

Agilent 4396B Network/Spectrum/Impedance Analyzer

GPIB Command Reference

SERIAL NUMBERS

This manual applies directly to instruments which have the serial number prefix JP1KE.
For additional important information about serial numbers,
read “Serial Number” in Appendix A of this Manual.



Agilent Technologies

Agilent Part No. 04396-90044
Printed in JAPAN March 2000

Fourth Edition

Notice

The information contained in this document is subject to change without notice.

This document contains proprietary information that is protected by copyright. All rights are reserved. No part of this document may be photocopied, reproduced, or translated to another language without the prior written consent of the Agilent Technologies.



Agilent Technologies Japan, Ltd.
Component Test PGU-Kobe
1-3-2, Murotani, Nishi-ku, Kobe-shi,
Hyogo, 651-2241 Japan

Manual Printing History

The manual printing date and part number indicate its current edition. The printing date changes when a new edition is printed. (Minor corrections and updates that are incorporated at reprint do not cause the date to change.) The manual part number changes when extensive technical changes are incorporated.

| | |
|------------------|---|
| March 1997 | First Edition (part number: 04396-90024) |
| July 1997 | Second Edition (part number: 04396-90034) |
| March 1998 | Third Edition (part number: 04396-90044) |
| March 2000 | Fourth Edition (part number: 04396-90044) |

Typeface Conventions

| | |
|-----------------|--|
| Bold | Boldface type is used when a term is defined. For example: icons are symbols. |
| <i>Italics</i> | Italic type is used for emphasis and for titles of manuals and other publications. Italic type is also used for keyboard entries when a name or a variable must be typed in place of the words in italics. For example: copy <i>filename</i> means to type the word <i>copy</i> , to type a space, and then to type the name of a file such as <i>file1</i> . |
| Computer | Computer font is used for on-screen prompts and messages. |
| HARDKEYS | Labeled keys on the instrument front panel are enclosed in  . |
| SOFTKEYS | Softkeys located to the right of the CRT are enclosed in  . |

Documentation Map

The following manuals are available for the analyzer.

User's Guide (Agilent Part Number 04396-90031)

The User's Guide walks you through system setup and initial power-on, shows how to make basic measurements, explains commonly used features, and typical application measurement examples. After you receive your analyzer, begin with this manual.

Task Reference (Agilent Part Number 04396-90030)

Task Reference helps you to learn how to use the analyzer. This manual provides simple step-by-step instructions without concepts.

Function Reference (Agilent Part Number 04396-90052)

The Function Reference describes all function accessed from the front panel keys and softkeys. It also provides information on options and accessories available, specifications, system performance, and some topics about the analyzer's features.

Programming Guide (Agilent Part Number 04396-90043)

The Programming Guide shows how to write and use BASIC program to control the analyzer and describes how HP Instrument BASIC works with the analyzer.

GPIB Command Reference (Agilent Part Number 04396-90044)

The GPIB Command Reference provides a summary of all available GPIB commands. It also provides information on the status reporting structure and the trigger system (these features conform to the SCPI standard).

Option 010 Operating Handbook (Option 010 only) (Agilent Part Number 04396-90036)

The option 010 Operation Handbook describes the unique impedance measurement functions of the 4396B with option 010.

HP Instrument BASIC Manual Set (Agilent Part Number E2083-90000)

The HP Instrument BASIC User's Handbook introduces you to the HP Instrument BASIC programming language, provide some helpful hints on getting the most use from it, and provide a general programming reference. It is divided into three books, *HP Instrument BASIC Programming Techniques*, *HP Instrument BASIC Interface Techniques*, and *HP Instrument BASIC Language Reference*.

Performance Test Manual (Agilent Part Number 04396-90130)

The Performance Test Manual explains how to verify conformance to published specifications.

Service Manual (Option 0BW only) (Agilent Part Number 04396-90121)

The Service Manual explains how to adjust, troubleshoot, and repair the instrument. This manual is option 0BW only.

Contents

1. Introduction

| | |
|---|-----|
| GPIB Commands | 1-3 |
| Simple Commands | 1-3 |
| SCPI Commands | 1-3 |
| Common Commands | 1-3 |
| SCPI Subsystem Commands | 1-3 |
| Subsystem Command Tree | 1-4 |
| Command Abbreviations | 1-5 |
| Program Message Syntax | 1-5 |
| Case | 1-5 |
| Program Message Terminator | 1-5 |
| Multiple Messages | 1-5 |
| Query and Response Message Syntax | 1-5 |
| Parameters | 1-6 |
| Parameter Types | 1-6 |

2. Commands Reference

| | |
|---|------|
| Simple Commands | 2-3 |
| ADDRCONT \square <i><numeric></i> | 2-3 |
| ANAOCH1 | 2-3 |
| ANAOCH2 | 2-3 |
| ANAO DATA | 2-3 |
| ANAO MEMO | 2-4 |
| ANARANG \square <i><numeric1></i> , <i><numeric2></i> | 2-4 |
| ANARFULL | 2-4 |
| ATT \square <i><numeric></i> [DB] | 2-4 |
| ATTAUTO \square {OFF ON 0 1} | 2-5 |
| ATTP1 \square <i><numeric></i> [DB] | 2-5 |
| ATTP2 \square <i><numeric></i> [DB] | 2-5 |
| AUTO | 2-6 |
| AVER \square {OFF ON 0 1} | 2-6 |
| AVERFACT \square <i><numeric></i> | 2-6 |
| AVERREST | 2-6 |
| BACI \square <i><numeric></i> [PCT] | 2-7 |
| BEEPDONE \square {OFF ON 0 1} | 2-7 |
| BEEPFAIL \square {OFF ON 0 1} | 2-7 |
| BEEPWARN \square {OFF ON 0 1} | 2-8 |
| BLIGHT \square {OFF ON 0 1} | 2-8 |
| BOTV \square <i><numeric></i> | 2-8 |
| BW \square <i><numeric></i> [HZ KHZ MAHZ] | 2-9 |
| BWAUTO \square {OFF ON 0 1} | 2-9 |
| BWSRAT \square <i><numeric></i> | 2-10 |
| C0 \square <i><numeric></i> | 2-10 |
| C1 \square <i><numeric></i> | 2-10 |
| C2 \square <i><numeric></i> | 2-11 |
| CALCASSI | 2-11 |

| | |
|--|------|
| CALECPARA | 2-11 |
| CALI□{NONE RESP RAI S111 S221 FUL2 ONE2 IMP} | 2-11 |
| CALK□{APC7 APC35 N50 N75 USED} | 2-12 |
| CALS□<numeric> | 2-13 |
| CBRI□<numeric>[PCT] | 2-13 |
| CENT□<numeric>[HZ KHZ MAHZ GHZ DBM] | 2-13 |
| CHAD□<string> | 2-14 |
| CHAN1 | 2-14 |
| CHAN2 | 2-15 |
| CIRF□{RI LIN LOG RX GB SWR} | 2-15 |
| CLAD | 2-16 |
| CLASIMP{A B C} | 2-16 |
| CLASS11{A B C} | 2-16 |
| CLASS22{A B C} | 2-16 |
| CLEL | 2-16 |
| CLES | 2-17 |
| CLOSE | 2-17 |
| CNTS□<numeric>[HZ KHZ MAHZ GHZ] | 2-17 |
| CNTSAUTO□{OFF ON 0 1} | 2-17 |
| COLO□<parameter> | 2-18 |
| COLOR□<numeric>[PCT] | 2-20 |
| COMC{A B C} | 2-20 |
| COMCDAT{A B C}□{OFF ON 0 1} | 2-20 |
| COMKDONE | 2-21 |
| COMP | 2-21 |
| COMS | 2-21 |
| COMSDONE | 2-21 |
| CONT | 2-21 |
| CONV□<parameter> | 2-22 |
| COPA | 2-22 |
| COPT□{OFF ON 0 1} | 2-23 |
| CORR□{OFF ON 0 1} | 2-23 |
| COUC□{OFF ON 0 1} | 2-24 |
| CRED□<string> | 2-24 |
| CRSC□{OFF ON 0 1} | 2-25 |
| CWD? | 2-25 |
| CWFREQ□<numeric>[HZ KHZ MAHZ GHZ] | 2-25 |
| DATAOVAL□<numeric> | 2-26 |
| DATGAIN□<numeric> | 2-26 |
| DATMEM | 2-26 |
| DATOVAL□<numeric> | 2-27 |
| DAYMYEAR | 2-27 |
| DEFC | 2-27 |
| DEFEC{R1 C1 L1 C0}□<numeric> | 2-28 |
| DEFGO | 2-28 |
| DEFS□{1-8} | 2-29 |
| DEFSLOAD{R L}□<numeric> | 2-29 |
| DEFSOPEN{G C}□<numeric> | 2-30 |
| DEFSSHOR{R L}□<numeric> | 2-30 |
| DET□{POS NEG SAM} | 2-31 |
| DFLT | 2-31 |
| DHOLD□{OFF MAX MIN} | 2-32 |
| DISA□{ALLI HIHB ALLB BASS} | 2-32 |
| DISECIRC | 2-33 |
| DISECPARA {OFF ON 0 1} | 2-33 |

| | |
|--|------|
| DISF□{DOS LIF} | 2-34 |
| DISL | 2-34 |
| DISLLIST | 2-34 |
| DISMAMP□{UL MD} | 2-34 |
| DISMPRM□{STSP CTSP} | 2-35 |
| DISP□{DATA MEMO DATM} | 2-35 |
| DMKR□{ON FIX TRAC OFF} | 2-36 |
| DMKRAUV□<numeric> | 2-36 |
| DMKRPRM□<numeric>[HZ KHZ MAHZ GHZ DBM] | 2-37 |
| DMKRVAL□<numeric> | 2-37 |
| DONE | 2-37 |
| DPI□<numeric> | 2-38 |
| DSKEY | 2-38 |
| DUAC□{OFF ON 0 1} | 2-39 |
| EDITDONE | 2-39 |
| EDITLIML | 2-39 |
| EDITLIST | 2-40 |
| ELED□<numeric>[S MS US NS PS FS] | 2-40 |
| ENKEY | 2-40 |
| EQUC□CIR{A B C D E} | 2-40 |
| EQUC0?□<numeric> | 2-41 |
| EQUCPARA? | 2-41 |
| EQUCPARS? | 2-41 |
| EQUCPARS4? | 2-41 |
| EQUM?□<numeric> | 2-41 |
| ESB? | 2-42 |
| ESNB□<numeric> | 2-42 |
| EXPP□{OFF ON 0 1} | 2-42 |
| FILC□<string1>,<string2>,<string3>,<string4> | 2-43 |
| FIXE□<numeric> | 2-43 |
| FIXKDONE | 2-43 |
| FIXT□{NONE 16191 16192 16193 16194 USED} | 2-43 |
| FMT□<parameter> | 2-44 |
| FNAME?□<numeric> | 2-45 |
| FNUM? | 2-46 |
| FORM2 | 2-46 |
| FORM3 | 2-46 |
| FORM4 | 2-46 |
| FORM5 | 2-46 |
| FORMFEED□{OFF ON 0 1} | 2-47 |
| FREQ | 2-47 |
| FSIZE?□<string> | 2-47 |
| FULS | 2-47 |
| FWDI | 2-48 |
| FWDM | 2-48 |
| FWDT | 2-48 |
| GATCTL□{LEV EDG} | 2-48 |
| GATDLY□<numeric>[US MS S] | 2-49 |
| GATLEN□<numeric>[US MS S] | 2-49 |
| GRODAPER□<numeric>[PCT] | 2-49 |
| HOLD | 2-50 |
| INID | 2-50 |
| INP8IO? | 2-50 |
| INPUCALC{1-12}□<numeric (1)>,<numeric (2)>,...,<numeric (n)> | 2-51 |
| INPUCALK□<block> | 2-51 |

| | |
|---|------|
| INPUCOMC{1 2 3}□<numeric> | 2-51 |
| INPUTD | 2-52 |
| INPUTDATA□<numeric (1)>,<numeric (2)>, . . . ,<numeric (n)> | 2-52 |
| INPUTTRC□<numeric (1)>,<numeric (2)>, . . . ,<numeric (n)> | 2-53 |
| INPUTLOAA□<numeric (1)>,<numeric (2)>, . . . ,<numeric (n)> | 2-53 |
| INPUTOPEA□<numeric (1)>,<numeric (2)>, . . . ,<numeric (n)> | 2-53 |
| INPUTRAW{1-4}□<numeric (1)>,<numeric (2)>, . . . ,<numeric (n)> | 2-54 |
| INPUTSHOA□<numeric (1)>,<numeric (2)>, . . . ,<numeric (n)> | 2-54 |
| INPZ□{50 75}[OHM] | 2-54 |
| INTE□<numeric>[PCT] | 2-55 |
| ISOD | 2-55 |
| ISOL | 2-55 |
| KEY□<numeric> | 2-55 |
| KITD | 2-56 |
| LABECOMK□<string> | 2-56 |
| LABEFIX□<string> | 2-56 |
| LABEFDW{T M}□<string> | 2-57 |
| LABELIMP{A B C}□<string> | 2-57 |
| LABERES{P I}□<string> | 2-57 |
| LABEREV{T M}□<string> | 2-58 |
| LABES11{A B C}□<string> | 2-58 |
| LABES22{A B C}□<string> | 2-59 |
| LABK□<string> | 2-59 |
| LABS□<string> | 2-60 |
| LANDSCAPE□{OFF ON 0 1} | 2-60 |
| LIMCLEL | 2-60 |
| LIMD□<numeric> | 2-60 |
| LIMEDONE | 2-61 |
| LIMIAMPO□<numeric> | 2-61 |
| LIMILINE□{OFF ON 0 1} | 2-61 |
| LIMIPRMO□<numeric> | 2-62 |
| LIMITEST□{OFF ON 0 1} | 2-62 |
| LIML□<numeric> | 2-62 |
| LIMM□<numeric> | 2-63 |
| LIMPRM□<numeric>[HZ KHZ MAHZ GHZ DBM] | 2-63 |
| LIMSADD | 2-63 |
| LIMSDEL | 2-63 |
| LIMSDON | 2-64 |
| LIMSEDI□[<numeric>] | 2-64 |
| LIMU□<numeric> | 2-64 |
| LMAX?□<numeric> | 2-64 |
| LMARG□<numeric> | 2-65 |
| LMIN?□<numeric> | 2-65 |
| LISV | 2-65 |
| LVCDT□<numeric>[DB] | 2-66 |
| LVLCAL | 2-66 |
| MATH□{DATA DDVM DMNM DPLM} | 2-66 |
| MAXMLEV□<numeric>[DBM] | 2-67 |
| MEAS□<parameter> | 2-67 |
| MEASTAT□{OFF ON 0 1} | 2-70 |
| MKR□{OFF ON 0 1} | 2-70 |
| MKRAMPO | 2-70 |
| MKRAUV? | 2-70 |
| MKRCENT | 2-71 |
| MKRCONT□{OFF ON 0 1} | 2-71 |

| | |
|---------------------------------------|------|
| MKRCOUP□{OFF ON 0 1} | 2-71 |
| MKRCSTE | 2-71 |
| MKRDCENT | 2-72 |
| MKRDCSTE | 2-72 |
| MKRDELA | 2-72 |
| MKRDSPAN | 2-72 |
| MKRL□{OFF ON 0 1} | 2-72 |
| MKRMIDD | 2-73 |
| MKRNOI□{OFF ON 0 1} | 2-73 |
| MKRO□{DATA MEMO} | 2-73 |
| MKROFS | 2-73 |
| MKRP□<numeric> | 2-74 |
| MKRPKD | 2-74 |
| MKRPRM□<numeric>[HZ KHZ MAHZ GHZ DBM] | 2-74 |
| MKRREF | 2-74 |
| MKRSTAR | 2-75 |
| MKRSTOP | 2-75 |
| MKRSWPRM | 2-75 |
| MKRTHRE | 2-75 |
| MKRTIME□{OFF ON 0 1} | 2-76 |
| MKRVAL? | 2-76 |
| MKRZM | 2-76 |
| MOD1 | 2-76 |
| MODICOMK | 2-77 |
| MODIFIX | 2-77 |
| MONDYEAR | 2-77 |
| NA | 2-77 |
| NEXP | 2-78 |
| NEXPK? | 2-78 |
| NUMG□<numeric> | 2-78 |
| NUMLMAX? | 2-79 |
| NUMLMIN? | 2-79 |
| OFSD□<numeric>[S] | 2-79 |
| OFSL□<value> | 2-79 |
| OFSZ□<numeric>[OHM KOHM] | 2-80 |
| OMII | 2-80 |
| OPEP | 2-80 |
| OSE□<numeric> | 2-80 |
| OSER? | 2-81 |
| OSNT□<numeric> | 2-81 |
| OSPT□<numeric> | 2-81 |
| OSR? | 2-81 |
| OUT8IO□<numeric> | 2-82 |
| OUTPCALC{1-12}? | 2-82 |
| OUTPCALK? | 2-82 |
| OUTPCERR? | 2-82 |
| OUTPCFIL? | 2-83 |
| OUTPCOMC{1 2 3}? | 2-83 |
| OUTPDATA? | 2-83 |
| OUTPDATAP?□<numeric> | 2-84 |
| OUTPDMKR? | 2-84 |
| OUTPDTRC? | 2-85 |
| OUTPDTRCP?□<numeric> | 2-85 |
| OUTPERRO? | 2-86 |
| OUTPFAIP? | 2-86 |

| | |
|---------------------------------------|-------|
| OUTPFILT? | 2-86 |
| OUTPLIMF? | 2-86 |
| OUTPLIML? | 2-87 |
| OUTPLIMM? | 2-88 |
| OUTPMAX? | 2-88 |
| OUTPMEAN? | 2-88 |
| OUTPMEMO? | 2-88 |
| OUTPMEMOP?□<numeric> | 2-89 |
| OUTPMIN? | 2-89 |
| OUTPMINMAX? | 2-89 |
| OUTPMKR? | 2-89 |
| OUTPMSTA? | 2-90 |
| OUTPMTRC? | 2-90 |
| OUTPMTRCP?□<numeric> | 2-91 |
| OUTPMWID? | 2-91 |
| OUTPRAW{1-4}? | 2-92 |
| OUTPRESF? | 2-92 |
| OUTPRESO? | 2-92 |
| OUTPRESR? | 2-92 |
| OUTPSMKR{1-7}? | 2-92 |
| OUTPSWPRM? | 2-93 |
| OUTPSWPRMP?□<numeric> | 2-93 |
| OUTPXFIL? | 2-93 |
| PARS□{OFF ON 0 1} | 2-94 |
| PEAK? | 2-94 |
| PEAKCENT | 2-94 |
| PHAO□<numeric>[DEG] | 2-95 |
| PHAU {RAD DEG} | 2-95 |
| PKDLTX□<numeric>[HZ KHZ MAHZ GHZ DBM] | 2-95 |
| PKDLTY□<numeric> | 2-96 |
| PKPOL□{POS NEG} | 2-96 |
| PKTHRE□{OFF ON 0 1} | 2-97 |
| PKTHVAL□<value> | 2-97 |
| POIN□<numeric> | 2-98 |
| POLE?□<numeric> | 2-98 |
| PORE□{OFF ON 0 1} | 2-98 |
| PORT1□<numeric>[S MS US NS PS] | 2-99 |
| PORT2□<numeric>[S MS US NS PS] | 2-99 |
| PORTA□<numeric>[S MS US NS PS] | 2-99 |
| PORTB□<numeric>[S MS US NS PS] | 2-100 |
| PORTR□<numeric>[S MS US NS PS] | 2-100 |
| PORTZ□<numeric> | 2-100 |
| POWE□<numeric>[DBM] | 2-101 |
| PREP | 2-101 |
| PRES | 2-101 |
| PRIC | 2-102 |
| PRICFIXE | 2-102 |
| PRICVARI | 2-102 |
| PRINALL | 2-103 |
| PRIS | 2-103 |
| PRSOFT□{OFF ON 0 1} | 2-103 |
| PRSMKRS | 2-104 |
| PURG□<string> | 2-104 |
| RAID | 2-104 |
| RAISOL | 2-104 |

| | |
|---------------------------|-------|
| RAIRESP | 2-105 |
| READ? | 2-105 |
| RECC | 2-105 |
| RECD□<string> | 2-106 |
| REFD | 2-106 |
| REFL | 2-106 |
| REFP□<numeric> | 2-107 |
| REFV□<numeric> | 2-107 |
| REFX□<numeric> | 2-108 |
| REFY□<numeric> | 2-108 |
| REPTSMP□{OFF ON 0 1} | 2-109 |
| RESAVD□<string> | 2-109 |
| RESC | 2-109 |
| RESCOM | 2-110 |
| RESD | 2-110 |
| RESPDONE | 2-110 |
| REST | 2-110 |
| REVI | 2-110 |
| REVM | 2-111 |
| REVT | 2-111 |
| RFO□{OFF ON 0 1} | 2-111 |
| ROPEN□<string> | 2-112 |
| RPLENV? | 2-112 |
| RPLHEI? | 2-112 |
| RPLLHEI? | 2-113 |
| RPLMEA? | 2-113 |
| RPLPP? | 2-113 |
| RPLRHEI? | 2-113 |
| RPLVAL? | 2-113 |
| RSCO | 2-113 |
| SA | 2-114 |
| SADD | 2-114 |
| SAUNIT□{DBM DBV DBUV W V} | 2-114 |
| SAV1 | 2-115 |
| SAV2 | 2-115 |
| SAVC | 2-115 |
| SAVCAL□{OFF ON 0 1} | 2-116 |
| SAVCOM | 2-116 |
| SAVDASC□<string> | 2-116 |
| SAVDAT□{OFF ON 0 1} | 2-117 |
| SAVDDAT□<string> | 2-117 |
| SAVDSTA□<string> | 2-117 |
| SAVDSTAC□<string> | 2-118 |
| SAVDTIF□<string> | 2-118 |
| SAVDTRC□{OFF ON 0 1} | 2-119 |
| SAVEUSEK | 2-119 |
| SAVIMP | 2-119 |
| SAVMEM□{OFF ON 0 1} | 2-119 |
| SAVMTRC□{OFF ON 0 1} | 2-120 |
| SAVRAW□{OFF ON 0 1} | 2-120 |
| SAVUCOMK | 2-120 |
| SAVUFIXT | 2-120 |
| SCAC□{OFF ON 0 1} | 2-121 |
| SCAF□{DATA MEMO} | 2-121 |
| SCAL□<numeric> | 2-121 |

| | | |
|----------|---|-------|
| SCRN | □{OFF ON 0 1} | 2-122 |
| SDEL | | 2-122 |
| SDON | | 2-122 |
| SEAL | | 2-122 |
| SEAM | □{PEAK MAX MIN TARG PKSA PKSR PKSL OFF} | 2-123 |
| SEANPK | | 2-123 |
| SEANPKL | | 2-124 |
| SEANPKR | | 2-124 |
| SEAR | | 2-124 |
| SEARSTR | | 2-124 |
| SEARSTRL | | 2-124 |
| SEARSTRR | | 2-125 |
| SEATARG | □<numeric>[DB DEG S OHM] | 2-125 |
| SEDI | □<numeric> | 2-125 |
| SETCDATE | □<numeric (year)>,<numeric (month)>,<numeric (day)> | 2-126 |
| SETCTIME | □<numeric (hour)>,<numeric (minute)>,<numeric (second)> | 2-126 |
| SETZ | □<numeric>[OHM KOHM MAOHM] | 2-126 |
| SGTRK | □{OFF ON 0 1} | 2-127 |
| SIMFCHAR | | 2-127 |
| SING | | 2-128 |
| SLOP | □{OFF ON 0 1} | 2-128 |
| SLOPE | □<numeric> | 2-129 |
| SMKR | {1-7}□{OFF ON 0 1} | 2-129 |
| SMKRAUV | {1-7}? | 2-130 |
| SMKRP | {1-7}□<numeric> | 2-130 |
| SMKRPRM | {1-7}□<numeric>[HZ KHZ MAHZ GHZ DBM] | 2-131 |
| SMKRVAL | {1-7}? | 2-131 |
| SPAN | □<numeric>[HZ KHZ MAHZ GHZ DBM] | 2-132 |
| SPECFWD | {M T}□<numeric 1>[,<numeric 2>[,... [<numeric 7>] | 2-132 |
| SPECIMP | {A B C}□<numeric 1>[,<numeric 2>[,... [<numeric 7>] | 2-132 |
| SPECRES | {I P}□<numeric 1>[,<numeric 2>[,... [<numeric 7>] | 2-133 |
| SPECREV | {M T}□<numeric 1>[,<numeric 2>[,... [<numeric 7>] | 2-133 |
| SPECS11 | {A B C}□<numeric 1>[,<numeric 2>[,... [<numeric 7>] | 2-134 |
| SPECS22 | {A B C}□<numeric 1>[,<numeric 2>[,... [<numeric 7>] | 2-134 |
| SPLD | □{OFF ON 0 1} | 2-135 |
| SQUI | | 2-135 |
| STAN | {A-G} | 2-135 |
| STAR | □<numeric>[HZ KHZ MAHZ GHZ DBM] | 2-135 |
| STDD | | 2-136 |
| STDT | □{OPEN SHOR LOAD DELA ARBI} | 2-136 |
| STOD | {DISK MEMO} | 2-136 |
| STOP | □<numeric>[HZ KHZ MAHZ GHZ DBM] | 2-137 |
| SVCO | | 2-137 |
| SWET | □<numeric>[S MS] | 2-137 |
| SWETAUTO | □{OFF ON 0 1} | 2-138 |
| SWPT | □{LINF LOGF LIST POWE} | 2-138 |
| TARL? | | 2-139 |
| TARR? | | 2-139 |
| TERI | □<numeric>[OHM KOHM] | 2-139 |
| TESS? | | 2-139 |
| THRR | □<numeric> | 2-139 |
| TINT | □<numeric> | 2-140 |
| TITL | □<string> | 2-140 |
| TMARG | □<numeric> | 2-140 |
| TOPV | □<numeric> | 2-141 |

| | | |
|---|---|-------|
| TRACK | {OFF ON 0 1} | 2-141 |
| TRAD | | 2-141 |
| TRAN | | 2-142 |
| TRGEVE | {SWE POIN} | 2-142 |
| TRGP | {POS NEG} | 2-142 |
| TRGS | {INT EXT BUS VID MAN GAT} | 2-143 |
| USKEY | | 2-143 |
| VBW | <numeric>[HZ KHZ MAHZ] | 2-144 |
| VBWT | {LIN LOG} | 2-144 |
| VELOFACT | <numeric> | 2-144 |
| VIDLVL | <numeric> | 2-145 |
| WIDSIN | | 2-145 |
| WIDSOUT | | 2-145 |
| WIDT | {OFF ON 0 1} | 2-145 |
| WIDV | <numeric>[DB DEG S OHM] | 2-146 |
| WIDVTYPE | {DIVS2 MULS2 DIV2 FIXed} | 2-146 |
| WOPEN | <string>[,<numeric>] | 2-147 |
| WRITE | <block> | 2-147 |
| ZA | | 2-147 |
| ZMAPER | <numeric> | 2-148 |
| Common Commands | | 2-149 |
| *CLS | | 2-149 |
| *ESE | <numeric> | 2-149 |
| *ESR? | | 2-149 |
| *IDN? | | 2-149 |
| *OPC | | 2-150 |
| *OPT? | | 2-150 |
| *PCB | <numeric> | 2-150 |
| *RST | | 2-151 |
| *SRE | <numeric> | 2-151 |
| *STB? | | 2-151 |
| *TRG | | 2-151 |
| *TST? | | 2-152 |
| *WAI | | 2-152 |
| SCPI Commands With No Equivalent Simple Command | | 2-153 |
| :CALCulate:MATH1[:EXPReSSion]:CATalog? | | 2-153 |
| :CALCulate:MATH2[:EXPReSSion]:CATalog? | | 2-153 |
| :CALCulate:PATH? | | 2-153 |
| :PROGram:CATalog? | | 2-154 |
| :PROGram[:SELeCted]:DEFine | <block> | 2-154 |
| :PROGram[:SELeCted]:DELete[:SELeCted] | | 2-155 |
| :PROGram[:SELeCted]:DELete:ALL | | 2-155 |
| :PROGram[:SELeCted]:EXECute | <string> | 2-155 |
| :PROGram[:SELeCted]:MALLocate | {<numeric> DEFault} | 2-155 |
| :PROGram[:SELeCted]:NAME | <string> | 2-155 |
| :PROGram[:SELeCted]:NUMBer | <string>,<numeric 1>[,<numeric 2> [, ... [,<numeric n>] | 2-156 |
| :PROGram[:SELeCted]:STATe | {RUN PAUSe STOP CONTInue} | 2-156 |
| :PROGram[:SELeCted]:STRing | <string (varname)>,<string (value 1)>[,<string (value 2)> [, ... [,<string (value n)>] | 2-157 |
| :PROGram[:SELeCted]:WAIT | | 2-157 |
| :PROGram:EXPLicit:DEFine | "PROG",<block> | 2-158 |
| :PROGram:EXPLicit:DELete | "PROG" | 2-158 |
| :PROGram:EXPLicit:EXECute | "PROG",<string> | 2-158 |
| :PROGram:EXPLicit:MALLocate | "PROG",{<numeric> DEFault} | 2-158 |

| | |
|---|-------|
| :PROG:EXPLicit:NAME□“PROG”,<string> | 2-158 |
| :PROG:EXPLicit:NUMBer□“PROG”,<varname>,<numeric 1>[,<numeric 2> [,...[,<numeric n>]...]] | 2-158 |
| :PROG:EXPLicit:STATe□“PROG”,{RUN PAUSe STOP CONTInue} | 2-158 |
| :PROG:EXPLicit:STRing□“PROG”,<varname>,<string 1>[,<string 2> [,... [,<string n>]...]] | 2-158 |
| :PROG:EXPLicit:WAIT “PROG” | 2-158 |
| :STATus:PRESet | 2-159 |
| :STATus:QUESTionable:CONDition? | 2-159 |
| :STATus:QUESTionable:ENABle□<numeric> | 2-159 |
| :STATus:QUESTionable[:EVENT]? | 2-159 |
| :SYSTem:VERSIon? | 2-159 |
| Service Related Commands | 2-160 |
| :DIAG:EREFErence:STATe? | 2-160 |
| :DIAG:FREVIion? | 2-160 |
| :DIAG:INIT:RESult? | 2-161 |
| :DIAG:SERVice:BUS:AZERo□{OFF ON 0 1} | 2-161 |
| :DIAG:SERVice:BUS:DC□<numeric> | 2-161 |
| :DIAG:SERVice:BUS:FREQ□<numeric> | 2-162 |
| :DIAG:SERVice:BUS:STATe□{OFF ON 0 1} | 2-162 |
| :DIAG:SERVice:BUS:WAIT□<numeric> | 2-162 |
| :DIAG:SERVice:CCONstant:FRESponse□{OFF ON 0 1} | 2-163 |
| :DIAG:SERVice:CCONstant:IFGain□{OFF ON 0 1} | 2-163 |
| :DIAG:SERVice:CCONstant:SOURce□{OFF ON 0 1} | 2-163 |
| :DIAG:SERVice:CCONstant:XTAL□{OFF ON 0 1} | 2-164 |
| :DIAG:SERVice:IF:ADMX:MODE□{AUTO ALTErnate DEG0 DEG90} | 2-164 |
| :DIAG:SERVice:IF:BPFilter:MODE□{AUTO BW3M BW1M XTAL} | 2-165 |
| :DIAG:SERVice:IF:GAIN:MODE□{AUTO MANual} | 2-165 |
| :DIAG:SERVice:IF:GAIN:W□{AUTO DB0 DB10} | 2-165 |
| :DIAG:SERVice:IF:GAIN:X□{AUTO DB0 DB18} | 2-166 |
| :DIAG:SERVice:IF:GAIN:Y□{AUTO DB0 DB6 DB12 DB18} | 2-166 |
| :DIAG:SERVice:IF:GAIN:Z□{AUTO DB0 DB2 DB4 DB18} | 2-167 |
| :DIAG:SERVice:IF:LPFilter:MODE□{AUTO BW5K BW15K BW50K BW150K THROugh} | 2-167 |
| :DIAG:SERVice:IF:RANGE:F□{HIGH LOW} | 2-168 |
| :DIAG:SERVice:IF:RANGE:MODE□{AUTO MANual} | 2-168 |
| :DIAG:SERVice:IF:RANGE:R□{HIGH LOW} | 2-168 |
| :DIAG:SERVice:IF:SHBW:MODE□{AUTO NARRow MIDdle WIDE} | 2-168 |
| :DIAG:SERVice:IF:TLOCAL:MODE□{AUTO AC DC} | 2-169 |
| :DIAG:SERVice:MODE□{ON 1} | 2-169 |
| :DIAG:SERVice:SOURce:ALCLoop□{OPEN CLOSE} | 2-169 |
| :DIAG:SERVice:SOURce:ATTenuator□{AUTO DB0 DB10 DB20 DB30 DB40 DB50 DB60} | 2-170 |
| :DIAG:SERVice:SOURce:GAIN:DAC:MODE□{AUTO MANual} | 2-170 |
| :DIAG:SERVice:SOURce:GAIN:DAC:VALue□<numeric> | 2-171 |
| :DIAG:SERVice:SOURce:LEVel:DAC:MODE□{AUTO MANual} | 2-171 |
| :DIAG:SERVice:SOURce:LEVel:DAC:VALue□<numeric> | 2-171 |
| :DIAG:SERVice:SOURce:MODE□{AUTO MANual} | 2-172 |
| :DIAG:SERVice:SYNThesizer:FLOCAL:MODE□{AUTO SINGle TRIPle} | 2-172 |
| :DIAG:SERVice:SYNThesizer:FN:MODE□{AUTO NARRow WIDE} | 2-172 |
| :DIAG:SERVice:SYNThesizer:FREQuency:OFFSet□<numeric> | 2-173 |
| :DIAG:SERVice:SYNThesizer:STEP:DAC:MODE□{AUTO MANual} | 2-173 |
| :DIAG:SERVice:SYNThesizer:STEP:DAC:VALue□<numeric> | 2-173 |
| :DIAG:SERVice:SYNThesizer:STEP:LOOP□{OPEN CLOSE} | 2-174 |
| :DIAG:SERVice:SYNThesizer:STEP:MODE□{AUTO MANual} | 2-174 |

| | |
|--|-------|
| :DIAG:SERvice:SYNThesizer:STEP:OUTPut□{OFF ON 0 1} | 2-174 |
| :DIAG:SERvice:SYNThesizer:STEP:POLarity□{AUTO POSitive NEGative} | 2-174 |
| :DIAG:TEST□<numeric> | 2-175 |
| :DIAG:TEST:CONTInue | 2-175 |
| :DIAG:TEST:EXECute | 2-175 |
| :DIAG:TEST:RESult?□<numeric> | 2-176 |
| A. Manual Changes | |
| Introduction | A-1 |
| Manual Changes | A-1 |
| Serial Number | A-2 |
| Change 1 | A-2 |
| Change 2 | A-2 |
| B. Command Summary | |
| C. SCPI Commands Summary | |
| D. Status Reporting | |
| OSPT, OSNT | D-5 |
| OSPT (Operation Status Positive Transition Filter) | D-5 |
| OSNT (Operation Status Negative Transition Filter) | D-5 |
| E. Trigger System | |
| Trigger System | E-1 |
| Analyzer Trigger System Configuration | E-1 |
| Idle State | E-2 |
| Initiate State | E-2 |
| Trigger Event Detection State | E-2 |
| Sequence Operation State | E-2 |
| F. Calibration Types and Standard Classes, and Calibration Arrays | |
| G. Key Codes | |
| H. Data Format and Data Levels | |
| Data Format | H-1 |
| Data Levels | H-2 |
| Marker Readout | H-3 |
| I. Waveform Analysis Commands | |
| Conventions and Definitions | I-2 |
| Waveform Analysis Setup Commands | I-3 |
| ANAOCH1 | I-3 |
| ANAOCH2 | I-3 |
| ANARANG | I-4 |
| ANARFULL | I-4 |
| ANAODATA | I-5 |
| ANAOMEMO | I-5 |
| THRR | I-6 |
| Maximum/Minimum/Mean Value Search Commands | I-7 |
| OUTPMAX? | I-7 |
| OUTPMIN? | I-7 |
| OUTPMINMAX? | I-8 |
| OUTPMEAN? | I-8 |

| | |
|--|------|
| PEAK? | I-9 |
| NEXPK? | I-10 |
| NUMLMAX? | I-11 |
| NUMLMIN? | I-11 |
| LMAX? | I-12 |
| LMIN? | I-12 |
| TARR? | I-13 |
| TARL? | I-14 |
| Ripple Analysis Commands | I-15 |
| RPLPP? | I-15 |
| RPLHEI? | I-17 |
| RPLRHEI? | I-18 |
| RPLLHEI? | I-19 |
| RPLENV? | I-20 |
| RPLMEA? | I-21 |
| RPLVAL? | I-22 |
| POLE? | I-23 |
| Filter and Resonator Analysis Commands | I-25 |
| OUTPFILT? | I-26 |
| OUTPXFIL? | I-28 |
| OUTPCFIL? | I-31 |
| OUTPRESO? | I-34 |
| OUTPRESR? | I-36 |
| OUTPRESF? | I-38 |
| OUTPCERR? | I-40 |
| Equivalent circuit analysis commands | I-42 |
| EQUCPARA? | I-43 |
| EQUCPARS? | I-46 |
| EQUC0? <i>value</i> | I-47 |
| EQUCPARS4? | I-49 |

Messages

Index

Figures

| | |
|---|-------|
| 1-1. Proper Use of the Colon and Semicolon | 1-4 |
| 2-1. Fixed length block format | 2-105 |
| 2-2. Procedure of executing commands to read/write data | 2-112 |
| A-1. Serial Number Plate | A-2 |
| D-1. Status Reporting Structure | D-1 |
| D-2. Example of Reading Status Byte (1) | D-2 |
| D-3. Example of Reading Status Byte (2) | D-2 |
| D-4. Example of Generating a Service Request (SRQ) | D-5 |
| E-1. Trigger System Configuration | E-1 |
| E-2. Inside an Trigger Event Detection State | E-2 |
| G-1. Key Codes | G-1 |
| H-1. Form 2 Data Transfer Format | H-1 |
| H-2. Form 3 Data Transfer Format | H-1 |
| I-1. THRR | I-6 |
| I-2. RPLPP? | I-15 |
| I-3. RPLHEI? | I-17 |
| I-4. RPLRHEI? | I-18 |
| I-5. RPLLHEI? | I-19 |
| I-6. RPLENV? | I-20 |
| I-7. RPLMEA? | I-21 |
| I-8. RPLVAL? | I-22 |
| I-9. POLE? | I-23 |
| I-10. OUTPFILT? | I-26 |
| I-11. OUTPXFIL? | I-29 |
| I-12. OUTPCFIL? | I-32 |
| I-13. OUTPRESO? | I-34 |
| I-14. OUTPRESR? | I-36 |
| I-15. OUTPRESF? | I-38 |
| I-16. OUTPCERR? | I-40 |
| I-17. Six-Device Equivalent Circuit of Crystal Resonator | I-43 |
| I-18. Four-Device Equivalent Circuit of Crystal Resonator | I-49 |
| I-19. Admittance Characteristic Circle Diagram | I-50 |

Tables

| | |
|---|-----|
| A-1. Manual Changes by Serial Number | A-1 |
| A-2. Manual Changes by Firmware Version | A-1 |
| D-1. Status Bit Definitions of the Status Byte (STB) | D-2 |
| D-2. Status Bit Definitions of the Standard Event Status Register (ESR) | D-3 |
| D-3. Status Bit Definitions of the Event Status Register B (ESB) | D-4 |
| D-4. Status Bit Definitions of the Operation Status Register (OSR) | D-4 |
| F-1. Calibration Types and Standard Classes | F-1 |
| F-2. Calibration Array | F-2 |
| H-1. Marker Readout | H-3 |

Introduction

This manual provides a reference for the General Purpose Interface Bus (GPIB) commands used to control the 4396B Network/Spectrum/Impedance Analyzer (analyzer). These commands are implemented using an external controller or the HP instrument BASIC.

The following is a brief description of each chapter and appendix.

Chapter 2 explains all the GPIB commands.

Appendix A contains the information required to adapt this manual to earlier versions or configurations of the analyzer than the current printing date of this manual.

Appendix B lists all the GPIB commands sorted by the function (key label).

Appendix C lists all the GPIB commands in alphabetical order for the Standard Commands for Programmable Instruments (SCPI) commands.

Appendix D provides information about the status reporting structure for service request functions.

Appendix E provides information about the trigger system, which corresponds to the SCPI standard.

Appendix F describes the calibration types and the standard classes, and the calibration coefficients.

Appendix G provides the front-panel key codes for the KEY GPIB commands.

Appendix H provides information about data formats and data levels.

Appendix I provides detail information about the waveform analysis commands.

Error Messages lists all error messages with an explanation for each error.

See the *GPIB Programming Guide* for introduction to using the analyzer's GPIB commands and for a description of how the HP instrument BASIC works with the analyzer.

Note



You should become familiar with the operation of the analyzer before you attempt to control it using GPIB commands. See the following documents which are better suited to this task.

For more information concerning the operation of the analyzer, see the following:

User's Guide

Task Reference

Function Reference

Option 010 Operating Handbook for impedance measurement mode.

Note

This manual is not intended to teach the BASIC programming language or the Standard Commands for Programmable Instruments (SCPI) commands. It also does not discuss GPIB theory. See the following documents that are better suited to these tasks.

For more information concerning BASIC, see the manual set for the BASIC version being used:

BASIC Programming Techniques

BASIC Language Reference

For more information concerning SCPI, see the following:

Beginner's Guide to SCPI

For more information concerning GPIB operation, see the following:

BASIC Interfacing Techniques

Tutorial Description of the General Purpose Interface Bus

Condensed Description of the General Purpose Interface Bus

GPIB Commands

Most of the analyzer's functions have two corresponding GPIB commands. One is unique to the analyzer (called a **Simple command**) and another corresponds to the Standard Commands for Programmable Instruments (called a **SCPI command**). You can use both commands in one program.

- For example, the command to select the analyzer type is as follows:

```
Simple command:  NA
SCPI command:   :INSTRument:TYPE NA
```

The analyzer also has other commands (called **Common command**) that are not measurement related. These include commands for functions such as status register control or synchronization.

- For example, the command to clear status registers is as follows:

```
*CLS
```

Simple Commands

All the analyzer's front-panel keys have corresponding GPIB commands. The names of the simple commands are derived from their front panel key titles (where possible). Commands that have no equivalent front-panel key use a similar convention based on the common name of the function.

SCPI Commands

SCPI is the instrument command language for controlling instrument that goes beyond IEEE 488.2 standard to address a wide variety of instrument functions in a standard manner.

Common Commands

All common commands begin with an asterisk (*). Common commands are defined by IEEE 488.2.

SCPI Subsystem Commands

Subsystem commands include all measurement functions and some general purpose functions. Each subsystem is a set of commands that roughly corresponds to a functional block inside the instrument.

Subsystem commands have a hierarchical structure, called a **command tree**, that consists of several key words separated by a colon between each word.

Subsystem Command Tree

The top of the subsystem command tree is called the **root command**, or simply the **root**. To reach the low-level commands, you must specify a particular **path** (like a DOS file directory path). After Power ON or after presetting, the current path is set to the root. The path settings are changed as follows:

Program Message Terminator A program message terminator, such as <new line> character, sets the current path to the root.

Colon (:) When a colon is placed between two command mnemonics, the colon moves the current path down one level on the command tree. When the colon is the first character of a command, it specifies that the following command mnemonic is a root-level command.

Semicolon (;) A semicolon separates two commands in the same message without changing the current path.

Figure 1-1 shows examples of how to use the colon and semicolon to navigate efficiently through the command tree.

Common commands, such as *RST, are not part of any subsystem. The analyzer interprets them in the same way, regardless of the current path setting.

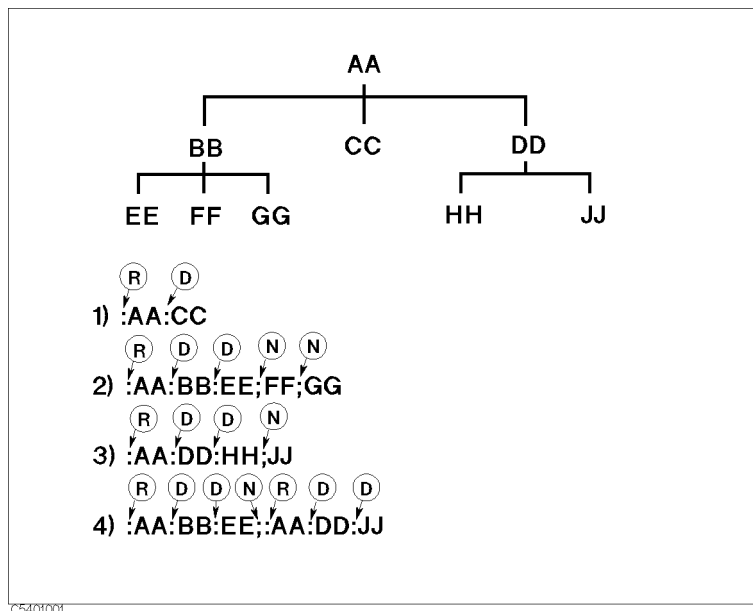


Figure 1-1. Proper Use of the Colon and Semicolon

- \textcircled{R} sets the current path to the root.
- \textcircled{D} moves the current path down one level.
- \textcircled{N} does not change the current path.

Command Abbreviations

Many commands have a long and a short form. In this manual, all commands are spelled out in the long form. The short form is obtained by deleting the lower case letters.

For example, the short form of :INITiate is :INIT and the long form of it is :INITIATE. (SCPI does *not* accept anything in between, such as :INITIA.)

Program Message Syntax

This section provides the construction of program messages. A program message is the message that you send from a computer to an instrument. Program messages consist of commands combined with appropriate punctuation and program message terminators.

Case

Letter cases (upper and lower) are ignored.

Program Message Terminator

A program message must end with one of the three **program message terminators** , <new line>, <^END>, or <new line><^END>. <^END> means that End Of Identify (EOI) is asserted on the GPIB interface at the same time the preceding data byte is sent. For example, the HP BASIC OUTPUT statement is automatically sent after last data byte. If you are using a PC, you can usually configure your system to send whatever terminator you specify.

Multiple Messages

To send more than one command in the same message, you must separate them with a semicolon:

```
NA;CHAN1
```

Query and Response Message Syntax

All commands can be queried except the commands described as “no query” in the command reference. To send a query message, add ? after the last command mnemonic.

```
NA?
```

A response message may contain both commas and semicolons as separators. When a single query command returns multiple values, a comma is used to separate each data item. When multiple queries are sent in the same message, the group of data items corresponding to each query are separated by a semicolon. For example, the fictitious query :QUERY1?;QUERY2? might return a response message of:

```
<data1>,<data1>;<data2>,<data2>
```

After the message, <new line><^END> is always sent as a response message terminator.

Parameters

There must be a <space> between the last command mnemonic and the first parameter in a subsystem command.

CENT□*parameter*

□ means a space (ASCII character (decimal 32)).

If you send more than one parameter with a single command, each parameter must be separated by a comma.

Parameter Types

The analyzer accepts commands and parameters in various formats and responds to a particular query in a predefined and fixed format. Each command reference contains information about the parameter types available for the individual commands.

- <numeric> represents numeric parameters as follows:

| | |
|-------------|---------------------------------------|
| 100 | no decimal point required |
| 100. | fractional digits optional |
| -1.23, +235 | leading signs allowed |
| 4.56e□3 | space allowed after e in exponentials |
| -7.89E-01 | use either E or e in exponentials |
| .5 | digits left of decimal point optional |

The analyzer setting programmed with a numeric parameter can assume a finite number of values, so the analyzer automatically rounds off the parameter. For example, the analyzer has a programmable input attenuator value. If you specified 50.1, it would be rounded off to 50.

Query response of <numeric_value> is always a numeric value in <NR1> (integer) or <NR3> (floating point) format.

- Suffix

When a command has a specified suffix, the suffix multiplier and suffix units can be used with parameters as follows. (The suffix multiplier must be used with the suffix unit.):

| | |
|--------------|---|
| Frequency: | HZ (Hz; default), KHZ (kHz), MAHZ or MHZ (MHz), GHZ (GHz) |
| Power: | DBM (dBm; default) |
| Attenuator: | DB (dB; default) |
| Time: | S (second; default), MS (ms), US (μ s), NS (ns), PS (ps), FS (fs) |
| Scale: | DB (dB), DEG ($^{\circ}$), S (second), DBM (dBm), DBV (dBV), DBUV ($\text{dB}\mu\text{V}$), W (watt), V (Volt), OHM (Ω), SIE (siemens) |
| Phase: | DEG ($^{\circ}$; default) |
| Capacitance: | F (farad; default) |
| Percent: | PCT (%; default) |
| Impedance: | OHM (Ω ; default), KOHM (k Ω) |
| Loss: | DB (dB; default) |

The suffix is optional and can be omitted.

- <string> is a string parameter that contains ASCII characters. A string must begin with a single quote (ASCII 39 decimal) or a double quote (ASCII 34 decimal) and end with the same corresponding character, a single or double quote. The quote to mark the beginning and end of the string is called the delimiter. You can include the delimiter as part of the string by typing it twice without any characters in between.

Example of *<string>* TXT,

```
OUTPUT @Meter;"ASCE 'TXT'"      using single quote
```

```
OUTPUT @Meter;"ASCE ""TXT"""    using double quote
```

The query response is the string between double quote delimiters.

- *<block>* is typically used to transfer large quantities of related data. *<block>* can be sent as the definite length blocks.

General form of block parameters:

```
#<num_digits><num_bytes><data bytes>
```

The single decimal digit *<num_digits>* specifies how many digits are contained in *<num_bytes>*. The decimal number *<num_bytes>* specifies how many data bytes will follow in *<data bytes>*.

Example of *<block>* ABC+XYZ,

```
OUTPUT @Meter;"#17ABC+XYZ"
```

(1 means one digit follows, 7 means seven bytes follow.)

Commands Reference

This chapter provides a reference for the GPIB commands of the analyzer. Use this information as a reference to the syntax requirements and general function of the individual commands.

This chapter is organized as follows:

- Simple Commands
- Common Commands
- SCPI Commands With No Equivalent Simple Command
- Service Related Commands

Within each group the commands are listed in alphabetical order. See Appendix B for a functional list of the commands. See Appendix C for a list of the SCPI commands in alphabetical order.

See the *Function Reference* for the details of each function. See *GPIB Programming Guide* for an introduction to using the analyzer's GPIB Commands.

The following conventions and definitions are used to describe the commands.

① → **AVER**□{OFF|ON|0|1}

② → Turns the averaging function ON or OFF for the active channel. (**AVERAGING** on off under **(Bw/Avg)**)

③ →

| Parameter | Description |
|-----------|------------------------|
| OFF or 0 | Averaging function OFF |
| ON or 1 | Averaging function ON |

④ → ■ Query Response

{0|1} <new line> <END>

⑤ → ■ Equivalent SCPI Command

:SENSe:AVERage[:STATe] {OFF|ON|0|1}

⑥ → ■ Example

OUTPUT 717;"AVER ON"

OUTPUT 717;"AVER?"

ENTER 717;A

OUTPUT 717;":SENSe:AVER ON"

OUTPUT 717;":SENSe:AVER?"

ENTER 717;A

| | |
|---|---|
| ① | <p>Command name and required parameter.</p> <p>Upper case bold characters represent the command that must appear exactly as shown with no embedded spaces. Upper and lower case characters are equivalent.</p> <p>A constant or a pre-assigned simple or complex numeric or string variable transferred to the analyzer. There must be a space between it and the code. (□ indicates a space.)</p> <p>Characters enclosed in the { } brackets are qualifiers attached to the root mnemonic. There can be no spaces or symbols between the root mnemonic and its appendage. For example, {OFF ON 0 1} means OFF, ON, 0, or 1, and {1-4} means 1, 2, 3, or 4.</p> |
| ② | <p>Description.</p> <p>Key or softkey that has the same function is shown in the brackets. The brackets may include more additional information.</p> |
| ③ | <p>Parameter description of the Simple command and SCPI command.</p> |
| ④ | <p>Query response of the Simple command and SCPI command.</p> <p>If the query response of the SCPI command differs from the response of the Simple command, the query response of the SCPI command is described in "Equivalent SCPI Command."</p> |
| ⑤ | <p>Equivalent SCPI command to the Simple command.</p> <p>See "SCPI Commands" in Chapter 1 for more information about the SCPI command. Square brackets indicate that the enclosed information is optional.</p> |
| ⑥ | <p>Example of the usage of the Simple command and SCPI command (including their query forms).</p> |

Simple Commands

ADDRCONT□<numeric>

Sets the GPIB address the analyzer will use to communicate with the external controller.
(ADDRESS: CONTROLLER under (Local))

| Parameter | Range | Unit |
|-----------|---------|------|
| <numeric> | 0 to 30 | |

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

:SYSTEM:COMMunicate:GPIB2:ADDRESS□<numeric>

ANA0CH1

Selects channel 1 for waveform analysis. For details, see “ANA0CH1” in Appendix I.
(Instrument BASIC EXECUTE executable; No equivalent SCPI command)

■ Query Response

| Parameter | Description |
|-----------|--------------------------------|
| OFF or 0 | Analysis for channel 1 is off. |
| ON or 1 | Analysis for channel 1 is on. |

ANA0CH2

Selects channel 2 for waveform analysis. For details, see “ANA0CH2” in Appendix I.
(Instrument BASIC EXECUTE executable; No equivalent SCPI command)

■ Query Response

| Parameter | Description |
|-----------|--------------------------------|
| OFF or 0 | Analysis for channel 1 is off. |
| ON or 1 | Analysis for channel 1 is on. |

ANAODATA

Selects a data trace for waveform analysis. For details, refer to “ANAODATA” in Appendix I.
(Instrument BASIC EXECUTE executable; No equivalent SCPI command)

■ Query Response

| Parameter | Description |
|-----------|---------------------------------|
| OFF or 0 | Analysis for data trace is off. |
| ON or 1 | Analysis for data trace is on. |

ANAOMEMO

Selects a memory trace for waveform analysis. For details, refer to “ANAOMEMO” in Appendix I. (Instrument BASIC EXECUTE executable; No equivalent SCPI command)

■ Query Response

| Parameter | Description |
|-----------|-----------------------------------|
| OFF or 0 | Analysis for memory trace is off. |
| ON or 1 | Analysis for memory trace is on. |

ANARANG \square <numeric1>, <numeric2>

Sets the waveform analysis stimulus range by entering the START and STOP values. For details, see “ANARANG” in Appendix I. (Instrument BASIC EXECUTE executable; No equivalent SCPI command)

| Parameter | Range | Description (Unit) |
|------------|---|---|
| <numeric1> | 100000 (= 100 k) to 1.82×10^9 (= 1.82 G) (Network and Impedance analyzer) 0 to 1.82×10^9 (= 1.82 G) (Spectrum analyzer) -70 to 20 | Start Frequency (Hz) Hz (frequency) Start Power (dBm) |
| <numeric2> | 100000 (= 100 k) to 1.82×10^9 (= 1.82 G) (Network and Impedance analyzer) 0 to 1.82×10^9 (= 1.82 G) (Spectrum analyzer) -70 to 20 | Stop Frequency (Hz) Hz (frequency) Stop Power (dBm) |

■ Query Response

{numeric1},{numeric2}<new line><^END>

ANARFULL

Sets the analysis range equal to the full stimulus range. For details, see “ANARFULL” in Appendix I. (Instrument BASIC EXECUTE executable; No equivalent SCPI command)

ATT \square <numeric> [DB]

Changes the input attenuation when the S input is selected. Because the attenuators at the R, A, and B inputs are fixed, if either R, A, or B is selected, you can enter the value but not change. (Spectrum analyzer only) (ATTEN under Scale Ref)

| Parameter | Range | Unit |
|-----------|---------------------------|------|
| <numeric> | 0, 10, 20, 30, 40, 50, 60 | dB |

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

```
:SENSe:POWer:AC:ATTenuation:AUTO  $\square$  {OFF|0}  
:SENSe:POWer:AC:ATTenuation  $\square$  <numeric>
```


■ Example

```
OUTPUT 717;"ATT 10DB"
OUTPUT 717;":SENS:POW:AC:ATT:AUTO OFF"
OUTPUT 717;":SENS:POW:AC:ATT 10"
```

ATTAUTO□{OFF|ON|0|1}

Sets the automatic and manual spectrum analyzer input attenuator of the S input. (Spectrum analyzer only) (ATTEN AUTO man under [Scale Ref](#))

| Parameter | Description |
|-----------|----------------------|
| OFF or 0 | Manual attenuator |
| ON or 1 | Automatic attenuator |

■ Query Response

{0|1} <new line><END>

■ Equivalent SCPI Command

:SENSe:POWer:AC:ATTenuation:AUTO□{OFF|ON|0|1}

ATTP1□<numeric>[DB]

Controls the attenuation at port 1 of an S-parameter Test Set connected to the analyzer. (Network analyzer only) (ATTENUATOR PORT 1 under [Source](#))

| Parameter | Range | Unit |
|-----------|-------------------------------|------|
| <numeric> | 0, 10, 20, 30, 40, 50, 60, 70 | dB |

■ Query Response

{numeric} <new line><END>

■ Equivalent SCPI Command

:OUTPut:ATTenuation1□<numeric>

ATTP2□<numeric>[DB]

Controls the attenuation at port 2 of an S-parameter Test Set connected to the analyzer. (Network analyzer only) (ATTENUATOR PORT 2 under [Source](#))

| Parameter | Range | Unit |
|-----------|-------------------------------|------|
| <numeric> | 0, 10, 20, 30, 40, 50, 60, 70 | dB |

■ Query Response

{numeric} <new line><END>

■ Equivalent SCPI Command

:OUTPut:ATTenuation2□<numeric>

AUTO

Brings the trace data, defined by the SCAF command, in view on the display. (Network and impedance analyzer only) (AUTO SCALE under (Scale Ref); No query)

■ Equivalent SCPI Command

```
:DISPlay[:WINDow]:TRACe{[1]|2}:Y[:SCALe]:AUTO|ONCE  
(TRACe[1] for the data trace; TRACe2 for the memory trace.)
```

AVER|{OFF|ON|0|1}

Turns the averaging function ON or OFF for the active channel. (AVERAGING ON off under (Bw/Avg))

| Parameter | Description |
|-----------|------------------------|
| OFF or 0 | Averaging function OFF |
| ON or 1 | Averaging function ON |

■ Query Response

```
{0|1} <new line><^END>
```

■ Equivalent SCPI Command

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

AVERFACT|<numeric>

Makes the averaging factor for the active function. (AVERAGING FACTOR under (Bw/Avg))

| Parameter | Range | Unit |
|-----------|----------|------|
| <numeric> | 1 to 999 | |

■ Query Response

```
{numeric} <new line><^END>
```

■ Equivalent SCPI Command

```
:SENSe:AVERAge:COUNT|<numeric>
```

AVERREST

Resets the sweep-to-sweep averaging and restarts the sweep count at 1 at the beginning of the next sweep. (AVERAGING RESTART under (Bw/Avg); No query)

■ Equivalent SCPI Command

```
:SENSe:AVERAge:CLear
```

BACI□<numeric>[PCT]

Sets the background intensity of the display as a percent of the white level.

(BACKGROUND INTENSITY under (Display))

| Parameter | Range | Unit |
|-----------|---------------------------|------|
| <numeric> | 0 to 100 (simple command) | % |
| <numeric> | 0 to 1 (SCPI command) | |

- Query Response

{numeric} <new line><^END>

- Equivalent SCPI Command

:DISPlay:CONTrast□<numeric>

BEEPDONE□{OFF|ON|0|1}

Sets an annunciator that sounds to indicate completion of certain operations such as calibration or instrument state save. (BEEP DONE ON off under (System))

| Parameter | Description |
|-----------|---------------------------------|
| OFF or 0 | Operation completion beeper OFF |
| ON or 1 | Operation completion beeper ON |

- Query Response

{0|1} <new line><^END>

- Equivalent SCPI Command

:SYSTem:BEEPer1:STATe□{OFF|ON|0|1}

BEEPFAIL□{OFF|ON|0|1}

Turns the limit fail beeper ON or OFF. When the limit testing is ON and the fail beeper is ON, a beep is emitted each time a limit test is performed and a failure is detected.

(BEEP FAIL ON off under (System))

| Parameter | Description |
|-----------|-----------------------|
| OFF or 0 | Limit fail beeper OFF |
| ON or 1 | Limit fail beeper ON |

- Query Response

{0|1} <new line><^END>

- Equivalent SCPI Command

:CALCulate:LIMit:BEEPer[:STATe]□{OFF|ON|0|1}

BEEPWARN \square {OFF|ON|0|1}

Sets the warning annunciator. When the annunciator is ON, it sounds a warning when a cautionary message is displayed. (BEEP WARN ON off under System)

| Parameter | Description |
|-----------|--------------------|
| OFF or 0 | Warning beeper OFF |
| ON or 1 | Warning beeper ON |

■ Query Response

{0|1} <new line><^END>

■ Equivalent SCPI Command

:SYSTEM:BEEPer2:STATe \square {OFF|ON|0|1}

BLIGHT \square {OFF|ON|0|1}

Sets backlighting the LCD screen ON or OFF.

| Parameter | Description |
|-----------|------------------|
| OFF or 0 | Backlighting OFF |
| ON or 1 | Backlighting ON |

■ Query Response

{0|1} <new line><^END>

■ Equivalent SCPI Command

:DISPlay:BACKlight \square {OFF|ON|0|1}

BOTV \square <numeric>

Defines the bottom border of the display and adjusts the scale value. (BOTTOM VALUE under Scale Ref); Impedance analyzer only)

| Parameter | Range | Unit |
|-----------|--|----------------------------------|
| <numeric> | -1 \times 10 ⁹ to 1 \times 10 ⁹ -200 \times 10 ⁶ to 200 \times 10 ⁶ | y-axis unit y-axis unit (Log) |

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

:DISPlay[:WINDow]:TRACe{1|2}:Y[:SCALe]:BOTTom \square <numeric>

BW □ <numeric> [HZ|KHZ|MAHZ]

Sets the bandwidth value for IF bandwidth reduction, or sets the IF bandwidth of the list sweep table. (Network analyzer and Impedance analyzer) (IF BW under Bw/Avg), or IF BW under Sweep)

Sets the bandwidth value for the resolution bandwidth reduction, or sets the resolution bandwidth of the list sweep table. (Spectrum analyzer) (RES BW under Bw/Avg), or RES BW under Sweep)

| Parameter | Range | Unit |
|-----------|--|------|
| <numeric> | 10, 30, 100, 300, 1000 (= 1 k), 3000 (= 3 k), 10000 (= 10 k), 40000 (= 40 k) (network analyzer and impedance analyzer) 1, 3, 10, 30, 100, 300, 1000 (= 1 k), 3000 (= 3 k), 10000 (= 10 k), 30000 (= 30 k), 100000 (= 100 k), 300000 (= 300 k), 1000000 (= 1 M), 3000000 (= 3 M) (spectrum analyzer) | Hz |

■ Query Response

{numeric} <new line><END>

■ Equivalent SCPI Command

Network and impedance analyzer :SENSe:BANDwidth[:RESolution] □ <numeric>

Spectrum analyzer :SENSe:BANDwidth[:RESolution]:AUTO □ {OFF|0}
:SENSe:BANDwidth[:RESolution] □ <numeric>

IF BW, RES BW under Sweep :SENSe:LIST:SEGment:BANDwidth □ <numeric>

BWAUTO □ {OFF|ON|0|1}

Sets either the automatic or manual resolution bandwidth ON. (Spectrum analyzer only) (RES BW AUTO man under Bw/Avg)

| Parameter | Description |
|-----------|---------------------------------|
| OFF or 0 | Manual resolution bandwidth. |
| ON or 1 | Automatic resolution bandwidth. |

■ Query Response

{0|1} <new line><END>

■ Equivalent SCPI Command

:SENSe:BANDwidth[:RESolution]:AUTO □ {OFF|ON|0|1}

BWAUTO \square {OFF|ON|0|1}

BWSRAT \square \langle numeric \rangle

Sets the RBW/SPAN ratio that specifies the resolution bandwidth in the AUTO mode. (Spectrum analyzer only) (RBW/SPAN RATIO under (Bw/Avg))

| Parameter | Range | Unit |
|-----------------------------|---------------------------------------|------|
| \langle numeric \rangle | 0.01 to 10 (of SPAN) (simple command) | % |
| \langle numeric \rangle | 0.0001 to 0.1 (SCPI command) | |

■ Query Response

{numeric} \langle new line \rangle \langle ^END \rangle

■ Equivalent SCPI Command

:SENSe:BANDwidth[:RESolution]:RATio \square \langle numeric \rangle

C0 \square \langle numeric \rangle

Enters the C_0 term, which is the constant term of the capacitance model equation. (Network and impedance analyzer only) (C0 under (Cal); No query)

| Parameter | Range | Unit |
|-----------------------------|---------------------------------------|------|
| \langle numeric \rangle | -10000 to 10000 ($\times 10^{-15}$) | F |

■ Equivalent SCPI Command (Query)

:SENSe:CORRection:CKIT:STANdard:C0 \square \langle numeric \rangle

■ Query Response

{numeric} \langle new line \rangle \langle ^END \rangle

C1 \square \langle numeric \rangle

Enters the C_1 term, which is the constant term of the capacitance model equation. (Network and impedance analyzer only) (C1 under (Cal); No query)

| Parameter | Range | Unit |
|-----------------------------|---------------------------------------|------|
| \langle numeric \rangle | -10000 to 10000 ($\times 10^{-27}$) | F/Hz |

■ Equivalent SCPI Command (Query)

:SENSe:CORRection:CKIT:STANdard:C1 \square \langle numeric \rangle

■ Query Response

{numeric} \langle new line \rangle \langle ^END \rangle

C2 <numeric>

Enters the C₂ term, which is the constant term of the capacitance model equation. (Network and impedance analyzer only) (C2 under Cal); No query)

| Parameter | Range | Unit |
|-----------|--|-------------------|
| <numeric> | -10000 to 10000 (× 10 ⁻³⁶) | F/Hz ² |

- Equivalent SCPI Command (Query)
 - :SENSe:CORRection:CKIT:STANdard:C2<numeric>
- Query Response
 - {numeric} <new line><^END>

CALCASSI

Shows the tabular listing of the calibration kit class assignment. (Network and impedance analyzer only) (CLASS ASSIGNMENT under Copy); No query)

- Equivalent SCPI Command
 - :DISPlay[:WINDow]:TEXT3:PAGE1
 - :DISPlay[:WINDow]:TEXT3:STATe{ON|1}

CALECPARA

Calculates and displays the equivalent circuit parameters. (CALCULATE EQV PARAMS under Display); No query; Impedance analyzer only)

- Equivalent SCPI Command
 - :CALCulate:EVALuate:EPARameters
 - :DISPlay[:WINDow]:TEXT18:STATe{ON|1}

CAL{NONE|RESP|RAI|S111|S221|FUL2|ONE2|IMP}

Selects the measurement calibration type. (Network analyzer and impedance analyzer only) (CALIBRATE:NONE, RESPONSE, RESPONSE & ISOL'N, S11 1-PORT, S22 1-PORT, FULL 2-PORT, ONE PATH 2-PORT under Cal) of network analyzer mode or CALIBRATE MENU under Cal) of impedance analyzer mode.)

| Parameter | Description |
|-----------|--|
| NONE | No calibration (Network and impedance analyzer only) |
| RESP | Response measurement calibration (Network analyzer only) |
| RAI | Response and isolation measurement calibration (Network analyzer only) |
| S111 | 1-Port measurement calibration at port 1 (Network analyzer only) |
| S221 | 1-Port measurement calibration at port 2 (Network analyzer only) |
| FUL2 | Full 2-Port measurement calibration (Network analyzer only) |
| ONE2 | One-path 2-Port measurement calibration (Network analyzer only) |
| IMP | Calibration of the impedance analyzer mode. (Impedance analyzer only) |

- Query Response

CALI {NONE|RESP|RAI|S111|S221|FUL2|ONE2|IMP}

{NONE|RESP|RAI|S111|S221|FUL2|ONE2|IMP} <new line><END>

■ Equivalent SCPI Command

CALI NONE :SENSe:CORRection1:COLLect:METhod NONE
CALI RESP :SENSe:CORRection1:COLLect:METhod RESPonse
CALI RAI :SENSe:CORRection1:COLLect:METhod RAIsol
CALI S111 :SENSe:CORRection1:COLLect:METhod S111
CALI S221 :SENSe:CORRection1:COLLect:METhod S221
CALI FUL2 :SENSe:CORRection1:COLLect:METhod TPORt
CALI ONE2 :SENSe:CORRection1:COLLect:METhod OPTPort
CALI IMP :SENSe:CORRection1:COLLect:METhod IMPedance

■ Example

```
OUTPUT 717;"CALI NONE"  
  
OUTPUT 717;"CALI?"  
ENTER 717;A$  
  
OUTPUT 717;":SENS:CORR:COLL:METH NONE"  
  
OUTPUT 717;":SENS:CORR:COLL:METH?"  
ENTER 717;A$
```

CALK {APC7|APC35|N50|N75|USED}

Selects one of the default calibration kits available for different connector types. (Network and impedance analyzer only) (CAL KIT:7mm, 3.5mm, N 50 ohm, N 75 ohm, or USER KIT under (Cal))

| Parameter | Description |
|-----------|--------------|
| APC7 | 7 mm |
| APC35 | 3.5 mm |
| N50 | Type-N 50 Ω |
| N75 | Type-N 75 Ω |
| USED | User-defined |

■ Query Response

{APC7|APC35|N50|N75|USED} <new line><END>

■ Equivalent SCPI Command

CALK APC7 :SENSe:CORRection:CKIT APC7
CALK APC35 :SENSe:CORRection:CKIT APC35
CALK N50 :SENSe:CORRection:CKIT N50
CALK N75 :SENSe:CORRection:CKIT N75
CALK USED :SENSe:CORRection:CKIT UDEFined

■ Example

```
OUTPUT 717;"CALC APC7"  
  
OUTPUT 717;"CALC?"  
ENTER 717;A$  
  
OUTPUT 717;":SENS:CORR:CKIT APC7"
```


CENT□<numeric>[HZ|KHZ|MAHZ|GHZ|DBM]

CALS□<numeric>

Provides the tabular listing of the standard definitions. (Network and impedance analyzer only) (STD NO.1 to STD NO.8 under (Copy); No query)

| Parameter | Range | Unit |
|-----------|--------|------|
| <numeric> | 1 to 8 | |

■ Equivalent SCPI Command

```
CALS□1      :DISPlay[:WINDow]:TEXT4:PAGE□1
             :DISPlay[:WINDow]:TEXT4:STATe□{0N|1}

CALS□2      :DISPlay[:WINDow]:TEXT5:PAGE□1
             :DISPlay[:WINDow]:TEXT5:STATe□{0N|1}

CALS□3      :DISPlay[:WINDow]:TEXT6:PAGE□1
             :DISPlay[:WINDow]:TEXT6:STATe□{0N|1}

CALS□4      :DISPlay[:WINDow]:TEXT7:PAGE□1
             :DISPlay[:WINDow]:TEXT7:STATe□{0N|1}

CALS□5      :DISPlay[:WINDow]:TEXT8:PAGE□1
             :DISPlay[:WINDow]:TEXT8:STATe□{0N|1}

CALS□6      :DISPlay[:WINDow]:TEXT9:PAGE□1
             :DISPlay[:WINDow]:TEXT9:STATe□{0N|1}

CALS□7      :DISPlay[:WINDow]:TEXT10:PAGE□1
             :DISPlay[:WINDow]:TEXT10:STATe□{0N|1}

CALS□8      :DISPlay[:WINDow]:TEXT11:PAGE□1
             :DISPlay[:WINDow]:TEXT11:STATe□{0N|1}
```

CBRI□<numeric>[PCT]

Adjusts the brightness of the color being modified. (BRIGHTNESS under (Display); No equivalent SCPI command)

| Parameter | Range | Unit |
|-----------|----------|------|
| <numeric> | 0 to 100 | % |

■ Query Response

{numeric} <new line><END>

CENT□<numeric>[HZ|KHZ|MAHZ|GHZ|DBM]

Defines the center value of the sweep range, or the center value of the segment to be edited in the list sweep table. (Center, or CENTER under (Sweep))

CENT \square *<numeric>* [HZ|KHZ|MAHZ|GHZ|DBM]

| Parameter | Range | Unit |
|------------------------|--|---|
| <i><numeric></i> | 100k to 1.82G (network analyzer and impedance analyzer) 0 to 1.8199999995 G (spectrum analyzer) 0 to 1.8199999990234 G (spectrum analyzer with span = 195mHz) 0 to 1.82 G (spectrum analyzer with span = 0Hz) -60 to +20 (network analyzer and impedance analyzer) | Hz (frequency) dBm (power) |

■ Query Response

{*numeric*} <new line><^END>

■ Equivalent SCPI Command

Center :SENSe:FREQuency:CENTer \square *<numeric>* (frequency) or
:SOURce:POWer:CENTer \square *<numeric>* (power)

CENTER under **Sweep** :SENSe:LIST:SEGment:FREQuency:CENTer \square *<numeric>*
(List sweep table)

■ Example

```
OUTPUT 717;"CENT 899.95MAHZ"
```

CHAD \square *<string>*

Specifies changing the current directory of a DOS format disk. (**CHANGE DIRECTORY** under **Save**); No query

| Parameter | Description |
|-----------------------|----------------|
| <i><string></i> | Directory path |

■ Equivalent SCPI Command

:MMEMory:CDIRectory \square *<string>*

■ Example

```
OUTPUT 717;"CHAD "".."""
```

CHAN1

Selects channel 1 as the active channel. (**Chan 1**)

■ Query Response

{0|1} <new line><^END>

■ Equivalent SCPI Command

:INSTrument[:SElect] \square CH2 or :INSTrument:NSElect \square 2
:INSTrument:STATe \square OFF
:INSTrument[:SElect] \square CH1 or :INSTrument:NSElect \square 1
:INSTrument:STATe \square ON

■ Example

```
OUTPUT 717;"CHAN1"
OUTPUT 717;":INST CH2"
OUTPUT 717;":INST:STAT OFF"
OUTPUT 717;":INST CH1"
OUTPUT 717;":INST:STAT ON"
```

CHAN2

Selects channel 2 as the active channel. ((Chan 2))

■ Query Response

{0|1} <new line><^END>

■ Equivalent SCPI Command

```
:INSTrument[:SElect] CH1 or :INSTrument:NSElect 1
:INSTrument:STATe OFF
:INSTrument[:SElect] CH2 or :INSTrument:NSElect 2
:INSTrument:STATe ON
```

CIRF{RI|LIN|LOG|RX|GB|SWR}

Selects format to readout the value of a Smith, polar, or admittance chart using markers. (Network and impedance analyzer only) (REAL IMAG , LIN MAG PHASE , LOG MAG PHASE , R+jX , G+jB , SWR PHASE under (Utility))

| Parameter | Description |
|-----------|---------------------------------|
| RI | Real and imaginary form |
| LIN | Linear magnitude and phase form |
| LOG | Log magnitude and phase form |
| RX | Complex impedance form (R+jX) |
| GB | Complex admittance form (G+jB) |
| SWR | SWR and phase form |

■ Query Response

{RI|LIN|LOG|RX|GB|SWR} <new line><^END>

■ Equivalent SCPI Commands

```
CIRF RI :CALCulate:EVALuate:R:FORMat RIMaginary
CIRF LIN :CALCulate:EVALuate:R:FORMat MLIPhase
CIRF LOG :CALCulate:EVALuate:R:FORMat MLOPhase
CIRF RX :CALCulate:EVALuate:R:FORMat RX
CIRF GB :CALCulate:EVALuate:R:FORMat GB
CIRF SWR :CALCulate:EVALuate:R:FORMat SWRPhase
```

■ Example

```
OUTPUT 717;"CIRF GB"
```

CLAD

Completes the class assignment and stores it. (Network and impedance analyzer only)
(`CLASS DONE (SPE'D)` under `(Cal)`; No query)

- Equivalent SCPI Command

```
:SENSe:CORRection:CKIT:SAVE CLASs
```

- Example

```
OUTPUT 717;"CLAD"  
OUTPUT 717;":SENS:CORR:CKIT:SAVE CLAS"
```

CLASIMP{A|B|C}

Selects and acquires the impedance calibration classes. (`CALIBRATION OPEN`, `SHORT`, or `LOAD` under `(Cal)`, respectively; No query; Impedance analyzer only)

The order in which you acquire the `OPEN`, `SHORT`, and `LOAD` is changable. You can suspend a calibration sequence and do a different operation, and then resume the calibration sequence.

