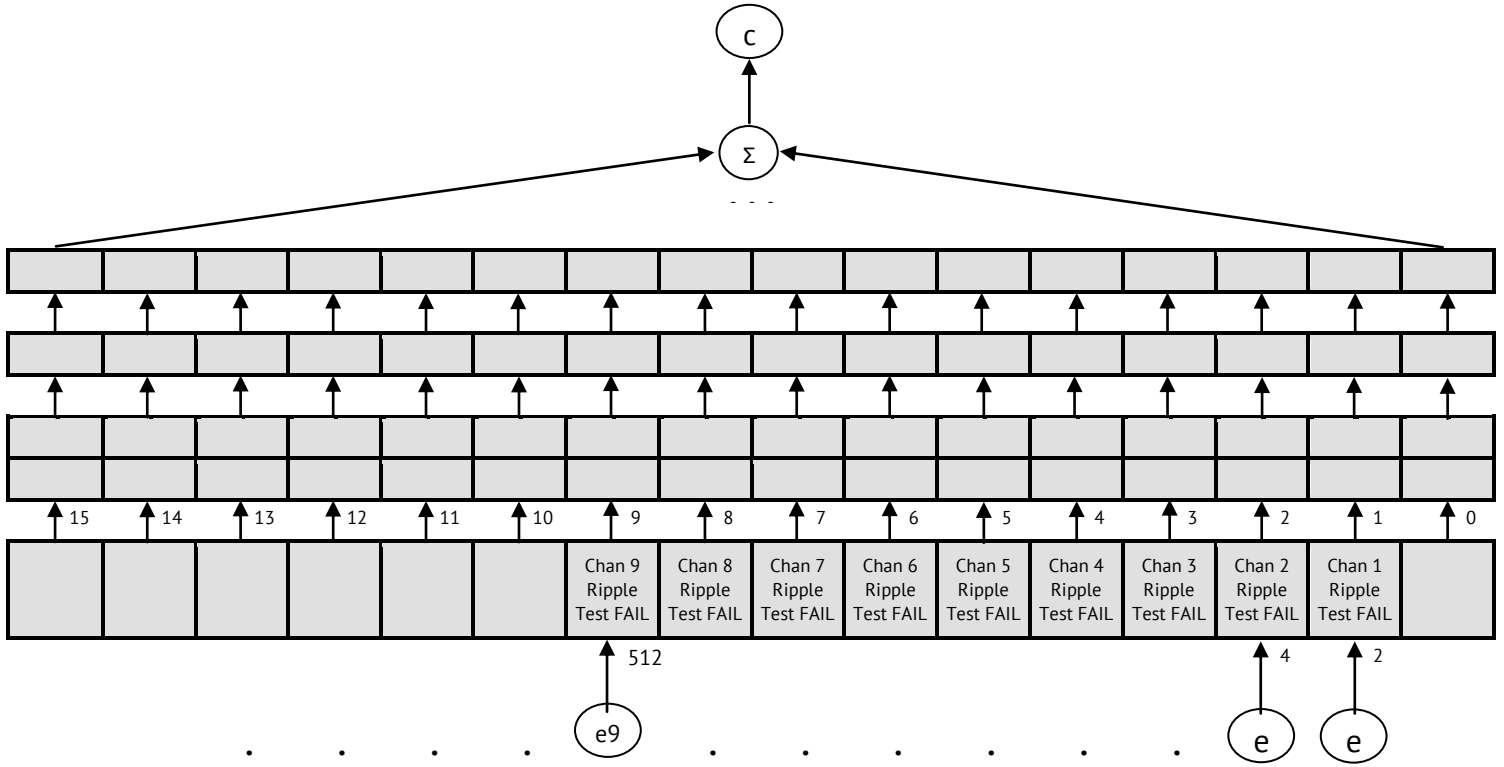
Questionable Limit Channel 9 Status Event Register
STAT:QUES:CHAN9:LIM2

Positive Transition Filter
STAT:QUES:LIM:CHAN9:PTR

Negative Transition Filter
STAT:QUES:LIM:CHAN9:NTR

Questionable Limit Channel 9 Status Condition Register
STAT:QUES:LIM:CHAN9:COND?