- Equivalent SCPI Command

```
:SENSe:CORRection1:COLLect[:ACQuire] IMP{A|B|C}
```

CLASS11{A|B|C}

Selects port 1 (S11) calibration standard class: S11A (open), S11B (short), or S11C (load). (Network analyzer only) (`[S11] : OPEN`, `SHORT`, `LOAD` under `(Cal)`; No query)

- Equivalent SCPI Command

```
:SENSe:CORRection:COLLect[:ACQuire] STANdard{1|2|3}
```

CLASS22{A|B|C}

Selects port 2 (S22) calibration standard class: S22A (open), S22B (short), or S22C (load). (Network analyzer only) (`[S22] : OPEN`, `SHORT`, `LOAD` under `(Cal)`; No query)

- Equivalent SCPI Command

```
:SENSe:CORRection:COLLect[:ACQuire] CS22{A|B|C}
```

CLEL

Clears the entire list. (`CLEAR LIST` under `(Sweep)`; No query)

- Equivalent SCPI Command

```
:SENSe:LIST:CLEAr
```

CLES

Clears the Status Byte register, the Standard Event Status register, the Event Status register B (Instrument Event Status register), and the Operational Status register. (No query)

- Equivalent Common Command
 - *CLS

CLOSE

Returns a file, which has been read/write-enabled using the ROPEN command or WOPEN command, to access-disabled status. If this command is executed before reading process using the READ? command completes, an error occurs.

Generally, this command is used in combination with the ROPEN command and READ? command or the WOPEN command and the WRITE command, as shown in Figure 2-2. (No query)

CNTS □ <numeric> [HZ|KHZ|MAHZ|GHZ]

Changes the step size for the center frequency function. (CENTER STEP SIZE under **Center**)

| Parameter | Range | Unit |
|-----------|-------------------------------|------|
| <numeric> | 0 to 1.8199 × 10 ⁹ | Hz |

- Query Response
 - {numeric} <new line><END>
- Equivalent SCPI Command
 - :SENSe:FREQuency:CENTer:STEP[:INCRement] □ <numeric>
- Example
 - OUTPUT 717;"CNTS 1MAHZ"

CNTSAUTO □ {OFF|ON|0|1}

Sets CENTER step policy. (STEP SIZE AUTO man under **Center**)

| Parameter | Description |
|-----------|-------------|
| OFF or 0 | Linear step |
| ON or 1 | 1-2-5 step |

- Query Response
 - {0|1} <new line><END>
- Equivalent SCPI Commands
 - :SENSe:FREQuency:CENTer:STEP[:INCRement]:AUTO □ {OFF|ON|0|1}

COLO□ <parameter>

Specifies the display element to change color. (CH1 DATA, CH1 MEM LIMIT LN, CH2 DATA, CH2 MEM LIMIT LN, GRATICULE, IBASIC, PEN 1, PEN 2, PEN 3, PEN 4, PEN 5, PEN 6, TEXT, WARNING under (Display))

| Parameter | Description |
|-----------|---|
| CH1D | Channel 1 data |
| CH1M | Channel 1 memory and limit lines |
| CH2D | Channel 2 data |
| CH2M | Channel 2 memory and limit lines |
| GRAT | Graticule and a portion of softkey text |
| WARN | Warning annotation |
| TEXT | All the non-data text |
| IBT | Text on the BASIC screen |
| PEN1 | Pen 1 |
| PEN2 | Pen 2 |
| PEN3 | Pen 3 |
| PEN4 | Pen 4 |
| PEN5 | Pen 5 |
| PEN6 | Pen 6 |

■ Query response

{CH1D|CH1M|CH2D|CH2M|WARN|TEXT|GRAT|IBT|PEN1|PEN2|PEN3|PEN4|PEN5|PEN6}
<new line><END>

■ Equivalent SCPI Command

- COLO□CH1D :DISPlay:CMAP:COLor1:HSL□<numeric (hue)>, <numeric (sat)>, <numeric (lum)>
- COLO□CH1M :DISPlay:CMAP:COLor2:HSL□<numeric (hue)>, <numeric (sat)>, <numeric (lum)>
- COLO□CH2D :DISPlay:CMAP:COLor3:HSL□<numeric (hue)>, <numeric (sat)>, <numeric (lum)>
- COLO□CH2M :DISPlay:CMAP:COLor4:HSL□<numeric (hue)>, <numeric (sat)>, <numeric (lum)>
- COLO□GRAT :DISPlay:CMAP:COLor5:HSL□<numeric (hue)>, <numeric (sat)>, <numeric (lum)>
- COLO□WARN :DISPlay:CMAP:COLor6:HSL□<numeric (hue)>, <numeric (sat)>, <numeric (lum)>
- COLO□TEXT :DISPlay:CMAP:COLor7:HSL□<numeric (hue)>, <numeric (sat)>, <numeric (lum)>
- COLO□IBT :DISPlay:CMAP:COLor8:HSL□<numeric (hue)>, <numeric (sat)>, <numeric (lum)>
- COLO□PEN1 :DISPlay:CMAP:COLor9:HSL□<numeric (hue)>, <numeric (sat)>, <numeric (lum)>
- COLO□PEN2 :DISPlay:CMAP:COLor10:HSL□<numeric (hue)>, <numeric (sat)>, <numeric (lum)>
- COLO□PEN3 :DISPlay:CMAP:COLor11:HSL□<numeric (hue)>, <numeric (sat)>, <numeric (lum)>
- COLO□PEN4 :DISPlay:CMAP:COLor12:HSL□<numeric (hue)>, <numeric (sat)>, <numeric (lum)>
- COLO□PEN5 :DISPlay:CMAP:COLor13:HSL□<numeric (hue)>, <numeric (sat)>, <numeric (lum)>
- COLO□PEN6 :DISPlay:CMAP:COLor14:HSL□<numeric (hue)>, <numeric (sat)>, <numeric (lum)>

| Parameter | Range | Unit |
|-----------------|--|------|
| <numeric (hue)> | (Hue) 0 to 100, circular, with a value of 0 resulting in the same hue as a value of 100. The approximate color is (starting at 0): red, orange, yellow, green, cyan, blue, magenta, and back to red. | % |
| <numeric (sat)> | (Saturation) 0 to 100, with 0 specifying no color (only white or gray, depending on intensity) and 1 specifying no white. | % |
| <numeric (lum)> | (Luminance) 0 to 100, with 0 resulting in black and 1 resulting in the brightest color available. | % |

□ Query Response

{numeric (hue)}, {numeric (sat)}, {numeric (lum)} <new line><END>

■ Example

OUTPUT 717;"COLO CH1D"

OUTPUT 717;":DISP:CMAP:COL1:HSL 17,100,100"

COLO \square *<parameter>*

COLOR \square *<numeric>* [PCT]

Adjusts the degree of whiteness of the color being modified. (COLOR under Display); No equivalent SCPI Command)

| Parameter | Range | Unit |
|------------------------|----------|------|
| <i><numeric></i> | 0 to 100 | % |

■ Query response

{*numeric*} \square *<new line>* \square \langle END \rangle

COMC{A|B|C}

Measures the standards for the fixture compensation. (COMPEN OPEN, SHORT, or LOAD under Cal); No query; Impedance analyzer only)

| Parameter | Description |
|-----------|-----------------|
| A | Measures OPEN. |
| B | Measures SHORT. |
| C | Measures LOAD. |

■ Equivalent SCPI Commands

COMCA :SENSe:CORRection2:COLLect[:ACQuire] \square STANdard1
COMCB :SENSe:CORRection2:COLLect[:ACQuire] \square STANdard2
COMCC :SENSe:CORRection2:COLLect[:ACQuire] \square STANdard3

COMCDAT{A|B|C} \square {OFF|ON|0|1}

Sets the OPEN, SHORT, and LOAD fixture compensation ON or OFF. (OPEN ON off, SHORT ON off, or LOAD ON off under Cal); Impedance analyzer only)

| Parameter | Description |
|-----------|------------------------------|
| A | Uses OPEN compensation data. |
| B | Uses SHORT compensation data |
| C | Uses LOAD compensation data |
| ON or 1 | Turns on the selected data. |
| OFF or 0 | Turns off the selected data. |

■ Query response

{1|0} \square *<new line>* \square \langle END \rangle

■ Equivalent SCPI Commands

COMCDATA {ON|OFF} :SENSe:CORRection2:OPEN \square {ON|OFF|1|0}
COMCDATB {ON|OFF} :SENSe:CORRection2:SHORT \square {ON|OFF|1|0}
COMCDATC {ON|OFF} :SENSe:CORRection2:LOAD \square {ON|OFF|1|0}

COMKDONE

Complete modifying the fixture compensation kit. (KIT DONE (MODIFIED) under (Cal) COMPEN KIT [USER] MODIFY [USER] ; Impedance analyzer only)

- Equivalent SCPI Command

```
:SENSE:CORRection2:CKIT:SAVE
```

COMP

Call the fixture compensation menu. You need send this command before sending COMC. (COMPEN MENU under (Cal) FIXTURE COMPEN ; No query; Impedance analyzer only)

- Equivalent SCPI Command

```
:SENSE:CORRection2:COLLect:METhodIMPedance
```

COMS

Displays the fixture compensation definition on the display. (COMPEN KIT DEFINITION under (Copy); No query; Impedance analyzer only)

- Equivalent SCPI Commands

```
DISPlay[:WINDow]:TEXT20:PAGE 1
DISPlay[:WINDow]:TEXT20:STATe ON
```

COMSDONE

Complete defining the standard for the fixture compensation kit. (STD DONE (DEFINED) under (Cal) COMPEN KIT [USER] MODIFY [USER] ; Impedance analyzer only)

- Equivalent SCPI Command

```
:SENSE:CORRection2:CKIT:STANdard:SAVE
```

CONT

Triggers sweep automatically and continuously and the trace is updated with each sweep. (CONTINUOUS under (Trig))

- Query Response

```
{0|1} <new line><END>
```

- Equivalent SCPI Command

```
:INITiate:CONTInuous{1|ON}
```

CONT

CONV□<parameter>

Selects the measurement data conversion setting (impedance, admittance, or multiple phase). (Network analyzer only) (OFF, Z:Refl, Z:Trans, Y:Refl, Y:Trans, 1/S, 4xPHASE, 8xPHASE, 16xPHASE under **Meas**)

| Parameter | Description |
|-----------|----------------------|
| OFF | Conversion OFF |
| ZREF | Z: reflection |
| ZTRA | Z: transmission |
| YREF | Y: reflection |
| YTRA | Y: transmission |
| ONEDS | Reciprocal (1/S) |
| MP4 | Multiply phase by 4 |
| MP8 | Multiply phase by 8 |
| MP16 | Multiply phase by 16 |

■ Query Response

```
{OFF|ZREF|ZTRA|YREF|YTRA|ONEDS|MP4|MP8|MP16} <new line><END>
```

■ Equivalent SCPI Command

```
CONV□OFF      :CALCulate:MATH1[:EXPRESSion]:NAME□OFF  
CONV□ZREF     :CALCulate:MATH1[:EXPRESSion]:NAME□ZREF  
CONV□ZTRA     :CALCulate:MATH1[:EXPRESSion]:NAME□ZTRA  
CONV□YREF     :CALCulate:MATH1[:EXPRESSion]:NAME□YREF  
CONV□YTRA     :CALCulate:MATH1[:EXPRESSion]:NAME□YTRA  
CONV□ONEDS    :CALCulate:MATH1[:EXPRESSion]:NAME□INVS  
CONV□MP4      :CALCulate:MATH1[:EXPRESSion]:NAME□MP4  
CONV□MP8      :CALCulate:MATH1[:EXPRESSion]:NAME□MP8  
CONV□MP16     :CALCulate:MATH1[:EXPRESSion]:NAME□MP16
```

□ Query Response

```
{OFF|ZREF|ZTRA|YREF|YTRA|ONEDS|MP4|MP8|MP16} <new line><END>
```

■ Example

```
OUTPUT 717;"CONV ZREF"  
OUTPUT 717;":CALC:MATH1:NAME ZREF"
```

COPA

Aborts a print in progress. (**COPY ABORT** under **Copy**); No query)

■ Equivalent SCPI Command

```
:HCOPY:ABORt
```

COPT□{OFF|ON|0|1}

Turns printing time and date (the time stamp function) ON or OFF. (COPY TIME ON off under **Copy**)

| Parameter | Description |
|-----------|-------------------------|
| OFF or 0 | Time stamp function OFF |
| ON or 1 | Time stamp function ON |

- Query Response

{0|1} <new line><END>

- Equivalent SCPI Command

:HCOpy:ITeM:TDSamp:STATe□{OFF|ON|0|1}

CORR□{OFF|ON|0|1}

Turns error correction ON or OFF. (Network and impedance analyzer only)
(CORRECTION ON off under **Cal**, This softkey is Network analyzer only)

| Parameter | Description |
|-----------|----------------------|
| OFF or 0 | Error correction OFF |
| ON or 1 | Error correction ON |

- Query Response

{0|1} <new line><END>

- Equivalent SCPI Command

:SENSe:CORRection[:STATe]□{OFF|ON|0|1}

- Example

```
OUTPUT 717;"CORR OFF"
OUTPUT 717;"CORR?"
ENTER 717;A
OUTPUT 717;":SENS:CORR OFF"
OUTPUT 717;":SENS:CORR?"
ENTER 717;A
```

CORR□{OFF|ON|0|1}

COUC□{OFF|ON|0|1}

Sets the channel coupling of sweep parameter values. (Between network or between impedance analyzers only) (COUPLED CH ON off under **Sweep**)

| Parameter | Description |
|-----------|----------------------|
| OFF or 0 | Channel coupling OFF |
| ON or 1 | Channel coupling ON |

■ Query Response

{0|1} <new line><END>

■ Equivalent SCPI Command

:INSTRument:COUPlE□{OFF|ON|0|1}

CRED□<*string*>

Create a new directory in a DOS format disk. (CREATE DIRECTORY under **Save**); No query)

| Parameter | Description |
|-------------------|--|
| < <i>string</i> > | Up to 8 characters for directory name (and up to 3 characters for extension) |

■ Equivalent SCPI Command

:MMEMory:CREate:DIRectory□<*string*>

■ Example

```
OUTPUT 717;"CRED ""DATA""
```

```
OUTPUT 717;" :MMEM:CRE:DIR ""DATA""
```

CWFREQ \square $\langle numeric \rangle$ [**HZ**|**KHZ**|**MAHZ**|**GHZ**]

CRSC \square {**OFF**|**ON**|**0**|**1**}

Selects the destination channel of the marker \rightarrow functions. When a marker \rightarrow function is performed, the sweep parameter or amplitude value of the destination channel is changed.

(**CROSS CHAN ON off** under (**Marker** \rightarrow))

| Parameter | Description |
|-----------|--|
| OFF or 0 | Current active channel as the destination channel |
| ON or 1 | Current inactive channel as the destination channel ¹ |

¹ Can be selected only when the dual channel function is ON.

■ Query Response

{0|1} \langle new line \rangle \langle END \rangle

■ Equivalent SCPI Command

:CALCulate:EVALuate:EFFect:ON \square {1|2}

When channel 1 is active,

| Parameter | Description |
|-----------|---|
| 1 | The channel currently active is selected. |
| 2 | The channel currently not active is selected. |

When channel 2 is active,

| Parameter | Description |
|-----------|---|
| 1 | The channel currently not active is selected. |
| 2 | The channel currently active is selected. |

CWD?

Returns the name of the current directory. (Query only)

■ Query Response

{string} \langle new line \rangle \langle END \rangle

CWFREQ \square $\langle numeric \rangle$ [**HZ**|**KHZ**|**MAHZ**|**GHZ**]

Sets the frequency for power sweep. (Network and impedance analyzer only) (**CWFREQ** under (**Source**))

| Parameter | Range | Unit |
|---------------------------|---|------|
| $\langle numeric \rangle$ | 100000 (= 100 k) to 1.82×10^9 (= 1.82 G) | Hz |

■ Query Response

{numeric} \langle new line \rangle \langle END \rangle

■ Equivalent SCPI Command

:SOURce:FREQuency[:CW] \square $\langle numeric \rangle$

CWFREQ \square *<numeric>* [HZ|KHZ|MAHZ|GHZ]

■ Example

```
OUTPUT 717;"CWFREQ 500MAHZ"
```

DATAOVAL \square *<numeric>*

Defines the imaginary part of the offset value when using the Smith, Polar, and admittance chart format. (AUX OFFSET under [Display](#))

| Parameter | Range | Unit |
|------------------------|-------------------|------|
| <i><numeric></i> | -500000 to 500000 | |

■ Query Response

```
{numeric} <new line><^END>
```

■ Equivalent SCPI Command

```
:DATA[:DATA]  $\square$ AOFF, <numeric>
```

DATGAIN \square *<numeric>*

Defines the gain value of the data math function. (GAIN under [Display](#))

| Parameter | Range | Unit |
|------------------------|---------------------------------|------|
| <i><numeric></i> | -100 to -0.001, or 0.001 to 100 | |

■ Query Response

```
{numeric} <new line><^END>
```

■ Equivalent SCPI Command

```
:DATA[:DATA]  $\square$ GAIN, <numeric>
```

DATMEM

Stores the current active measurement data in the memory of the active channel. (DATA—MEMORY under [Display](#)); No query)

■ Equivalent SCPI Command

```
:TRACe{1|2}:COPY  $\square$ MTRace, DTRace
```

■ Example

```
OUTPUT 717;":TRAC:COPY MTR,DTR"
```

DATVAL □ *<numeric>*

Defines the offset value. When using Smith, Polar, and admittance chart format, this command defines the real part of the offset value. (**OFFSET** under **Display**)

| Parameter | Range | Unit |
|------------------------|-------------------|------|
| <i><numeric></i> | -500000 to 500000 | |

■ Query Response

{*numeric*} <new line><END>

■ Equivalent SCPI Command

:DATA[:DATA] □ OFFS, *<numeric>*

DAYMYEAR

Changes the displayed date to the “day:month:year” format. (**DayMonYear** under **System**)

■ Query Response

{0|1} <new line><END>

(0 for the “month:day:year” format; 1 for the “day:month:year” format.)

■ Equivalent SCPI Command

:SYSTem:DATE:MODE □ DMY

DEFC

Returns all the color settings back to the factory-set default values. (**DEFAULT COLORS** under **Display**); No query)

■ Equivalent SCPI Command

:DISPlay:CMAP:DEFault

DEFEC

DEFEC{R1|C1|L1|C0}␣<numeric>

Defines the specified equivalent circuit parameter for simulation. (PARAMETER R1, C1, L1, C0 under `[Display]`; Impedance analyzer only)

| Parameter | Description |
|-----------|--------------------------|
| R1 | Parameter R ₁ |
| C1 | Parameter C ₁ |
| L1 | Parameter L ₁ |
| C0 | Parameter C ₀ |

| Parameter | Range | Unit |
|-----------|---|-------------------------|
| <numeric> | -1×10 ¹⁸ to 1×10 ¹⁸ | F(C0, C1) H(L1) OHM(R1) |

■ Query Response

<numeric> <new line><^END>

■ Equivalent SCPI Command

:DATA[:DATA]␣EQ{R1|C1|L1|C0},<numeric>

■ Example

```
OUTPUT @Hp4396;"DEFECR1 350HM"  
OUTPUT @Hp4396;"DEFECC1?"  
ENTER @Hp4396;C1
```

DEFGO

Returns the gain and offset values back to the default values (gain=1, offset=0). (DEFAULT GAIN & OFS under `[Display]`; No query)

■ Equivalent SCPI Command

```
:DATA[:DATA]␣GAIN,1  
:DATA[:DATA]␣OFFS,0
```

■ Example

```
OUTPUT 717;"DEFGO"  
OUTPUT 717;":DATA GAIN,1"  
OUTPUT 717;":DATA OFFS,0"
```


DEFS <1-8>

Defines the number of the calibration standards to be modified. (Network and impedance analyzer only) (DEFINE STANDARD under **Cal**); No query)

| Parameter | Description |
|-----------|---------------------------|
| 1 | Standard no. 1 (SHORT) |
| 2 | Standard no. 2 (OPEN) |
| 3 | Standard no. 3 (LOAD) |
| 4 | Standard no. 4 (DEL/THRU) |
| 5 | Standard no. 5 (LOAD) |
| 6 | Standard no. 6 (LOAD) |
| 7 | Standard no. 7 (SHORT) |
| 8 | Standard no. 8 (OPEN) |

■ Equivalent SCPI Command (Query)

:SENSe:CORRection:CKIT:SElect <STANDARD>{1-8}

□ Query response

{ STAN{1-8} } <new line><END>

■ Example

```
OUTPUT 717;"DEFS 1"
OUTPUT 717;":SENS:CORR:CKIT:SEL STAN1"
OUTPUT 717;":SENS:CORR:CKIT:SEL?"
ENTER 717;A$
```

DEFSLOAD{R|L} <numeric>

Defines the LOAD standard by entering resistance and reactance value. (LOAD: RESIST. (R), INDUCT. (L) under **Cal** CAL KIT []). Impedance analyzer only)

| Parameter | Description |
|-----------|---|
| R | Resistance value of the LOAD fixture compensation standard. |
| L | Inductance value of the LOAD fixture compensation standard. |

| Parameter | Range | Unit |
|-----------|---|-------|
| <numeric> | -1×10 ⁶ to 1×10 ⁶ | Ω (R) |
| <numeric> | -1×10 ⁶ to 1×10 ⁶ | H (L) |

■ Query Response

{ numeric } <new line><END>

■ Equivalent SCPI Command

```
DEFSLOADR :SENSe:CORRection2:CKIT:STANdard3:R<numeric>
DEFSLOADL :SENSe:CORRection2:CKIT:STANdard3:L<numeric>
```

DEFSOPEN{G|C}□<numeric>

Defines the OPEN standard by entering conductance and capacitance value.

(OPEN: CONDUCT.(G), CAP.(C) under **Cal** CAL KIT []). Impedance analyzer only)

| Parameter | Description |
|-----------|--|
| G | Conductance value of the OPEN fixture compensation standard. |
| C | Capacitance value of the OPEN fixture compensation standard. |

| Parameter | Range | Unit |
|-----------|---|-------|
| <numeric> | -1×10 ⁶ to 1×10 ⁶ | S (G) |
| <numeric> | -1×10 ⁻⁶ to 1×10 ⁶ (simple command) | f (C) |
| <numeric> | -1×10 ⁻⁹ to 1×10 ⁹ (SCPI command) | F (C) |

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

DEFSOPENG SENSE:CORREction2:CKIT:STANdard1:G□<numeric>

DEFSOPENC SENSE:CORREction2:CKIT:STANdard1:C□<numeric>

■ Example

```
OUTPUT @Hp4396;"DEFSOPENG OS"
```

```
OUTPUT @Hp4396;"DEFSOPENC 53E-6F"
```

```
OUTPUT @Hp4396;"DEFSOPENG?"
```

```
ENTER @Hp4396;G
```

```
OUTPUT @Hp4396;"SENS:CORR2:CKIT:STAN1:G OS;C 53E-6F"
```

DEFSSHOR{R|L}□<numeric>

Defines the SHORT calibration standard by entering resistance and inductance value.

(SHORT: RESIST.(R), INDUCT.(L) under **Cal** CAL KIT []). Impedance analyzer only)

| Parameter | Description |
|-----------|--|
| R | Resistance value of the SHORT fixture compensation standard. |
| L | Inductance value of the SHORT fixture compensation standard. |

| Parameter | Range | Unit |
|-----------|---|-------|
| <numeric> | -1×10 ⁶ to 1×10 ⁶ | Ω (R) |
| <numeric> | -1×10 ⁶ to 1×10 ⁶ | H (L) |

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

DEFSSHORR SENSE:CORREction2:CKIT:STANdard2:R□<numeric>

DEFSSHORL SENSE:CORREction2:CKIT:STANdard2:L□<numeric>

- Example

```
OUTPUT @Hp4396;"DEFSSHORTL 00HM"
OUTPUT @Hp4396;"DEFSSHORTL OH"

OUTPUT @Hp4396;"DEFSSHORTL?"
ENTER @Hp4396;R

OUTPUT @Hp4396;"SENS:CORR2:CKIT:STAN2:R 00HM;L OH"
```

DET {POS|NEG|SAM}

Selects the detection mode for the active channel. (Spectrum analyzer only) (POS PEAK, NEG PEAK, SAMPLE under (Meas))

| Parameter | Description |
|-----------|--------------------|
| POS | Positive Detection |
| NEG | Negative Detection |
| SAM | Sample Detection |

- Query Response

```
{POS|NEG|SAM} <new line><^END>
```

- Equivalent SCPI Command

```
:SENSe:DETEctor[:FUNction]  {POSitive|NEGative|SAMple}
```

- Example

```
OUTPUT 717;"DET POS"
```

DFLT

Returns the printing parameters to their default values. (DEFAULT SETUP under (Copy); No query)

The table below lists the default values.

| Command | Default value | Unit |
|-----------|---------------|------|
| DPI | 75 | dpi |
| FORMFEED | ON | |
| LANDSCAPE | OFF | |
| LMARG | 1.0 | inch |
| TMARG | 1.0 | inch |
| SKEY | OFF | |

- Equivalent SCPI Command

```
:HCOpy:DEFault
```

DFLT

DHOLD □ {OFF|MAX|MIN}

Selects the data hold operation. When the format is changed, the value held is initiated.
(HOLD: OFF, MAX, MIN under [Display](#))

| Parameter | Description |
|-----------|-----------------------------------|
| OFF | Data hold operation is turned off |
| MAX | Maximum data hold |
| MIN | Minimum data hold |

■ Query Response

{OFF|MAX|MIN} <new line><END>

■ Equivalent SCPI Command

```
DHOLD □ OFF      :CALCulate:AVERage:STATe □ OFF
DHOLD □ MAX      :CALCulate:AVERage:TYPE □ MAXimum
                  :CALCulate1:AVERage:STATe □ ON
DHOLD □ MIN      :CALCulate:AVERage:TYPE □ MINimum
                  :CALCulate1:AVERage:STATe □ ON
```

■ Example

```
OUTPUT 717;"DHOLD MAX"
OUTPUT 717;"DHOLD?"
ENTER 717;A$
OUTPUT 717;":CALC:AVER:TYPE MAX"
OUTPUT 717;":CALC:AVER:STAT ON"
OUTPUT 717;":CALC:AVER:TYPE?"
ENTER 717;A$
OUTPUT 717;":CALC:AVER:STAT?"
ENTER 717;A
```

DISA □ {ALLI|HIHB|ALLB|BASS}

Selects the display allocation mode. ([DISPLAY ALLOCATION](#) under [Display](#))

| Parameter | Description |
|-----------|--|
| ALLI | All instrument |
| HIHB | Half instrument and half HP instrument BASIC |
| ALLB | All HP instrument BASIC |
| BASS | HP instrument BASIC status |

- Query Response

```
{ ALLI|HIHB|ALLB|BASS } <new line><END>
```

- Equivalent SCPI Command

```
DISA|ALLI      :DISPlay[:WINDow]:ALLocation|INSTRument
DISA|HIHB      :DISPlay[:WINDow]:ALLocation|HIHB
DISA|ALLB      :DISPlay[:WINDow]:ALLocation|BASic
DISA|BASS      :DISPlay[:WINDow]:ALLocation|BStatus
```

- Example

```
OUTPUT 717;"DISA HIHB"
OUTPUT 717;"DISA?"
ENTER 717;A$
```

DISECIRC

Displays the equivalent circuit models. (**SELECT EQV CKT [A]** under **Display**); Impedance analyzer only)

| Parameter | Description |
|-----------|---|
| OFF or 0 | Turns off the equivalent circuit parameter display. |
| ON or 1 | Turns on the equivalent circuit parameter display. |

- Query Response

```
{0|1} <new line><END>
```

- Equivalent SCPI Command

```
:DISPlay[:WINDow]:TEXT19:STATe|{OFF|ON|0|1}
```

DISECPARA {OFF|ON|0|1}

Displays the equivalent circuit parameters. (**DISP EQV PARM [ON]** or **[OFF]** under **Display**); Impedance analyzer only)

| Parameter | Description |
|-----------|---|
| OFF or 0 | Turns off the equivalent circuit parameter display. |
| ON or 1 | Turns on the equivalent circuit parameter display. |

- Query Response

```
{0|1} <new line><END>
```

- Equivalent SCPI Command

```
:DISPlay[:WINDow]:TEXT18:STATe|{OFF|ON|0|1}
```

DISECPARA {OFF|ON|0|1}

DISF□{DOS|LIF}

Selects the disk format (LIF or DOS) to be used when initializing a new disk. (FORMAT [] under Save); No equivalent SCPI command)

| Parameter | Description |
|-----------|------------------------------|
| DOS | Disk Operating System format |
| LIF | Logical Interchange format |

■ Query Response

{DOS|LIF} <new line><END>

■ Example

OUTPUT 717;"DISF DOS"

DISL

Displays the list sweep table on the display. (DISPLAY LIST under Copy); No query)

■ Equivalent SCPI Command

:DISPlay[:WINDow]:TEXT{12|13}:PAGE□1
:DISPlay[:WINDow]:TEXT{12|13}:STATe□{0N|1}

(TEXT12 for the “start & stop” format; TEXT13 for the “center & span” format)

DISLLIST

Displays the limit testing table on the display. (DISPLAY LIST under Copy); No query)

■ Equivalent SCPI Command

:DISPlay[:WINDow]:TEXT{14|15}:PAGE□1
:DISPlay[:WINDow]:TEXT{14|15}:STATe□{0N|1}

(TEXT14 for the “upper & lower” format; TEXT15 for the “middle & delta” format)

DISMAMP□{UL|MD}

Selects the amplitude format to display the limit testing table to list on the screen. (DISP MODE: UPR & LWR , MID & DLT under Copy); No equivalent SCPI command)

| Parameter | Description |
|-----------|-------------------------|
| UL | Upper and lower format |
| MD | Middle and delta format |

■ Query Response

{UL|MD} <new line><END>

- Example

```
OUTPUT 717;"DISMAMP UL"
OUTPUT 717;"DISMAMP?"
ENTER 717;A$
```

DISMPRM{STSP|CTSP}

Selects the sweep parameter range format to display the list sweep table on the screen.
(DISP MODE: ST & SP, CTR & SPAN under **Copy**); No equivalent SCPI command)

| Parameter | Description |
|-----------|------------------------|
| STSP | Start and stop format |
| CTSP | Center and span format |

- Query Response

```
{STSP|CTSP} <new line><END>
```

- Example

```
OUTPUT 717;"DISMPRM STSP"
```

DISP{DATA|MEMO|DATM}

Selects the display trace type. (DISPLAY: DATA, MEMORY, DATA and MEMORY under **Display**)

| Parameter | Description |
|-----------|--------------------------------|
| DATA | Current data trace |
| MEMO | Memory trace |
| DATM | Current data and memory traces |

- Query Response

```
{DATA|MEMO|DATM} <new line><END>
```

- Equivalent SCPI Command

```
DISPDATA :DISPlay[:WINDow]:TRACe1:STATe{ON|1}
           :DISPlay[:WINDow]:TRACe2:STATe{OFF|0}

DISPMEMO :DISPlay[:WINDow]:TRACe1:STATe{OFF|0}
           :DISPlay[:WINDow]:TRACe2:STATe{ON|1}

DISPDATM :DISPlay[:WINDow]:TRACe1:STATe{ON|1}
           :DISPlay[:WINDow]:TRACe2:STATe{ON|1}
```

- Example

```
OUTPUT 717;"DISP DATA"
OUTPUT 717;":DISP:TRAC1:STAT ON"
OUTPUT 717;":DISP:TRAC2:STAT OFF"
```

DISP□{**DATA**|**MEMO**|**DATM**}

DMKR□{**ON**|**FIX**|**TRAC**|**OFF**}

Displays the Δmarker (ON, FIX, TRAC) at the point of the marker and the marker mode changes to the Δmode. Erases (OFF) the Δmarker and the Δmode is turned off. (ΔMKR, FIXED ΔMKR, TRACKING ΔMKR, ΔMODE OFF under (Marker))

| Parameter | Description |
|-----------|---|
| ON | Puts the Δmarker on a current position of the marker. |
| FIX | Sets a user-specified fixed reference marker. |
| TRAC | Puts a Δmarker at the present active marker position and turns on the tracking Δmarker. |
| OFF | Turns off the Δmode. |

■ Query Response

{ON|FIX|TRAC|OFF} <new line><^END>

■ Equivalent SCPI Command

DMKR□ON :DISPlay[:WINDow]:TRACe{[1]|2}:MARKer[1]:RELative□{ON|1}
:DISPlay[:WINDow]:TRACe{[1]|2}:MARKer[1]:RELative:REFerence□MARKer

DMKR□FIX :DISPlay[:WINDow]:TRACe{[1]|2}:MARKer[1]:RELative□{ON|1}
:DISPlay[:WINDow]:TRACe{[1]|2}:MARKer[1]:RELative:REFerence□FIXed

DMKR□TRAC :DISPlay[:WINDow]:TRACe{[1]|2}:MARKer[1]:RELative□{ON|1}
:DISPlay[:WINDow]:TRACe{[1]|2}:MARKer[1]:RELative:REFerence□TRACked

DMKR□OFF :DISPlay[:WINDow]:TRACe{[1]|2}:MARKer[1]:RELative□{OFF|0}

(TRACe[1] for the data trace; TRACe2 for the memory trace)

■ Example

```
OUTPUT 717;"DMKR ON"  
OUTPUT 717;":DISP:TRAC:MARK:REL ON"  
OUTPUT 717;":DISP:TRAC:MARK:REL:REF MARK"
```

DMKRAUV□<numeric>

Sets the auxiliary amplitude value of the fixed Δmarker. This command is used with a polar, Smith, or admittance chart. (Network and impedance analyzer only) (ΔMKR AUX VALUE under (Marker))

| Parameter | Range | Unit |
|-----------|-------------------|------|
| <numeric> | -500000 to 500000 | |

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

:CALCulate:EVALuate:REFerence:Y2□<numeric>

■ Example

```
OUTPUT 717;"DMKRAUV 0"
```


DMKRPRM \square *<numeric>* [**HZ**|**KHZ**|**MAHZ**|**GHZ**|**DBM**]

Sets the sweep parameter value of the Δ marker. (**DMKR SWP PRM** under **(Marker)**)

| Parameter | Range | Unit |
|------------------------|---------------------------|-------------------------------|
| <i><numeric></i> | Start value to stop value | Hz (frequency) dBm (power) |

- Query Response

{*numeric*} <new line><^END>

- Equivalent SCPI Command

:CALCulate:EVALuate:REFerence:X \square *<numeric>*

DMKRVAL \square *<numeric>*

Sets the amplitude value of the fixed Δ marker. (**DMKR VALUE** under **(Marker)**)

| Parameter | Range | Format |
|------------------------|---------------------------------|--------------------------------|
| <i><numeric></i> | -500 to 500 (spectrum analyzer) | (dBm, dB μ V, dBV formats) |
| <i><numeric></i> | -500 to 500 (network analyzer) | (Log magnitude format) |
| <i><numeric></i> | -500k to 500k | |

- Query Response

{*numeric*} <new line><^END>

- Equivalent SCPI Command

:CALCulate:EVALuate:REFerence:Y1 \square *<numeric>*

DONE

Completes the measurement of the selected standard calibration. (Network and impedance analyzer only) (**DONE: RESPONSE** under **(Cal)**). When Type-N calkits or user calkit, **DONE: OPEN** **DONE: SHORT** under **(Cal)**. No query)

- Equivalent SCPI Command

:SENSe:CORRection:COLLect:SAVE1

DONE

DPI □ <numeric>

Specifies the printing resolution value for the printer. (DPI under Copy)

| Parameter | Range | Unit |
|-----------|-----------|------|
| <numeric> | 75 to 600 | dpi |

■ Query Response

{numeric} <new line><END>

■ Equivalent SCPI Command

:HCOpy:DRIVer:DPI □ <numeric>

DSKEY

Disables the front panel keys and the rotary knob. To enable the keys and knob again, send the ENKEY command. (No query for the Simple command)

■ Equivalent SCPI Command

:SYSTem:KLOCK □ {1 | ON}

■ Example

OUTPUT 717;"DSKEY"

OUTPUT 717;":SYST:KLOC ON"

DUAC \square {OFF|ON|0|1}

Selects the display of both measurement channels or the active channel only.

(DUAL CHAN ON off under (Display))

| Parameter | Description |
|-----------|---------------------|
| OFF or 0 | Active channel only |
| ON or 1 | Both channels |

- Query Response

{0|1} <new line><END>

- Equivalent SCPI Command

When channel 1 is active,

```
DUAC ON      :INSTRument[:SElect] CH2
              :INSTRument:STATe ON
              :INSTRument[:SElect] CH1
              :INSTRument:STATe ON
```

When channel 2 is active,

```
DUAC ON      :INSTRument[:SElect] CH1
              :INSTRument:STATe ON
              :INSTRument[:SElect] CH2
              :INSTRument:STATe ON
```

```
DUAC OFF     :INSTRument[:SElect] {CH1|CH2}
              :INSTRument:STATe OFF
              :INSTRument[:SElect] {CH1|CH2}
```

- Example

```
OUTPUT 717;"DUAC ON"
OUTPUT 717;":INST CH1"
OUTPUT 717;":INST:STAT ON"
OUTPUT 717;":INST CH2"
OUTPUT 717;":INST:STAT ON"
```

EDITDONE

Completes editing the frequency sweep list. (LIST DONE under (Sweep); No query)

- Equivalent SCPI Command

```
:SENSe:LIST:SAVE
```

EDITLIML

Begins editing the limit line table. (EDIT LIMIT LINE under (System); No query; No equivalent SCPI command)

EDITLIML

EDITLIST

Begins editing the frequency sweep list. (**EDIT LIST** under **Sweep**); No query; No equivalent SCPI command)

ELED \square *<numeric>* [**S|MS|US|NS|PS|FS**]

Adjusts the electrical delay to balance the phase of the DUT. (Network analyzer only)
(**ELECTRICAL DELAY** under **Scale Ref**)

| Parameter | Range | Unit |
|------------------------|-----------|------|
| <i><numeric></i> | -10 to 10 | s |

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

:SENSe:CORRection:EDELAy2 \square *<numeric>*

ENKEY

Reenables the front panel keys and the rotary knob that have been disabled by the DSKEY command. (No query for the Simple command)

■ Equivalent SCPI Command

SYSTem:KLOCK \square {OFF|0}

■ Example

```
OUTPUT 717;"ENKEY"  
OUTPUT 717;":SYST:KLOC OFF"
```

EQUC \square CIR {**A|B|C|D|E**}

Selects the equivalent circuit. (**SELECT EQV CKT** \square under **Display**). Impedance analyzer only)

| Parameter | Description |
|-----------|--------------------------------|
| CIRA | For coils with high core loss. |
| CIRB | For coils and resistance. |
| CIRC | For high-value resistors. |
| CIRD | For capacitors. |
| CIRE | For resonators. |

■ Query Response

CIR{A|B|C|D|E} <new line><^END>

■ Equivalent SCPI Command

:CALCulate:EVALuate:EPARameters:CIRCuit \square {A|B|C|D|E}

■ Example

```
OUTPUT @Hp4396;"EQUC CIRA"
```

EQUC0?□<numeric>

Returns a C_0 at the specified frequency. For more information, see “EQUC0? *value*” in Appendix I. (No equivalent SCPI command; Network analyzer only)

| Parameter | Range | Unit |
|-----------|------------------------------------|------|
| <numeric> | 0 to 1.82×10^9 (= 1.82 G) | Hz |

- Query Response

<numeric><new line><^END>

EQUCPARA?

Executes six elements equivalent circuit analysis of a crystal resonator, and outputs parameters. For more information, see “EQUCPARA?” in Appendix I. (No equivalent SCPI command; Network analyzer only)

- Query Response

See “EQUCPARA?” in Appendix I.

EQUCPARS?

Executes six elements equivalent circuit analysis of a crystal resonator, and outputs parameters. For more information, see “EQUCPARS?” in Appendix I. (No equivalent SCPI command; Network analyzer only)

- Query Response

See “EQUCPARS?” in Appendix I.

EQUCPARS4?

Executes four elements equivalent circuit analysis of a crystal resonator, and outputs parameters. For more information, see “EQUCPARS4?” in Appendix I. (No equivalent SCPI command; Network analyzer only)

- Query Response

See “EQUCPARS4?” in Appendix I.

EQUM?□<numeric>

Specifies how many points are used for an approximation of a admittance circle for EQUCPARA? and EQUCPARS? equivalent circuit analysis commands for the crystal resonator. EQUCPARA? (or EQUCPARS?) thins the measured points out for the specified points, then make circle approximation. When the EQUM parameter is set greater than the number of points, EQUCPARA? (or EQUCPARS?) uses all points for the circle approximation. Default value is 8. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Network analyzer only)

| Parameter | Range | Unit |
|-----------|----------|------|
| <numeric> | 2 to 801 | N/A |

- Query Response

<numeric><new line><^END>

EQU? \square \langle numeric \rangle

ESB?

Outputs the Event Status register B (Instrument Event Status register) value. (Query Only)

■ Query Response

{numeric} \langle new line \rangle \langle END \rangle

■ Equivalent SCPI Command

:STATus:INSTRument[:EVENT]?

ESNB \square \langle numeric \rangle

Enables the bits of Event Status register B (Instrument Event Status register).

| Parameter | Range | Unit |
|-----------------------------|--|------|
| \langle numeric \rangle | Decimal expression of the contents of the register; 0 to 65535 ($=2^{16}-1$) | |

■ Query Response

{numeric} \langle new line \rangle \langle END \rangle

■ Equivalent SCPI Command

:STATus:INSTRument:ENABle \square \langle numeric \rangle

EXPP \square {OFF|ON|0|1}

Turns on and off the expanded phase display. (EXP PHASE on OFF under **Format**); Impedance analyzer only)

| Parameter | Description |
|-----------|---------------------------------------|
| OFF or 0 | Turns off the expanded phase display. |
| ON or 1 | Turns on the expanded phase display. |

■ Query Response

{0|1} \langle new line \rangle \langle END \rangle

■ Equivalent SCPI Command

EXPP \square OFF :CALCulate:MATH1:STATe \square OFF
:CALCulate:FORMat \square PHASe

EXPP \square ON :CALCulate:MATH1:STATe \square OFF
:CALCulate:FORMat \square UPHASe

FIXT□{NONE|16191|16192|16193|16194|USED}

FILC□<string1>,<string2>,<string3>,<string4>

Copies files. (COPY FILE under (Save); No query)

| Parameter | Description |
|-----------|--|
| <string1> | Source file name |
| <string2> | Source device name ("DISK" or "MEMORY") ¹ |
| <string3> | Destination file name |
| <string4> | Destination device name ("DISK" or "MEMORY") |

¹ "DISK" for the built-in flexible disk drive; "MEMORY" for the RAM disk memory.

■ Equivalent SCPI Command

:MMEMory:COPIE□<string1>,<string2>,<string3>,<string4>

■ Example

```
OUTPUT @Hp4396;"FILC ""DAT1.TXT"",""MEMORY"",""DAT1.TXT"",""DISK"""
```

```
OUTPUT @Hp4396;":MMEM:COPIE ""DAT1.TXT"",""MEMORY"",""DAT1.TXT"",""DISK"""
```

FIXE□<numeric>

Sets the electrical length of the fixture. (DEFINE EXTENSION under (Meas); Impedance analyzer only.)

| Parameter | Range | Unit |
|-----------|-----------|------|
| <numeric> | -10 to 10 | m |

■ Query Response

{numeric} <new line><END>

■ Equivalent SCPI Command

:SYSTem:FIXTure:DIStance□<numeric>

FIXKDONE

Terminates the user fixture setting. (DONE under (Meas); No query; Impedance analyzer only.)

■ Equivalent SCPI Command

:SYSTem:FIXTure:SAVE

FIXT□{NONE|16191|16192|16193|16194|USED}

Specifies the fixture in use in order to select which electrical length (recorded in the analyzer) is to be used. (FIXTURE:NONE, 16191, 16192, 16193, 16194, USED under (Meas) SELECT FIXTURE; Impedance analyzer only.)

■ Query Response

{NONE|16191|16192|16193|16194|USED} <new line><END>

■ Equivalent SCPI Command

FIXT {NONE|16191|16192|16193|16194|USED}

:SYSTem:FIXTure {NONE|16191|16192|16193|16194|USED}

FMT <parameter>

Selects the display format. (FORMAT: LOG MAG, PHASE, DELAY, SMITH [Re Im], POLAR [Re Im], LIN MAG, SWR, FORMAT: REAL, IMAGINARY, EXPANDED PHASE, ADMITTANCE [Re Im], FORMAT: SPECTRUM, NOISE, LIN Y-AXIS, LOG Y-AXIS, COPLEX PLANE under (Format))

| Parameter | Description |
|-----------|--|
| LOGM | Log magnitude format (Network analyzer only) |
| PHAS | Phase format (Network analyzer only) |
| DELA | Delay format (Network analyzer only) |
| SMITH | Smith chart format (Network and impedance analyzer only) |
| POLA | Polar chart format (Network and impedance analyzer only) |
| LINM | Linear magnitude format (Network analyzer only) |
| SWR | SWR format (Network analyzer only) |
| REAL | Real format (Network analyzer only) |
| IMAG | Imaginary format (Network analyzer only) |
| EXPP | Expanded phase format (Network analyzer only) |
| ADMIT | Admittance Smith chart (Network and impedance analyzer only) |
| SPECT | Spectrum measurement (Spectrum analyzer only) |
| NOISE | Noise level measurement (Spectrum analyzer only) |
| LINY | Linear Y-axis measurement (Impedance analyzer only) |
| LOGY | Log Y-axis measurement (Impedance analyzer only) |
| COMP | Complex plane measurement (Impedance analyzer only) |

■ Query Response

{LOGM|PHAS|DELA|SMITH|POLA|LINM|SWR|REAL|IMAG|EXPP|ADMIT|SPECT|NOISE|LINY|LOGY|COMP} <new line><END>

■ Equivalent SCPI Command

FMT LOGM :DISPlay[:WINDow]:TRACe{[1]|2}:GRATicule:FORMat RECTangle
:CALCulate:FORMat MLOGarithmic

FMT PHAS :DISPlay[:WINDow]:TRACe{[1]|2}:GRATicule:FORMat RECTangle
:CALCulate:FORMat PHASe

FMT DELA :DISPlay[:WINDow]:TRACe{[1]|2}:GRATicule:FORMat RECTangle
:CALCulate:FORMat GDElay

FMT SMITH :DISPlay[:WINDow]:TRACe{[1]|2}:GRATicule:FORMat SMITH
:CALCulate:FORMat COMPLEX

FMT POLA :DISPlay[:WINDow]:TRACe{[1]|2}:GRATicule:FORMat POLar
:CALCulate:FORMat COMPLEX

FMT LINM :DISPlay[:WINDow]:TRACe{[1]|2}:GRATicule:FORMat RECTangle
:CALCulate:FORMat MLINear

FMT SWR :DISPlay[:WINDow]:TRACe{[1]|2}:GRATicule:FORMat RECTangle
:CALCulate:FORMat SWR

FNAME? <numeric>

```

FMT␣REAL      :DISPlay[:WINDow]:TRACe{[1]|2}:GRATicule:FORMat␣RECTangle
                :CALCulate:FORMat␣REAL
FMT␣IMAG      :DISPlay[:WINDow]:TRACe{[1]|2}:GRATicule:FORMat␣RECTangle
                :CALCulate:FORMat␣IMAGinary
FMT␣EXPP      :DISPlay[:WINDow]:TRACe{[1]|2}:GRATicule:FORMat␣RECTangle
                :CALCulate:FORMat␣UPHase
FMT␣ADMIT     :DISPlay[:WINDow]:TRACe{[1]|2}:GRATicule:FORMat␣ADMittance
                :CALCulate:FORMat␣COMPLex
FMT␣SPECT     :SENSe:FUNCTion␣"POWer␣{1|2|3|4}"
FMT␣NOISE     :SENSe:FUNCTion␣"POWer{1|2|3|4}:PSDensity"

FMT␣LINY      DISPlay[:WINDow]:TRACe{[1]|2}:GRATicule:FORMat RECTangle
                DISPlay[:WINDow]:TRACe{[1]|2}:Y::SPACing LINear
FMT␣LOGY      DISPlay[:WINDow]:TRACe{[1]|2}:GRATicule:FORMat RECTangle
                DISPlay[:WINDow]:TRACe{[1]|2}:Y::SPACing LOGarithmic
FMT␣COMP      DISPlay[:WINDow]:TRACe{[1]|2}:GRATicule:FORMat CPLane
                CALCulate:FORMat:COMPLex

                (TRACe[1] for the data trace; TRACe2 for the memory trace.)

```

■ Example

```

OUTPUT @Hp4396;"FMT LOGM"
OUTPUT @Hp4396;"FMT?"
ENTER @Hp4396;A$

OUTPUT @Hp4396;":DISP:TRAC:GRAT:FORM RECT"
OUTPUT @Hp4396;":CALC:FORM MLOG"

OUTPUT @Hp4396;":DISP:TRAC:GRAT:FORM?"
ENTER @Hp4396;A$
OUTPUT @Hp4396;":CALC:FORM?"
ENTER @Hp4396;B$

```

FNAME? <numeric>

Returns the file name corresponding to a specified number in the current directory. To each file, a number is assigned from 1 to “the number of the files” in alphabetical order. Use the FNUM? command to verify the number of the files in the current directory. (Query only)

| Parameter | Description | Range |
|-----------|--------------------|---|
| <numeric> | Specified file No. | 1 to “the number of the files in the current directory” |

■ Query Response

```
{string} <new line><^END>
```

FNAME? □ <numeric>

FNUM?

Returns the number of the files in the current directory. (Query only)

- Query Response

{*numeric*} <new line><END>

FORM2

Sets the IEEE 32-bit floating point format to transfer trace data via GPIB. See Appendix H for more information. (No query for the Simple command)

- Equivalent SCPI Command

:FORMat[:DATA] □REAL,32

FORM3

Sets the IEEE 64-bit floating point format to transfer the trace data via GPIB. See Appendix H for more information. (No query for the Simple command)

- Equivalent SCPI Command

:FORMat[:DATA] □REAL,64

FORM4

Sets the ASCII transfer format to transfer the trace data via GPIB. See Appendix H for more information. (No query for the Simple command)

- Equivalent SCPI Command

:FORMat[:DATA] □ASCIi

FORM5

Sets MS-DOS format to transfer the trace data via GPIB. See Appendix H for more information. (No query for the Simple command)

- Equivalent SCPI Command

:FORMat[:DATA] □PACKed,32

FORMFEED □ {OFF|ON|0|1}

Sets the printer ON or OFF for delivering printed paper each time printing an entire screen is finished. When the paper orientation is set to Landscape, the setting by this FORMFEED command will not take effect and the printer delivers printed paper screen by screen.

| Parameter | Description |
|-----------|--------------------------------|
| OFF or 0 | Does not deliver printed paper |
| ON or 1 | Delivers printed paper |

- Query Response

{0|1} <new line><END>

- Equivalent SCPI Command

:HCOPIy:DRIVER:FORMFeed □ {OFF|ON|0|1}

FREQ

Blanks the displayed frequency notation for security purposes. Frequency notation cannot be restored except by sending the :SYSTEM:PRESet or *RST command, or by turning the power OFF and ON. (FREQUENCY BLANK under [Display](#))

- Query Response

{0|1} <new line><END>

- Equivalent SCPI Command

:DISPlay:ANNotation:FREQuency □ {OFF|0}

:SYSTEM:SECurity[:STATE] □ {ON|1}

FSIZE? □ <string>

Returns the size of a specified file in bytes. If the file does not exist, this command returns -1. (Query only)

| Parameter | Description |
|-----------|--|
| <string> | File name of up to 12 characters including its extension (for the LIF format, up to 10 characters) |

- Query Response

{numeric} <new line><END>

FULL

Sets the SPAN to the maximum range. (FULL SPAN under [Span](#)); No query)

- Equivalent SCPI Command

:SENSe:FREQuency:SPAN:FULL (frequency) or

:SOURce:POWer:SPAN:FULL (power)

FULS

FWDI

Measures S_{21} isolation. (Network analyzer only) (FWD ISOL'N ISOL'N STD under **Cal**); No query)

■ Equivalent SCPI Command

:SENSe:CORRection:COLLect[:ACQuire]␣FWDI

FWDM

Measures S_{11} load match. (Network analyzer only) (FWD. MATCH THRU under **Cal**); No query)

■ Equivalent SCPI Command

:SENSe:CORRection:COLLect[:ACQuire]␣FWDM

FWDT

Measures S_{21} frequency response. (Network analyzer only) (FWD. TRANS. THRU under **Cal**); No query)

■ Equivalent SCPI Command

:SENSe:CORRection:COLLect[:ACQuire]␣FWDT

GATCTL␣{LEV|EDG}

Specifies the gate trigger mode. (Spectrum analyzer only) (Option 1D6 only)
(GATE CTL: LEVEL, EDGE under **Trigger**)

| Parameter | Description |
|-----------|-------------------------|
| LEV | Level gate trigger mode |
| EDG | Edge gate trigger mode |

■ Query Response

{LEV|EDG} <new line><END>

■ Equivalent SCPI Command

GATCTL␣LEV :SENSe:SWEep:GATed:TRIGger␣LEVel

GATCTL␣EDG :SENSe:SWEep:GATed:TRIGger␣EDGE

GRODAPER \square *<numeric>* [PCT]**GATDLY** \square *<numeric>* [US|MS|S]

Sets the gate delay. (Spectrum analyzer only) (Option 1D6 only) (GATE DELAY under (Trigger))

| Parameter | Range | Unit |
|------------------------|-----------------------------|------|
| <i><numeric></i> | 0.000002 (=2 μ) to 3.2 | s |

- Query Response

{*numeric*} <new line><END>

- Equivalent SCPI Command

:SENSe:SWEep:GATed:DELay \square *<numeric>*

- Example

```
OUTPUT @Hp4396;"GATDLY 10US"
```

GATLEN \square *<numeric>* [US|MS|S]

Sets the gate length. (Spectrum analyzer only) (Option 1D6 only) (GATE LENGTH under (Trigger))

| Parameter | Range | Unit |
|------------------------|-----------------------------|------|
| <i><numeric></i> | 0.000002 (=2 μ) to 3.2 | s |

- Query Response

{*numeric*} <new line><END>

- Equivalent SCPI Command

:SENSe:SWEep:GATed:LENGth \square *<numeric>*

- Example

```
OUTPUT @Hp4396;"GATLEN 100US"
```

```
OUTPUT @Hp4396;":SENS:SWE:GAT:LENG 100US"
```

GRODAPER \square *<numeric>* [PCT]

Sets the aperture for the group delay measurement as a percentage of the span. (Network analyzer only) (GROUP DELY APERTURE under (Bw/Avg))

| Parameter | Range | Unit |
|------------------------|---------------------------------------|------|
| <i><numeric></i> | 0.25 to 20 (of span) (simple command) | % |
| <i><numeric></i> | 0.0025 to 0.2 (SCPI command) | |

- Query Response

{*numeric*} <new line><END>

- Equivalent SCPI Command

:CALCulate:GDAPerture:APERture \square *<numeric>*

GRODAPER \square *<numeric>* [PCT]

HOLD

Freezes the data trace on the display. the analyzer stops sweeping and taking data.

(SWEEP: HOLD under (Trigger))

■ Query Response

{0|1} <new line><^END>

| Parameter | Description |
|-----------|--------------------------|
| 0 | Sweeping (not hold mode) |
| 1 | Hold mode |

■ Equivalent SCPI Command

:INITiate:CONTinuous \square {OFF|0}
:ABORt

INID

Initializes the disk in the flexible disk drive or the RAM disk memory. (INITIALIZE under (Save); No query) Floppy disks can be initialized in the 2HD format only.

■ Equivalent SCPI Command

:MMEMory:INITialize \square *<string (msus)>*, {LIF|DOS}

| Parameter | Description |
|------------------------------|--|
| <i><string (msus)></i> | "DISK" for the internal flexible disk drive "MEMORY" for the internal RAM disk memory |

■ Example

```
OUTPUT @Hp4396;":MMEM:INIT ""DISK"",DOS"
```

INP8IO?

Inputs data from the 4-bit parallel input to the analyzer, and outputs the data to a controller. (Query only)

■ Query Response

{*numeric*} <new line><^END>

■ Equivalent SCPI Command

:SYSTem:COMMunicate:PARAllel[:RECEive]:DATA?

INPUCALC{1-12}□<numeric (1)>,<numeric (2)>, ... ,<numeric (n)>

Stores the measurement calibration error coefficient set of real/imaginary pairs input via GPIB into the analyzer's memory. The command definition changes to INPUCALC{1-3} when used in the impedance analyzer. See Appendix F for calibration array assignments. (Network and impedance analyzer only; No query)

| Parameter | Description |
|-----------|---|
| <numeric> | Complex number (Data format: real, imaginary) |

■ Equivalent SCPI Command

:DATA[:DATA]□CC0{1-12},{<numeric (1)>,<numeric (2)>, ... ,<numeric(n)>|<block>}

■ Example

```
DIM A(1:201,1:2)          NOP: 201
OUTPUT @Hp4396;"INPUCALC1 ";A(*)  Set the measurement calibration error coefficient.

DIM A(1:201,1:2)          NOP: 201
OUTPUT @Hp4396;":DATA CC01,";A(*)  Set the measurement calibration error coefficient.
```

INPUCALK□<block>

Stores the calibration kit data transmitted by the OUTPCALK? command. (Network and impedance analyzer only) (No query)

| Parameter | Description |
|-----------|--|
| <block> | Block data (Data format: 4396B internal format (714 bytes of binary data)) |

■ Equivalent SCPI Command

:DATA[:DATA]□CKIT,{<block>|<numeric (1)>,<numeric (2)>, ... ,<numeric (n)>}

■ Example

```
OUTPUT @Hp4396;"INPUCALK ";A$
OUTPUT @Hp4396;":DATA CKIT,";A$
```

INPUCOMC{1|2|3}□<numeric>

Inputs data into the fixture compensation coefficient arrays. (No query; Impedance analyzer only.)

The analyzer handles a reflection coefficient data for the intermediate processing. Thus, the fixture compensation is performed for the reflection coefficient as follows:

$$\Gamma = \frac{\Gamma_M - A}{B \times (\Gamma_M - A) + C}$$

Where,

- A, B, and C Fixture compensation coefficients. (complex)
- Γ_M Measured reflection data. (converted from V and I.)
- Γ Corrected reflection data.

INPUCOMC{1|2|3} □ <numeric>

By using this command, you can change the contents of the fixture compensation coefficient arrays.

| Parameter | Description |
|-----------|---|
| 1 | coefficient A |
| 2 | coefficient B |
| 3 | coefficient C |
| <numeric> | Complex number (Data format: real, imaginary) |

■ Equivalent SCPI Command

```
:DATA[:DATA] □ CMP{1|2|3}, {<numeric>
```

■ Example

```
OUTPUT @Hp4396;"INPUCOMC1 ";Dat(*)
```

```
OUTPUT @Hp4396;"DATA CMP1,";Dat(*)
```

INPUD

Executes a 3-term calibration by using real data which are set with INPULOAA, INPUOPEA, and INPUSHOA commands. (No equivalent SCPI command; No query; Network analyzer only)

INPUDATA □ <numeric (1)>, <numeric (2)>, ... , <numeric (n)>

Inputs the error corrected data. (No query)

| Parameter | Description |
|-----------|---|
| <numeric> | Complex number (Data format: real, imaginary) for the Network analyzer Real number for the Spectrum analyzer |

■ Equivalent SCPI Command

```
DATA[:DATA] □ DATA, {<numeric (1)>, <numeric (2)>, ... , <numeric (n)> | <block>}
```

■ Example

```
DIM A(1:201,1:2) Network Analyzer; NOP: 201
```

```
! Set the error corrected data.
```

```
OUTPUT @Hp4396;"INPUDATA ";A(*)
```

```
DIM A(1:201,1:2)
```

```
! Set the error corrected data.
```

```
OUTPUT @Hp4396;" :DATA DATA,";A(*)
```


INPUOPEA \square \langle numeric (1) \rangle, \langle numeric (2) \rangle, \dots, \langle numeric (n) \rangle

INPUDTRC \square \langle numeric (1) \rangle, \langle numeric (2) \rangle, \dots, \langle numeric (n) \rangle

Inputs data to DATA TRACE memory. (No query)

| Parameter | Description |
|-----------------------------|---|
| \langle numeric \rangle | Complex number (Data format: real, imaginary) for the Network analyzer Real number for the Spectrum analyzer |

■ Equivalent SCPI Command

:TRACe[:DATA] \square DTR, { \langle numeric (1) \rangle, \langle numeric (2) \rangle, \dots, \langle numeric (n) \rangle | \langle block \rangle }

■ Example

```
DIM A(1:201,1:2)           Network analyzer, NOP: 201
! Set the trace data.
OUTPUT @Hp4396;"INPUDTRC ";A(*)

DIM A(1:201,1:2)
! Set the trace data.
OUTPUT @Hp4396;" :TRAC DTR, ";A(*)
```

INPULOAA \square \langle numeric (1) \rangle, \langle numeric (2) \rangle, \dots, \langle numeric (n) \rangle

Inputs the real LOAD data array for a 3-term calubtaion. (No equivalent SCPI command; No query; Network analyzer only)

| Parameter | Description |
|-----------------------------|---|
| \langle numeric \rangle | Complex number (Data format: real, imaginary) |

INPUOPEA \square \langle numeric (1) \rangle, \langle numeric (2) \rangle, \dots, \langle numeric (n) \rangle

Inputs the real OPEN data array for a 3-term calubtaion. (No equivalent SCPI command; No query; Network analyzer only)

| Parameter | Description |
|-----------------------------|---|
| \langle numeric \rangle | Complex number (Data format: real, imaginary) |

INPUOPEA \square \langle numeric (1) \rangle , \langle numeric (2) \rangle , ... , \langle numeric (n) \rangle

INPURAW {1-4} \square \langle numeric (1) \rangle , \langle numeric (2) \rangle , ... , \langle numeric (n) \rangle

Inputs raw data. The command definition changes to INPURAW{1} when used in the impedance analyzer and spectrum analyzer. (No query)

| Parameter | Description |
|-----------------------------|---|
| \langle numeric \rangle | Complex number (Data format: real, imaginary) for the Network analyzer Real number for the Spectrum analyzer |

■ Equivalent SCPI Command

:DATA[:DATA] \square RAW{1-4}, { \langle numeric (1) \rangle , \langle numeric (2) \rangle , ... , \langle numeric (n) \rangle } | \langle block \rangle }

■ Example

```
DIM A(1:201,1:2)           Network Analyzer; NOP: 201
! Set the raw data.
OUTPUT @Hp4396;"INPURAW1 ";A(*)

DIM A(1:201,1:2)
! Set the raw data.
OUTPUT @Hp4396;" :DATA RAW1,";A(*)
```

INPUSHOA \square \langle numeric (1) \rangle , \langle numeric (2) \rangle , ... , \langle numeric (n) \rangle

Inputs the real SHORT data array for a 3-term calubtaion. (No equivalent SCPI command; No query; Network analyzer only)

| Parameter | Description |
|-----------------------------|---|
| \langle numeric \rangle | Complex number (Data format: real, imaginary) |

INPZ \square {50|75} [OHM]

Sets the input impedance. (Spectrum analyzer only) (**INPUT Z** under **Cal**)

■ Query Response

{numeric} \langle new line \rangle \langle END \rangle

■ Equivalent SCPI Command

:INPut:IMPedance \square \langle numeric \rangle

■ Example

```
OUTPUT @Hp4396;"INPZ 50"
```

INTE□<numeric>**[PCT]**

Sets the display intensity as a percent of the brightest setting. (INTENSITY under (Display))

| Parameter | Range | Unit |
|-----------|---------------------------|------|
| <value> | 0 to 100 (simple command) | % |
| <value> | 0 to 1 (SCPI command) | |

- Query Response

{numeric} <new line><END>

- Equivalent SCPI Command

:DISPlay:BRIGhtness□<numeric>

ISOD

Completes isolation calibration. The error coefficients are calculated and stored. (Network analyzer only) (ISOLATION DONE under (Cal); No query)

- Equivalent SCPI Command

:SENSe:CORRection:COLLect:SAVE7

ISOL

Starts the isolation calibration. (Network analyzer only) (ISOLATION under (Cal); No query)

- Equivalent SCPI Command

:SENSe:CORRection:COLLect[:ACQuire]□ISOL2

KEY□<numeric>

Sends the key code for a key or a softkey on the front panel. This is equivalent to actually pressing a key. See Appendix G for key codes.

| Parameter | Description |
|-----------|-------------|
| <numeric> | 0 to 52 |

- Query Response

{numeric} <new line><END>

- Equivalent SCPI Command

:SYSTem:KEY□<numeric>

- Example

OUTPUT @Hp4396;"KEY 8" *Equivalent to pressing (Chan 1).*

KEY \square *<numeric>*

KITD

Completes the procedure to define user cal kit. (Network analyzer only)
(KIT DONE (MODIFIED) under (Cal); No query)

■ Equivalent SCPI Command

:SENSe:CORRection:CKIT:SAVE \square ALL

LABECOMK

 \square *<string>*

Modifies the label of user defined fixture compensation kit. (Impedance analyzer only)
(LABEL KIT under (Cal); No query)

| Parameter | Description |
|-----------------------|---------------------|
| <i><string></i> | Up to 8 characters. |

■ Equivalent SCPI Command (Query)

:SENSe:CORRection2:CKIT:LABel \square *<string>*

■ Query Response

{*numeric*} \langle new line \rangle \langle END \rangle

■ Example

OUTPUT @Hp4396;"LABECOMK ""NEW""

LABEFIX

 \square *<string>*

Modifies the label of user defined test fixture. (LABEL FIXTURE under (Meas) FIXTURE \square);
Impedance analyzer only.)

| Parameter | Description |
|-----------------------|-----------------------|
| <i><string></i> | Up to ten characters. |

■ Query Response

<string> \langle new line \rangle \langle END \rangle

■ Equivalent SCPI Command

:SYSTem:FIXTure:LABel \square *<string>*

■ Example

OUTPUT @Hp4396;"LABEFIX ""NEW""

LABEFWD{T|M}□<string>

Defines the label for the forward transmission (THRU) or the forward match (THRU) calibration. (Network analyzer only) (LABEL: FWD. TRANS., FWD. MATCH under Cal); No query)

| Parameter | Description |
|-----------|-----------------------|
| <string> | Up to ten characters. |

- Equivalent SCPI Command (Query)

LABEFWDT□<string> :SENSe:CORRection:CKIT:CLASs7:LABel□<string>

LABEFWDM□<string> :SENSe:CORRection:CKIT:CLASs9:LABel□<string>

- Query Response

{numeric} <new line><^END>

LABEIMP{A|B|C}□<string>

Defines the label for the first class, second class, or the third class required for an impedance measurement calibration. (Impedance analyzer only) (LABEL CLASS under Cal); No query)

| Parameter | Description |
|-----------|-----------------------|
| <string> | Up to ten characters. |

- Equivalent SCPI Command (Query)

LABEIMPA□<string> :SENSe:CORRection:CKIT:CLASs13:LABel□<string>

LABEIMPB□<string> :SENSe:CORRection:CKIT:CLASs14:LABel□<string>

LABEIMPC□<string> :SENSe:CORRection:CKIT:CLASs15:LABel□<string>

- Query Response

{numeric} <new line><^END>

LABERES{P|I}□<string>

Defines the label for the response, or the response and isolation calibration. (Network analyzer only) (RESPONSE, RESPONSE & ISOL'N under Cal); No query)

| Parameter | Description |
|-----------|-----------------------|
| <string> | Up to ten characters. |

- Equivalent SCPI Command (Query)

LABERESP□<string> :SENSe:CORRection:CKIT:CLASs11:LABel□<string>

LABERESI□<string> :SENSe:CORRection:CKIT:CLASs12:LABel□<string>

- Query Response

{numeric} <new line><^END>

LABERES{P|I}□<string>

■ Example

```
OUTPUT @Hp4396;"LABERESP ""RESPONSE""
```

```
OUTPUT @Hp4396;":SENS:CORR:CKIT:CLAS11:LAB ""RESPONSE""
```

LABEREV{T|M}□<string>

Defines the label for reverse transmission (THRU) or the reverse match (THRU) calibration. (Network analyzer only) (REV.TRANS. , REV.MATCH under **Cal**); No query

| Parameter | Description |
|-----------|-----------------------|
| <string> | Up to ten characters. |

■ Equivalent SCPI Command (Query)

```
LABEREVT□<string> :SENSe:CORRection:CKIT:CLASs8:LABel□<string>
```

```
LABEREVM□<string> :SENSe:CORRection:CKIT:CLASs10:LABel□<string>
```

■ Query Response

```
{numeric} <new line><^END>
```

LABES11{A|B|C}□<string>

Defines the label for the first class, the second class, or the third class required for an S₁₁ 1-port calibration. (Network analyzer only) (LABEL: S11A , S11B , S11C under **Cal**); No query

| Parameter | Description |
|-----------|-----------------------|
| <string> | Up to ten characters. |

■ Equivalent SCPI Command (Query)

```
LABES11A□<string> :SENSe:CORRection:CKIT:CLASs1:LABel□<string>
```

```
LABES11B□<string> :SENSe:CORRection:CKIT:CLASs2:LABel□<string>
```

```
LABES11C□<string> :SENSe:CORRection:CKIT:CLASs3:LABel□<string>
```

■ Query Response

```
{numeric} <new line><^END>
```

LABES22{A|B|C}□<string>

Defines the label for the first class, the second class, or the third class required for an S₂₂ 1-port calibration. (Network analyzer only) (LABEL: S22A, S22B, S22C under Cal); No query)

| Parameter | Description |
|-----------|-----------------------|
| <string> | Up to ten characters. |

- Equivalent SCPI Command (Query)

LABES22A□<string> :SENSe:CORRection:CKIT:CLASs4:LABel□<string>

LABES22B□<string> :SENSe:CORRection:CKIT:CLASs5:LABel□<string>

LABES22C□<string> :SENSe:CORRection:CKIT:CLASs6:LABel□<string>

- Query Response

{numeric} <new line><^END>

- Example

```
OUTPUT @Hp4396;"LABES22A ""OPENS""
```

```
OUTPUT @Hp4396;":SENS:CORR:CKIT:CLAS4:LAB ""OPENS""
```

LABK□<string>

Defines a label for a new calibration kit. (Network and impedance analyzer only) (LABEL KIT under Cal); No query)

| Parameter | Description |
|-----------|-------------------------|
| <string> | Up to eight characters. |

- Equivalent SCPI Command (Query)

:SENSe:CORRection:CKIT:LABel□<string>

- Query Response

{numeric} <new line><^END>

- Example

```
OUTPUT @Hp4396;"LABK ""7mm""
```

```
OUTPUT @Hp4396;":SENS:CORR:CKIT:LAB ""7mm""
```

LABK□<string>

LABS□<string>

Defines a label for the standard. (Network and impedance analyzer only) (**LABEL STD** under **Cal**); No query)

| Parameter | Description |
|-----------|-----------------------|
| <string> | Up to ten characters. |

■ Equivalent SCPI Command

:SENSe:CORRection:CKIT:STANdard:LABel□<string>

■ Example

```
OUTPUT @Hp4396;"LABS ""SHORT""
```

```
OUTPUT @Hp4396;" :SENS:CORR:CKIT:STAN:LAB ""SHORT""
```

LANDSCAPE□{OFF|ON|0|1}

Sets the orientation of paper landscape or not, using ON or OFF. This setting takes effect for printers which support printing paper placed in the landscape orientation. Setting the paper orientation mode will invalidate the setting by the FORMFEED command. (**LANDSCAPE** under **Copy**)

| Parameter | Description |
|-----------|--|
| OFF or 0 | The orientation of paper is not set to Landscape. (Thus, Portrait) |
| ON or 1 | The orientation of paper is set to Landscape. |

■ Query Response

{0|1} <new line><END>

■ Equivalent SCPI Command

:HCOPY:DRIVER:LANDScape□{OFF|ON|0|1}

LIMCLEL

Clears all segments in the limit line. (**CLEAR LIST YES** under **System**); No query)

■ Equivalent SCPI Command

:CALCulate:LIMit:CLEar

LIMD□<numeric>

Sets the limits an equal amount above and below a specified middle value, instead of setting upper and lower limits separately. (**DELTA LIMITS** under **System**)

| Parameter | Range | Unit |
|-----------|--------------|------|
| <numeric> | 0 to 5000000 | |

■ Query Response

{*numeric*} <new line><^END>

■ Equivalent SCPI Command

:CALCulate:LIMit:SEGment:DELTA□<*numeric*>

LIMEDONE

Completes editing the limit table. (DONE under (System); No query)

■ Equivalent SCPI Command

:CALCulate:LIMit:SAVE

LIMIAMPO□<*numeric*>

Adds or subtracts an offset in amplitude value. (AMPLITUDE OFFSET under (System))

| Parameter | Range | Unit |
|--------------------|---------------------|------|
| < <i>numeric</i> > | -5000000 to 5000000 | |

■ Query Response

{*numeric*} <new line><^END>

■ Equivalent SCPI Command

:CALCulate:LIMit:OFFSet□<*numeric*>

LIMILINE□{OFF|ON|0|1}

Sets limit lines ON or OFF. (LIMIT LINE ON off under (System))

| Parameter | Description |
|-----------|-----------------|
| OFF or 0 | Limit lines OFF |
| ON or 1 | Limit lines ON |

■ Query Response

{0|1} <new line><^END>

■ Equivalent SCPI Command

:CALCulate:LIMit:LINE□{OFF|ON|0|1}

■ Example

OUTPUT @Hp4396;"LIMILINE ON"

LIMLINE \square {OFF|ON|0|1}**LIMIPRMO** \square \langle numeric \rangle

Adds or subtracts an offset from the sweep parameter value. (SWP PARAM OFFSET under **System**)

| Parameter | Range | Unit |
|-----------------------------|---|------------------------------|
| \langle numeric \rangle | -1.82×10^9 (–1.82 G) to 1.82×10^9 (1.82 G) –60 to 60 | Hz (frequency) dB (power) |

■ Query Response

{numeric} \langle new line \rangle \langle ^END \rangle

■ Equivalent SCPI Command

:CALCulate:LIMit:CONTRol:OFFSet \square \langle numeric \rangle

LIMITEST \square {OFF|ON|0|1}

Sets the limit testing ON or OFF. (LIMIT TEST ON off under **System**)

| Parameter | Description |
|-----------|-------------------|
| OFF or 0 | Limit testing OFF |
| ON or 1 | Limit testing ON |

■ Query Response

{numeric} \langle new line \rangle \langle ^END \rangle

■ Equivalent SCPI Command

:CALCulate:LIMit:STATe \square {OFF|ON|0|1}

LIML \square \langle numeric \rangle

Sets the lower limit value for the segment. (LOWER LIMIT under **System**)

| Parameter | Range | Unit |
|---------------------------|---------------------|------|
| \langle value \rangle | –5000000 to 5000000 | |

■ Query Response

{numeric} \langle new line \rangle \langle ^END \rangle

■ Equivalent SCPI Command

:CALCulate:LIMit:SEGment:LOWer \square \langle numeric \rangle

LIMM \square *<numeric>*

Sets the midpoint for delta limits. (**MIDDLE VALUE** under **(System)**)

| Parameter | Range | Unit |
|----------------------|---------------------|------|
| <i><value></i> | -5000000 to 5000000 | |

■ Query Response

{*numeric*} \langle new line \rangle \langle END \rangle

■ Equivalent SCPI Command

:CALCulate:LIMit:SEGment:MIDDLE \square *<numeric>*

LIMPRM \square *<numeric>* [**HZ**|**KHZ**|**MAHZ**|**GHZ**|**DBM**]

Sets the starting sweep parameter value of a segment, using entry block controls. (**SWP PARAM** under **(System)**)

| Parameter | Range | Unit |
|------------------------|---|----------------|
| <i><numeric></i> | 100000 (= 100 k) to 1.82×10^9 (= 1.82 G) (Network and impedance analyzer) | Hz (frequency) |
| <i><numeric></i> | 0 to 1.82×10^9 (= 1.82 G) (Spectrum analyzer) | Hz (frequency) |
| | -60 to 20 | dBm (power) |

■ Query Response

{*numeric*} \langle new line \rangle \langle END \rangle

■ Equivalent SCPI Command

:CALCulate:LIMit:SEGment:CONTRol[:DATA] \square *<numeric>*

LIMSADD

Adds a new segment to the end of the limit list. (**ADD** under **(System)**; No query)

■ Equivalent SCPI Command

:CALCulate:LIMit:SEGment:ADD

LIMSDEL

Deletes a limit testing segment. (**DELETE** under **(System)**; No query)

■ Equivalent SCPI Command

:CALCulate:LIMit:SEGment:DELeTe

LIMSDEL

LIMSDON

Terminates a limit segment definition. (DONE under (System); No query)

■ Equivalent SCPI Command

:CALCulate:LIMit:SEGment:SAVE

LIMSEDI□[<numeric>]

Specifies which limit segment in the table to edit. When you want to define or modify the values of the specified segment, you do not have to enter <numeric> (the segment number).

(SEGMENT, EDIT under (System))

| Parameter | Description |
|-----------|--------------------------|
| <numeric> | Segment number, 1 to 18. |

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

LIMSEDI□<numeric> :CALCulate:LIMit:SEGment□<numeric>
(SEGMENT)

LIMSEDI□[<numeric>] :CALCulate:LIMit:SEGment:EDIT□<numeric> (No query)
(EDIT)

LIMU□<numeric>

Sets the upper limit value for a limit testing segment. (UPPER LIMIT under (System))

| Parameter | Range | Unit |
|-----------|---------------------|------|
| <numeric> | -5000000 to 5000000 | |

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

:CALCulate:LIMit:SEGment:UPPer□<numeric>

LMAX?□<numeric>

Outputs the *n*th peak value from the left of the analysis range. See “LMAX?” in Appendix I. (Instrument BASIC EXECUTE executable; No equivalent SCPI command)

| Parameter | Range | Unit |
|-----------|----------|------|
| <numeric> | 1 to 801 | |

■ Query Response

| Parameter | Description |
|-----------|---|
| <numeric> | Complex number (Data format: real, imaginary) |

LMARG □ <numeric>

Specify the value for the left margin of printed paper. (**PRINT SETUP** , **LEFT MARGIN** under **Copy**)

| Parameter | Range | Unit |
|-----------|--------|------|
| <numeric> | 0 to 5 | inch |

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

:HCOpy:DRIVer:LEFTMarg: □<numeric>

LMIN? □ <numeric>

Outputs the *n*th negative peak value from the left of the analysis range. See “LMIN?” in Appendix I. (Instrument BASIC EXECUTE executable; No equivalent SCPI command)

| Parameter | Range | Unit |
|-----------|----------|------|
| <numeric> | 1 to 801 | |

■ Query Response

| Parameter | Description |
|-----------|---|
| <numeric> | Complex number (Data format: real, imaginary) |

LISV

Displays a tabular listing of all the measured data points and their current values. (**LIST VALUES** under **Copy**); No query)

■ Equivalent SCPI Command

:DISPlay[:WINDow]:TEXT1:PAGE □1
:DISPlay[:WINDow]:TEXT1:STATe □{ON|1}

LISV

LVCDT \square \langle numeric \rangle [DB]

Sets the level cal data (adds an offset value to the measured value). (Spectrum analyzer only)
(LVL CAL DATA under \square Cal)

| Parameter | Range | Unit |
|-----------------------------|-----------|------|
| \langle numeric \rangle | -10 to 10 | dB |

■ Query Response

{numeric} \langle new line \rangle \langle END \rangle

■ Equivalent SCPI Command

:SENSe:CORRection:OFFSet[:MAGNitude] \square \langle numeric \rangle

LVLCAL

Measures the CAL OUT signal (20 MHz, -20 dBm) at the input selected by the MEAS command (automatically sets the level cal data). After executing this function, the instrument state is returned to the state that existed before executing this function. (Spectrum analyzer only)
(EXECUTE LVL CAL under \square Cal; No query)

■ Equivalent SCPI Command

:CALibration:AUTO \square ONCE

MATH \square {DATA|DDVM|DMNM|DPLM}

Sets the trace math operation. (DATA MATH: DATA, DATA-MEM, DATA+MEM, DATA/MEM under DATA MATH [] under \square Display)

| Parameter | Description |
|-----------|-------------------------------------|
| DATA | Turns OFF all data math functions. |
| DMNM | Subtracts the memory from the data. |
| DPLM | Adds the memory to the data. |
| DDVM | Divides the data by the memory. |

■ Query Response

{DATA|DMNM|DPLM|DDVM} \langle new line \rangle \langle END \rangle

■ Equivalent SCPI Command

MATH \square DATA :CALCulate:MATH2[:EXPRession]:NAME \square OFF
MATH \square DMNM :CALCulate:MATH2[:EXPRession]:NAME \square SUB
MATH \square DPLM :CALCulate:MATH2[:EXPRession]:NAME \square ADD
MATH \square DDVM :CALCulate:MATH2[:EXPRession]:NAME \square DIV

■ Example

```
OUTPUT @Hp4396;"MATH DATA"  
OUTPUT @Hp4396;":CALC:MATH2:NAME OFF"
```

MAXMLEV \square *<numeric>* **[DBM]**

Sets the maximum mixer level. (Spectrum analyzer only) (MAX MIXER LEVEL under **Scale Ref**)

| Parameter | Range | Unit |
|------------------------|-------------|------|
| <i><numeric></i> | -100 to -10 | dBm |

- Query Response

{*numeric*} \langle new line \rangle \langle END \rangle

- Equivalent SCPI Command

:SENSe:POWer:AC:RANGe[:UPPer] \square *<numeric>*

MEAS \square *<parameter>*

Selects the parameters or inputs to be measured. (NETWORK: A/R, B/R, R, A, B, Refl: FWD S11 [A/R], Trans:FWD S21 [B/R], Trans:REV S12 [A/R], Refl: REV S22 [B/R], SPECTRUM: S, R, A, B, IMPEDANCE: MAG(|Z|), PHASE(θ_Z), RESIST(R), REACT(X), ADMITTNCE:MAG(|Y|), PHASE(θ_Y), CONDUCT(G), SUSCEPT(B), REFL.COEF:MAX(| Γ |), PHASE(θ_Γ)), REAL(Γ_X), IMAG(Γ_Y), CAPCITNCE:PRL(Cp), SER(Cs), INDUCTNCE:PRL(Lp) SER(Ls), RESISTNCE:PRL(Rp), SER(Rs), D FACTOR(D), Q FACTOR(Q) under **Meas**)

MEAS□<parameter>

| Parameter | Description |
|-----------|---|
| AR | A/R measurement (Network analyzer only) |
| BR | B/R measurement (Network analyzer only) |
| R | R measurement (Both Network and Spectrum analyzers) |
| A | A measurement (Both Network and Spectrum analyzers) |
| B | B measurement (Both Network and Spectrum analyzers) |
| S11 | S11 measurement (Network analyzer only) |
| S12 | S12 measurement (Network analyzer only) |
| S21 | S21 measurement (Network analyzer only) |
| S22 | S22 measurement (Network analyzer only) |
| S | S measurement (Spectrum analyzer only) |
| IMAG | Z measurement (Impedance analyzer only) |
| IPH | θ_z (Impedance analyzer only) |
| IRE | R (Impedance analyzer only) |
| IIM | X (Impedance analyzer only) |
| AMAG | Y (Impedance analyzer only) |
| APH | θ_y (Impedance analyzer only) |
| ARE | G (Impedance analyzer only) |
| AIM | B (Impedance analyzer only) |
| RCM | Γ (Impedance analyzer only) |
| RCPH | θ_{Γ} (Impedance analyzer only) |
| RCR | Γ_x (Impedance analyzer only) |
| RCIM | Γ_y (Impedance analyzer only) |
| CP | Parallel Capacitance, C_p (Impedance analyzer only) |
| CS | Series Capacitance, C_s (Impedance analyzer only) |
| LP | Parallel Inductance, L_p (Impedance analyzer only) |
| LS | Series Inductance, L_s (Impedance analyzer only) |
| D | Disipation Factor, D (Impedance analyzer only) |
| Q | Quality Factor, Q (Impedance analyzer only) |
| RP | Parallel Resistance, R_p (Impedance analyzer only) |
| RS | Series Resistance, R_s (Impedance analyzer only) |

■ Query Response

{ AR|RB|R|A|B|S11|S12|S21|S22|S|IMAG|IPH|IRE|IIM|AMAG|APH|ARE|AIM|RCM|RCPH|RCM|RCPH|RCR|RCIM|CP|CS|LP|LS|D|Q|RP|RS} <new line><END>

■ Equivalent SCPI Command

```
MEAS:AR :SENSe:FUNction"POWer:RATio3,2"
MEAS:BR :SENSe:FUNction"POWer:RATio4,2"
MEAS:R :SENSe:FUNction"POWer2"
MEAS:A :SENSe:FUNction"POWer3"
MEAS:B :SENSe:FUNction"POWer4"
MEAS:S11 :SENSe:FUNction"POWer:S11"
MEAS:S21 :SENSe:FUNction"POWer:S21"
MEAS:S12 :SENSe:FUNction"POWer:S12"
MEAS:S22 :SENSe:FUNction"POWer:S22"
MEAS:S :SENSe:FUNction"POWer1"
MEAS:IMAG CALCulate:MATH1[:EXPRession]:NAME IMPedance
CALCulate:FORMat MLINear
MEAS:IPH CALCulate:MATH1[:EXPRession]:NAME IMPedance
CALCulate:FORMat PHASe
```



```

MEAS□IRE      CALCulate:MATH1[:EXPRession]:NAME IMPedance
               CALCulate:FORMat REAL
MEAS□IIM      CALCulate:MATH1[:EXPRession]:NAME IMPedance
               CALCulate:FORMat IMAGinary
MEAS□AMAG     CALCulate:MATH1[:EXPRession]:NAME ADMittance
               CALCulate:FORMat MLINear
MEAS□APHS     CALCulate:MATH1[:EXPRession]:NAME ADMittance
               CALCulate:FORMat PHASe
MEAS□ARE      CALCulate:MATH1[:EXPRession]:NAME ADMittance
               CALCulate:FORMat REAL
MEAS□AIM      CALCulate:MATH1[:EXPRession]:NAME ADMittance
               CALC:FORMat IMAGinary
MEAS□RCM      CALCulate:MATH1[:EXPRession]:NAME OFF
               CALCulate:FORMat MLINear
MEAS□RCPH     CALCulate:MATH1[:EXPRession]:NAME OFF
               CALCulate:FORMat PHASe
MEAS□RCR      CALCulate:MATH1[:EXPRession]:NAME OFF
               CALCulate:FORMat REAL
MEAS□RCIM     CALCulate:MATH1[:EXPRession]:NAME OFF
               CALCulate:FORMat IMAGinary
MEAS□CP       CALCulate:MATH1[:EXPRession]:NAME IMPedance
               CALCulate:FORMat CP
MEAS□CS       CALCulate:MATH1[:EXPRession]:NAME IMPedance
               CALCulate:FORMat CS
MEAS□LP       CALCulate:MATH1[:EXPRession]:NAME IMPedance
               CALCulate:FORMat LP
MEAS□LS       CALCulate:MATH1[:EXPRession]:NAME IMPedance
               CALCulate:FORMat LS
MEAS□D        CALCulate:MATH1[:EXPRession]:NAME IMPedance
               CALCulate:FORMat D
MEAS□Q        CALCulate:MATH1[:EXPRession]:NAME IMPedance
               CALCulate:FORMat Q
MEAS□RP       CALCulate:MATH1[:EXPRession]:NAME IMPedance
               CALCulate:FORMat RP
MEAS□RS       CALCulate:MATH1[:EXPRession]:NAME IMPedance
               CALCulate:FORMat RS

```

■ Example

```

OUTPUT @Hp4396;"MEAS AR"
OUTPUT @Hp4396;"MEAS?"
ENTER @Hp4396;A$

OUTPUT @Hp4396;":SENS:FUNC ""POW:RAT 3,2""
OUTPUT @Hp4396;":SENS:FUNC?"
ENTER @Hp4396;A$

```

MEAS \square *<parameter>*

MEASSTAT \square {OFF|ON|0|1}

Calculates the mean, standard deviation, and peak-to-peak values in the portion of the displayed trace that is in the search range. (STATISTICS ON off under [Utility](#))

| Parameter | Description |
|-----------|---|
| OFF or 0 | Does not display the statistical values |
| ON or 1 | Displays the statistical values |

■ Query Response

{0|1} <new line><END>

■ Equivalent SCPI Command

:CALCulate:EVALuate:MStatistics[:STATe] \square {OFF|ON|0|1}

MKR \square {OFF|ON|0|1}

Sets the marker to active (ON) or inactive (OFF). When the MKR is turned off, the marker, sub-marker, and Δ marker are tuned to be off. ([MKR](#))

| Parameter | Description |
|-----------|--------------------------------|
| OFF or 0 | Turns off the marker function. |
| ON or 1 | Turns on the marker function. |

■ Query Response

{0|1} <new line><END>

■ Equivalent SCPI Command

:DISPlay[:WINDow]:TRACe{[1]|2}:MARKer[1]:ALL:STATe \square {OFF|ON|0|1}

(TRACe[1] for the data trace; TRACe2 for the memory trace.)

MKRAMPO

Moves the limits so that they are centered an equal amount above and below the marker at the sweep parameter value. ([MAKER \$\rightarrow\$ AMP.OFS.](#) under [System](#)); No query)

■ Equivalent SCPI Command

:CALCulate:LIMit:OFFSet \square MARKer

MKRAUV?

Outputs the auxiliary amplitude value (value 2) of the measurement value at the marker position. See “Marker Readout” in Appendix H for the auxiliary amplitude value of each display format. (Query only)

■ Query Response

{*numeric*} <new line><END>

■ Equivalent SCPI Command

:CALCulate:EVALuate:Y[1]:VALue2?

MKRCENT

Sets the sweep parameter center value of the destination channel to the sweep parameter value of the marker and centers the new span about that value. (MKR→CENTER under (Marker→); No query)

- Equivalent SCPI Command

:SENSe:FREQuency:CENTer␣MARKer (frequency) or
:SOURce:POWer:CENTer␣MARKer (power)

MKRCONT␣{OFF|ON|0|1}

Sets the continuous or discontinuous marker mode. (Network and impedance analyzer only) (MKR [] under (Marker))

| Parameter | Description |
|-----------|----------------------------|
| OFF or 0 | Discontinuous marker mode. |
| ON or 1 | Continuous marker mode. |

- Query Response

{0|1} <new line><END>

- Equivalent SCPI Command

:CALCulate:EVALuate:INTerpolate␣{OFF|ON|0|1}

MKRCOUP␣{OFF|ON|0|1}

Sets the coupled or uncoupled marker mode. (MKR [] under (Marker))

| Parameter | Description |
|-----------|-----------------------|
| OFF or 0 | Uncoupled marker mode |
| ON or 1 | Coupled marker mode |

- Query Response

{0|1} <new line><END>

- Equivalent SCPI Command

:CALCulate:EVALuate:COUPle␣{OFF|ON|0|1}

MKRCSTE

Sets the CENTER step size to the marker's sweep parameter value. (MKR→CNTR STEP under (Center); No query)

- Equivalent SCPI Command

:SENSe:FREQuency:CENTter:STEP[:INCRement]␣MARKer

MKRCSTE

MKRDCENT

Sets the sweep parameter center value of the destination channel to the difference value between the marker and the Δ marker values. (`MKRA→CENTER` under `Center` and `Marker→`); No query)

■ Equivalent SCPI Command

```
:SENSe:FREquency:CENTer $\Delta$ DMARker (frequency) or  
:SOURce:POWer:CENTer $\Delta$ DMARker (power)
```

MKRDCSTE

Sets the CENTER step size to the difference between the marker and Δ marker values. (`MKRA→CNTR STEP` under `Center`); No query)

■ Equivalent SCPI Command

```
:SENSe:FREquency:CENTer:STEP[:INCRement] $\Delta$ DMARker
```

MKRDELA

Sets the group delay at the marker point of a fixed frequency aperture, 20% of the span, to the electrical delay to balance the phase of the DUT. (Network analyzer only) (`MKR→DELAY` under `Scale Ref`); No query)

■ Equivalent SCPI Command

```
:SENSe:CORRection:EDELay2 $\Delta$ MARKer
```

MKRDSPAN

Sets the SPAN of the destination channel to the difference between the marker and the Δ marker values. (`MKRA→CENTER` under `Span` or `Marker→`); No query)

■ Equivalent SCPI Command

```
:SENSe:FREquency:SPAN $\Delta$ DMARker (frequency) or  
:SOURce:POWer:SPAN $\Delta$ DMARker (power)
```

MKRL Δ {OFF|ON|0|1}

Sets the maker list function ON or OFF. (`MKR LIST ON off` under `Utility`)

| Parameter | Description |
|-----------|--------------------------|
| OFF or 0 | Marker list function OFF |
| ON or 1 | Marker list function ON |

■ Query Response

```
{0|1} <new line><END>
```

■ Equivalent SCPI Command

```
:DISPlay[:WINDow]:TEXT16:STATe $\Delta$ {OFF|ON|0|1}
```

MKRMIDD

Sets the midpoint the LIMD command using the marker to set the middle amplitude value of a limit segment. (MKR→MIDDLE under (System); No query)

- Equivalent SCPI Command

```
:CALCulate:LIMit:SEGment:MIDDLE|MARKer
```

MKRNOI{OFF|ON|0|1}

Sets the noise format of the marker ON or OFF. This marker reads out the average noise level at the marker position (referenced to a 1 Hz noise power bandwidth). (Spectrum analyzer only) (NOISE FORM ON off under (Utility))

- Query Response

```
{0|1} <new line><END>
```

- Equivalent SCPI Command

```
:CALCulate:EVALuate:NOISe[:STATe]|{OFF|ON|0|1}
```

MKRO{DATA|MEMO}

Sets a trace from data or memory to be applied for the marker values. (MKR ON [] under (Marker))

| Parameter | Description |
|-----------|--------------|
| DATA | DATA TRACE |
| MEMO | MEMORY TRACE |

- Query Response

```
{DATA|MEMO} <new line><END>
```

- Equivalent SCPI Command

```
MKRO|DATA :CALCulate:EVALuate:ON|"DTR"
MKRO|MEMO :CALCulate:EVALuate:ON|"MTR"
```

- Example

```
OUTPUT @Hp4396;"MKRO DATA"
```

MKROFS

Sets the marker's amplitude value into the offset value. (MKR→OFFSET under (Display); No query)

- Equivalent SCPI Command

```
:DATA[:DATA]|OFFS,MARKer
```

MKROFS

MKRP \square \langle numeric \rangle

Moves the marker to the specified data point number.

| Parameter | Description |
|-----------------------------|-----------------------|
| \langle numeric \rangle | 1 to Number of Points |

■ Query Response

{*numeric*} \langle new line \rangle \langle ^END \rangle

■ Equivalent SCPI Command

:CALCulate:EVALuate:Y[1]:XPOSITION:POINT \square \langle numeric \rangle

MKRPKD

Sets the peak delta value to the smaller value of the difference of amplitude values between the present marker position and both side display points of the marker. (Network and impedance analyzer only) (MKR→PEAK DELTA under [Search](#)); No query)

■ Equivalent SCPI Command

:CALCulate:EVALuate:PEAK:EXCursion \square DMARKer

MKRPRM \square \langle numeric \rangle [HZ|KHZ|MAHZ|GHZ|DBM]

Sets the marker at the point of the specified sweep parameter, when the marker is ON.

| Parameter | Range | Unit |
|-----------------------------|---------------------------|-------------------------------|
| \langle numeric \rangle | start value to stop value | Hz (frequency) dBm (power) |

■ Query Response

{*numeric*} \langle new line \rangle \langle ^END \rangle

■ Equivalent SCPI Command

:CALCulate:EVALuate:Y[1]:XPOSITION \square \langle numeric \rangle

MKRREF

Makes the reference value of the destination channel equal to the marker's absolute value (regardless of the Δ marker value). (MKR→REFERENCE under [Scale Ref](#) and [Marker→](#)); No query)

■ Equivalent SCPI Command

:DISPlay[:WINDow]:TRACe{[1]|2}:Y[:SCALe]:RLEVEL \square MARKer

(TRACe[1] for the data trace; TRACe2 for the memory trace.)

MKRSTAR

Sets the sweep parameter start value of the destination channel to the sweep parameter value of the marker. (SEGMENT: MKR→START under (Sweep), or MKR→START under (Marker→); No query)

- Equivalent SCPI Command

```
SEGMENT: MKR→START under (Sweep) :SENSe:LIST:SEGment:FREQuency:STARt␣MARKer
MKR→START under (Marker→) :SENSe:FREQuency:STARt␣MARKer (frequency) or
:SOURce:POWer:STARt␣MARKer (power)
```

MKRSTOP

Sets the sweep parameter stop value of the destination channel to the sweep parameter value of the marker. (MKR→STOP under (Sweep), or MKR→STOP under (Marker→); No query)

- Equivalent SCPI Command

```
MKR→STOP under (Sweep) :SENSe:LIST:SEGment:FREQuency:STOP␣MARKer
MKR→STOP under (Marker→) :SENSe:FREQuency:STOP␣MARKer (frequency) or
:SOURce:POWer:STOP␣MARKer (power)
```

MKRSWPRM

Sets the segment sweep parameter value to the present marker sweep parameter value. (MKR→SWP PARAM under (System); No query)

- Equivalent SCPI Command

```
:CALCulate:LIMit:SEGment:CONTrol[:DATA]␣MARKer
```

MKRTHRE

Sets the threshold value to the amplitude value of the present marker position. (MKR→THRESHOLD under (Search); No query)

- Equivalent SCPI Command

```
:CALCulate:EVALuate:PEAK:THReshold␣MARKer
```

MKRTHRE

MKRTIME [] {OFF|ON|0|1}

Sets the x-axis units to time, (the start point is zero and the stop point is the value of the sweep time). (MKR TIME ON off under [Utility](#))

| Parameter | Description |
|-----------|--|
| OFF or 0 | Sets the x-axis to the sweep parameter |
| ON or 1 | Sets the x-axis to time |

■ Query Response

{0|1} <new line><END>

■ Equivalent SCPI Command

:DISPlay[:WINDow]:TRACe{[1]|2}:MARKer[1]:UNIT:TIME [] {OFF|ON|0|1}

■ Example

```
OUTPUT @Hp4396;"MKRTIME ON"
```

MKRVAL?

Outputs the amplitude value of the measurement value at the marker position. See “Marker Readout” in Appendix H for the amplitude value of each display format. (Query only)

■ Query Response

{*numeric*} <new line><END>

■ Equivalent SCPI Command

:CALCulate:EVALuate:Y[1]:VALue1?

■ Example

```
OUTPUT @Hp4396;"MKRVAL?"  
ENTER @Hp4396;A
```

MKRZM

Sets the sweep parameter center value of the destination channel to the sweep parameter value of the marker, and changes the sweep parameter span value of the destination channel to “sweep parameter span × zooming aperture.” (MKR ZOOM under [Marker](#)); No query

■ Equivalent SCPI Command

:SENSe:FREQuency:SPAN [] MZAPerture (frequency) or
:SOURce:POWer:SPAN [] MZAPerture (power)

MODI1

Leads to the modify calibration kit menu, where a calibration kit can be user-modified. (Network and impedance analyzer only) (MODIFY [] under [Cal](#)); No query

■ Equivalent SCPI Command

:SENSe:CORRection:CKIT:MODify

MODICOMK

Leads to the modify fixture compensation kit menu. (MODIFY [] under (Cal) COMPEN KIT[]; No query; Impedance analyzer only.)

- Equivalent SCPI Command

```
:SENSe:CORRection2:CKIT:MODify
```

MODIFIX

Leads to the modify user fixture menu. (MODIFY [] under (Meas) FIXTURE[]; No query; Impedance analyzer only.)

- Equivalent SCPI Command

```
:SYSTem:FIXTure:MODify
```

MONDYEAR

Changes the displayed date to the “month:day:year” format. (DATE MODE: MonDayYear under (System))

- Query Response

```
{0|1} <new line><END>
```

| Parameter | Description |
|-----------|-------------------------|
| 0 | “day:month:year” format |
| 1 | “month:day:year” format |

- Equivalent SCPI Command

```
:SYSTem:DATE:MODE:MDY
```

NA

Selects the network analyzer as the analyzer type. (NETWORK ANALYZER under (Meas))

- Query Response

```
{0|1}<new line><END>
```

| Parameter | Description |
|-----------|-----------------------------------|
| 0 | Network analyzer is not selected. |
| 1 | Network analyzer is selected. |

- Equivalent SCPI Command

```
INSTRument:TYPE:NA
```

- Example

```
OUTPUT @Hp4396;"NA?"
ENTER @Hp4396;Na
If 1 THEN PRINT "Network Analyzer Mode is selected."
```

NA

NEXP

Displays the next page of information in a tabular listing. (NEXT PAGE under Copy); No query)

■ Equivalent SCPI Command

```
:DISPlay[:WINDow]:TEXT{1-17}:PAGEUP
```

| Parameter | Description |
|-----------|-------------------------------|
| 1 | List Value |
| 2 | Operation Parameter |
| 3 | Cal Class |
| 4 to 11 | Cal Standard No. 1 to No. 8 |
| 12 | Start and Stop |
| 13 | Center and Span (List) |
| 14 | Upper and Lower |
| 15 | Middle and Delta (Limit Test) |
| 16 | Marker List |
| 17 | Title |

NEXPK?

Outputs the maximum peak value and its stimulus next to the peak last found by the PEAK?, or NEXPK? commands. For more information, see “NEXPK?” in Appendix I. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only)

NUMG<numeric>

Triggers a user-specified number of sweeps and returns to the HOLD mode. (NUMBER OF GROUPS under Trigger); No query)

| Parameter | Description |
|-----------|--|
| <numeric> | 1 to 999 (if <numeric> is 0 or less than 0, it is set to 1.) |

■ Equivalent SCPI Command (Query)

```
:INITiate:CONTinuous{OFF|0}  
:ABORt  
:SENSe:SWEep:COUNt<numeric>  
:INITiate[:IMMediate]
```

■ Query Response

```
{numeric} <new line><^END>
```

■ Example

```
OUTPUT @Hp4396;"NUMG 10"  
OUTPUT @Hp4396;":INIT:CONT OFF"  
OUTPUT @Hp4396;":SENS:SWE:COUN 10"  
OUTPUT @Hp4396;":INIT"
```

NUMLMAX?

Outputs the number of peaks within the analysis range. See “NUMLMAX?” in Appendix I. (Instrument BASIC EXECUTE executable; No equivalent SCPI command)

■ Query Response

<numeric><new line><^END>

NUMLMIN?

Outputs the number of negative peaks within the analysis range. See “NUMLMIN?” in Appendix I. (Instrument BASIC EXECUTE executable; No equivalent SCPI command)

■ Query Response

<numeric><new line><^END>

OFSD□<numeric>[S]

Specifies the one-way electrical delay from the measurement (reference) plane to the standard. (Network and impedance analyzer only) (OFFSET DELAY under **Cal**); No query)

| Parameter | Range | Unit |
|-----------|------------|------|
| <numeric> | - 10 to 10 | s |

■ Equivalent SCPI Command (Query)

:SENSe:CORRection:CKIT:STANdard:ODELay□<numeric>

■ Query Response

{numeric} <new line><^END>

OFSL□<value>

Specifies energy loss, due to skin effect, along a one-way length of coaxial cable offset. (Network and impedance analyzer only) (OFFSET LOSS under **Cal**); No query)

| Parameter | Range | Unit |
|-----------|-------------------------|------|
| <value> | 0 to 1×10^{19} | Ω/s |

■ Equivalent SCPI Command (Query)

:SENSe:CORRection:CKIT:STANdard:OLOSS□<numeric>

■ Query Response

{numeric} <new line><^END>

OFSL \square *<value>*

OFSZ \square *<numeric>* [**OHM**|**KOHM**]

Specifies the characteristic impedance of the coaxial cable offset. (Network and impedance analyzer only) (**OFFSET Z0** under **[Cal]**); No query)

| Parameter | Range | Unit |
|------------------------|------------------------|----------|
| <i><numeric></i> | 0.001 to 5000000 (=5M) | Ω |

■ Equivalent SCPI Command (Query)

:SENSe:CORRection:CKIT:STANdard:OCIMpedance \square *<numeric>*

■ Query Response

{*numeric*} <new line><^END>

OMII

Omits correction for isolation of a 2-port calibration. (Network analyzer only) (**OMIT ISOLATION** under **[Cal]**); No query)

■ Equivalent SCPI Command

:SENSe:CORRection:COLLect[:ACQuire] \square OMII

OPEP

Provides a tabular listing on the display of the key parameters for both channels. (**OPERATING PARAMETERS** under **[Copy]**); No query)

■ Equivalent SCPI Command

:DISPlay[:WINDow]:TEXT2:PAGE \square 1
:DISPlay[:WINDow]:TEXT2:STATe \square {ON|1}

OSE \square *<numeric>*

Enables the operational status register.

| Parameter | Description |
|------------------------|--|
| <i><numeric></i> | Decimal expression of the contents of the register; 0 to 65535 ($=2^{16}-1$) |

■ Query Response

{*numeric*} <new line><^END>

■ Equivalent SCPI Command

:STATus:OPERation:ENABle \square *<numeric>*

OSER?

Outputs the current value in the event register of an operational status register. (Query only)

- Query Response

{*numeric*} <new line><^END>

- Equivalent SCPI Command

:STATus:OPERation[:EVENT]?

OSNT□<*numeric*>

Sets the negative transition filter of an operational status register. For details, refer to Appendix D.

| Parameter | Description |
|--------------------|--|
| < <i>numeric</i> > | Decimal expression of the contents of the register, 0 to 65535 ($=2^{16}-1$) |

- Query Response

{*numeric*} <new line><^END>

- Equivalent SCPI Command

:STATus:OPERation:NTRansition□<*numeric*>

OSPT□<*numeric*>

Sets the positive transition filter of an operational status register. For details, refer to Appendix D.

| Parameter | Description |
|--------------------|--|
| < <i>numeric</i> > | Decimal expression of the contents of the register, 0 to 65535 ($=2^{16}-1$) |

- Query Response

{*numeric*} <new line><^END>

- Equivalent SCPI Command

:STATus:OPERation:PTRansition□<*numeric*>

OSR?

Outputs the operational status register value. (Query only)

- Query Response

{*numeric*} <new line><^END>

- Equivalent SCPI Command

:STATus:OPERation:CONDition?

OSR?

OUT8IO \square \langle numeric \rangle

Outputs the data to the 8-bit parallel output port. (No query)

| Parameter | Description |
|-----------------------------|-------------|
| \langle numeric \rangle | 0 to 255 |

■ Equivalent SCPI Command

```
:SYSTem:COMMunicate:PARAllel:TRANsmit:DATA  $\square$   $\langle$ numeric $\rangle$ 
```

OUTPCALC{1-12}?

Outputs the active calibration set array of the active channel. Refer to Appendix F for the calibration set array. (Network and impedance analyzer only) (Query only)

■ Query Response

$\{$ numeric (1) $\}$ $\{$ numeric (2) $\}$... $\{$ numeric (n) $\}$ \langle new line \rangle \langle END \rangle (n is the number of points.)

numeric is a complex number (data format: real, imaginary).

■ Equivalent SCPI Command

```
:DATA[:DATA]?  $\square$ CC0{1-12}
```

■ Example

```
DIM A(1:201,1:2)          NOP: 201
OUTPUT 717;"OUTCALC1?"
ENTER 717;A(*)

DIM A(1:201,1:2)
OUTPUT 717;" :DATA? CC01"
ENTER 717;A(*)
```

OUTPCALK?

Outputs the active calibration kit. (Network and impedance analyzer only) (Query only)

■ Query Response

$\{$ block data (714 bytes of binary data) $\}$ \langle new line \rangle \langle END \rangle

■ Equivalent SCPI Command

```
:DATA[:DATA]?  $\square$ CKIT
```

OUTPCERR?

Outputs ceramic resonator parameters. See "OUTPCERR?" in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only; Network analyzer only)

OUTPCFIL?

Outputs ceramic filter parameters. See “OUTPCFIL?” in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only; Network analyzer only)

OUTPCOMC{1|2|3}?

Outputs data of the fixture compensation arrays. See “INPUCOMC{1|2|3}□<numeric>” for details about the fixture compensation arrays. (Impedance analyzer only)

■ Query Response

{*numeric (1)*} {*numeric (2)*} ... {*numeric (n)*} <new line><END>

(n is the number of points.)

numeric is a complex number. (data format: real, imaginary)

■ Equivalent SCPI Command

:DATA[:DATA]?□CMP{1|2|3}

OUTPDATA?

Outputs the error corrected data. (Query only)

■ Query Response

{*numeric (1)*} {*numeric (2)*} ... {*numeric (n)*} <new line><END>

(n is the number of points.)

numeric is a complex number (data format: real, imaginary) for the Network analyzer, or a real number for the Spectrum analyzer.

■ Equivalent SCPI Command

:DATA[:DATA]?□DATA

■ Example

```
DIM A(1:201,1:2)           Network Analyzer, NOP: 201
OUTPUT 717;"OUTPDATA?"
ENTER 717;A(*)

DIM A(1:201,1:2)
OUTPUT 717;":DATA? DATA"
ENTER 717;A(*)
```

OUTPDATA?

OUTPDATAP? □ <numeric>

Outputs the error corrected data at the specified point. (Query only)

| Parameter | Description |
|-----------|---|
| <numeric> | 1 to "number of points" (If <numeric> is 0 or less than 0, it is set to 1. If <numeric> is greater than "number of points," it is set to "number of points.") |

■ Query Response

{*numeric (real)*} {*numeric (imaginary)*} <new line><END> (Network analyzer)

{*numeric (val)*} <new line><END> (Spectrum analyzer)

■ Equivalent SCPI Command

:DATA[:DATA]:VALue? □ DATA, <numeric>

■ Example

```
OUTPUT 717;"OUTPDATAP? 1"  
ENTER 717;A,B           Network Analyzer  
  
OUTPUT 717;":DATA:VAL? DATA,1"  
ENTER 717;A,B
```

OUTPDMKR?

Outputs sweep parameter and measurement value at the Δmarker position. (Query only)

■ Query Response

{*numeric (val1)*} {*numeric (val2)*} {*numeric (stimulus)*} <new line><END>

(Val1: Amplitude value, Val2: Auxiliary amplitude value. See "Marker Readout" in Appendix H.)

■ Equivalent SCPI Command

:CALCulate:EVALuate:REFerence:DATA?

■ Example

```
OUTPUT 717;"OUTPDMKR?"  
ENTER 717;A,B,C  
  
OUTPUT 717;":CALC:EVAL:REF:DATA?"  
ENTER 717;A,B,C
```


OUTPDTRC?

Outputs DATA TRACE data. (Query only)

■ Query Response

{*numeric (1:val1)*} {*numeric (1:val2)*} {*numeric (2:val1)*} {*numeric (2:val2)*} ...
 {*numeric (n:val2)*} {*numeric (n:val2)*} <new line><END> (Network analyzer)

{*numeric (1)*} {*numeric (2)*} ... {*numeric (n)*} <new line><END> (Spectrum analyzer)

(n is the number of points.) (Val1: Amplitude value, Val2: Auxiliary amplitude value. See “Marker Readout” in Appendix H.)

■ Equivalent SCPI Command

:TRACe[:DATA]?□DTR

■ Example

```
DIM A(1:201,1:2)           Network Analyzer, NOP: 201
OUTPUT 717;"OUTPDTRC?"
ENTER 717;A(*)

DIM A(1:201,1:2)
OUTPUT 717;":TRAC? DTR"
ENTER 717;A(*)
```

OUTPDTRCP?□<numeric>

Outputs DATA TRACE data at the specified point. (Query only)

| Parameter | Description |
|-----------|---|
| <numeric> | 1 to “number of points” (If <numeric> is 0 or less than 0, it is set to 1. If <numeric> is greater than “number of points,” it is set to “number of points.”) |

■ Query Response

{*numeric (val1)*} {*numeric (val2)*} <new line><END> (Network analyzer)

{*numeric (val)*} <new line><END> (Spectrum analyzer)

(Val1: Amplitude value, Val2: Auxiliary amplitude value. See “Marker Readout” in Appendix H.)

■ Equivalent SCPI Command

:TRACe[:DATA]:VALue?□DTR,<numeric>

■ Example

```
OUTPUT 717;"OUTPDTRCP? 1"
ENTER 717;A,B

OUTPUT 717;":TRAC:VAL? DTR,1"
ENTER 717;A,B
```

OUTPDTRCP? \square \langle numeric \rangle

OUTPERRO?

Outputs the error message in the error queue.

■ Query Response

$\{$ numeric (Error number) $\}$ $\{$ string (Error message) $\}$ \langle new line \rangle \langle END \rangle

■ Equivalent SCPI Command

:SYSTem:ERRor?

■ Example

```
OUTPUT 717;"OUTPERRO?"
ENTER 717;A,A$

OUTPUT 717;" :SYST:ERR?"
ENTER 717;A,A$
```

OUTPFAIP?

Outputs number of the failed point of the limit test. (Query only)

■ Query Response

$\{$ numeric $\}$ \langle new line \rangle \langle END \rangle

■ Equivalent SCPI Command

:DATA:POINts? \square LFA

■ Example

```
OUTPUT 717;"OUTPFAIP?"
ENTER 717;A

OUTPUT 717;" :DATA:POIN? LFA"
ENTER 717;A
```

OUTPFILT?

Outputs filter parameters. See "OUTPFILT?" in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only; Network analyzer only)

OUTPLIMF?

Outputs the limit test results only for the failed points. (Query only)

■ Query Response

$\{$ numeric (stimulus 1) $\}$ $\{0\}$ $\{$ numeric (upper_limit 1) $\}$ $\{$ numeric (lower_limit 1) $\}$
 $\{$ numeric (stimulus 2) $\}$ $\{0\}$ $\{$ numeric (upper_limit 2) $\}$ $\{$ numeric (lower_limit 2) $\}$
 \vdots
 $\{$ numeric (stimulus n) $\}$ $\{0\}$ $\{$ numeric (upper_limit n) $\}$ $\{$ numeric (lower_limit n) $\}$ \langle new
line \rangle \langle END \rangle (Form 4)

(n is the number of failed points.)

$\{0\}$ \langle new line \rangle \langle END \rangle (for no failed points.)

- Equivalent SCPI Command

```
:DATA[:DATA]?LFA
```

- Example

```
DIM A(1:201,1:4)          NOP: 201
OUTPUT 717;"OUTPLIMF?"
ENTER 717 USING "%,K";A(*)

DIM A(1:201,1:4)
OUTPUT 717;" :DATA? LFA"
ENTER 717 USING "%,K";A(*)
```

OUTPLIML?

Outputs the limit test results for each point. (Query only)

- Query Response

```
{numeric (stimulus 1)} {numeric (result 1)} {numeric (upper_limit 1)}
{numeric (lower_limit 1)}

{numeric (stimulus 2)} {numeric (result 2)} {numeric (upper_limit 2)}
{numeric (lower_limit 2)}

:

{numeric (stimulus n)} {numeric (result n)} {numeric (upper_limit n)}
{numeric (lower_limit n)} <new line><END> (Form 4)
```

(n is the number of points.) (*result* is 1 for pass, 0 for fail, or -1 for no test.)

- Equivalent SCPI Command

```
:DATA[:DATA]?LLIS
```

- Example

```
DIM A(1:201,1:4)          NOP: 201
OUTPUT 717;"OUTPLIML?"
ENTER 717;A(*)

DIM A(1:201,1:4)
OUTPUT 717;" :DATA? LLIS"
ENTER 717;A(*)
```

OUTPLIML?

OUTPLIMM?

Outputs the limit test result for the marker position. (Query only)

■ Query Response

```
{numeric (stimulus)} {numeric (result)} {numeric (upper_limit)} {numeric (lower_limit)}  
<new line><END>
```

(*result* is 1 for pass, 0 for fail, or -1 for no test)

■ Equivalent SCPI Command

```
:DATA[:DATA]?LIMAR
```

■ Example

```
OUTPUT 717;"OUTPLIMM?"  
ENTER 717;A,B,C,D  
  
OUTPUT 717;":DATA? LMAR"  
ENTER 717;A,B,C,D
```

OUTPMAX?

Outputs maximum value within analysis range. See "OUTPMAX?" in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only)

OUTPMEAN?

Outputs mean value within analysis range. See "OUTPMEAN?" in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only)

OUTPMEMO?

Outputs the memory data from the active channel. (Query only)

■ Query Response

```
{numeric (1)} {numeric (2)} ... {numeric (n)} <new line><END>
```

(n is the number of points.)

numeric is a complex number (data format: real, imaginary) for the Network analyzer, or a real number for the Spectrum analyzer.

■ Equivalent SCPI Command

```
:DATA[:DATA]?MEM
```

■ Example

```
DIM A(1:201,1:2)           Network Analyzer; NOP: 201  
OUTPUT 717;"OUTPMEMO?"  
ENTER 717;A(*)  
  
DIM A(1:201,1:2)  
OUTPUT 717;":DATA? MEM"  
ENTER 717;A(*)
```

OUTPMEMOP? \square *<numeric>*

Outputs the memory data from the active channel at a specified point. (Query only)

| Parameter | Description |
|----------------------|---|
| <i><value></i> | 1 to "number of points" (If <i><numeric></i> is 0 or less than 0, it is set to 1. If <i><numeric></i> is greater than "number of points," it is set to "number of points.") |

- Query Response

{real} {imaginary} \langle new line \rangle \langle END \rangle (Network analyzer)

{numeric} \langle new line \rangle \langle END \rangle (Spectrum analyzer)

- Equivalent SCPI Command

:DATA[:DATA]:VALue? \square MEM, *<numeric>*

OUTPMIN?

Outputs minimum value within analysis range. See "OUTPMIN?" in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only)

OUTPMINMAX?

Outputs maximum and minimum value within analysis range. See "OUTPMIN?" in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only)

OUTPMKR?

Outputs the sweep parameter and measurement values at the marker position. (Query only)

- Query Response

{*numeric (val1)*} {*numeric (val2)*} {*numeric (stimulus)*} \langle new line \rangle \langle END \rangle

(Val1: Amplitude value, Val2: Auxiliary amplitude value. See "Marker Readout" in Appendix H.)

- Equivalent SCPI Command

:CALCulate:EVALuate:Y:DATA?

- Example

```
OUTPUT 717;"OUTPMKR?"
ENTER 717;A,B,C

OUTPUT 717;":CALC:EVAL:Y:DATA?"
ENTER 717;A,B,C
```

OUTPMKR?

OUTPMSTA?

Outputs the marker statistics. (STATISTICS ON off under Utility); Query only)

■ Query Response

{*numeric (mean)*} {*numeric (standard deviation)*} {*numeric (peak to peak)*}
<new line><END>

■ Equivalent SCPI Command

:CALCulate:EVALuate:MStatistics:DATA?

■ Example

```
OUTPUT 717;"OUTPMSTA?"
ENTER 717;A,B,C

OUTPUT 717;":CALC:EVAL:MST:DATA?"
ENTER 717;A,B,C
```

OUTPMTRC?

Outputs the MEMORY TRACE data. (Query only)

■ Query Response

{*numeric (1:val1)*} {*numeric (1:val2)*} {*numeric (2:val1)*} {*numeric (2:val)*} ...
{*numeric (n:val1)*} {*numeric (n:val2)*} <new line><END> (Network analyzer)

{*numeric (1)*} {*numeric (2)*} ... {*numeric (n)*} <new line><END> (Spectrum analyzer)

(n is the number of points.) (Val1: Amplitude value, Val2: Auxiliary amplitude value. See "Marker Readout" in Appendix H.)

■ Equivalent SCPI Command

:TRACe[:DATA]?LMTR

■ Example

```
DIM A(1:201,1:2)           Network Analyzer, NOP: 201
OUTPUT 717;"OUTPMTRC?"
ENTER 717;A(*)

DIM A(1:201,1:2)
OUTPUT 717;":TRAC? MTR"
ENTER 717;A(*)
```

OUTPMTRCP? \square *<numeric>*

Outputs the MEMORY TRACE data at the specified point. (Query only)

| Parameter | Description |
|------------------------|---|
| <i><numeric></i> | 1 to "number of points" (If <i><numeric></i> is 0 or less than 0, it is set to 1. If <i><numeric></i> is greater than "number of points," it is set to "number of points.") |

- Query Response

{numeric (val1)} *{numeric (val2)}* <new line><END> (Network analyzer)

{numeric (val)} <new line><END> (Spectrum analyzer)

(Val1: Amplitude value, Val2: Auxiliary amplitude value. See "Marker Readout" in Appendix H.)

- Equivalent SCPI Command

:TRACe[:DATA]:VALue?LMTR,<numeric>

- Example

```
OUTPUT 717;"OUTPMTRCP? 1" Network Analyzer
ENTER 717;A,B
```

f

```
OUTPUT 717;":TRAC:VAL? MTR,1"
ENTER 717;A,B
```

OUTPMWID?

Outputs the results of the bandwidth search. (Network and impedance analyzer only)

(WIDTHS ON off under Search); Query only)

- Query Response

{numeric (bandwidth)} *{numeric (center)}* *{numeric (Q)}* <new line><END>

- Equivalent SCPI Command

:CALCulate:EVALuate:WIDTh:DATA?

- Example

```
OUTPUT 717;"OUTPMWID?"
ENTER 717;A,B,C
```

```
OUTPUT 717;":CALC:EVAL:WIDT:DATA?"
ENTER 717;A,B,C
```

OUTPMWID?

OUTPRAW{1-4}?

Outputs the uncorrected data arrays for the active channel. (Query only)

■ Query Response

{*numeric (1)*} {*numeric (2)*} ... {*numeric (n)*} <new line><END>

(n is the number of points.)

numeric is a complex number (data format: real, imaginary) for the Network analyzer, or a real number for the Spectrum analyzer.

■ Equivalent SCPI Command

:DATA[:DATA]? \square RAW{1-4}

OUTPRESF?

Outputs resonator parameters. See “OUTPRESF?” in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only; Network analyzer only)

OUTPRESO?

Outputs resonator parameters. See “OUTPRESO?” in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only; Network analyzer only)

OUTPRESR?

Outputs resonator parameters. See “OUTPRESR?” in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only; Network analyzer only)

OUTPSMKR{1-7}?

Outputs the measurement values and sweep parameter at the sub-marker position. (Query only)

■ Query Response

{*numeric (val1)*} {*numeric (val2)*} {*numeric (stimulus)*} <new line><END>

(Val1: Amplitude value, Val2: Auxiliary amplitude value. See “Marker Readout” in Appendix H.)

■ Equivalent SCPI Command

```
OUTPSMKR1? :CALCulate:EVALuate:Y2:DATA?  
OUTPSMKR2? :CALCulate:EVALuate:Y3:DATA?  
OUTPSMKR3? :CALCulate:EVALuate:Y4:DATA?  
OUTPSMKR4? :CALCulate:EVALuate:Y5:DATA?  
OUTPSMKR5? :CALCulate:EVALuate:Y6:DATA?  
OUTPSMKR6? :CALCulate:EVALuate:Y7:DATA?  
OUTPSMKR7? :CALCulate:EVALuate:Y8:DATA?
```


OUTPSWPRM?

Outputs the sweep parameter data. (Query only)

■ Query Response

{*numeric 1*} {*numeric 2*} ... {*numeric n*} <new line><END>

(n is the number of points.)

■ Equivalent SCPI Command

:DATA[:DATA]? \square SPAR

OUTPSWPRMP? \square <*numeric*>

Outputs the sweep parameter data at a specified point. (Query only)

| Parameter | Description |
|--------------------|---|
| < <i>numeric</i> > | 1 to “number of points” (If < <i>numeric</i> > is 0 or less than 0, it is set to 1. If < <i>numeric</i> > is greater than “number of points,” it is set to “number of points.”) |

■ Query Response

{numeric} <new line><END>

■ Equivalent SCPI Command

:DATA[:DATA]:VALue? \square SPAR,<*numeric*>

OUTPXFIL?

Outputs crystal filter parameters. See “OUTPXFIL?” in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only; Network analyzer only)

OUTPXFIL?

PARS □ {OFF|ON|0|1}

Sets the partial search of the marker search function ON or OFF. (PART SRCH ON off under **Search**)

| Parameter | Description |
|-----------|--------------------|
| OFF or 0 | Partial search OFF |
| ON or 1 | Partial search ON |

■ Query Response

{0|1} <new line><END>

■ Equivalent SCPI Command

:CALCulate:EVALuate:BAND:FULL[:STATe] □ {OFF|ON|0|1}

| Parameter | Description |
|-----------|--------------------|
| OFF or 0 | Partial search ON |
| ON or 1 | Partial search OFF |

■ Example

```
OUTPUT 717;"PARS ON"  
OUTPUT 717;"PARS?"  
ENTER 717;A  
  
OUTPUT 717;":CALC:EVAL:BAND:FULL OFF"  
OUTPUT 717;":CALC:EVAL:BAND:FULL?"  
ENTER 717;A
```

PEAK?

Outputs maximum peak within analysis range, and memorizes its position for the NEXPK? command. See "PEAK?" in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only)

PEAKCENT

Searches for a peak using the marker and then changes the CENTER of the destination channel to the sweep parameter value of that peak. (PEAK—CENTER under **Center** or **Marker→**); No query)

■ Equivalent SCPI Command

:SENSe:FREQUency:CENTer □ TPEak (frequency) or
:SOURce:POWer:CENTer □ TPEak (power)

PKDLTX \square *<numeric>* [**HZ**|**KHZ**|**MAHZ**|**GHZ**|**DBM**]

PHAO \square *<numeric>* [**DEG**]

Adds or subtracts a phase offset that is constant with frequency. (Network analyzer only)
(**PHASE OFFSET** under **Scale Ref**)

| Parameter | Range | Unit |
|------------------------|--------------|------|
| <i><numeric></i> | -360 to +360 | ° |

■ Query Response

{*numeric*} <new line><END>

■ Equivalent SCPI Command

:SENSe:CORRection:OFFSet:PHASe \square *<numeric>*

PHAU {**RAD**|**DEG**}

Selects the unit of phase format. (**PHASE UNIT** \square under **Format**); Impedance analyzer only.)

| Parameter | Description |
|-----------|-------------|
| DEG | Degree. |
| RAD | Radian. |

■ Query Response

{DEG|RAD}<new line><END>

■ Equivalent SCPI Command

:CALCulate:FORMat:UNIT:ANGLE \square {DEG|RAD}

PKDLTX \square *<numeric>* [**HZ**|**KHZ**|**MAHZ**|**GHZ**|**DBM**]

Sets the peak ΔX value that is used to define the peak. (Network and impedance analyzer only)
(**PEAK DEF: ΔX** under **Search**)

| Parameter | Range | Unit |
|------------------------|----------------------|-------------------------------|
| <i><numeric></i> | 0 to 8 G 0 to 500 | Hz (frequency) dBm (power) |

■ Query Response

{*numeric*} <new line><END>

■ Equivalent SCPI Command

:CALCulate:EVALuate:PEAK:EXCursion:X \square *<numeric>*

PKDLTX \square *<numeric>* [**HZ|KHZ|MAHZ|GHZ|DBM**]

■ Example

```
OUTPUT 717;"PKDLTX 1E6"  
OUTPUT 717;"PKDLTX?"  
ENTER 717;A  
OUTPUT 717;":CALC:EVAL:PEAK:EXC:X 1E6"  
OUTPUT 717;":CALC:EVAL:PEAK:EXC:X?"  
ENTER 717;A
```

PKDLTY \square *<numeric>*

Sets the peak ΔY value that is used to define the peak. (**PEAK DEF: ΔY** under **Search**)

| Parameter | Range | Format |
|------------------------|--|--|
| <i><numeric></i> | 0 to 500 (spectrum analyzer) 0 to 500k 0 to 500 (network analyzer) 0 to 500k (any format in network analyzer except log magnitude format, and impedance analyzer) | (dBm, dB μ V, dBV formats) (Watt format, Volt format) (Log magnitude format) |

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

:CALCulate:EVALuate:PEAK:EXCursion:Y \square *<numeric>*

PKPOL \square {**POS|NEG**}

Sets the peak polarity for the marker search functions. (Network and impedance Analyzer only) (**PEAK PLRTY pos neg** under **Search**)

| Parameter | Description |
|-----------|---------------|
| POS | Positive peak |
| NEG | Negative peak |

■ Query Response

{POS|NEG} <new line><^END>

■ Equivalent SCPI Command

:CALCulate:EVALuate:PEAK:POLarity \square {POSitive|NEGative}

PKTHRE \square {OFF|ON|0|1}Sets the threshold ON or OFF. (THRESHOLD ON off under [Search](#))

| Parameter | Description |
|-----------|---------------|
| OFF or 0 | Threshold OFF |
| ON or 1 | Threshold ON |

■ Query Response

{0|1} <new line><^END>

■ Equivalent SCPI Command

:CALCulate:EVALuate:PEAK:THReshold:STATe \square {OFF|ON|0|1}**PKTHVAL** \square *<value>*Sets the threshold values. (THRESHOLD VALUE under [Search](#))

| Parameter | Range | Unit |
|------------------------|--|--|
| <i><numeric></i> | -500 to 500 | (Log mag format) |
| | -5×10^6 to 5×10^6 | (Phase, Expanded phase, Lin man, Real, Imaginary, SWR formats) |
| | -0.5 to 0.5 | (Delay format) |
| | 1×10^{-11} to 500 | (Smith chart, Admittance chart, Polar formats) |
| | -100 to 30 | (dBm format) |
| | -113 to 17 (50 Ω) | (dBV format) |
| | -111.2 to 18.8 (75 Ω) | |
| | 7 to 137 (50 Ω) | (dB μ V format) |
| | 8.8 to 138.8 (75 Ω) | |
| | 1×10^{-13} to 1 | (Watt format) |
| | 1×10^{-6} to 7.071 (50 Ω) | (Volt format) |
| | 1×10^{-6} to 8.66 (75 Ω) | |

■ Query Response

<numeric> <new line><^END>

■ Equivalent SCPI Command

:CALCulate:EVALuate:PEAK:THReshold \square *<numeric>*

PKTHVAL □ *<value>*

POIN □ *<numeric>*

Sets the number of points for the segment, or sets the number of points for the list sweep table. (**NUMBER OF POINTS** under **(Sweep)**)

| Parameter | Description |
|------------------------|------------------------|
| <i><numeric></i> | 2 to 801. ¹ |

¹ For the spectrum analyzer, *<numeric>* can be set when the SPAN is set to zero. When the SPAN is not zero, this command is query only.

■ Query Response

{*numeric*} <new line><^END>

■ Equivalent SCPI Command

:SENSe:SWEEp:POINts □ *<numeric>* or

:SENSe:LIST:SEGMENT:POINts □ *<numeric>* (List sweep)

POLE? □ *<numeric>*

Outputs the first found negative peaks for both side from the maximum peak. Negative peaks must be lower than the *<numeric>* down from the maximum peak. See “POLE?” in Appendix I for details, command paramter, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; No query)

PORE □ {**OFF**|**ON**|**0**|**1**}

Sets the reference plane extension mode ON or OFF. (Network and impedance analyzer only) (**EXTENSIONS ON off** under **(Cal)**)

| Parameter | Description |
|-----------|------------------------------------|
| OFF or 0 | Reference plane extension mode OFF |
| ON or 1 | Reference plane extension mode ON |

■ Query Response

{0|1} <new line><^END>

■ Equivalent SCPI Command

:SENSe:CORRection:EDElay:STATe □ {OFF|ON|0|1}

PORT1□<numeric>[S|MS|US|NS|PS]

Extends the reference plane for measurement of S_{11} , S_{21} , and S_{12} . (Network analyzer only)
 (EXTENSION PORT 1 under Cal)

| Parameter | Range | Unit |
|-----------|------------|------|
| <numeric> | - 10 to 10 | s |

■ Query Response

{numeric} <new line><END>

■ Equivalent SCPI Command

:SENSe:CORRection:EDELay:PORT1:TIME□<numeric>

PORT2□<numeric>[S|MS|US|NS|PS]

Extends the reference plane for measurement of S_{22} , S_{12} , and S_{21} . (Network analyzer only)
 (EXTENSION PORT 2 under Cal)

| Parameter | Range | Unit |
|-----------|------------|------|
| <numeric> | - 10 to 10 | s |

■ Query Response

{numeric} <new line><END>

■ Equivalent SCPI Command

:SENSe:CORRection:EDELay:PORT2:TIME□<numeric>

PORTA□<numeric>[S|MS|US|NS|PS]

Adds electrical delay to the input A reference plan for all A input measurements (including S-parameters). (Network analyzer only) (EXTENSION INPUT A under Cal)

| Parameter | Range | Unit |
|-----------|------------|------|
| <value> | - 10 to 10 | s |

■ Query Response

{numeric} <new line><END>

■ Equivalent SCPI Command

:SENSe:CORRection:EDELay:PORT4:TIME□<numeric>

PORTA□<numeric>[S|MS|US|NS|PS]

PORTB□<numeric>[S|MS|US|NS|PS]

Adds electrical delay to the input B reference plane for all B input measurements (including S-parameters). (Network analyzer only) (EXTENSION INPUT B under Cal)

| Parameter | Range | Unit |
|-----------|-----------|------|
| <numeric> | -10 to 10 | s |

■ Query Response

{numeric} <new line><END>

■ Equivalent SCPI Command

:SENSe:CORRection:EDELay:PORT5:TIME□<numeric>

PORTR□<numeric>[S|MS|US|NS|PS]

Adds electrical delay to extend the reference plane at input R to the end of cable. (Network analyzer only) (EXTENSION INPUT R under Cal)

| Parameter | Range | Unit |
|-----------|-----------|------|
| <value> | -10 to 10 | s |

■ Query Response

{numeric} <new line><END>

■ Equivalent SCPI Command

:SENSe:CORRection:EDELay:PORT3:TIME□<numeric>

PORTZ□<numeric>

Sets the port extension value. (EXTENSION VALUE under Cal); Impedance analyzer only.)

■ Query response

<numeric><new line><END>

■ Equivalent SCPI command

:SENSe:CORRection1:EDELay:PORT6:[[:TIME]]□<numeric>

POWE□<numeric>**[DBM]**

Sets the power level segment by segment, or sets the power level for the list sweep table.

(**POWER** under (**Sweep**))

| Parameter | Range | Unit |
|-----------|------------|------|
| <numeric> | -60 to +20 | dBm |

- Query Response

{numeric} <new line><^END>

- Equivalent SCPI Command

:SOURCE:POWer[:LEVel][:IMMediate][:AMPLitude]□<numeric> or

:SENSe:LIST:SEGment:POWer <numeric> (List sweep)

PREP

Displays the previous page of information in a tabular listing. (**PREV PAGE** under (**Copy**); No query)

- Equivalent SCPI Command

:DISPlay[:WINDow]:TEXT{1-17}:PAGE□DOWN

| Parameter | Description |
|-----------|-------------------------------|
| 1 | List Value |
| 2 | Operation Parameter |
| 3 | Cal Class |
| 4 to 11 | Cal Standard No. 1 to No. 8 |
| 12 | Start and Stop |
| 13 | Center and Span (List) |
| 14 | Upper and Lower |
| 15 | Middle and Delta (Limit Test) |
| 16 | Marker List |
| 17 | Title |

PRES

Presets the ANALYZER to the preset default values. See Appendix D of the *Function Reference* for the default values. The PRES command does *not* preset the HP instrument BASIC.

(**PRESET**); No query)

- Equivalent SCPI Command

:SYSTem:PRESet

PRES

PRIC

Sets the print command to the color printing. (`COLOR` under `Copy`)

■ Query Response

{0|1} <new line><END>

| Parameter | Description |
|-----------|-----------------------|
| 0 | Single-color printing |
| 1 | Color printing |

■ Equivalent SCPI Command

:HCOpy:DRIVer:COLor{0N|1}

PRICFIXE

Sets the default colors for printing a hard copy. (`PRINT COLOR [FIXED]` under `Copy`)

■ Query Response

{0|1} <new line><END>

| Parameter | Description |
|-----------|---|
| 0 | Variable colors (colors similar to the display) |
| 1 | Fixed colors (default colors) |

■ Equivalent SCPI Command

:HCOpy:DRIVer:CMAP:COLor{FIXEd}

PRICVARI

Sets the colors used for printing a hard copy as close as possible to the display colors. Refer to “System Accessory Printer” in Chapter 9 of *FuncRef* for the printers which support the variable color printing. (`PRINT COLOR [VARIABLE]` under `Copy`)

■ Query Response

{0|1} <new line><END>

| Parameter | Description |
|-----------|---|
| 0 | Fixed colors (default colors) |
| 1 | Variable colors (colors similar to the display) |

■ Equivalent SCPI Command

:HCOpy:DRIVer:CMAP:COLor{VARIable}

PRINALL

Causes an extra copy of the display to be printed. (PRINT [] under **Copy**); No query)

- Equivalent SCPI Command

:HCOpy[:IMMediate]

PRIS

Sets the print command to the single color printing. (PRINT: STANDARD under **Copy**)

- Query Response

{0|1} <new line><END>

| Parameter | Description |
|-----------|-----------------------|
| 0 | Color printing |
| 1 | Single color printing |

- Equivalent SCPI Command

:HCOpy:DRIVer:COLor□{OFF|0}

- Query Response

| Parameter | Description |
|-----------|-------------------------------|
| 0 | Default printing (black only) |
| 1 | Color printing |

PRSOFT□{OFF|ON|0|1}

Sets printing the softkeys displayed in the screen ON or OFF. (COPY SKEY under **Copy**)

| Parameter | Description |
|-----------|------------------------------|
| OFF or 0 | Does not print the soft keys |
| ON or 1 | Print the soft keys |

- Query Response

{0|1} <new line><END>

- Equivalent SCPI Command

:HCOpy:DRIVer:SKEY□{OFF|ON|0|1}

PRSOFT \square {OFF|ON|0|1}

PRSMKRS

Turns off all markers and cancels all settings of the marker functions. (PRESET MKRS under (Marker); No query)

■ Equivalent SCPI Command

:DISPly[:WINDow]:TRACe{[1]|2}:MARKer[1]:ALL \square DEFault
(TRACe[1] for the data trace; TRACe2 for the memory trace.)

PURG

 \square <string>

Removes the file. (PURGE FILE under (SAVE); No query)

| Parameter | Description |
|-----------|--|
| <string> | File name, up to 10 characters including the extension |

■ Equivalent SCPI Command

:MMEMory:DELeTe \square <string (file_name)> [, <string (msus)>]

| Parameter | Description |
|-----------------|--|
| <string (msus)> | "DISK" for the flexible disk drive "MEMORY" for the RAM disk memory |

■ Example

```
OUTPUT 717;"PURG ""TEST_S""  
OUTPUT 717;":MMEM:DEL ""TEST_S""
```

RAID

Completes the response and isolation calibration. Computes and stores the error coefficients. (Network analyzer only) (DONE RESP ISOL'N CAL under (Cal); No query)

■ Equivalent SCPI Command

:SENSe:CORRection:COLLect:SAVE3

RAISOL

Selects the isolation class for the response and isolation calibration. (Network analyzer only) (ISOL'N STD under (Cal); No query)

■ Equivalent SCPI Command

:SENSe:CORRection:COLLect[:ACQuire] \square ISOL

RAIRESP

Selects the response class for the response and isolation calibration. (Network analyzer only)
(**RESPONSE** under **Cal**); No query)

- Equivalent SCPI Command

```
:SENSe:CORRection:COLLect[:ACQuire]LRRESP
```

READ?

Reads data from a file that has been read-enabled using the **ROPEN** command. The returned data is in the fixed length block format defined in IEEE488.2. The fixed length block format, as shown in Figure 2-1, consists of a header part indicating the data size and an actual data part. In the case of the 4396B, the number of digits to indicate the data size is 6 and the maximum length of the actual data part is 16 Kbytes. If a file contains data greater than 16 Kbytes, execute this command repeatedly to read it. Note that acceptable file formats for this command are the DOS format and the LIF format BDAT type.

Generally, this command is used in combination with the **ROPEN** command and the **CLOSE** command, as shown in Figure 2-2. (Query only)

- Query Response

```
{block} <new line><END>
```

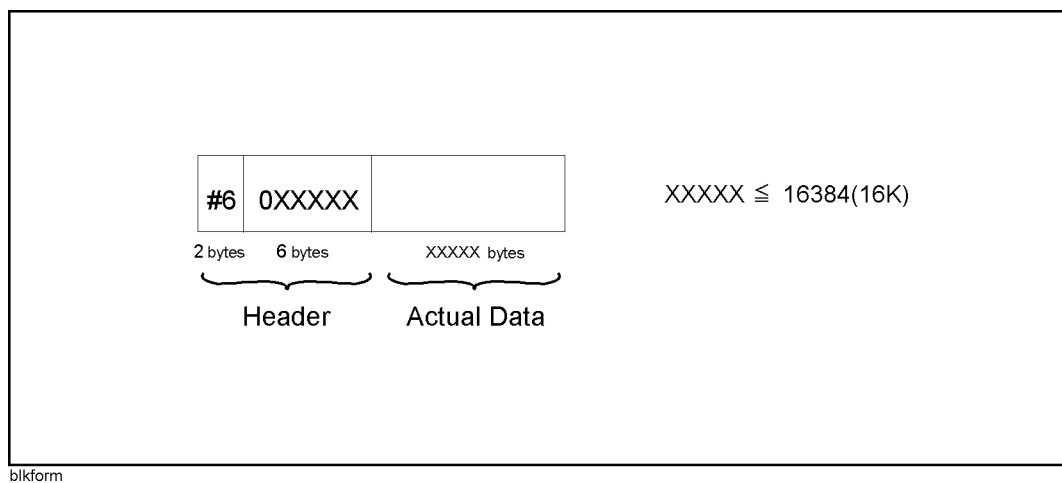


Figure 2-1. Fixed length block format

RECC

Recalls the previously saved version of the color set from the non-volatile memory.
(**RECALL COLORS** under **Display**); No query)

- Equivalent SCPI Command

```
:DISPlay:CMAP:LOAD
```

RECC

RECD \square *<string>*

Loads the instrument states or data. (`file name` under `Recall`); No query)

| Parameter | Description |
|-----------------------|--|
| <i><string></i> | File name, Up to 10 characters including the extension |

■ Equivalent SCPI Command

:MMEMory:LOAD:STATe \square *<string (file_name)>* [, *<string (msus)>*] (State)

:MMEMory:LOAD:TRACe \square SEL, *<string (file_name)>* [, *<string (msus)>*] (Data)

| Parameter | Description |
|-----------------------------------|--|
| <i><string (file_name)></i> | File name, Up to 10 characters including the extension |
| <i><string (msus)></i> | “DISK” for the flexible disk drive “MEMORY” for the RAM disk memory |

■ Example

```
OUTPUT 717;"RECD ""TEST_S""
```

```
OUTPUT 717;":MMEM:LOAD:STAT ""TEST_S""
```

REFD

Completes with the reflection part of the full 2-port or one-path 2-port calibration. (Network analyzer only) (`REFLECT'N DONE` under `Cal`); No query)

■ Equivalent SCPI Command

:SENSe:CORRection:COLLect:SAVE5

REFL

Begins the reflection part of the full 2-port or one-path 2-port calibration. (Network analyzer only) (`REFLECT'N` under `Cal`); No query)

■ Equivalent SCPI Command

:SENSe:CORRection:COLLect[:ACQuire] \square REFL2

REFP□<numeric>

Sets the position of the reference line on the graticule of a Cartesian display. (Network and impedance analyzer only) (**REFERENCE POSITION** under **Scale Ref**)

| Parameter | Range | Unit |
|-----------|---|------|
| <numeric> | 0 to 10 (simple command) 0 to 100 (SCPI command) | Div |

■ Equivalent SCPI Command

:DISPlay[:WINDow]:TRACe{[1]|2}:Y[:SCALe]:RPOSition□<numeric>

Where,

TRACe1 Data trace
TRACe2 Memory trace

REFV□<numeric>

Sets the value of the reference line, moving the measurement trace correspondingly.

(**REFERENCE VALUE** under **Scale Ref**)

| Parameter | Range | Format |
|-----------|---|--|
| <numeric> | -500 to 500 (Network analyzer) | (Log mag format) |
| <numeric> | -500 M to 500 M (Impedance analyzer) | (Linear magnitude format) |
| | -500 M to 500 M (Network analyzer) | (Phase, Expanded phase, Lin man, Real, Imaginary, SWR formats) |
| | -0.5 to 0.5 (Network analyzer) | (Delay format) |
| | 1×10^{-11} to 500 (Network analyzer) | (Smith chart, Admittance chart, Polar formats) |
| | 10 f to 500 M (Impedance analyzer) | (Smith chart, Admittance chart, Polar formats) |
| | -100 to 30 (Spectrum analyzer) | (dBm format) |
| | -113 to 17 (50 Ω) | (dBV format) |
| | -111.2 to 18.8 (75 Ω) (Spectrum analyzer) | |
| | 7 to 137 (50 Ω) | (dBμV format) |
| | 8.8 to 138.8 (75 Ω) (Spectrum analyzer) | |
| | 1×10^{-13} to 1 (Spectrum analyzer) | (Watt format) |
| | 2.236 μ to 7.071 (50 Ω) | (Volt format) |
| | 2.739 μ to 8.66 (75 Ω) (Spectrum analyzer) | |

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

:DISPlay[:WINDow]:TRACe{[1]|2}:Y[:SCALe]:RLEVel□<numeric>

Where,

TRACe1 Data trace
TRACe2 Memory trace

REFX□<numeric>

Sets the value of the x-axis reference line in complex plane format, moving the measurement trace correspondingly, when the measurement format is set to the complex plane.

(REFERENCE X VALUE under (Scale Ref); Impedance analyzer only.)

| Parameter | Range | Unit |
|-----------|---|------|
| <numeric> | -5.0×10 ⁸ to 5.0×10 ⁸ | U |

■ Query Response

<numeric><new line><END>

■ Equivalent SCPI Command

:DISPlay[:WINDow]:TRACe{1|2}:X[:SCALE]:RLEVel□<numeric>

Where,

TRACe1 Data trace
TRACe2 Memory trace

REFY□<numeric>

Sets the value of the y-axis reference line in complex plane format, moving the measurement trace correspondingly, when the measurement format is set to the complex plane.

(REFERENCE Y VALUE under (Scale Ref))

| Parameter | Range | Unit |
|-----------|---|------|
| <numeric> | -5.0×10 ⁸ to 5.0×10 ⁸ | U |

■ Query Response

<numeric><new line><END>

■ Equivalent SCPI Command

:DISPlay[:WINDow]:TRACe{1|2}:Y[:SCALE]:RLEVel□<numeric>

Where,

TRACe1 Data trace
TRACe2 Memory trace

REPTSMP □ {OFF|ON|0|1}

Sets the normal or repetitive sampling mode for zero span. (Spectrum analyzer only)
 (SAMPLING NORMAL repet under (Sweep))

| Parameter | Description |
|-----------|----------------------------------|
| OFF or 0 | Normal sampling |
| ON or 1 | Repetitive sampling ¹ |

¹ Can be set only when the trigger source is the external or video trigger; the frequency span is 0 Hz, and the sweep type is linear frequency.

- Query response

{*numeric*} <new line><END>

- Equivalent SCPI Command

:SENSe:DETEctor:CONTinuous □ {OFF|ON|0|1}

RESAVD □ <string>

Updates a file that is already saved. (RE-SAVE FILE under (Save); No query)

| Parameter | Description |
|-----------|---|
| <string> | File name up to 10 characters including the extension |

- Equivalent SCPI Command

STATE :MMEMory:DELeTe □ <string (file_name)> [, <string (msus)>]
 :MMEMory:STORe:STATe □ <string (file_name)> [, <string (msus)>]
 TRACE :MMEMory:DELeTe □ <string (file_name)> [, <string (msus)>]
 :MMEMory:STORe:TRACe □ SEL, <string (file_name)> [, <string (msus)>]

| Parameter | Description |
|-----------------|--|
| <string (msus)> | "DISK" for the flexible disk drive "MEMORY" for the RAM disk memory |

- Example

```
OUTPUT 717;"RESAVD ""TEST_S""
OUTPUT 717;":MMEM:DEL ""TEST_S""
OUTPUT 717;":MMEM:STOR:STAT ""TEST""
```

RESC

Eliminates the need to restart a calibration sequence that was interrupted to access some other menu. (Network and impedance analyzer only) (RESUME CAL SEQUENCE under (Cal); No query)

- Equivalent SCPI Command

:SENSe:CORRection:COLLect:RESume

RESCOM

Resume the last measured compensation sequence. (**RESUME COMP SEQ** under **Cal**); No query; Impedance analyzer only.)

■ Equivalent SCPI Command

`:SENSe:CORRection2:COLLect:RESume`

RESD

Turns off the tabular listing and returns the measurement display to the screen. (**RESTORE DISPLAY** under **Copy**); No query)

■ Equivalent SCPI Command

`:DISPLay[:WINDow]:TEXT{1-17}:STATe␣OFF`

| Parameter | Description |
|-----------|-------------------------------|
| 1 | List Value |
| 2 | Operation Parameter |
| 3 | Cal Class |
| 4 to 11 | Cal Standard No. 1 to No. 8 |
| 12 | Start and Stop |
| 13 | Center and Span (List) |
| 14 | Upper and Lower |
| 15 | Middle and Delta (Limit Test) |
| 16 | Marker List |
| 17 | Title |

RESPDONE

Completes the response calibration. Computes and stores the error coefficients. (Network analyzer only) (**DONE: RESPONSE** under **Cal**); No query)

■ Equivalent SCPI Command

`:SENSe:CORRection:COLLect:SAVE2`

REST

Aborts the sweep in progress and then restarts the measurement. (**MEASURE RESTART** under **Trigger**); No query)

■ Equivalent SCPI Command

`:INITiate[:IMMEDIATE]:AGain:ALL`

REVI

Measures S_{12} isolation for the full 2-port calibration. (Network analyzer only) (**REV ISOL'N ISOL'N STD** under **Cal**); No query)

■ Equivalent SCPI Command

`:SENSe:CORRection:COLLect[:ACquire]␣REVI`

REVM

Measures S_{22} load match for the full 2-port calibration. (Network analyzer only)
 (REV. MATCH THRU under **Cal**); No query

- Equivalent SCPI Command

```
:SENSe:CORRection:COLLect[:ACQuire]□REVM
```

REVT

Measures S_{12} frequency response for the full 2-port calibration. (Network analyzer only)
 (REV. TRANS. THRU under **Cal**); No query

- Equivalent SCPI Command

```
:SENSe:CORRection:COLLect[:ACQuire]□REVT
```

RFO □ {OFF|ON|0|1}

Sets the signal output on the RF OUT port ON or OFF. (RF OUT ON off under **Source**)

| Parameter | Description |
|-----------|-----------------|
| OFF or 0 | RF OUT port OFF |
| ON or 1 | RF OUT port ON |

- Query Response

```
{0|1} <new line><END>
```

- Equivalent SCPI Command

```
:SOURce:POWer:STATe□{OFF|ON|0|1}
```

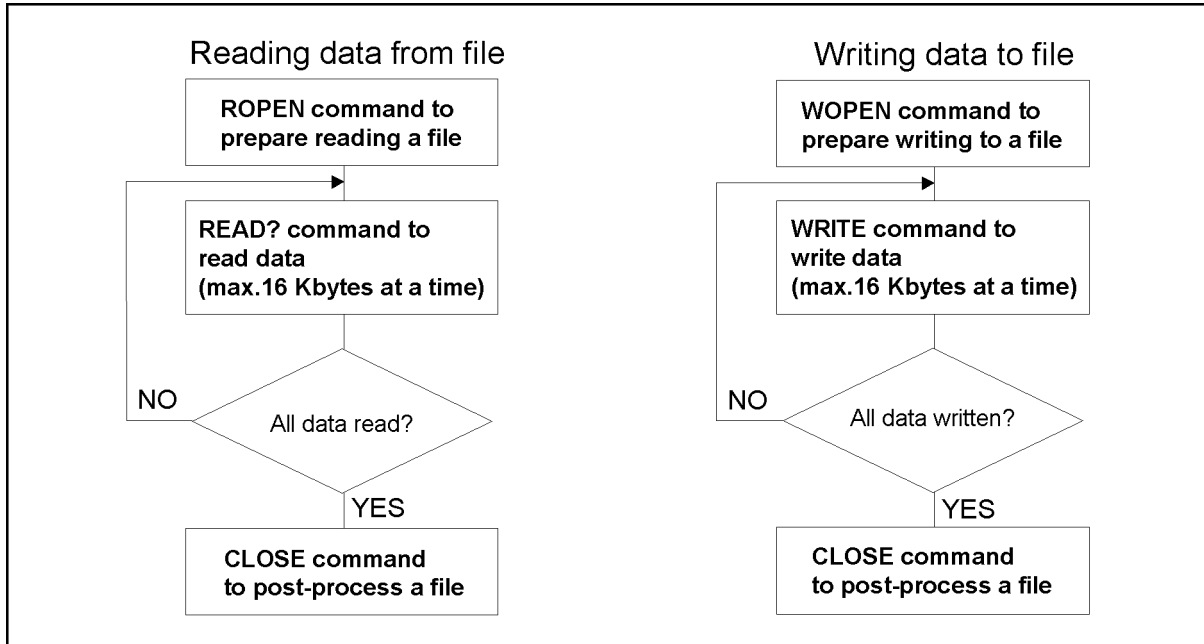
RFO □ {OFF|ON|0|1}

ROPEN □ <string>

Makes a specified file read-enabled. If the file does not exist, an error occurs.

Generally, this command is used in combination with the READ? command and the CLOSE command, as shown in Figure 2-2. (No query)

| Parameter | Description |
|-----------|--|
| <string> | File name of up to 12 characters including its extension (for the LIF format, up to 10 characters) |



rwflow

Figure 2-2. Procedure of executing commands to read/write data

RPLENV?

Returns the maximum height between the negative peak and intersection of an imaginary slope line between the adjacent positive peaks. See “RPLENV?” in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only)

RPLHEI?

Returns the maximum difference between adjacent positive and negative peaks. See “RPLHEI?” in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only)

RPLLHEI?