Appendix 1. IEE488.2 Status Reporting System



Questionable Ripple Limit Status Enable Register
STAT:QUES:RLIM:ENAB

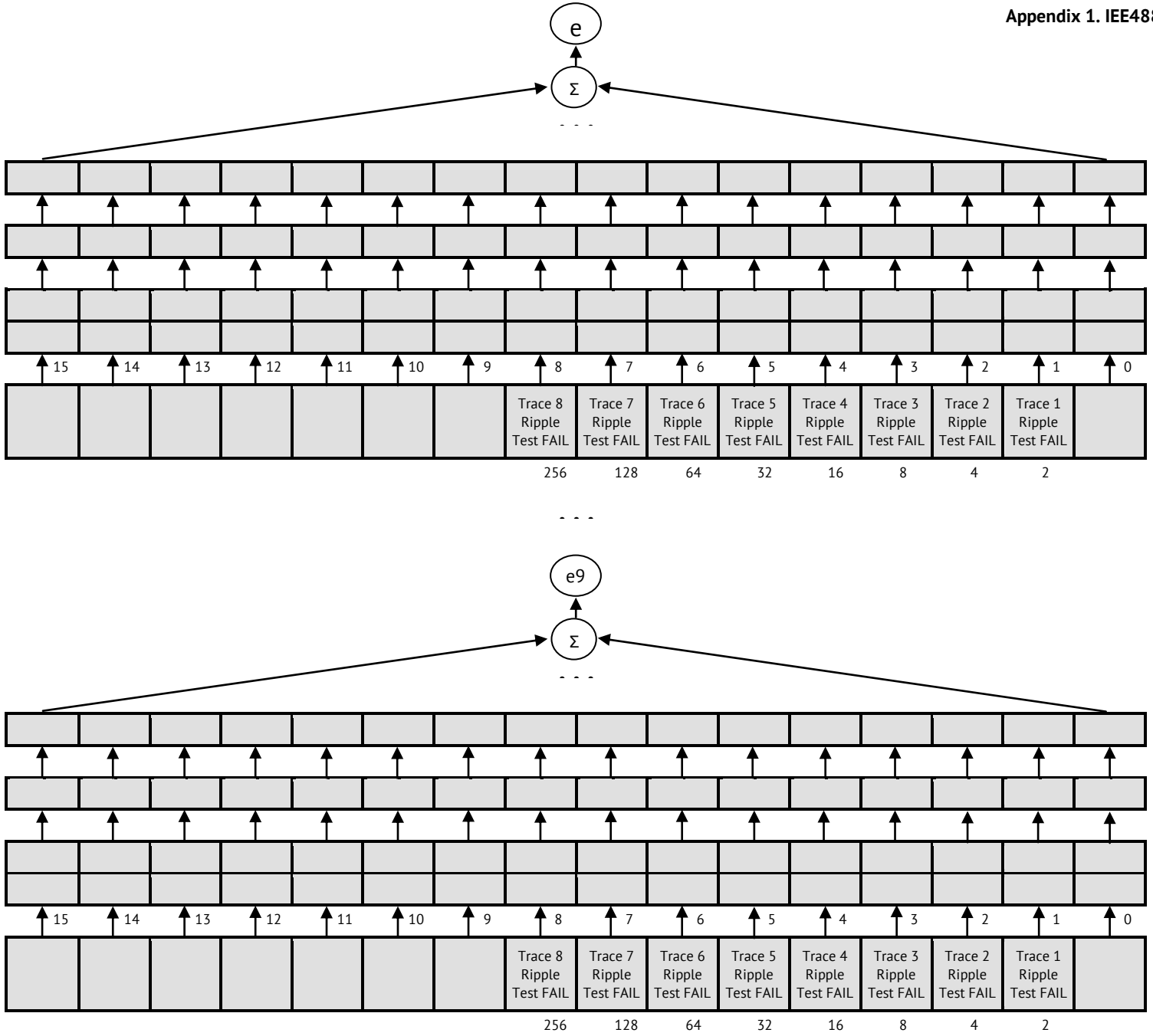
Questionable Ripple Limit Status Event Register
STAT:QUES:RLIM?