Returns the maximum difference between the positive peak and right-hand adjacent negative peak. See “RPLLHEI?” in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only)

RPLMEA?

Returns the mean of the difference between adjacent positive and negative peaks within range. See “RPLPP?” in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only)

RPLPP?

Returns the maximum difference between the positive peak and the negative peak within range. See “RPLPP?” in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only)

RPLRHEI?

Returns the maximum difference between the positive peak and left-hand adjacent negative peak. See “RPLRHEI?” in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only)

RPLVAL?

Returns the maximum total of the differences between the negative peaks and the adjacent positive peaks on both sides and the stimulus of the corresponding negative peak. See “RPLPP?” in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only)

RSCO

Resets the color being modified to the default color. (**RESET COLOR** under **Display**); No query)

- Equivalent SCPI Command

DISPlay:CMAP:COLor{1-14}:DEFault

| Parameter | Description |
|-----------|---|
| 1 | Channel 1 data |
| 2 | Channel 1 memory and limit lines |
| 3 | Channel 2 data |
| 4 | Channel 2 memory and limit lines |
| 5 | Graticule and a portion of softkey text |
| 6 | Warning annotation |
| 7 | All the non-data text |
| 8 | Text on the BASIC screen |
| 9-14 | Pen 1-6 |

RSCO

SA

Selects the spectrum analyzer as the analyzer type. (SPECTRUM ANALYZER under Meas)

■ Query Response

{0|1} <new line><END>

| Parameter | Description |
|-----------|------------------------------------|
| 0 | Spectrum analyzer is not selected. |
| 1 | Spectrum analyzer is selected. |

■ Equivalent SCPI Command

:INSTrument:TYPE|SA

■ Example

```
OUTPUT 717;"SA"  
OUTPUT 717;"SA?"  
ENTER 717;A  
OUTPUT 717;":INST:TYPE SA"  
OUTPUT 717;":INST:TYPE?"  
ENTER 717;A$
```

SADD

Adds a new segment to a list sweep table. (ADD under Sweep); No query)

■ Equivalent SCPI Command

:SENSe:LIST:SEGMENT:ADD

SAUNIT|{DBM|DBV|DBUV|W|V}

Selects the measurement data unit of the spectrum analyzer on the active channel. (Spectrum analyzer only) (UNIT: dBm, dBV, dB μ V, WATT, VOLT under Format)

| Parameter | Description |
|-----------|-------------|
| DBM | dBm |
| DBV | dBV |
| DBUV | dB μ V |
| W | Watt |
| V | Volt |

- Query Response

```
{DBM|DBV|DBUV|W|V} <new line><^END>
```

- Equivalent SCPI Command

```
SAUNIT:DBM :CALCulate:FORMat:MLOGarithmic
:UNIT:POWer:DBM
```

```
SAUNIT:DBV :CALCulate:FORMat:MLOGarithmic
:UNIT:POWer:DBV
```

```
SAUNIT:DBUV :CALCulate:FORMat:MLOGarithmic
:UNIT:POWer:DBUV
```

```
SAUNIT:W :CALCulate:FORMat:MLINear
:UNIT:POWer:W
```

```
SAUNIT:V :CALCulate:FORMat:MLINear
:UNIT:POWer:V
```

- Example

```
OUTPUT 717;"SAUNIT DBM"
```

```
OUTPUT 717;"SAUNIT?"
```

```
ENTER 717;A$
```

```
OUTPUT 717;":CALC:FORM MLOG"
```

```
OUTPUT 717;":UNIT:POW DBM"
```

```
OUTPUT 717;":CALC:FORM?"
```

```
ENTER 717;A$
```

```
OUTPUT 717;":UNIT:POW?"
```

```
ENTER 717;B$
```

SAV1

Completes the S_{11} or S_{22} 1-port calibration. The error coefficients are computed and stored. (Network analyzer only) (DONE: 1-PORT CAL under Cal); No query)

- Equivalent SCPI Command

```
:SENSe:CORRection:COLLect:SAVE4
```

SAV2

Completes the full or one-path 2-port calibration. The error coefficients are computed and stored. (Network analyzer only) (DONE: 2-PORT CAL under Cal); No query)

- Equivalent SCPI Command

```
:SENSe:CORRection:COLLect:SAVE8
```

SAVC

Redraws a trace using the current error coefficient array data. (Network and impedance analyzer only) (No query)

- Equivalent SCPI Command

```
:SENSe:CORRection:COLLect:SAVE9
```

SAVC

SAVCAL \square {OFF|ON|0|1}

Selects whether or not to save the calibration coefficients arrays. (CAL ON off under **Save**); No query for the SCPI command)

| Parameter | Description |
|-----------|--|
| OFF or 0 | Does not save the calibration coefficients arrays. |
| ON or 1 | Saves the calibration coefficients arrays. |

■ Query Response

{0|1} <new line><END>

■ Equivalent SCPI Command

SAVCAL \square {OFF|0} :MMEMory:STORe:ITEM:TRACe{1|2}:DELete \square CC0

SAVCAL \square {ON|1} :MMEMory:STORe:ITEM:TRACe{1|2}:SELect \square CC0

SAVCOM

Calculates the fixture compensation coefficients and store it. (DONE: COMPEN under **Cal**); No query; Impedance analyzer only)

■ Equivalent SCPI Command

:SENSe:CORRection2:COLLect:SAVE

SAVDASC \square <string>

Specifies saving the internal data arrays as an ASCII file. (SAVE ASCII under **Save**); No query)

| Parameter | Description |
|-----------|-------------------------------|
| <string> | File name, up to 8 characters |

■ Equivalent SCPI Command

:MMEMory:STORe:DINTerchange:TRACe \square SEL, <string (file_name)> [, <string (msus)>]

| Parameter | Description |
|-----------------|--|
| <string (msus)> | "DISK" for the flexible disk drive "MEMORY" for the RAM disk memory |

■ Example

```
OUTPUT 717;"SAVDASC ""DATA1""
```

```
OUTPUT 717;":MMEM:STOR:DINT:TRAC SEL, ""DATA1""
```


SAVDAT \square {OFF|ON|0|1}

Selects whether or not to save the data arrays. (DATA ON off under **Save**); No query for the SCPI command)

| Parameter | Description |
|-----------|--------------------------------|
| OFF or 0 | Does not save the data arrays. |
| ON or 1 | Saves the data arrays. |

- Query Response

{0|1} <new line><END>

- Equivalent SCPI Command

SAVDAT \square {OFF|0} :MMEMory:STORe:ITEM:TRACe{1|2}:DELete \square DATA
 SAVDAT \square {ON|1} :MMEMory:STORe:ITEM:TRACe{1|2}:SELect \square DATA

- Example

OUTPUT 717;"SAVDAT ON"

SAVDDAT \square *<string>*

Specifies saving the internal data arrays which are defined by the SAVRAW, SAVCAL, SAVDAT, SAVMEM, SAVTDAT, and SAVTMEM commands. (SAVE BINARY under **Save**); No query)

| Parameter | Description |
|-----------------------|------------------------------|
| <i><string></i> | File name up to 8 characters |

- Equivalent SCPI Command

:MMEMory:STORe:TRACe \square SEL, *<string (file_name)>* [, *<string (msus)>*]

| Parameter | Description |
|------------------------------|--|
| <i><string (msus)></i> | "DISK" for the flexible disk drive "MEMORY" for the RAM disk memory |

- Example

OUTPUT 717;"SAVDDAT ""DATA1"""

OUTPUT 717;":MMEM:STOR:TRAC SEL, ""DATA1"""

SAVDSTA \square *<string>*

Specifies saving only the instrument states and the calibration coefficients. (STATE under **Save**); No query)

| Parameter | Description |
|-----------------------|------------------------------|
| <i><string></i> | File name up to 8 characters |

SAVDSTA \square <string>

■ Equivalent SCPI Command

:MMEMory:STORe:STATe \square <string (file_name)> [, <string (msus)>]

| Parameter | Description |
|-----------------|--|
| <string (msus)> | "DISK" for the flexible disk drive "MEMORY" for the RAM disk memory |

■ Example

```
OUTPUT 717;"SAVDSTA ""STA1""
```

```
OUTPUT 717;":MMEM:STOR:STAT ""STA1""
```

SAVDSTAC \square <string>

Specifies saving the instrument states the calibration coefficients and measurement data which are compatible with 4196A. (4396A STATE under **(Save)**; No query)

Note

The following settings are not saved:



- dpi
- Printer Orientation
- Form feed
- Top Margin
- Left Margin
- Print Softkey

| Parameter | Description |
|-----------|------------------------------|
| <string> | File name up to 8 characters |

■ Equivalent SCPI Command

None

■ Example

```
OUTPUT 717;"SAVDSTA ""STA1""
```

SAVDTIF \square <string>

Saves the displayed screen in the TIFF format. (GRAPHICS under **(Save)**; No Query)

| Parameter | Description |
|-----------|---------------------------------------|
| <string> | File name having maximum 8 characters |

■ Equivalent SCPI Command

:MMEMory:STORe:DINTerchange:TIFF \square <string (file name)>

SAVDTRC{OFF|ON|0|1}

Sets whether or not to save the trace arrays. (DATA TRACE ON off under **Save**); No query for the SCPI command)

| Parameter | Description |
|-----------|---------------------------------|
| OFF or 0 | Does not save the trace arrays. |
| ON or 1 | Saves the trace arrays. |

- Query Response

{0|1} <new line><END>

- Equivalent SCPI Command

SAVDTRC{OFF|0} :MMEMory:STORe:ITEM:TRACe{1|2}:DELeteLDTR
 SAVDTRC{ON|1} :MMEMory:STORe:ITEM:TRACe{1|2}:SELEctLDTR

SAVEUSEK

Stores the user-modified or user-defined calibration kit into memory. (Network and impedance analyzer only) (SAVE USER KIT under **Cal**); No query)

- Equivalent SCPI Command

:SENSe:CORRection:CKIT:MODify:SAVE

SAVIMP

Calculates the error-correction coefficients from the calibration data and stores the coefficients. (DONE:CAL under **CAL**); No query; Impedance analyzer only)

- Equivalent SCPI Command

:SENSe:CORRection1:COLLect:SAVE

SAVMEM{OFF|ON|0|1}

Specifies whether or not to save the memory arrays. (MEM ON off under **Save**); No query for the SCPI command)

| Parameter | Description |
|-----------|----------------------------------|
| OFF or 0 | Does not save the memory arrays. |
| ON or 1 | Saves the memory arrays. |

- Query Response

{0|1} <new line><END>

- Equivalent SCPI Command

SAVMEM{OFF|0} :MMEMory:STORe:ITEM:TRACe{1|2}:DELeteLMEM
 SAVMEM{ON|1} :MMEMory:STORe:ITEM:TRACe{1|2}:SELEctLMEM

SAVMEM \square {OFF|ON|0|1}

SAVMTRC \square {OFF|ON|0|1}

Specifies whether or not to save the memory trace arrays. (MEM TRACE ON off under **Save**); No query for the SCPI command)

| Parameter | Description |
|-----------|--|
| OFF or 0 | Does not save the memory trace arrays. |
| ON or 1 | Saves the memory trace arrays. |

■ Query Response

{0|1} <new line><END>

■ Equivalent SCPI Command

SAVMTRC \square {OFF|0} :MMEMory:STORe:ITEM:TRACe{1|2}:DELete \square MTR
SAVMTRC \square {ON|1} :MMEMory:STORe:ITEM:TRACe{1|2}:SELect \square MTR

SAVRAW \square {OFF|ON|0|1}

Specifies whether or not to save the raw data arrays. (RAW ON off under **Save**); No query for the SCPI command)

| Parameter | Description |
|-----------|------------------------------------|
| OFF or 0 | Does not save the raw data arrays. |
| ON or 1 | Saves the raw data arrays. |

■ Query Response

{0|1} <new line><END>

■ Equivalent SCPI Command

SAVRAW \square {OFF|0} :MMEMory:STORe:ITEM:TRACe{1|2}:DELete \square RAW
SAVRAW \square {ON|1} :MMEMory:STORe:ITEM:TRACe{1|2}:SELect \square RAW

SAVUCOMK

Stores the user-modified compensation kit into memory. (SAVE COMPEN KIT under **Cal**); No query; Impedance analyzer only)

■ Equivalent SCPI Command

:SENSe:CORRection2:CKIT:MODifySAVE

SAVUFIXT

Saves the settings of user difined fixture. (SAVE USER FXTR KIT under **Meas** FIXTURE \square); No query; Impedance analyzer only)

■ Equivalent SCPI Command

:SYSTem:FIXTure:SAVE

SCAC {OFF|ON|0|1}

Couples or uncouples the “DATA” and “MEMORY” traces to be scaled. (D&M SCALE [] under Scale Ref);

| Parameter | Description |
|-----------|---|
| OFF or 0 | Uncouples the “DATA” and “MEMORY” traces. |
| ON or 1 | Couples the “DATA” and “MEMORY” traces. |

■ Query Response

{0|1} <new line><^END>

■ Equivalent SCPI Command

:DISPlay[:WINDow]:TRACe{[1]|2}:Y[:SCALE]:COUPlE{OFF|ON|0|1}
 (TRACe[1] for the data trace; TRACe2 for the memory trace.)

SCAF {DATA|MEMO}

Selects one of the “DATA” or “MEMORY” traces to be scaled. (SCALE FOR [] under Scale Ref);
 No equivalent SCPI command)

■ Query Response

{DATA|MEMO} <new line><^END>

SCAL <numeric>

Sets the response value scale per graticule trace. (SCALE/DIV under Scale Ref)

| Parameter | Range | Format |
|-----------|---|---|
| <numeric> | 0.001 to 500 (Network analyzer) | (Log magnitude format) |
| | 1p to 500 (Network analyzer) | (Phase format) |
| | 1x10 ⁻¹⁴ to 10 (Network analyzer) | (Delay format) |
| | 1x10 ⁻¹¹ to 10000 (Network analyzer) | (Smith chart, Admittance chart, Polar, Lin mag, Real, Imaginary, SWR, and Expanded phase formats) |
| | 1f to 100M (Network analyzer) | (Lin mag, Real, Imaginary, SWR, and Expanded phase formats) |
| | 0.1 to 20 (Network analyzer) | (dBm, dBμV, dBV formats) |
| | 1f to 0.1 (Network analyzer) | (Watt format) |
| | 1n to 1 (Network analyzer) | (Volt format) |
| | 1f to 100M (Impedance analyzer) | (Log mag, Lin mag, and complex plane formats) |
| | 10f to 500M (Impedance analyzer) | (Smith chart, Admittance chart, and Polar formats) |

SCAL \square *<numeric>*

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

:DISPlay[:WINDow]:TRACe{[1]|2}:Y[:SCALe]:PDIVision \square *<numeric>*

(TRACe[1] for the data trace; TRACe2 for the memory trace.)

SCRN \square {OFF|ON|0|1}

Controls whether the LCD display is visible or not. (No equivalent SCPI command)

| Parameter | Description |
|-----------|--|
| OFF or 0 | Invisible (only softkey labels are displayed.) |
| ON or 1 | Visible |

■ Query Response

{0|1} <new line><^END>

SDEL

Deletes a segment from a list sweep table. (DELETE under Sweep); No query)

■ Equivalent SCPI Command

:SENSe:LIST:SEGment:DELeTe

SDON

Saves the modified segment of a list sweep table. (SEGMENT DONE under Sweep); No query)

■ Equivalent SCPI Command

:SENSe:LIST:SEGment:SAVE

■ Example

```
OUTPUT 717;"SDON"
```

```
OUTPUT 717;" :SENS:LIST:SEG:SAVE"
```

SEAL

Searches the trace for the next occurrence of the target value to the left of the marker. (Network and impedance analyzer only) (SEARCH LEFT under Search); No query)

■ Equivalent SCPI Command

:CALCulate:EVALuate:Y:XPOSITION:LTARget

SEAM□{**PEAK**|**MAX**|**MIN**|**TARG**|**PKSA**|**PKSR**|**PKSL**|**OFF**}

Selects the marker search function. (SEARCH: PEAK , MAX , MIN , TARGET , SEARCH: PEAKS ALL , PEAKS RIGHT , PEAKS LEFT under (Search); No query for the SCPI command)

| Parameter | Description |
|-----------|---|
| PEAK | Peak search |
| MAX | Maximum search |
| MIN | Minimum search |
| TARG | Target search (Network and impedance analyzer only) |
| PKSA | Peak all |
| PKSR | Peak right all |
| PKSL | Peak left all |
| OFF | Marker search function OFF |

■ Query Response

{PEAK|MAX|MIN|TARG|PKSA|PKSR|PKSL|OFF} <new line><^END>

■ Equivalent SCPI Command

```
SEAM□PEAK      :CALCulate:EVALuate:Y:XPOSITION:PEAK
SEAM□MAX       :CALCulate:EVALuate:Y:XPOSITION:MAXimum
SEAM□MIN       :CALCulate:EVALuate:Y:XPOSITION:MINimum
SEAM□TARG      :CALCulate:EVALuate:Y:XPOSITION:TARGe□<numeric>
SEAM□PKSA      :CALCulate:EVALuate:Y:XPOSITION:PALL
SEAM□PKSR      :CALCulate:EVALuate:Y:XPOSITION:PRIGHt
SEAM□PKSL      :CALCulate:EVALuate:Y:XPOSITION:PLEFt
SEAM□OFF       None
```

| Parameter | Range | Unit |
|---|---|------|
| <numeric> for :CALC:EVAL:Y:XPOS:TARG | -5×10 ⁵ to 5×10 ⁵ | |

■ Example

```
OUTPUT 717;"SEAM PEAK"
OUTPUT 717;"SEAM?"
ENTER 717;A$
OUTPUT 717;":CALC:EVAL:Y:XPOS:PEAK"
```

SEANPK

Moves the marker to the next peak. (NEXT PEAK under (Search); No query)

■ Equivalent SCPI Command

```
:CALCulate:EVALuate:Y:XPOSITION:NPEak
```

SEANPK

SEANPKL

Moves the marker to the peak to the left of the present marker position. (NEXT PEAK LEFT under `(Search)`; No query)

■ Equivalent SCPI Command

:CALCulate:EVALuate:Y:XPOsition:LPEak

SEANPKR

Moves the marker to the peak to the right of the present marker position. (NEXT PEAK RIGHT under `(Search)`; No query)

■ Equivalent SCPI Command

:CALCulate:EVALuate:Y:XPOsition:RPEak

SEAR

Searches the trace for the next occurrence of the target value to the right of the marker. (Network and impedance analyzer only) (SEARCH RIGHT under `(Search)`; No Query)

■ Equivalent SCPI Command

:CALCulate:EVALuate:Y:XPOsition:RTARget

SEARSTR

Sets the partial search range to the range between the marker and the Δ marker. (MKRA—SEARCH RNG under `(Search)`; No query)

■ Equivalent SCPI Command

:CALCulate:EVALuate:BAND:SPAN Δ DMARker

SEARSTRL

Sets the left (lower) border of the partial search range at the current position of the marker. (MKR—LEFT RNG under `(Search)`; No query)

■ Equivalent SCPI Command

:CALCulate:EVALuate:BAND:STARt Δ MARker

■ Example

```
OUTPUT 717;"SEARSTRL"
```

```
OUTPUT 717;":CALC:EVAL:BAND:STAR MARK"
```


SEARSTRR

Sets the right (higher) border of the partial search range at the current position of the marker. (MKR—RIGHT RNG under **Search**); No query

- Equivalent SCPI Command

:CALCulate:EVALuate:BAND:STOP□MARKer

- Example

OUTPUT 717;"SEARSTRR"

OUTPUT 717;":CALC:EVAL:BAND:STOP MARK"

SEATARG□<numeric>[DB|DEG|S|OHM]

Makes the target value to the active function to enter a value and moves the marker to a specified target point on the trace. (Network and impedance analyzer only) (**TARGET** under **Search**)

| Parameter | Range | Unit |
|-----------|---|------|
| <numeric> | -5×10 ⁵ to 5×10 ⁵ | |

- Query Response

{numeric} <new line><^END>

- Equivalent SCPI Command

:CALCulate:EVALuate:Y:XPOSITION:TARGGet□<numeric>

- Example

OUTPUT 717;"SEATARG 0"

SEDI□<numeric>

Determines the segment of the list sweep table to be modified. (**EDIT** under **Sweep**); No query for the SCPI command)

| Parameter | Description |
|-----------|-------------|
| <numeric> | 1 to 15 |

- Query Response

{numeric} <new line><^END>

- Equivalent SCPI Command

:SENSe:LIST:SEGMENT:EDIT

SEDI□<numeric>

SETCDATE□<numeric (year)>,<numeric (month)>,<numeric (day)>

Sets the date of the internal clock. (DATE MM/DD/YY under (System))

| Parameter | Description |
|-------------------|--------------|
| <numeric (year)> | 1900 to 2099 |
| <numeric (month)> | 1 to 12 |
| <numeric (day)> | 1 to 31 |

■ Query Response

{numeric (year)} {numeric (month)} {numeric (day)} <new line><END>

■ Equivalent SCPI Command

:SYSTem:DATE□<numeric (year)>,<numeric (month)>,<numeric (day)>

■ Example

OUTPUT 717;"SETCDATE 1993,1,1"

SETCTIME□<numeric (hour)>,<numeric (minute)>,<numeric (second)>

Sets the time of the internal clock. (SETCTIME under (System))

| Parameter | Description |
|--------------------|-------------|
| <numeric (hour)> | 0 to 23 |
| <numeric (minute)> | 0 to 59 |
| <numeric (second)> | 0 to 59 |

■ Query Response

{numeric (hour)} {numeric (minute)} {numeric (second)} <new line><END>

■ Equivalent SCPI Command

:SYSTem:TIME□<numeric (hour)>,<numeric (minute)>,<numeric (second)>

■ Example

OUTPUT 717;"SETCTIME 10,30,0"

SETZ□<numeric> [OHM|KOHM|MAOHM]

Sets the characteristic impedance of the coaxial cable offset. (Network analyzer only) (SET Z0 under (Cal))

| Parameter | Range | Unit |
|-----------|--------------|------|
| <numeric> | 0.001 to 5 M | Ω |

- Query Response

{*numeric*} <new line><^END>

- Equivalent SCPI Command

:SENSe:CORRection:CIMPedance<numeric>

SGTRK{OFF|ON|0|1}

Sets the signal tracking function ON or OFF. (Spectrum analyzer only) (SGNL TRACK ON off under [Search](#))

| Parameter | Description |
|-----------|---------------------|
| OFF or 0 | Signal tracking OFF |
| ON or 1 | Signal tracking ON |

- Query Response

{0|1} <new line><^END>

- Equivalent SCPI Command

:SENSe:TRACk:SIGNa1:MARKer{OFF|ON|0|1}

SIMFCHAR

Simulates frequency response of the equivalent circuit. (SIMULTE F-CHAR under [Display](#)); No query; Impedance analyzer only)

- Equivalent SCPI Command

CALCulate:EVALuate:EPARameters:SIMulation

SIMFCHAR

SING

Makes one sweep of the data and returns to the hold mode. (Instrument BASIC EXECUTE executable; **SINGLE** under **Trigger**); No query;)

When you execute this command by EXECUTE command of the instrument BASIC, the analyzer sweeps once and then back the control to the analyzer. The program waits the completion of sweep. You can use this method instead of detecting the sweep end by monitoring the status register to synchronize the program with the analyzer.

■ Equivalent SCPI Command

```
:INITiate:CONTinuous{OFF|0}  
:ABORt  
:SENSe:SWEep:COUNt1  
:INITiate[:IMMediate]
```

■ Example

```
OUTPUT 717;"SING"  
  
OUTPUT 717;":INIT:CONT OFF"  
OUTPUT 717;":SENS:SWE:COUN 1"  
OUTPUT 717;":INIT"  
  
EXECUTE "SING"
```

SLOP{OFF|ON|0|1}

Sets the power slope function ON or OFF. With the slope ON, the output power increases with frequency (starting at the selected power level). (Network analyzer only) (**SLOPE ON off** under **Source**)

| Parameter | Description |
|-----------|--------------------------|
| OFF or 0 | Power slope function OFF |
| ON or 1 | Power slope function ON |

■ Query Response

```
{0|1} <new line><END>
```

■ Equivalent SCPI Command

```
:SOURce:POWer[:LEVel][:IMMediate][:AMPLitude]:SLOPe:STATe{OFF|ON|0|1}
```

SLOPE□<numeric>

Compensates for the power loss versus the frequency sweep, by sloping the output power upwards proportionally to the frequency. (Network analyzer only) (SLOPE under (Source))

| Parameter | Range | Unit |
|-----------|--------|--------|
| <numeric> | 0 to 2 | dB/GHz |

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

:SOURce:POWer[:LEVel][:IMMediate][:AMPLitude]:SLOPe□<numeric>

SMKR{1-7}□{OFF|ON|0|1}

Displays the specified sub-marker at the point of the marker (ON), or erases the sub-marker (OFF). (SUB MKR {1-7} under (Marker))

| Parameter | Description |
|-----------|----------------|
| OFF or 0 | Sub-marker ON |
| ON or 1 | Sub-marker OFF |

■ Query Response

{0|1} <new line><^END>

■ Equivalent SCPI Command

```
SMKR1□{OFF|ON|0|1} :DISPlay[:WINDow]:TRACe{[1]|2}:MARKer2:STATe□{OFF|ON|0|1}
SMKR2□{OFF|ON|0|1} :DISPlay[:WINDow]:TRACe{[1]|2}:MARKer3:STATe□{OFF|ON|0|1}
SMKR3□{OFF|ON|0|1} :DISPlay[:WINDow]:TRACe{[1]|2}:MARKer4:STATe□{OFF|ON|0|1}
SMKR4□{OFF|ON|0|1} :DISPlay[:WINDow]:TRACe{[1]|2}:MARKer5:STATe□{OFF|ON|0|1}
SMKR5□{OFF|ON|0|1} :DISPlay[:WINDow]:TRACe{[1]|2}:MARKer6:STATe□{OFF|ON|0|1}
SMKR6□{OFF|ON|0|1} :DISPlay[:WINDow]:TRACe{[1]|2}:MARKer7:STATe□{OFF|ON|0|1}
SMKR7□{OFF|ON|0|1} :DISPlay[:WINDow]:TRACe{[1]|2}:MARKer8:STATe□{OFF|ON|0|1}
```

(TRACe [1] for the data trace; TRACe2 for the memory trace.)

SMKR{1-7} □ {OFF|ON|0|1}

SMKRAUV{1-7}?

Outputs the auxiliary amplitude value of the measurement value at the sub-marker position. See “Marker Readout” in Appendix H for the auxiliary amplitude value of each display format. (SUB MKR {1-7} under **Marker**; Query only)

■ Query Response

{*numeric*} <new line><^END>

■ Equivalent SCPI Command

```
SMKRAUV1? :CALCulate:EVALuate:Y2:VALue2?
SMKRAUV2? :CALCulate:EVALuate:Y3:VALue2?
SMKRAUV3? :CALCulate:EVALuate:Y4:VALue2?
SMKRAUV4? :CALCulate:EVALuate:Y5:VALue2?
SMKRAUV5? :CALCulate:EVALuate:Y6:VALue2?
SMKRAUV6? :CALCulate:EVALuate:Y7:VALue2?
SMKRAUV7? :CALCulate:EVALuate:Y8:VALue2?
```

■ Example

```
OUTPUT 717;"SMKRAUV1?"
ENTER 717;A
```

SMKRP{1-7} □ <*numeric*>

Moves the sub-marker to the specified data point number.

| Parameter | Description |
|--------------------|---|
| < <i>numeric</i> > | 1 to “number of points” (If < <i>numeric</i> > is 0 or less than 0, it is set to 1. If < <i>numeric</i> > is greater than “number of points,” it is set to “number of points.”) |

■ Query Response

{*numeric*} <new line><^END>

■ Equivalent SCPI Command

```
SMKRP1□<numeric> :CALCulate:EVALuate:Y2:XPOSition:POINt□<numeric>
SMKRP2□<numeric> :CALCulate:EVALuate:Y3:XPOSition:POINt□<numeric>
SMKRP3□<numeric> :CALCulate:EVALuate:Y4:XPOSition:POINt□<numeric>
SMKRP4□<numeric> :CALCulate:EVALuate:Y5:XPOSition:POINt□<numeric>
SMKRP5□<numeric> :CALCulate:EVALuate:Y6:XPOSition:POINt□<numeric>
SMKRP6□<numeric> :CALCulate:EVALuate:Y7:XPOSition:POINt□<numeric>
SMKRP7□<numeric> :CALCulate:EVALuate:Y8:XPOSition:POINt□<numeric>
```

■ Example

```
OUTPUT 717;"SMKRP1 1"
OUTPUT 717;"SMKRP1?"
ENTER 717;A
```

SMKRPRM{1-7} □ <numeric> [HZ|KHZ|MAHZ|GHZ|DBM]

Moves the sub-marker to the specified sweep parameter value. (SUB MKR {1-7} under (Marker))

| Parameter | Range | Unit |
|-----------|---------------------------|-------------------------------|
| <numeric> | start value to stop value | Hz (frequency) dBm (power) |

- Query Response

{numeric} <new line><^END>

- Equivalent SCPI Command

```
SMKRPRM1 □ <numeric>      :CALCulate:EVALuate:Y2:XPOSITION □ <numeric>
SMKRPRM2 □ <numeric>      :CALCulate:EVALuate:Y3:XPOSITION □ <numeric>
SMKRPRM3 □ <numeric>      :CALCulate:EVALuate:Y4:XPOSITION □ <numeric>
SMKRPRM4 □ <numeric>      :CALCulate:EVALuate:Y5:XPOSITION □ <numeric>
SMKRPRM5 □ <numeric>      :CALCulate:EVALuate:Y6:XPOSITION □ <numeric>
SMKRPRM6 □ <numeric>      :CALCulate:EVALuate:Y7:XPOSITION □ <numeric>
SMKRPRM7 □ <numeric>      :CALCulate:EVALuate:Y8:XPOSITION □ <numeric>
```

SMKRVAL{1-7}?

Outputs the primary part of the measurement value at the sub-marker position.

(SUB MKR {1-7} under (Marker); Query only)

- Query Response

{numeric} <new line><^END>

- Equivalent SCPI Command

```
SMKRVAL1?      :CALCulate:EVALuate:Y2:VALue1?
SMKRVAL2?      :CALCulate:EVALuate:Y3:VALue1?
SMKRVAL3?      :CALCulate:EVALuate:Y4:VALue1?
SMKRVAL4?      :CALCulate:EVALuate:Y5:VALue1?
SMKRVAL5?      :CALCulate:EVALuate:Y6:VALue1?
SMKRVAL6?      :CALCulate:EVALuate:Y7:VALue1?
SMKRVAL7?      :CALCulate:EVALuate:Y8:VALue1?
```

SMKRVAL{1-7}?

SPAN \square <numeric> [HZ|KHZ|MAHZ|GHZ|DBM]

Sets the frequency span of a segment about a specified center frequency, or sets the frequency span of the list sweep table. (**SPAN**) or **SPAN** under **Sweep**)

| Parameter | Range | Unit |
|-----------|--|-------------------------------|
| <numeric> | 0 to 1.8199×10^9 (= 1.8199 G) 0 to 20 or 30 ¹ | Hz (frequency) dBm (power) |

1 The maximum range depends on the center value.

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

Span :SENSe:FREQuency:SPAN \square <numeric> (frequency) or
:SOURce:POWer:SPAN <numeric> (power)

Span under **Sweep** :SENSe:LIST:SEGment:FREQuency:SPAN \square <numeric>
(List sweep table)

SPECFWD{M|T} \square <numeric 1> [, <numeric 2> [, ... [, <numeric 7>]

Enters the standard numbers for the forward match (THRU) or forward transmission (THRU) calibration. (Network analyzer only) (**FWD.MATCH**, **FWD.TRANS.** under **Cal**); No query)

| Parameter | Description |
|-----------|-------------|
| <numeric> | 1 to 8 |

■ Equivalent SCPI Command

SPECFWDM :SENSe:CORRection:CKIT:CLASs9:STANdard \square <numeric 1> [, <numeric 2>
[, ... [, <numeric 7>]

SPECFWDT :SENSe:CORRection:CKIT:CLASs7:STANdard \square <numeric 1> [, <numeric 2>
[, ... [, <numeric 7>]

■ Example

OUTPUT 717;"SPECFWDM 1"

OUTPUT 717;":SENS:CORR:CKIT:CLAS9:STAN 1"

SPECIMP{A|B|C} \square <numeric 1> [, <numeric 2> [, ... [, <numeric 7>]

Enters the standard numbers for the first, second, or third standard class required for an impednace calibration. (**SPECIFY CLASS** under **Cal**); No query. Impedance analyzer only.)

| Parameter | Description |
|-----------|-------------|
| <numeric> | 1 to 8 |

■ Equivalent SCPI Command

SPECREV{M|T}□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]

SPECIMPA :SENSe:CORRection:CKIT:CLASs13:STANdard□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]
SPECIMPB :SENSe:CORRection:CKIT:CLASs14:STANdard□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]
SPECIMPC :SENSe:CORRection:CKIT:CLASs15:STANdard□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]

SPECRES{I|P}□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]

Enters the standard numbers for a response and isolation, or a response calibration. (Network analyzer only) (RESPONSE & ISOL'N, RESPONSE under Cal); No query)

| Parameter | Description |
|-----------|-------------|
| <numeric> | 1 to 8 |

■ Equivalent SCPI Command

SPECRESI :SENSe:CORRection:CKIT:CLASs12:STANdard□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]
SPECRESP :SENSe:CORRection:CKIT:CLASs11:STANdard□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]

SPECREV{M|T}□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]

Enters the standard numbers for the reverse match (THRU) or reverse transmission (THRU) calibration. (Network analyzer only) (REV.MATCH, REV.TRANS. under Cal); No query)

| Parameter | Description |
|-----------|-------------|
| <numeric> | 1 to 8 |

■ Equivalent SCPI Command

SPECREVM :SENSe:CORRection:CKIT:CLASs10:STANdard□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]
SPECREVT :SENSe:CORRection:CKIT:CLASs8:STANdard□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]

SPECREV{**M|T**}□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]

SPECS11{**A|B|C**}□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]

Enters the standard numbers for the first, second, or third standard class required for an S₁₁ 1-port calibration. (Network analyzer only) (SPECIFY: S11A, S11B, S11C under **Cal**); No query)

| Parameter | Description |
|-----------|-------------|
| <numeric> | 1 to 8 |

■ Equivalent SCPI Command

SPECS11A :SENSe:CORRection:CKIT:CLASs1:STANdard□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]

SPECS11B :SENSe:CORRection:CKIT:CLASs2:STANdard□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]

SPECS11C :SENSe:CORRection:CKIT:CLASs3:STANdard□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]

SPECS22{**A|B|C**}□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]

Enters the standard numbers for the first, second, or third standard class required for an S₂₂ 1-port calibration. (Network analyzer only) (SPECIFY: S22A, S22B, S22C under **Cal**); No query)

| Parameter | Description |
|-----------|-------------|
| <numeric> | 1 to 8 |

■ Equivalent SCPI Command

SPECS22A :SENSe:CORRection:CKIT:CLASs4:STANdard□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]

SPECS22B :SENSe:CORRection:CKIT:CLASs5:STANdard□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]

SPECS22C :SENSe:CORRection:CKIT:CLASs6:STANdard□<numeric 1>[,<numeric 2>[, ... [,<numeric 7>]

STAR \square *<numeric>* [HZ|KHZ|MAHZ|GHZ|DBM]

SPLD \square {OFF|ON|0|1}

Sets the dual channel display mode. (SPLIT DISP ON off under (Display))

| Parameter | Description |
|-----------|---|
| OFF or 0 | Full-screen single graticule display |
| ON or 1 | Split display with two half-screen graticules |

■ Query Response

{0|1} <new line><END>

■ Equivalent SCPI Command

SPLD \square {OFF|0} :DISPlay[:WINDow]:FORMat \square FBACk
SPLD \square {ON|1} :DISPlay[:WINDow]:FORMat \square ULOWer

SQUI

Terminates editing a segment of the list sweep table. (SEGMENT QUIT under (Sweep); No query)

■ Equivalent SCPI Command

:SENSe:LIST:SEGment:QUIT

STAN{A-G}

Measures the calibration standard in the current standard class. (Network analyzer only)
(OPEN, SHORT, THRU, OPEN [], SHORT [], defined std {1-7} under (Ca); No query)

■ Equivalent SCPI Command

STANA :SENSe:CORRection:COLLect[:ACQuire] \square STANdard1
STANB :SENSe:CORRection:COLLect[:ACQuire] \square STANdard2
STANC :SENSe:CORRection:COLLect[:ACQuire] \square STANdard3
STAND :SENSe:CORRection:COLLect[:ACQuire] \square STANdard4
STANE :SENSe:CORRection:COLLect[:ACQuire] \square STANdard5
STANF :SENSe:CORRection:COLLect[:ACQuire] \square STANdard6
STANG :SENSe:CORRection:COLLect[:ACQuire] \square STANdard7

STAR \square *<numeric>* [HZ|KHZ|MAHZ|GHZ|DBM]

Sets the start value of a segment, or sets the start value of the list sweep table. ((Start) or SEGMENT: START under (Sweep))

| Parameter | Range | Unit |
|------------------------|---|-------------------------------|
| <i><numeric></i> | 100000 (= 100 k) to 1.82×10^9 (= 1.82 G) (Network analyzer) 0 to 1.82×10^9 (= 1.82 G) (Spectrum analyzer) -60 to 20 | Hz (frequency) dBm (power) |

STAR<numeric>[HZ|KHZ|MAHZ|GHZ|DBM]

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

START :SENSe:FREQUency:STARt<numeric> (frequency) or
:SOURce:POWer:STARt<numeric> (power)

SEGMENT: START under **Sweep** :SENSe:LIST:SEGment:FREQUency:STARt<numeric>
(List sweep table)

STDD

Terminates the standard definition. (Network and impedance analyzer only)

(STD DONE (DEFINED) under **Cal**); No query

■ Equivalent SCPI Command

:SENSe:CORRection:CKIT:STANdard:SAVE

STDT{OPEN|SHOR|LOAD|DELA|ARBI}

Defines the standard type. (Network and impedance analyzer only) (STD TYPE: OPEN, SHORT, LOAD, DELAY/THRU, ARBITRARY IMPEDANCE under **Cal**)

| Parameter | Description |
|-----------|---------------------------------------|
| OPEN | OPEN |
| SHOR | SHORT |
| LOAD | LOAD |
| DELA | Transmission line of specified length |
| ARBI | LOAD with an arbitrary impedance |

■ Query Response

{OPEN|SHOR|LOAD|DELA|ARBI} <new line><^END>

■ Equivalent SCPI Command

:SENSe:CORRection:CKIT:STANdard:TYPE{OPEN|SHORt|LOAD|DELay|AIMPedance}

STOD{DISK|MEMO}

Sets the storage device. (STOR DEV[] under **Save**); No query; No equivalent SCPI command

| Parameter | Description |
|-----------|---------------------|
| STODDISK | Flexible disk drive |
| STODMEMO | RAM disk memory |

SWET□<numeric>[S|MS]

STOP□<numeric>[HZ|KHZ|MAHZ|GHZ|DBM]

Sets the stop value frequency of a segment, or the stop value of the list table. ((STOP) or STOP under (Sweep))

| Parameter | Range | Unit |
|-----------|---|-----------------------------------|
| <numeric> | 100000 (= 100 k) to 1.82×10^9 (= 1.82 G) (Network analyzer) 0 to 1.82×10^9 (= 1.82 G) (Spectrum analyzer) -60 to 20 | Hz (frequency) dBm (power) |

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

(Stop) :SENSe:FREQuency:STOP□<numeric> (frequency) or
:SOURce:POWer:STOP□<numeric> (power)

Stop under (Sweep) :SENSe:LIST:SEGment:FREQuency:STOP□<numeric>
(List sweep table)

SVCO

Saves the modified version of the color set to the non-volatile memory. (SAVE COLORS under (Display); No query)

■ Equivalent SCPI Command

:DISPlay:CMAP:STORe

SWET□<numeric>[S|MS]

Sets the sweep time. (SWEEP TIME under (Sweep))

| Parameter | Range | Unit |
|-----------|-------------------------------------|------|
| <numeric> | (depends on the analyzer's setting) | s |

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

:SENSe:SWEep:TIME□<numeric>

SWET \sqcup *<numeric>* [**S**|**MS**]

SWETAUTO \sqcup {**OFF**|**ON**|**0**|**1**}

Sets the automatic or manual sweep time. The automatic mode gives the fastest sweep time at the analyzer's current settings of the channel. (SWEEP TIME AUTO man under **Sweep**)

| Parameter | Description |
|-----------|----------------------|
| OFF or 0 | Manual sweep time |
| ON or 1 | Automatic sweep time |

■ Query Response

{0|1} <new line><END>

■ Equivalent SCPI Command

:SENSe:SWEEP:TIME:AUTO \sqcup {OFF|ON|0|1}

SWPT \sqcup {**LINF**|**LOGF**|**LIST**|**POWE**}

Selects the sweep type. (SWEEP TYPE:LIN FREQ, LOG FREQ, LIST FREQ, POWER SWEEP under **Sweep**)

| Parameter | Description |
|-----------|---|
| LINF | Linear frequency |
| LOGF | Log frequency (Network and impedance analyzer only) |
| LIST | Frequency list |
| POWE | Power (Network and impedance analyzer only) |

■ Query Response

{LINF|LOGF|LIST|POWE} <new line><END>

■ Equivalent SCPI Command

SWPT \sqcup LINF :SENSe:FREQuency:MODE \sqcup SWEEP
:SOURce:POWer:MODE \sqcup FIXEd
:SENSe:SWEEP:SPACing \sqcup LINear

SWPT \sqcup LOGF :SENSe:FREQuency:MODE \sqcup SWEEP
:SOURce:POWer:MODE \sqcup FIXEd
:SENSe:SWEEP:SPACing \sqcup LOGarithmic

SWPT \sqcup LIST :SENSe:FREQuency:MODE \sqcup LIST
:SOURce:POWer:MODE \sqcup LIST
:SENSe:SWEEP:SPACing \sqcup LINear

SWPT \sqcup POWE :SENSe:FREQuency:MODE \sqcup FIXEd
:SOURce:POWer:MODE \sqcup SWEEP
:SENSe:SWEEP:SPACing \sqcup LINear

TARL?

Searches for the point having the parameter-specified value leftward from the right end of the range, and returns its stimulus. See “TARL?” in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only)

TARR?

Searches for the point having the parameter-specified value rightward from the left end of the range, and returns its stimulus. See “TARR?” in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only)

TERI□<numeric>[**OHM**|**KOHM**]

Specifies the (arbitrary) impedance of the standard. (Network and impedance analyzer only) (TERMINAL IMPEDANCE under **Cal**); No query)

| Parameter | Range | Unit |
|-----------|---------------------|------|
| <numeric> | 0 to 10000 (- 10 k) | Ω |

■ Equivalent SCPI Command

:SENSe:CORRection:CKIT:STANdard:TIMPedance□<numeric>

TESS?

Outputs the test set identifier. (Network analyzer only) (Query only)

■ Query Response

{0|1} <new line><^END>

| Parameter | Description |
|-----------|----------------------|
| 0 | None |
| 1 | S-parameter test set |

■ Equivalent SCPI Command

:SYSTem:COMMunicate:TSET?

THRR□<numeric>

Sets threshold ripple height for waveform analysis commands. See “THRR” in Appendix I for details, query response, and examples. (Instrument BASIC EXECUTE executable; No equivalent SCPI command; Query only)

THRR □ <numeric>

TINT □ <numeric>

Adjusts the hue of the specified display element. (TINT under (Display)); No equivalent SCPI command)

| Parameter | Range | Unit |
|-----------|----------|------|
| <numeric> | 0 to 100 | % |

■ Query Response

{numeric} <new line><^END>

TITL □ <string>

Sends the string to the title area on the display. (TITLE under (Display))

| Parameter | Description |
|-----------|---------------------|
| <string> | up to 53 characters |

■ Query Response

{string} <new line><^END>

■ Equivalent SCPI Command

:DISPlay[:WINDow]:TEXT17[:DATA] □ <string>

■ Example

```

OUTPUT 717;"TITL ""COMMENT""
OUTPUT 717;"TITL?"
ENTER 717;A$

OUTPUT 717;":DISP:TEXT17 ""COMMENT""
OUTPUT 717;":DISP:TEXT17?"
ENTER 717;A$

```

TMARG □ <numeric>

Specify the value for the top margin of printed paper. (TOP MARGIN under (Copy))

| Parameter | Range | Unit |
|-----------|--------|------|
| <numeric> | 0 to 5 | inch |

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

:HCOPY:DRIVER:TOPMarg □ <numeric>

TOPV \square *<numeric>*

Defines the top border of the display and adjusts the scale value. (TOP VALUE under **Scale Ref**); Impedance analyzer only.)

| Parameter | Range | Unit |
|------------------------|---|-------------|
| <i><numeric></i> | -1x10 ⁹ to 1x10 ⁹ | y-axis unit |

■ Query Response

<numeric> \square \square *<new line>* \square *<END>*

■ Equivalent SCPI Command

:DISPlay[:WINDow]:TRACe{1|2}:Y[:SCALe]:TOP \square *<numeric>*

TRACK \square {OFF|ON|0|1}

Sets the search tracking function ON or OFF. (SRCH TRACK ON off under **Search**)

■ Query Response

{0|1} \square *<new line>* \square *<END>*

■ Equivalent SCPI Command

TRACK \square {OFF|0} :CALCulate:EVALuate:Y:XPOSITION:TRACk \square OFF
 TRACK \square {ON|1} :CALCulate:EVALuate:Y:XPOSITION:TRACk \square
 {MAXimum|MINimum|TARGet|PEAK|PALL|PLEFt|PRIGHt}

| Parameter | Description |
|-----------|----------------|
| MAXimum | Maximum search |
| MINimum | Minimum search |
| TARGet | Target search |
| PEAK | Peak search |
| PALL | Peaks all |
| PLEFt | Peaks left |
| PRIGHt | Peaks right |

TRAD

Completes the transmission calibration of the full or one-path 2-port calibration. (Network analyzer only) (TRANS. DONE under **Cal**); No query)

■ Equivalent SCPI Command

:SENSe:CORRection:COLLect:SAVE6

■ Example

OUTPUT 717;"TRAD"

OUTPUT 717;"SENS:CORR:COLL:SAVE6"

TRAD

TRAN

Starts the transmission part of the full or one-path 2-port calibration. (Network analyzer only) (TRANSMISSION under (Cal); No query)

■ Equivalent SCPI Command

```
:SENSe:CORRection:COLLect[:ACQuire]␣TRAN2
```

TRGEVE␣{SWE|POIN}

Selects the trigger event mode. (Network and impedance analyzer only) (TRIG EVENT [] under (Trigger))

| Parameter | Description |
|-----------|-------------------------------------|
| SWE | Trigger event on sweep |
| POIN | Trigger event on point ¹ |

¹ Available only when the trigger source is the GPIB, Manual, or External trigger.

■ Query Response

```
{SWE|POIN} <new line><END>
```

■ Equivalent SCPI Command

```
:TRIGger:EVENT:TYPE␣{SWEp|POINT}
```

TRGP␣{POS|NEG}

Sets the trigger signal polarity of an external signal connected to the rear panel EXT TRIGGER input. (TRIG PLRTY pos neg under (Trigger))

| Parameter | Description |
|-----------|---|
| POS | Positive trigger (low-to-high transition) |
| NEG | Negative trigger (high-to-low transition) |

■ Query Response

```
{POS|NEG} <new line><END>
```

■ Equivalent SCPI Command

```
:TRIGger:SLOPe␣{POSitive|NEGative}
```

TRGS {INT|EXT|BUS|VID|MAN|GAT}

Selects the trigger source, which is common to both channels. (TRIGGER: [] under (Trigger))

| Parameter | Description |
|-----------|---|
| INT | Internal trigger |
| EXT | External trigger input from BNC on the rear panel |
| BUS | GPIB trigger |
| VID | Video trigger (Spectrum analyzer only) |
| MAN | Manual trigger |
| GAT | External gate trigger (Spectrum analyzer and option 1D6 only) |

- Query Response

{INT|EXT|BUS|VID|MAN|GAT} <new line><END>

- Equivalent SCPI Command

```
TRGS INT      :TRIGGER:SOURCE:INTERNAL1
               :SENSE:SWEEP:GATED{OFF|0}

TRGS EXT      :TRIGGER:SOURCE:EXTERNAL
               :SENSE:SWEEP:GATED{OFF|0}

TRGS BUS      :TRIGGER:SOURCE:BUS
               :SENSE:SWEEP:GATED{OFF|0}

TRGS VID      :TRIGGER:SOURCE:INTERNAL2
               :SENSE:SWEEP:GATED{OFF|0}

TRGS MAN      :TRIGGER:SOURCE:MANUAL
               :SENSE:SWEEP:GATED{OFF|0}

TRGS GAT      :TRIGGER:SOURCE:EXTERNAL
               :SENSE:SWEEP:GATED{ON|1}
```

USKEY

Displays the user menu of the soft keys. The user menu display returns to the ordinary measurement keys when the program ends. (No query; No equivalent SCPI command)

The USKEY command is equivalent to executing the program shown below;

```
OUTPUT @Hp4396;"KEY 47"
OUTPUT @Hp4396;"KEY 0"
OUTPUT @Hp4396;"KEY 6"
```

USKEY

VBW \square \langle numeric \rangle [HZ|KHZ|MAHZ]

Sets the bandwidth of the video bandwidth filter. (Spectrum analyzer only) (VIDEO BW under \square Bw/Avg)

| Parameter | Description |
|-----------------------------|--|
| \langle numeric \rangle | RBW, RBW/3, RBW/10, RBW/30, RBW/100, and RBW/300 |

■ Query Response

{numeric} \langle new line \rangle \langle END \rangle

■ Equivalent SCPI Command

```
:SENSe:BANDwidth[:RESolution]:AUTO  $\square$  {OFF|0}
:SENSe:BANDwidth[:RESolution]  $\square$   $\langle$ numeric $\rangle$ 
```

VBWT \square {LIN|LOG}

Selects either the linear or logarithmic video filter. (VBW TYPE [LIN] or [LOG] under \square Bw/Avg)

■ Query Response

{LIN|LOG} \langle new line \rangle \langle END \rangle

■ Equivalent SCPI Command

```
SENSe:BANDwidth:VIDeo:TYPE  $\square$  {LIN|LOG}
```

■ Examples

```
OUTPUT @Hp4396;"SENS:BAND:VID:TYPE LOG"
OUTPUT @Hp4396;"SENS:BAND:VID:TYPE?"
ENTER @Hp4396;Type$
PRINT "Current VBW setting is ";Type$
```

VELOFACT \square \langle numeric \rangle

Enters the velocity factor used by the analyzer to calculate the equivalent electrical length. (Network and impedance analyzer only) (VELOCITY FACTOR under \square Cal)

| Parameter | Range | Unit |
|-----------------------------|---------|------|
| \langle numeric \rangle | 0 to 10 | |

■ Query Response

{numeric} \langle new line \rangle \langle END \rangle

■ Equivalent SCPI Command

```
:SENSe:CORRection:RVELOCITY  $\square$   $\langle$ numeric $\rangle$ 
```

VIDLVL <numeric>

Sets the video trigger level. (Spectrum analyzer only) (VIDEO under Trigger)

| Parameter | Range | Unit |
|-----------|----------|------|
| <numeric> | 0 to 100 | % |

- Query Response

{numeric} <new line><END>

- Equivalent SCPI Command

:TRIGger:LEVel<numeric>

WIDSIN

Searches for the cutoff point on the trace within the current cutoff points. (Network and impedance analyzer only; SEARCH IN under Search); No query)

- Equivalent SCPI Command

:CALCulate:EVALuate:WIDTh:XPOSition:IN

- Example

```
OUTPUT 717;"WIDSIN"
```

```
OUTPUT 717;":CALC:EVAL:WIDT:XPOS:IN"
```

WIDSOUT

Searches for the cutoff point on the trace outside of the current cutoff points. (Network and impedance analyzer only; SEARCH OUT under Search); No query)

- Equivalent SCPI Command

:CALCulate:EVALuate:WIDTh:XPOSition:OUT

WIDTh{OFF|ON|0|1}

Sets the bandwidth search feature ON or OFF. (Network and impedance analyzer only) (WIDTHS ON off under Search)

| Parameter | Description |
|-----------|---|
| OFF or 0 | Bandwidth search feature OFF |
| ON or 1 | Bandwidth search feature ON (calculates the center stimulus value, bandwidth, Q, insertion loss, and cutoff point deviation from the center of a bandpass or band reject shape on the trace.) |

- Query Response

{0|1} <new line><END>

- Equivalent SCPI Command

:CALCulate:EVALuate:WIDTh:STATe{OFF|ON|0|1}

WIDT \square {OFF|ON|0|1}

WIDV \square \langle numeric \rangle [DB|DEG|S|OHM]

Sets an amplitude parameter that defines the start and stop points for a bandwidth search. (Network and impedance analyzer only) (WIDTH VALUE under [Search](#))

| Parameter | Range | Format |
|-----------------------------|---|----------------------|
| \langle numeric \rangle | -5×10^5 to 5×10^5 -500 to 500 (Network analyzer) | Log magnitude format |

■ Query Response

$\{$ numeric $\}$ \langle new line \rangle \langle ^END \rangle

■ Equivalent SCPI Command

:CALCulate:EVALuate:WIDTh:Y \square \langle numeric \rangle

WIDVTYPE \square {DIVS2|MULS2|DIV2|FIXed}

Select Marker Width Value Type. When you use FIXed, you must specify the bandwidth value by using WIDV. ($\text{MKRVAL}/(\sqrt{2})$, $\text{MKRVAL}*(\sqrt{2})$, $\text{MKRVAL}/2$, or FIXED VALUE under [Search](#))
WIDTH \square WIDHT VALUE. Impedance analyzer only.)

■ Query Response

{DIVS2|MULS2|DIV2|FIX} $\}$ \langle new line \rangle \langle ^END \rangle

■ Equivalent SCPI Command

:CALCulate:EVALuate:WIDTh:Y:TYPE \square {DIVS2|MULS2|DIV2|FIXed, \langle numeric \rangle }

■ Samples

```
OUTPUT @Hp4396;"WIDVTYPE DIV2"
```

```
OUTPUT @Hp4396;"WIDVTYPE FIXed"
```

```
OUTPUT @Hp4396;"WIDV ";3
```

WOPEN □ *<string>* [, *<numeric>*]

If the specified file exists, this command makes it write-enabled; otherwise, creates a new file and makes it write-enabled. This command takes its arguments in a different way, depending on the file format. For a DOS format file you do not have to specify its file size, for a LIF format file you must. Specify the file size, 0 or greater, so that the file can contain the maximum number of bytes used. Note that only the BDAT type is available as the LIF file format.

The format and size of an existing file cannot be changed. Therefore, if you want to change them, delete the file itself using the PURG command and then create a new file using this command.

This command is used in combination with the WRITE command and the CLOSE commands, as shown in Figure 2-2. (No query)

| Parameter | Description |
|------------------------|--|
| <i><string></i> | File name of up to 12 characters including its extension (for the LIF format, up to 10 characters) |
| <i><numeric></i> | File size (required only for the LIF format) |

WRITE □ *<block>*

Writes data in a file that has been write-enabled using the WOPEN command. Written data must take the fixed length block format (see Figure 2-1) defined in IEEE488.2. The maximum length of data is 16 Kbytes. If data is greater than 16 Kbytes, execute this command repeatedly to write it. (No query)

Generally, this command is used in combination with the WOPEN command and the CLOSE command, as shown in Figure 2-2. (No query)

| Parameter | Description |
|----------------------|---------------------------------------|
| <i><block></i> | Data in the fixed length block format |

ZA

Selects the impedance analyzer mode. (IMPEDANCE ANALYZER under Meas)

■ Query Response

{0|1}<new line><END>

| Parameter | Description |
|-----------|--|
| 0 | Impedance analyzer mode is not selected. |
| 1 | Impedance analyzer mode is selected. |

■ Equivalent SCPI Command

:INSTrument:TYPE □ ZA

ZA

ZMAPER □ <numeric>

Sets the zooming aperture value as a percentage of the span. (ZOOMING APERTURE under Marker →)

| Parameter | Range | Unit |
|-----------|---------------------|------|
| <numeric> | 0.01 to 100 of SPAN | % |

■ Query Response

{numeric} <new line><^END>

■ Equivalent SCPI Command

:DATA[:DATA] □MZAP ,<numeric>

■ Example

```
OUTPUT 717;"ZMAPER 50"
```

```
OUTPUT 717;"ZMAPER?"
```

```
ENTER 717;A
```

```
OUTPUT 717;":DATA MZAP,50"
```

```
OUTPUT 717;":DATA? MZAP"
```

```
ENTER 717;A
```

Common Commands

***CLS**

Clears the Status Byte Register, and the Event Register of the Operation Status Register, the Standard Event Status Register, and the Event Status Register B (Instrument Event Status Register). (No query)

- Example

```
OUTPUT 717;"*CLS"
```

***ESE** \square *<numeric>*

Sets the enable bits of the Standard Event Status Register.

| Parameter | Description |
|------------------------|---|
| <i><numeric></i> | 0 to 255 (decimal expression of enable bits of the operation status register) |

- Query Response

```
{numeric} <new line><^END>
```

- Example

```
OUTPUT 717;"*ESE 1"
```

```
OUTPUT 717;"*ESE?"
```

```
ENTER 717;A
```

***ESR?**

Returns the contents of the Standard Event Status Register. (Query only)

- Query Response

```
{numeric} <new line><^END>
```

- Example

```
OUTPUT 717;"*ESR?"
```

```
ENTER 717;A
```

***IDN?**

Returns the analyzer's ID.

- Query Response

```
{manufacturer} {model} {serial no.} {firmware rev.} <new line><^END>
```

- Example

```
OUTPUT 717;"*IDN?"
```

```
ENTER 717;A$
```

***IDN?**

***OPC**

Tells the analyzer to set bit 0 (Operation Complete bit) in the Standard Event Status Register when it completes all pending operations.

*OPC? query places an ASCII character 1 into the analyzer's output queue when all pending operations have been completed.

■ Query Response

{1} <new line><^END>

■ Example

```
OUTPUT 717;"*OPC"  
OUTPUT 717;"*OPC?"  
ENTER 717;A
```

***OPT?**

Queries the options installed. (Query only)

■ Query Response

{parameter} <new line><^END>

| Parameter | Description |
|-----------|------------------------------|
| (Null) | None |
| 1C2 | HP instrument BASIC |
| 1D6 | Time-gated spectrum analysis |

■ Example

```
OUTPUT 717;"*OPT?"  
ENTER 717;A$
```

***PCB** □ <numeric>

Specifies the address of a controller that is temporarily passing GPIB control to the analyzer. (Option 1C2 only; No query)

| Parameter | Description |
|-----------|-------------|
| <numeric> | 0 to 30 |

■ Example

```
OUTPUT 717;"*PCB 0"
```

***RST**

Resets the analyzer to its default values, (see Appendix D of the *Function Reference* for information on the default values), stops sweeping and taking data, and resets the HP instrument BASIC (option 1C2 only). (No query)

■ Example

```
OUTPUT 717;"*RST"
```

***SRE** □ <numeric>

Sets the enable bits of the Status Byte Register.

| Parameter | Description |
|-----------|--|
| <numeric> | 0 to 255 (decimal expression of enable bits of the status byte register) |

■ Query Response

```
{numeric} <new line><^END>
```

■ Example

```
OUTPUT 717;"*SRE 1"
OUTPUT 717;"*SRE?"
ENTER 717;A
```

***STB?**

Reads the Status Byte Register by reading the master summary status bit. (Query only)

■ Query Response

```
{numeric} <new line><^END>
```

■ Example

```
OUTPUT 717;"*STB?"
ENTER 717;A
```

***TRG**

Triggers the analyzer when the trigger mode is set to BUS trigger. (No query)

■ Example

```
OUTPUT 717;"*TRG"
```

***TRG**

***TST?**

Executes an internal self-test and returns the test result. (Query only)

■ Query Response

{*numeric*} <new line><END>

| Parameter | Description |
|-----------|-------------|
| 0 | Pass |
| 1 | Fail |

■ Example

```
OUTPUT 717;"*TST?"  
ENTER 717;A
```

***WAI**

Makes the analyzer wait until all previously sent commands are completed. (No query)

■ Example

```
OUTPUT 717;"*WAI"
```

SCPI Commands With No Equivalent Simple Command

:CALCulate:MATH1[:EXPRession]:CATalog?

Returns the available parameters that can be used with the :CALCulate:MATH1[:EXPRession]:NAME command. (Query only)

■ Query Response

“OFF,YREF,YTRA,ZREF,ZTRA,INVS,MP4,MP8,MP16” <new line><^END>

■ Example

```
OUTPUT 717;":CALC:MATH1:CAT?"
ENTER 717;A$
```

:CALCulate:MATH2[:EXPRession]:CATalog?

Returns the available parameters that can be used with the :CALCulate:MATH2[:EXPRession]:NAME command. (Query only)

■ Query Response

“ADD,SUB,DIV,OFF” <new line><^END>

■ Example

```
OUTPUT 717;":CALC:MATH2:CAT?"
ENTER 717;A$
```

:CALCulate:PATH?

Returns the order in which CALCulate subsystems are to be performed.

■ Query Response

“MATH1,FORM,AVER,MATH2,LIM” <new line><^END>

■ Example

```
OUTPUT 717;":CALC:PATH?"
ENTER 717;A$
```

:PROG:CATalog?

Returns all the defined program names. The program name is always "PROG", because the analyzer's HP instrument BASIC only executes a single program at a time. This command can be used from an external controller only. (Query only)

■ Query Response

```
{ "PROG" } <new line><END>
```

■ Example

```
OUTPUT 717;" :PROG:CAT?"  
ENTER 717;A$
```

:PROG[:SElected]:DEFine □ <block>

Creates and downloads programs. The DEFine query uploads programs. This command can be used from an external controller only.

| Parameter | Description |
|-----------|-------------|
| <block> | program |

The <block> must be arbitrary block program data containing the lines of program code. The first line of <block> must be a header, which shows the program size. There are two formats for the header as follows:

- #0 Allows the OUTPUT statement to send program line until END is specified in the OUTPUT statement.
- #NMM... M Specifies the program size.
 - N specifies the number of digits that define the program size
 - M... M is program size in byte (N digits)

Each line of the program must be separated by <CR> or <CR> <LF>. When the size of the <block> exceeds the amount of available memory in the instrument, the program lines are saved up to the point of memory overflow.

In the response to the DEFine query, the selected program and its size are returned. The selected program must be in either the paused or stopped state for the program to be uploaded. The <block> is uploaded as definite length arbitrary block response data. The program size is returned in the first line as the header, then program lines are returned.

:PROGram[:SElected]:NAME□<string>

■ Example

```
OUTPUT 717;":PROG:DEF #0"
OUTPUT 717;"10 PRINT ""HELLO!""
OUTPUT 717;"20 END"
OUTPUT 717;" ",END

DIM A$[100000]
OUTPUT 717;":PROG:DEF?"
ENTER 717 USING "%,2A";HEAD$ ! Gets the header.
B=VAL(HEAD$[2]) !
FOR I=1 TO B !
  ENTER 717 USING "%,A";HEAD$ !
NEXT I !
ENTER 717 USING "-K";A$ ! Gets the program.
```

:PROGram[:SElected]:DELeTE[:SElected]

Deletes the program in the BASIC editor of the analyzer. This command can be used from an external controller only. (No query)

■ Example

```
OUTPUT 717;":PROG:DEL"
```

:PROGram[:SElected]:DELeTE:ALL

Deletes the program in the BASIC editor of the analyzer. This command can be used from an external controller only. (No query)

■ Example

```
OUTPUT 717;":PROG:DEL:ALL"
```

:PROGram[:SElected]:EXECute□<string>

Executes the program command. The program must be in either paused or stopped before the EXECute command is allowed. This command can be used from an external controller only. (No query)

| Parameter | Description |
|-----------|-----------------------|
| <string> | Legal program command |

■ Example

```
OUTPUT 717;":PROG:EXEC ""STEP""
```

:PROGram[:SElected]:MALLocate□{<numeric>|DEFault}

Performs no function in the analyzer's HP instrument BASIC. This command can be used from an external controller only.

:PROGram[:SElected]:NAME□<string>

Performs no function in the analyzer's HP instrument BASIC. This command can be used from an external controller only.

:PROG[:SElected]:NAME□<string>

:PROG[:SElected]:NUMBer□<string>,<numeric 1>[,<numeric 2> [, ...
[,<numeric n>]

Sets or queries the contents of numeric program variables and arrays in the program on the BASIC editor of the analyzer. This command can be used from an external controller only.

| Parameter | Description |
|-----------|---|
| <string> | Name of an existing variable in the selected program (either character data or string data) |
| <numeric> | Variable value |

■ Query Response

{<numeric 1> [{<numeric 2> [... [{<numeric n>} ...]]} <new line><^END>

(n is the number of the array.)

■ Example

```
OUTPUT 717;":PROG:NUMB A,1"
```

```
OUTPUT 717;":PROG:NUMB? A"
```

```
ENTER 717;B
```

:PROG[:SElected]:STATe□{**RUN**|**PAUSE**|**STOP**|**CONTInue**}

Sets or queries the state of the program in the BASIC editor of the analyzer. The table below defines the affect of setting the state to the specified state from each of the possible current states. This command can be used from an external controller only.

| Desired State | Current State | | |
|---------------|---------------|-------|--------------|
| | RUN | PAUSE | STOP |
| RUN | error (-221) | RUN | RUN |
| CONT | error (-221) | RUN | error (-221) |
| PAUSE | PAUSE | PAUSE | STOP |
| STOP | STOP | STOP | STOP |

■ Query Response

{ "RUN"|"PAUS"|"STOP"|"CONT" } <new line><^END>

■ Example

```
OUTPUT 717;":PROG:STAT ""STOP""
```

```
OUTPUT 717;":PROG:STAT?"
```

```
ENTER 717;A$
```


:PROGram[:SElected]:STRing □ <string (varname)>, <string (value 1)> [, <string (value 2)> [, ... [, <string (value n)>]

Sets or queries the contents of string program variables and arrays in the program in the BASIC editor of the analyzer. If a string value is too long it is truncated when stored in the program's variable. This command can be used from an external controller only.

| Parameter | Description |
|--------------------|--|
| <string (varname)> | Name of an existing variable in the selected program (either character data or string data). |
| <string (value)> | Variable value |

■ Query Response

{string 1} [{string 2} [... [{string n} ...]] <new line><^END>

(n is the number of the array.)

■ Example

```
OUTPUT 717;":PROG:STR A,""HELLO""
OUTPUT 717;":PROG:STR? A"
ENTER 717;B$
```

:PROGram[:SElected]:WAIT

Causes no further commands or queries to be executed until the defined program exits from the RUN state. That is, the program is either stopped or paused. This command can be used from an external controller only.

■ Query Response

{1} <new line><^END>

1 is returned when the program is either stopped or paused.

■ Example

```
OUTPUT 717;":PROG:WAIT"
OUTPUT 717;":PROG:WAIT?"
ENTER 717;A
```

Note



The following commands under the EXPLicit node perform the specified functions in the same manner as the corresponding commands under the SElected node. The EXPLicit commands are included in the analyzer's GPIB commands to maintain compatibility with other SCPI instruments. Therefore, you can use either the EXPLicit or the SElected commands for the analyzer. However, you should select one set and use it consistently to avoid confusion.

:PROGrama:EXPLICIT:DEFine □ “PROG”, <block>

See “:PROGrama[:SElected]:DEFine □ <block>”.

:PROGrama:EXPLICIT:DElete □ “PROG”

See “:PROGrama[:SElected]:DElete[:SElected]”.

:PROGrama:EXPLICIT:EXECute □ “PROG”, <string>

See “:PROGrama[:SElected]:EXECute □ <string>”.

:PROGrama:EXPLICIT:MALLocate □ “PROG”, {<numeric>|DEFAULT}

See “:PROGrama[:SElected]:MALLocate □ {<numeric>|DEFAULT}”.

:PROGrama:EXPLICIT:NAME □ “PROG”, <string>

See “:PROGrama[:SElected]:NAME □ <string>”.

:PROGrama:EXPLICIT:NUMBer □ “PROG”, <varname>, <numeric 1>
[, <numeric 2> [, ... [, <numeric n>] ...]]

See “:PROGrama[:SElected]:NUMBer □ <string>, <numeric 1>[, <numeric 2> [, ... [, <numeric n>]”.

:PROGrama:EXPLICIT:STATe □ “PROG”, {RUN|PAUSE|STOP|CONTinue}

See “:PROGrama[:SElected]:STATe □ {RUN|PAUSE|STOP|CONTinue}”.

:PROGrama:EXPLICIT:STRing □ “PROG”, <varname>, <string 1>[, <string 2>
[, ... [, <string n>] ...]]

See “:PROGrama[:SElected]:STRing □ <string (varname)>, <string (value 1)>[, <string (value 2)>
[, ... [, <string (value n)>]”.

:PROGrama:EXPLICIT:WAIT “PROG”

See “:PROGrama[:SElected]:WAIT”.

:SYSTem:VERSion?

:STATus:PRESet

Clears the Operational and Questionable Status Register groups. Both event and enable registers are cleared. (No query)

■ Example

```
OUTPUT 717;":STAT:PRES"
```

:STATus:QUEStionable:CONDition?

Returns the contents of the condition register of the Questionable Register group. (Query only)
The analyzer has no operation that reports an event to the questionable register.

:STATus:QUEStionable:ENABle□<numeric>

Sets or queries the enable register of the questionable register group.
The analyzer has no operation that reports an event to the questionable register.

:STATus:QUEStionable[:EVENT]?

Returns the contents of the event register of the Questionable register group. (Query only)
The analyzer has no operation that reports an event to the questionable register.

:SYSTem:VERSion?

Returns the value corresponding to the SCPI version to which the instrument complies. (Query only)

■ Query Response

```
{YYYY.V} <new line><^END>
```

| Parameter | Description |
|-----------|------------------------------|
| YYYY | Year-version |
| V | Revision number for the year |

■ Example

```
OUTPUT 717;":SYST:VERS?"  
ENTER 717;A$
```

Service Related Commands

Note See the *Service Manual* for more information about each function.



:DIAG:EREFerence:STATe?

Tells whether an external frequency reference signal is connected to the rear-panel EXT REF INPUT connector. (Query only)

■ Query Response

{0|1} <new line><END>

| Parameter | Description |
|-----------|--|
| 0 | The external reference is not connected. |
| 1 | The external reference is connected. |

■ Example

```
OUTPUT 717;":DIAG:EREF:STAT?"  
ENTER 717;A
```

:DIAG:FREVision?

Returns the current firmware revision information. (**FIRMWARE REVISION** under **System**); Query only)

■ Query Response

"4396B REVN.NN MON DD YEAR HH:MM:SS" <new line><END>

| Parameter | Description |
|-----------|------------------------------|
| N.NN | Revision number |
| MON | Implementation date (month) |
| DD | Implementation date (date) |
| YEAR | Implementation date (year) |
| HH | Implementation time (hour) |
| MM | Implementation time (minute) |
| SS | Implementation time (second) |

■ Example

```
OUTPUT 717;":DIAG:FREV?"  
ENTER 717;A$
```

:DIAG:SERVice:BUS:DC□<numeric>

:DIAG:INIT:RESult?

Returns the result of the power on test. (Query only)

■ Query Response

{PASS|FAIL} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:INIT:RES?"
ENTER 717;A$
```

:DIAG:SERVice:BUS:AZERo□{OFF|ON|0|1}

Sets the Auto Zero Switch of the Bus Measurement. (AZ SWITCH ON off under **System**)

■ Query Response

{0|1} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:BUS:AZER ON"
OUTPUT 717;":DIAG:SERV:BUS:AZER?"
ENTER 717;A
```

:DIAG:SERVice:BUS:DC□<numeric>

Selects the DC Bus Nodes of the Bus Measurement. (DC BUS [] under **System**)

| Parameter | Description |
|-----------|-------------|
| <numeric> | 0 to 26 |

■ Query Response

{numeric} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:BUS:DC 0"
OUTPUT 717;":DIAG:SERV:BUS:DC?"
ENTER 717;A
```

:DIAG:SERVice:BUS:DC□<numeric>

:DIAG:SERVice:BUS:FREQ□<numeric>

Selects the Frequency Bus Nodes of the Bus Measurement. (FREQ BUS [] under (System))

| Parameter | Description |
|-----------|-------------|
| <numeric> | 0 to 7 |

■ Query Response

{numeric} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:BUS:FREQ 0"  
OUTPUT 717;":DIAG:SERV:BUS:FREQ?"  
ENTER 717;A
```

:DIAG:SERVice:BUS:STATe□{OFF|ON|0|1}

Sets the Bus Measurement ON or OFF. (BUS MEAS ON off under (System))

| Parameter | Description |
|-----------|---------------------|
| OFF or 0 | Bus measurement OFF |
| ON or 1 | Bus measurement ON |

■ Query Response

{0|1} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:BUS:STAT ON"  
OUTPUT 717;":DIAG:SERV:BUS:STAT?"  
ENTER 717;A
```

:DIAG:SERVice:BUS:WAIT□<numeric>

Waits starting the Bus Measurement for the specified count. (WAIT COUNT under (System))

| Parameter | Description |
|-----------|-------------|
| <numeric> | 2 to 32767 |

■ Query Response

{numeric} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:BUS:WAIT 2"  
OUTPUT 717;":DIAG:SERV:BUS:WAIT?"  
ENTER 717;A
```

:DIAG:SERVice:CCONstant:SOURce{OFF|ON|0|1}

:DIAG:SERVice:CCONstant:FRESponse{OFF|ON|0|1}

Sets the correction constants of the Frequency Response ON or OFF. (FRQ RSP CC ON off under **(System)**)

| Parameter | Description |
|-----------|--|
| OFF or 0 | Frequency response correction constant OFF |
| ON or 1 | Frequency response correction constant ON |

■ Query Response

{0|1} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:CCON:FRES ON"  
OUTPUT 717;":DIAG:SERV:CCON:FRES?"  
ENTER 717;A
```

:DIAG:SERVice:CCONstant:IFGain{OFF|ON|0|1}

Sets the correction constant of the IF Gain Error ON or OFF. (IF GAIN CC ON off under **(System)**)

| Parameter | Description |
|-----------|---------------------------------------|
| OFF or 0 | IF gain error correction constant OFF |
| ON or 1 | IF gain error correction constant ON |

■ Query Response

{0|1} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:CCON:IFG ON"  
OUTPUT 717;":DIAG:SERV:CCON:IFG?"  
ENTER 717;A
```

:DIAG:SERVice:CCONstant:SOURce{OFF|ON|0|1}

Sets the correction constant of the RF Output Level ON or OFF. (SOURCE CC ON off under **(System)**)

| Parameter | Description |
|-----------|---|
| OFF or 0 | RF output level correction constant OFF |
| ON or 1 | RF output level correction constant ON |

:DIAG:SERVice:CCONstant:SOURce {OFF|ON|0|1}

■ Query Response

{0|1} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:CCON:SOUR ON"  
OUTPUT 717;":DIAG:SERV:CCON:SOUR?"  
ENTER 717;A
```

:DIAG:SERVice:CCONstant:XTAL {OFF|ON|0|1}

Sets the correction constant of the Crystal Filter ON or OFF. (XTAL CC ON off under **System**)

| Parameter | Description |
|-----------|--|
| OFF or 0 | Crystal filter correction constant OFF |
| ON or 1 | Crystal filter correction constant ON |

■ Query Response

{0|1} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:CCON:XTAL ON"  
OUTPUT 717;":DIAG:SERV:CCON:XTAL?"  
ENTER 717;A
```

:DIAG:SERVice:IF:ADMX:MODE {AUTO|ALTErnate|DEG0|DEG90}

Sets the A/D Multiplexer of the A6 Receiver IF. (A/D MUX [] under **System**)

| Parameter | Description |
|-----------|-------------|
| AUTO | Automatic |
| ALTErnate | Alternate |
| DEG0 | 0 ° |
| DEG90 | 90 ° |

■ Query Response

{AUTO|ALT|DEG0|DEG90} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:IF:ADMX:MODE AUTO"  
OUTPUT 717;":DIAG:SERV:IF:ADMX:MODE?"  
ENTER 717;A$
```


:DIAG:SERVice:IF:GAIN:W{**AUTO|DB0|DB10**}

:DIAG:SERVice:IF:BPFilter:MODE{**AUTO|BW3M|BW1M|XTAL**}

Sets the IF Band Pass Filter of the A6 Receiver IF. (**IF BPF** under **(System)**)

| Parameter | Description |
|-----------|-------------|
| AUTO | Automatic |
| BW3M | 3 MHz |
| BW1M | 1 MHz |
| XTAL | Crystal |

■ Query Response

{**AUTO|BW3M|BW1M|XTAL**} <new line><^END>

■ Example

```
OUTPUT 717;":DIAG:SERV:IF:BPFilter:MODE AUTO"  
OUTPUT 717;":DIAG:SERV:IF:BPFilter:MODE?"  
ENTER 717;A$
```

:DIAG:SERVice:IF:GAIN:MODE{**AUTO|MANual**}

Sets the IF Gain mode of the A6 Receiver IF to Auto or Manual. Setting this mode to Manual enables setting the Gain W, Gain X, Gain Y, and Gain Z values. (**IF GAIN AUTO man** under **(System)**)

■ Query Response

{**AUTO|MAN**} <new line><^END>

■ Example

```
OUTPUT 717;":DIAG:SERV:IF:GAIN:MODE AUTO"  
OUTPUT 717;":DIAG:SERV:IF:GAIN:MODE?"  
ENTER 717;A$
```

:DIAG:SERVice:IF:GAIN:W{**AUTO|DB0|DB10**}

Sets Gain W of the A6 Receiver IF. (**GAIN W []** under **(System)**)

| Parameter | Description |
|-----------|-------------|
| AUTO | Automatic |
| DB0 | 0 dB |
| DB10 | 10 dB |

■ Query Response

{**AUTO|DB0|DB10**} <new line><^END>

■ Example

```
OUTPUT 717;":DIAG:SERV:IF:GAIN:W AUTO"  
OUTPUT 717;":DIAG:SERV:IF:GAIN:W?"  
ENTER 717;A$
```

:DIAG:SERVice:IF:GAIN:W{**AUTO|DB0|DB10**}

:DIAG:SERVice:IF:GAIN:X{**AUTO|DB0|DB18**}

Sets Gain X of the A6 Receiver IF. (**GAIN X []**) under (**System**)

| Parameter | Description |
|-----------|-------------|
| AUTO | Automatic |
| DB0 | 0 dB |
| DB18 | 18 dB |

■ Query Response

{**AUTO|DB0|DB18**} <new line><^END>

■ Example

```
OUTPUT 717;":DIAG:SERV:IF:GAIN:X AUTO"  
OUTPUT 717;":DIAG:SERV:IF:GAIN:X?"  
ENTER 717;A$
```

:DIAG:SERVice:IF:GAIN:Y{**AUTO|DB0|DB6|DB12|DB18**}

Sets Gain Y of the A6 Receiver IF. (**GAIN Y []**) under (**System**)

| Parameter | Description |
|-----------|-------------|
| AUTO | Automatic |
| DB0 | 0 dB |
| DB6 | 6 dB |
| DB12 | 12 dB |
| DB18 | 18 dB |

■ Query Response

{**AUTO|DB0|DB6|DB12|DB18**} <new line><^END>

■ Example

```
OUTPUT 717;":DIAG:SERV:IF:GAIN:Y AUTO"  
OUTPUT 717;":DIAG:SERV:IF:GAIN:Y?"  
ENTER 717;A$
```

:DIAG:SERVice:IF:LPFilter:MODE {**AUTO|BW5K|BW15K|BW50K|BW150K|THRough**}

:DIAG:SERVice:IF:GAIN:Z {**AUTO|DB0|DB2|DB4|DB18**}

Sets Gain Z of the A6 Receiver IF. (**GAIN Z []** under **(System)**)

| Parameter | Description |
|-----------|-------------|
| AUTO | Automatic |
| DB0 | 0 dB |
| DB2 | 2 dB |
| DB4 | 4 dB |
| DB18 | 18 dB |

■ Query Response

{**AUTO|DB0|DB2|DB4|DB18**} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:IF:GAIN:Z AUTO"
```

```
OUTPUT 717;":DIAG:SERV:IF:GAIN:Z?"
```

```
ENTER 717;A$
```

:DIAG:SERVice:IF:LPFilter:MODE {**AUTO|BW5K|BW15K|BW50K|BW150K|THRough**}

Sets the IF Low Pass Filter of the A6 Receiver IF. (**IF LPF []** under **(System)**)

| Parameter | Description |
|-----------|-------------|
| AUTO | Automatic |
| BW5K | 5 kHz |
| BW15K | 15 kHz |
| BW50K | 50 kHz |
| BW150K | 150 kHz |
| THRough | Through |

■ Query Response

{**AUTO|BW5K|BW15K|BW50K|BW150K|THR**} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:IF:LPF:MODE AUTO"
```

```
OUTPUT 717;":DIAG:SERV:IF:LPF:MODE?"
```

```
ENTER 717;A$
```

:DIAG:SERVice:IF:LPFILTER:MODE { **AUTO** | **BW5K** | **BW15K** | **BW50K** | **BW150K** | **THROUGH** }

:DIAG:SERVice:IF:RANGE:F { **HIGH** | **LOW** }

Sets Range F of the A6 Receiver IF. (**RANGE F: HIGH**, **LOW** under **(System)**)

■ Query Response

{ **HIGH** | **LOW** } <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:IF:RANG:F HIGH"
```

```
OUTPUT 717;":DIAG:SERV:IF:RANG:F?"
```

```
ENTER 717;A$
```

:DIAG:SERVice:IF:RANGE:MODE { **AUTO** | **MANual** }

Sets the Range mode of the A6 Receiver IF to Auto or Manual. Setting this mode to Manual enables setting Range F and Range R. (**IF RANGE AUTO man** under **(System)**)

■ Query Response

{ **AUTO** | **MAN** } <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:IF:RANG:MODE AUTO"
```

```
OUTPUT 717;":DIAG:SERV:IF:RANG:MODE?"
```

```
ENTER 717;A$
```

:DIAG:SERVice:IF:RANGE:R { **HIGH** | **LOW** }

Sets Range R of the A6 Receiver IF. (**RANGE R: HIGH**, **LOW** under **(System)**)

■ Query Response

{ **HIGH** | **LOW** } <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:IF:RANG:R HIGH"
```

```
OUTPUT 717;":DIAG:SERV:IF:RANG:R?"
```

```
ENTER 717;A
```

:DIAG:SERVice:IF:SHBW:MODE { **AUTO** | **NARRow** | **MIDDLE** | **WIDE** }

Sets the Sample and Hold of the A6 Receiver IF. (**S/H BW []** under **(System)**)

■ Query Response

{ **AUTO** | **NARR** | **MIDD** | **WIDE** } <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:IF:SHBW:MODE AUTO"
```

```
OUTPUT 717;":DIAG:SERV:IF:SHBW:MODE?"
```

```
ENTER 717;A$
```

:DIAG:SERVice:SOURce:ALCLoop□{OPEN|CLOSe}

:DIAG:SERVice:IF:TLOCAl:MODE□{AUTO|AC|DC}

Sets the 3rd Local Oscillator of the A6 Receiver IF. (S/H BW [] under (System))

■ Query Response

{AUTO|AC|DC} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:IF:TLOC:MODE AUTO"
OUTPUT 717;":DIAG:SERV:IF:TLOC:MODE?"
ENTER 717;A$
```

:DIAG:SERVice:MODE□{ON|1}

Activates the service mode. (SERVICE MODE under (System))

■ Query Response

{0|1} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:MODE ON"
OUTPUT 717;":DIAG:SERV:MODE?"
ENTER 717;A
```

:DIAG:SERVice:SOURce:ALCLoop□{OPEN|CLOSe}

Sets the ALC Loop of the A3A2 ALC. (ALC LOOP ON off under (System))

■ Query Response

{OPEN|CLOS}

■ Example

```
OUTPUT 717;":DIAG:SERV:SOUR:ALCL OPEN"
OUTPUT 717;":DIAG:SERV:SOUR:ALCL?"
ENTER 717;A
```

:DIAG:SERVice:SOURce:ALCLoop□{OPEN|CLOSE}

:DIAG:SERVice:SOURce:ATTenuator□{AUTO|DB0|DB10|DB20|DB30|DB40|DB50|DB60}

Sets the Output Attenuator of the A3A2 ALC. (OUTPUT ATT under (System))

| Parameter | Description |
|-----------|-------------|
| AUTO | Automatic |
| DB0 | 0 dB |
| DB10 | 10 dB |
| DB20 | 20 dB |
| DB30 | 30 dB |
| DB40 | 40 dB |
| DB50 | 50 dB |
| DB60 | 60 dB |

■ Query Response

{AUTO|DB0|DB10|DB20|DB30|DB40|DB50|DB60} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:SOUR:ATT AUTO"  
OUTPUT 717;":DIAG:SERV:SOUR:ATT?"  
ENTER 717;A$
```

:DIAG:SERVice:SOURce:GAIN:DAC:MODE□{AUTO|MANual}

Sets the Gain DAC mode of the A3A2 ALC to Auto or Manual. Setting this mode to Manual enables setting Gain DAC value. (GAIN DAC AUTO man under (System))

■ Query Response

{AUTO|MAN} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:SOUR:GAIN:DAC:MODE AUTO"  
OUTPUT 717;":DIAG:SERV:SOUR:GAIN:DAC:MODE?"  
ENTER 717;A$
```

:DIAG:SERVice:SOURce:LEVel:DAC:VALue□<numeric>

:DIAG:SERVice:SOURce:GAIN:DAC:VALue□<numeric>

Sets the Gain DAC value of the A3A2 ALC. (GAIN DAC VALUE under (System))

| Parameter | Description |
|-----------|-------------|
| <numeric> | 0 to 15 |

■ Query Response

{numeric} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:SOUR:GAIN:DAC:VAL 0"  
OUTPUT 717;":DIAG:SERV:SOUR:GAIN:DAC:VAL?"  
ENTER 717;A
```

:DIAG:SERVice:SOURce:LEVel:DAC:MODE□{AUTO|MANual}

Sets the Level DAC mode of the A3A2 ALC to Auto or Manual. Setting this mode to Manual enables setting the Level DAC value. (LVL DAC AUTO man under (System))

■ Query Response

{AUTO|MAN}

■ Example

```
OUTPUT 717;":DIAG:SERV:SOUR:LEV:DAC:MODE AUTO"  
OUTPUT 717;":DIAG:SERV:SOUR:LEV:DAC:MODE?"  
ENTER 717;A$
```

:DIAG:SERVice:SOURce:LEVel:DAC:VALue□<numeric>

Sets the Level DAC value of the A3A2 ALC. (LVL DAC VALUE under (System))

| Parameter | Description |
|-----------|-------------|
| <numeric> | 0 to 4095 |

■ Query Response

{numeric} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:SOUR:LEV:DAC:VAL 0"  
OUTPUT 717;":DIAG:SERV:SOUR:LEV:DAC:VAL?"  
ENTER 717;A
```

:DIAG:SERVice:SOURce:LEVel:DAC:VALue□<numeric>

:DIAG:SERVice:SOURce:MODE□{**AUTO**|**MANual**}

Sets the Source mode of the A3A2 ALC to Auto or Manual. Setting this mode to Manual enables setting the ALC Loop, Output attenuator, Level DAC, and Gain DAC. (SOURCE AUTO man under (System))

■ Query Response

{AUTO|MANual} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:SOUR:MODE AUTO"
OUTPUT 717;":DIAG:SERV:SOUR:MODE?"
ENTER 717;A$
```

:DIAG:SERVice:SYNThesizer:FLOCal:MODE□{**AUTO**|**SINGLE**|**TRIPle**}

Sets the 1st Local Oscillator of the A5 Synthesizer. (1st LO OSC [] under (System))

■ Query Response

{AUTO|SING|TRIP} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:SYNT:FLOC:MODE AUTO"
OUTPUT 717;":DIAG:SERV:SYNT:FLOC:MODE?"
ENTER 717;A$
```

:DIAG:SERVice:SYNThesizer:FN:MODE□{**AUTO**|**NARRow**|**WIDE**}

Sets the Fractional N Oscillator of the A5 Synthesizer. (FN OSC [] under (System))

■ Query Response

{AUTO|NARR|WIDE} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:SYNT:FN:MODE AUTO"
OUTPUT 717;":DIAG:SERV:SYNT:FN:MODE?"
ENTER 717;A$
```


:DIAG:SERVice:SYNThesizer:STEP:DAC:VALue□<numeric>

:DIAG:SERVice:SYNThesizer:FREQuency:OFFSet□<numeric>

Sets the Frequency Offset of the A5 Synthesizer. (Network analyzer only) (**FREQUENCY OFFSET** under **(System)**)

| Parameter | Description |
|-----------|---|
| <numeric> | -8×10^9 (–8 G) to 8×10^9 (8 G) Hz |

■ Query Response

{numeric} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:SYNT:FREQ:OFFS 0"  
OUTPUT 717;":DIAG:SERV:SYNT:FREQ:OFFS?"  
ENTER 717;A
```

:DIAG:SERVice:SYNThesizer:STEP:DAC:MODE□{**AUTO**|**MANual**}

Sets the Step DAC mode of the A5 Synthesizer to Auto or Manual. Setting this mode to Manual enables setting the Step DAC value. (**STEP DAC** under **(System)**)

■ Query Response

{AUTO|MANual} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:SYNT:STEP:DAC:MODE AUTO"  
OUTPUT 717;":DIAG:SERV:SYNT:STEP:DAC:MODE?"  
ENTER 717;A$
```

:DIAG:SERVice:SYNThesizer:STEP:DAC:VALue□<numeric>

Sets the Step DAC value of the A5 Synthesizer. (**DAC VALUE** under **(System)**)

| Parameter | Description |
|-----------|-------------|
| <numeric> | 0 to 4095 |

■ Query Response

{numeric} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:SYNT:STEP:DAC:VAL 0"  
OUTPUT 717;":DIAG:SERV:SYNT:STEP:DAC:VAL?"  
ENTER 717;A$
```

:DIAG:SERVice:SYNThesizer:STEP:DAC:VALue□<numeric>

:DIAG:SERVice:SYNThesizer:STEP:LOOP□{OPEN|CLOSe}

Sets the Step Oscillator Loop of the A5 Synthesizer. (LOOP open close under (System))

■ Query Response

{OPEN|CLOS} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:SYNT:STEP:LOOP OPEN"
OUTPUT 717;":DIAG:SERV:SYNT:STEP:LOOP?"
ENTER 717;A$
```

:DIAG:SERVice:SYNThesizer:STEP:MODE□{AUTO|MANual}

Sets the Step Oscillator mode of the A5 Synthesizer to Auto or Manual. Setting this mode to Manual enables setting the Oscillator Output, Loop, Polarity, and Step DAC. (STEP OSC AUTO man under (System))

■ Query Response

{AUTO|MAN} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:SYNT:STEP:MODE AUTO"
OUTPUT 717;":DIAG:SERV:SYNT:STEP:MODE?"
ENTER 717;A$
```

:DIAG:SERVice:SYNThesizer:STEP:OUTPut□{OFF|ON|0|1}

Sets the Step Oscillator Output ON or OFF of the A5 Synthesizer. (OSC OUT ON off under (System))

■ Query Response

{0|1} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:SYNT:STEP:OUTP ON"
OUTPUT 717;":DIAG:SERV:SYNT:STEP:OUTP?"
ENTER 717;A
```

:DIAG:SERVice:SYNThesizer:STEP:POLarity□{AUTO|POSitive|NEGative}

Sets the Step Oscillator Polarity of the A5 Synthesizer. (POLARITY [] under (System))

■ Query Response

{AUTO|POS|NEG} <new line><END>

■ Example

```
OUTPUT 717;":DIAG:SERV:SYNT:STEP:POL AUTO"
OUTPUT 717;":DIAG:SERV:SYNT:STEP:POL?"
ENTER 717;A$
```

:DIAG:TEST□<*numeric*>

Selects the diagnostic tests. (TESTS, INTERNAL TESTS, EXTERNAL TESTS, ADJUSTMENT TESTS, DISPLAY TESTS, ALL EXT TESTS under System; No query)

| Parameter | Description |
|-----------|--|
| 0 | First internal test (ALL INT). |
| 17 | First external test (FRONT PANEL DIAG). |
| 41 | First adjustment test (DC OFFST/HLD STEP). |
| 48 | First display test (TEST PATTERN 1). |
| 53 | First ALL EXT test (ALL EXT 1). |
| 58 | Miscellaneous test (IMPEDANCE TEST KIT). |

■ Example

```
OUTPUT 717;":DIAG:TEST 0"
```

:DIAG:TEST:CONTinue

Continues the diagnostic test when the test is paused for user interaction. (CONTINUE under System); No query)

■ Example

```
OUTPUT 717;":DIAG:TEST:CONT"
```

:DIAG:TEST:EXECute

Runs the selected diagnostic tests. (EXECUTE TEST under System); No query)

■ Example

```
OUTPUT 717;":DIAG:TEST:EXEC"
```

:DIAG:TEST:EXECute

:DIAG:TEST:RESult?□<numeric>

Returns the diagnostic test result. (Query only)

| Parameter | Description |
|-----------|----------------------|
| <numeric> | Test number; 0 to 67 |

■ Query Response

{ "PASS" | "FAIL" | "BUSY" | "NDON" | "DONE" } <new line> <END>

| Parameter | Description |
|-----------|-------------|
| "PASS" | Pass |
| "FAIL" | Fail |
| "BUSY" | In progress |
| "NDON" | Not done |
| "DONE" | DONE |

■ Example

```
OUTPUT 717;":DIAG:TEST:RES? 0"  
ENTER 717;A$
```

Manual Changes

Introduction

This appendix contains the information required to adapt this manual to earlier versions or configurations of the analyzer than the current printing date of this manual. The information in this manual applies directly to the 4396B Network/Spectrum/Impedance Analyzer serial number prefix listed on the title page of this manual.

Manual Changes

To adapt this manual to your 4396B, see Table A-1 and Table A-2, and make all the manual changes listed opposite your instrument's serial number and firmware version.

Instruments manufactured after the printing of this manual may be different from those documented in this manual. Later instrument versions will be documented in a manual changes supplement that will accompany the manual shipped with that instrument. If your instrument's serial number is not listed on the title page of this manual or in Table A-1, it may be documented in a *yellow MANUAL CHANGES* supplement.

In additions to change information, the supplement may contain information for correcting errors (Errata) in the manual. To keep this manual as current and accurate as possible, Agilent Technologies recommends that you periodically request the latest *MANUAL CHANGES* supplement.

For information concerning serial number prefixes not listed on the title page or in the *MANUAL CHANGE* supplement, contact the nearest Agilent Technologies office.

Turn on the line switch or execute the *IDN? command by GPIB to confirm the firmware version. See the "*IDN?" in Chapter 2 for information on the *IDN? command.

Table A-1. Manual Changes by Serial Number

| Serial Prefix or Number | Make Manual Changes |
|-------------------------|---------------------|
| | |

Table A-2. Manual Changes by Firmware Version

| Version | Make Manual Changes |
|---------|---------------------|
| 1.00 | Change 1 |
| 1.0X | Change 2 |

Serial Number

Agilent Technologies uses a two-part, nine-character serial number that is stamped on the serial number plate (see Figure A-1) attached to the rear panel. The first four digits and the letter are the serial prefix and the last five digits are the suffix.



Figure A-1. Serial Number Plate

Change 1

Please delete the command of "SAVDSTAC".

Change 2

Please delete the following commands.

CLOSE
CWD?
FNAME?
FNUM?
FSIZE?
READ?
ROPEN
WOPEN
WRITE

Command Summary

This appendix summarizes the GPIB commands (and SCPI commands) according to the equivalent front panel keys as follows. It also summarizes the GPIB only commands and the common commands.

- Chan 1
- Chan 2
- Meas
- Format
- Display
- Scale Ref
- Cal
- Sweep
- Source
- Trigger
- Start
- Stop
- Center
- Span
- Marker
- Marker→
- Search
- Utility
- System
- Local
- Copy
- Save
- Recall
- Preset
- GPIB Only Commands

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---|--|---|
| Chan 1 | CHAN1 | INSTRument[:SElect] CH2 or INSTRument:NSElect 2 INSTRument:STATe OFF INSTRument[:SElect] CH1 or INSTRument:NSElect 1 INSTRument:STATe ON |
| Chan 2 | CHAN2 | INSTRument[:SElect] CH1 or INSTRument:NSElect 1 INSTRument:STATe OFF INSTRument[:SElect] CH2 or INSTRument:NSElect 2 INSTRument:STATe ON |
| Meas Network Analyzer NETWORK: A/R B/R R A B CONVERSION [OFF] → See Conversion menu S-PARAMETERS → See S-parameters menu ANALYZERTYPE → See Analyzer type menu | MEAS AR MEAS BR MEAS R MEAS A MEAS B | SENSe:FUNCTion "POWer:RATio 3,2" SENSe:FUNCTion "POWer:RATio 4,2" SENSe:FUNCTion "POWer 2" SENSe:FUNCTion "POWer 3" SENSe:FUNCTion "POWer 4" |
| S-parameters menu Refl: FWD S11 [A/R] Trans:FWD S21 [B/R] Trans:REV S12 [A/R] Refl: REV S22 [B/R] CONVERSION [OFF] → See Conversion menu INPUT PORTS → See Network input port menu ANALYZER TYPE → See Analyzer type menu | MEAS S11 MEAS S21 MEAS S12 MEAS S22 | SENSe:FUNCTion "POWer:S11" SENSe:FUNCTion "POWer:S21" SENSe:FUNCTion "POWer:S12" SENSe:FUNCTion "POWer:S22" |
| Spectrum Analyzer SPECTRUM: S R A B DETECTION [POS] → See Detection menu ANALYZER TYPE → See Analyzer type menu | MEAS S MEAS R MEAS A MEAS B | SENSe:FUNCTion "POWer 1" SENSe:FUNCTion "POWer 2" SENSe:FUNCTion "POWer 3" SENSe:FUNCTion "POWer 4" |
| Impedance Analyzer IMPEDANCE:MAG(--Z--) PHASE(θ_Z) RESIST(R) REACT(X) ADMITTANCE:MAG(--Y--) PHASE(θ_Y) | MEAS IMAG MEAS IPH MEAS IRE MEAS IIM MEAS AMAG MEAS APH | CALCulate:MATH1:NAME IMPedance CALCulate:FORMat MLINear CALCulate:MATH1:NAME IMPedance CALCulate:FORMat PHASe CALCulate:MATH1:NAME IMPedance CALCulate:FORMat REAL CALCulate:MATH1:NAME IMPedance CALCulate:FORMat IMAGinary CALCulate:MATH1:NAME ADMittance CALCulate:FORMat MLINear CALCulate:MATH1:NAME ADMittance CALCulate:FORMat PHASe |

B-2 Command Summary

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---|----------------|---|
| Impedance Analyzer (Continued) | | |
| CONDUCT(G) | MEAS ARE | CALCulate:MATH1:NAME ADMittance CALCulate:FORMat REAL |
| SUSCEPT(B) | MEAS AIM | CALCulate:MATH1:NAME ADMittance CALCulate:FORMat IMAGinary |
| REFL.COEF:MAG(Γ) | MEAS RCM | CALCulate:MATH1:NAME OFF CALCulate:FORMat MLINear |
| PHASE(θ_{Γ}) | MEAS RCPH | CALCulate:MATH1:NAME OFF CALCulate:FORMat PHASe |
| REAL(Γ_X) | MEAS RCR | CALCulate:MATH1:NAME OFF CALCulate:FORMat REAL |
| IMAG(Γ_Y) | MEAS RCIM | CALCulate:MATH1:NAME OFF CALCulate:FORMat IMAGinary |
| CAPCITNCE:PRL(Cp) | MEAS CP | CALCulate:MATH1:NAME IMPedance CALCulate:FORMat CP |
| SER(Cs) | MEAS CS | CALCulate:MATH1:NAME IMPedance CALCulate:FORMat CS |
| INDUCTNCE:PRL(Lp) | MEAS LP | CALCulate:MATH1:NAME IMPedance CALCulate:FORMat LP |
| SER(Ls) | MEAS LS | CALCulate:MATH1:NAME IMPedance CALCulate:FORMat LS |
| RESISTNCE:PRL(Rp) | MEAS RP | CALCulate:MATH1:NAME IMPedance CALCulate:FORMat RP |
| SER(Rs) | MEAS RS | CALCulate:MATH1:NAME IMPedance CALCulate:FORMat RS |
| D FACTOR | MEAS D | CALCulate:MATH1:NAME IMPedance CALCulate:FORMat D |
| Q FACTOR | MEAS Q | CALCulate:MATH1:NAME IMPedance CALCulate:FORMat Q |
| FIXTURE | MODIFIX | SYSTem:FIXTure:MODify → See Fixture menu |
| ANALYZER TYPE → See Analyzer type menu | | |
| Analyzer type menu | | |
| NETWORK ANALYZER | NA | INSTrument:TYPE NA |
| SPECTRUM ANALYZER | SA | INSTrument:TYPE SA |
| IMPEDANCE ANALYZER | ZA | INSTrument:TYPE ZA |
| Conversion menu | | |
| CONVERSION OFF | CONV OFF | CALCulate:MATH1[:EXPReSSion]:NAME OFF |
| Z:RefI | CONV ZREF | CALCulate:MATH1[:EXPReSSion]:NAME ZREF |
| Z:Trans | CONV ZTRA | CALCulate:MATH1[:EXPReSSion]:NAME ZTRA |
| Y:RefI | CONV YREF | CALCulate:MATH1[:EXPReSSion]:NAME YREF |
| Y:Trans | CONV YTRA | CALCulate:MATH1[:EXPReSSion]:NAME YTRA |
| IS | CONV ONEDS | CALCulate:MATH1[:EXPReSSion]:NAME INVS |
| MORE | | |
| CONVERSION 4*PHASE | CONV MP4 | CALCulate:MATH1[:EXPReSSion]:NAME MP4 |
| 8*PHASE | CONV MP8 | CALCulate:MATH1[:EXPReSSion]:NAME MP8 |
| 16*PHASE | CONV MP16 | CALCulate:MATH1[:EXPReSSion]:NAME MP16 |
| RETURN | | |
| RETURN | | |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---|---|--|
| Detection menu DETECTION POS PEAK NEG PEAK SAMPLE RETURN | DET POS DET NEG DET SAM | SENSE:DETECTOR[:FUNCTION] POSITIVE SENSE:DETECTOR[:FUNCTION] NEGATIVE SENSE:DETECTOR[:FUNCTION] SAMPLE |
| Fixture Menu SELECT FIXTURE FIXTURE:NONE 16191 16192 16193 16194 USER RETURN SAVE USER FXTR KIT MODIFY [NONE] DEFINE EXTENSION LABEL FIXTURE KIT DONE (MODIFIED) | FIXT NONE FIXT 16191 FIXT 16192 FIXT 16193 FIXT 16194 FIXT USED SAVUFIXT MODIFIX FIXE <numeric> LABEFIX <string> FIXKDONE | SYSTEM:FIXTURE NONE SYSTEM:FIXTURE 16191 SYSTEM:FIXTURE 16192 SYSTEM:FIXTURE 16193 SYSTEM:FIXTURE 16194 SYSTEM:FIXTURE UDEFINED SYSTEM:FIXTURE:SAVE SYSTEM:FIXTURE:MODIFY SYSTEM:FIXTURE:DISTANCE <numeric> SYSTEM:FIXTURE:LABEL <string> SYSTEM:FIXTURE:MODIFY:SAVE |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---|---|--|
| Format Network Analyzer FORMAT:LOG MAG PHASE DELAY SMITH [Re Im] POLAR [Re Im] LIN MAG SWR MORE FORMAT-REAL IMAGINARY EXPANDED PHASE ADMITTANCE [Re Im] RETURN | FMT LOGM FMT PHAS FMT DELA FMT SMITH FMT POLA FMT LINM FMT SWR FMT REAL FMT IMAG FMT EXPP FMT ADMIT | DISPlay[:WINDow]:TRACe:GRATICule:FORMat RECTangle CALCulate:FORMat MLOGarithmic DISPlay[:WINDow]:TRACe:GRATICule:FORMat RECTangle CALCulate:FORMat PHASe DISPlay[:WINDow]:TRACe:GRATICule:FORMat RECTangle CALCulate:FORMat GDELAy DISPlay[:WINDow]:TRACe:GRATICule:FORMat SMITH CALCulate:FORMat COMPLex DISPlay[:WINDow]:TRACe:GRATICule:FORMat POLar CALCulate:FORMat COMPLex DISPlay[:WINDow]:TRACe:GRATICule:FORMat RECTangle CALCulate:FORMat MLINear DISPlay[:WINDow]:TRACe:GRATICule:FORMat RECTangle CALCulate:FORMat SWR DISPlay[:WINDow]:TRACe:GRATICule:FORMat RECTangle CALCulate:FORMat REAL DISPlay[:WINDow]:TRACe:GRATICule:FORMat RECTangle CALCulate:FORMat IMAGinary DISPlay[:WINDow]:TRACe:GRATICule:FORMat RECTangle CALCulate:FORMat UPHase DISPlay[:WINDow]:TRACe:GRATICule:FORMat ADMittance CALCulate:FORMat COMPLex |
| Spectrum Analyzer FORMAT:SPECTRUM NOISE UNIT: dBm dBV dBuV WATT VOLT | FMT SPECT FMT NOISE SAUNIT DBM SAUNIT DBV SAUNIT DBUV SAUNIT W SAUNIT V | SENSe:FUNCTion "POWer {1-4}" SENSe:FUNCTion "POWer{1-4}:PSDensity" CALCulate:FORMat MLOGarithmic UNIT:POWer DBM CALCulate:FORMat MLOGarithmic UNIT:POWer DBV CALCulate:FORMat MLOGarithmic UNIT:POWer DBUV CALCulate:FORMat MLINear UNIT:POWer W CALCulate:FORMat MLINear UNIT:POWer V |
| Impedance Analyzer FORMAT: LIN Y-AXIS LOG Y-AXIS POLAR CHART SMITH CHART ADMITTANCE CHART COMPLEX PLANE PHASE UNIT [DEG] EXP PHASE ON off EXP PHASE on OFF | FMT LINY FMT LOGY FMT POLA FMT SMITH FMT ADMIT FMT COMP PHAU {DEG RAD} EXPP ON EXPP OFF | DISPlay[:WINDow]:TRACe:GRATICule:FORMat RECTangle DISPlay[:WINDow]:TRACe:Y:SPACE LINear DISPlay[:WINDow]:TRACe:GRATICule:FORMat RECTangle DISPlay[:WINDow]:TRACe:Y:SPACE LOGarithmic DISPlay[:WINDow]:TRACe:GRATICule:FORMat POLar CALCulate:FORMat COMPLex DISPlay[:WINDow]:TRACe:GRATICule:FORMat SMITH CALCulate:FORMat COMPLex DISPlay[:WINDow]:TRACe:GRATICule:FORMat ADMittance CALCulate:FORMat COMPLex DISPlay[:WINDow]:TRACe:GRATICule:FORMat CPLane CALCulate:FORMat:COMPLex CALCulate:FORMat:UNIT:ANGLE {DEG RAD} CALCulate:FORMat UPHase CALCulate:FORMat PHASe |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|--------------------|--------------------|--|
| Display | | |
| DUAL CHAN ON off | DUAC {ON 1} | When channel 1 is active, INSTRument[:SElect] CH2 INSTRument:STATe ON INSTRument[:SElect] CH1 INSTRument:STATe ON When channel 2 is active, INSTRument[:SElect] CH1 INSTRument:STATe ON INSTRument[:SElect] CH2 INSTRument:STATe ON |
| | DUAC {OFF 0} | INSTRument[:SElect] {CH1 CH2} INSTRument:STATe OFF INSTRument[:SElect] {CH1 CH2} |
| DISPLAY: DATA | DISP DATA | DISPlay[:WINDow]:TRACe1:STATe {ON 1} DISPlay[:WINDow]:TRACe2:STATe {OFF 0} |
| MEMORY | DISP MEMO | DISPlay[:WINDow]:TRACe2:STATe {ON 1} DISPlay[:WINDow]:TRACe1:STATe {OFF 0} |
| DATA & MEMORY | DISP DATM | DISPlay[:WINDow]:TRACe1:STATe {ON 1} DISPlay[:WINDow]:TRACe2:STATe {ON 1} |
| DATA → MEMORY | DATMEM | TRACe:COpy MTRace,DTRace |
| DATA HOLD [OFF] | | |
| HOLD: OFF | DHOLD OFF | CALCulate:AVERAge:STATe {OFF 0} |
| MAX | DHOLD MAX | CALCulate:AVERAge:TYPE MAXimum CALCulate:AVERAge:STATe {ON 1} |
| MIN | DHOLD MIN | CALCulate:AVERAge:TYPE MINimum CALCulate:AVERAge:STATe {ON 1} |
| RETURN | | |
| DATA MATH [DATA] | | |
| DATA MATH: DATA | MATH DATA | CALCulate:MATH2[:EXPRession]:NAME OFF |
| DATA-MEM | MATH DMNM | CALCulate:MATH2[:EXPRession]:NAME SUB |
| DATA + MEM | MATH DPLM | CALCulate:MATH2[:EXPRession]:NAME ADD |
| DATA/MEM | MATH DDVM | CALCulate:MATH2[:EXPRession]:NAME DIV |
| DEFAULT GAIN & OFS | DEFGO | DATA[:DATA] GAIN,1 DATA[:DATA] OFFS,0 |
| OFFSET | | |
| MKR → OFFSET | MKROFS | DATA[:DATA] OFFS,MARKer |
| OFFSET VALUE | DATOVAL <numeric> | DATA[:DATA] OFFS,<numeric> |
| | DATOVAL? | DATA[:DATA]? OFFS |
| AUX OFFSET VALUE | DATAOVAL <numeric> | DATA[:DATA] AOFF,<numeric> |
| | DATAOVAL? | DATA[:DATA]? AOFF |
| RETURN | | |
| GAIN | DATGAIN <numeric> | DATA[:DATA] GAIN,<numeric> |
| | DATGAIN? | DATA[:DATA]? GAIN |
| RETURN | | |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---|-----------------------------|--|
| MORE | | |
| SPLIT DISP ON off | SPLD {ON 1} SPLD {OFF 0} | DISPlay[:WINDow]:FORMat ULOWer DISPlay[:WINDow]:FORMat FBACk |
| DISPLAY ALLOCATION | | |
| ALL INSTRUMENT | DISA ALLI | DISPlay[:WINDow]:ALLocation INSTrument |
| HALF INSTR HALF BASIC | DISA HIHB | DISPlay[:WINDow]:ALLocation HIHB |
| ALL BASIC | DISA ALLB | DISPlay[:WINDow]:ALLocation BASic |
| BASIC STATUS | DISA BASS | DISPlay[:WINDow]:ALLocation BSTatus |
| RETURN | | |
| EQUIV CKT MENU → See Equivalent circuit menu | | |
| TITLE → See Title menu | TITL <string> | DISPlay[:WINDow]:TEXT17[:DATA] <string> |
| ADJUST DISPLAY → See Adjust display menu | | |
| FREQUENCYBLANK | FREQ | DISPlay:ANNotation:FREQuency {OFF 0} SYSTem:SECurity[:STATe] {ON 1} |
| RETURN | | |
| Equivalent Circuit Menu | | |
| SELECT EQV CKT [A] | | |
| CKT A | EQUC CIRA | CALCulate:EVALuate:EPARameters:CIRCuit A |
| B | EQUC CIRB | CALCulate:EVALuate:EPARameters:CIRCuit B |
| C | EQUC CIRC | CALCulate:EVALuate:EPARameters:CIRCuit C |
| D | EQUC CIRD | CALCulate:EVALuate:EPARameters:CIRCuit D |
| E | EQUC CIRE | CALCulate:EVALuate:EPARameters:CIRCuit E |
| CALCULATE EQV PARAMS | CALECPARA | CALCulate:EVALuate:EPARameters DISPlay[:WINDow]:TEXT18:STATe {ON 1} |
| SIMULATE F-CHAR | SIMFCHAR | CALCulate:EVALuate:EPARameters:SIMulation |
| RETURN | | |
| DISP EQUIV PARM [OFF] | DISECPARA OFF | DISPlay[:WINDow]:TEXT18:STAT {0 OFF} |
| DISP EQUIV PARM [ON] | DISECPARA ON | DISPlay[:WINDow]:TEXT18:STAT {1 ON} |
| DEFINE EQV PARAMS | | |
| PARAMETER R1 | DEFECR1 <numeric> | DATA[:DATA] EQR1,<numeric> |
| C1 | DEFEC1 <numeric> | DATA[:DATA] EQC1,<numeric> |
| L1 | DEFEL1 <numeric> | DATA[:DATA] EQL1,<numeric> |
| C0 | DEFEC0 <numeric> | DATA[:DATA] EQC0,<numeric> |
| SIMULATE F-CHAR | SIMFCHAR | CALCulate:EVALuate:EPARameters:SIMulation |
| RETURN | | |
| CALCULATE EQV PARAMS | CALECPARA | CALCulate:EVALuate:EPARameters |
| SIMULATE F-CHAR | SIMFCHAR | CALCulate:EVALuate:EPARameters:SIMulation |
| RETURN | | |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|----------------------|-----------------|--|
| Adjust display menu | | |
| INTENSITY | INTE <numeric> | DISPlay:BRIGhtness <numeric> |
| BACKGROUND INTENSITY | BACI <numeric> | DISPlay:CONTrast <numeric> |
| MODIFY COLORS | | |
| CHI DATA | COLO CH1D | DISPlay:CMAP:COLor1:HSL <hue>,<sat>,<lum> |
| CHI MEM/LIMIT LINE | COLO CH1M | DISPlay:CMAP:COLor2:HSL <hue>,<sat>,<lum> |
| CH2 DATA | COLO CH2D | DISPlay:CMAP:COLor3:HSL <hue>,<sat>,<lum> |
| CH2 MEM/LIMIT LINE | COLO CH2M | DISPlay:CMAP:COLor4:HSL <hue>,<sat>,<lum> |
| GRATICULE | COLO GRAT | DISPlay:CMAP:COLor5:HSL <hue>,<sat>,<lum> |
| WARNING | COLO WARN | DISPlay:CMAP:COLor6:HSL <hue>,<sat>,<lum> |
| MORE | | |
| TEXT | COLO TEXT | DISPlay:CMAP:COLor7:HSL <hue>,<sat>,<lum> |
| IBASIC | COLO IBT | DISPlay:CMAP:COLor8:HSL <hue>,<sat>,<lum> |
| MORE | | |
| PEN 1 | COLO PEN1 | DISPlay:CMAP:COLor9:HSL <hue>,<sat>,<lum> |
| PEN 2 | COLO PEN2 | DISPlay:CMAP:COLor10:HSL <hue>,<sat>,<lum> |
| PEN 3 | COLO PEN3 | DISPlay:CMAP:COLor11:HSL <hue>,<sat>,<lum> |
| PEN 4 | COLO PEN4 | DISPlay:CMAP:COLor12:HSL <hue>,<sat>,<lum> |
| PEN 5 | COLO PEN5 | DISPlay:CMAP:COLor13:HSL <hue>,<sat>,<lum> |
| PEN 6 | COLO PEN6 | DISPlay:CMAP:COLor14:HSL <hue>,<sat>,<lum> |
| RETURN | | |
| RETURN | | |
| RETURN | | |
| DEFAULT COLORS | DEFC | DISPlay:CMAP:DEFault |
| SAVE COLORS | SVCO | DISPlay:CMAP:STORe |
| RECALL COLORS | RECC | DISPlay:CMAP:LOAD |
| RETURN | | |
| Color adjust menu | | |
| TINT | TINT <numeric> | |
| BRIGHTNESS | CBRI <numeric> | |
| COLOR | COLOR <numeric> | |
| RESET COLOR | RSCO | DISPlay:CMAP:COLor{1-14}:DEFault |
| RETURN | | |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---|---|---|
| Scale Ref Network Analyzer AUTO SCALE SCALE/DIV REFERENCE POSITION REFERENCE VALUE MKR→REFERENCE SCALE FOR [DATA] D&M SCALE [COUPLE] ELECTRICAL DELAY MENU MKR→DELAY ELECTRICAL DELAY PHASE OFFSET RETURN | AUTO SCAL <numeric> REFP <numeric> REFV <numeric> MKRREF SCAF DATA SCAF MEMO SCAC {OFF ON 0 1} MKRDELA ELED <numeric> PHAO <numeric> | DISPlay[:WINDow]:TRACe{1 2}:Y[:SCALe]:AUTO ONCE DISPlay[:WINDow]:TRACe{1 2}:Y[:SCALe]:PDIVision <numeric> DISPlay[:WINDow]:TRACe{1 2}:Y[:SCALe]:RPOStion <numeric> DISPlay[:WINDow]:TRACe{1 2}:Y[:SCALe]:RLEVel <numeric> DISPlay[:WINDow]:TRACe{1 2}:Y[:SCALe]:RLEVel MARKer DISPlay[:WINDow]:TRACe{1 2}:Y[:SCALe]:COUPle {OFF ON 0 1} SENSe:CORRection:EDELAy2 MARKer SENSe:CORRection:EDELAy2 <numeric> SENSe:CORRection:OFFSet:PHASe <numeric> |
| Spectrum Analyzer ATTEN AUTO man ATTEN SCALE/DIV REFERENCE VALUE MKR→REFERENCE SCALE FOR [DATA] D&M SCALE [COUPLE] MAX MIXER LEVEL | ATTAUTO {OFF ON 0 1} ATT <numeric> SCAL <numeric> REFV <numeric> MKRREF SCAF DATA SCAF MEMO SCAC {OFF ON 0 1} MAXMLEV <numeric> | SENSe:POWer:AC:ATTenuation:AUTO {OFF ON 0 1} SENSe:POWer:AC:ATTenuation:AUTO {OFF 0} SENSe:POWer:AC:ATTenuation <numeric> DISPlay[:WINDow]:TRACe{1 2}:Y[:SCALe]:PDIVision <numeric> DISPlay[:WINDow]:TRACe{1 2}:Y[:SCALe]:RLEVel <numeric> DISPlay[:WINDow]:TRACe{1 2}:Y[:SCALe]:RLEVel MARKer DISPlay[:WINDow]:TRACe{1 2}:Y[:SCALe]:COUPle {OFF ON 0 1} SENSe:POWer:AC:RANGe[:UPPer] <numeric> |
| Impedance Analyzer AUTO SCALE SCALE/DIV REFERENCE POSITION REFERENCE VALUE MKR→REFERENCE TOP VALUE BOTTOM VALUE MORE SCALE FOR [DATA] D&M SCALE [COUPLE] REFERENCE X VALUE REFERENCE Y VALUE RETURN | AUTO SCAL <numeric> REFP <numeric> REFV <numeric> MKRREF TOPV <numeric> BOTV <numeric> SCAF DATA SCAF MEMO SCAC {OFF ON 0 1} REFX <numeric> REFY <numeric> | DISPlay[:WINDow]:TRACe{1 2}:Y[:SCALe]:AUTO ONCE DISPlay[:WINDow]:TRACe{1 2}:Y[:SCALe]:PDIVision <numeric> DISPlay[:WINDow]:TRACe{1 2}:Y[:SCALe]:RPOStion <numeric> DISPlay[:WINDow]:TRACe{1 2}:Y[:SCALe]:RLEVel <numeric> DISPlay[:WINDow]:TRACe{1 2}:Y[:SCALe]:RLEVel MARKer DISPlay:TRACe{1 2}:Y[:SCALe]:TOP <numeric> DISPlay:TRACe{1 2}:Y[:SCALe]:BOTTOm <numeric> DISPlay[:WINDow]:TRACe{1 2}:Y[:SCALe]:COUPle {OFF ON 0 1} DISPlay:TRACe{1 2}:X[:SCALe]:RLEVel <numeric> DISPlay:TRACe{1 2}:Y[:SCALe]:RLEVel <numeric> |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---------------------|---------------------|---|
| Bw/Avg | | |
| Network Analyzer | | |
| AVERAGING RESTART | AVERREST | SENSe:AVERAge:CLEAr |
| AVERAGING ON off | AVER {OFF ON 0 1} | SENSe:AVERAge[:STATe] {OFF ON 0 1} |
| AVERAGING FACTOR | AVERFACT <numeric> | SENSe:AVERAge:COUNT <numeric> |
| IF BW [40 kHz] | BW <numeric> | SENSe:BANDwidth[:RESolution]:AUTO {OFF 0} SENSe:BANDwidth[:RESolution] <numeric> |
| GROUP DELY APERTURE | GRODAPER <numeric> | CALCulate:GDAPerture:APERture <numeric> |
| Spectrum Analyzer | | |
| | BW <numeric> | SENSe:BANDwidth[:RESolution]:AUTO {OFF 0} SENSe:BANDwidth[:RESolution] <numeric> |
| AVERAGING RESTART | AVERREST | SENSe:AVERAge:CLEAr |
| AVERAGING ON off | AVER {OFF ON 0 1} | SENSe:AVERAge[:STATe] {OFF ON 0 1} |
| AVERAGING FACTOR | AVERFACT <numeric> | SENSe:AVERAge:COUNT <numeric> |
| RES BW AUTO man | BWAUTO {OFF ON 0 1} | SENSe:BANDwidth[:RESolution]:AUTO {OFF ON 0 1} |
| RES BW [3 MHz] | BW <numeric> | SENSe:BANDwidth[:RESolution]:AUTO {OFF 0} SENSe:BANDwidth[:RESolution] <numeric> |
| RBW/SPAN RATIO | BWSRAT <numeric> | SENSe:BANDwidth[:RESolution]:RATio <numeric> |
| VIDEO BW | VBW <numeric> | SENSe:BANDwidth:VIDeo <numeric> |
| Impedance Analyzer | | |
| AVERAGING RESTART | AVERREST | SENSe:AVERAge:CLEAr |
| AVERAGING ON off | AVER {OFF ON 0 1} | SENSe:AVERAge[:STATe] {OFF ON 0 1} |
| AVERAGING FACTOR | AVERFACT <numeric> | SENSe:AVERAge:COUNT <numeric> |
| IF BW [40 kHz] | BW <numeric> | SENSe:BANDwidth[:RESolution] <numeric> |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---|--|---|
| Cal Network Analyzer CORRECTION ON off CALIBRATE MENU CALIBRATE: NONE RESPONSE → See Response menu RESPONSE & ISOL/N → See Response/isolation menu S11 1-PORT → See S11 1-port menu S22 1-PORT → See S22 1-port menu FULL 2-PORT → See Full 2-port menu ONE PATH 2-PORT → See One-path 2-port menu RESUME CAL SEQUENCE CAL KIT [7mm] CAL KIT: 7mm 3.5mm N 50 ohm N 75 ohm USER KIT SAVE USER KIT MODIFY [7mm] → See Modify calkit menu RETURN MORE PORT EXTENSIONS → See Reference plane menu VELOCITY FACTOR SET Z0 RETURN | CORR {OFF ON 0 1} CALI NONE CALI RESP CALI RAI CALI S111 CALI S221 CALI FUL2 CALI ONE2 RESC CALK APC7 CALK APC35 CALK N50 CALK N75 CALK USED SAVEUSEK MODI1 VELOFACT <numeric> SETZ <numeric> | SENSe:CORRection[:STATe] {OFF ON 0 1} SENSe:CORRection:COLLect:METhod NONE SENSe:CORRection:COLLect:METhod RESPonse SENSe:CORRection:COLLect:METhod RAIsol SENSe:CORRection:COLLect:METhod S111 SENSe:CORRection:COLLect:METhod S221 SENSe:CORRection:COLLect:METhod TPORT SENSe:CORRection:COLLect:METhod OPTPort SENSe:CORRection:COLLect:RESume SENSe:CORRection:CKIT APC7 SENSe:CORRection:CKIT APC35 SENSe:CORRection:CKIT N50 SENSe:CORRection:CKIT N75 SENSe:CORRection:CKIT UDEFined SENSe:CORRection:CKIT:MODify:SAVE SENSe:CORRection:CKIT:MODify SENSe:CORRection:RVELocity <numeric> SENSe:CORRection:CIMPedance <numeric> |
| Response standard menu defined std 1 defined std 2 defined std 3 defined std 4 defined std 5 defined std 6 defined std 7 DONE: RESPONSE | STANA STANB STANC STAND STANE STANF STANG RESPDONE | SENSe:CORRection:COLLect[:ACQuire] STANdard1 SENSe:CORRection:COLLect[:ACQuire] STANdard2 SENSe:CORRection:COLLect[:ACQuire] STANdard3 SENSe:CORRection:COLLect[:ACQuire] STANdard4 SENSe:CORRection:COLLect[:ACQuire] STANdard5 SENSe:CORRection:COLLect[:ACQuire] STANdard6 SENSe:CORRection:COLLect[:ACQuire] STANdard7 SENSe:CORRection:COLLect:SAVE2 |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---|--|---|
| Response/isolation menu RESPONSE → See OPEN/SHORT/Response standard menu ISOL'N STD DONE RESP ISOL'N CAL | RAIRESP RAISOL RAID | SENSe:CORRection:COLLect[:ACQuire] RESP SENSe:CORRection:COLLect[:ACQuire] ISOL SENSe:CORRection:COLLect:SAVE3 |
| S11 1-port menu [S11]: OPEN ¹ SHORT ¹ LOAD DONE: 1-PORT CAL | CLASS11A CLASS11B CLASS11C SAV1 | SENSe:CORRection:COLLect[:ACQuire] CS11A SENSe:CORRection:COLLect[:ACQuire] CS11B SENSe:CORRection:COLLect[:ACQuire] CS11C SENSe:CORRection:COLLect:SAVE4 |
| S22 1-port menu [S22]: OPEN ¹ SHORT ¹ LOAD DONE: 1-PORT CAL | CLASS22A CLASS22B CLASS22C SAV1 | SENSe:CORRection:COLLect[:ACQuire] CS22A SENSe:CORRection:COLLect[:ACQuire] CS22B SENSe:CORRection:COLLect[:ACQuire] CS22C SENSe:CORRection:COLLect:SAVE4 |
| Full 2-port menu REFLECT'N [S11]: OPEN ¹ SHORT ¹ LOAD [S22]: OPEN ¹ SHORT ¹ LOAD REFLECT'N DONE TRANSMISSION FWD. TRANS. THRU FWD. MATCH THRU REV. TRANS. THRU REV. MATCH THRU TRANS. DONE ISOLATION OMIT ISOLATION FWD.ISOL'N ISOL'N STD REV.ISOL'N ISOL'N STD ISOLATION DONE DONE: 2-PORT CAL | REFL CLASS11A CLASS11B CLASS11C CLASS22A CLASS22B CLASS22C REFD TRAN FWDT FWDM REVT REVM TRAD ISOL OMII FWDI REVI ISOD SAV2 | SENSe:CORRection:COLLect[:ACQuire] REFL2 SENSe:CORRection:COLLect[:ACQuire] CS11A SENSe:CORRection:COLLect[:ACQuire] CS11B SENSe:CORRection:COLLect[:ACQuire] CS11C SENSe:CORRection:COLLect[:ACQuire] CS22A SENSe:CORRection:COLLect[:ACQuire] CS22B SENSe:CORRection:COLLect[:ACQuire] CS22C SENSe:CORRection:COLLect:SAVE5 SENSe:CORRection:COLLect[:ACQuire] TRAN2 SENSe:CORRection:COLLect[:ACQuire] FWDT SENSe:CORRection:COLLect[:ACQuire] FWDM SENSe:CORRection:COLLect[:ACQuire] REVT SENSe:CORRection:COLLect[:ACQuire] REVM SENSe:CORRection:COLLect:SAVE6 SENSe:CORRection:COLLect[:ACQuire] ISOL2 SENSe:CORRection:COLLect[:ACQuire] OMII SENSe:CORRection:COLLect[:ACQuire] FWDI SENSe:CORRection:COLLect[:ACQuire] REVI SENSe:CORRection:COLLect:SAVE7 SENSe:CORRection:COLLect:SAVE8 |

1 See OPEN/SHORT/Response standard menu when Type-N calkits or user calkit is selected.

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|--|--|---|
| One-path 2-port menu REFLECT'N [S11]: OPEN SHORT LOAD REFLECT'N DONE TRANSMISSION FWD. TRANS. THRU FWD. MATCH THRU TRANS. DONE ISOLATION OMITISOLATION FWD.ISOL'N ISOL'N STD REV.ISOL'N ISOL'N STD ISOLATION DONE DONE: 2-PORT CAL | REFL CLASS11A CLASS11B CLASS11C REFD TRAN FWDT FWDM TRAD ISOL OMII FWDI REVI ISOD SAV2 | SENSe:CORRection:COLLect[:ACQuire] REFL2 SENSe:CORRection:COLLect[:ACQuire] CS11A SENSe:CORRection:COLLect[:ACQuire] CS11B SENSe:CORRection:COLLect[:ACQuire] CS11C SENSe:CORRection:COLLect:SAVE5 SENSe:CORRection:COLLect[:ACQuire] TRAN2 SENSe:CORRection:COLLect[:ACQuire] FWDT SENSe:CORRection:COLLect[:ACQuire] FWDM SENSe:CORRection:COLLect:SAVE6 SENSe:CORRection:COLLect[:ACQuire] ISOL2 SENSe:CORRection:COLLect[:ACQuire] OMII SENSe:CORRection:COLLect[:ACQuire] FWDI SENSe:CORRection:COLLect[:ACQuire] REVI SENSe:CORRection:COLLect:SAVE7 SENSe:CORRection:COLLect:SAVE8 |
| OPEN/SHORT/Response standard menu defined std 1 defined std 2 defined std 3 defined std 4 defined std 5 defined std 6 defined std 7 DONE: OPENS/SHORTS /Response | STANA STANB STANC STAND STANE STANF STANG DONE | SENSe:CORRection:COLLect[:ACQuire] STANdard1 SENSe:CORRection:COLLect[:ACQuire] STANdard2 SENSe:CORRection:COLLect[:ACQuire] STANdard3 SENSe:CORRection:COLLect[:ACQuire] STANdard4 SENSe:CORRection:COLLect[:ACQuire] STANdard5 SENSe:CORRection:COLLect[:ACQuire] STANdard6 SENSe:CORRection:COLLect[:ACQuire] STANdard7 SENSe:CORRection:COLLect:SAVE1 |
| Reference plane menu EXTENSIONS ON off EXTENSION INPUT R EXTENSION INPUT A EXTENSION INPUT B EXTENSION PORT 1 EXTENSION PORT 2 RETURN | PORE {OFF ON 0 1} PORTR <numeric> PORTA <numeric> PORTB <numeric> PORT1 <numeric> PORT2 <numeric> | SENSe:CORRection:EDELay:STATe {OFF ON 0 1} SENSe:CORRection:EDELay:PORT3:TIME <numeric> SENSe:CORRection:EDELay:PORT4:TIME <numeric> SENSe:CORRection:EDELay:PORT5:TIME <numeric> SENSe:CORRection:EDELay:PORT1:TIME <numeric> SENSe:CORRection:EDELay:PORT2:TIME <numeric> |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---------------------|------------------------|--|
| Modify calkit menu | | |
| DEFINE STANDARD | | |
| STD NO.1 [SHORT] | DEFS 1 | SENSe:CORRection:CKIT:SELEct STANdard1 |
| STD NO.2 [OPEN] | DEFS 2 | SENSe:CORRection:CKIT:SELEct STANdard2 |
| STD NO.3 [LOAD] | DEFS 3 | SENSe:CORRection:CKIT:SELEct STANdard3 |
| STD NO.4 [DEL/THRU] | DEFS 4 | SENSe:CORRection:CKIT:SELEct STANdard4 |
| STD NO.5 [LOAD] | DEFS 5 | SENSe:CORRection:CKIT:SELEct STANdard5 |
| STD NO.6 [LOAD] | DEFS 6 | SENSe:CORRection:CKIT:SELEct STANdard6 |
| STD NO.7 [SHORT] | DEFS 7 | SENSe:CORRection:CKIT:SELEct STANdard7 |
| STD NO.8 [OPEN] | DEFS 8 | SENSe:CORRection:CKIT:SELEct STANdard8 |
| SPECIFY CLASS | | |
| SPECIFY: S11A | SPECS11A <value, ... > | SENSe:CORRection:CKIT:CLASs1:STANdard <n,n, ... > |
| S11B | SPECS11B <value, ... > | SENSe:CORRection:CKIT:CLASs2:STANdard <n,n, ... > |
| S11C | SPECS11C <value, ... > | SENSe:CORRection:CKIT:CLASs3:STANdard <n,n, ... > |
| SPECIFY: S22A | SPECS22A <value, ... > | SENSe:CORRection:CKIT:CLASs4:STANdard <n,n, ... > |
| S22B | SPECS22B <value, ... > | SENSe:CORRection:CKIT:CLASs5:STANdard <n,n, ... > |
| S22C | SPECS22C <value, ... > | SENSe:CORRection:CKIT:CLASs6:STANdard <n,n, ... > |
| MORE | | |
| SPECIFY:FWD.TRANS. | SPECFWDT <value, ... > | SENSe:CORRection:CKIT:CLASs7:STANdard <n,n, ... > |
| REV.TRANS. | SPECREVT <value, ... > | SENSe:CORRection:CKIT:CLASs8:STANdard <n,n, ... > |
| FWD.MATCH | SPECFWDM <value, ... > | SENSe:CORRection:CKIT:CLASs9:STANdard <n,n, ... > |
| REV.MATCH | SPECREVM <value, ... > | SENSe:CORRection:CKIT:CLASs10:STANdard <n,n, ... > |
| RESPONSE | SPECRESP <value, ... > | SENSe:CORRection:CKIT:CLASs11:STANdard <n,n, ... > |
| RESPONSE & ISO'N | SPECRESI <value, ... > | SENSe:CORRection:CKIT:CLASs12:STANdard <n,n, ... > |
| CLASS DONE (SPEC'D) | CLAD | SENSe:CORRection:CKIT:SAVE CLASs |
| CLASS DONE (SPEC'D) | CLAD | SENSe:CORRection:CKIT:SAVE CLASs |
| LABEL CLASS | | |
| LABEL: S11A | LABES11A <string> | SENSe:CORRection:CKIT:CLASs1:LABel <string> |
| S11B | LABES11B <string> | SENSe:CORRection:CKIT:CLASs2:LABel <string> |
| S11C | LABES11C <string> | SENSe:CORRection:CKIT:CLASs3:LABel <string> |
| LABEL: S22A | LABES22A <string> | SENSe:CORRection:CKIT:CLASs4:LABel <string> |
| S22B | LABES22B <string> | SENSe:CORRection:CKIT:CLASs5:LABel <string> |
| S22C | LABES22C <string> | SENSe:CORRection:CKIT:CLASs6:LABel <string> |
| MORE | | |
| LABEL:FWD.TRANS. | LABEFWDT <string> | SENSe:CORRection:CKIT:CLASs7:LABel <string> |
| REV.TRANS. | LABEREVT <string> | SENSe:CORRection:CKIT:CLASs8:LABel <string> |
| FWD.MATCH | LABEFWDM <string> | SENSe:CORRection:CKIT:CLASs9:LABel <string> |
| REV.MATCH | LABEREVM <string> | SENSe:CORRection:CKIT:CLASs10:LABel <string> |
| RESPONSE | LABERESP <string> | SENSe:CORRection:CKIT:CLASs11:LABel <string> |
| RESPONSE&ISO'N | LABERESI <string> | SENSe:CORRection:CKIT:CLASs12:LABel <string> |
| LABEL DONE | | |
| LABEL DONE | | |
| LABEL KIT | LABK <string> | SENSe:CORRection:CKIT:LABel <string> |
| KIT DONE (MODIFIED) | KITD | SENSe:CORRection:CKIT:SAVE ALL |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---|--|--|
| Specify offset menu OFFSET DELAY OFFSET LOSS OFFSET Z0 STD OFFSET DONE | OFSD <numeric> OFSL <numeric> OFSZ <numeric> | SENSe:CORRection:CKIT:STANdard:ODELAY <numeric> SENSe:CORRection:CKIT:STANdard:OLOs <numeric> SENSe:CORRection:CKIT:STANdard:OCIMpedance <numeric> |
| Standard type menu STD TYPE: OPEN C0 C1 C2 SPECIFY OFFSET → See Specify offset menu LABEL STD STD DONE (DEFINED) SHORT SPECIFY OFFSET → See Specify offset menu LABEL STD STD DONE (DEFINED) LOAD SPECIFY OFFSET → See Specify offset menu LABEL STD STD DONE (DEFINED) DELAY/THRU SPECIFY OFFSET → See Specify offset menu LABEL STD STD DONE (DEFINED) ARBITRARY IMPEDANCE TERMINAL IMPEDANCE SPECIFY OFFSET → See Specify offset menu LABEL STD STD'DONE(DEFINED) RETURN | STDT OPEN C0 <numeric> C1 <numeric> C2 <numeric> LABS <string> STDD STDT SHOR LABS <string> STDD STDT LOAD LABS <string> STDD STDT DELA LABS <string> STDD STDT ARBI TERI <numeric> LABS <string> STDD | SENSe:CORRection:CKIT:STANdard:TYPE OPEN SENSe:CORRection:CKIT:STANdard:C0 <numeric> SENSe:CORRection:CKIT:STANdard:C1 <numeric> SENSe:CORRection:CKIT:STANdard:C2 <numeric> SENSe:CORRection:CKIT:STANdard:LABel <string> SENSe:CORRection:CKIT:STANdard:SAVE SENSe:CORRection:CKIT:STANdard:TYPE SHORT SENSe:CORRection:CKIT:STANdard:LABel <string> SENSe:CORRection:CKIT:STANdard:SAVE SENSe:CORRection:CKIT:STANdard:TYPE LOAD SENSe:CORRection:CKIT:STANdard:LABel <string> SENSe:CORRection:CKIT:STANdard:SAVE SENSe:CORRection:CKIT:STANdard:TYPE DELay SENSe:CORRection:CKIT:STANdard:LABel <string> SENSe:CORRection:CKIT:STANdard:SAVE SENSe:CORRection:CKIT:STANdard:TYPE AIMpedance SENSe:CORRection:CKIT:STANdard:TIMPedance <numeric> SENSe:CORRection:CKIT:STANdard:LABel <string> SENSe:CORRection:CKIT:STANdard:SAVE |
| Spectrum Analyzer EXECUTE LVL CAL LVL CAL DATA INPUT Z | LVL CAL LVCDT <numeric> INPZ <numeric> | CaLibration:AUTO ONCE SENSe:CORRection:OFFSet[:MAGNitude] <numeric> INPut:IMPedance <numeric> |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|-----------------------------|--------------------|---|
| Impedance Analyzer | | |
| CALIBRATE MENU | CALI IMP | SENSe:CORRection:COLLect:METhod IMP |
| OPEN ¹ | CLASIMPA | SENSe:CORRection:COLLect[:ACQuire] IMPA |
| SHORT ¹ | CLASIMPB | SENSe:CORRection:COLLect[:ACQuire] IMPB |
| LOAD | CLASIMPC | SENSe:CORRection:COLLect[:ACQuire] IMPC |
| DONE:CAL | SAVIMP | SENSe:CORRection:COLLect:SAVE |
| RESUME CAL SEQUENCE | RESC | SENSe:CORRection:COLLect:RESume |
| FIXTURE COMPEN | | |
| COMPEN MENU | COMP | SENSe:CORRection2:COLLect:METhod IMPedance |
| OPEN | COMCA | SENSe:CORRection2:COLLect[:ACQuire] STANdard1 |
| SHORT | COMCB | SENSe:CORRection2:COLLect[:ACQuire] STANdard2 |
| LOAD | COMCC | SENSe:CORRection2:COLLect[:ACQuire] STANdard3 |
| DONE:COMPEN | SAVCOM | SENSe:CORRection2:COLLect:SAVE |
| RESUME COMP SEQ | RESCOM | SENSe:CORRection2:COLLect:RESume |
| OPEN on OFF | COMCDATA {ON OFF} | SENSe:CORRection2:OPEN {ON OFF}1 0} |
| SHORT on OFF | COMCDATB {ON OFF} | SENSe:CORRection2:SHORT {ON OFF}1 0} |
| LOAD on OFF | COMCDATC {ON OFF} | SENSe:CORRection2:LOAD {ON OFF}1 0} |
| RETURN | | |
| CAL KIT [IMP 7mm] | | |
| CAL KIT : IMP 7mm | CALK APC7 | SENSe:CORRection:CKIT APC7 |
| 3.5mm | CALK APC35 | SENSe:CORRection:CKIT APC35 |
| N 50Ω | CALK N50 | SENSe:CORRection:CKIT N50 |
| N 75Ω | CALK N75 | SENSe:CORRection:CKIT N75 |
| USER KIT | CALK USED | SENSe:CORRection:CKIT UDEFined |
| SAVE USER KIT | SAVEUSEK | SENSe:CORRection:CKIT:MODify:SAVE |
| MODIFY (IMP 7mm) | MODII | SENSe:CORRection:CKIT:MODify |
| DEFINE STNADARD | | |
| → See Define standard menu | | |
| SPECIFY CLASS | | |
| SPECIFY: IMP A | SPECIMPA <numeric> | :SENSe:CORRection:CKIT:CLASs13:STANdard <num1> [, <num2> [, . . . [, <num7>]]] |
| IMP B | SPECIMPB <numeric> | :SENSe:CORRection:CKIT:CLASs14:STANdard <num1> [, <num2> [, . . . [, <num7>]]] |
| IMP C | SPECIMPC <numeric> | :SENSe:CORRection:CKIT:CLASs15:STANdard <num1> [, <num2> [, . . . [, <num7>]]] |
| CLASS DONE | CLAD | SENSe:CORRection:CKIT:SAVE CLASs |
| LABEL CLASS | | |
| LABEL: IMP A | LABEIMPA <string> | :SENSe:CORRection:CKIT:CLASs13:LABel <string> |
| IMP B | LABEIMPB <string> | :SENSe:CORRection:CKIT:CLASs14:LABel <string> |
| IMP C | LABEIMPC <string> | :SENSe:CORRection:CKIT:CLASs15:LABel <string> |
| LABEL KIT | LABK <string> | SENSe:CORRection:CKIT:LABel <string> |
| KIT DONE (MODIFIED) | KITD | SENSe:CORRection:CKIT:SAVE ALL |
| COMPEN KIT [USER] | | |
| → See Compensation kit menu | | |

1 See OPEN/SHORT/Response standard menu when Type-N calkits or user calkit is selected.

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|------------------|--------------------|---|
| PORT EXTENSIONS | | |
| EXTENSION ON off | PORE {OFF ON 0 1} | SENSe:CORRection:EDELay:STAtE {OFF ON 0 1} |
| EXTENSION VALUE | PORTZ <numeric> | SENSe:CORRection1:EDELay:PORT6[:TIME] <numeric> |
| VELOCITY FACTOR | VELOFACT <numeric> | SENSe:CORRection:RVELocity <numeric> |
| RETURN | | |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|-----------------------|---------------------|--|
| Compensation Kit Menu | | |
| SAVE COMPEN KIT | SAVUCOMK | SENSe:CORRection2:CKIT:MODify:SAVE |
| MODIFY [USER] | MODICOMK | SENSe:CORRection2:CKIT:MODify |
| DEFINE STANDARD | | |
| OPEN CONDUCT(G) | DEFSOPENG <numeric> | SENSe:CORRection2:CKIT:STNAdard1:G <numeric> |
| CAP.(C) | DEFSOPENC <numeric> | SENSe:CORRection2:CKIT:STNAdard1:C <numeric> |
| SHORT RESIST.(R) | DEFSSHORR <numeric> | SENSe:CORRection2:CKIT:STNAdard2:R <numeric> |
| INDUCT.(L) | DEFSSHORL <numeric> | SENSe:CORRection2:CKIT:STNAdard2:L <numeric> |
| LOAD RESIST.(R) | DEFSLOADR <numeric> | SENSe:CORRection2:CKIT:STNAdard3:R <numeric> |
| INDUCT.(L) | DEFSLOADL <numeric> | SENSe:CORRection2:CKIT:STNAdard3:L <numeric> |
| STD DONE (DEFINED) | COMSDONE | SENSe:CORRection2:CKIT:STANdard:SAVE |
| LABEL KIT | LABECOMK <string> | SENSe:CORRection2:CKIT:LABel <string> |
| KIT DONE (MODIFIED) | COMKDONE | SENSe:CORRection2:CKIT:SAVE |
| RETURN | | |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---|--|--|
| Sweep Network/Impedance Analyzer SWEEP TIME AUTO man SWEEP TIME h:m:s RETURN NUMBER OF POINTS COUPLED CH ON off SWEEP TYPE MENU SWEEP TYPE:LIN FREQ LOG FREQ LIST FREQ POWER SWEEP EDIT LIST SEGMENT EDIT → See NA/ZA segment menu DELETE ADD → See NA/ZA segment menu CLEAR LIST LIST DONE RETURN | SWETAUTO {OFF ON 0 1} SWET <numeric> POIN <numeric> COUC {OFF ON 0 1} SWPT LINF SWPT LOGF SWPT LIST SWPT POWE EDITLIST SEDI <numeric> SEDI [<numeric>] SDEL SADD CLEL EDITDONE | SENSE:SWEep:TIME:AUTO {OFF ON 0 1} SENSE:SWEep:TIME <numeric> SENSE:SWEep:POINts <numeric> INSTRument:COUPlE {OFF ON 0 1} SENSE:FREQuency:MODE SWEep SOURce:POWer:MODE FIXEd SENSE:SWEep:SPACing LINear SENSE:FREQuency:MODE SWEep SOURce:POWer:MODE FIXEd SENSE:SWEep:SPACing LOGarithmic SENSE:FREQuency:MODE LIST SOURce:POWer:MODE LIST SENSE:SWEep:SPACing LINear SENSE:FREQuency:MODE FIXEd SOURce:POWer:MODE SWEep SENSE:SWEep:SPACing LINear SENSE:LIST:SEGment <numeric> SENSE:LIST:SEGment:EDIT SENSE:LIST:SEGment:DELEte SENSE:LIST:SEGment:ADD SENSE:LIST:CLEAr SENSE:LIST:SAVE |
| NA/ZA segment menu SEGMENT: MKR→START MKR→STOP NUMBER of POINTS POWER IF BW MORE SEGMENT: START STOP CENTER SPAN RETURN SEGMENT QUIT SEGMENT DONE | MKRSTAR MKRSTOP POIN <numeric> POWE <numeric> BW <numeric> STAR <numeric> STOP <numeric> CENT <numeric> SPAN <numeric> SQUI SDON | SENSE:LIST:SEGment:FREQuency:STARt MARKer SENSE:LIST:SEGment:FREQuency:STOP MARKer SENSE:LIST:SEGment:POINts <numeric> SENSE:LIST:SEGment:POWer <numeric> SENSE:LIST:SEGment:BANdwidth <numeric> SENSE:LIST:SEGment:FREQuency:STARt <numeric> SENSE:LIST:SEGment:FREQuency:STOP <numeric> SENSE:LIST:SEGment:FREQuency:CENter <numeric> SENSE:LIST:SEGment:FREQuency:SPAN <numeric> SENSE:LIST:SEGment:QUIT SENSE:LIST:SEGment:SAVE |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|--------------------------|-----------------------|---|
| Spectrum Analyzer | | |
| SWEEP TIME AUTO man | SWETAUTO {OFF ON 0 1} | SENSe:SWEep:TIME:AUTO {OFF ON 0 1} |
| SWEEP TIME | SWET <numeric> | SENSe:SWEep:TIME <numeric> |
| h:m:s | | |
| RETURN | | |
| SAMPLING NORMAL repet | REPTSMP {OFF ON 0 1} | SENSe:DETEctor:CONTinuous {OFF ON 0 1} |
| NUMBER OF POINTS | POIN <numeric> | SENSe:SWEep:POINts <numeric> |
| SWEEP TYPE MENU | | |
| SWEEP TYPE:LIN FREQ | SWPT LINF | SENSe:FREQuency:MODE SWEep SOURce:POWer:MODE FIXed SENSe:SWEep:SPACing LINear |
| LIST FREQ | SWPT LIST | SENSe:FREQuency:MODE LIST SOURce:POWer:MODE LIST SENSe:SWEep:SPACing LINear |
| EDIT LIST | EDITLIST | |
| SEGMENT | | SENSe:LIST:SEGment <numeric> |
| EDIT | SEDI [<numeric>] | SENSe:LIST:SEGment:EDIT |
| → See SA segment menu | | |
| DELETE | SDEL | SENSe:LIST:SEGment:DELeTe |
| ADD | SADD | SENSe:LIST:SEGment:ADD |
| → See SA segment menu | | |
| CLEAR LIST | CLEL | SENSe:LIST:CLEar |
| LIST DONE | EDITDONE | SENSe:LIST:SAVE |
| RETURN | | |
| SA segment menu | | |
| SEGMENT: MKR→ START | MKRSTAR | SENSe:LIST:SEGment:FREQuency:STARt MARKer |
| MKR→ STOP | MKRSTOP | SENSe:LIST:SEGment:FREQuency:STOP MARKer |
| NUMBER of POINTS | POIN <numeric> | SENSe:LIST:SEGment:POINts <numeric> |
| POWER | POWE <numeric> | SENSe:LIST:SEGment:POWer <numeric> |
| RES'BW | BW <numeric> | SENSe:LIST:SEGment:BANDwidth <numeric> |
| MORE | | |
| SEGMENT: START | STAR <numeric> | SENSe:LIST:SEGment:FREQuency:STARt <numeric> |
| STOP | STOP <numeric> | SENSe:LIST:SEGment:FREQuency:STOP <numeric> |
| CENTER | CENT <numeric> | SENSe:LIST:SEGment:FREQuency:CENTer <numeric> |
| SPAN | SPAN <numeric> | SENSe:LIST:SEGment:FREQuency:SPAN <numeric> |
| RETURN | | |
| SEGMENT QUIT | SQUI | SENSe:LIST:SEGment:QUIT |
| SEGMENT DONE | SDON | SENSe:LIST:SEGment:SAVE |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---------------------------|-------------------|--|
| Source | | |
| Network Analyzer | | |
| POWER | POWE <numeric> | SOURce:POWer[:LEVel][:IMMediate][:AMPLitude] <numeric> |
| SLOPE | SLOPE <numeric> | SOURce:POWer[:LEVel][:IMMediate][:AMPLitude]:SLOPe <numeric> |
| SLOPE ON off | SLOP {OFF ON 0 1} | SOURce:POWer[:LEVel][:IMMediate][:AMPLitude]:SLOPe:STATe {OFF ON 0 1} |
| CW FREQ | CWFREQ <numeric> | SOURce:FREQuency[:CW] <numeric> |
| ATTENUATOR PORT 1 | ATTP1 <numeric> | OUTPut:ATTenuation1 <numeric> |
| ATTENUATOR PORT 2 | ATTP2 <numeric> | OUTPut:ATTenuation2 <numeric> |
| RF OUT ON off | RFO {OFF ON 0 1} | SOURce:POWer:STATe {OFF ON 0 1} |
| Spectrum Analyzer | | |
| POWER | POWE <numeric> | SOURce:POWer[:LEVel][:IMMediate][:AMPLitude] <numeric> |
| RF OUT ON off | RFO {OFF ON 0 1} | SOURce:POWer:STATe {OFF ON 0 1} |
| Impedance Analyzer | | |
| POWER | POWE <numeric> | SOURce:POWer[:LEVel][:IMMediate][:AMPLitude] <numeric> |
| CW FREQ | CWFREQ <numeric> | SOURce:FREQuency[:CW] <numeric> |
| RF OUT ON off | RFO {OFF ON 0 1} | SOURce:POWer:STATe {OFF ON 0 1} |
| Trigger | | |
| SWEEP:HOLD | HOLD | INITiate:CONTInuous {OFF 0} ABORT |
| SINGLE | SING | INITiate:CONTInuous {OFF 0} SENSe:SWEEp:COUNT 1 INITiate[:IMMediate] |
| NUMBER of GROUPS | NUMG <numeric> | INITiate:CONTInuous {OFF 0} SENSe:SWEEp:COUNT <numOfGroups> INITiate[:IMMediate] |
| CONTINUOUS | CONT | INITiate:CONTInuous {ON 1} |
| TRIGGER:[FREE RUN] | | |
| FREE RUN | TRGS INT | TRIGger:SOURce INTernal1 SENSe:SWEEp:GATed {OFF 0} |
| EXTERNAL | TRGS EXT | TRIGger:SOURce EXTernal SENSe:SWEEp:GATed {OFF 0} |
| (GPIB) | TRGS BUS | TRIGger:SOURce BUS SENSe:SWEEp:GATed {OFF 0} |
| VIDEO ¹ | TRGS VID | TRIGger:SOURce INTernal2 SENSe:SWEEp:GATed {OFF 0} |
| | VIDLVL <numeric> | TRIGger:LEVel <numeric> |
| MANUAL | TRGS MAN | TRIGger:SOURce MANual SENSe:SWEEp:GATed {OFF 0} |
| GATE [LEVEL] ¹ | TRGS GAT | TRIGger:SOURce EXTernal SENSe:SWEEp:GATed {ON 1} |
| GATE CTL:LEVEL | GATCTL LEV | SENSe:SWEEp:GATed:TRIGger LEVel |
| EDGE | GATCTL EDG | SENSe:SWEEp:GATed:TRIGger EDGE |
| GATE DELAY | GATDLY <numeric> | SENSe:SWEEp:GATed:DELay <numeric> |
| GATE LENGTH | GATLEN <numeric> | SENSe:SWEEp:GATed:LENGth <numeric> |
| RETURN | | |
| RETURN | | |
| TRIG EVENT ² | TRGEVE SWE | TRIGger:EVENT:TYPE SWEEp |
| | TRGEVE POIN | TRIGger:EVENT:TYPE POINT |
| TRIG PLRTY POS neg | TRGP {POS NEG} | TRIGger:SLOPe {POSitive NEGative} |
| MEASURE RESTART | REST | INITiate[:IMMediate]:AGAI:n:ALL |

1 Spectrum analyzer only

2 Network and impedance analyzer only

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|--|---|--|
| Start | STAR <numeric> | SENSe:FREQuency:STARt <numeric> (Frequency) or SOURce:POWer:STARt <numeric> (Power) |
| Stop | STOP <numeric> | SENSe:FREQuency:STOP <numeric> (Frequency) or SOURce:POWer:STOP <numeric> (Power) |
| Center STEP SIZE AUTO man CENTER STEP SIZE MKR→CNTR STEP MKRΔ→CNTR STEP MKR→CENTER MKRΔ→CENTER PEAK→CENTER | CENT <numeric> CNTSAUTO {OFF ON 0 1} CNTS <numeric> MKRCSTE MKRDCSTE MKRCENT MKRDCENT PEAKCENT | SENSe:FREQuency:CENTer <numeric> (Frequency) or SOURce:POWer:CENTer <numeric> (Power) SENSe:FREQuency:CENTer:STEP[:INCRement]:AUTO {OFF ON 0 1} SENSe:FREQuency:CENTter:STEP[:INCRement] <numeric> SENSe:FREQuency:CENTter:STEP[:INCRement] MARKer SENSe:FREQuency:CENTter:STEP[:INCRement] DMARKer SENSe:FREQuency:CENTer MARKer (Frequency) or SOURce:POWer:CENTer MARKer (Power) SENSe:FREQuency:CENTer DMARKer (Frequency) or SOURce:POWer:CENTer DMARKer (Power) SENSe:FREQuency:CENTer TPEak (Frequency) or SOURce:POWer:CENTer TPEak (Power) |
| Span FULL SPAN ZERO SPAN MKRΔ→SPAN | SPAN <numeric> FULS SPAN 0 MKRDSPAN | SENSe:FREQuency:SPAN <numeric> (Frequency) or SOURce:POWer:SPAN <numeric> (Power) SENSe:FREQuency:SPAN:FULL (Frequency) or SOURce:POWer:SPAN:FULL (Power) SENSe:FREQuency:SPAN 0 (Frequency) or SOURce:POWer:SPAN 0 (Power) SENSe:FREQuency:SPAN DMARKer (Frequency) or SOURce:POWer:SPAN DMARKer (Power) |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---|--|--|
| Marker | MKR {OFF ON 0 1} | DISPlay[:WINDow]:TRACe:MARKer:ALL:STATe {OFF ON 0 1} |
| Network/Impedance Analyzer SUB MKR → See Sub-marker menu CLEAR SUB MKR → See Sub-marker menu PRESET MKRS MKR ON [DATA] MKR [UNCOUPLE] MKR [CONT] ΔMODE MENU → See NA/ZA Delta mode menu | PRSMKRS MKRO DATA MKRO MEMO MKRCOUP {OFF ON 0 1} MKRCONT {OFF ON 0 1} | DISPlay[:WINDow]:TRACe:MARKer:ALL DEFault CALCulate:EVALuate:ON "DTR" CALCulate:EVALuate:ON "MTR" CALCulate:EVALuate:COUPle {OFF ON 0 1} CALCulate:EVALuate:INTerpolate {OFF ON 0 1} |
| Spectrum Analyzer SUB MKR → See Sub marker menu CLEAR SUB MKR → See Sub marker menu PRESET MKRS MKR ON [DATA] MKR [UNCOUPLE] ΔMODE MENU → See SA Delat mode menu | PRSMKRS MKRO DATA MKRO MEMO MKRCOUP {OFF ON 0 1} | DISPlay[:WINDow]:TRACe:MARKer:ALL DEFault CALCulate:EVALuate:ON "DTR" CALCulate:EVALuate:ON "MTR" CALCulate:EVALuate:COUPle {OFF ON 0 1} |
| NA/ZA Delta mode menu ΔMKR FIXED ΔMKR TRACKING ΔMKR ΔMODE OFF ΔMKR SWP PRM ΔMKR VALUE ΔMKR AUX VALUE RETURN | DMKR ON DMKR FIX DMKR TRAC DMKR OFF OUTPDMKR? DMKRPRM <numeric> DMKRVAL <numeric> DMKRAUV <numeric> | DISPlay[:WINDow]:TRACe:MARKer:RELative {ON 1} DISPlay[:WINDow]:TRACe:MARKer:RELative:REFerence MARKer DISPlay[:WINDow]:TRACe:MARKer:RELative {ON 1} DISPlay[:WINDow]:TRACe:MARKer:RELative:REFerence FIXed DISPlay[:WINDow]:TRACe:MARKer:RELative {ON 1} DISPlay[:WINDow]:TRACe:MARKer:RELative:REFerence TRACKed DISPlay[:WINDow]:TRACe:MARKer:RELative {OFF 0} CALCulate:EVALuate:REFerence:DATA? CALCulate:EVALuate:REFerence:X <numeric> CALCulate:EVALuate:REFerence:Y1 <numeric> CALCulate:EVALuate:REFerence:Y2 <numeric> |
| SA Delta mode menu ΔMKR FIXED ΔMKR TRACKING ΔMKR ΔMODE OFF ΔMKR SWP PRM ΔMKR VALUE RETURN | DMKR ON DMKR FIX DMKR TRAC DMKR OFF DMKRPRM <numeric> DMKRVAL <numeric> | DISPlay[:WINDow]:TRACe:MARKer:RELative {ON 1} DISPlay[:WINDow]:TRACe:MARKer:RELative:REFerence MARKer DISPlay[:WINDow]:TRACe:MARKer:RELative {ON 1} DISPlay[:WINDow]:TRACe:MARKer:RELative:REFerence FIXed DISPlay[:WINDow]:TRACe:MARKer:RELative {ON 1} DISPlay[:WINDow]:TRACe:MARKer:RELative:REFerence TRACKed DISPlay[:WINDow]:TRACe:MARKer:RELative {OFF 0} CALCulate:EVALuate:REFerence:X <numeric> CALCulate:EVALuate:REFerence:Y1 <numeric> |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|--|--|--|
| Sub-marker menu SUB MKR 1 2 3 4 5 6 7 RETURN | SMK{1-7} {OFF ON 0 1} SMKPRM{1-7} <numeric> OUTPSMKR{1-7}? SMKPRM{1-7}? SMKVAL{1-7}? SMKRAUV{1-7}? | DISPlay[:WINDow]:TRACe:MARKer{2-8}:STATe {OFF ON 0 1} CALCulate:EVALuate:Y{2-8}:XPOSITION <numeric> CALCulate:EVALuate:Y{2-8}:DATA? CALCulate:EVALuate:Y{2-8}:XPOSITION? CALCulate:EVALuate:Y{2-8}:VALue1? CALCulate:EVALuate:Y{2-8}:VALue2? |
| Marker→ MKR→CENTER MKR→START MKR→STOP MKR→REFERENCE PEAK→CENTER CROSS CHAN ON off MORE MKR ZOOM ZOOMING APERTURE MKRA→SPAN MKRA→CENTER CROSS CHAN ON off RETURN | MKRCENT MKRSTAR MKRSTOP MKRREF PEAKCENT CRSC {OFF ON 0 1} MKRZM ZMAPER <numeric> MKRDSPAN MKRDCENT CRSC {OFF ON 0 1} | SENSE:FREQUENCY:CENTer MARKer (Frequency) or SOURCE:POWER:CENTer MARKer (Power) SENSE:FREQUENCY:STARt MARKer (Frequency) or SOURCE:POWER:STARt MARKer (Power) SENSE:FREQUENCY:STOP MARKer (Frequency) or SOURCE:POWER:STOP MARKer (Power) DISPlay[:WINDow]:TRACe{1 2}:Y[:SCALE]:RLEVel MARKer SENSE:FREQUENCY:CENTer TPEak (Frequency) or SOURCE:POWER:CENTer TPEak (Power) CALCulate:EVALuate:EFFect:ON {1 2} SENSE:FREQUENCY:SPAN MZAPerture (Frequency) SOURCE:POWER:SPAN MZAPerture (Power) DATA[:DATA] MZAP <numeric> SENSE:FREQUENCY:SPAN DMARker (Frequency) or SOURCE:POWER:SPAN DMARker (Power) SENSE:FREQUENCY:CENTer DMARker (Frequency) or SOURCE:POWER:CENTer DMARker (Power) CALCulate:EVALuate:EFFect:ON {1 2} |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|--|---|--|
| <p>Search</p> <p>Network Analyzer/Impedance Analyzer</p> <p>SEARCH: PEAK → See Peak menu</p> <p>MAX</p> <p>MIN</p> <p>TARGET</p> <p>TARGET</p> <p>SEARCH LEFT</p> <p>SEARCH RIGHT</p> <p>SUB MKR → See Sub-marker menu</p> <p>RETURN</p> <p>MULTIPLE PEAKS → See Multiple peaks menu</p> <p>WIDTHS [OFF]</p> <p>SEARCH IN</p> <p>SEARCH OUT</p> <p>WIDTHS ON off</p> <p>WIDTH VALUE → See Width Value Menu for impedance analyzer</p> <p>RETURN</p> <p>SRCH TRACK ON off</p> <p>SRCH RANGE MENU → See Search range menu</p> | <p>if mkr = off then MKR ON</p> <p>SEAM PEAK</p> <p>SEAM MAX</p> <p>SEAM MIN</p> <p>SEAM TARG</p> <p>SEATARG <numeric></p> <p>SEAL</p> <p>SEAR</p> <p>WIDSIN</p> <p>WIDSOUT</p> <p>WIDT {OFF ON 0 1}</p> <p>OUTPMWID?</p> <p>WIDV <numeric></p> <p>TRACK {OFF ON 0 1}</p> | <p>DISPlay[:WINDow]:TRACe:MARKer:STATe {ON 1}</p> <p>CALCulate:EVALuate:Y:XPOSITION:PEAK</p> <p>CALCulate:EVALuate:Y:XPOSITION:MAXimum</p> <p>CALCulate:EVALuate:Y:XPOSITION:MINimum</p> <p>CALCulate:EVALuate:Y:XPOSITION:TARGeT <numeric></p> <p>CALCulate:EVALuate:Y:XPOSITION:TARGeT <numeric></p> <p>CALCulate:EVALuate:Y:XPOSITION:LTARGET</p> <p>CALCulate:EVALuate:Y:XPOSITION:RTARGeT</p> <p>CALCulate:EVALuate:WIDTh:XPOSITION:IN</p> <p>CALCulate:EVALuate:WIDTh:XPOSITION:OUT</p> <p>CALCulate:EVALuate:WIDTh:STATe {OFF ON 0 1}</p> <p>CALCulate:EVALuate:WIDTh:DATA?</p> <p>CALCulate:EVALuate:WIDTh:Y <numeric></p> <p>CALCulate:EVALuate:Y:XPOSITION:TRACK {MAXimum MINimum TARGeT PEAK PALL PLEFt PRIGHt OFF}</p> |
| <p>Spectrum Analyzer</p> <p>SEARCH: PEAK → See Peak menu</p> <p>MAX</p> <p>MIN</p> <p>MULTIPLE PEAKS → See Multiple peaks menu</p> <p>SGNL TRACK ON off</p> <p>SRCH TRACK ON off</p> <p>SRCH RANGE MENU → See Search range menu</p> | <p>SEAM PEAK</p> <p>SEAM MAX</p> <p>SEAM MIN</p> <p>SGTRK {OFF ON 0 1}</p> <p>TRACK {OFF ON 0 1}</p> | <p>CALCulate:EVALuate:Y:XPOSITION:PEAK</p> <p>CALCulate:EVALuate:Y:XPOSITION:MAXimum</p> <p>CALCulate:EVALuate:Y:XPOSITION:MINimum</p> <p>SENSe:TRACk:SIGNal:MARKer {OFF ON 0 1}</p> <p>CALCulate:EVALuate:Y:XPOSITION:TRACK {MAXimum MINimum TARGeT PEAK PALL PLEFt PRIGHt OFF}</p> |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---|--|--|
| Peak menu PEAK NEXT PEAK NEXT PEAK LEFT NEXT PEAK RIGHT PEAK DEF MENU → See Peak definition menu SUB MKR → See Sub-marker menu RETURN | SEAM PEAK SEANPK SEANPKL SEANPKR | CALCulate:EVALuate:Y:XPOSITION:PEAK CALCulate:EVALuate:Y:XPOSITION:NPEak CALCulate:EVALuate:Y:XPOSITION:LPEak CALCulate:EVALuate:Y:XPOSITION:RPEak |
| Multiple peaks menu SEARCH: PEAKS ALL PEAKS RIGHT PEAKS LEFT PEAK DEF MENU → See Peak definition menu SRCH TRACK ON off RETURN | SEAM PKSA SEAM PKSR SEAM PKSL TRACK {OFF ON 0 1} | CALCulate:EVALuate:Y:XPOSITION:PALL CALCulate:EVALuate:Y:XPOSITION:PRIGHT CALCulate:EVALuate:Y:XPOSITION:PLEft CALCulate:EVALuate:Y:XPOSITION:TRACK {MAXimum MINimum TARGET PEAK PALL PLEft PRIGHT OFF} |
| Peak definition menu THRESHOLD ON off THRESHOLD VALUE MKR→THRESHOLD PEAK PLRTY POS neg ¹ PEAK DEF: ΔX ¹ PEAK DEF: ΔY MKR→PEAK'DELTA ¹ RETURN | PKTHRE {OFF ON 0 1} PKTHVAL <numeric> MKRTHRE PKPOL {POS NEG} PKDLTX <numeric> PKDLTY <numeric> MKRPKD | CALCulate:EVALuate:PEAK:THReshold:STATe {OFF ON 0 1} CALCulate:EVALuate:PEAK:THReshold <numeric> CALCulate:EVALuate:PEAK:THReshold MARKer CALCulate:EVALuate:PEAK:POLarity {POSitive NEGative} CALCulate:EVALuate:PEAK:EXCursion:X <numeric> CALCulate:EVALuate:PEAK:EXCursion[:Y] <numeric> CALCulate:EVALuate:PEAK:EXCursion DMARKer |
| Width value menu MKRVAL/(√2) MKRVAL*(√2) MKRVAL/2 FIXED VALUE RETURN | WIDVTYPE DIVS2 WIDVTYPE MULS2 WIDVTYPE DIV2 WIDVTYPE FIX WIDV <numeric> | CALCulate:EVALuate:WIDTH:Y:TYPE DIVS2 CALCulate:EVALuate:WIDTH:Y:TYPE MULS2 CALCulate:EVALuate:WIDTH:Y:TYPE DIV2 CALCulate:EVALuate:WIDTH:Y:TYPE FIXed,<numeric> |
| Search range menu PART SRCH ON off MKRΔ→SEARCH RNG MKR→LEFT RNG MKR→RIGHT RNG RETURN | PARS {OFF ON 0 1} SEARSTR SEARSTRL SEARSTRR | CALCulate:EVALuate:BAND:FULL[:STATe] {OFF ON 0 1} CALCulate:EVALuate:BAND:SPAN DMARKer CALCulate:EVALuate:BAND:START MARKer CALCulate:EVALuate:BAND:STOP MARKer |

¹ Network and impedance analyzer only

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|--|---|--|
| Utility Network/Impedance Analyzer MKR LIST ON off STATISTICS ON off MKR TIME ON off SMTH/POLAR MENU → See Circle data menu | MKRL {OFF ON 0 1} MEASTAT {OFF ON 0 1} OUTPMSTA? MKRTIME {OFF ON 0 1} | DISPLAY[:WINDOW]:TEXT16:STATE {OFF ON 0 1} CALCulate:EVALuate:MStatistics[:STATE] {OFF ON 0 1} CALCulate:EVALuate:MStatistics:DATA? DISPLAY[:WINDOW]:TRACE:MARKer:UNIT:TIME {OFF ON 0 1} |
| Circle data menu REAL IMAG LIN MAG PHASE LOG MAG PHASE R+jX G+jB SWR PHASE RETURN | CIRF RI CIRF LIN CIRF LOG CIRF RX CIRF GB CIRF SWR | CALCulate:EVALuate:R:FORMat RIMaginary CALCulate:EVALuate:R:FORMat MLIPhase CALCulate:EVALuate:R:FORMat MLOPhase CALCulate:EVALuate:R:FORMat RX CALCulate:EVALuate:R:FORMat GB CALCulate:EVALuate:R:FORMat SWRPhase |
| Spectrum Analyzer MKR LIST ON off STATISTICS ON off MKR TIME ON off NOISE FORM ON off | MKRL {OFF ON 0 1} MEASTAT {OFF ON 0 1} OUTPMSTA? MKRTIME {OFF ON 0 1} MKRNOI {OFF ON 0 1} | DISPLAY[:WINDOW]:TEXT16:STATE {OFF ON 0 1} CALCulate:EVALuate:MStatistics[:STATE] {OFF ON 0 1} CALCulate:EVALuate:MStatistics:DATA? DISPLAY[:WINDOW]:TRACE:MARKer:UNIT:TIME {OFF ON 0 1} CALCulate:EVALuate:NOISE[:STATE] {OFF ON 0 1} |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|-----------------------|----------------|---|
| System | | |
| IBASIC | | |
| Step | | PROGRAM[:SElected]:EXECute "STEP" |
| Continue | | PROGRAM[:SElected]:STATe CONTINUE or PROGRAM[:SElected]:EXECute "CONT" |
| Run | | PROGRAM[:SElected]:STATe RUN or PROGRAM[:SElected]:EXECute "RUN" |
| Pause | | PROGRAM[:SElected]:STATe PAUSE or PROGRAM[:SElected]:EXECute "PAUSE" |
| Stop | | PROGRAM[:SElected]:STATe STOP or PROGRAM[:SElected]:EXECute "STOP" |
| Edit | | PROGRAM[:SElected]:EXECute "EDIT" |
| ASSIGN @Hp4396 | | |
| OUTPUT @Hp4396 | | |
| ENTER @Hp4396 | | |
| END | | |
| GOTO LINE | | |
| RECALL LINE | | |
| END EDIT | | |
| ON KEY LABELS | | |
| [USER DEFINE] | | |
| [USER DEFINE] | | |
| [USER DEFINE] | | |
| [USER DEFINE] | | |
| [USER DEFINE] | | |
| [USER DEFINE] | | |
| [USER DEFINE] | | |
| [USER DEFINE] | | |
| [USER DEFINE] | | |
| [USER DEFINE] | | |
| [USER DEFINE] | | |
| CAT | | |
| SAVE | | |
| RE-SAVE | | |
| GET | | |
| PURGE | | |
| INITIALIZE | | |
| MSI [INTERNAL/MEMORY] | | |
| SCRATCH | | |
| RENumber | | |
| LIST | | |
| COMMAND ENTRY | | |
| SELECT LETTER | | |
| SPACE | | |
| BACK SPACE | | |
| ERASE TITLE | | |
| DONE | | |
| CANCEL | | |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|-------------------|----------------|-------------------------|
| CLEAR I/O | | |
| RESET | | |
| MEMORY PARTITION | | |
| mmK RAM nnK BASIC | | |
| mmK RAM nnK BASIC | | |
| mmK RAM nnK BASIC | | |
| mmK RAM nnK BASIC | | |
| mmK RAM nnK BASIC | | |
| DONE | | |
| CHANGE YES | | |
| NO | | |
| CANCEL | | |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|----------------------|-----------------------|---|
| SET CLOCK | | |
| TIME HH:MM:SS | | |
| HOUR | | |
| MIN | | |
| SEC | | |
| ENTER | SETCTIME <h,m,s> | SYSTem:TIME <hour>,<minute>,<second> |
| CANCEL | | |
| DATE DD/MM/YY | | |
| MON | | |
| DAY | | |
| YEAR | | |
| ENTER | SETCDATE <y,m,d> | SYSTem:DATE <year>,<month>,<day> |
| CANCEL | | |
| DATE MODE:MonDayYear | MONDYEAR | SYSTem:DATE:MODE MDY |
| DayMonYear | DAYMYEAR | SYSTem:DATE:MODE DMY |
| RETURN | | |
| BEEPER MENU | | |
| BEEP DONE ON off | BEEPDONE {OFF ON 0 1} | SYSTem:BEEPer1:STATe {OFF ON 0 1} |
| BEEP WARN ON off | BEEPWARN {OFF ON 0 1} | SYSTem:BEEPer2:STATe {OFF ON 0 1} |
| RETURN | | |
| LIMIT MENU | | |
| LIMIT LINE ON off | LIMLINE {OFF ON 0 1} | CALCulate:LIMit:LINE {OFF ON 0 1} |
| LIMIT TEST ON off | LIMITEST {OFF ON 0 1} | CALCulate:LIMit:STATe {OFF ON 0 1} |
| BEEP FAIL ON off | BEEPFAIL {OFF ON 0 1} | CALCulate:LIMit:BEEPer[:STATe] {OFF ON 0 1} |
| EDIT LIMIT LINE | EDITLIML | |
| SEGMENT | LIMSEDI <numeric> | CALCulate:LIMit:SEGment <numeric> |
| EDIT | LIMSEDI [<numeric>] | CALCulate:LIMit:SEGment:EDIT |
| DELETE | LIMSDEL | CALCulate:LIMit:SEGment:DELeTe |
| ADD | LIMSADD | CALCulate:LIMit:SEGment:ADD |
| CLEAR LIST | LIMCLEL | CALCulate:LIMit:CLEar |
| CLEAR LIST YES | | |
| NO | | |
| DONE | LIMEDONE | CALCulate:LIMit:SAVE |
| LIMIT LINE OFFSETS | | |
| SWP PARAM OFFSET | LIMPRMO <numeric> | CALCulate:LIMit:CONTRol:OFFSet <numeric> |
| AMPLITUDE OFFSET | LIMIAMPO <numeric> | CALCulate:LIMit:OFFSet <numeric> |
| MKR→AMP.OFS. | MKRAMPO | CALCulate:LIMit:OFFSet MARKer |
| RETURN | | |
| RETURN | | |
| SERVICE MENU | | |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|-----------------------|--------------------|--|
| Limit line entry menu | | |
| SWP PARAM | LIMPRM <numeric> | CALCulate:LIMit:SEGment:CONTRol[:DATA] <numeric> |
| MKR→SWP PARAM | MKRSWPRM | CALCulate:LIMit:SEGment:CONTRol[:DATA] MARKer |
| UPPER LIMIT | LIMU <numeric> | CALCulate:LIMit:SEGment:UPPer <numeric> |
| LOWER LIMIT | LIML <numeric> | CALCulate:LIMit:SEGment:LOWer <numeric> |
| DELTA LIMIT | LIMD <numeric> | CALCulate:LIMit:SEGment:DELTA <numeric> |
| MIDDLE VALUE | LIMM <numeric> | CALCulate:LIMit:SEGment:MIDDle <numeric> |
| MKR→MIDDLE | MKRMIDD | CALCulate:LIMit:SEGment:MIDDle MARKer |
| DONE | LIMSDON | CALCulate:LIMit:SEGment:SAVE |
| Local | | |
| SYSTEM CONTROLLER | | |
| ADDRESSABLE ONLY | | |
| SET ADDRESSES | | |
| ADDRESS:4396 | | |
| ADDRESS:CONTROLLER | ADDRCONT <numeric> | SYSTem:COMMunicate:GPIB2:ADDRes <numeric> |
| RETURN | | |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|--|--|--|
| <p>Copy</p> <p>PRINT [STANDARD]</p> <p>COPY ABORT</p> <p>COPY SKEY on OFF</p> <p>COPY TIME ON off</p> <p>PRINT SETUP → See Print setup menu</p> <p>ORIENT [PORTRAIT]</p> <p>FORM FEED ON off</p> <p>MORE → See Copy more menu</p> | <p>PRIN ALL</p> <p>COPA</p> <p>PRSOFT {OFF ON 0 1}</p> <p>COPT {OFF ON 0 1}</p> <p>LANDSCAPE {OFF ON 0 1}</p> <p>FORMFEED {OFF ON 0 1}</p> | <p>HCOPY[:IMMediate]</p> <p>HCOPY:ABORt</p> <p>HCOPY:DRIVER:SKEY {OFF ON 0 1}</p> <p>HCOPY:ITEM:TDS'amp:STATe {OFF ON 0 1}</p> <p>HCOPY:DRIVER:LANDScape {OFF ON 0 1}</p> <p>HCOPY:DRIVER:FORMFeed {OFF ON 0 1}</p> |
| <p>Copy more menu</p> <p>Network Analyzer</p> <p>LIST VALUES</p> <p>OPERATING PARAMETERS</p> <p>CAL KIT DEFINITION → See Copy cal kit menu</p> <p>LIST SWEEP TABLE → See Copy list sweep menu</p> <p>LIMIT TEST TABLE → See Copy limit test menu</p> <p>RETURN</p> <p>Spectrum Analyzer</p> <p>LIST VALUES</p> <p>OPERATING PARAMETERS</p> <p>LIST SWEEP TABLE → See Copy list sweep menu</p> <p>LIMIT TEST TABLE → See Copy limit test menu</p> <p>RETURN</p> <p>Impedance Analyzer</p> <p>LIST VALUES</p> <p>OPERATING PARAMETERS</p> <p>CAL KIT DEFINITION → See Copy cal kit menu</p> <p>COMPEN KIT DEFINITION</p> <p>LIST SWEEP TABLE → See Copy list sweep menu</p> <p>LIMIT TEST TABLE → See Copy limit test menu</p> <p>RETURN</p> | <p>LISV</p> <p>OPEP</p> <p>LISV</p> <p>OPEP</p> <p>LISV</p> <p>OPEP</p> <p>COMS</p> | <p>DISPlay[:WINDow]:TEXT1:PAGE 1 DISPlay[:WINDow]:TEXT1:STATe {ON 1}</p> <p>DISPlay[:WINDow]:TEXT2:PAGE 1 DISPlay[:WINDow]:TEXT2:STATe {ON 1}</p> <p>DISPlay[:WINDow]:TEXT1:PAGE 1 DISPlay[:WINDow]:TEXT1:STATe {ON 1}</p> <p>DISPlay[:WINDow]:TEXT2:PAGE 1 DISPlay[:WINDow]:TEXT2:STATe {ON 1}</p> <p>DISPlay[:WINDow]:TEXT1:PAGE 1 DISPlay[:WINDow]:TEXT1:STATe {ON 1}</p> <p>DISPlay[:WINDow]:TEXT2:PAGE 1 DISPlay[:WINDow]:TEXT2:STATe {ON 1}</p> <p>DISPlay[:WINDow]:TEXT20:PAGE 1 DISPlay[:WINDow]:TEXT20:STATe ON</p> |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|--|--|--|
| Copy cal kit menu STANDARD DEFINITION → See Copy standard number menu CLASS ASSIGNMENT RETURN | CALCASSI | DISPlay[:WINDow]:TEXT3:PAGE 1 DISPlay[:WINDow]:TEXT3:STATe {ON 1} |
| Copy limit test menu DISPLAY LIST DISP MODE: UPR & LWR MID & DLT RETURN | DISLLIST DISMAMP UL DISMAMP MD DISMAMP? | DISPlay[:WINDow]:TEXT{14 15}:PAGE 1 DISPlay[:WINDow]:TEXT{14 15}:STATe {ON 1} |
| Copy list sweep menu DISPLAY LIST DISP MODE: ST & SP CTR & SPAN RETURN | DISL DISMPRM STSP DISMPRM CTSP DISMPRM? | DISPlay[:WINDow]:TEXT{12 13}:PAGE 1 DISPlay[:WINDow]:TEXT{12 13}:STATe {ON 1} |
| Copy standard number menu → See Screen menu STD NO.1 STD NO.2 STD NO.3 STD NO.4 STD NO.5 STD NO.6 STD NO.7 STD NO.8 | CALS 1 CALS 2 CALS 3 CALS 4 CALS 5 CALS 6 CALS 7 CALS 8 | DISPlay[:WINDow]:TEXT4:PAGE 1 DISPlay[:WINDow]:TEXT4:STATe {ON 1} DISPlay[:WINDow]:TEXT5:PAGE 1 DISPlay[:WINDow]:TEXT5:STATe {ON 1} DISPlay[:WINDow]:TEXT6:PAGE 1 DISPlay[:WINDow]:TEXT6:STATe {ON 1} DISPlay[:WINDow]:TEXT7:PAGE 1 DISPlay[:WINDow]:TEXT7:STATe {ON 1} DISPlay[:WINDow]:TEXT8:PAGE 1 DISPlay[:WINDow]:TEXT8:STATe {ON 1} DISPlay[:WINDow]:TEXT9:PAGE 1 DISPlay[:WINDow]:TEXT9:STATe {ON 1} DISPlay[:WINDow]:TEXT10:PAGE 1 DISPlay[:WINDow]:TEXT10:STATe {ON 1} DISPlay[:WINDow]:TEXT11:PAGE 1 DISPlay[:WINDow]:TEXT11:STATe {ON 1} |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---|---|--|
| Print setup menu PRINT STANDARD COLOR PRINT COLOR [FIXED] DPI TOP MARGIN LEFT MARGIN DEFAULT SETUP RETURN | PRIS PRIC PRICFIXE PRICVARI DPI <numeric> TMARG <numeric> LMARG <numeric> DFLT | HCOPY:DRIVER:COLor {OFF 0} HCOPY:DRIVER:COLor {ON 1} HCOPY:DRIVER:CMAP:COLor FIXed HCOPY:DRIVER:CMAP:COLor VARIable HCOPY:DRIVER:DPI <numeric> HCOPY:DRIVER:TOPMarg <numeric> HCOPY:DRIVER:LEFTMarg <numeric> HCOPY:DEFault |
| Screen menu PRINT [STANDARD] COPY ABORT COPY TIME ON off PRINT SETUP → See Print setup menu NEXT PAGE PREV PAGE RESTORE DISPLAY | PRIN ALL COPA COPT {OFF ON 0 1} NEXP PREP RESD | HCOPY:DRIVER:LANGUage PCL HCOPY[:IMMedia te] HCOPY:ABORt HCOPY:ITEM:TDSamp:STATe {OFF ON 0 1} DISPlay[:WINDow]:TEXT{1-17}:PAGE UP DISPlay[:WINDow]:TEXT{1-17}:PAGE DOWN DISPlay[:WINDow]:TEXT{1-17}:STATe {OFF 0} |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---|--|---|
| Save | | |
| STATE | SAVDSTA <string> | MMEMory:STORe:STATe <string(file name)>[,<string(msus)>] |
| DATA ONLY | | |
| SAVE BINARY | SAVDDAT <string> | MMEMory:STORe:TRACe SEL,<string(file name)>[,<string(msus)>] |
| SAVE ASCII | SAVDASC <string> | MMEMory:STORe:DINTerchange:TRACe SEL,<string(file name)>[,<string(msus)>] |
| DEFINE SAVE DATA → See Define save data menu | | |
| STOR DEV [DISK] | STOD {DISK MEMO} | |
| GRAPHICS | SAVDTIF <string> | MMEMory:STORe:DINTerchange:TIFf <string(file name)>[,<string(msus)>] |
| 4396A STATE | SAVDSTAC <string> | |
| RE-SAVE FILE | RESAVD <string> | MMEMory:DELeTe <file_name>[,<msus>] MMEMory:STORe:{STATe TRACe} [SEL,]<file_name>[,<msus>] |
| file name | | |
| file name | | |
| file name | | |
| file name | | |
| PREV FILES | | |
| NEXT FILES | | |
| RETURN | | |
| FILE UTILITIES | | |
| PURGE FILE | PURG <string> | MMEMory:DELeTe <file_name>[,<msus>] |
| file name | | |
| file name | | |
| file name | | |
| file name | | |
| PREV FILES | | |
| NEXT FILES → See purge yes no menu | | |
| STOR DEV [DISK] | STOD {DISK MEMO} | |
| CREATE DIRECTORY | CRED <string> | MMEMory:CREate:DIRectory <string> |
| CHANGE DIRECTORY | CHAD <string> | MMEMory:CDIRectory [<string>] |
| COPY FILE | FILC <string>,<string>,<string>,<string> | MMEMory:COpy <string(s)>,<msus>,<string(d)>,<msus> |
| file name | | |
| file name | | |
| file name | | |
| file name | | |
| PREV FILES | | |
| NEXT FILES | | |
| STOR DEV [DISK] | STOD {DISK MEMO} | |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---|--|--|
| INITIALIZE DISK INITIALIZE: YES NO FORMAT [LIF] STOR DEV [] RETURN STOR DEV [] | INID DISF LIF DISF DOS STOD{DISK MEMO} STOD{DISK MEMO} | MMEmory:INITiaLize <msus>,{LIF DOS} |
| Define save data menu RAW ON off CAL ON off DATA ON off MEM ON off DATA TRACE ON off MEM TRACE ON off RETURN | SAVRAW {ON 1} SAVRAW {OFF 0} SAVCAL {ON 1} SAVCAL {OFF 0} SAVDAT {ON 1} SAVDAT {OFF 0} SAVMEM {ON 1} SAVMEM {OFF 0} SAVDTRC {ON 1} SAVDTRC {OFF 0} SAVMTRC {ON 1} SAVMTRC {OFF 0} | MMEmory:STORe:ITEM:TRACe:SElect RAW MMEmory:STORe:ITEM:TRACe:DELeTe RAW MMEmory:STORe:ITEM:TRACe:SElect CCO MMEmory:STORe:ITEM:TRACe:DELeTe CCO MMEmory:STORe:ITEM:TRACe:SElect DATA MMEmory:STORe:ITEM:TRACe:DELeTe DATA MMEmory:STORe:ITEM:TRACe:SElect MEM MMEmory:STORe:ITEM:TRACe:DELeTe MEM MMEmory:STORe:ITEM:TRACe:SElect DTR MMEmory:STORe:ITEM:TRACe:DELeTe DTR MMEmory:STORe:ITEM:TRACe:SElect MTR MMEmory:STORe:ITEM:TRACe:DELeTe MTR |
| Purge yes no menu PURGE: YES NO | | |
| Recall file name file name file name file name PREV FILES NEXT FILES STOR DEV [] | RECD <string> STOD{DISK MEMO} | MMEmory:LOAD:STAtE <file_name>[,<msus>] (State) MMEmory:LOAD:TRACe SEL,<file_name>[,<msus>] (Data) |
| Preset | PRES | SYSTem:PRESet |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|--|--|--|
| GPIB only commands Marker related commands | MKR {ON 1} MKRPRM <numeric> OUTPMKR? MKRVAL? MKRAUV? MKRP <numeric> SMKRP{1-7} <numeric> | DISPlay[:WINDow]:TRACe:MARKer:STATe {ON 1} CALCulate:EVALuate:Y:XPOStion <numeric> CALCulate:EVALuate:Y:DATA? CALCulate:EVALuate:Y:VALue1? CALCulate:EVALuate:Y:VALue2? CALCulate:EVALuate:Y:XPOStion:POINt <numeric> CALCulate:EVALuate:Y{2-8}:XPOStion:POINt <numeric> |
| 8-bit IO related commands | INP8IO? OUT8IO <numeric> | SYSTem:COMMunicate:PARAllel[:RECeive]:DATA? SYSTem:COMMunicate:PARAllel:TRANsmit:DATA <numeric> |
| KEY related commands | KEY <NUMERIC> ENKEY DSKEY USKEY | SYSTem:KEY <numeric> SYSTem:KLOCK {OFF 0} SYSTem:KLOCK {ON 1} |
| Input/output data array related commands | INPURAW1 INPURAW2 INPURAW3 INPURAW4 INPUDATA INPUDTRC OUTPRAW1? OUTPRAW2? OUTPRAW3? OUTPRAW4? OUTPDATA? OUTPDATAP? <numeric> OUTPMEMO? OUTPMEMOP? <numeric> OUTPDTRC? OUTPDTRCP? <numeric> OUTPMTRC? OUTPMTRCP? <numeric> OUTPSWPRM? OUTPSWPRMP? <numeric> | DATA[:DATA] RAW1,{<block> <numeric>[,<numeric>]} DATA[:DATA] RAW2,{<block> <numeric>[,<numeric>]} DATA[:DATA] RAW3,{<block> <numeric>[,<numeric>]} DATA[:DATA] RAW4,{<block> <numeric>[,<numeric>]} DATA[:DATA] DATA,{<block> <numeric>[,<numeric>]} TRACe[:DATA] DTR,{<block> <numeric>[,<numeric>]} DATA[:DATA]? RAW1 DATA[:DATA]? RAW2 DATA[:DATA]? RAW3 DATA[:DATA]? RAW4 DATA[:DATA]? DATA DATA[:DATA]:VALue? DATA,<point> DATA[:DATA]? MEM DATA[:DATA]:VALue? MEM,<point> TRACe[:DATA]? DTR TRACe[:DATA]:VALue? DTR,<point> TRACe[:DATA]? MTR TRACe[:DATA]:VALue? MTR,<point> DATA[:DATA]? SPAR DATA[:DATA]:VALue? SPAR,<point> |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|---|--|--|
| Calibration and compensation related commands | INPUCALC{1-12} <numeric> OUTPCALC{1-12}? INPUCALK <numeric> OUTPCALK? SAVC INPUCOMC{1-3} OUTPCOMC{1-3}? | DATA[:DATA] CCO{1-12},{<block> <numeric>[,<numeric>]} DATA[:DATA]? CCO{1-12} DATA[:DATA] CKIT,{<block> <numeric>[,<numeric>]} DATA[:DATA]? CKIT SENSE:CORRection:COLLect:SAVE9 DATA[:DATA] CMP{1-3},{<block> <numeric>[,<numeric>]} DATA[:DATA]? CMP{1-3} |
| Transfer data format related commands | FORM2 FORM3 FORM4 FORM5 | FORMat[:DATA] REAL,32 FORMat[:DATA] REAL,64 FORMat[:DATA] ASCii FORMat[:DATA] PACKed,32 |
| Limit test related commands | OUTPLIMF? OUTPFAIP? OUTPLIMM? | DATA[:DATA]? LFA DATA:POINts? LFA DATA[:DATA]? LMAR |
| Error related command | OUTPERRO? | SYSTem:ERRor? |
| Status byte related commands | CLES ESB? ESNB <numeric> OSR? OSE <numeric> OSER? OSPT <numeric> OSNT <numeric> | *CLS (Common Command) STATus:INSTrument[:EVENT]? STATus:INSTrument:ENABle <numeric> STATus:OPERation:CONDition? STATus:OPERation:ENABle <numeric> STATus:OPERation[:EVENT]? STATus:OPERation:PTRansition <numeric> STATus:OPERation:NTRansition <numeric> |
| Test set related command | TESS? | SYSTem:COMMunicate:TSET? |
| LCD display related command | SCRn {OFF ON 0 1} BLIGHT {OFF ON 0 1} | |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|--|---|-------------------------|
| Waveform analysis commands | | |
| Waveform analysis setup commands | ANAOCH1 ANAOCH2 ANARANG ANARFULL ANAODATA ANAOMEMO THRR | |
| Maximum/Minimum/Mean search commands | OUTPMAX? OUTPMIN? OUTPMINMAX? OUTPMEAN? PEAK? NEXPK? NUMLMAX? NUMLMIN? LMAX? LMIN? TARR? TARL? | |
| Ripple analysis commands | RPLPP? RPLHEI? RPLRHEI? RPLLHEI? RPLENV? RPLMEA? RPLVAL? POLE? | |
| Filter and Resonator analysis commands | OUTPFILT? OUTPXFIL? OUTPCFIL? OUTPRESO? OUTPRESR? OUTPRESF? OUTPCERR? | |
| Equivalent circuit analysis commands | EQUCPARA? EQU EQUCPARS? EQU0? EQUCPARS4? | |

| Front Panel Key | Simple Command | Equivalent SCPI Command |
|------------------------------------|---|-------------------------|
| 3-term calibration commands | INPUOPEA INPUSHOA INPULOAA INPUD | |
| File transfer commands | CLOSE CWD? FNAME? <numeric> FNUM? FSIZE? <string> READ? ROPEM <string> WOPEN <string>[,<numeric>] WRITE <block> | |
| Common commands | *CLS *IDN? *OPT? *RST *TST? *OPC *WAI *CLS *ESE <numeric> *ESR? *SRE <numeric> *STB? *TRG *PCB <numeric> | |

SCPI Commands Summary

This appendix summarizes the SCPI commands in alphabetical order. It also shows the equivalent simple command when applicable.

| Command | Parameter | Note | Equivalent Simple Command |
|----------------------|---------------------|--------------|---------------------------|
| ABORt | | | HOLD |
| CALCulate | | | |
| :AVERAge | | | |
| :STATe | {OFF ON 0 1} | | DHOLD |
| :TYPE | {MAXimum MINimum} | | DHOLD |
| :EVALuate | | | |
| :BAND | | | |
| :FULL | | | |
| [:STATe] | {OFF ON} | | PARS |
| :SPAN | DMARker | [no query] | SEARSTR |
| :STARt | MARKer | [no query] | SEARSTRL |
| :STOP | MARKer | [no query] | SEARSTRR |
| :COUPlE | {OFF ON 0 1} | | MKRCOUP |
| :EFFect | | | |
| :ON | {1 2} | | CRSC |
| :EPARameters | | CALECPARA | |
| :CIRCuit {A B C D E} | | EQUC | |
| :SIMulation | | SIMFCHAR | |
| :INTerpolate | {OFF ON 0 1} | | MKRCONT |
| :MSTATistics | | | |
| :DATA? | | [query only] | OUTPMSTA? |
| [:STATe] | {OFF ON 0 1} | | MEASTAT |
| :NOISe | | | |
| [:STATe] | {OFF ON 0 1} | | MKRNOI |
| :ON | {"DTR" "MTR"} | | MKRO |
| :PEAK | | | |
| :EXCursion | | | |
| :X | {<numeric> DMARker} | | MKRPKD, PKDLTX |
| [:Y] | {<numeric> DMARker} | | MKRPKD, PKDLTY |
| :POLarity | {POSitive NEGative} | | PKPOL |
| :THReshold | {<numeric> MARKer} | | MKRTHRE, PKTHVAL |
| :STATe | {OFF ON 0 1} | | PKTHRE |

| Command | Parameter | Note | Equivalent Simple Command |
|-----------------------|---|--------------|---------------------------|
| CALCulate (continued) | | | |
| :EVALuate (continued) | | | |
| :REFerence | | | |
| :DATA? | | [query only] | OUTPDMKR? |
| :X | <numeric> | | DMKRPRM |
| :Y1 | <numeric> | | DMKRVAL |
| :Y2 | <numeric> | | DMKRAUV |
| :WIDTh | | | |
| :DATA? | | [query only] | OUTPMWID? |
| :STATe | {OFF ON 0 1} | | WIDT |
| :XPOSITion | | | |
| :IN | | [no query] | WIDSIN |
| :OUT | | [no query] | WIDSOUT |
| :Y | <numeric> | | WIDV |
| :TYPE | {DIVS2 MULS2 DIV2 FIXed} | | WIDVTYPE |
| :Y[1] | | | |
| :DATA? | | [query only] | OUTPMKR? |
| :VALue1? | | [query only] | MKRVAL? |
| :VALue2? | | [query only] | MKRAUV? |
| :XPOSITion | <numeric> | | MKRPRM |
| :LPEak | | [no query] | SEANPKL |
| :LTARget | | [no query] | SEAL |
| :MAXimum | | [no query] | SEAM |
| :MINimum | | [no query] | SEAM |
| :NPEak | | [no query] | SEANPK |
| :PALL | | [no query] | SEAM |
| :PEAK | | [no query] | SEAM |
| :PLEFt | | [no query] | SEAM |
| :POINT | <numeric> | | MKRP |
| :PRIGHt | | [no query] | SEAM |
| :RPEak | | [no query] | SEANPKR |
| :RTARget | | [no query] | SEAR |
| :TARGet | <numeric> | | SEAM, SEATARG |
| :TRACK | {MAXimum MINimum TARGet PEAK PALL PLEFt PRIGHt OFF} | | TRACK |
| :Y{2-8} | | | |
| :DATA? | | [query only] | OUTPSMKR{1-7}? |
| :VALue1? | | [query only] | SMKRVAL{1-7}? |
| :VALue2? | | [query only] | SMKRAUV{1-7}? |
| :XPOSITion | <numeric> | | SMKRPRM{1-7} |
| :POINT | <numeric> | | SMKRP{1-7} |

| Command | Parameter | Note | Equivalent Simple Command |
|-----------------------|---|--------------|---------------------------|
| CALCulate (continued) | | | |
| :FORMat | {GDElay REAL IMAGinary MLINear MLOGarithmic PHASe UPHase SWR COMPLex} | | FMT, SAUNIT |
| :FORMat | {RIMaginary MLIPhase MLOPhase RX GB SWRPhase MLINear PHASe UPHase REAL IMAGinary CP SC LP LS D Q RP RS COMPLex} | | FMT, SAUNIT, CIRF, MEAS |
| :UNIT | | | |
| :ANGLE | {DEG RAD} | | PHAU |
| :GDAPerture | | | |
| :APERture | <numeric> | | GRODAPER |
| :LIMit | | | |
| :BEEPer | | | |
| [:STATe] | {OFF ON 0 1} | | BEEPFail |
| :CLEar | | [no query] | LIMCLEL |
| :CONTRol | | | |
| :OFFSet | <numeric> | | LIMIPRMO, MKRAMPO |
| :LINE | {OFF ON 0 1} | | LIMLINE |
| :OFFSet | {<numeric> MARKer} | | LIMIAMPO |
| :SAVE | | [no query] | LIMEDONE |
| :SEGment | <numeric> | | LIMSEDI |
| :ADD | | [no query] | LIMSADD |
| :CONTRol | | | |
| [:DATA] | {<numeric> MARKer} | | LIMPRM, MKRSWPRM |
| :DELete | | [no query] | LIMSDEL |
| :DELTA | <numeric> | | LIMD |
| :EDIT | | [no query] | LIMSEDI |
| :LOWer | <numeric> | | LIML |
| :MIDDLE | {<numeric> MARKer} | | LIMM, MKRMIDD |
| :SAVE | | [no query] | LIMSDON |
| :UPPer | <numeric> | | LIMU |
| :STATe | {OFF ON 0 1} | | LIMITEST |
| :MATH1 | | | |
| [:EXPReSSion] | | | |
| :CATalog? | | [query only] | (None) |
| :NAME | {OFF YREF YTRA ZREF ZTRA INVS MP4 MP8 MP16} | | CONV |
| :NAME | {OFF YREF YTRA ZREF ZTRA INVS MP4 MP8 MP16} | (NA) | CONV |
| | {OFF IMPedance ADMittance} | (ZA) | MEAS |

| Command | Parameter | Note | Equivalent Simple Command |
|----------------------|--|------------|---------------------------|
| DIAG (continued) | | | |
| :SERVice (continued) | | | |
| :IF | | | |
| :ADMX | | | |
| :MODE | {AUTO ALTErnate DEGO DEG90} | | (None) |
| :BPFilter | | | |
| :MODE | {AUTO BW3M BW1M XTAL} | | (None) |
| :GAIN | | | |
| :MODE | {AUTO MANual} | | (None) |
| :W | {AUTO DB0 DB10} | | (None) |
| :X | {AUTO DB0 DB18} | | (None) |
| :Y | {AUTO DB0 DB6 DB12 DB18} | | (None) |
| :Z | {AUTO DB0 DB2 DB4 DB18} | | (None) |
| :LPFilter | | | |
| :MODE | {AUTO BW5K BW15K BW50K BW150K THROUGH} | | (None) |
| :RANGe | | | |
| :F | {HIGH LOW} | | (None) |
| :MODE | {AUTO MANual} | [no query] | (None) |
| :R | {HIGH LOW} | | (None) |
| :SHBW | | | |
| :MODE | {AUTO NARRow MIDDLE WIDE} | | (None) |
| :TLOCAl | | | |
| :MODE | {AUTO AC DC} | | (None) |
| :MODE | ON | | (None) |
| :SOURce | | | |
| :ALCLoop | {OPEN CLOSE} | | (None) |
| :ATTenuator | {AUTO DB0 DB10 DB20 DB30 DB40 DB50 DB60} | | (None) |
| :GAIN | | | |
| :DAC | | | |
| :MODE | {AUTO MANual} | | (None) |
| :VALue | <numeric> | | (None) |
| :LEVel | | | |
| :DAC | | | |
| :MODE | {AUTO MANual} | | (None) |
| :VALue | <numeric> | | (None) |
| :MODE | {AUTO MANual} | | (None) |

| Command | Parameter | Note | Equivalent Simple Command |
|----------------------|--------------------------|--------------|---------------------------|
| DIAG (continued) | | | |
| :SERVice (continued) | | | |
| :SYNThesizer | | | |
| :FLOCAl | | | |
| :MODE | {AUTO SINGle TRIPle} | | (None) |
| :FREQuency:OFFSet | <numeric> | | |
| :FN | | | |
| :MODE | {AUTO NARRow WIDE} | | (None) |
| :STEP | | | |
| :DAC | | | |
| :MODE | {AUTO MANual} | | (None) |
| :VALue | <numeric> | | (None) |
| :LOOP | {OPEN CLOSe} | | (None) |
| :MODE | {AUTO MANual} | | (None) |
| :OUTPut | {OFF ON 0 1} | | (None) |
| :POLarity | {AUTO POSitive NEGative} | | (None) |
| :TEST | <numeric> | [no query] | (None) |
| :EXECute | | [no query] | (None) |
| :CONTinue | | [no query] | (None) |
| :RESult? | <numeric> | [query only] | (None) |
| DISPlay | | | |
| :ANNotation | | | |
| :BACKlight | {OFF ON 0 1} | | BLIGHT |
| :FREQuency | OFF | | FREO |
| :BRIGHtness | <numeric> | | INTE |
| :CMAP | | | |
| :COLor{1-14} | | | |
| :DEFault | | [no query] | RSCO |
| :HSL | <hue>,<sat>,<lum> | | COLO |
| :DEFault | | | DEFC |
| :LOAD | | [no query] | RECC |
| :STORe | | [no query] | SVCO |
| :CONTrast | <numeric> | | BACI |

| Command | Parameter | Note | Equivalent Simple Command |
|---|---|------------|--|
| DISPlay (continued) | | | |
| [:WINDow] | | | |
| :ALLocation | {INSTRument HIHB BASic BSTatus} | | DISA |
| :FORMat | {FBACK ULOWer} | | SPLD |
| :TEXT{1-17} | | | |
| [:DATA] | <title_string> (only for TEXT17) | | TITL |
| :PAGE | {UP DOWN <numeric>} (except TEXT16 and TEXT17) | | CALCASSI, CALS, DISL, DISLLIST, LISV, NEXP, OPEP, PREP |
| :STATe | {OFF ON 0 1} (except TEXT17) | | CALCASSI, CALS, DISL, DISLLIST, LISV, MKRL, OPEP, RESD |
| :TEXT18 (18: Equivalent circuit parameters) | | | |
| :STATe | {OFF ON 0 1} | | DESECPARA |
| :TEXT19 (19: Equivalent circuit model) | | | |
| :STATe | {OFF ON 0 1} | | DISECIRC |
| :TEXT20 (20: Fixture compensation kit definition table) | | | |
| :STATe | {OFF ON 0 1} | | COMS |
| :TRACe{[1] 2} (1: Data Trace, 2: Memory Trace) | | | |
| :GRATticule | | | |
| :FORMat | {RECTangle POLar SMITH ADMittance} | (NA) | FMT |
| :FORMat | {RECTangle POLar SMITH ADMittance CPLane} | (ZA) | FMT |
| :MARKer{[1] 2-8} (1: Main Marker, 2-8: Sub Marker) | | | |
| :ALL | DEFault | [no query] | PRSMKRS |
| :STATe | {OFF ON 0 1} (MARKer{2-8} only) | | MKR |
| :RELative | {OFF ON 0 1} | | DMKR |
| :REFerence | {MARKer FIXed TRACked} | | DMKR |
| :STATe | {OFF ON 0 1} (Only for MARKer{2-8}) | | SMKR{1-7} |
| :UNIT | | | |
| :TIME | {OFF ON 0 1} | | |
| :X | | | |
| [:SCALe] | | | |
| :RLEVel | <numeric> | | REFX |

| Command | Parameter | Note | Equivalent Simple Command |
|---|--|------------|---|
| :Y [:SCALe] :AUTO :BOTTom :COUPle :PDIVision :RLEVel :RPOStion :TOP [:SCALe] :SPACing [:SCALe] :STATe | ONCE <numeric> {OFF ON 0 1} <numeric> {<numeric> MARKer} <numeric> <numeric> {LINear LOGarithmic} {OFF ON 0 1} | [no query] | AUTO BOTV SCAC SCAL REFV, REFY, MKRREF REFP TOPV FMT |
| FORMat [:DATA] | {ASCIi REAL,32 REAL,64 PACKed,32} | | FORM2, FORM3, FORM4, FORM5 |

| Command | Parameter | Note | Equivalent Simple Command |
|--------------|------------------|------------|---------------------------|
| HCOPY | | | |
| :ABORT | | [No query] | COPA |
| :DEFAULT | | [No query] | DFLT |
| :DRIVER | | | |
| :CMAP | | | |
| :COLOR | {FIXED VARIABLE} | | PRICFIXE, PRICVARI |
| :COLOR | {OFF ON 0 1} | | PRIC, PRIS |
| :DPI | <numeric> | | DPI |
| :FORMFEED | {OFF ON 0 1} | | FORMFEED |
| :LANDSCAPE | {OFF ON 0 1} | | LANDSCAPE |
| :LEFTMARG | <numeric> | | LMARG |
| :SKEY | {OFF ON 0 1} | | PRSOFT |
| :TOPMARG | <numeric> | | TMARG |
| :ITEM | | | |
| :TDSAMP | | | |
| :STATE | {OFF ON 0 1} | | COPT |
| INITIATE | | | |
| [:IMMEDIATE] | | [no query] | NUMG, SING |
| :AGAIN | | | |
| :ALL | | [no query] | REST |
| :CONTINUOUS | {OFF ON 0 1} | | CONT, HOLD, NUMG, SING |
| INPUT | | | |
| :IMPEDANCE | <numeric> | | INPZ |
| INSTRUMENT | | | |
| :COUPLE | {OFF ON 0 1} | | COUC |
| :NSELECT | {1 2} | | CHAN1, CHAN2 |
| [:SELECT] | {CH1 CH2} | | CHAN1, CHAN2, DUAC |
| :STATE | {OFF ON 0 1} | | CHAN1, CHAN2, DUAC |
| :TYPE | {NA SA ZA} | | NA, SA, ZA |

| Command | Parameter | Note | Equivalent Simple Command |
|-------------------|--|--------------|--|
| MMEmory | | | |
| :CDIRectory | [<string>] | [no query] | CHAD |
| :COpy | <string(src)> ,<msus> , <string(des)> ,<msus> | [no query] | FILC |
| :CREate | | | |
| :DIRectory | <string> | [no query] | CRED |
| :DELeTe | <file_name>[,<msus>] | [no query] | PURG, RESAVD |
| :INITialize | <msus> ,{LIF DOS} | [no query] | INID |
| :LOAD | | | |
| :STATe | <file_name>[,<msus>] | | RECD |
| :TRACe | SEL,<file_name>[,<msus>] | | RECD |
| :STORe | | | |
| :DINTerchange | | | |
| :TIFf | <file_name>[,<msus>] | [no query] | SAVDTIF |
| :TRACe | SEL,<file_name>[,<msus>] | | SAVDASC |
| :ITeM | | | |
| :TRACe | | | |
| :CATalog? | | [query only] | SAVCAL?, SAVDAT?, SAVDTR?, SAVMEM?, SAVMTR?, SAVRAW? |
| :DELeTe | {CCO DATA DTR MEM MTR RAW} | [no query] | SAVCAL, SAVDAT, SAVDTR, SAVMEM, SAVMTR, SAVRAW |
| :SELeCt | {CCO DATA DTR MEM MTR RAW} | [no query] | SAVCAL, SAVDAT, SAVDTR, SAVMEM, SAVMTR, SAVRAW |
| :STATe | <file_name>[,<msus>] | | SAVDSTA |
| :TRACe | SEL,<file_name>[,<msus>] | | SAVDDAT |
| OUTPut | | | |
| :ATTenuation{1 2} | <numeric> | | ATTP1, ATTP2 |

| Command | Parameter | Note | Equivalent Simple Command |
|-------------|--------------------------------------|--------------|---------------------------|
| PROGRAM | | | |
| :CATalog? | | [query only] | (None) |
| :EXPLicit | | | |
| :DEFine | <programe>,<program> | | (None) |
| :DEFine? | <programe> | [query only] | (None) |
| :DELete | <programe> | [no query] | (None) |
| :EXECute | <programe>,<program command> | [no query] | (None) |
| :MALLocate | <programe>,{<nbytes> DEFault} | | (None) |
| :NAME | <programe>,<program name> | | (None) |
| :NUMBer | <programe>,<varname> [,<nvalues>] | | (None) |
| :STATe | <programe>,{RUN PAUSE STOP CONTinue} | | (None) |
| :STRing | <programe>,<varname> [,<svalues>] | | (None) |
| :WAIT | <programe> | | (None) |
| [:SELEcted] | | | |
| :DEFine | <program> | | (None) |
| :DELete | | | |
| [:SELEcted] | | [no query] | (None) |
| :ALL | | [no query] | (None) |
| :EXECute | {“RUN” “PAUSE” “STOP” “STEP” “CONT”} | [no query] | (None) |
| :MALLocate | {<nbytes> DEFault} | | (None) |
| :NAME | <program name> | | (None) |
| :NUMBer | <varname>{,<nvalues>} | | (None) |
| :STATe | {RUN PAUSE STOP CONTinue} | | (None) |
| :STRing | <varname>{,<svalues>} | | (None) |
| :WAIT | | | (None) |

| Command | Parameter | Note | Equivalent Simple Command |
|---------------|-------------------------------|------------|---|
| SENSE | | | |
| :AVERage | | | |
| :CLEar | | [no query] | AVERREST |
| :COUNt | <numeric> | | AVERFACT |
| [:STATe] | {OFF ON 0 1} | | AVER |
| :BANDwidth | | | |
| [:RESolution] | <numeric> | | BW |
| :AUTO | {OFF ON 0 1} | | BW, BWAUTO |
| :RATio | <numeric> | | BWSRAT |
| :VIDeo | <numeric> | | VBW |
| :TYPE | {LINear LOGarithmic} | | VBWT |
| :CORRection | | | |
| :CIMPedance | <numeric> | | SETZ |
| :CKIT | {APC35 APC7 N50 N75 UDEFined} | | CALK |
| :CLASs{1-12} | | | |
| :STANdard | <n,n, ... > (max 7 items) | [no query] | SPECS11A, SPECS11B, SPECS11C, SPECS22A, SPECS22B, SPECS22C, SPECFWDm, SPECFWDt, SPECRESI, SPECRESP, SPECREVM, SPECREVT |
| :LABel | <string> | | LABEFD{ T M}, LABERE{ P I}, LABEREV{ T M}, LABES11{ A B C}, LABES22{ A B C} |
| :LABel | <string> | | LABK |
| :MODify | | [no query] | MODI1 |
| :SAVE | [no query] | | SAVEUSEK |
| :SAVE | {ALL CLASs} | [no query] | CLAD, KITD |
| :SELEct | STANdard{1-8} | | DEFS |

| Command | Parameter | Note | Equivalent Simple Command |
|-----------------------------|---|------------|--|
| SENSe (continued) | | | |
| :CORRection (continued) | | | |
| :CKIT (continued) | | | |
| :CLASs{1-13} (continued) | | | |
| :STANdard | | | |
| :C0 | <numeric> | | C0 |
| :C1 | <numeric> | | C1 |
| :C2 | <numeric> | | C2 |
| :LABel | <string> | | LABS |
| :OCIMpedance | <numeric> | | OFSZ |
| :ODELay | <numeric> | | OFSD |
| :OLOSS | <numeric> | | OFSL |
| :SAVE | | [no query] | STDD |
| :TIMPedance | <numeric> | | TERI |
| :TYPE | {AIMPedance DElay LOAD OPEN SHORT} | | STDT |
| :COLLect | | | |
| [:ACQuire] | {CS11A CS11B CS11C CS22A CS22B CS22C FWDI FWDM FWDT ISOL ISOL2 OMI REFL REFL2 RESP REVI REVM REVT TRAN TRAN2 STANdard{1-7}} | [no query] | CLASS11{A B C}, CLASS22{A B C}, FWDI, FWDM, FWDT, ISOL, OMI, RAIISOL, RAIRESP, REFL, REVI, REVM, REVT, STAN{A-G}, TRAN |
| :METhod | {NONE RESPonse RAISol S111 S221 TPORt OPTPort} | | CALI |
| :RESume | | [no query] | RESC |
| :SAVE{1-9} | | [no query] | DONE, ISOD, RAID, REFD, RESPDONE, SAV1, SAV2, SAVC, TRAD |
| :EDELay{1} | | | |
| :PORT{1-5} | | | |
| [:TIME] | <numeric> | | PORT1, PORT2, PORTA, PORTB, PORTR |
| :STATe | {OFF ON 0 1} | | PORE |
| :EDELay2 | {<numeric> MARKer} | | ELED, MKRDELA |
| :OFFSet | | | |
| [:MAGNitude] | <numeric> | | LVCDT |
| :PHASe | <numeric> | | PHAO |
| :RVELocity | <numeric> | | VELOFACT |
| [:STATe] | {OFF ON 0 1} | | CORR |

| Command | Parameter | Note | Equivalent Simple Command |
|-------------------|-------------------------------------|------------|---------------------------|
| :CORRection1 | | (ZA) | |
| :CKIT (continued) | | | |
| :CLASs{13-15} | | | |
| :STANdard | <num1>[,<num2>[, ... [,<num7>]]] | | SPECIMP{A B C} |
| :LABel | <string> | | LABIMP{A B C} |
| :COLLect | | | |
| [:ACQuire] | STANdard{1 2} | [no query] | STAN{A B} |
| [:ACQuire] | IMP{A B C} | [no query] | CALSIMP{A B C} |
| :METhod | {NONE IMP} | | CALI |
| :RESume | | | RESCOM |
| :SAVE | | | SAVIMP |
| :EDELay | | | |
| :PORT6 | | | |
| [:TIME] | <numeric> | | PORTZ |
| :CORRection2 | | (ZA) | |
| :CKIT | | | |
| :LABel | <string> | | LABECOMK |
| :MODify | | | MODICOMK |
| :SAVE | | | SAVUCOMK |
| :SAVE | | | COMKDONE |
| :STANdard | | | |
| :SAVE | | | COMSDONE |
| :STANdard1 | | | |
| :G | <numeric> | | DEFSOPENG |
| :C | <numeric> | | DEFSOPENC |
| :STANdard2 | | | |
| :R | <numeric> | | DEFSSHORR |
| :L | <numeric> | | DEFSSHORL |
| :STANdard3 | | | |
| :R | <numeric> | | DEFSLOADR |
| :L | <numeric> | | DEFSLOADL |
| :COLLect | | | |
| :METhod | IMPedance | | COMP |
| [:ACQuire] | STANdard{1 2 3} | [no query] | COMC |
| :SAVE | | | SAVCOM |
| :RESUME | | | RESCOM |

| Command | Parameter | Note | Equivalent Simple Command |
|-------------------|--|------------|-----------------------------------|
| SENSe (continued) | | | |
| :OPEN | {OFF ON 0 1} | | COMCDATA |
| :SHORT | {OFF ON 0 1} | | COMCDATB |
| :LOAD | {OFF ON 0 1} | | COMCDATC |
| :DETECTOR | | | |
| :CONTINUOUS | {OFF ON 0 1} | | REPTSMP |
| [:FUNCTION] | {NEGATIVE POSITIVE SAMPLE} | | DET |
| :FREQUENCY | | | |
| :CENTER | {<numeric> TPEAK DMARKER MARKER} | | CENT, MKRCENT, MKRDCENT, PEAKCENT |
| :STEP | | | |
| [:INCREMENT] | {<numeric> DMARKER MARKER} | | CNTS |
| :AUTO | {OFF ON 0 1} | | CNTSAUTO |
| :MODE | {FIXED LIST SWEPT} | | SWPT |
| :SPAN | {<numeric> DMARKER MZAPERTURE} | | MKRSPAN, MKRZM, SPAN |
| :FULL | | [no query] | FULS |
| :START | {<numeric> MARKER} | | MKRSTAR, STAR |
| :STOP | {<numeric> MARKER} | | MKRSTOP, STOP |
| :FUNCTION | {“POWER 1” “POWER 2” “POWER 3” “POWER 4” “POWER:S11” “POWER:S12” “POWER:S21” “POWER:S22” “POWER{1-4}:PSDENSITY” “POWER:RATIO 3,2” “POWER:RATIO 4,2”} | | FMT, MEAS |
| :LIST | | | |
| :CLEAR | | [no query] | CLEL |
| :SAVE | | [no query] | EDITDONE |
| :SEGMENT | <numeric> | | (None) |
| :ADD | | [no query] | SADD |
| :BANDWIDTH | <numeric> | | BW |
| :DELETE | | [no query] | SDEL |
| :EDIT | | [no query] | SEDI |
| :FREQUENCY | | | |
| :CENTER | <numeric> | | CENT |
| :SPAN | <numeric> | | SPAN |
| :START | {<numeric> MARKER} | | MKRSTAR, STAR |
| :STOP | {<numeric> MARKER} | | MKRSTOP, STOP |
| :POINTS | <numeric> | | POIN |
| :POWER | <numeric> | | POWE |
| :QUIT | | [no query] | SQUI |
| :SAVE | | [no query] | SDON |

| Command | Parameter | Note | Equivalent Simple Command |
|-------------------|----------------------------------|------------|-----------------------------------|
| SENSe (continued) | | | |
| :POWer | | | |
| :AC | | | |
| :ATTenuation | <numeric> | | ATT |
| :AUTO | {OFF ON 0 1} | | ATT, ATTAUTO |
| :RANGe | | | |
| [:UPPer] | <numeric> | | MAXMLEV |
| :SWEep | | | |
| :COUNt | <numeric> | | NUMG, SING |
| :GATed | {OFF ON 0 1} | | TRGS |
| :DELay | <numeric> | | GATDLY |
| :LENGth | <numeric> | | GATLEN |
| :TRIGger | {EDGE LEVel} | | GATCTL |
| :POINts | <numeric> | | POIN |
| :SPACing | {LINear LOGarithmic} | | SWPT |
| :TIME | <numeric> | | SWET |
| :AUTO | {OFF ON 0 1} | | SWETAUTO |
| :TRACk | | | |
| :SIGNal | | | |
| :MARKer | {OFF ON 0 1} | | SGTRK |
| SOURce | | | |
| :FREQuency | | | |
| [:CW] | <numeric> | | CWFREQ |
| :POWer | | | |
| :CENTer | {<numeric> TPEak DMARker MARKer} | | CENT, MKRCENT, MKRDCENT, PEAKCENT |
| [:LEVel] | | | |
| [:IMMediate] | | | |
| [:AMPLitude] | <numeric> | | POWE |
| :SLOPe | <numeric> | | SLOPE |
| :STATe | {OFF ON 0 1} | | SLOP |
| :MODE | {FIXed LIST SWEep} | | SWPT |
| :SPAN | {<numeric> DMARker MZAPerture} | | MKRSPAN, MKRZM, SPAN |
| :FULL | | [no query] | FULS |
| :STARt | {<numeric> MARKer} | | MKRSTAR, STAR |
| :STATe | {OFF ON 0 1} | | RFO |
| :STOP | {<numeric> MARKer} | | MKRSTOP, STOP |

| Command | Parameter | Note | Equivalent Simple Command |
|---------------|-----------|--------------|---------------------------|
| STATUS | | | |
| :INSTrument | | | |
| :ENABLE | <numeric> | | ESNB |
| [:EVENT]? | | [query only] | ESB? |
| :OPERation | | | |
| :CONDition? | | [query only] | OSR? |
| :ENABLE | <numeric> | | OSE |
| [:EVENT]? | | [query only] | OSER? |
| :NTRansition | <numeric> | | OSNT |
| :PTRansition | <numeric> | | OSPT |
| :PRESet | | | |
| :QUEStionable | | | |
| :CONDition? | | [query only] | |
| :ENABLE | <numeric> | | |
| [:EVENT]? | | [query only] | |

| Command | Parameter | Note | Equivalent Simple Command |
|--|--|---|--|
| <p>SYSTEM</p> <p>:BEEPPer{1 2}</p> <p>:STATe</p> <p>:COMMunicate</p> <p>:PARAllel</p> <p>[:RECeive]</p> <p>:DATA?</p> <p>:TRANsmit</p> <p>:DATA</p> <p>:TSET?</p> <p>:DATE</p> <p>:MODE</p> <p>:ERRor?</p> <p>:FIXTuer</p> <p>:DISTance</p> <p>:LABel</p> <p>:MODify</p> <p>:SAVE</p> <p>:SAVE</p> <p>:KEY</p> <p>:KLOCK</p> <p>:PRESet</p> <p>:SECurity</p> <p>[:STATe]</p> <p>:TIME</p> <p>:VERSion?</p> | <p>{OFF ON 0 1}</p> <p><numeric></p> <p><year> ,< month> ,< day></p> <p>{DMY MDY}</p> <p>{NONE HP16191 HP16192 HP16193 HP16194 USER}</p> <p><numeric></p> <p><string></p> <p><numeric></p> <p>{OFF ON 0 1}</p> <p>ON</p> <p><hour> ,< minute> ,< second></p> | <p>[query only]</p> <p>[no query]</p> <p>[query only]</p> <p>[query only]</p> <p>[no query]</p> <p>[query only]</p> <p>[no query]</p> <p>[query only]</p> <p>[no query]</p> <p>[query only]</p> | <p>BEEPDONE, BEEPWARN</p> <p>INP8IO?</p> <p>OUT8IO</p> <p>TESS?</p> <p>SETCDATE</p> <p>DAYYEAR, MONDYEAR</p> <p>OUTPERRO?</p> <p>FIXT</p> <p>FIXE</p> <p>LABEFIX</p> <p>MODIFIX</p> <p>FIXKDONE</p> <p>SAVUFIXT</p> <p>KEY</p> <p>DSKEY, ENKEY</p> <p>PRES</p> <p>FREO</p> <p>SETCTIME</p> <p>(None)</p> |
| <p>TRACe</p> <p>:COPY</p> <p>[:DATA]</p> <p>:VALue?</p> | <p>{MTRace,DTRace}</p> <p>{DTR MTR}, {<block> <numeric> [,<numeric>]}</p> <p>{DTR MTR}, <point></p> | <p>[no query]</p> <p>[query only]</p> | <p>DATMEM</p> <p>INPUDTRC, OUTPDTRC?, OUTPMTRC?</p> <p>OUTPDTRCP?, OUTPMTRCP?</p> |

| Command | Parameter | Note | Equivalent Simple Command |
|---|---|------|----------------------------------|
| TRIGger :EVENT :TYPE :LEVel :SLOPe :SOURce | {POINT SWEEp} <numeric> {POSitive NEGative} {BUS EXTerna INTerna1 INTerna2 MANual} | | TRGEVE VIDLVL TRGP TRGS |
| UNIT :POWer | {DBM DBUV DBV V W} | | SAUNIT |

Status Reporting

The Status byte register (STB) summarizes four status registers that indicate the internal condition of the analyzer. Figure D-1 shows the status reporting structure of the analyzer.

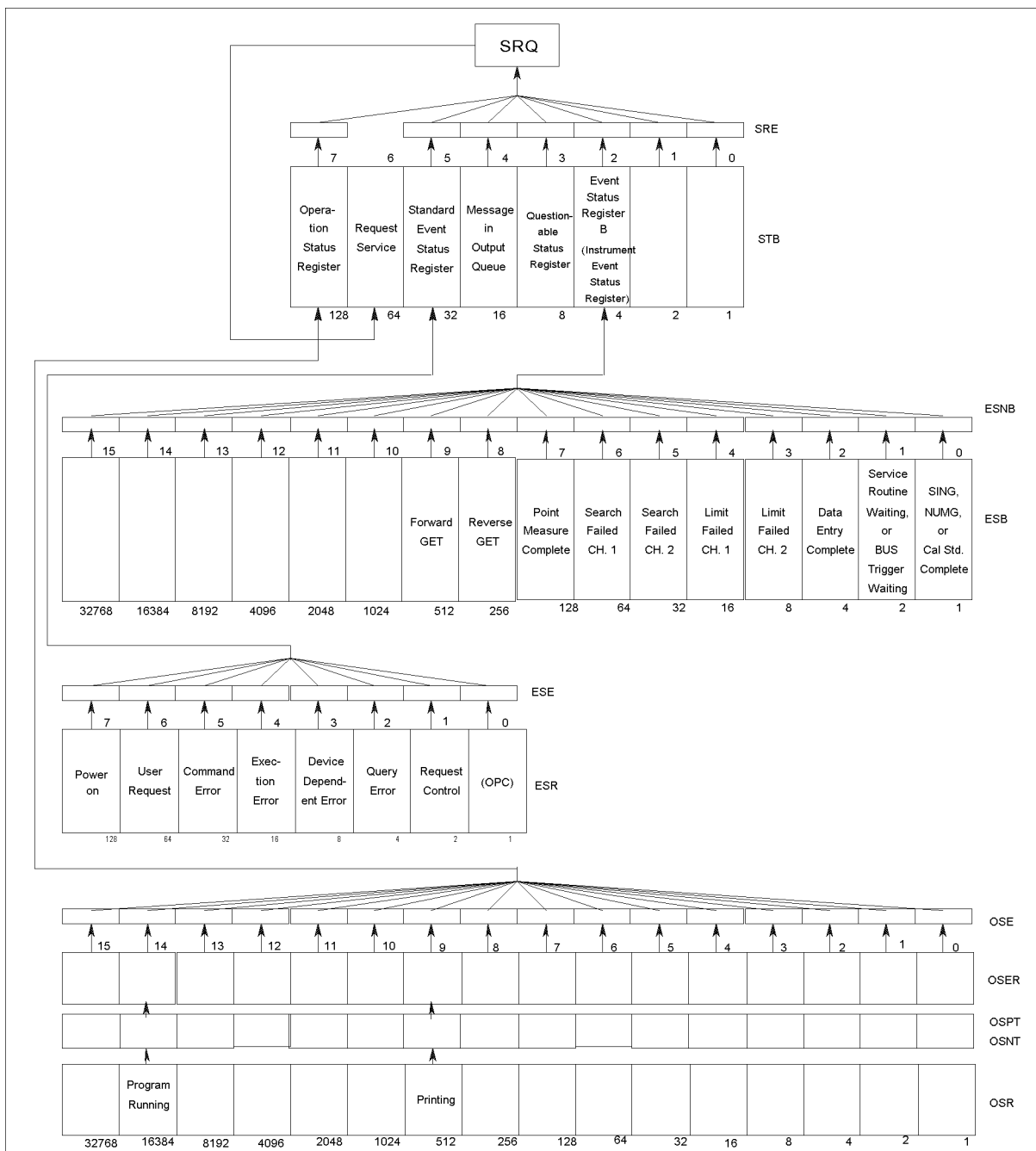


Figure D-1. Status Reporting Structure

The analyzer has a status reporting system to report the condition of the analyzer. The status bytes consist of 8-bit registers, with each bit representing a specific analyzer condition. The value of the Status Byte can be read by using SPOLL(717) statement from an external controller. This command reads a value directly from the analyzer without being set to remote. So, you can operate front panel keys while a controller is reading the Status Byte. Contents of the Status Byte can also be read by using the *STB? command. Reading the Status Byte does not affect the contents of the Status Byte. Table D-1 shows contents of Status Byte.

Table D-1. Status Bit Definitions of the Status Byte (STB)

| Bit | Name | Description |
|-----|--|---|
| 2 | Event Status Register B Summary Bit | One of the enabled bits in Event Status Register B (Instrument Event Status Register) has been set. |
| 3 | Questionable Status Register Summary Bit | The analyzer has no operation to report the event to the Questionable Status Register group. This register is available to keep the consistency with other SCPI compatible instruments. |
| 4 | Message in Output Queue | A command has prepared information to be output, but it has not been read yet. |
| 5 | Standard Event Status Register Summary Bit | One of the enabled bits in the Standard Event Status Register has been set. |
| 6 | Request Service | One of the enabled Status Byte bits is causing an SRQ. |
| 7 | Operation Status Register Summary Bit | One of the enabled bits in the Operation Status Register has been set. |

For example, to read the contents of Message in the output queue,

```

10 Stat=SPOLL(717)
20 Stb4=BIT(Stat,4)
30 PRINT Stb4
40 END

```

Figure D-2. Example of Reading Status Byte (1)

or,

```

10 ASSIGN @Hp4396 TO 717
20 OUTPUT @Hp4396;"*STB?"
30 ENTER @Hp4396;Stat
40 Stb4=BIT(Stat,4)
50 PRINT Stb4
60 END

```

Figure D-3. Example of Reading Status Byte (2)

The Standard Event Status Register (ESR), Event Status Register B (ESB; Instrument Event Status Register), and Operation Status Register (OSR) are subordinate to the Status Byte. Each

D-2 Status Reporting

register can set a bit with a condition that is watched by status bit. A status bit is cleared when it is read by query or the CLES or *CLS command is executed.

**Table D-2.
Status Bit Definitions of the Standard Event Status Register (ESR)**

| Bit | Name | Description |
|------------|------------------------|---|
| 0 | Operation Complete | A command for which OPC has been enabled, and completed an operation. |
| 1 | Request Control | The analyzer has been commanded to perform an operation that requires control of a peripheral, and needs control of GPIB. |
| 2 | Query Error | <ol style="list-style-type: none"> 1. The analyzer has been addressed to talk, but there is nothing in the output queue to transmit. 2. Data in the Output Queue has been lost. |
| 3 | Device Dependent Error | An error, other than a command error, a query error, and an execution error has occurred. |
| 4 | Execution Error | <ol style="list-style-type: none"> 1. A program data element following a header exceeded its input range, or is inconsistent with the analyzer's capabilities. 2. A valid program message could not be properly executed due to some analyzer condition. |
| 5 | Command Error | <ol style="list-style-type: none"> 1. An IEEE 488.2 syntax error has occurred. Possible violations include, a data element violated the analyzer listening formats or a data element type is unacceptable to the analyzer. 2. A semantic error that indicates an unrecognized header was received has occurred. Unrecognized headers include incorrect device-specific headers and incorrect or unimplemented IEEE 488.2 common commands. 3. A Group Execute Trigger (GET) was entered into the Input Buffer of a program message. |
| 6 | User Request | The operator pressed a front panel key or an optional keyboard key or turned the rotary knob. |
| 7 | Power ON | This bit is set when a power-on sequence occurs. |

Table D-3. Status Bit Definitions of the Event Status Register B (ESB)

| Bit | Name | Description |
|-----|--|---|
| 0 | SING, NUMG, or Cal Std. Complete | A single, group sweep, calibration, or compensation has been completed since the last read of the register. Operates in conjunction with SING or NUMG. |
| 1 | Service Routine Waiting or Bus Trigger Waiting | 1. An internal service routine has completed an operation, or is waiting for an operator response. 2. The analyzer has set the manual trigger to the point mode and is waiting for a manual trigger. |
| 2 | Data Entry Complete | A terminator key has been pressed. |
| 3 | Limit Failed, Ch 2 | Limit test failed on channel 2. |
| 4 | Limit Failed, Ch 1 | Limit test failed on channel 1. |
| 5 | Search Failed, Ch 2 | A marker search was executed on channel 2, but the target value was not found. |
| 6 | Search Failed, Ch 1 | A marker search was executed on channel 1, but the target value was not found. |
| 7 | Point Measurement Complete ¹ | One measurement point of a sweep has been completed. |
| 8 | Reverse GET | A one-path 2-port calibration is active, and the analyzer has stopped, waiting for the operator to connect the device for a reverse measurement. |
| 9 | Forward GET | A one-path 2-port calibration is active, and the analyzer has stopped, waiting for the operator to connect the device for a forward measurement. |

¹ This bit is set only when the related bits of both SRE and ESNB are enabled.

In the case of the manual trigger on point mode, the analyzer accepts the next trigger while the current measurement is in progress (up to the number of points). Use bit 1 and bit 7 correctly to synchronize the measurement and external triggering. For example, 1) wait until bit 1 is set, 2) trigger, and 3) wait until bit 7 is set.

Table D-4. Status Bit Definitions of the Operation Status Register (OSR)

| Bit | Name | Description |
|-----|-----------------|--|
| 14 | Program running | An HP instrument BASIC program is running. |
| 9 | Printing | Data is being transferred to the printer. |

Each status register has a register that enables generating a Service Request (SRQ) with a condition of a status bit. For instance, to generate an SRQ when the analyzer completes the specified number of sweeps, enable ESNB bit 1. Bit 1 of ESNB is the mask register for ESB 0 (“SING, NUMG, or Cal Std. Complete”) which shows sweep completion and SRE bit 2. This enables a path from ESB bit 0 to generate an SRQ. Figure D-4 shows a program listing that can be used to generate an SRQ.

```

10 ASSIGN @Hp4396 TO 717
20 !
30 OUTPUT @Hp4396;"CLES" ! Clears status registers
40 OUTPUT @Hp4396;"ESNB 1" ! Enables mask register of "SING. NUMG. or
50 ! ! Cal Std. Complete" of ESB
60 OUTPUT @Hp4396;"*SRE 4" ! Enables mask register of "Event Status
70 ! ! Register B" of STB
80 !
90 ON INTR 7 GOTO End ! Declare SRQ interrupt
100 ENABLE INTR 7;2
110 OUTPUT @Hp4396;"SING" ! Execute single sweep
120 GOTO 120 ! Endless loop
130 !
140 End: ! Exit from loop when sweep is completed
150 END

```

Figure D-4. Example of Generating a Service Request (SRQ)

OSPT, OSNT

OSPT (Operation Status Positive Transition Filter)

Sets the positive transition filter. Setting a bit in OSPT will cause a 0 to 1 transition in the corresponding bit of the associated Operation Status Register (OSR) to cause a 1 to be written in the associated bit of corresponding Operation Status Event Register (OSER).

Bit 14 of the analyzer's OSR is used to show program status. When bit 14 of OSPT is set to 1, starting a program causes a 1 to be written in bit 14 of OSER. (This sets bit 7 of STB to 1.)

OSNT (Operation Status Negative Transition Filter)

Sets the negative transition filter. Setting a bit in the negative transition filter will cause a 1 to 0 transition in the corresponding bit of the associated Operation Status Register to cause a 1 to be written in the associated bit of corresponding Operation Status Event Register.

Bit 14 of the analyzer's OSR is used to show program status. When bit 14 of OSNT is set to 1, stopping a program causes a 1 to be written in bit 14 of OSER. (This sets bit 7 of STB to 1.)

Trigger System

Trigger System

This section provides information about the trigger system of the analyzer. SCPI defines a common trigger model for several types of instruments. A trigger system allows you to have control of your measurements.

Information on the trigger system requires more technical expertise than most other topics covered in this chapter. But you can avoid having to learn the information in this chapter by using the `:INITiate` commands to make your measurements.

Analyzer Trigger System Configuration

The trigger system synchronizes the analyzer measurement with specified events. Events include an GPIB trigger command or input pulse on the EXT TRIGGER input. The trigger system also allows you to specify the number of times to repeat a measurement and the delays between measurements.

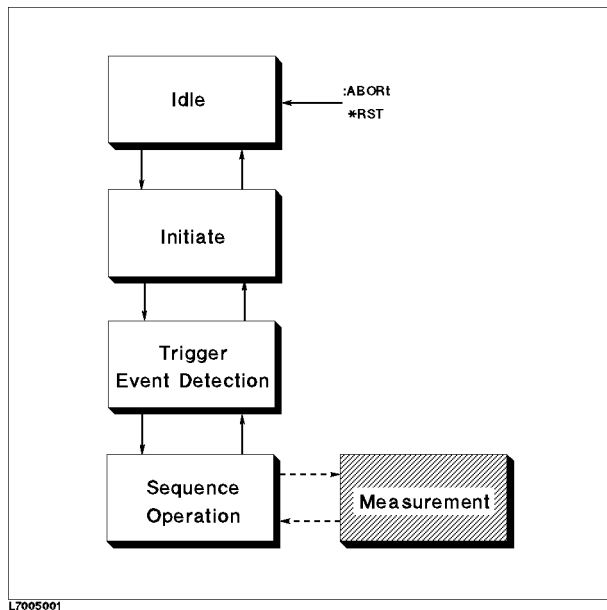


Figure E-1. Trigger System Configuration

Figure E-1 shows the configuration of the analyzer trigger system. Each unshaded block is called a **trigger state**. The analyzer moves between adjacent states depending on its conditions. The power ON state is called the **Idle** state. You can force the analyzer to the idle state using the `:ABORT` or `*RST` command. The **Initiate** and **Trigger Event Detection** state branches to next state, whether the analyzer satisfies the specified conditions or not. The **Sequence Operation** state signals the instrument hardware to take a measurement and listens for a signal saying that the measurement has been taken.

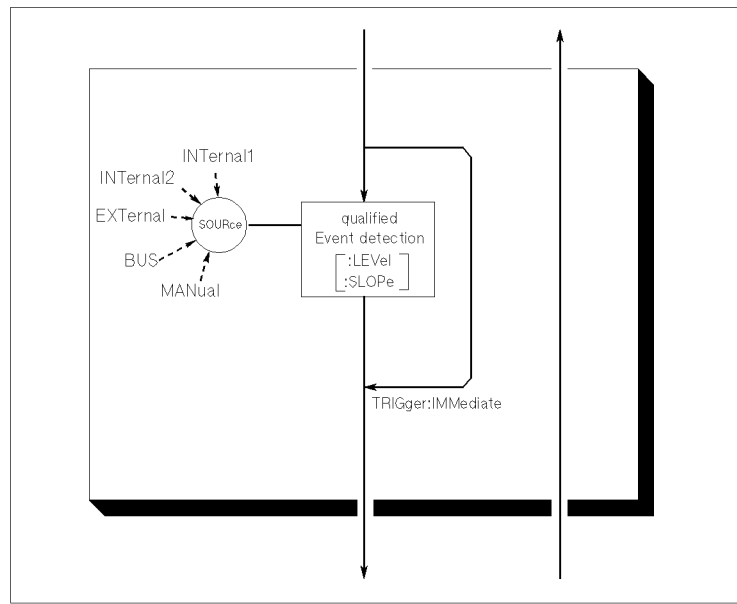
Idle State

The trigger system remains in the Idle state until it is initiated by `:INITiate:IMMediate` or `:INITiate:CONTinuous ON`. Once one of these conditions is satisfied, the trigger system exits downward to the Initiate state. Note that `*RST` sets `:INITiate:CONTinuous OFF`.

Initiate State

If the trigger system is on a downward path, it travels directly through the Initiate state without restrictions. If the trigger system is on an upward path, and `:INITiate:CONTinuous` is `ON`, then it exits downward to a Trigger Event Detection state. If the trigger system is on an upward path and `:INITiate:CONTinuous` is `OFF`, then it exits upward to the Idle state.

Trigger Event Detection State



C540E001

Figure E-2. Inside an Trigger Event Detection State

SOURCE

The `:TRIGger:SOURce` command specifies which particular input can generate the event required to continue the downward path. If the source chosen is a nonanalog signal such as `IMMediate` or `BUS`, then no further qualifications are required to generate an event. However when an `INTERNAL1`, `INTERNAL2`, or `EXTERNAL` analog signal is chosen, you can specify additional qualifications by using the appropriate `LEVel` and `SLOPe` commands. Sending `*RST` sets the `SOURce` to `IMMediate`.

IMMediate

The `:TRIGger:IMMediate` command bypasses the event detection and `DELaY` qualifications one time. The upward path through the Trigger Event Detection state contains only one condition. If the condition is satisfied, the trigger system exits upward.

Sequence Operation State

The downward entrance to the Sequence Operation state forces the analyzer to start a measurement. An upward exit is not allowed until the measurement is complete.

Calibration Types and Standard Classes, and Calibration Arrays

Table F-1 lists which standard classes are required for each calibration type. Table F-2 specifies where the calibration coefficients are stored for different calibration types.

Table F-1. Calibration Types and Standard Classes

| Class | Response | Response and Isolation | S ₁₁ 1-port | S ₂₂ 1-port | One-path 2-port | Full 2-port | ZA cali- bration |
|--|----------|------------------------|---------------------------|---------------------------|--------------------|----------------|---------------------|
| Response: | • | | | | | | |
| Response and isolation: Response Isolation | | • • | | | | | |
| Reflection: ¹ S11A (opens) S11B (shorts) S11C (loads) S22A (opens) S22B (shorts) S22C (loads) | | | • • • | • • • | • • • | • • • | |
| Transmission: ¹ Forward match Forward thru Reverse match Reverse thru | | | | | • • • | • • • | |
| Isolation: ¹ Forward Reverse | | | | | • • | • • | |
| Impedance analyzer cal IMPA (OPEN) IMPB (SHORT) IMPC (LOAD) | | | | | | | • • • |

¹ These subheadings must be called when doing 2-port calibrations.

Table F-2. Calibration Array

| Array | Response ¹ | Response and Isolation ¹ | 1-port ¹ ZA cal | 2-port ¹² |
|-------|----------------------------------|---|----------------------------|----------------------|
| 1 | E _R or E _T | E _X (E _D) ³ E _T (E _R) | E _D | E _{DF} |
| 2 | | | | E _{SF} |
| 3 | | | | E _{RF} |
| 4 | | | | E _{XF} |
| 5 | | | | E _{LF} |
| 6 | | | | E _{TF} |
| 7 | | | | E _{DR} |
| 8 | | | | E _{SR} |
| 9 | | | | E _{RR} |
| 10 | | | | E _{XR} |
| 11 | | | | E _{LR} |
| 12 | | | | E _{TR} |

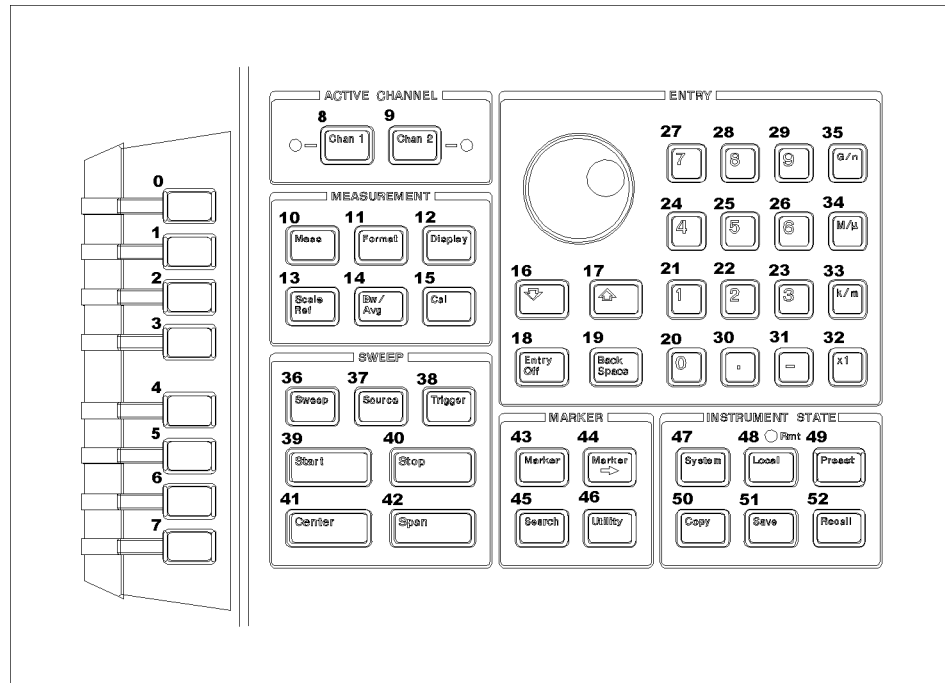
1 Meaning of first subscript: D=directivity; S=source match; X=crosstalk; L=load match; T=transmission tracking.
 Meaning of second subscript: F=forward; R=reverse.

2 One path, 2-port cal duplicates arrays 1 to 6 in arrays 7 to 12.

3 Response and isolation corrects for crosstalk and transmission tracking in transmission measurements, and for directivity and reflection tracking in reflection measurements.

Key Codes

Figure G-1 shows the codes of the front panel keys for using the KEY GPIB command.



CB40G001

Figure G-1. Key Codes

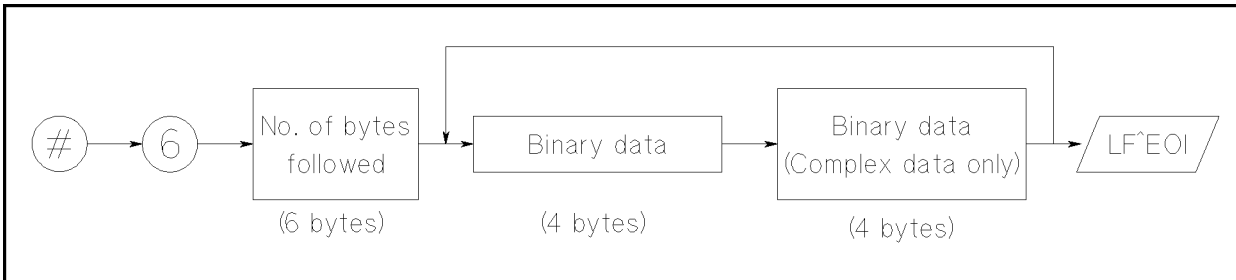
Data Format and Data Levels

Data Format

The analyzer can transmit data over GPIB in four different formats. The type of format affects what kind of data array is declared (real or integer), because the format determines what type of data is transferred.

■ Form 2

IEEE 32-bit floating point format. Figure H-1 shows the data transfer format of Form 2. In this mode, each number takes 4 bytes.

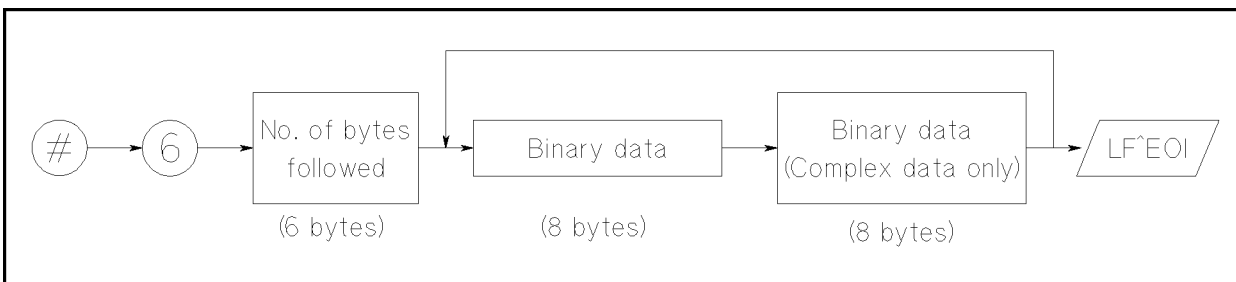


C4502004

Figure H-1. Form 2 Data Transfer Format

■ Form 3

IEEE 64-bit floating point format. Figure H-2 shows the data transfer format of Form 3. Data is stored internally in the 200/300 series computer with the IEEE 64-bit floating point format, eliminating the need for any reformatting by the computer. In this mode, each number takes 8 bytes.



C4502003

Figure H-2. Form 3 Data Transfer Format

■ Form 4

ASCII data transfer format. In this mode, each number is sent as a 24-character string, each character being a digit, sign, or decimal point.

- Form 5

MS-DOS[®] personal computer format. This mode is a modification of IEEE 32-bit floating point format with the byte order reversed. Form 5 also has a four-byte header that must be read in so that data order is maintained. In this mode, an MS-DOS[®] PC can store data internally without reformatting it.

Data Levels

The analyzer has the following data arrays in internal memory:

- Raw data

These arrays store the results of all the preceding data processing operations. Note that the numbers here are still complex pairs.

When the Network analyzer mode and the full 2-port error correction are on, the raw data arrays contain all four S-parameter measurements required for accuracy enhancement.

- Error corrected data

The results of error correction are stored in the data arrays as complex number pairs.

- Formatted data

This is the array of data being displayed. It reflects all post-processing functions such as electrical delay, and the units of the array read out depends on the current display format.

- Calibration coefficients (Network and impedance analyzer only)

The results of a calibration are stored arrays of calibration coefficients that are used by the error correction routines. Each array corresponds to a specific error term in the error model. The calibration coefficients are read out with `OUTPCALC{1-12}?`.

- fixture compensation coefficients (Impedance analyzer only)

The results of a fixture compensation are stored arrays of fixture compensation coefficients that are used by the error correction routines. Each array corresponds to a specific error term in the error model. The fixture compensation coefficients are read out with `OUTPCOMC{1-3}?`.

Formatted data is generally the most useful, because it is the same information as that seen on the display. However, if post-processing is not necessary, as may be the case with smoothing, error corrected data is more desirable. Error corrected data also gives you the opportunity to load the data into the instrument and apply post-processing at a later time.

For more information of the data processing, refer to “Data Processing Flow” in chapter 12 of the *Reference Manual*.

Marker Readout

The values specified by the marker, sub-marker, or delta-marker can be read using the following commands.

OUTPMKR?, Amplitude value (Value 1), Auxiliary amplitude value (Value 2), Sweep
 OUTPSMKR?, Parameter
 OUTPDMKR?

MKRVAL?, Amplitude value (Value 1)
 SMKRVAL{1-7}?

MKRAUV?, Auxiliary amplitude value (Value 2)
 SMKRAUV{1-7}?

The following table lists the amplitude value (value 1) and the auxiliary amplitude value (value 2) for each display format.

Table H-1. Marker Readout

| Analyzer Type | Display Format | Parameter of CIRF Command | Amplitude Value (Value 1) | Auxiliary Amplitude Value (Value 2) |
|---|-------------------------|---------------------------|--|-------------------------------------|
| Network Analyzer | Log Magnitude | — | Log Magnitude (dB) | 0 |
| | Phase | — | Phase (degrees) | 0 |
| | Expanded Phase | — | Phase (degrees) | 0 |
| | Delay | — | Delay (seconds) | 0 |
| | Linear Magnitude | — | Linear Magnitude | 0 |
| | SWR | — | SWR | 0 |
| | Real | — | Real | 0 |
| Network/impedance Analyzer ¹ | Smith Chart | RI | Real | Imaginary |
| | Polar | LIN | Linear Magnitude | Phase (degrees) |
| | Addmittance | LOG | Log Magnitude (dB) | Phase (°) |
| | | RX | Resistance (Ω) | Reactance (Ω) |
| | | GB | Conductance (S) | Suseptance (S) |
| | | SWR | SWR | Phase (°) |
| | | | | |
| Spectrum Analyzer | Spectrum Measurement | — | Magnitude (dBm, dBV, dBμV, W, or V) ² | 0 |
| | Noise Level Measurement | — | Magnitude (dBm, dBV, dBμV, W, or V) ² | 0 |

¹ For the other format than listed above in the impedance analyzer mode, the marker readout has the unit of the selected parameter by **[Meas]** key.

² Unit is specified by the SAUNIT command. (default: dBm)

Waveform Analysis Commands

The 4396A has added a command set that can be used to analyze waveforms of specific devices. The waveform analysis commands analyze and output the results using only a single command. This appendix provides information about the added waveform analysis commands.

The commands are divided into five groups as follows:

- Waveform analysis setup commands
- Maximum/Minimum/Mean search commands
- Ripple analysis commands
- Filter and Resonator analysis commands
- Equivalent circuit analysis commands

All of the commands that are described in this appendix are executable using the Instrument BASIC EXECUTE command. By using the EXECUTE command, you can execute the waveform analysis commands much faster than by using the OUTPUT statement. If you use Instrument BASIC, it is recommended that you use EXECUTE with the waveform analysis commands.

For detail information about EXECUTE command, see the *Using HP instrument BASIC with 4396A* manual.

Conventions and Definitions

This section describes the conventions and definitions that are used to describe the waveform analysis commands.

- ① → **ANARANG**
- ② → Sets the stimulus range for the waveform. . . .
- ③ → **Syntax** ANARANG *start,stop*
- ④ → Where,

| | | |
|---|--------------|-----------------------------------|
| 0 | <i>start</i> | Start value of the analysis range |
| 1 | <i>stop</i> | Stop value of the analysis range |
- ⑤ → **Query**
- Response**
- ⑥ → **Semantics**
- ⑦ → **Note**
- ⑧ → **Examples**

| | |
|---|---|
| ① | Command name. |
| ② | Command description. |
| ③ | <p>Command syntax</p> <p>This part shows the syntax of the command. You must put a space between the command and the parameters.</p> |
| ④ | <p>Command parameter description</p> <p>The first column of the table lists the register number that is used by the EXECUTE command. You must put the parameter in the indicated register before using the EXECUTE command. For example (in the above case):</p> <pre style="margin-left: 40px;">WRITEIO 15,0;Start Put "Start" in register 0. WRITEIO 15,1;Stop Put "Stop" in register 1. EXECUTE "ANARANG" Execute "ANARANG".</pre> <p>The second column lists the parameter name that is shown in the Syntax area. The third column describes the parameters.</p> |
| ⑤ | <p>Query response.</p> <p>This part shows what values will be returned as the query response. The description of the query response is similar to the description of the Syntax area shown above.</p> |
| ⑥ | <p>Semantics</p> <p>This part describes how the command obtains the values for the query response.</p> |
| ⑦ | <p>Note</p> <p>This part describes the required conditions or limitations when using the command.</p> |
| ⑧ | <p>Examples</p> <p>This part shows examples of how to use the command. Examples are provided for both HP BASIC on an external controller and Instrument BASIC on the analyzer.</p> |

Waveform Analysis Setup Commands

The following commands are used for setting up the conditions for waveform analysis:

- ANAOCH1
- ANAOCH2
- ANARANG
- ANARFULL
- ANAODATA
- ANAOMEMO
- THRR

The settings are effective for all of the waveform analysis commands.

ANA0CH1

Selects channel 1 for waveform analysis.

Syntax ANAOCH1

Query *boolean*

Response Where,

| Register | Parameter | Description |
|----------|----------------|---|
| 0 | <i>boolean</i> | 1 or 0. Channel 1 is selected (1) or is not selected (0) for waveform analysis. |

Note ■ The ANAOCH1 and ANAOCH2 channel setting is independent of the active channel setting.

ANA0CH2

Selects channel 2 for waveform analysis.

Syntax ANAOCH2

Query *boolean*

Response Where,

| Register | Parameter | Description |
|----------|----------------|---|
| 0 | <i>boolean</i> | 1 or 0. Channel 2 is selected (1) or is not selected (0) for waveform analysis. |

Note ■ The ANAOCH1 and ANAOCH2 channel setting is independent of the active channel setting.

ANARANG

Sets the stimulus range for waveform analysis commands by start and stop value.

Syntax ANARANG *start, stop*

Where,

| Register | Parameter | Description |
|----------|--------------|------------------------------------|
| 0 | <i>start</i> | Start value of the analysis range. |
| 1 | <i>stop</i> | Stop value of the analysis range. |

Query *start, stop*

Response

Note

- The waveform analysis range is independent of the marker search range.
- You can set the range for each channel independently. Therefore, you need to set the analysis channel using ANAOCH1 or ANAOCH2 before using ANARANG.
- The waveform analysis range will be truncated to fit the displayed stimulus range if the setting is exceeded.
- If the displayed stimulus range is changed, the waveform analysis range is set equal to the displayed range.
- Store the waveform analysis range setting using **(SAVE)** **ALL** or **STATE ONLY**.
- The waveform analysis range is set to equal to the displayed stimulus range when the power is turned on.

Examples For External Controller

```
INPUT "Enter Start for Analysis Range.",Start
INPUT "Enter Stop for Analysis Range.",Stop
OUTPUT @Hp4396;"ANARANG ";Start,Stop
```

For Instrument BASIC

```
INPUT "Enter Start for Analysis Range.",Start
INPUT "Enter Stop for Analysis Range.",Stop
WRITEIO 8,0;Start
WRITEIO 8,1;Stop
EXECUTE "ANARANG"
```

ANARFULL

Sets the waveform analysis range equal to the displayed stimulus range. (No Query)

Syntax ANARFULL

Note

- You can set the range for each channel independently. Therefore, you need to set the analysis channel using ANAOCH1 or ANAOCH2 before using ANARFULL.

ANAODATA

Selects the data trace for waveform analysis.

Syntax ANAODATA

Query *boolean*

Response Where,

| Register | Parameter | Description |
|----------|----------------|--|
| 0 | <i>boolean</i> | 1 or 0. Data trace is selected (1) or is not selected (0) for waveform analysis. |

Note ■ You can select the trace for each channel independently. Therefore, you need to set the analysis channel using ANA0CH1 or ANA0CH2 before using ANAODATA.

ANAOMEMO

Selects the data trace for waveform analysis.

Syntax ANAOMEMO

Query *boolean*

Response Where,

| Register | Parameter | Description |
|----------|----------------|--|
| 0 | <i>boolean</i> | 1 or 0. Memory trace is selected (1) or is not selected (0) for waveform analysis. |

Note ■ You can select the trace for each channel independently. Therefore, you need to set the analysis channel using ANA0CH1 or ANA0CH2 before using ANAOMEMO.

THRR

Sets threshold ripple height for waveform analysis commands.

Syntax THRR *height*

Where,

| Register | Parameter | Description |
|----------|---------------|--|
| 0 | <i>height</i> | (Peak height) - (negative peak height) |

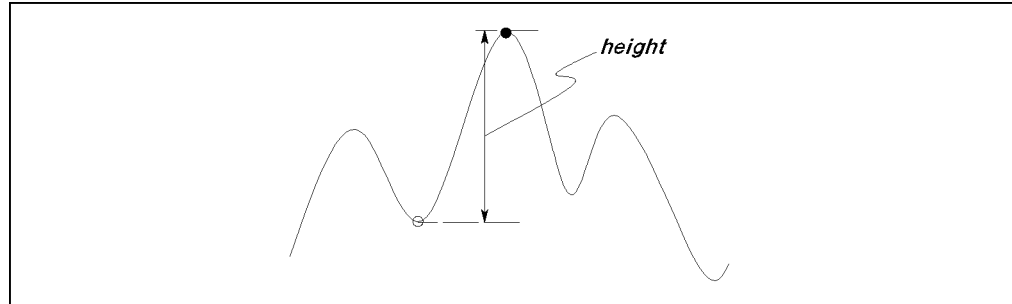


Figure I-1. THRR

Query *height*

Response

- Semantics**
- Ripple height is defined as the difference between the positive peak and the negative peak.
 - Waveform analysis commands search only for ripples greater than the threshold value, any others are ignored.

Note ■ Default threshold value is 0.

Examples **For External Controller**

```
INPUT "Enter Pos. Peak Gain [dB].",Local_max
INPUT "Enter Neg. Peak Gain [dB].",Local_min
Height=Local_max-Local_min
OUTPUT @Hp4396;"THRR ";Height
```

For Instrument BASIC

```
INPUT "Enter Pos. Peak Gain [dB].",Local_max
INPUT "Enter Neg. Peak Gain [dB].",Local_min
Height=Local_max-Local_min
WRITEIO 8,0;Height
EXECUTE "THRR"
```

Maximum/Minimum/Mean Value Search Commands

The following commands return the maximum, minimum, and mean value of a trace within the range specified using the ANARANG command.

- OUTPMAX?
- OUTPMIN?
- OUTPMINMAX?
- OUTPMEAN?
- PEAK?
- NEXPK?
- NUMLMAX?
- NUMLMIN?
- LMAX?
- LMIN?
- TARR?
- TARL?

OUTPMAX?

Returns the maximum point value and its stimulus within the specified range. (Query only)

Syntax OUTPMAX?

Query *MAX, f_{max}*

Response Where,

| Register | Parameter | Description |
|----------|------------------------|--|
| 0 | <i>MAX</i> | Maximum value |
| 1 | <i>f_{max}</i> | Stimulus at maximum point (Frequency or Power) |

Examples For External Controller

```
OUTPUT @Hp4396;"OUTPMAX?"
ENTER @Hp4396;Max_value,F_max
PRINT Max_value,F_max
```

For Instrument BASIC

```
EXECUTE "OUTPMAX?"
PRINT READIO(8,0),READIO(8,1)
```

OUTPMIN?

Returns the minimum point value and its stimulus within the specified range. (Query only)

Syntax OUTPMIN?

Query *MIN, f_{min}*

Response Where,

| Register | Parameter | Description |
|----------|------------------------|--|
| 0 | <i>MIN</i> | Minimum value |
| 1 | <i>f_{min}</i> | Stimulus at minimum point (Frequency or Power) |

OUTPMINMAX?

Returns the maximum and minimum values and their stimulus values within the specified range. (Query only)

Syntax OUTPMINMAX?

Query $MIN, f_{min}, MAX, f_{max}$

Response Where,

| Register | Parameter | Description |
|----------|-----------|--|
| 0 | MIN | Minimum value |
| 1 | f_{min} | Stimulus at minimum point (Frequency or Power) |
| 2 | MAX | Maximum value |
| 3 | f_{max} | Stimulus at maximum point (Frequency or Power) |

Examples For External Controller

```
OUTPUT @Hp4396;"OUTPMINMAX?"
ENTER @Hp4396;Min_value,F_min,Max_value,F_max
PRINT "MIN:",Min_value,F_min
PRINT "MAN:",Max_value,F_max
```

For Instrument BASIC

```
EXECUTE "OUTPMINMAX?"
PRINT "MIN:",READIO(8,0),READIO(8,1)
PRINT "MAX:",READIO(8,2),READIO(8,3)
```

OUTPMEAN?

Returns the mean value within the specified range. (Query only)

Syntax OUTPMEAN?

Query $mean$

Response Where,

| Register | Parameter | Description |
|----------|-----------|-------------|
| 0 | $mean$ | Mean value. |

Examples For External Controller

```
OUTPUT @Hp4396;"OUTPMEAN?"
ENTER @Hp4396;Mean
PRINT Mean
```

For Instrument BASIC

```
EXECUTE "OUTPMEAN?"
PRINT READIO(8,0)
```

PEAK?

Returns maximum peak and its stimulus within the specified range. (Query only)

Syntax PEAK?

Query $MAX_{peak}, f_{maxpeak}$

Response Where,

| Register | Parameter | Description |
|----------|---------------|--------------------------|
| 0 | MAX_{peak} | Maximum peak value |
| 1 | $f_{maxpeak}$ | Stimulus at maximum peak |

Semantics ■ The analyzer defines the searched value and point as a reference point for the next NEXPK? command. The reference point is stored using `(SAVE) ALL` or `STATE ONLY`.

Note ■ If the search fails, the analyzer returns 0,0.

Examples For External Controller

```
OUTPUT @Hp4396;"PEAK?"
ENTER @Hp4396;Peak,F_maxpeak
PRINT "Peak:",Peak," [dB]"," ,F_maxpeak," [Hz]"
```

For Instrument BASIC

```
EXECUTE "PEAK?"
PRINT "Peak:",READIO(8,0)," [dB]"," ,READIO(8,1)," [Hz]"
```

NEXPK?

Returns the maximum peak having a value less than the value that was found using last PEAK? or NEXPK? command within the specified range. It also returns the corresponding stimulus value. (Query only)

Syntax NEXPK?

Query $Peak, f_{Peak}$

Response Where,

| Register | Parameter | Description |
|----------|------------|---------------------|
| 0 | $Peak$ | Searched peak value |
| 1 | f_{Peak} | Searched stimulus |

Note

- The analyzer defines the searched value and point as a reference point for the next NEXPK? command. The reference point is stored using `(SAVE) ALL` or `STATE ONLY`.
- If the multiple corresponded points are found, the analyzer returns right-hand nearest peak of the reference point.
- If the search fails, the analyzer returns 0,0.

Examples For External Controller

```
OUTPUT @Hp4396;"NEXPK?"
ENTER @Hp4396;N_peak,F_npeak
PRINT N_peak,F_npeak
```

For Instrument BASIC

```
EXECUTE "PEAK?"
I=1
REPEAT
  PRINT I,READIO(8,0),READIO(8,1)
  EXECUTE "NEXPK?"
  I=I+1
UNTIL READIO(8,0)=0
```

NUMLMAX?

Returns the number of positive peaks within the specified range. (Query only)

Syntax NUMLMAX?

Query *n*

Response Where,

| Register | Parameter | Description |
|----------|-----------|-----------------|
| 0 | <i>n</i> | Number of peaks |

Note ■ If the search fails, the analyzer returns 0.

Examples For External Controller

```
OUTPUT @Hp4396;"NUMLMAX?"  
ENTER @Hp4396;N  
PRINT N
```

For Instrument BASIC

```
EXECUTE "NUMLMAX?"  
PRINT READIO(8,0)
```

NUMLMIN?

Returns the number of negative peaks within the specified range. (Query only)

Syntax NUMLMIN?

Query *n*

Response Where,

| Register | Parameter | Description |
|----------|-----------|--------------------------|
| 0 | <i>n</i> | Number of negative peaks |

Note ■ If the search fails, the analyzer returns 0.

LMAX?

Returns the n th positive peak counted from the left end of the range.

Syntax LMAX? n

Where,

| Register | Parameter | Description |
|----------|-----------|--|
| 0 | n | Peak counted from the left end of the range. |

Query $LMAX_n$

Response Where,

| Register | Parameter | Description |
|----------|-----------|----------------------|
| 0 | $LMAX_n$ | Value of n th peak |

Note ■ If the search fails, the analyzer returns 3.40282346639E+38.

Examples For External Controller

```
OUTPUT @Hp4396;"LMAX? 5"  
ENTER @Hp4396;Lmax  
PRINT Lmax
```

For Instrument BASIC

```
INPUT "?",N  
WRITEIO 8,0;N  
EXECUTE "LMAX?"  
PRINT READIO(8,0)
```

LMIN?

Returns the n th negative peak counted from the left end of the range.

Syntax LMIN? n

Where,

| Register | Parameter | Description |
|----------|-----------|---|
| 0 | n | Negative peak counted from the left end of the range. |

Query $LMIN_n$

Response Where,

| Register | Parameter | Description |
|----------|-----------|-------------------------------|
| 0 | $LMIN_n$ | Value of n th negative peak |

Note ■ If the search fails, the analyzer returns 3.40282346639E+38.

TARR?

Searches to the right for the point having the specified parameter-value from the left end of the range, and returns its stimulus.

Syntax TARR? *target*

Where,

| Register | Parameter | Description |
|----------|---------------|---------------|
| 0 | <i>target</i> | Search value. |

Query *f_{target}*

Response Where,

| Register | Parameter | Description |
|----------|---------------------------|------------------------------------|
| 0 | <i>f_{target}</i> | Stimulus of the first point found. |

Note ■ If the search fails, the analyzer returns 0.

Examples For External Controller

```
INPUT "Enter Target Value.",Target
OUTPUT @Hp4396;"TARR? ";Target
ENTER @Hp4396;F_target
PRINT F_target
```

For Instrument BASIC

```
INPUT "Enter Target Value.",Target
WRITEIO 8,0;Target
EXECUTE "TARR?"
PRINT READIO(8,0)
```

TARL?

Searches to the left for the point having the specified parameter-value from the right end of the range, and returns its stimulus.

Syntax TARL? *target*

Where,

| Register | Parameter | Description |
|----------|---------------|---------------|
| 0 | <i>target</i> | Search value. |

Query *f_{target}*

Response Where,

| Register | Parameter | Description |
|----------|---------------------------|------------------------------------|
| 0 | <i>f_{target}</i> | Stimulus of the first point found. |

Note ■ If the search fails, the analyzer returns 0.

Ripple Analysis Commands

Ripple analysis commands analyze the ripples of the waveform and return the results.

- RPLPP?
- RPLHEI?
- RPLRHEI?
- RPLLHEI?
- RPLENV?
- RPLMEA?
- RPLVAL?
- POLE?

RPLPP?

Returns the maximum difference between the positive peak and the negative peak within the specified range. (Query only)

Syntax RPLPP?

Query MAX_{diff}

Response Where,

| Register | Parameter | Description |
|----------|--------------|--|
| 0 | MAX_{diff} | Maximum difference between positive and negative peak. |

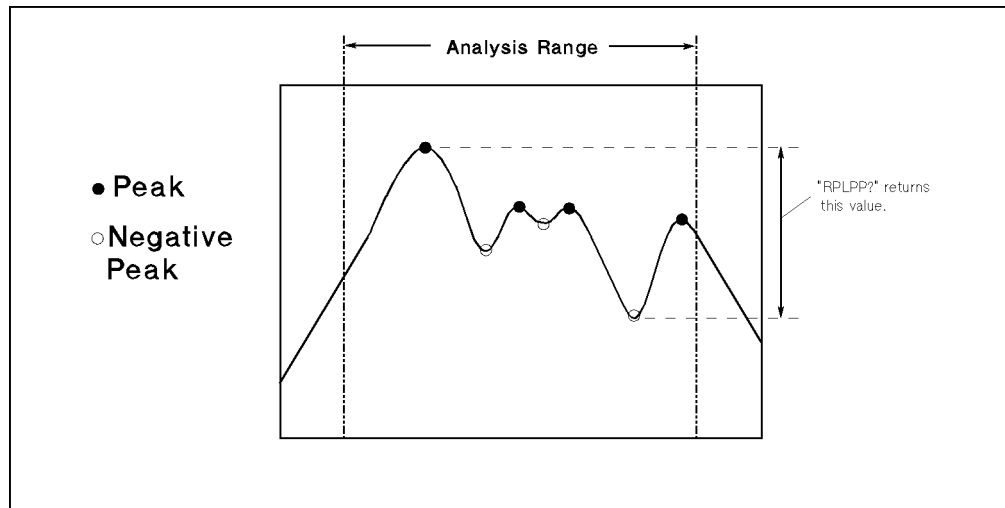


Figure I-2. RPLPP?

Note ■ If the search fails, the analyzer returns 0.

Examples For External Controller

```

ASSIGN @Hp4396 TO 717
OUTPUT @Hp4396;"ANAOCH1;ANARFULL;ANAODATA"
OUTPUT @Hp4396;"RPLPP?"
ENTER @Hp4396;Max_diff
PRINT Max_diff;"[dB]"
END

```

For Instrument BASIC

```
EXECUTE "ANAOCH1"  
EXECUTE "ANARFULL"  
EXECUTE "ANAODATA"  
EXECUTE "RPLPP?"  
PRINT READIO(8,0); "[dB]"  
END
```

RPLHEI?

Returns the maximum difference between adjacent positive and negative peaks. (Query only)

Syntax RPLHEI?

Query *value*

Response Where,

| Register | Parameter | Description |
|----------|--------------|--|
| 0 | <i>value</i> | Maximum difference between adjacent positive and negative peaks. |

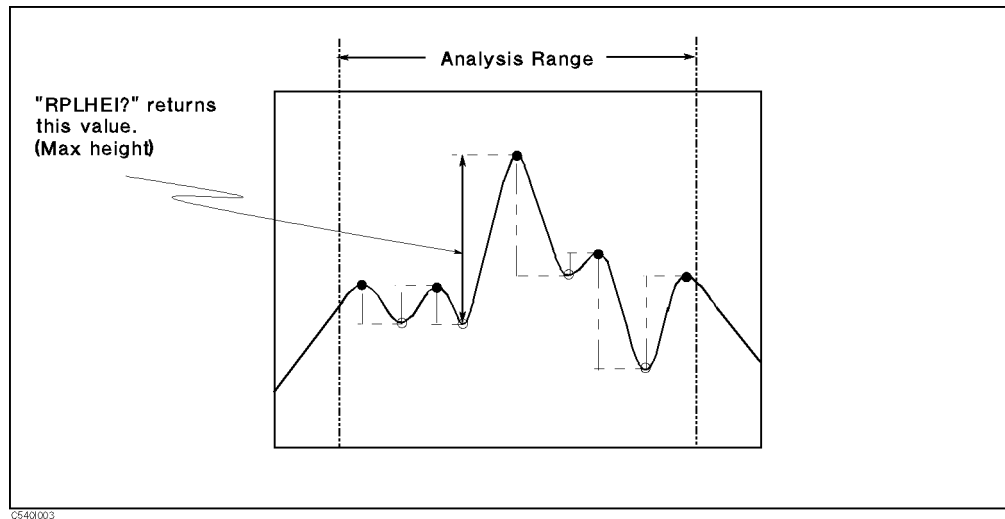


Figure I-3. RPLHEI?

Note ■ If the search fails, the analyzer returns 0.

Examples For External Controller

```
ASSIGN @Hp4396 TO 717
OUTPUT @Hp4396;"ANAOCH1;ANARFULL;ANAODATA"
OUTPUT @Hp4396;"RPLHEI?"
ENTER @Hp4396;Adj_diff
PRINT Adj_diff;"[dB]"
END
```

For Instrument BASIC

```
EXECUTE "ANAOCH1"
EXECUTE "ANARFULL"
EXECUTE "ANAODATA"
EXECUTE "RPLHEI?"
PRINT READIO(8,0);"[dB]"
END
```

RPLRHEI?

Returns the maximum difference between the positive peak and the right-hand adjacent negative peak. (Query only)

Syntax RPLRHEI?

Query *value*

Response Where,

| Register | Parameter | Description |
|----------|--------------|---|
| 0 | <i>value</i> | Maximum difference between the positive peak and the right-hand adjacent negative peak. |

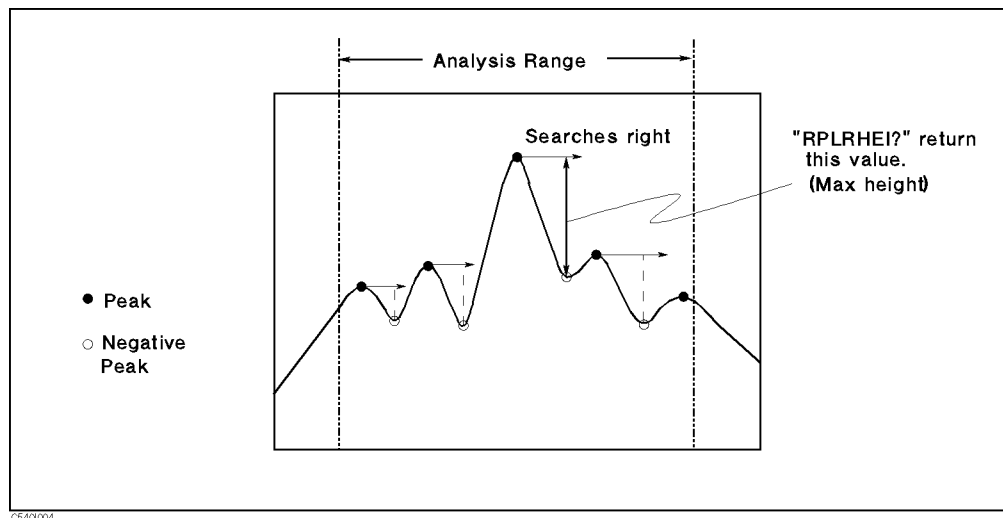


Figure I-4. RPLRHEI?

Note ■ If the search fails, the analyzer returns 0.

Examples For External Controller

```
ASSIGN @Hp4396 TO 717
OUTPUT @Hp4396;"ANAOCH1;ANARFULL;ANAODATA"
OUTPUT @Hp4396;"RPLRHEI?"
ENTER @Hp4396;Adj_diff
PRINT Adj_diff;"[dB]"
END
```

For Instrument BASIC

```
EXECUTE "ANAOCH1"
EXECUTE "ANARFULL"
EXECUTE "ANAODATA"
EXECUTE "RPLRHEI?"
PRINT READIO(8,0);"[dB]"
END
```

RPLLHEI?

Returns the maximum difference between the positive peak and the left-hand adjacent negative peak. (Query only)

Syntax RPLLHEI?

Query *value*

Response Where,

| Register | Parameter | Description |
|----------|--------------|--|
| 0 | <i>value</i> | Maximum difference between the positive peak and the left-hand adjacent negative peak. |

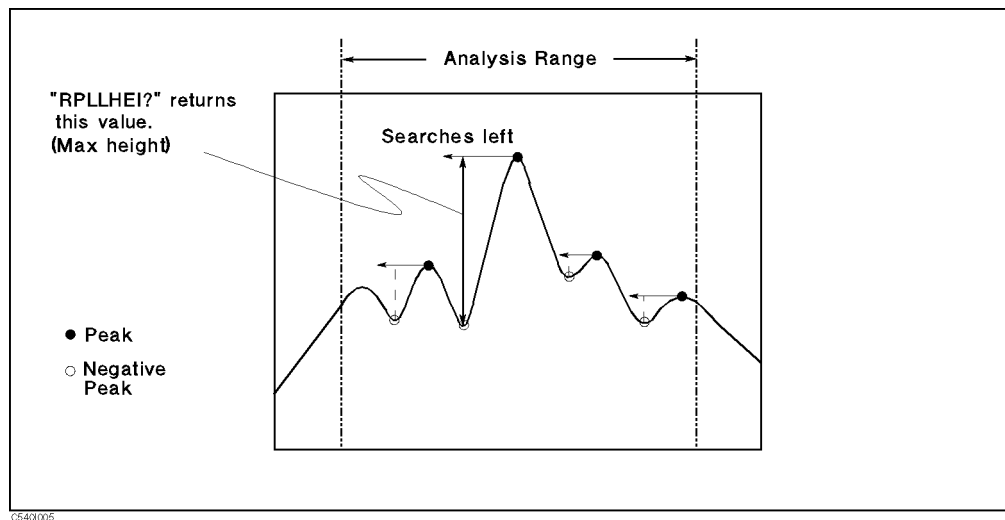


Figure I-5. RPLLHEI?

Note ■ If the search fails, the analyzer returns 0.

Examples For External Controller

```
ASSIGN @Hp4396 TO 717
OUTPUT @Hp4396;"ANAOCH1;ANARFULL;ANAODATA"
OUTPUT @Hp4396;"RPLLHEI?"
ENTER @Hp4396;Adj_diff
PRINT Adj_diff;"[dB]"
END
```

For Instrument BASIC

```
EXECUTE "ANAOCH1"
EXECUTE "ANARFULL"
EXECUTE "ANAODATA"
EXECUTE "RPLLHEI?"
PRINT READIO(8,0);"[dB]"
END
```

RPLENV?

Returns the maximum height between the negative peak and the intersection of an imaginary slope line between the adjacent positive peaks. (Query only)

Syntax RPLENV?

Query *value*

Response Where,

| Register | Parameter | Description |
|----------|--------------|---|
| 0 | <i>value</i> | Maximum height between the negative peak and the intersection of an imaginary slope line between the adjacent positive peaks. (See Figure I-6.) |

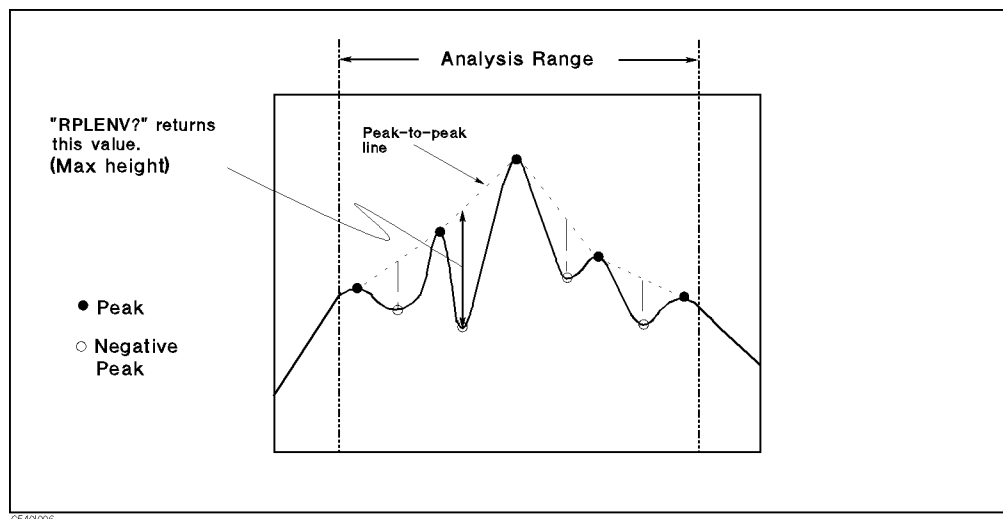


Figure I-6. RPLENV?

Note ■ If the search fails, the analyzer returns 0.

Examples For External Controller

```
ASSIGN @Hp4396 TO 717
OUTPUT @Hp4396;"ANAOCH1;ANARFULL;ANAODATA"
OUTPUT @Hp4396;"RPLENV?"
ENTER @Hp4396;Env_diff
PRINT Env_diff;"[dB]"
END
```

For Instrument BASIC

```
EXECUTE "ANAOCH1"
EXECUTE "ANARFULL"
EXECUTE "ANAODATA"
EXECUTE "RPLENV?"
PRINT READIO(8,0);"[dB]"
END
```

RPLMEA?

Returns the mean of the difference between the adjacent positive and negative peaks within the specified range. (Query only)

Syntax RPLMEA?

Query *value*

Response Where,

| Register | Parameter | Description |
|----------|--------------|---|
| 0 | <i>value</i> | Mean of the difference between the adjacent positive and negative peaks. (See Figure I-7) |

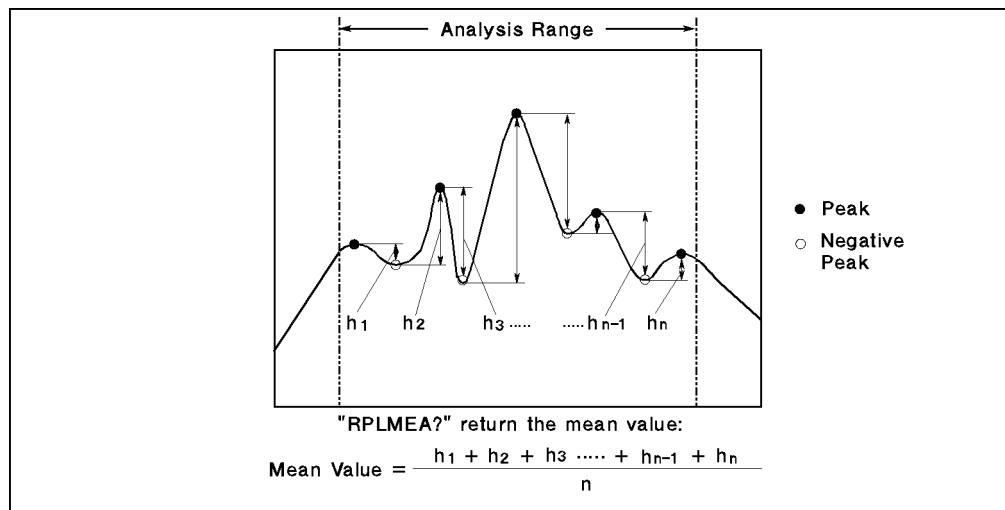


Figure I-7. RPLMEA?

Note ■ If the search fails, the analyzer returns 0.

Examples For External Controller

```

ASSIGN @Hp4396 TO 717
OUTPUT @Hp4396;"ANAOCH1;ANARFULL;ANAODATA"
OUTPUT @Hp4396;"RPLMEA?"
ENTER @Hp4396;Mean_diff
PRINT Mean_diff;"[dB]"
END

```

For Instrument BASIC

```

EXECUTE "ANAOCH1"
EXECUTE "ANARFULL"
EXECUTE "ANAODATA"
EXECUTE "RPLMEA?"
PRINT READIO(8,0);"[dB]"
END

```

RPLVAL?

Returns the maximum total of the differences between the negative peaks and the adjacent positive peaks on both sides and the stimulus of the corresponding negative peak. (Query only)

Syntax RPLVAL?

Query *Rpl_{val}*, *stimulus*

Response Where,

| Register | Parameter | Description |
|----------|--------------------------|---|
| 0 | <i>Rpl_{val}</i> | Maximum total of the differences between the negative peaks and the adjacent positive peaks on both sides. (See Figure I-8) |
| 1 | <i>stimulus</i> | Stimulus of the corresponding negative peak |

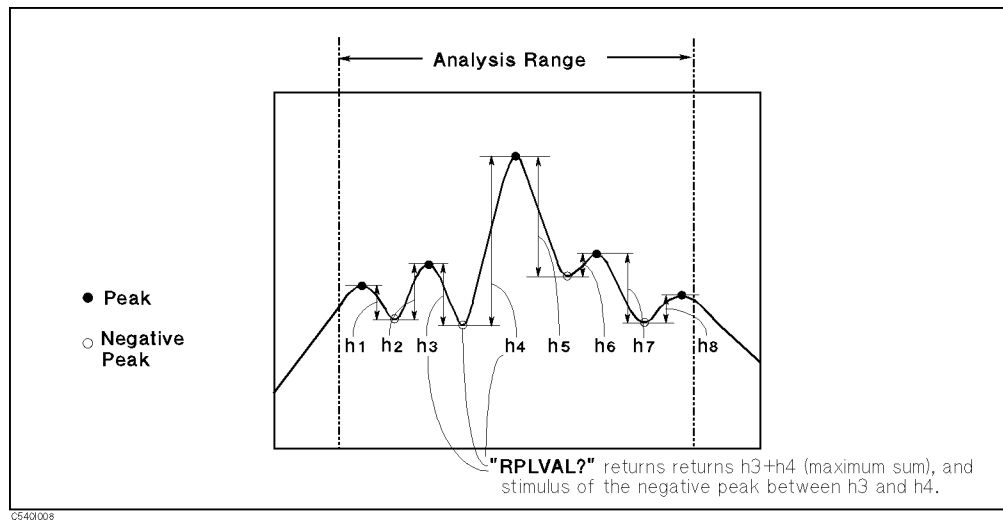


Figure I-8. RPLVAL?

Note ■ If the search fails, the analyzer returns 0.

Examples For External Controller

```
ASSIGN @Hp4396 TO 717
OUTPUT @Hp4396;"ANAOCH1;ANARFULL;ANAODATA"
OUTPUT @Hp4396;"RPLVAL?"
ENTER @Hp4396;Val,Stim
PRINT Val;"[dB]";Stim;"[Hz]"
END
```

For Instrument BASIC

```
EXECUTE "ANAOCH1"
EXECUTE "ANARFULL"
EXECUTE "ANAODATA"
EXECUTE "RPLVAL?"
PRINT READIO(8,0);"[dB]"
PRINT READIO(8,1);"[Hz]"
END
```


POLE?

Returns the stimulus and value of the first negative peak found on each side of the maximum point that are below the specified value from the maximum peak. (Query only)

Syntax POLE? *D*

Where,

| Register | Parameter | Description |
|----------|-----------|-----------------------------------|
| 0 | <i>D</i> | Difference from the maximum peak. |

Query $x_1, stim_1, x_2, stim_2$

Response Where,

| Register | Parameter | Description |
|----------|-----------|----------------------------|
| 0 | x_1 | Left negative peak value. |
| 1 | $stim_1$ | Stimulus of x_1 . |
| 2 | x_2 | Right negative peak value. |
| 3 | $stim_2$ | Stimulus of x_2 . |

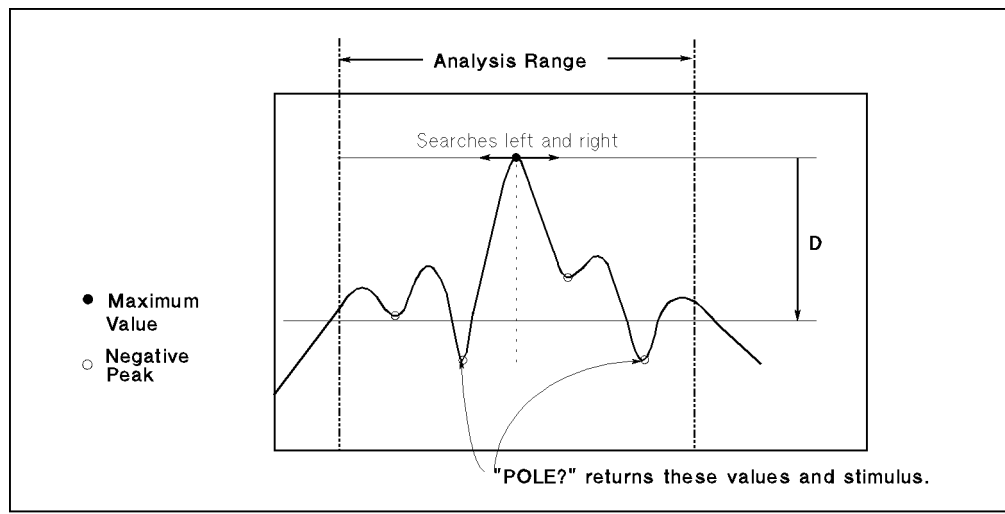


Figure I-9. POLE?

- Note**
- If the search fails, the analyzer returns 0.
 - Give the command parameter as a negative value. For instance, to specify 50 dB down from the maximum peak as a reference level, the parameter is -50.

Examples For External Controller

```

ASSIGN @Hp4396 TO 717
OUTPUT @Hp4396;"ANAOCH1;ANARFULL;ANAODATA"
OUTPUT @Hp4396;"POLE? -50"
ENTER @Hp4396;X1,S1,X2,S2
PRINT "LEFT :";X1;"[dB]";S1;"[Hz]"
PRINT "RIGHT:";X2;"[dB]";S2;"[Hz]"

```

END

For Instrument BASIC

```
EXECUTE "ANAOCH1"  
EXECUTE "ANARFULL"  
EXECUTE "ANAODATA"  
WRITEIO 8,0;-50  
EXECUTE "POLE?"  
PRINT "LEFT :";READIO(8,0);"[dB]";READIO(8,1);"[Hz]"  
PRINT "RIGHT:";READIO(8,2);"[dB]";READIO(8,3);"[Hz]"  
END
```

Filter and Resonator Analysis Commands

The following commands are device related. They are easy to use for specific device analysis because they can output many parameters using only a single command.

- `OUTPFILT?`
- `OUTPXFIL?`
- `OUTPCFIL?`
- `OUTPRESO?`
- `OUTPRESR?`
- `OUTPRESF?`
- `OUTPCERR?`

OUTPFILT?

Analyzes the filter and returns the parameters.

Syntax OUTPFILT? *x*

Where,

| Register | Parameter | Description |
|----------|-----------|---|
| 0 | <i>x</i> | The dB value down the bandwidth filter. |

Query Response *Loss, BW, f_{cent}, Q, Δf_{left}, Δf_{right}* (Total6)

Response

| Register | Parameter | Description |
|----------|---------------------------|--|
| 0 | <i>Loss</i> | Insertion loss |
| 1 | <i>BW</i> | <i>x</i> dB down bandwidth |
| 2 | <i>f_{cent}</i> | Center frequency |
| 3 | <i>Q</i> | Q (Quality factor) |
| 4 | <i>Δf_{left}</i> | Frequency difference between the left cutoff point and the middle of the range. |
| 5 | <i>Δf_{right}</i> | Frequency difference between the right cutoff point and the middle of the range. |

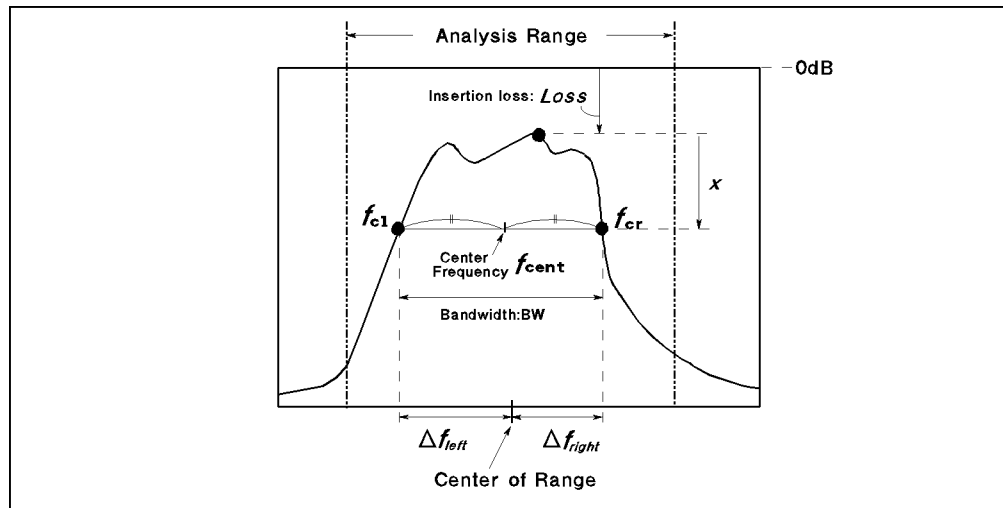


Figure I-10. OUTPFILT?

- Semantics**
- Insertion loss is the maximum value within the specified range.
 - *x* dB bandwidth is the frequency difference between both of the *x*dB down cutoff points.
 - Center frequency is the middle point of both cutoff points.
 - *Q* is calculated using the following equation:

$$Q = \frac{\sqrt{f_{cl} \times f_{cr}}}{BW}$$

Note

- If both of the two cutoff points are not found, the analyzer returns 0 for all values of the query response.

Examples For External Controller

```

10 ASSIGN @Hp4396 TO 717
20 CALL Sweep(1) ! Goes to the subroutine.
30 OUTPUT @Hp4396;"ANAOCH1;ANARFULL;ANAODATA"
40 OUTPUT @Hp4396;"OUTPFILT? -3"
50 ENTER @Hp4396;Loss,Bw,Fc,Q,Dfl,Dfr
60 PRINT "Loss: ";Loss;"[dB] BW: ";Bw;"[Hz]"
70 PRINT "fc: ";Fc;"[Hz] Q: ";Q
80 PRINT "Dfl: ";Dfl;"[Hz] Dfr: ";Dfr;"[Hz]"
90 END
100 SUB Sweep(Ch)! Sweep End Detection Subroutine
101         ! (Parameter: No. of channel)
110  ASSIGN @Hp4396 TO 717
120  ON INTR 7 GOTO Sweep_end
130  OUTPUT @Hp4396;"TRGS BUS"
140  OUTPUT @Hp4396;"ESNB 2; *SRE 4"
150  FOR I=1 TO Ch
160  OUTPUT @Hp4396;"*CLS;*OPC?"
170  ENTER @Hp4396;0pc
180  ENABLE INTR 7;2
190  TRIGGER @Hp4396
200  Waiting:GOTO Waiting
210  Sweep_end:!
220  NEXT I
230  SUBEND

```

For Instrument BASIC

```

10 ASSIGN @Hp4396 TO 800
20 OUTPUT @Hp4396;"CENT 70MHZ; SPAN 100KHZ"
30 EXECUTE "SING"
40 EXECUTE "ANAOCH1"
50 EXECUTE "ANARFULL"
60 EXECUTE "ANAODATA"
70 WRITEIO 8,0;-50
80 EXECUTE "POLE?"
90 PRINT "Loss: ";READIO(8,0);"[dB] BW: ";READIO(8,1);"[Hz]"
100 PRINT "fc: ";READIO(8,2);"[Hz] Q: ";READIO(8,3)
110 PRINT "Dfl: ";READIO(8,4);"[Hz] Dfr: ";READIO(8,5);"[Hz]"
120 END

```

OUTPXFIL?

Syntax OUTPXFIL? x_1, x_2, D, f_1, f_2

Where,

| Register | Parameter | Description |
|----------|-----------|---|
| 0 | x_1 | The dB value down the bandwidth filter. (1) x_1 [dB] |
| 1 | x_2 | The dB value down the bandwidth filter. (2) x_2 [dB] |
| 2 | D | Difference from maximum value. (Same as POLE? parameter.) |
| 3 | f_1 | Stop frequency of the range for the rejection level. |
| 4 | f_2 | Start frequency of the range for the spurious level. |

Query Response $Loss, BW, f_{cent}, Q, \Delta f_{left1}, \Delta f_{right1}, \Delta f_{left2}, \Delta f_{right2}, Pass, Reject, Spurious, Pole_{x1}, Pole_{stim1}, Pole_{x2}, Pole_{stim2}$ (15)

| Register | Parameter | Description |
|----------|---------------------|--|
| 0 | $Loss$ | Insertion loss |
| 1 | BW | x_1 dB down bandwidth |
| 2 | f_{cent} | Center frequency |
| 3 | Q | Q |
| 4 | Δf_{left} | Frequency difference between the left cutoff point (f_{cl}) and the middle of the range. |
| 5 | Δf_{right} | Frequency difference between the right cutoff point (f_{cr}) and the middle of the range. |
| 6 | Δf_{left2} | Frequency difference between the left cutoff point (f_{cl2}) and the middle of the range. |
| 7 | Δf_{right2} | Frequency difference between the right cutoff point (f_{cr2}) and the middle of the range. |
| 8 | $Pass$ | Passband ripple |
| 9 | $Reject$ | Rejection level |
| 10 | $Spurious$ | Spurious level |
| 11 | $Pole_{x1}$ | First negative peak found to the left of the maximum point. |
| 12 | $Pole_{stim1}$ | Stimulus of $Pole_{x1}$. |
| 13 | $Pole_{x2}$ | First negative peaks found to the right of the maximum point. |
| 14 | $Pole_{stim2}$ | Stimulus of $Pole_{x2}$. |

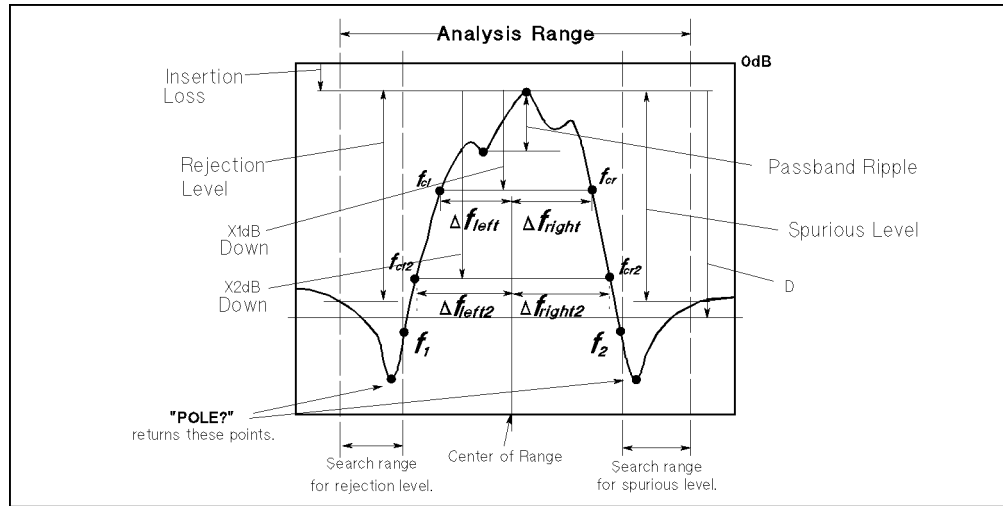


Figure I-11. OUTPXFIL?

- Semantics**
- Insertion loss, x_1 dB bandwidth, center frequency, Q , Δf_{left} , and Δf_{right} are the same as the responses of OUTPFILT?.
 - Δf_{left2} and Δf_{right2} are the frequency differences between both sides at the x_2 dB down cutoff points (f_{cl2} and f_{cr2}) and the middle of the range.
 - Passband ripple is the frequency difference of the maximum positive peak and the minimum negative peak between the x_1 dB down cutoff points (f_{cl} , f_{cr}).
 - Rejection level is the frequency difference from the insertion loss to the maximum level in the range from the left edge of analysis range to f_1 .
 - Spurious level is the frequency difference from the insertion loss to the maximum level between f_2 and the right edge of analysis range.
 - $Pole_{x1}$, $Pole_{stim1}$, $Pole_{x2}$, $Pole_{stim2}$ are the same as the query response of POLE? with the parameter D .

- Note**
- If both of the two x_1 dB down cutoff points are not found, the analyzer returns 0 for all values of the query response.
 - If both of the two x_2 dB down cutoff points are not found, the analyzer returns 0 for Δf_{left2} and Δf_{right2} .
 - If the corresponding peak for POLE? is not found, the analyzer returns 0 for $Pole_{x1}$, $Pole_{stim1}$, $Pole_{x2}$, and $Pole_{stim2}$.

Examples For External Controller

```

10 ASSIGN @Hp4396 TO 717
20 OUTPUT @Hp4396;"CENT 70MHZ; SPAN 100KHZ"
30 CALL Sweep(1) ! Goes to sub routine.
40 OUTPUT @Hp4396;"ANAOCH1;ANARFULL;ANAODATA"
50 OUTPUT @Hp4396;"OUTPXFIL? -3,-10,-50,69.98MHz,70.02MHz"
60 ENTER @Hp4396;Loss,Bw,Fc,Q,Df1,Dfr,Df12,Dfr2,Pass,Reject,
Spurious,Pole1,Fp1,Pole2,Fp2
70 PRINT "Loss:";Loss;"[dB] BW:";Bw;"[Hz] fc:";Fc;"[Hz]"
80 PRINT "Q:";Q;" Df1:";Df1;"[Hz] Dfr:";Dfr;"[Hz]"
90 PRINT "Df12:";Df12;"[Hz] Dfr2:";Dfr2;"[Hz] Pass:";Pass;"[dB]"

```

```

100 PRINT "Reject: ";Reject;"[dB] Spurious: ";Spurious;" [dB]"
110 PRINT "Pole (left): ";Pole1;"[dB] ";Fp1;"[Hz]"
120 PRINT "Pole (right): ";Pole2;"[dB] ";Fp2;"[Hz]"
130 END

```

For Instrument BASIC

```

10 ASSIGN @Hp4396 TO 800
20 OUTPUT @Hp4396;"CENT 70MHZ; SPAN 100KHZ"
30 EXECUTE "SING"
40 EXECUTE "ANAOCH1"
50 EXECUTE "ANARFULL"
60 EXECUTE "ANAODATA"
70 WRITEIO 8,0;-3
80 WRITEIO 8,1;-10
90 WRITEIO 8,2;-50
100 WRITEIO 8,3;6.998E+7
110 WRITEIO 8,4;7.002E+7
120 EXECUTE "OUTPXFIL?"
130 PRINT "Passband Ripple: ";READIO(8,8);" [dB]"
140 PRINT "Rejection Level: ";READIO(8,9);" [dB]"
150 PRINT "Spurious Level: ";READIO(8,10);" [dB]"
160 END

```


OUTPCFIL?

Analyzes the filter at the nominal frequency, and returns the parameters.

Syntax OUTPCFIL? $f_c, x_1, x_2, D, f_1, f_2$

Where,

| Register | Parameter | Description |
|----------|-----------|---|
| 0 | f_c | Nominal frequency |
| 1 | x_1 | The dB value down the bandwidth filter. (1) x_1 [dB] |
| 2 | x_2 | The dB value down the bandwidth filter. (2) x_2 [dB] |
| 3 | D | Difference from maximum value. (Same as POLE? parameter.) |
| 4 | f_1 | Stop frequency of the range for the rejection level. |
| 5 | f_2 | Start frequency of the range for the spurious level. |

Query Response $Loss, Loss_c, BW, f_{cent}, Q, \Delta f_{left1}, \Delta f_{right1}, \Delta f_{left2}, \Delta f_{right2}, Pass, Reject, Spurious, Pole_{x1}, Pole_{stim1}, Pole_{x2}, Pole_{stim2}$ (Total 16)

| Register | Parameter | Description |
|----------|---------------------|---|
| 0 | $Loss$ | Insertion loss |
| 1 | $Loss_c$ | Const Loss |
| 2 | BW | x_1 dB down bandwidth |
| 3 | f_{cent} | Center frequency |
| 4 | Q | Q |
| 5 | Δf_{left} | Frequency difference between the left cutoff point (f_{cl}) and the middle of the range. |
| 6 | Δf_{right} | Frequency difference between the right cutoff point (f_{cr}) and the middle of the range. |
| 7 | Δf_{left2} | Frequency difference between the left cutoff point (f_{cl2}) and the middle of the range. |
| 8 | Δf_{right2} | Frequency difference between the left cutoff point (f_{cr2}) and the middle of the range. |
| 9 | $Pass$ | Passband ripple |
| 10 | $Reject$ | Rejection level |
| 11 | $Spurious$ | Spurious level |
| 12 | $Pole_{x1}$ | First negative peaks found to the left of the maximum point. |
| 13 | $Pole_{stim1}$ | Stimulus of $Pole_{x1}$. |
| 14 | $Pole_{x2}$ | First negative peak found to the right of the maximum point. |
| 15 | $Pole_{stim2}$ | Stimulus of $Pole_{x2}$. |

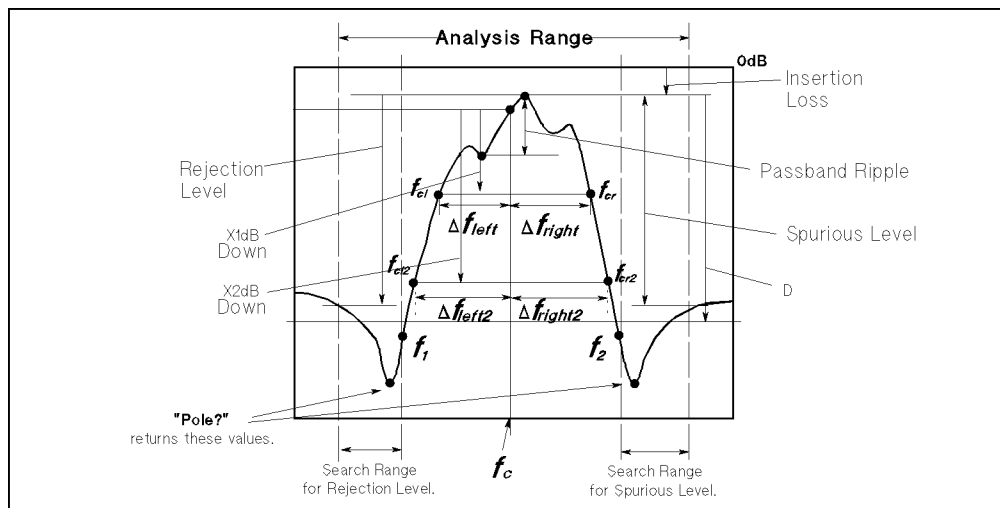


Figure I-12. OUTPCFIL?

- Semantics**
- Insertion loss, rejection level, spurious level, $Pole_{x1}$, $Pole_{stim1}$, $Pole_{x2}$, and $Pole_{stim2}$ are the same as the responses of OUTPCFIL?.
 - The const loss is the value of the point that is specified by command parameter, f_c .
 - x_1 dB bandwidth is the frequency difference between two x_1 dB down cutoff points (f_{cl} , f_{cr}) from the const loss point.
 - Center frequency is the middle point of f_{cl} and f_{cr} .
 - Q is calculated using the following equation:

$$Q = \frac{\sqrt{f_{cl} \times f_{cr}}}{BW}$$

- Δf_{left} and Δf_{right} are the frequency differences between both sides at the x_1 dB down cutoff points (f_{cl} and f_{cr}) and f_c .
- Δf_{left2} and Δf_{right2} are the frequency differences between both sides at the x_2 dB down cutoff points (f_{cl2} and f_{cr2}) and f_c .
- Passband ripple is the frequency difference of maximum positive peak and minimum negative peak between x_1 dB down cutoff points (f_{cl1} , f_{cr1}).

- Note**
- If both of the two x_1 dB down cutoff points are not found, the analyzer returns 0 for all values of the query response.
 - If both of the two x_2 dB down cutoff points are not found, the analyzer returns 0 for Δf_{left2} and Δf_{right2} .
 - If the corresponding peak for POLE? is not found, the analyzer returns 0 for $Pole_{x1}$, $Pole_{stim1}$, $Pole_{x2}$, and $Pole_{stim2}$.

Examples For External Controller

```

10 ASSIGN @Hp4396 TO 717
20 OUTPUT @Hp4396;"CENT 70MHZ; SPAN 100KHZ"
30 CALL Sweep(1) ! Goes to sub routine. (See OUTPCFIL?)
40 OUTPUT @Hp4396;"ANAOCH1;ANARFULL;ANAO DATA"

```

```

50 OUTPUT @Hp4396;"OUTPCFIL? 70MHz,-3,-10,-50,69.98MHz,70.02MHz"
60 ENTER @Hp4396;Loss,Lc,Bw,Fc,Q,Df1,Dfr,Df12,Dfr2,Pass,Reject,
Spurious,Pole1,Fp1,Pole2,Fp2
70 PRINT "Loss: ";Loss;" [dB] Const Loss: ";Lc;" [dB]"
80 PRINT "BW: ";Bw;" [Hz] Fc: ";Fc;" [Hz]"
90 PRINT "Q: ";Q;" DF1: ";Df1;" [Hz] DFR: ";Dfr;" [Hz]"
100 PRINT "Df12: ";Df12;" [Hz] Dfr2: ";Dfr2;" [Hz] Pass: ";Pass;" [dB]"
110 PRINT "Reject: ";Reject;" [dB] Spurious: ";Spurious;" [dB]"
120 PRINT "Pole (left): ";Pole1;" [dB] ";Fp1;" [Hz]"
130 PRINT "Pole (right): ";Pole2;" [dB] ";Fp2;" [Hz]"
140 END

```

For Instrument BASIC

```

10 ASSIGN @Hp4396 TO 800
20 OUTPUT @Hp4396;"CENT 70MHZ; SPAN 100KHZ"
30 EXECUTE "SING"
40 EXECUTE "ANAOCH1"
50 EXECUTE "ANARFULL"
60 EXECUTE "ANAODATA"
70 WRITEIO 8,0;7.E+7
80 WRITEIO 8,1;-3
90 WRITEIO 8,2;-10
100 WRITEIO 8,3;-50
110 WRITEIO 8,4;6.998E+7
120 WRITEIO 8,5;7.002E+7
130 EXECUTE "OUTPCFIL?"
140 PRINT "Const Loss: ";READIO(8,1);" [dB]"
150 END

```

OUTPRESO?

Returns resonator specific parameters. (Query only)

Syntax OUTPRESO?

Query Z_r, f_r, Z_a, f_a (Total 4)

Response Where,

| Register | Parameter | Description |
|----------|-----------|-------------------------|
| 0 | Z_r | Resonant impedance |
| 1 | f_r | Resonant frequency |
| 2 | Z_a | Anti-resonant impedance |
| 3 | f_a | Anti-resonant frequency |

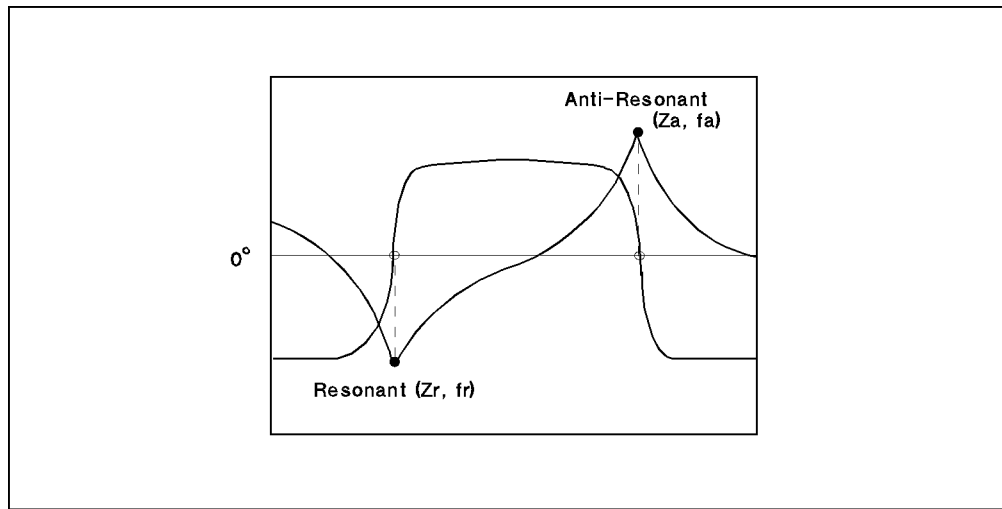


Figure I-13. OUTPRESO?

- Semantics** ■ OUTPRESO? executes the following actions and returns their values:
1. Searches for the 0° phase point from the left edge of the analysis range.
 2. Defines the first point found as the resonant point, and then returns its impedance and its frequency.
 3. Defines the next point found as the anti-resonant point, and then returns its impedance and its frequency.

- Note** ■ You must select the following conditions to use this command:
- Dual Channel & Coupled Channel: ON
 - Impedance Conversion: ON
 - Analysis channel: LOG MAG format
 - Non-analysis channel: Phase format
- OUTPRESO? returns the first two found 0° phase point events if there are more than three corresponding points.
- If there is only one 0° phase point in the range, OUTPRESO? defines that point as a resonant point and returns 0 for Z_a and f_a .

- If there is no 0° point, OUTPRES0? returns 0 for all parameters.
- If the impedance conversion is off, OUTPRES0? returns the magnitude (dB) at the 0° phase point.

Examples For External Controller

```

10 ASSIGN @Hp4396 TO 717
20 OUTPUT @Hp4396;"DUAC ON; COUC ON"
30 OUTPUT @Hp4396;"CHAN2; FMT PHAS"
40 OUTPUT @Hp4396;"CHAN1; FMT LOGM; CONV ZTRA"
50 OUTPUT @Hp4396;"CENT 70MHZ; SPAN 100KHZ"
60 CALL Sweep(2) ! Goes to sub routine. (See OUTPFILT?)
61             ! Parameter is 2 because of Dual Channel ON.
70 OUTPUT @Hp4396;"ANAOCH1;ANARFULL;ANAODATA"
80 OUTPUT @Hp4396;"OUTPRES0?"
90 ENTER @Hp4396;Zr,Fr,Za,Fa
100 PRINT "Resonant: ";Zr;" [ohm] ," ;Fr;" [Hz] "
110 PRINT "Anti-Resonant: ";Za;" [ohm] ," ;Fa;" [Hz] "
120 END

```

For Instrument BASIC

```

10 ASSIGN @Hp4396 TO 800
20 OUTPUT @Hp4396;"DUAC ON; COUC ON"
30 OUTPUT @Hp4396;"CHAN2; FMT PHAS"
40 OUTPUT @Hp4396;"CHAN1; FMT LOGM; CONV ZTRA"
50 OUTPUT @Hp4396;"CENT 70MHZ; SPAN 100KHZ"
60 EXECUTE "SING" ! This line waits for the end of both channel sweep.
70 EXECUTE "ANAOCH1"
80 EXECUTE "ANARFULL"
90 EXECUTE "ANAODATA"
100 EXECUTE "OUTPRES0?"
110 PRINT "Resonant: ";READIO(8,0);" [ohm] ";READIO(8,1);" [Hz] "
120 PRINT "Anti-Resonant: ";READIO(8,2);" [ohm] ";READIO(8,3);" [Hz] "
130 END

```

OUTPRESR?

Returns the resonator specific parameters. (Query only)

Syntax OUTPRESR?

Query $Z_r, f_r, Z_a, f_a, Rpl_1, Rpl_2, Rpl_3$ (Total 7)

Response Where,

| Register | Parameter | Description |
|----------|-----------|---|
| 0 | Z_r | Resonant impedance |
| 1 | f_r | Resonant frequency |
| 2 | Z_a | Anti-resonant impedance |
| 3 | f_a | Anti-resonant frequency |
| 4 | Rpl_1 | Maximum left height of the ripple where is on the left side of the resonant point. |
| 5 | Rpl_2 | Maximum height right of the ripple that is between the resonant and anti-resonant points. |
| 6 | Rpl_3 | Maximum height left of the ripple that is on the right side of the resonant point. |

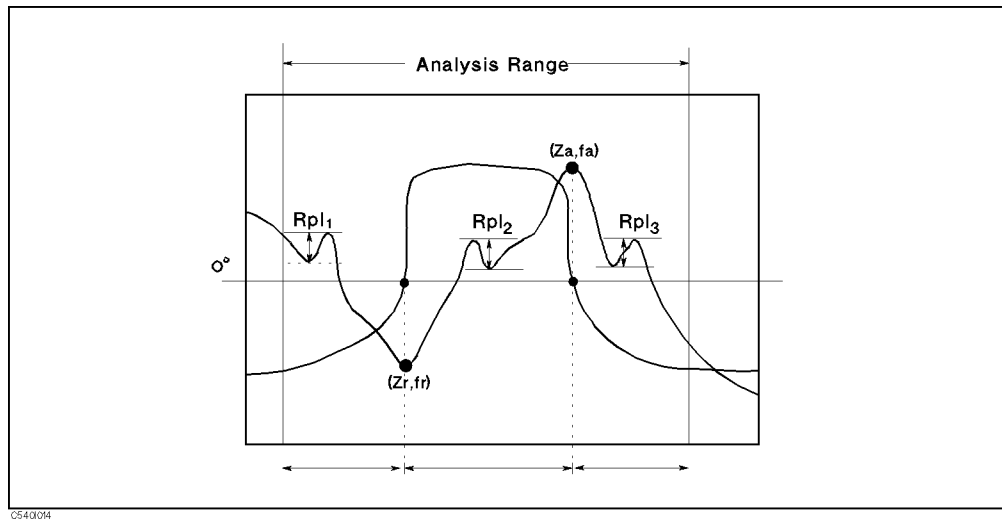


Figure I-14. OUTPRESR?

Semantics ■ OUTPRESR? executes the following actions:

1. Searches for the 0° phase point from the left edge of the analysis range.
2. Defines the first point found as the resonant point, and then returns its impedance and its frequency.
3. Defines the next point found as the anti-resonant point, and then returns its impedance and its frequency.
4. Returns the maximum height of the ripple, Rpl_1 , that is the difference between the peak and left adjacent negative peak.
5. Returns the maximum height of the ripple, Rpl_2 , that is the difference between the peak and right adjacent negative peak.

6. Returns the maximum height of the ripple, Rpl_3 , that is the difference between the peak and left adjacent negative peak.

Note

- You must select the following conditions to use this command:
 - Dual Channel & Coupled Channel: ON
 - Impedance Conversion: ON
 - Analysis channel: LOG MAG format
 - Non-analysis channel: Phase format
- OUTPRESR? returns the first two 0° phase point events found if there are more than three corresponding points points.
- If there is only one 0° phase point in the range, OUTPRESR? defines that point as a resonant point and returns 0 for Z_a , f_a , Rpl_1 , Rpl_2 , and Rpl_3 .
- If there is no 0° point, OUTPRESR? returns 0 for all parameters.

Examples For External Controller

```

10 ASSIGN @Hp4396 TO 717
20 OUTPUT @Hp4396;"DUAC ON; COUC ON"
30 OUTPUT @Hp4396;"CHAN2; FMT PHAS"
40 OUTPUT @Hp4396;"CHAN1; FMT LOGM; CONV ZTRA"
50 OUTPUT @Hp4396;"CENT 70MHZ; SPAN 100KHZ"
60 CALL Sweep(2) ! Goes to sub routine. (See OUTPFILT?)
70 OUTPUT @Hp4396;"ANAOCH1;ANARFULL;ANAODATA"
80 OUTPUT @Hp4396;"OUTPRESR?"
90 ENTER @Hp4396;Zr,Fr,Za,Fa,R1,R2,R3
100 PRINT "Resonant: ";Zr;" [ohm] ," ;Fr;" [Hz]"
110 PRINT "Anti-Resonant: ";Za;" [ohm] ," ;Fa;" [Hz]"
120 PRINT "Ripple L: ";R1;" [dB]"
130 PRINT "Ripple M: ";R2;" [dB]"
140 PRINT "Ripple R: ";R3;" [dB]"
150 END

```

For Instrument BASIC

```

10 ASSIGN @Hp4396 TO 800
20 OUTPUT @Hp4396;"DUAC ON; COUC ON"
30 OUTPUT @Hp4396;"CHAN2; FMT PHAS"
40 OUTPUT @Hp4396;"CHAN1; FMT LOGM; CONV ZTRA"
50 OUTPUT @Hp4396;"CENT 70MHZ; SPAN 100KHZ"
60 EXECUTE "SING"
70 EXECUTE "ANAOCH1"
80 EXECUTE "ANARFULL"
90 EXECUTE "ANAODATA"
100 EXECUTE "OUTPRESR?"
110 PRINT "Resonant: ";READIO(8,0);" [ohm] ," ;READIO(8,1);" [Hz]"
120 PRINT "Anti-Resonant: ";READIO(8,2);" [ohm] ," ;READIO(8,3);" [Hz]"
130 PRINT "Ripple L: ";READIO(8,4);" [dB]"
140 PRINT "Ripple M: ";READIO(8,5);" [dB]"
150 PRINT "Ripple R: ";READIO(8,6);" [dB]"
160 END

```

OUTPRESF?

Returns the resonator specific parameters. (Query only)

Syntax OUTPRESF? x_1, x_2

Where,

| Register | Parameter | Description |
|----------|-----------|-----------------------------------|
| 0 | x_1 | Value down from the maximum peak. |
| 1 | x_2 | Value above the minimum peak. |

Query Response $f_s, f_p, f_{s1}, f_{s2}, f_{p1}, f_{p2}$ (Total 6)

Where,

| Register | Parameter | Description |
|----------|-----------|---|
| 0 | f_s | Middle point frequency between f_{s1} and f_{s2} . |
| 1 | f_p | Middle point frequency between f_{p1} and f_{p2} . |
| 2 | f_{s1} | Left one of the two points x_1 dB down from the maximum peak. |
| 3 | f_{s2} | Right one of the two points x_1 dB down from the maximum peak. |
| 4 | f_{p1} | Left one of the two points x_2 dB above the minimum negative peak. |
| 5 | f_{p2} | Right one of the two points x_2 dB above the minimum negative peak. |

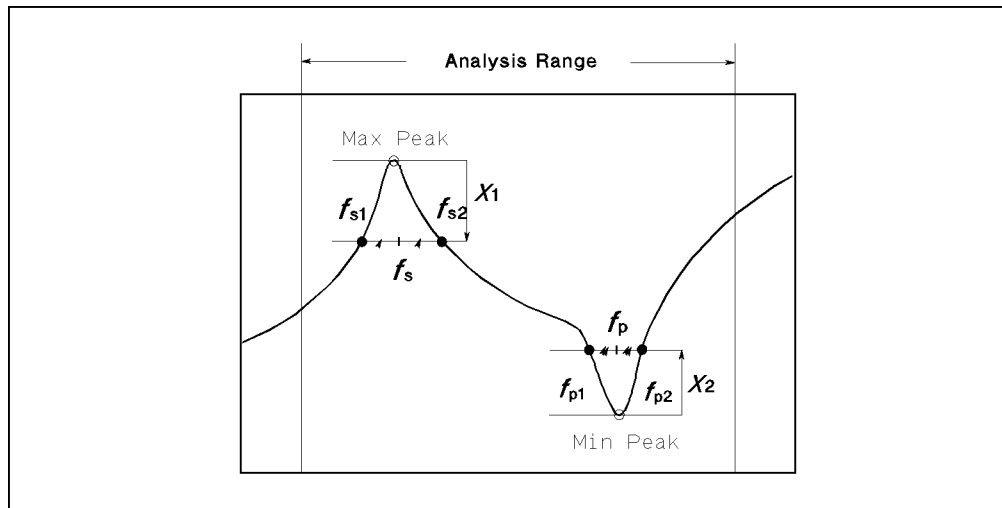


Figure I-15. OUTPRESF?

Semantics ■ OUTPRESF? executes the following actions:

1. Searches for the maximum peak in the analysis range.
2. Searches for the x_1 dB below points on both sides, and defines the first found left and right side points as f_{s1} and f_{s2} , respectively.

3. Defines the middle point between f_{s1} and f_{s2} to f_s .
4. Searches for the x_2 dB above points on both sides, and defines the first found left and right side points as f_{p1} and f_{p2} , respectively.
5. Defines the middle point between f_{p1} and f_{p2} as f_p .

Note

- If there is no corresponding peak in the range, OUTPRESF? returns 0 for all parameters.
- If the maximum peak cannot be found, OUTPRESF? returns 0 for f_s , f_{s1} , and f_{s2} .
- If the minimum negative peak cannot be found, OUTPRESF? returns 0 for f_p , f_{p1} , and f_{p2} .
- Specify the negative value for x_1 and positive value for x_2 .

Examples For External Controller

```

10 ASSIGN @Hp4396 TO 717
20 OUTPUT @Hp4396;"FMT LOGM; CENT 60.06MHz; SPAN 20kHz"
30 CALL Sweep(1) ! Goes to sub routine. (See OUTPFILT?)
40 OUTPUT @Hp4396;"ANAOCH1;ANARFULL;ANAODATA"
50 OUTPUT @Hp4396;"OUTPRESF? -3dB,3dB"
60 ENTER @Hp4396;Fs,Fp,Fs1,Fs2,Fp1,Fp2
70 PRINT "Series-Resonant: ";Fs;" [Hz]"
80 PRINT "Parallel-Resonant: ";Fp;" [Hz]"
90 END

```

For Instrument BASIC

```

10 ASSIGN @Hp4396 TO 800
20 OUTPUT @Hp4396;"FMT LOGM"
30 OUTPUT @Hp4396;"CENT 60.06MHz; SPAN 20kHz"
40 EXECUTE "SING"
50 EXECUTE "ANAOCH1"
60 EXECUTE "ANARFULL"
70 EXECUTE "ANAODATA"
80 WRITEIO 8,0;-3
90 WRITEIO 8,1;3
100 EXECUTE "OUTPRESF?"
110 PRINT "Series-Resonant: ";READIO(8,0);" [Hz]"
120 PRINT "Pararel-Resonant: ";READIO(8,1);" [Hz]"
130 END

```

OUTPCERR?

Returns the ceramic resonator specific parameters. (Query only)

Syntax OUTPCERR?

Query $Z_r, f_r, Z_a, f_a, Rpl_1, Rpl_2, Rpl_3$ (Total7)

Response Where,

| Register | Parameter | Description |
|----------|-----------|---|
| 0 | Z_r | Resonant impedance |
| 1 | f_r | Resonant frequency |
| 2 | Z_a | Anti-resonant impedance |
| 3 | f_a | Anti-resonant frequency |
| 4 | Rpl_1 | Maximum height of the ripple that is on the left side of the resonant point. |
| 5 | Rpl_2 | Maximum height of the ripple that is between the resonant and anti-resonant points. |
| 6 | Rpl_3 | Maximum height of the ripple that is on the right side of the resonant point. |

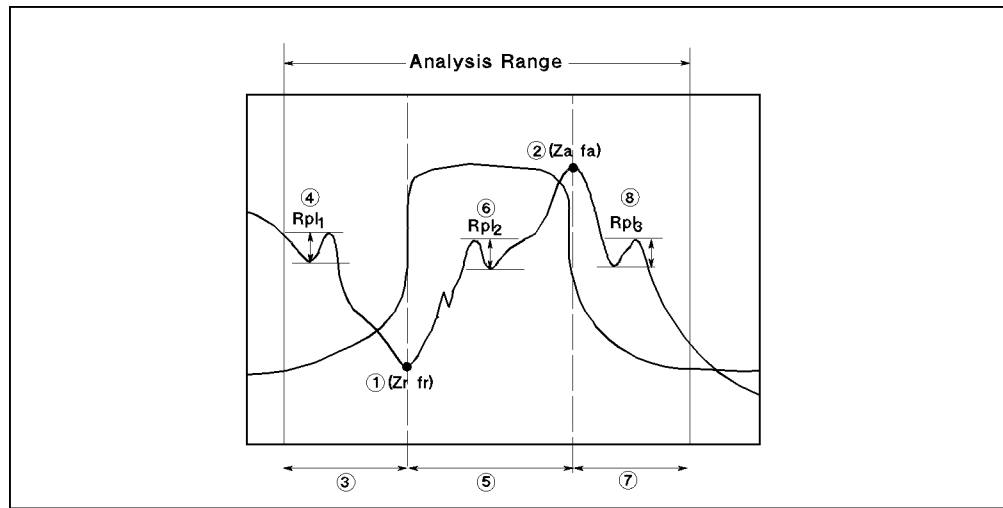


Figure I-16. OUTPCERR?

- Semantics**
- You need to select the LOG MAG format (FMT LOGM) and turn impedance conversion on (CONV ZTRA) to use this command.
 - OUTPCERR? executes the following actions:
 1. Searches for the minimum negative peak in the range and defines it as a resonant point. Then returns the resonant impedance, Z_r , and resonant frequency, f_r .
 2. Searches for the maximum peak in the range and defines it as a anti-resonant point. Then returns the anti-resonant impedance, Z_a , and anti-resonant frequency, f_p .

3. Returns the maximum height of the ripple, Rpl_1 , that is the difference between the peak and left adjacent negative peak.
4. Returns the maximum height of the ripple, Rpl_2 , that is the difference between the peak and right adjacent negative peak.
5. Returns the maximum height of the ripple, Rpl_3 , that is the difference between the peak and left adjacent negative peak.

Note

- This command can be used when the LOG MAG format (FMT LOGM) is selected. If another format is selected, OUTPCERR? returns 0 for all parameters.
- If no corresponding ripple is found, OUTPCERR? returns 0.

Examples

```

10 ASSIGN @Hp4396 TO 717
20 OUTPUT @Hp4396;"FMT LOGM; CONV ZTRA; CENT 60.02MHz; SPAN 20kHz"
30 CALL Sweep(1) ! Goes to sub routine. (See OUTPFILT?)
40 OUTPUT @Hp4396;"ANAOCH1;ANARFULL;ANAODATA"
50 OUTPUT @Hp4396;"OUTPCERR?"
60 ENTER @Hp4396;Zr,Fr,Za,Fa,R1,R2,R3
70 PRINT "Resonant: ";Zr;" [ohm] ,";Fr;" [Hz] "
80 PRINT "Anti-Resonant: ";Za;" [ohm] ,";Fa;" [Hz] "
90 PRINT "Ripple L: ";R1;" [dB] "
100 PRINT "Ripple M: ";R2;" [dB] "
110 PRINT "Ripple R: ";R3;" [dB] "
120 END

```

For Instrument BASIC

```

10 ASSIGN @Hp4396 TO 800
20 OUTPUT @Hp4396;"FMT LOGM"
30 OUTPUT @Hp4396;"CENT 60.02MHz; SPAN 20kHz"
40 EXECUTE "SING"
50 EXECUTE "ANAOCH1"
60 EXECUTE "ANARFULL"
70 EXECUTE "ANAODATA"
80 EXECUTE "OUTPCERR?"
90 PRINT "Resonant: ";READIO(8,0);" [ohm] ,";READIO(8,1);" [Hz] "
100 PRINT "Anti-Resonant: ";READIO(8,2);" [ohm] ,";READIO(8,3);" [Hz] "
110 PRINT "Ripple L: ";READIO(8,4);" [dB] "
120 PRINT "Ripple M: ";READIO(8,5);" [dB] "
130 PRINT "Ripple R: ";READIO(8,6);" [dB] "
140 END

```

Equivalent circuit analysis commands

The following commands are for the equivalent circuit analysis. They are easy to use for specific device analysis because they can output many parameters using only a single command.

- EQUCPARA?
 - EQU
- EQUCPARS?
- EQUCO?
- EQUCPARS4?

EQUCPARA?

Returns the six-device equivalent circuit parameters of the crystal resonator. (Query only)

Syntax EQUCPARA?

Query $C_0, C_1, L_1, R_1, G_0, R_0$ (Total 6)

Response Where,

| Register | Parameter | Description |
|----------|-----------|-----------------------|
| 0 | C_0 | Parallel capacitance |
| 1 | C_1 | Motional capacitance |
| 2 | L_1 | Motional inductance |
| 3 | R_1 | Motional resistance |
| 4 | G_0 | Electrode conductance |
| 5 | R_0 | Electrode resistance |

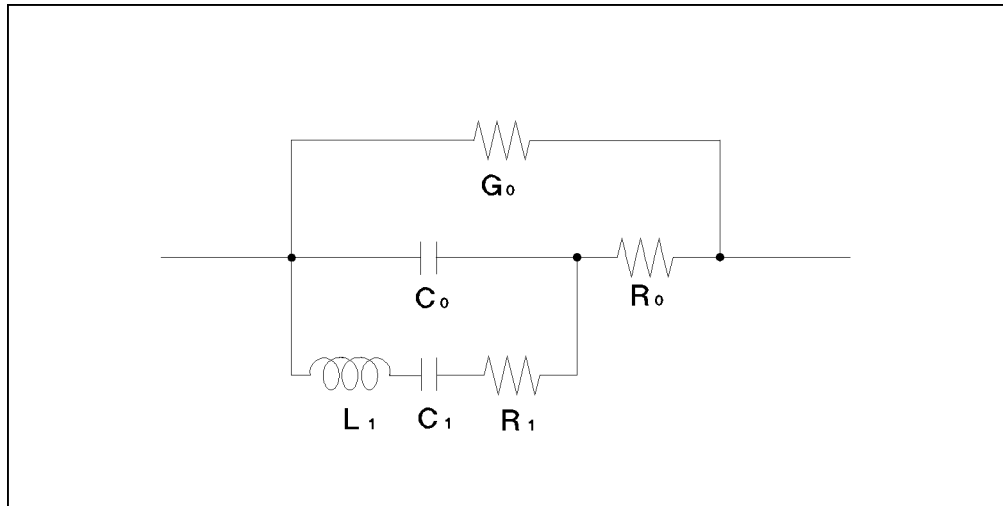


Figure I-17. Six-Device Equivalent Circuit of Crystal Resonator

Semantics ■ EQUCPARA? executes the following actions:

1. Obtains the admittance characteristic circle diagram.
2. Obtains the maximum conductance. (G_{max})
3. Obtains frequencies f_1 and f_2 ($f_1 < f_2$) of the two points where the conductance is half the maximum conductance (G_{max}).
4. Calculates f_s by $f_s = \sqrt{f_1 \times f_2}$.
5. Obtains susceptance B_{fs} at f_s .
6. Calculates ω_s by $\omega_s = 2 \times \pi \times f_s$.
7. Assumes that the frequency at which the phase becomes 0° * near the parallel resonance frequency is f_a , and obtains its conductance G_a .
8. Calculates ω_a by $\omega_a = 2 \times \pi \times f_a$.

9. Assumes that the frequency at which the phase becomes 0° * near the series resonance frequency is f_r .
10. Calculates the constants using the above values and the following equations:

$$\begin{aligned}
 Q_s &= \frac{f_s}{f_2 - f_1} & C_o' &= \frac{B_1 + B_2}{2\omega_s} \\
 L_1 &= \frac{Q_s}{\omega_s G_{max}} & R_1 &= \frac{C_o'}{C_o G_{max}} \\
 C_1 &= \frac{G_{max}}{\omega_s Q_s} & R_o &= \frac{1}{G_{max}} - R_1 \\
 C_o &= \frac{B_{fs}}{\omega_s} & G_o &= G_a - \frac{R_1 \omega_a^2 C_o^2}{1 + R_o R_1 \omega_a^2 C_o^2}
 \end{aligned}$$

* "EQUCPARA?" interpolates the 0° phase points even if it does not exist in measured data.

- If the number of points between the maximum peak point (f_{Bmax}) and the minimum peak point (f_{Bmin}) of the conductance is less than 10 points, EQUCPARA? approximates an admittance circle. The circle approximation can be performed if there are 3 points for analysis. You can specify how many points are used for circle approximation using the EQU command to reduce the analysis time.
- If EQUCPARA? fails the circle approximation, 0 will be return for all parameters.
- If there are only 2 points for analysis, EQUCPARA? returns four-device equivalent circuit parameters. In this case, EQUCPARA? returns 0 for G_o and R_o .
- If there is only 1 point for analysis, EQUCPARA? returns 0 for all parameters.

EQU
value

Specifies how many points are used for an approximation of a circle with the EQUCPARA? and EQUCPARS? commands. EQUCPARA? (or EQUCPARS?) thins the measured points out for the specified points, then makes a circle approximation. When the EQU parameter is set greater than the number of points, EQUCPARA? uses all the points for the circle approximation. Default value is 8.
value 2 to 801

Note

- You must select the following conditions or Polar format to use this command:
 - Dual Channel & Coupled Channel: ON
 - Impedance Conversion: ON
 - Analysis channel: LOG MAG format
 - Non-analysis channel: Phase format

If another format is selected, 0 will be returned for query response.

Examples For External Controller

```

10 ASSIGN @Hp4396 TO 717
20 OUTPUT @Hp4396;"FMT POLA; CONV YTRA; CENT 60.06MHz; SPAN 20kHz"
30 CALL Sweep(1) ! Goes to sub routine. (See OUTPFILT?)
40 OUTPUT @Hp4396;"ANAOCH1;ANARFULL;ANAODATA"
50 OUTPUT @Hp4396;"EQUCPARA?"
60 ENTER @Hp4396;C0,C1,L1,R1,GO,RO
70 PRINT "C0:";C0;" C1:";C1
80 PRINT "L1:";L1;" R1:";R1
90 PRINT "GO:";GO;" RO:";RO
100 END

```

For Instrument BASIC

```
10 ASSIGN @Hp4396 TO 800
20 OUTPUT @Hp4396;"FMT POLA; CONV YTRA"
30 OUTPUT @Hp4396;"CENT 60.06MHZ; SPAN 20kHz"
40 EXECUTE "SING"
50 EXECUTE "ANAOCH1"
60 EXECUTE "ANARFULL"
70 EXECUTE "ANAODATA"
80 WRITEIO 8,0;4
90 EXECUTE "EQUM"
100 EXECUTE "EQUCPARA?"
110 PRINT "CO:";READIO(8,0);" C1:";READIO(8,1)
120 PRINT "L1:";READIO(8,2);" R1:";READIO(8,3)
130 PRINT "GO:";READIO(8,4);" RO:";READIO(8,5)
140 END
```

EQUCPARS?

Outputs the six-device equivalent circuit parameters of the crystal resonator. (Query only)

Syntax EQUCPARS?

Query $C_0, C_1, L_1, R_1, f_s, f_a, f_r, f_1^*, f_2^*, G_0, R_0$ (Total 11)

Response * $f_1 < f_2$

For information about each parameter, see “EQUCPARA?”.

Note ■ You must select the following conditions or Polar format to use this command:

- Dual Channel & Coupled Channel: ON
- Impedance Conversion: ON
- Analysis channel: LOG MAG format
- Non-analysis channel: Phase format

If another format is selected, 0 will be returned for query response.

EQUC0? value

Returns the parallel capacitance (C_0) of the equivalent circuit of the resonator at the specified frequency. (Query only)

Syntax EQUC0? value

Where,

| Register | Parameter | Description |
|----------|-----------|---------------------|
| 0 | value | Frequency for C_0 |

Query C_0

Response Where,

| Register | Parameter | Description |
|----------|-----------|----------------------|
| 0 | C_0 | Parallel capacitance |

Semantics ■ C_0 is calculated using the following equation:

$$C_0 = \frac{B_s}{\omega_s}$$

Where,

B_s Imaginary part of the point on f_s .

$\omega_s = 2\pi f_s$

f_s Frequency that is specified by the command parameter.

■ If the impedance conversion is selected, C_0 is calculated using the following equation:

$$C_0 = \frac{-1}{B_s \times \omega_s}$$

Note ■ You must select the following conditions or Polar format to use this command:

- Dual Channel & Coupled Channel: ON
- Impedance Conversion: ON
- Analysis channel: LOG MAG format
- Non-analysis channel: Phase format

If another format is selected, 0 will be returned for query response.

■ If the specified frequency is out of analysis range, 0 will be returned.

■ If B_s is 0 when the impedance conversion is selected, EQUC0? returns 0.

Examples For External Controller

```
10 ASSIGN @Hp4396 TO 717
20 OUTPUT @Hp4396;"DUAC ON; COUC ON"
30 OUTPUT @Hp4396;"CHAN2; FMT PHAS"
40 OUTPUT @Hp4396;"CHAN1; FMT LOGM; CONV ZTRA"
50 OUTPUT @Hp4396;"CENT 60.06MHz; SPAN 20kHz"
60 CALL Sweep(2) ! Goes to sob routine. (See OUTPFILT?)
70 OUTPUT @Hp4396;"ANAOCH1;ANARFULL;ANAODATA"
80 OUTPUT @Hp4396;"EQUC0? 60.06MHz"
```

```
90 ENTER @Hp4396;CO
100 PRINT "CO: ";CO
110 END
```

For Instrument BASIC

```
10 ASSIGN @Hp4396 TO 800
20 OUTPUT @Hp4396;"FMT POLA"
30 OUTPUT @Hp4396;"CENT 60.06MHZ; SPAN 20kHz"
40 EXECUTE "SING"
50 EXECUTE "ANAOCH1"
60 EXECUTE "ANARFULL"
70 EXECUTE "ANAODATA"
80 WRITEIO 8,0;6.006E+7
90 EXECUTE "EQUCO?"
100 PRINT "CO: ";READIO(8,0)
110 END
```

EQUCPARS4?

Returns the 4-device equivalent circuit parameters of the crystal resonator. (Query only)

Syntax EQUCPARS4?

Query $C_0, C_1, L_1, R_1, f_s, f_a, f_r, f_1, f_2$ (Total9)

Response Where,

| Register | Parameter | Description |
|----------|-----------|---|
| 0 | C_0 | Parallel capacitance |
| 1 | C_1 | Motional capacitance |
| 2 | L_1 | Motional inductance |
| 3 | R_1 | Motional resistance |
| 4 | f_s | Motional (parallel) resonant frequency |
| 5 | f_a | Anti-resonant frequency |
| 6 | f_r | Resonant frequency |
| 7 | f_1 | Frequency at the point where the half of maximum conductance. |
| 8 | f_2 | Frequency at the point where the half of maximum conductance. ($f_1 < f_2$) |

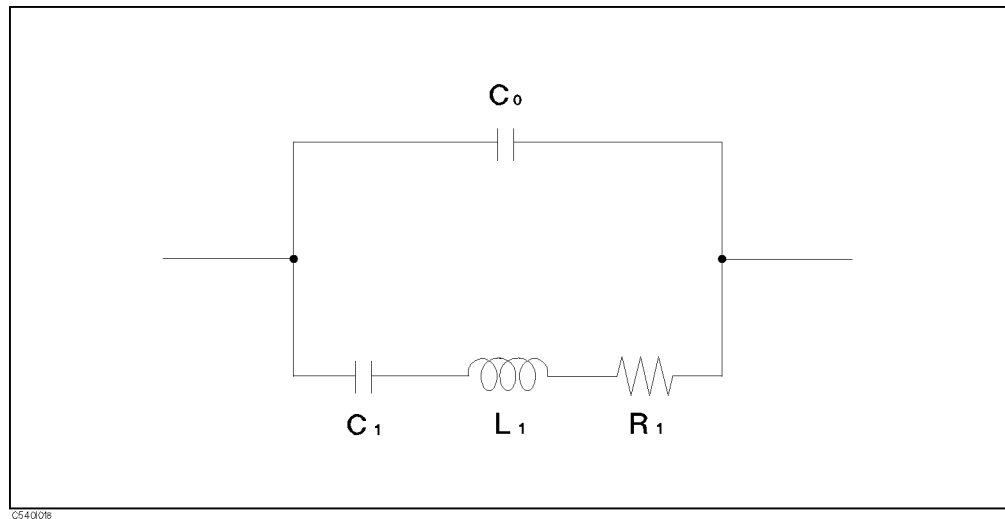


Figure I-18. Four-Device Equivalent Circuit of Crystal Resonator

- Semantics**
- You need to select the polar format (FMT POLA) and turn the admittance conversion on to use this command.
 - EQUCPARS4? executes the following actions:
 1. Obtains the admittance characteristic circle diagram. (See Figure I-19.)
 2. Obtains the susceptance (B_{fs}) and its frequency (f_s) at the maximum conductance (G_{max}) point.
 3. Obtains frequencies f_1 and f_2 ($f_1 < f_2$) of the two points where the conductance is half the maximum conductance (G_{max}).

4. Assumes that the frequency at which the phase becomes 0° near the parallel resonance frequency is f_a .
5. Assumes that the frequency at which the phase becomes 0° near the series resonance frequency is f_r .
6. Calculates the constants using the above values and the following equations:

$$C_0 = \frac{f_r^2}{f_a^2 - f_r^2} \times C_1$$

$$C_1 = \frac{1}{QR_1 2\pi f_s}$$

$$L_1 = \frac{QR_1}{2\pi f_s}$$

$$Q = \left| \frac{f_s}{f_2 - f_1} \right|$$

$$R_1 = \frac{1}{G_{max}}$$

If there are no f_r and f_a points on the admittance chart, C_0 is calculated using the following equation:

$$C_0 = \frac{B_{fs}}{2\pi f_s}$$

Where, B_{fs} is the susceptance at the G_{max} point.

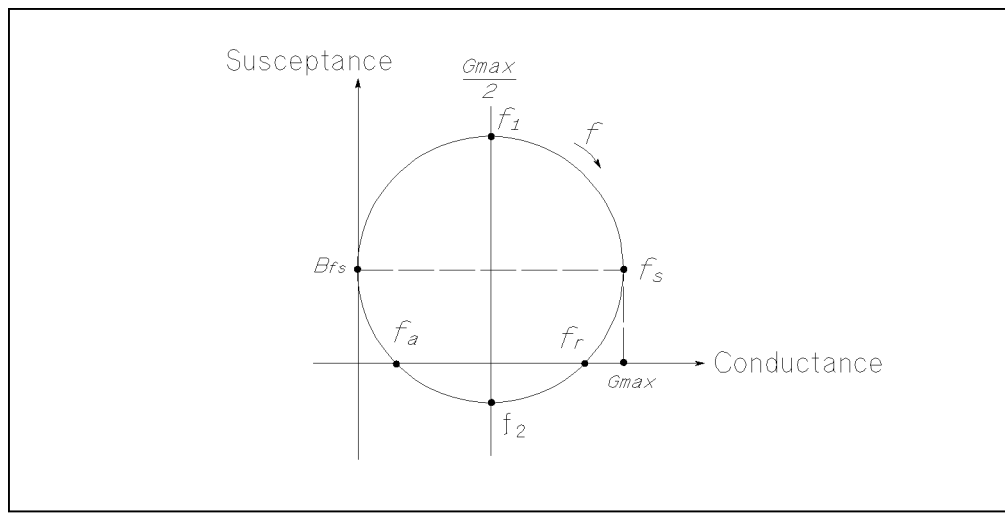


Figure I-19. Admittance Characteristic Circle Diagram

Note * This command is only available when Polar format and the admittance conversion is on. If these are not selected, 0 will be returned.

Examples For External Controller

```

10 ASSIGN @Hp4396 TO 717
20 OUTPUT @Hp4396;"FMT POLA; CONV YTRA; CENT 60.06MHz; SPAN 20kHz"
30 CALL Sweep(1) ! Goes to sub routine. (See OUTPFILT?)
40 OUTPUT @Hp4396;"ANAOCH1; ANARFULL; ANAODATA"
50 OUTPUT @Hp4396;"EQUCPARS4?"
60 ENTER @Hp4396;C0,C1,L1,R1
70 PRINT "C0:";C0;" C1:";C1
80 PRINT "L1:";L1;" R1:";R1
90 END

```

For Instrument BASIC

```
10 ASSIGN @Hp4396 TO 800
20 OUTPUT @Hp4396;"FMT POLA; CONV YTRA"
30 OUTPUT @Hp4396;"CENT 60.06MHZ; SPAN 20kHz"
40 EXECUTE "ANAOCH1"
50 EXECUTE "ANARFULL"
60 EXECUTE "ANAODATA"
70 EXECUTE "EQUCPARS4?"
80 PRINT "CO=";READIO(8,0);",C1=";READIO(8,1)
90 PRINT "L1=";READIO(8,2);",R1=";READIO(8,3)
100 END
```


Error Messages

This section lists the error messages that are displayed on the analyzer display or transmitted by the instrument over GPIB. Each error message is accompanied by an explanation, and suggestions are provided to help in solving the problem. Where applicable, references are provided to the related chapter of the appropriate manual. The messages are listed in numerical order.

In the explanation of many error commands, section numbers of the IEEE standard 488.2 are included. Refer to them for further information about an error with these IEEE section numbers.

+0 No error

The error queue is empty. Every error in the queue has been read (OUTPERRO? query) or the queue was cleared by power-on or the *CLS command.

1 CAN'T SET RBW AUTO IN ZERO SPAN

The RBW AUTO mode cannot be selected in the zero span. The RBW must be specified manually in the zero span. See Chapter 2 of the *Function Reference*. (spectrum analyzer mode only).

10 ADDITIONAL STANDARDS NEEDED

Error correction for the selected calibration class cannot be computed until all the necessary standards have been measured.

11 CALIBRATION REQUIRED

No valid calibration coefficients were found when you attempted to turn calibration ON. See *Task Reference* for information on how to perform calibration.

12 NO CALIBRATION CURRENTLY IN PROGRESS

The **RESUME CAL SEQUENCE** softkey is not valid unless a calibration is in progress. Start a new calibration. See “**Ca**” key” in the *Function Reference*.

13 CALIBRATION ABORTED

The calibration in progress was terminated due to a change of the active channel or stimulus parameters.

14 NOT VALID FOR PRESENT TEST SET

The calibration requested is inconsistent with the test set present. This message occurs in the following situations:

- A full 2-port calibration is requested with a test set other than an S-parameter test set.
- A one-path 2-port calibration is requested with an S-parameter test set (this procedure is typically used with a transmission/reflection test set).

15 **EXCEEDED 7 STANDARDS PER CLASS**

A maximum of seven standards can be defined for any class. See “Modifying Calibration Kits” in the *Function Reference*.

16 **CURRENT PARAMETER NOT IN CAL SET**

GPIB only. Correction is not valid for the selected measurement parameter.

17 **BACKUP DATA LOST**

Data checksum error on the battery backup memory has occurred. The battery is recharged for approximately 10 minutes after power was turned ON.

18 **NOT ALLOWED IN LIST SWEEP**

The level cal cannot be executed in the list sweep. The sweep type must be the linear frequency (spectrum analyzer mode only). See Chapter 5 of the *Function Reference*.

19 **UNEXPECTED DATA DETECTED: CAL ABORTED**

The signal measured for the level cal is not adequate for the calibration signal. (spectrum analyzer mode only.) See Chapter 5 of the *Function Reference*.

26 **PRINTER: not on, not connected, out of paper**

The printer does not respond to control. Verify power to the printer, and check the interface connection between the analyzer and the printer.

34 **NO VALID MEMORY TRACE**

If a memory trace is to be displayed or otherwise used, a data trace must first be stored to memory.

44 **OVERLOAD ON INPUT B**

45 **OVERLOAD ON INPUT A**

46 **OVERLOAD ON INPUT R**

47 **OVERLOAD ON INPUT S**

The power level at one of the four receiver inputs exceeds a certain level greater than the maximum input level.

48 **PHASE LOCK LOOP UNLOCKED**

Sever error. Contact your nearest Agilent Technologies office.

49 **POWER FAILED ON *nnn***

Sever error. Contact your nearest Agilent Technologies office. One or more power is failed. *nnn* is one of -5 V, -15 V, +5 V, +15 V, +65 V, and PostRegHot. It shows that which power line is failed. When this error occurs, the system halts so a controller cannot read this error by GPIB.

50 **CONT SWITCHING MAY DAMAGE MECH SWITCH**

RF output power switch or input attenuator switch at input S is switching sweep by sweep, because RF power level or the input attenuator setting is different between two channels and the dual channel is turn on. To avoid premature wearing out of the output power switch and input attenuator switch, change trigger type to HOLD, SINGLE, or NUMBER of GROUP to hold sweep after measurement required. Or turn off the dual channel, or set the power level and the input attenuator of both channels to the same setting.

51 **MEASUREMENT INVALID AT $f \leq 1\text{MHz}$, $\text{IFBW} \geq 10\text{kHz}$**

This message will displayed when whole frequency measured is less than or equal to 1 MHz and IFBW is set to 10 kHz or 40 kHz because the network measurement performance is not warranted at frequency ≤ 1 MHz with 10 kHz or 40 kHz IFBW.

54 **TOO MUCH DATA**

Either there is too much binary data to send to the analyzer when the data transfer format is FORM 2, FORM 3 or FORM 5, or the amount of data is greater than the number of points.

55 **NOT ENOUGH DATA**

The amount of data sent to the analyzer is less than that expected (*GPIB only*).

56 **OPTION NOT INSTALLED**

This error occurs when an GPIB command which is optional command is sent and the analyzer is not installed the option (*GPIB only*). Please confirm options installed to the analyzer using *OPT? command (see “*OPT?” in Chapter 2.)

64 **TOO MANY SEGMENTS**

The maximum number of segments for the limit line table is 18. See Chapter 8 of the *Task Reference*.

74 **CURRENT EDITING SEGMENT SCRATCHED**

The current editing segment for the list table and the limit line is scratched when the following cases occur (*GPIB only*) :

- When EDITLIST (edit list table) command is received while editing a segment for the list table.
- When EDITLIML (edit limit line) command is received while editing a segment for the limit line.

Send LIMSDON (limit segment done) or SDON (segment done) to terminate editing segment.

75 **COMMAND IGNORED - SEGMENT NOT DONE YET**

The GPIB command the analyzer received is ignored, because the segment is editing (*GPIB only*). Send LIMSDON (limit segment done) or SDON (segment done) to terminate editing segment. (See “LIMSDON” in Chapter 2 and “SDON” in Chapter 2.)

76 **SEGMENT START/STOP OVERLAPPED**

Segments are not allowed to be overlapped. Reenter appropriate value for start or stop value of segments to avoid that segment is not overlapped.

77 **TOO MANY SEGMENTS OR POINTS**

Frequency list mode is limited to 31 segments or 801 points.

78 **TOO SMALL POINTS OR TOO LARGE STOP**

STOP+SPAN/(NOP-1) is out of sweep range. Increase NOP or change STOP value to lower frequency to avoid this error.

82 **CAN'T CHANGE- ANOTHER CONTROLLER ON BUS**

The analyzer cannot assume the mode of system controller until the active controller is removed from the bus or relinquishes the bus. See Chapter 7 of the *GPIB Programming Guide*.

83 **FORMAT NOT VALID FOR MEASUREMENT**

The conversion function except the 1/S and the multiple phase modes is not valid for the Smith, admittance, and SWR formats.

84 **ANALYZER TYPE MISMATCH**

The analyzer receives a command that is not available for the current analyzer type. Please confirm GPIB command or change analyzer type before sending the command.

93 **NO DATA TRACE**

The **MARKER ON [DATA]** is selected when the data trace is not displayed.

94 **NO MEMORY TRACE**

The **MARKER ON [MEMORY]** is selected when the memory trace is not displayed.

95 **NO MARKER DELTA - SPAN NOT SET**

The **MKRD→SPAN** softkey requires that delta marker mode be turned ON.

96 **NO MARKER DELTA - RANGE NOT SET**

The **MKRD→SEARCH RNG** softkey requires that delta marker is turned ON.

98 **NO ACTIVE MARKER**

The marker→ command cannot be execute when no marker is displayed on the screen. Turn on the marker before executing the marker→ commands.

99 **CAN'T CHANGE WHILE DUAL CHAN OFF**

The Cross channel cannot be turned on when dual channel is off. Turn on the dual channel before the cross channel is turned on.

110 **SAVE ERROR**

A serious error, for example physically damaged disk surface, is detected on saving a file.

111 **RECALL ERROR: INSTR STATE PRESET**

A serious error, for example corrupted data, is detected on recalling a file, and this forced the analyzer to be PRESET.

112 **INVALID FILE NAME**

GPIB only. The file name for the RECALL, PURGE, or RE-SAVE function must have a “_D” or “_S” extension for LIF format.

113 **NO STATE/DATA FILES ON DISK**

There are no files on the flexible disk with extensions, “_D” or “_S” for LIF format, or “STA” or “.DTA” for DOS format.

114 **CAN'T SAVE GRAPHICS WHEN COPY IN PROGRESS**

If you attempt to save graphics when a print is in progress, this error message is displayed.

115 **LIF-DOS COPY NOT ALLOWED**

If you try to copy a file between the RAM disk and the flexible disk when the format of the RAM disk is different from the format of the flexible disk, this message is displayed.

116 **NO STATE/DATA FILES ON MEMORY**

There are no files on the RAM disk memory with extensions, “_D” or “_S” for LIF format, or “STA” or “.DTA” for DOS format.

119 **NO DATA TRACE DISPLAYED**

The **SCALE FOR [DATA]** is selected when the data trace is not displayed.

120 **NO MEMORY TRACE DISPLAYED**

The **SCALE FOR [MEMORY]** is selected when the memory trace is not displayed.

124 **LIST TABLE EMPTY OR INSUFFICIENT TABLE**

The frequency list is empty. To implement the list frequency mode, add segments to the list table.

125 **CAN'T SET SLOPE ON IN POWER SWEEP**

The slope function can be turned on in frequency sweep.

126 **CAN'T CHANGE NUMBER OF POINTS**

The number of points of the spectrum analyzer mode cannot be to change manually, except in zero span.

127 **CAN'T SET SWEEP TIME AUTO IN ZERO SPAN**

The automatic sweep time cannot be in zero span of the spectrum analyzer mode. (The network analyzer mode allows that the automatic sweep time is turned on.)

128 **Repet Smpling : SPAN = 0 ONLY**

The repetitive sampling is turn on when span must be zero, the sweep type must be linear frequency, and the trigger source must be EXT or VIDEO only. Confirm the analyzer setting and set appropriate setting for the repetitive sampling mode.

129 **Repet Smpling : LIN FREQ ONLY**

The repetitive sampling is turn on when span must be zero, the sweep type must be linear frequency, and the trigger source must be EXT or VIDEO only. Confirm the analyzer setting and set appropriate setting for the repetitive sampling mode.

130 **Repet Smpling : TRIG = EXT OR VIDEO ONLY**

The repetitive sampling is turn on when span must be zero, the sweep type must be linear frequency, and the trigger source must be EXT or VIDEO only. Confirm the analyzer setting and set appropriate setting for the repetitive sampling mode.

131 **FREQUENCY SWEEP ONLY**

The sweep type must be frequency sweep when the center step size is set.

133 **CAN'T CHANGE ON LIST SWEEP**

When list sweep is selected, the following parameters are not allowed to be changed:

- CENTER, SPAN, START, STOP
- NOP
- IFBW or RBW
- POWER

Modify the list table to change these parameters in the list sweep.

134 **CAN'T COUPLE IN CURRENT INPUTS**

When one channel measures a ratio measurement, and the other one measures an absolute measurement (for example: A/R and B), COUPLED CH can not be turned on.

135 **COUPLED CHAN - BETWEEN NA & NA OR ZA & ZA**

The analyzer types of both channels must be the network analyzer mode when the coupled channel is turned on.

141 **INSUFFICIENT MEMORY**

If a lot of tasks is executed at same time, memory might be insufficient for a while. (For example, running HP instrument BASIC program, printing a screen, and sending or receiving data array by GPIB are required at same time.) Please wait until finishing some tasks then execute the next task.

146 **ON POINT NOT ALLOWD FOR THE CURRENT TRIG**

The trigger event mode cannot be changed to the ON POINT mode because the current trigger source setting does not allow the ON POINT mode. The trigger event ON POINT mode is available for only MANUAL, EXTERNAL, and BUS trigger sources of the network analyzer mode.

154 **INVALID DATE**

The date entered to set the real time clock is invalid. Reenter correct date.

193 **POWER ON TEST FAILED**

An internal test fails in the power on sequence (the power on self-test fails). Contact your nearest Agilent Technologies office or see the *Service Manual* for troubleshooting.

194 **EEPROM WRITE ERROR**

Data cannot be stored properly into the EEPROM on the A1 CPU, when performing the display background adjustment or updating correction constants in the EEPROM using the adjustment program. See the *Service Manual* for troubleshooting.

195 **ALL INT TEST FAILED**

An “internal test 0: ALL INT” fails. See the *Service Manual* for troubleshooting.

196 **FLASH MEMORY CHECK SUM ERROR**

The data (firmware) stored in the A1 flash memory are invalid. This message is displayed in the bootloader menu. See the *Service Manual* for troubleshooting.

197 **BACKUP SRAM CHECK SUM ERROR**

An “internal test 1: A1 CPU” fails. The data (GPIB Address and so on) stored in the A1 CPUs BACKUP SRAM are invalid. Replace the A1 CPU with a new one. See the *Service Manual* for troubleshooting.

198 **EEPROM CHECK SUM ERROR**

An “internal test 1: A1 CPU” fails. The data (Correction Constants and so on) stored in the A1 CPU’s EEPROM are invalid. See the *Service Manual* for troubleshooting.

199 **DSP CHIP TEST FAILED**

An “internal test 1: A1 CPU” fails. The A1 CPU’s DSP (Digital Signal Processor) does not work properly. Replace the A1 CPU with a new one. See the *Service Manual* for troubleshooting.

200 **F-BUS TIMER CHIP TEST FAILED**

An “internal test 1: A1 CPU” fails. The A1 CPU’s F-BUS (Frequency Bus) timer does not work properly. Replace the A1 CPU with a new one. See the *Service Manual* for troubleshooting.

201 **RTC CHIP TEST FAILED**

An “internal test 1: A1 CPU” fails. The A1 CPU’s RTC (Real Time Clock) does not work properly. Replace the A1 CPU with a new one. See the *Service Manual* for troubleshooting.

202 **KEY CHIP TEST FAILED**

An “internal test 1: A1 CPU” fails. The A1 CPU’s front keyboard control chip does not work properly. Replace the A1 CPU with a new one. See the *Service Manual* for troubleshooting.

203 **FDC CHIP TEST FAILED**

An “internal test 1: A1 CPU” fails. The A1 CPU’s FDC (Flexible Disk drive control) ship does not work properly. Replace the A1 CPU with a new one. See the *Service Manual* for troubleshooting.

204 **HP-IB CHIP TEST FAILED**

An “internal test 1: A1 CPU” fails. The A1 CPU’s GPIB chip does not work properly. Replace the A1 CPU with a new one. See the *Service Manual* for troubleshooting.

205 **HP-HIL CHIP TEST FAILED**

An “internal test 1: A1 CPU” fails. The A1 CPU’s HP-HIL control chip does not work properly. Replace the A1 CPU with a new one. See the *Service Manual* for troubleshooting.

206 **CPU INTERNAL SRAM R/W ERROR**

An “internal test 2: A1 VOLATILE MEMORY” fails. The A1 CPU’s internal SRAM does not work properly. Replace the A1 CPU with a new one. See the *Service Manual* for troubleshooting.

207 **CPU BACKUP SRAM R/W ERROR**

An “internal test 2: A1 VOLATILE MEMORY” fails. The A1 CPU’s BACKUP SRAM does not work properly. Replace the A1 CPU with a new one. See the *Service Manual* for troubleshooting.

208 **DSP SRAM R/W ERROR**

An “internal test 2: A1 VOLATILE MEMORY” fails. The DSP’s SRAM on the A1 CPU does not work properly. Replace the A1 CPU with a new one. See the *Service Manual* for troubleshooting.

209 **DUAL PORT SRAM R/W ERROR**

An “internal test 2: A1 VOLATILE MEMORY” fails. The DSP’s dual port SRAM on the A1 CPU does not work properly. Replace the A1 CPU with a new one. See the *Service Manual* for troubleshooting.

210 **POST REGULATOR OUTPUT VOLTAGE OUT OF SPEC**

An “internal test 4: A2 POST REGULATOR” fails. A power supply voltage of the A2 post-regulator is out of its limits. See the *Service Manual* for troubleshooting.

211 **GND LEVEL OUT OF SPEC**

An “internal test 4: A2 POST REGULATOR” fails. The voltage of the GND (Ground) at the DC bus node 26 is out of its limits. See the *Service Manual* for troubleshooting.

212 **FAN POWER OUT OF SPEC**

An “internal test 4: A2 POST REGULATOR” fails. The voltage of the fan power supply at the DC bus node 11 is out of its limits. See the *Service Manual* for troubleshooting.

213 **FAILURE FOUND FROM A/D MUX TO A/D CONVERTER**

An “internal test 5: A6 A/D CONVERTER” fails. A trouble is found on the signal path from the A/D multiplexer to A/D converter on the A6 receiver IF. See the *Service Manual* for troubleshooting.

214 **REF OSC TEST FAILED**

An “internal test 6: A5 REFERENCE OSC” fails. The reference oscillator on the A5 synthesizer does not work properly. See the *Service Manual* for troubleshooting.

215 **FRACTIONAL N OSC TEST FAILED**

An “internal test 7: A5 FRACTIONAL N OSC” fails. The fractional N oscillator on the A5 synthesizer does not work properly. See the *Service Manual* for troubleshooting.

216 **STEP OSC TEST FAILED**

An “internal test 8: A5 STEP OSC” fails. The step oscillator on the A5 synthesizer does not work properly. See the *Service Manual* for troubleshooting.

217 **1st LO OSC TEST FAILED**

An “internal test 9: A4A1 1ST LO OSC” fails. The 1st LO OSC (first local oscillator) on the A4A1 1st LO does not work properly. See the *Service Manual* for troubleshooting.

218 **2nd LO OSC TEST FAILED**

An “internal test 10: A3A2 2ND LO” fails. The 2nd LO OSC (second local oscillator) on the A3A2 2nd LO does not work properly. See the *Service Manual* for troubleshooting.

219 **3rd LO OSC TEST FAILED**

An “internal test 12: A6 3RD LO OSC” fails. The 3rd LO OSC (third local oscillator) on the A6 receiver IF does not work properly. See the *Service Manual* for troubleshooting.

220 **SOURCE OSC TEST FAILURE**

An “internal test 13: A3A1 SOURCE OSC” fails. The source oscillator on the A3A1 ALC does not work properly. See the *Service Manual* for troubleshooting.

221 **DC OFFSET TOO BIG ON 0 DEG PATH**

An “internal test 14: A6 3rd IF DC OFFSET” fails. The DC offset on 0° path of the A6 receiver IF is larger than its limit. See the *Service Manual* for troubleshooting.

222 **DC OFFSET TOO BIG ON 90 DEG PATH**

An “internal test 14: A6 3rd IF DC OFFSET” fails. The DC offset on 90° path of the A6 receiver IF is larger than its limit. See the *Service Manual* for troubleshooting.

223 **SAMPLE FREQUENCY OUT OF SPEC**

An “internal test 15: A6 SEQUENCER” fails. The sampling frequency of the sample/hold circuit on the A6 receiver IF is out of its limits. See the *Service Manual* for troubleshooting.

224 **ALC TEST FAILED**

An “internal test 16: A3A1 ALC” fails. The ALC (Auto Level Control) circuit on the A3A1 ALC does not work properly. See the *Service Manual* for troubleshooting.

225 **A3 DIVIDER OUTPUT FREQUENCY OUT OF SPEC**

An “internal test 11: A3A1 DIVIDER” fails. The output frequency of the divider circuit on the A3A1 ALC is out of its limits. See the *Service Manual* for troubleshooting.

226 **FLOPPY DISK DRIVE FAILURE FOUND**

An “external test 18: DSK DR FAULT ISOL’N” fails. The A53 built-in FDD (flexible disk drive) does not work properly. Replace the A53 FDD with a new one. See the *Service Manual* for troubleshooting.

227 **POWER SWEEP LINEARITY OUT OF SPEC**

An “external test 19: POWER SWEEP LINEARITY” fails. See the *Service Manual* for troubleshooting.

228 **OUTPUT ATTENUATOR ACCURACY OUT OF SPEC**

An “external test 20: OUTPUT ATTENUATOR” fails. See the *Service Manual* for troubleshooting.

229 **INPUT ATTENUATOR ACCURACY OUT OF SPEC**

An “external test 21: INPUT ATTENUATOR” fails. See the *Service Manual* for troubleshooting.

230 **RF OUT TO S-INPUT FLATNESS TEST FAILED**

An “external test 22: RF TO S LVL & FLTNESS” fails. See the *Service Manual* for troubleshooting.

231 **S-INPUT TO A-INPUT CROSSTALK TEST FAILED**

An “external test 23: S TO A CROSSTALK” fails. See the *Service Manual* for troubleshooting.

232 **S-INPUT LEVEL COMPRESSION TEST FAILED**

An “external test 24: S INPUT COMPRESSION” fails. See the *Service Manual* for troubleshooting.

233 **S-INPUT RESIDUAL RESPONSE OUT OF SPEC**

An “external test 25: S INPUT RESIDUALS” fails. See the *Service Manual* for troubleshooting.

234 **1st LO LEAKAGE TEST FAILED**

An “external test 25: S INPUT RESIDUALS” fails. See the *Service Manual* for troubleshooting.

235 **S-INPUT NOISE LEVEL OUT OF SPEC**

An “external test 26: S INPUT NOISE LEVEL” fails. See the *Service Manual* for troubleshooting.

236 **FRACTION SPURIOUS OUT OF SPEC**

An “external test 27: FRACTION SPURIOUS” fails. See the *Service Manual* for troubleshooting.

237 **RF OUT TO R-INPUT FLATNESS TEST FAILED**

An “external test 28: RF TO A LVL & FLTNESS” fails. See the *Service Manual* for troubleshooting.

238 **R-INPUT TO A-INPUT CROSSTALK OUT OF SPEC**

An “external test 29: NA CROSSTALK & NOISE” fails. See the *Service Manual* for troubleshooting.

239 **R-INPUT TO B-INPUT CROSSTALK OUT OF SPEC**

An “external test 29: NA CROSSTALK & NOISE” fails. See the *Service Manual* for troubleshooting.

240 **R-INPUT NOISE LEVEL OUT OF SPEC**

An “external test 29: NA CROSSTALK & NOISE” fails. See the *Service Manual* for troubleshooting.

241 **A-INPUT NOISE LEVEL OUT OF SPEC**

An “external test 29: NA CROSSTALK & NOISE” fails. See the *Service Manual* for troubleshooting.

242 **B-INPUT NOISE LEVEL OUT OF SPEC**

An “external test 29: NA CROSSTALK & NOISE” fails. See the *Service Manual* for troubleshooting.

243 **R-INPUT LEVEL COMPRESSION TEST FAILED**

An “external test 30: R INPUT COMPRESSION” fails. See the *Service Manual* for troubleshooting.

244 **RANGING ACCURACY TEST FAILED**

An “external test 31: RANGING” fails. See the *Service Manual* for troubleshooting.

245 **A/R RATIO ACCURACY OUT OF SPEC**

An “external test 32: A/R RATIO ACCURACY” fails. See the *Service Manual* for troubleshooting.

246 **A/R RATIO RAW RESPONSE TEST FAILED**

An “external test 32: A/R RATIO ACCURACY” fails. See the *Service Manual* for troubleshooting.

247 **A-INPUT LEVEL COMPRESSION TEST FAILED**

An “external test 33: A INPUT COMPRESSION” fails. See the *Service Manual* for troubleshooting.

248 **B/R RATIO ACCURACY OUT OF SPEC**

An “external test 34: B/R RATIO ACCURACY” fails. See the *Service Manual* for troubleshooting.

249 **B/R RAW RESPONSE TEST FAILED**

An “external test 34: B/R RATIO ACCURACY” fails. See the *Service Manual* for troubleshooting.

250 **B-INPUT LEVEL COMPRESSION TEST FAILED**

An “external test 35: B INPUT COMPRESSION” fails. See the *Service Manual* for troubleshooting.

251 **SA RES FILTER 3 DB BW OUT OF SPEC**

An “external test 36: RESOLUTION BANDWIDTH” fails. See the *Service Manual* for troubleshooting.

252 **SA RES FILTER SHAPE FACTOR OUT OF SPEC**

An “external test 36: RESOLUTION BANDWIDTH” fails. See the *Service Manual* for troubleshooting.

253 **SA RES FILTER TRACK NOISE TEST FAILED**

An “external test 36: RESOLUTION BANDWIDTH” fails. See the *Service Manual* for troubleshooting.

254 **SA RES FILTER SWITCHING UNC. OUT OF SPEC**

An “external test 36: RESOLUTION BANDWIDTH” fails. See the *Service Manual* for troubleshooting.

255 **IF GAIN SWITCHING UNC. OUT OF SPEC**

An “external test 37: IF GAIN” fails. See the *Service Manual* for troubleshooting.

256 **SIDE BAND LEVEL OUT OF SPEC**

An “external test 38: PHASE NOISE” fails. See the *Service Manual* for troubleshooting.

257 **SA NON-HARMONIC SPURIOUS OUT OF SPEC**

An “external test 39: SPURIOUS” fails. See the *Service Manual* for troubleshooting.

258 **X-TAL FILTER RESPONSE OUT OF SPEC**

An “external test 40: X'TAL FILTER RESPONSE” fails. See the *Service Manual* for troubleshooting.

259 **X-TAL FILTER RAW RESPONSE TEST FAILED**

An “external test 40: X'TAL FILTER RESPONSE” fails. See the *Service Manual* for troubleshooting.

260 **ALL EXT TEST FAILED**

“External tests 63 to 67” fails. See the *Service Manual* for troubleshooting.

261 **IMPEDANCE ADAPTER TEST FAILED**

An error occurred while performing the impedance adapter test. Contact your nearest Agilent Technologies office or follow the instructions in the *Service Manual* to repair the product.

267 **COMPENSATION REQUIRED**

OPEN on OFF, **SHORT on OFF** or **LOAD on OFF** key is pressed when the fixture compensation is not being performed. The fixture compensation is required before you attempt to press these keys.

268 **NO COMPENSATION CURRENTLY IN PROGRESS**

RESUME COMP SEQ is pressed when the fixture compensation is not suspended. Press **COMPEN MENU** to start the fixture compensation.

270 **COMPENSATION STD LIST UNDEFINED**

The value of the calibration kit used for the fixture compensation is not defined. Define the value using **MODIFY []**.

-100 **Command error**

This is a generic syntax error that the analyzer cannot detect more specific errors. This code indicates only that a command error, as defined in IEEE 488.2, 11.5.1.1.4, has occurred.

-101 **Invalid character**

A syntax element contains a character that is invalid for that type. For example, a header containing an ampersand (**SENSE&**).

-102 **Syntax error**

An unrecognized command or data type was encountered. For example, a string was received when the analyzer was not expecting to receive a string.

-103 **Invalid separator**

The parser was expecting a separator and encountered an illegal character. For example, the semicolon was omitted after a program message unit, ***RST:TRIG**.

-104 **Data type error**

The parser recognized an unallowed data element. For example, numeric or string data was expected but block data was encountered.

-105 **GET not allowed**

A Group Execute Trigger (GET) was received within a program message (see IEEE 488.2, 7.7).

-108 **Parameter not allowed**

More parameters were received than expected for the header. For example, the *SRE command only accepts one parameter, so receiving *SRE 4,16 is not allowed.

-109 **Missing parameter**

Fewer parameters were received than required for the header. For example, the *SRE command requires one parameter, so receiving only *SRE is not allowed.

-110 **Command header error**

An error was detected in the header. This error message is used when the analyzer cannot detect the more specific errors described for errors –111 through –119.

-111 **Header separator error**

A character that is not a legal header separator was encountered while parsing the header. For example, no white space followed the header, thus *SRE4 is an error.

-112 **Program mnemonic too long**

The header contains more than twelve characters (see IEEE 488.2, 7.6.1.4.1).

-113 **Undefined header**

The header is syntactically correct, but it is undefined for the analyzer. For example, *XYZ is not defined for the analyzer.

-114 **Header Suffix out of range**

The value of a numeric suffix attached to a program mnemonic makes the header invalid.

-120 **Numeric data error**

This error, as well as errors –121 through –129, are generated when parsing a data element that appears to be numeric, including the nondecimal numeric types. This particular error message is used if the analyzer cannot detect a more specific error.

-121 **Invalid character in number**

An invalid character for the data type being parsed was encountered. For example, an alpha character in a decimal numeric or a “9” in octal data.

-123 **Exponent too large**

The magnitude of the exponent was larger than 32000 (see IEEE 488.2, 7.7.2.4.1).

-124 **Too many digits**

The mantissa of a decimal numeric data element contains more than 255 digits excluding leading zeros (see IEEE 488.2, 7.7.2.4.1).

-128 **Numeric data not allowed**

A legal numeric data element was received, but the analyzer does not accept it in this position for a header.

-130 **Suffix error**

This error, as well as errors –131 through –139, are generated when parsing a suffix. This particular error message is used if the analyzer cannot detect a more specific error.

-131 **Invalid suffix**

The suffix does not follow the syntax described in IEEE 488.2, 7.7.3.2, or the suffix is inappropriate for the analyzer.

-134 **Suffix too long**

The suffix contained more than 12 characters (see IEEE 488.2, 7.7.3.4).

-138 **Suffix not allowed**

A suffix was encountered after a numeric element that does not allow suffixes.

-140 **Character data error**

This error, as well as errors –141 through –148, are generated when analyzing the syntax of a character data element. This particular error message is used if the analyzer cannot detect a more specific error.

-141 **Invalid character data**

Either the character data element contains an invalid character or the particular element received is not valid for the header.

-144 **Character data too long**

The character data element contains more than twelve characters (see IEEE 488.2, 7.7.1.4).

-148 **Character data not allowed**

A legal character data element was encountered where prohibited by the analyzer.

-150 **String data error**

This error, as well as errors –151 and –158, are generated when analyzing the syntax of a string data element. This particular error message is used if the analyzer cannot detect a more specific error.

-151 **Invalid string data**

A string data element was expected, but was invalid for some reason (see IEEE 488.2, 7.7.5.2). For example, an EMD message was received before the terminal quote character.

-158 **String data not allowed**

A string data element was encountered but was not allowed by the analyzer at this point in parsing.

-160 **Block data error**

This error, as well as errors –161 and –168, are generated when analyzing the syntax of a block data element. This particular error message is used if the analyzer cannot detect a more specific error.

-161 **Invalid block data**

A block data element was expected, but was invalid for some reason (see IEEE 488.2, 7.7.6.2). For example, an END message was received before the length was satisfied.

-168 **Block data not allowed**

A legal block data element was encountered but was not allowed by the analyzer at this point in parsing.

-200 **Execution error**

This is the generic syntax error that the analyzer cannot detect more specific errors. This code indicates only that an execution error as defined in IEEE 488.2, 11.5.1.1.5 has occurred.

-210 **Trigger error**

A trigger related error occurred. This error message is used when the analyzer cannot detect the more specific errors described for errors –211 through –219.

-211 **Trigger ignored**

A GET, *TRG, or triggering signal was received and recognized by the analyzer but was ignored because of analyzer timing considerations. For example, the analyzer was not ready to respond.

-213 **Init ignored**

A request for a measurement initiation was ignored as another measurement was already in progress.

-220 **Parameter error**

Indicates that a program data element related error occurred. This error message is used when the analyzer cannot detect the more specific errors described for errors –221 through –229.

-221 **Settings conflict**

A legal program data element was parsed but could not be executed due to the current device state (see IEEE 488.2, 6.4.5.3 and 11.5.1.1.5).

-222 **Data out of range**

A legal program data element was parsed but could not be executed because the interpreted value was outside the legal range as defined by the analyzer (see IEEE 488.2, 11.5.1.1.5).

-223 **Too much data**

A legal program data element of block, expression, or string type was received that contained more data than the analyzer could handle due to memory or related device-specific requirements.

-224 **Illegal parameter value**

Used where exact value, from a list of possibilities, was expected.

-225 **Data out of memory**

The analyzer has insufficient memory to perform the requested operation.

-230 **Data corrupt or stale**

Possibly invalid data. New reading started but not completed since last access.

-231 **Data questionable**

Measurement accuracy is suspect.

-240 **Hardware error**

A legal program command or query could not be executed because of a hardware problem in the analyzer. Definition of what constitutes a hardware problem is completely device-specific. This error message is used when the analyzer cannot detect the more specific errors described for errors –241 through –249.

-241 **Hardware missing**

A legal program command or query could not be executed because of missing analyzer hardware. For example, an option was not installed.

-250 **Mass storage error**

A mass storage error occurred. This error message is used when the analyzer cannot detect the more specific errors described for errors –251 through –259.

-256 **File name not found**

A legal program command could not be executed because the file name on the device media was not found: for example, an attempt was made to read or copy a nonexistent file.

-257 **File name error**

A legal program command or query could not be executed because the file name on the device media was in error. For example, an attempt was made to copy to a duplicate file name. The definition of what constitutes a file name error is device-specific.

-280 **Program error**

A downloaded program-related execution error occurred. This error message is used when the analyzer cannot detect the more specific errors described for errors –281 through –289.

-281 **Cannot create program**

An attempt to create a program was unsuccessful. A reason for the failure might include not enough memory.

-282 **Illegal program name**

The name used to reference a program was invalid. For example, redefining an existing program, deleting a nonexistent program, or in general, referencing a nonexistent program.

-283 **Illegal variable name**

An attempt was made to reference a nonexistent variable in a program.

-284 **Program currently running**

Certain operations dealing with programs may be illegal while the program is running. For example, deleting a running program might not be possible.

-285 **Program syntax error**

A syntax error appears in a downloaded program. The syntax used when parsing the downloaded program is device-specific.

-286 **Program runtime error**

A program runtime error of the HP instrument BASIC has occurred. To get a more specific error information, use the `ERRM$` or `ERRN` command of the HP instrument BASIC.

-310 **System error**

Some error, termed "system error" by the analyzer, has occurred.

-311 **Memory error**

An error was detected in the analyzer's memory.

-330 **Self-test failed**

A self-test failed. Contact your nearest Agilent Technologies office or see the *Service Manual* for troubleshooting.

-350 **Queue overflow**

A specific code entered into the queue in lieu of the code that caused the error. This code indicates that there is no room in the queue and an error occurred but was not recorded.

-400 **Query errors**

This is the generic query error that the analyzer cannot detect more specific errors. This code indicates only that a query error as defined in IEEE 488.2, 11.5.1.1.7 and 6.3 has occurred.

-410 **Query INTERRUPTED**

A condition causing an interrupted query error occurred (see IEEE 488.2, 6.3.2.3). For example, a query followed by DAB or GET before a response was completely sent.

-420 **Query UNTERMINATED**

A condition causing an unterminated query error occurred (see IEEE 488.2, 6.3.2.2). For example, the analyzer was addressed to talk and an incomplete program message was received by the controller.

-430 **Query DEADLOCKED**

A condition causing a deadlocked query error occurred (see IEEE 488.2, 6.3.1.7). For example, both input buffer and output buffer are full and the analyzer cannot continue.

Index

B

<*block*>, 1-7

C

calibration array, F-1
calibration coefficient, H-2
calibration type, F-1
command abbreviation, 1-5
common command, 1-3
convention, 2-2

D

data format, H-1
data level, H-2
data transfer, H-1

E

error corrected data, H-2
error message, Messages-1
Event Status Register B, D-1, D-3

F

formatted data, H-2

I

Instrument Event Status Register, D-1, D-3

K

key code, G-1

M

manual changes, A-1

N

<*numeric*>, 1-6

O

Operation Status Negative Transition Filter,
D-5
Operation Status Positive Transition Filter,
D-5
Operation Status Register, D-1, D-4
OSNT, D-5
OSPT, D-5

Q

Questionable Status Register Bit, D-1

R

raw data, H-2

S

SCPI command, 1-3
serial number, A-2
simple command, 1-3
standard class, F-1
Standard Event Status Register, D-1, D-3
Status Byte, D-1, D-2
status reporting, D-1
<*string*>, 1-6
suffix, 1-6

T

trigger system, E-1