Positive Transition Filter
STAT:QUES:RLIM:PTR

Negative Transition Filter
STAT:QUES:RLIM:NTR

Questionable Ripple Limit Status Condition Register
STAT:QUES:RLIM:COND?

Appendix 1. IEE488.2 Status Reporting System



Questionable Ripple Limit Channel 1 Status Enable Register
STAT:QUES:RLIM:CHAN1:ENAB

Questionable Ripple Limit Channel 1 Status Event Register
STAT:QUES:CHAN1:RLIM2

Positive Transition Filter
STAT:QUES:RLIM:CHAN1:PTR

Negative Transition Filter
STAT:QUES:RLIM:CHAN1:NTR

Questionable Ripple Limit Channel 1 Status Condition Register
STAT:QUES:RLIM:CHAN1:COND?

Questionable Ripple Limit Channel 9 Status Enable Register
STAT:QUES:RLIM:CHAN9:ENAB

Questionable Ripple Limit Channel 9 Status Event Register
STAT:QUES:CHAN9:RLIM2

Positive Transition Filter
STAT:QUES:RLIM:CHAN9:PTR

Negative Transition Filter
STAT:QUES:RLIM:CHAN9:NTR

Questionable Ripple Limit Channel 9 Status Condition Register
STAT:QUES:RLIM:CHAN9:COND?

114	"Header suffix out of range"
200	"Execution error"
211	"Trigger ignored"
213	"Init ignored"
220	"Parameter Error"
222	"Data out of range"
224	"Illegal parameter value"
201	"Invalid channel index"
202	"Invalid trace index"
203	"Invalid marker index"
204	"Marker is not active"
205	"Invalid save type specifier"
206	"Invalid sweep type specifier"
207	"Invalid trigger source specifier"
208	"Invalid measurement parameter specifier"
209	"Invalid format specifier"
210	"Invalid data math specifier"
214	"Invalid limit data"
215	"Invalid segment data"
216	"Invalid standard type specifier"
217	"Invalid conversion specifier"
218	"Invalid gating shape specifier"
219	"Invalid gating type specifier"
300	"Device-specific error"
302	"Status reporting system error"

Example 1. Program Written in C

The following program shows the control over the instrument using the C language with the VISA library.

The instrument address is assigned in VISA Resource Name format in the command line at the start of the program. For more detail on VISA Resource Name see the VISA library documentation.

Program description:

1. Sets up the communication with the instrument.
2. Reads out and displays the instrument information string.
3. Sets some parameters of the instrument.
4. Triggers the measurement and waits for the sweep completion.
5. Reads out the measurement data and the frequency values at the measurement points.
6. Displays the measurement data

```
// Example1.cpp
//
// VISA Header: visa.h (must be included)
// VISA Library: visa32.lib (must be linked with)
#include "stdafx.h"
#include "visa.h"

int main(int argc, char* argv[])
{
    ViStatus status;                        // Error checking
    ViSession defaultRM, instr;       // Communication channels
    ViUInt32 retCount;                    // Return count from string I/O
    ViByte buffer[255];                   // Buffer for string I/O
    int temp;
    int NOP = 21;                           // Number of measurement
points
    const int maxCnt = 100;               // Maximum reading count
    double Data[maxCnt*2];            // Measurement data array
    double Freq[maxCnt];                  // Frequency array

    if (argc < 2)
    {
        printf("\nUsage: Example4 <VISA address>\n\n");
        printf("VISA address examples:\n");
        printf("       TCPIP::nnn.nnn.nnn.nnn::INSTR\n");
        printf("       TCPIP::devicename::INSTR\n");
        return -1;
    }

    status = viOpenDefaultRM(&defaultRM);
    if (status < VI_SUCCESS)
    {
        printf("Can't initialize VISA\n");
```

```
        return -1;
    }

    status = viOpen(defaultRM, argv[1], VI_NULL, VI_NULL, &instr);
    if (status < VI_SUCCESS)
    {
        printf("Can't open VISA address: %s\n", argv[1]);
        return -1;
    }
    //
    // Set the timeout for message-based communication
    //
    viSetAttribute(instr, VI_ATTR_TMO_VALUE, 5000);
    //
  // Read ID string from Analyzer
    //
    viPrintf(instr, "*IDN?\n");
    viRead(instr, buffer, sizeof(buffer), &retCount);
    printf("*IDN? Returned %d bytes: %.*s\n\n", retCount, retCount,
buffer);
    //
  // Set up the Analyzer
    //
    viPrintf(instr, "SYST:PRES\n");
    viPrintf(instr, "SENS:SWE:POIN %d\n", NOP);
    viPrintf(instr, "CALC:PAR1:DEF S21\n");
    viPrintf(instr, "CALC:PAR1:SEL\n");
    viPrintf(instr, "CALC:FORM MLOG\n");
    viPrintf(instr, "SENS:BAND 10\n");
    //
    // Trigger measurement and wait for completion
    //
    viPrintf(instr, ":TRIG:SOUR BUS\n");
    viPrintf(instr, ":TRIG:SING\n");
    viQueryf(instr, "*OPC?\n", "%d", &temp);
    //
    // Read out measurement data
    //
    retCount = maxCnt * 2;
    viQueryf(instr, "CALC:DATA:FDAT?\n", "%,#lf", &retCount, Data);
    retCount = maxCnt;
    viQueryf(instr, "SENS:FREQ:DATA?\n", "%,#lf", &retCount, Freq);
    //
    // Display measurement data
    //
    printf("%20s %20s %20s\n", "Frequency", "Data1", "Data2");
    for (int i = 0; i < NOP; i++)
    {
        printf("%20f %20f %20f\n", Freq[i], Data[i*2], Data[i*2+1]);
    }

    status = viClose(instr);
    status = viClose(defaultRM);
    return 0;
}
```

Example 2. Program Written in LabView

The following program shows the control over the instrument using the LabView language with the VISA library.

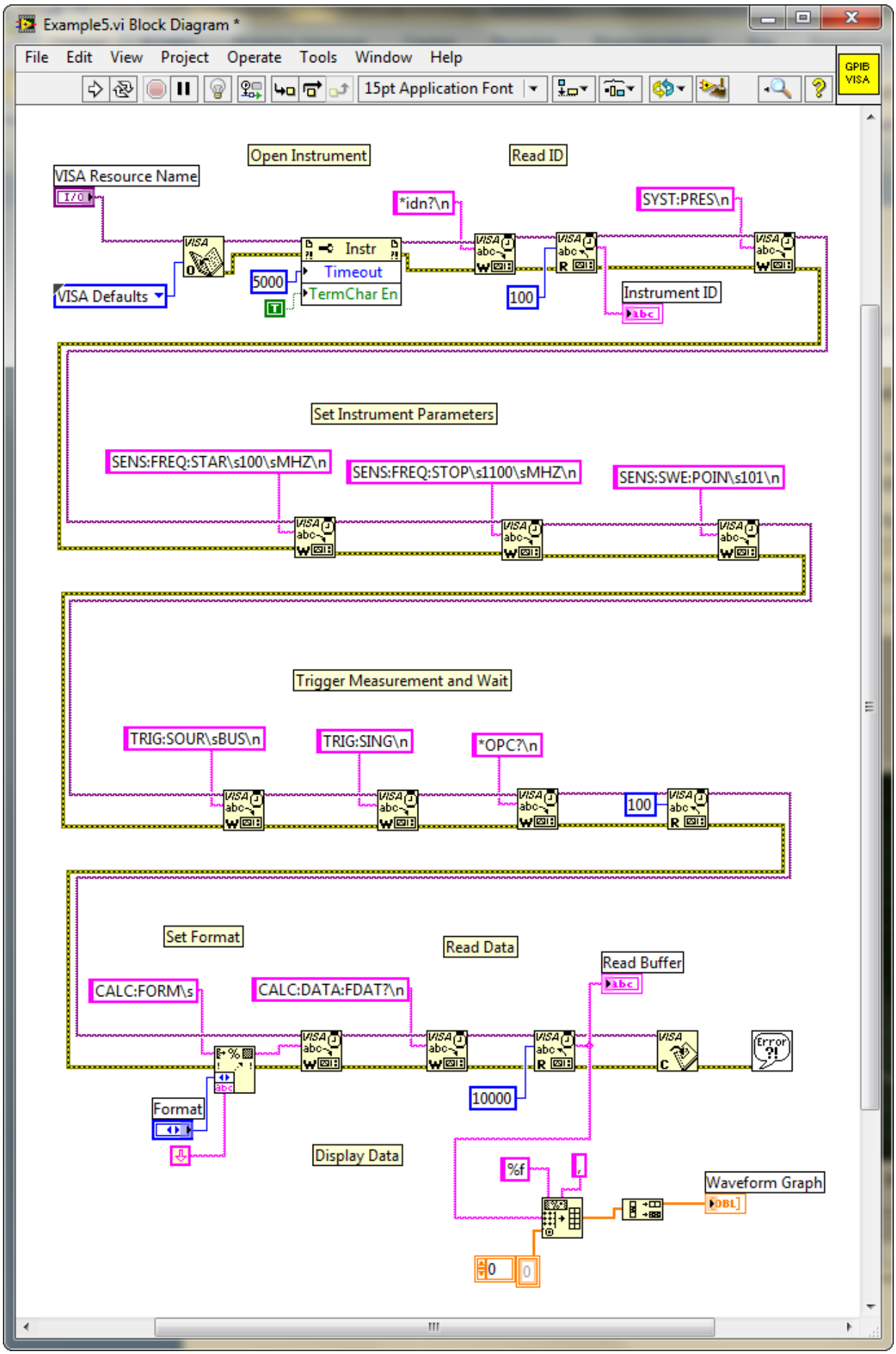
Below see the block diagram of the program and front panel of the program with the program execution result.

The front panel contains the entry field for the instrument name “VISA Resource Name”. For more detail on VISA Resource Name see the VISA library documentation.

The user must enter the instrument address, select the trace format in the “Format” field, and click the “Run” button. As the result of the program, the instrument information string will be displayed and the measurement trace will be plotted.

Program description:

1. Sets up the communication with the instrument.
2. Reads out and displays the instrument information string.
3. Sets some parameters of the instrument.
4. Generates the trigger and waits for the sweep completion.
5. Sets the trace format to the format entered by the user in the “Format” field.
6. Reads out the measurement data.
7. Displays the measurement data.


Example5.vi Block Diagram *
File Edit View Project Operate Tools Window Help
15pt Application Font
GPIB VISA
Open Instrument
VISA Resource Name
VISA Defaults
Instr
Timeout
TermChar En
5000
Read ID
*idn?\n
100
Instrument ID
SYST:PRES\n
Set Instrument Parameters
SENS:FREQ:STAR\s100\sMHZ\n
SENS:FREQ:STOP\s1100\sMHZ\n
SENS:SWE:POIN\s101\n
Trigger Measurement and Wait
TRIG:SOUR\sBUS\n
TRIG:SING\n
*OPC?\n
100
Set Format
CALC:FORM\s
Format
Read Data
CALC:DATA:FDAT?\n
10000
Read Buffer
Display Data
%f
Waveform Graph
DBL
0
0

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP