

**Standard
Commands for
Programmable
Instruments
(SCPI)**

Volume 2: Command Reference

VERSION 1999.0
May, 1999
Printed in U.S.A.

Table of Contents

Chapter 1	Introduction	1-1
Chapter 2	Instrument Model	2-1
Chapter 3	Measurement Instructions	3-1
Chapter 4	CALCulate Subsystem	4-1
Chapter 5	CALibration Subsystem	5-1
Chapter 6	CONTRol Subsystem	6-1
Chapter 7	DIAGnostic Subsystem	7-1
Chapter 8	DISPlay Subsystem	8-1
Chapter 9	FORMat Subsystem	9-1
Chapter 10	HCOPY	10-1
Chapter 11	INPut Subsystem	11-1
Chapter 12	INSTrument Subsystem	12-1
Chapter 13	MEMory Subsystem	13-1
Chapter 14	MMEMorySubsystem	14-1
Chapter 15	OUTPut Subsystem	15-1
Chapter 16	PROGram Subsystem	16-1
Chapter 17	ROUTe Subsystem	17-1
Chapter 18	SENSe Subsystem	18-1
Chapter 19	SOURce Subsystem	19-1
Chapter 20	STATus Subsystem	20-1
Chapter 21	SYSTem Subsystem	21-1
Chapter 22	TEST Subsystem	22-1
Chapter 23	TRACe DATA	23-1
Chapter 24	TRIGger Subsystem	24-1
Chapter 25	UNIT Subsystem	25-1
Chapter 26	VXI Subsystem	26-1

Table of Contents

Chapter 1 Introduction

1.1	Requirements	1-1
1.2	Organization	1-1

Chapter 2 Instrument Model

2.1	Signal Routing	2-2
2.1.1	Setting the Destination for Data Flow	2-2
2.2	Measurement Function	2-3
2.2.1	INPut	2-3
2.2.2	SENSe	2-4
2.2.3	CALCulate	2-4
2.3	Signal Generation	2-4
2.3.1	OUTPut	2-4
2.3.2	SOURce	2-5
2.3.3	CALCulate	2-5
2.4	TRIGger	2-5
2.5	MEMory	2-5
2.6	FORMat	2-5
2.7	Internal Routing	2-5
2.7.1	Data and Control Flow	2-6
2.7.2	Numeric Suffixes	2-7
2.7.3	Lamina and Cloned Models	2-7
2.7.3.1	FEED Exceptions	2-8
2.7.3.2	Amorphous SENSe Model	2-8
2.7.3.3	Other Subsystems	2-9
2.7.4	FEED <data_handle>	2-9
2.7.4.1	Sensor Function Selection	2-9
2.7.4.2	FEEDs from CALCulate Sub-Blocks	2-12
2.7.5	COMBine	2-12
2.7.6	Memory Associated with Data Flow	2-12
2.7.7	Querying the Data Flow	2-14
2.7.8	<event_handle>	2-14

Chapter 3 Measurement Instructions

3.1	CONFigure:<function> <parameters>[,<source list>]	3-2
3.2	FETCh[:<function>]? <parameters>[,<source list>]	3-3
3.3	READ[:<function>]? <parameters> [,<source list>]	3-4
3.4	MEASure:<function>? <parameters>[,<source list>]	3-5
3.5	<function>	3-5

1999 SCPI Command Reference

3.6	Presentation Layer	3-7
3.6.1	Presentation Command Summary	3-7
3.6.2	[:SCALar]	3-7
3.6.3	:ARRay <size>	3-7
3.7	Fundamental Measurement Layer	3-7
3.7.1	Fundamental Measurement Command Summary	3-7
3.7.2	:VOLTag [<expected_value>[,<resolution>]]	3-8
3.7.3	:CURREnt [<expected_value>[,<resolution>]]	3-8
3.7.4	:POWEr [<expected_value>[,<resolution>]]	3-8
3.7.5	:RESistance [<expected_value>[,<resolution>]]	3-8
3.7.6	:FRESistance [<expected_value>[,<resolution>]]	3-8
3.7.7	:TEMPerature [<transducer>[,<type> [,<expected_value>[,<resolution>]]]]	3-8
3.8	Measurement Function Layer	3-9
3.8.1	Simple Measurements	3-9
3.8.1.1	Simple Measurements Command Summary	3-9
3.8.1.2	:AC	3-9
3.8.1.3	[:DC]	3-9
3.8.1.4	:FREQuency [<expected_value>[,<resolution>]]	3-9
3.8.1.4.1	:BURSt [<expected_value>[,<resolution>]]	3-9
3.8.1.4.2	:PRF [<expected_value>[,<resolution>]]	3-10
3.8.1.5	:PERiod [<expected_value>[,<resolution>]]	3-10
3.8.1.6	:PHASe [<expected_value>[,<resolution>]]	3-10
3.8.2	Time Domain Waveform Measurements	3-10
3.8.2.1	Waveform Measurements Command Summary	3-10
3.8.2.2	:AMPLitude	3-11
3.8.2.3	:LOW	3-12
3.8.2.4	:HIGH	3-12
3.8.2.5	:RISE	3-12
3.8.2.5.1	:TIME [<low reference>[,<high reference> [,<expected_value>[,<resolution>]]]]	3-12
3.8.2.5.2	:OVERshoot	3-12
3.8.2.5.3	:PREShoot	3-12
3.8.2.6	:RTIME [<low reference>[,<high reference> [,<expected_value>[,<resolution>]]]]	3-12
3.8.2.7	:FALL	3-13
3.8.2.7.1	:TIME [<low reference>[,<high reference> [,<expected_value>[,<resolution>]]]]	3-13
3.8.2.7.2	:OVERshoot	3-13
3.8.2.7.3	:PREShoot	3-13
3.8.2.8	:FTIME [<low reference>[,<high reference> [,<expected_value>[,<resolution>]]]]	3-13
3.8.2.9	:PWIDth [<reference>]	3-13
3.8.2.10	:NWIDth [<reference>]	3-14
3.8.2.11	:PDUTYcycle[:DCYCLE [<reference>]	3-14
3.8.2.12	:NDUTYcycle [<reference>]	3-14

1999 SCPI Command Reference

3.8.2.13	:TMAXimum Layer:Time Domain Waveform Measurements	3-14
3.8.2.14	:TMINimum	3-14
3.8.2.15	:MINimum	3-14
3.8.2.16	:MAXimum	3-14
3.8.2.17	:PTPeak	3-14

Chapter 4 CALCulate Subsystem

4.1	:AVERage subsystem	4-6
4.1.1	:CLEar	4-6
4.1.2	:COUNT <numeric_value>	4-7
4.1.2.1	:AUTO <Boolean> ONCE	4-7
4.1.3	[:STATE] <Boolean>	4-7
4.1.4	:TCONtrol EXPonential MOVing NORMal REPeat	4-7
4.1.5	:TYPE COMpLex ENVELOpe MAXimum MINimum RMS SCALar	4-8
4.2	:CLIMits	4-9
4.2.1	:FAIL?	4-9
4.2.2	:FLIMits	4-9
4.2.2.1	[:DATA]?	4-9
4.2.2.2	:POINts?	4-9
4.3	:DATA?	4-10
4.3.1	:PREamble?	4-10
4.4	:DERivative	4-10
4.4.1	:STATe <Boolean>	4-10
4.4.2	:POINts <numeric_value>	4-10
4.5	:FEED <data_handle>	4-10
4.6	:FILTer	4-11
4.6.1	[:GATE]	4-11
4.6.1.1	:TIME	4-11
4.6.1.1.1	:STATe <Boolean>	4-11
4.6.1.1.2	[:TYPE] BPASs NOTCh	4-11
4.6.1.1.3	:START <numeric_value>	4-11
4.6.1.1.4	:STOP <numeric_value>	4-11
4.6.1.1.5	:SPAN <numeric_value>	4-12
4.6.1.1.6	:CENTER <numeric_value>	4-12
4.6.1.1.7	:POINts <numeric_value>	4-12
4.6.1.1.7.1	:AUTO <Boolean> ONCE	4-12
4.6.1.1.8	:WINDow RECTangular UNIFORM FLATtop HAMMING HANNing KBESSel FORCE EXPonential	4-12
4.6.1.1.9	:KBESSel <numeric_value>	4-12
4.6.1.1.10	:EXPonential <numeric_value>	4-12
4.6.1.1.11	:FORCE <numeric_value>	4-13
4.6.1.2	:FREQUency	4-13
4.6.1.2.1	:STATe <Boolean>	4-13
4.6.1.2.2	[:TYPE] BPASs NOTCh	4-13

1999 SCPI Command Reference

4.6.1.2.3	:START <numeric_value>	4-13
4.6.1.2.4	:STOP <numeric_value>	4-13
4.6.1.2.5	:SPAN <numeric_value>	4-14
4.6.1.2.6	:CENTer <numeric_value>	4-14
4.6.1.2.7	:POINts <numeric_value>	4-14
4.6.1.2.7.1	:AUTO <Boolean> ONCE	4-14
4.6.1.2.8	:WINDow RECTangular UNIFORM FLATtop HAMMING HANNing KBESsel FORCe EXPOntial	4-14
4.6.1.2.9	:KBESsel <numeric_value>	4-14
4.6.1.2.10	:EXPOntial <numeric_value>	4-14
4.6.1.2.11	:FORCe <numeric_value>	4-15
4.7	:FORMat NONE MLINear MLOGarithmic PHASel REAL IMAGinary SWRIGDELay COMPLex NYQuist NICHols POLar UPHase	4-15
4.7.1	:UPHase	4-16
4.7.1.1	:CREference <numeric_value>	4-17
4.7.1.2	:PREference <numeric value>	4-17
4.8	:GDAPerture	4-17
4.8.1	:SPAN <numeric_value>	4-17
4.8.2	:APERture <numeric_value>	4-17
4.9	:IMMediate	4-18
4.9.1	:AUTO <Boolean>	4-18
4.10	:INTEgral	4-18
4.10.1	:STATe <Boolean>	4-18
4.10.2	:TYPE SCALar MOVing	4-18
4.11	:LIMit	4-18
4.11.1	:STATe <Boolean>	4-19
4.11.2	:CONTRol	4-19
4.11.2.1	[:DATA] <numeric_value>{,<numeric_value>}	4-19
4.11.2.2	:POINts?	4-19
4.11.3	:UPPer	4-19
4.11.3.1	[:DATA] <numeric_value>{,<numeric_value>}	4-19
4.11.3.2	:POINts?	4-20
4.11.3.3	:STATe <Boolean>	4-20
4.11.4	:LOWer	4-20
4.11.4.1	[:DATA] <numeric_value>{,<numeric_value>}	4-20
4.11.4.2	:POINts?	4-20
4.11.4.3	:STATe <Boolean>	4-20
4.11.5	:FAIL?	4-20
4.11.6	:FCOunt?	4-20
4.11.7	:REPort	4-20
4.11.7.1	[:DATA]?	4-20
4.11.7.2	:POINts?	4-21
4.11.8	:CLEar	4-21

1999 SCPI Command Reference

4.11.8.1	:AUTO <Boolean> ONCE	4-21
4.11.8.2	[:IMMEDIATE]	4-21
4.11.9	:INTERpolate <Boolean>	4-21
4.12	:MATH	4-21
4.12.1	[:EXPRESSION] <numeric_expression>	4-21
4.12.1.1	:CATalog?	4-21
4.12.1.2	[:DEFine] <numeric_expression>	4-22
4.12.1.3	:DELete	4-22
4.12.1.3.1	[:SELected] <expression_name>	4-22
4.12.1.3.2	:ALL	4-22
4.12.1.4	:NAME <expression_name>	4-22
4.12.2	:STATe <Boolean>	4-23
4.13	:SMOothing	4-23
4.13.1	[:STATe] <Boolean>	4-23
4.13.2	:APERture <numeric_value>	4-23
4.13.3	:POINTs <numeric_value>	4-23
4.14	:STATe <Boolean>	4-23
4.15	:TRANsform	4-23
4.15.1	:HISTogram	4-24
4.15.1.1	:COUNT <numeric_value>	4-24
4.15.1.2	:ORDinate RATio PERCent PCT COUNT	4-24
4.15.1.3	:POINTs <numeric_value>	4-24
4.15.1.4	:RANGE	4-24
4.15.1.4.1	:AUTO <Boolean>	4-25
4.15.1.5	:STATe <Boolean>	4-25
4.15.2	:TIME	4-25
4.15.2.1	:STATe <Boolean>	4-25
4.15.2.2	[:TYPE] LPASS BPASS	4-25
4.15.2.3	:STIMulus STEP IMPulse	4-25
4.15.2.4	:STARt <numeric_value>	4-25
4.15.2.5	:STOP <numeric_value>	4-26
4.15.2.6	:SPAN <numeric_value>	4-26
4.15.2.7	:CENTer <numeric_value>	4-26
4.15.2.8	:POINTs <numeric_value>	4-26
4.15.2.8.1	:AUTO <Boolean> ONCE	4-26
4.15.2.9	:WINDOW RECTangular UNIFORM FLAT top HAMMING HANNing KBESSel FORCE EXPonential	4-26
4.15.2.10	:KBESSel <numeric_value>	4-27
4.15.2.11	:EXPonential <numeric_value>	4-27
4.15.2.12	:FORCE <numeric_value>	4-27
4.15.3	:DISTance	4-27
4.15.3.1	:STATe <Boolean>	4-27
4.15.3.2	[:TYPE] LPASS BPASS	4-27
4.15.3.3	:STIMulus STEP IMPulse	4-27
4.15.3.4	:STARt <numeric_value>	4-28

1999 SCPI Command Reference

4.15.3.5	:STOP <numeric_value>	4-28
4.15.3.6	:SPAN <numeric_value>	4-28
4.15.3.7	:CENTer <numeric_value>	4-28
4.15.3.8	:POINts <numeric_value>	4-28
4.15.3.8.1	:AUTO <Boolean> ONCE	4-28
4.15.3.9	:WINDow RECTangular UNIForm FLATtop HAMMING HANNing KBESsel FORCe EXponential	4-29
4.15.3.10	:KBESsel <numeric_value>	4-29
4.15.3.11	:EXponential <numeric_value>	4-29
4.15.3.12	:FORCe <numeric_value>	4-29
4.15.4	:FREquency	4-29
4.15.4.1	:STATe <Boolean>	4-29
4.15.4.2	[[:TYPE] LPASs BPASs	4-29
4.15.4.3	:STIMulus STEP IMPulse	4-30
4.15.4.4	:STARt <numeric_value>	4-30
4.15.4.5	:STOP <numeric_value>	4-30
4.15.4.6	:SPAN <numeric_value>	4-30
4.15.4.7	:CENTer <numeric_value>	4-30
4.15.4.8	:POINts <numeric_value>	4-31
4.15.4.8.1	:AUTO <Boolean> ONCE	4-31
4.15.4.9	:WINDow RECTangular UNIForm FLATtop HAMMING HANNing KBESsel FORCe EXponential	4-31
4.15.4.10	:KBESsel <numeric_value>	4-31
4.15.4.11	:EXponential <numeric_value>	4-31
4.15.4.12	:FORCe <numeric_value>	4-31
4.16	:PATH(MATH TRANsform FILTer SMOothing FORMat LIMit AVERage) {,(MATH TRANsform FILTer SMOothing FORMat LIMit AVERage)}	4-31

Chapter 5 CALibration Subsystem

5.1	[[:ALL]	5-2
5.2	[[:ALL]]?	5-2
5.3	:AUTO <Boolean> ONCE	5-2
5.4	:BINertia	5-2
5.4.1	:AVERage?	5-2
5.4.2	:HSPeet <numeric_value>	5-3
5.4.3	:INITiate	5-3
5.4.4	:LSPeet <numeric_value>	5-3
5.4.5	:NRUNs <numeric_value>	5-3
5.4.6	:SDEViation?	5-4
5.4.7	:UPDate	5-4
5.5	:DATA <arbitrary block program data>	5-4
5.6	:PLOSs	5-4
5.6.1	:APCoeff <numeric_value>,<numeric_value>,<numeric_value>[,<numeric_value>] ..	5-4
5.6.2	:INITiate	5-4

1999 SCPI Command Reference

5.6.3	:LATime	5-5
5.6.4	:STIME <numeric_value>	5-5
5.6.5	:UPDate	5-5
5.7	:SOURce INTernal EXTernal	5-6
5.8	:STATe <Boolean>	5-6
5.9	:VALue <numeric_value>	5-6
5.10	:WARMup	5-6
5.10.1	:INITiate	5-6
5.10.2	:SPEed <numeric_value>	5-7
5.10.3	:TIMEout <numeric_value>	5-7
5.11	:ZERO	5-7
5.11.1	:AUTO <Boolean> ONCE	5-7
5.11.2	:FSEnsor	5-7
5.11.2.1	:INITiate	5-8
5.11.2.2	:LATime <numeric_value>	5-8
5.11.2.3	:LEVel?	5-8
5.11.2.4	:SPEed <numeric_value>	5-8
5.11.2.5	:STIME <numeric_value>	5-8
5.11.2.6	:UPDate	5-9

Chapter 6 CONTROL Subsystem

6.1	:APOWer	6-1
6.1.1	[:STATe] <Boolean>	6-1
6.2	:BLOWer	6-2
6.2.1	[:STATe] <Boolean>	6-2
6.3	:BRAKe	6-2
6.3.1	[:STATe] <Boolean>	6-2
6.4	:COMPRESSor	6-2
6.4.1	[:STATe] <Boolean>	6-2
6.5	:COVER	6-2
6.5.1	[:ADJust] OPEN CLOSe SOPEn SCLOSe	6-2
6.5.2	:POSition?	6-3
6.6	:EBENch	6-3
6.6.1	:CLEan	6-3
6.6.1.1	:INITiate	6-3
6.6.1.2	:DURation <numeric_value>	6-3
6.7	:IDLE	6-3
6.7.1	:INITiate	6-3
6.8	:LIFT	6-3
6.8.1	[:ADJust] UP DOWN	6-4
6.8.2	:POSition?	6-4
6.9	:MCONtrol	6-4
6.9.1	[:STATe] <Boolean>	6-4
6.10	:ROTation	6-4

1999 SCPI Command Reference

6.10.1	:DIRection]	6-4
6.11	:VCDevice	6-4
6.11.1	:STATe] <Boolean>	6-4
6.11.2	:TDIameter <numeric_value>	6-5

Chapter 7 DIAGnostic Subsystem

Chapter 8 DISPlay Subsystem

8.1	:ANNotation	8-4
8.1.1	:ALL] <Boolean>	8-4
8.1.2	:AMPLitude <Boolean>	8-4
8.1.3	:FREQuency <Boolean>	8-4
8.2	:BRIGhtness <numeric_value>	8-4
8.3	:CMAP	8-4
8.3.1	:DEFault	8-4
8.3.2	:COLor	8-4
8.3.2.1	:HSL <hue>,<sat>,<lum>	8-5
8.3.2.2	:RGB <red>,<green>,<blue>	8-5
8.4	:CONTrast <numeric_value>	8-5
8.5	:ENABle <Boolean>	8-5
8.6	:MENU	8-6
8.6.1	:NAME] <menu_name>	8-6
8.6.2	:STATe <Boolean>	8-6
8.6.3	:KEY <string>	8-6
8.7	:WINDow]	8-6
8.7.1	:BACKground	8-6
8.7.1.1	:COLor <numeric_value>	8-6
8.7.2	:GEOMetry	8-6
8.7.2.1	:LLEFt <numeric_value>,<numeric_value>	8-7
8.7.2.2	:SIZE <numeric_value>,<numeric_value>	8-7
8.7.2.3	:URIGht <numeric_value>,<numeric_value>	8-7
8.7.3	:GRAPHics	8-7
8.7.3.1	:CLEar	8-7
8.7.3.2	:COLor <numeric_value>	8-7
8.7.3.3	:CSIZE <numeric_value>[,<numeric_value>]	8-7
8.7.3.4	:DRAW] <numeric_value>,<numeric_value>	8-8
8.7.3.5	:PCL <block>	8-8
8.7.3.6	:HPGL <block>	8-8
8.7.3.7	:IDRaw <numeric_value>,<numeric_value>	8-8
8.7.3.8	:IMOVe <numeric_value>,<numeric_value>	8-8
8.7.3.9	:LABel <string>	8-8
8.7.3.10	:LDIRection <numeric_value>	8-8
8.7.3.11	:LTYPE <numeric_value>[,<numeric_value>]	8-8
8.7.3.12	:MOVE <numeric_value>,<numeric_value>	8-9

1999 SCPI Command Reference

8.7.3.13	:STATe <Boolean>	8-9
8.7.4	[:STATe] <Boolean>	8-9
8.7.5	:TEXT	8-9
8.7.5.1	:ATTRIBUTES <Boolean>	8-9
8.7.5.2	:CLEAr	8-9
8.7.5.3	:COLor <numeric_value>	8-9
8.7.5.4	:CSIZe <numeric_value>[,<numeric_value>]	8-10
8.7.5.5	:FEED <data_handle>	8-10
8.7.5.6	[:DATA] <string> <block>	8-10
8.7.5.7	:LOCate <numeric_value>[,<numeric_value>]	8-10
8.7.5.8	:PAGE <numeric_value>	8-10
8.7.5.9	:STATe <Boolean>	8-10
8.7.6	:TRACe	8-10
8.7.6.1	:COLor <numeric_value>	8-10
8.7.6.2	:FEED <data_handle>	8-10
8.7.6.3	:GRATicule	8-11
8.7.6.3.1	:AXIS	8-11
8.7.6.3.1.1	[:STATe] <Boolean>	8-11
8.7.6.3.2	:FRAMe	8-11
8.7.6.3.2.1	[:STATe] <Boolean>	8-11
8.7.6.3.3	:GRID	8-11
8.7.6.3.3.1	:AUTO <Boolean>	8-11
8.7.6.3.3.2	[:STATe] <Boolean>	8-11
8.7.6.4	:PERSistence <numeric_value>	8-11
8.7.6.4.1	:AUTO <Boolean> ONCE	8-11
8.7.6.5	:STATe <Boolean>	8-11
8.7.6.6	:X	8-12
8.7.6.6.1	:LABel <string>	8-12
8.7.6.6.2	[:SCALE]	8-12
8.7.6.6.2.1	:AUTO <Boolean> ONCE	8-12
8.7.6.6.2.2	:CENTer <numeric_value>	8-12
8.7.6.6.2.3	:LEFT <numeric_value>	8-12
8.7.6.6.2.4	:PDIVision <numeric_value>	8-13
8.7.6.6.2.5	:RIGHT <numeric_value>	8-13
8.7.6.7	:Y	8-13
8.7.6.7.1	:LABel <string>	8-13
8.7.6.7.2	:RLINe <Boolean>	8-13
8.7.6.7.3	[:SCALE]	8-13
8.7.6.7.3.1	:AUTO <Boolean> ONCE	8-14
8.7.6.7.3.2	:BOTTom <numeric_value>	8-14
8.7.6.7.3.3	:PDIVision <numeric_value>	8-14
8.7.6.7.3.4	:RLEVel <numeric_value>	8-14
8.7.6.7.3.5	:RPOSition <numeric_value>	8-15
8.7.6.7.3.6	:TOP <numeric_value>	8-15
8.7.6.7.4	:SPACing LOGarithmic LINear	8-15

1999 SCPI Command Reference

8.7.6.8	:R	8-15
8.7.6.8.1	:LABel <string>	8-15
8.7.6.8.2	[[:SCALE]	8-15
8.7.6.8.2.1	:AUTO <Boolean> ONCE	8-15
8.7.6.8.2.2	:CPOint <numeric_value>	8-15
8.7.6.8.2.3	:OEDGE <numeric_value>	8-16
8.7.6.8.3	:SPACing LOGarithmic LINear	8-16

Chapter 9 FORMat Subsystem

9.1	:BORDER NORMallSWAPped	9-1
9.2	[[:DATA] <type>[,<length>]	9-1
9.3	:DINTERchange <Boolean>	9-2
9.4	:SREGister ASCii BINary HEXadecimal OCTal	9-3

Chapter 10 HCOPy

10.1	:ABORt	10-4
10.2	:DATA?	10-4
10.3	:DESTination<data_handle>	10-4
10.4	:DEVice	10-5
10.4.1	:CMAP	10-5
10.4.1.1	:COLor	10-5
10.4.1.1.1	:HSL <hue>,<sat>,<lum>	10-5
10.4.1.1.2	:RGB <red>,<green>,<blue>	10-5
10.4.1.2	:DEFault	10-6
10.4.2	:COLor <Boolean>	10-6
10.4.3	:LANGuage PCL[<n>] HPGL[<n>] POSTscript[<n>]	10-6
10.4.4	:MODE TABLE GRAPH	10-7
10.4.5	:RESolution <numeric_value>	10-7
10.4.5.1	:UNIT <SUFFIX PROGRAM DATA>	10-7
10.4.6	:SPEed <numeric_value>	10-7
10.4.6.1	:UNIT <SUFFIX PROGRAM DATA>	10-7
10.5	:FEED <data_handle>	10-7
10.6	[[:IMMediate]	10-7
10.7	:ITEM	10-7
10.7.1	:ALL	10-8
10.7.1.1	:DATA?	10-8
10.7.1.2	[[:IMMediate]	10-8
10.7.2	:ANNotation	10-8
10.7.2.1	:COLor <numeric_value>	10-8
10.7.2.2	:DATA?	10-8
10.7.2.3	[[:IMMediate]	10-8
10.7.2.4	:STATe <Boolean>	10-8
10.7.3	:CUT	10-8
10.7.3.1	:DATA?	10-9

1999 SCPI Command Reference

10.7.3.2	[:IMMediate]	10-9
10.7.3.3	:STATe <Boolean>	10-9
10.7.4	:FFeEd	10-9
10.7.4.1	:DATA?	10-9
10.7.4.2	[:IMMediate]	10-9
10.7.4.3	:STATe <Boolean>	10-9
10.7.5	:LABel	10-9
10.7.5.1	:COLor <numeric_value>	10-9
10.7.5.2	:DATA?	10-10
10.7.5.3	[:IMMediate]	10-10
10.7.5.4	:STATe <Boolean>	10-10
10.7.5.5	:TEXT <string>	10-10
10.7.6	:MENU	10-10
10.7.6.1	:COLor <numeric_value>	10-10
10.7.6.2	:DATA?	10-10
10.7.6.3	[:IMMediate]	10-10
10.7.6.4	:STATe <Boolean>	10-10
10.7.7	:TDSamp	10-11
10.7.7.1	:COLor <numeric_value>	10-11
10.7.7.2	:DATA?	10-11
10.7.7.3	[:IMMediate]	10-11
10.7.7.4	:STATe <Boolean>	10-11
10.7.8	[:WINDow]	10-11
10.7.8.1	:DATA?	10-11
10.7.8.2	[:IMMediate]	10-11
10.7.8.3	:STATe <Boolean>	10-11
10.7.8.4	:TEXT	10-11
10.7.8.4.1	:COLor <numeric_value>	10-12
10.7.8.4.2	:DATA?	10-12
10.7.8.4.3	[:IMMediate]	10-12
10.7.8.4.4	:STATe <Boolean>	10-12
10.7.8.5	:TRACe	10-12
10.7.8.5.1	:COLor <numeric_value>	10-12
10.7.8.5.2	:DATA?	10-12
10.7.8.5.3	:GRATICule	10-12
10.7.8.5.3.1	:COLor <numeric_value>	10-12
10.7.8.5.3.2	:DATA?	10-12
10.7.8.5.3.3	[:IMMediate]	10-13
10.7.8.5.3.4	:STATe <Boolean>	10-13
10.7.8.5.4	[:IMMediate]	10-13
10.7.8.5.5	:LTYPe SOLid DOTTed DASHed STYLE<n>	10-13
10.7.8.5.6	:STATe <Boolean>	10-13
10.8	:PAGE	10-13
10.8.1	:DIMensions	10-13
10.8.1.1	:AUTO <Boolean>	10-13

1999 SCPI Command Reference

10.8.1.2	:LLEFt <numeric_value>,<numeric_value>	10-14
10.8.1.3	:QUADrant[<n>]	10-14
10.8.1.4	:URIGht <numeric_value>,<numeric_value>	10-14
10.8.2	:LENGth <numeric_value>	10-14
10.8.3	:ORientation LANDscape PORTrait	10-14
10.8.4	:SCALE <numeric_value>	10-14
10.8.5	:SIZE CUSTom A B C D E A0 A1 A2 A3 A4 B0 B1 B2 B3 B4 B5	10-15
10.8.6	:UNIT <SUFFIX PROGRAM DATA>	10-15
10.8.7	:WIDTh <numeric_value>	10-15
10.9	:SDUMp	10-15
10.9.1	:DATA?	10-15
10.9.2	[:IMMediate]	10-15

Chapter 11 INPut Subsystem

11.1	:ATTenuation <numeric_value>	11-3
11.1.1	:AUTO <Boolean> ONCE	11-3
11.1.2	:STATe <Boolean>	11-3
11.2	:BIAS	11-3
11.2.1	:CURRent	11-3
11.2.1.1	:AC <numeric_value>	11-4
11.2.1.2	[:DC] <numeric_value>	11-4
11.2.2	[:STATe] <Boolean>	11-4
11.2.3	:TYPE CURRent VOLTage	11-4
11.2.4	:VOLTage	11-4
11.2.4.1	:AC <numeric_value>	11-4
11.2.4.2	[:DC] <numeric_value>	11-4
11.3	:COUPling AC DC GROund	11-4
11.4	:FILTer	11-5
11.4.1	:AWEighting	11-5
11.4.1.1	[:STATe] <Boolean>	11-5
11.4.2	:HPASs	11-5
11.4.2.1	:FREQuency <numeric_value>	11-5
11.4.2.2	[:STATe] <Boolean>	11-5
11.4.3	[:LPASs]	11-5
11.4.3.1	:FREQuency <numeric_value>	11-5
11.4.3.2	[:STATe] <Boolean>	11-5
11.5	:GAIN <numeric_value>	11-6
11.5.1	:AUTO <Boolean> ONCE	11-6
11.5.2	:STATe <Boolean>	11-6
11.6	:GUARd LOW FLOat	11-6
11.7	:IMPedance <numeric_value>	11-6
11.8	:LOW FLOat GROund	11-6
11.9	:OFFSet <numeric_value>	11-7
11.9.1	:STATe <Boolean>	11-7

1999 SCPI Command Reference

11.10	:POLarity NORMallINVerted	11-7
11.11	:POLarization <numeric_value>	11-7
11.11.1	:HORizontal	11-7
11.11.2	:VERTical	11-7
11.12	:POSition	11-7
11.12.1	[X]	11-8
11.12.1.1	:ANGLE	11-8
11.12.1.1.1	:DIRection UPIDOWN	11-8
11.12.1.1.2	[IMMediate] <numeric_value>	11-8
11.12.1.1.3	:LIMit	11-8
11.12.1.1.3.1	:HIGH <numeric_value>	11-8
11.12.1.1.3.2	:LOW <numeric_value>	11-8
11.12.1.1.3.3	:STATe <Boolean>	11-8
11.12.1.1.4	:OFFSet <numeric_value>	11-8
11.12.1.1.5	:VELocity <numeric_value>	11-9
11.12.1.2	[DISTance]	11-9
11.12.1.2.1	:DIRection UPIDOWN	11-9
11.12.1.2.2	[IMMediate] <numeric_value>	11-9
11.12.1.2.3	:LIMit	11-9
11.12.1.2.3.1	:HIGH <numeric_value>	11-9
11.12.1.2.3.2	:LOW <numeric_value>	11-9
11.12.1.2.3.3	:STATe <Boolean>	11-9
11.12.1.2.4	:OFFSet <numeric_value>	11-9
11.12.1.2.5	:VELocity <numeric_value>	11-10
11.12.2	:Y	11-10
11.12.2.1	:ANGLE	11-10
11.12.2.1.1	:DIRection UPIDOWN	11-10
11.12.2.1.2	[IMMediate] <numeric_value>	11-10
11.12.2.1.3	:LIMit	11-10
11.12.2.1.3.1	:HIGH <numeric_value>	11-10
11.12.2.1.3.2	:LOW <numeric_value>	11-10
11.12.2.1.3.3	:STATe <Boolean>	11-10
11.12.2.1.4	:OFFSet <numeric_value>	11-10
11.12.2.1.5	:VELocity <numeric_value>	11-11
11.12.2.2	[DISTance]	11-11
11.12.2.2.1	:DIRection UPIDOWN	11-11
11.12.2.2.2	[IMMediate] <numeric_value>	11-11
11.12.2.2.3	:LIMit	11-11
11.12.2.2.3.1	:HIGH <numeric_value>	11-11
11.12.2.2.3.2	:LOW <numeric_value>	11-11
11.12.2.2.3.3	:STATe <Boolean>	11-11
11.12.2.2.4	:OFFSet <numeric_value>	11-11
11.12.2.2.5	:VELocity <numeric_value>	11-12
11.12.3	:Z	11-12
11.12.3.1	:ANGLE	11-12

1999 SCPI Command Reference

11.12.3.1.1	:DIRection UPIDOWN	11-12
11.12.3.1.2	[:IMMediate] <numeric_value>	11-12
11.12.3.1.3	:LIMit	11-12
11.12.3.1.3.1	:HIGH <numeric_value>	11-12
11.12.3.1.3.2	:LOW <numeric_value>	11-12
11.12.3.1.3.3	:STATe <Boolean>	11-12
11.12.3.1.4	:OFFSet <numeric_value>	11-12
11.12.3.1.5	:VELocity <numeric_value>	11-13
11.12.3.2	[:DISTance]	11-13
11.12.3.2.1	:DIRection UPIDOWN	11-13
11.12.3.2.2	[:IMMediate] <numeric_value>	11-13
11.12.3.2.3	:LIMit	11-13
11.12.3.2.3.1	:HIGH <numeric_value>	11-13
11.12.3.2.3.2	:LOW <numeric_value>	11-13
11.12.3.2.3.3	:STATe <Boolean>	11-13
11.12.3.2.4	:OFFSet <numeric_value>	11-13
11.12.3.2.5	:VELocity <numeric_value>	11-14
11.13	[:STATe] <Boolean>	11-14
11.14	:TYPE <character data>	11-14

Chapter 12 INSTRument Subsystem

12.1	:CATalog?	12-1
12.1.1	:FULL?	12-1
12.2	:COUPle[:<subsystem>] ALL NONE <list>	12-2
12.3	:DEFine	12-2
12.3.1	:GROup <identifier>,<identifier_list>	12-2
12.3.2	[:NAME] <identifier>,<numeric_value>	12-3
12.4	:DELeTe	12-3
12.4.1	:ALL	12-3
12.4.2	[:NAME] <identifier>	12-3
12.5	:NSElect <numeric_value>	12-4
12.6	[:SElect] <identifier>	12-4
12.7	:STATe <Boolean>	12-4

Chapter 13 MEMory Subsystem

13.1	:CATalog	13-4
13.1.1	[:ALL]?	13-5
13.1.2	:ASCIi?	13-5
13.1.3	:BINary?	13-5
13.1.4	:MACRo?	13-5
13.1.5	:STATe?	13-5
13.1.6	:TABLe?	13-6
13.2	:CLEar	13-6
13.2.1	[:NAME] <name>	13-6

1999 SCPI Command Reference

13.2.2	:TABLE	13-6
13.3	:COPY	13-6
13.3.1	[:NAME] <name>,<name>	13-6
13.3.2	:TABLE <name>	13-7
13.4	:DATA<name>,<data>	13-7
13.5	:DELeTe	13-7
13.5.1	:ALL	13-7
13.5.2	[:NAME] <name>	13-7
13.6	:EXCHange	13-7
13.6.1	[:NAME] <name>,<name>	13-7
13.6.2	:TABLE <name>	13-8
13.7	:FREE	13-8
13.7.1	[:ALL]?	13-8
13.7.2	:ASCIi?	13-8
13.7.3	:BINary?	13-8
13.7.4	:MACRo?	13-8
13.7.5	:STATe?	13-9
13.7.6	:TABLE?	13-9
13.8	:NSTates?	13-9
13.9	:STATe	13-9
13.9.1	:CATalog?	13-9
13.9.2	:DEFine <name> ,<register_number>	13-9
13.10	:TYPE? <name>	13-9
13.11	:TABLE	13-10
13.11.1	:BNUMber <numeric_value> {,<numeric_value>}	13-10
13.11.1.1	:POINts?	13-10
13.11.2	:CCURve <numeric_value> {,<numeric_value>}	13-10
13.11.2.1	:POINts?	13-10
13.11.3	:CONCetration <numeric_value> {,<numeric_value>}	13-10
13.11.3.1	:POINts?	13-10
13.11.4	:CONDition	13-10
13.11.4.1	[:MAGNitude] <Boolean>{,<Boolean>}	13-11
13.11.4.1.1	:POINts?	13-11
13.11.5	:CPOint <numeric_value> {,<numeric_value>}	13-11
13.11.5.1	:POINts?	13-11
13.11.6	:CURRent	13-11
13.11.6.1	[:MAGNitude] <numeric_value>{,<numeric_value>}	13-11
13.11.6.1.1	:POINts?	13-11
13.11.6.2	:PHASe <numeric_value>{,<numeric_value>}	13-11
13.11.6.2.1	:POINts?	13-12
13.11.7	:DFACtory <numeric_value> {,<numeric_value>}	13-12
13.11.7.1	:POINts?	13-12
13.11.8	:DLASt <numeric_value> {,<numeric_value>}	13-12
13.11.8.1	:POINts?	13-13
13.11.9	:DLINearize <numeric_value> {,<numeric_value>}	13-13

1999 SCPI Command Reference

13.11.9.1	:POINts?	13-13
13.11.10	:EXPeCted <numeric_value> {,<numeric_value>}	13-13
13.11.10.1	:POINts?	13-14
13.11.11	:DEFiNe <structure_string>[,<numeric_value>]	13-14
13.11.12	:FORCe	13-14
13.11.12.1	[:MAGNitude] <numeric_value>{,<numeric_value>}	13-14
13.11.12.1.1	:POINts?	13-14
13.11.13	:FREQuency <numeric_value>{,<numeric_value>}	13-15
13.11.13.1	:POINts?	13-15
13.11.14	:LABEl <string> {,<string>}	13-15
13.11.14.1	:POINts?	13-15
13.11.15	:LLIMit <numeric_value> {,<numeric_value>}	13-15
13.11.15.1	:POINts?	13-15
13.11.16	:LOG <string> {,<string>}	13-15
13.11.16.1	:POINts?	13-16
13.11.17	:LOSS	13-16
13.11.17.1	[:MAGNitude] <numeric_value>{,<numeric_value>}	13-16
13.11.17.1.1	:POINts?	13-16
13.11.17.2	:PHASe <numeric_value>{,<numeric_value>}	13-16
13.11.17.2.1	:POINts?	13-16
13.11.18	:NCURve <numeric_value> {,<numeric_value>}	13-16
13.11.18.1	:POINts?	13-16
13.11.19	:POWeR	13-17
13.11.19.1	[:MAGNitude] <numeric_value>{,<numeric_value>}	13-17
13.11.19.1.1	:POINts?	13-17
13.11.20	:RAW <numeric_value> {,<numeric_value>}	13-17
13.11.20.1	:POINts?	13-17
13.11.21	:RESistance	13-17
13.11.21.1	[:MAGNitude] <numeric_value>{,<numeric_value>}	13-17
13.11.21.1.1	:POINts?	13-17
13.11.22	:SELeCt <name>	13-18
13.11.23	:SPEed	13-18
13.11.23.1	[:MAGNitude] <numeric_value>{,<numeric_value>}	13-18
13.11.23.1.1	:POINts?	13-18
13.11.24	:TIME	13-18
13.11.24.1	[:MAGNitude] <numeric_value>{,<numeric_value>}	13-18
13.11.24.1.1	:POINts?	13-18
13.11.25	:TOLerance <numeric_value> {,<numeric_value>}	13-18
13.11.25.1	:POINts?	13-19
13.11.26	:ULIMit <numeric_value> {,<numeric_value>}	13-19
13.11.26.1	:POINts?	13-19
13.11.27	:VOLTagE	13-19
13.11.27.1	[:MAGNitude] <numeric_value>{,<numeric_value>}	13-19
13.11.27.1.1	:POINts?	13-19
13.11.27.2	:PHASe <numeric_value>{,<numeric_value>}	13-19

1999 SCPI Command Reference

13.11.27.2.1	:POINts?	13-20
13.11.28	:WFACTOR <numeric_value> {,<numeric_value>}	13-20
13.11.28.1	:POINts?	13-20

Chapter 14 MMEMorySubsystem

14.1	:CATalog? [<msus>]	14-2
14.2	:CDIRectory [<directory_name>]	14-3
14.3	:CLOSE	14-3
14.4	:COPY <file_source>,<file_destination>	14-3
14.5	:DATA <file_name>,<data>	14-3
14.6	:DELeTe <file_name>[,<msus>]	14-3
14.7	:FEED <data_handle>	14-4
14.8	:INITialize [<msus>[(LIFIDOSIHFS)[,<numeric_value>]]]	14-4
14.9	:LOAD and :STORE	14-4
14.9.1	:DINTerchange <label>,<file_name>[,<msus>]	14-4
14.9.1.1	:TRACe <label>,<file_name>[,<msus>]	14-4
14.9.2	:MACRO <label>,<file_name>[,<msus>]	14-4
14.9.3	:STATe <numeric_value>,<file_name>[,<msus>]	14-4
14.9.4	:TABLe <label>,<file_name>[,<msus>]	14-5
14.9.5	:TRACe <label>,<file_name>[,<msus>]	14-5
14.10	:MSIS [<msus>]	14-5
14.11	:MOVE (<src_file>,<dest_file>) l(<src_file>,<src_msus>,<dest_file>,<dest_msus>)	14-5
14.12	:NAME <file_name>[,<msus>]	14-5
14.13	:OPEN	14-6
14.14	:PACK [<msus>]	14-6

Chapter 15 OUTPut Subsystem

15.1	:ATTenuation <numeric_value>	15-3
15.2	:COUPling ACIDC	15-3
15.3	:FILTer	15-3
15.3.1	:AUTO <Boolean> ONCE	15-3
15.3.2	:EXTernal	15-3
15.3.2.1	[:STATe] <Boolean>	15-4
15.3.3	:HPASs	15-4
15.3.3.1	:FREQuency <numeric_value>	15-4
15.3.3.2	[:STATe] <Boolean>	15-4
15.3.3.3	:TYPE BESSel CHEByshev	15-4
15.3.4	[:LPASs]	15-4
15.3.4.1	:FREQuency <numeric_value>	15-4
15.3.4.2	[:STATe] <Boolean>	15-4
15.3.4.3	:TYPE BESSel CHEByshev	15-4
15.4	:IMPedance <numeric_value>	15-5
15.5	:LOW FLOatIGROund	15-5

1999 SCPI Command Reference

15.6	:POLarity NORMal INVerted	15-5
15.7	:POLarization <numeric_value>	15-5
15.7.1	:HORizontal	15-5
15.7.2	:VERTical	15-5
15.8	:POSition	15-5
15.8.1	[:X]	15-5
15.8.1.1	:ANGLE	15-6
15.8.1.1.1	:DIRection UPIDOWN	15-6
15.8.1.1.2	[:IMMediate] <numeric_value>	15-6
15.8.1.1.3	:LIMit	15-6
15.8.1.1.3.1	:HIGH <numeric_value>	15-6
15.8.1.1.3.2	:LOW <numeric_value>	15-6
15.8.1.1.3.3	:STATE <Boolean>	15-6
15.8.1.1.4	:OFFSet <numeric_value>	15-6
15.8.1.1.5	:VELOCITY <numeric_value>	15-6
15.8.1.2	[:DISTance]	15-7
15.8.1.2.1	:DIRection UPIDOWN	15-7
15.8.1.2.2	[:IMMediate] <numeric_value>	15-7
15.8.1.2.3	:LIMit	15-7
15.8.1.2.3.1	:HIGH <numeric_value>	15-7
15.8.1.2.3.2	:LOW <numeric_value>	15-7
15.8.1.2.3.3	:STATE <Boolean>	15-7
15.8.1.2.4	:OFFSet <numeric_value>	15-7
15.8.1.2.5	:VELOCITY <numeric_value>	15-7
15.8.2	:Y	15-8
15.8.2.1	:ANGLE	15-8
15.8.2.1.1	:DIRection UPIDOWN	15-8
15.8.2.1.2	[:IMMediate] <numeric_value>	15-8
15.8.2.1.3	:LIMit	15-8
15.8.2.1.3.1	:HIGH <numeric_value>	15-8
15.8.2.1.3.2	:LOW <numeric_value>	15-8
15.8.2.1.3.3	:STATE <Boolean>	15-8
15.8.2.1.4	:OFFSet <numeric_value>	15-8
15.8.2.1.5	:VELOCITY <numeric_value>	15-8
15.8.2.2	[:DISTance]	15-9
15.8.2.2.1	:DIRection UPIDOWN	15-9
15.8.2.2.2	[:IMMediate] <numeric_value>	15-9
15.8.2.2.3	:LIMit	15-9
15.8.2.2.3.1	:HIGH <numeric_value>	15-9
15.8.2.2.3.2	:LOW <numeric_value>	15-9
15.8.2.2.3.3	:STATE <Boolean>	15-9
15.8.2.2.4	:OFFSet <numeric_value>	15-9
15.8.2.2.5	:VELOCITY <numeric_value>	15-9
15.8.3	:Z	15-10
15.8.3.1	:ANGLE	15-10

1999 SCPI Command Reference

15.8.3.1.1	:DIRection UPIDOWN	15-10
15.8.3.1.2	[:IMMediate] <numeric_value>	15-10
15.8.3.1.3	:LIMit	15-10
15.8.3.1.3.1	:HIGH <numeric_value>	15-10
15.8.3.1.3.2	:LOW <numeric_value>	15-10
15.8.3.1.3.3	:STATe <Boolean>	15-10
15.8.3.1.4	:OFFSet <numeric_value>	15-10
15.8.3.1.5	:VELocity <numeric_value>	15-10
15.8.3.2	[:DISTance]	15-11
15.8.3.2.1	:DIRection UPIDOWN	15-11
15.8.3.2.2	[:IMMediate] <numeric_value>	15-11
15.8.3.2.3	:LIMit	15-11
15.8.3.2.3.1	:HIGH <numeric_value>	15-11
15.8.3.2.3.2	:LOW <numeric_value>	15-11
15.8.3.2.3.3	:STATe <Boolean>	15-11
15.8.3.2.4	:OFFSet <numeric_value>	15-11
15.8.3.2.5	:VELocity <numeric_value>	15-11
15.9	:PROTection	15-12
15.9.1	:DELay <numeric_value>	15-12
15.9.2	[:STATe] <Boolean>	15-12
15.9.3	:TRIPped?	15-12
15.9.4	:CLEar	15-12
15.10	:ROSCillator	15-12
15.10.1	[:STATe] <Boolean>	15-12
15.11	:TTLTrg<n> :ECLTrg<n>	15-12
15.11.1	:IMMediate	15-12
15.11.2	:LEVel <Boolean>	15-13
15.11.3	:POLarity NORMal INVerted	15-13
15.11.4	:PROToCol SYNChronous SSYNchronous ASYNchronous	15-13
15.11.5	:WIDTh <numeric_value>	15-13
15.11.6	[:STATe] <Boolean>	15-13
15.11.7	:SOURce <character data>	15-13
15.12	[:STATe] <Boolean>	15-14
15.13	:TYPE <character data>	15-14

Chapter 16 PROgram Subsystem

16.1	:CATalog?	16-2
16.2	[:SELEcted]	16-2
16.2.1	:DEFine <program>	16-2
16.2.2	:DELeTe	16-2
16.2.2.1	[:SELEcted]	16-2
16.2.2.2	:ALL	16-2
16.2.3	:EXECute <program_command>	16-2
16.2.4	:MALLocate <nbytes> DEFault	16-3

1999 SCPI Command Reference

16.2.5	:NAME <progrname>	16-3
16.2.6	:NUMBer <varname>{,<nvalues>}	16-3
16.2.7	:STATe RUN PAUSelSTOP CONTinue	16-3
16.2.8	:STRing <varname>{,<svalues>}	16-4
16.2.9	:WAIT	16-4
16.3	:EXPLicit	16-4
16.3.1	:DEFine <progrname>,<program>	16-4
16.3.2	:DELete <progrname>	16-5
16.3.3	:EXECute <progrname>,<program_command>	16-5
16.3.4	:MALLocate <progrname>,<(nbytes> DEFault)	16-5
16.3.5	:NUMBer <progrname>,<varname>{,<nvalues>}	16-5
16.3.6	:STATe <progrname>,(RUN PAUSelSTOP CONTinue)	16-5
16.3.7	:STRing <progrname>,<varname>{,<svalues>}	16-6
16.3.8	:WAIT <progrname>	16-6

Chapter 17 ROUTe Subsystem

17.1	:CLOSe <channel_list>	17-1
17.1.1	:STATe?	17-2
17.2	:MODule	17-2
17.2.1	:CATalog?	17-2
17.2.2	[:DEFine] <module_name>,<module_address>	17-2
17.2.3	:DELete	17-2
17.2.3.1	:ALL	17-2
17.2.3.2	[:NAME] <module_name>	17-3
17.3	:OPeN <channel_list>	17-3
17.3.1	:ALL	17-3
17.4	:PATH	17-3
17.4.1	:CATalog?	17-3
17.4.2	[:DEFine] <path_name>,<channel_list>	17-3
17.4.3	:DELete	17-4
17.4.3.1	:ALL	17-4
17.4.3.2	[:NAME] <path_name>	17-4
17.5	:SAMPle	17-4
17.5.1	:CATalog?	17-4
17.5.2	[:OPeN] BAG DILutелPRE POST MID ICEFFiciency NONE ZERO SPAN VERify MANifold	17-4
17.6	:SCAN <channel_list>	17-6
17.7	TERMinals FRONt REAR BOTH NONE	17-6

Chapter 18 SENSE Subsystem

18.1	AM Subsystem	18-4
18.1.1	[:DEPTh]	18-4
18.1.1.1	:RANGe	18-4
18.1.1.1.1	:AUTO <Boolean> ONCE	18-4

1999 SCPI Command Reference

18.1.1.1.2	[[:UPPer] <numeric_value>	18-4
18.1.1.1.3	:LOWer <numeric_value>	18-4
18.1.2	:TYPE LINear LOGarithmic	18-4
18.2	AVERage Subsystem	18-6
18.2.1	A typical device action for SENS:AVERage	18-6
18.2.2	:COUNt <numeric_value>	18-7
18.2.2.1	:AUTO <Boolean> ONCE	18-7
18.2.3	[[:STATE] <Boolean>	18-7
18.2.4	:TCONtrol EXPOntial MOVing NORMal REPeat	18-7
18.2.5	:TYPE COMpLex ENVelope MAXimum MINimum RMS SCALar	18-8
18.3	BANDwidth BWIDth Subsystem	18-10
18.3.1	[[:RESolution] <numeric_value>	18-10
18.3.1.1	:AUTO <Boolean> ONCE	18-10
18.3.1.2	:RATio <numeric_value>	18-10
18.3.1.3	:TRACk <Boolean>	18-10
18.3.2	:VIDeo <numeric_value>	18-10
18.3.2.1	:AUTO <Boolean> ONCE	18-11
18.3.2.2	:RATio <numeric_value>	18-11
18.4	CONCentration Subsystem	18-12
18.4.1	:CSET <numeric_value>,<numeric_value>	18-12
18.4.2	:LOWer <numeric_value>	18-12
18.4.3	:LSET POLYnomial<n> SRATional<n>,<numeric_value>{,<numeric_value>} ..	18-12
18.4.4	:RANGe	18-13
18.4.4.1	:AUTO	18-13
18.4.4.1.1	:LOWer <numeric_value>	18-13
18.4.4.1.2	[[:STATe] <Boolean>	18-13
18.4.4.1.3	:UPPer <numeric_value>	18-13
18.4.4.2	[[:FIXed] <numeric_value>	18-13
18.4.5	:TALign <numeric_value>	18-13
18.4.6	:UPPer <numeric_value>	18-14
18.5	CONDition Subsystem	18-15
18.5.1	:LEVel <numeric_value> TTL ECL	18-15
18.6	CORRection Subsystem	18-16
18.6.1	:AUTO	18-17
18.6.2	:CALCulate	18-17
18.6.3	:COLlect	18-17
18.6.3.1	[[:ACQuire] STANdard	18-17
18.6.3.2	:METHod TPORT	18-18
18.6.3.3	:SAVE [<trace_name>]	18-18
18.6.4	:CSET	18-18
18.6.4.1	[[:SElect] <name>	18-18
18.6.4.2	:STATe <Boolean>	18-18
18.6.5	:EDELay	18-19
18.6.5.1	:DISTance <numeric_value>	18-19
18.6.5.2	:STATe <Boolean>	18-19

1999 SCPI Command Reference

18.6.5.3	[:TIME] <numeric_value>	18-19
18.6.6	:IMPedance	18-19
18.6.6.1	[:INPut]:OUTPut	18-19
18.6.6.1.1	[:MAGNitude] <numeric_value>	18-20
18.6.6.2	:STATe <Boolean>	18-20
18.6.7	:LOSSI:GAINI:SLOPe	18-20
18.6.7.1	[:INPut]:OUTPut	18-20
18.6.7.1.1	:AUTO ON OFF	18-20
18.6.7.1.2	[:MAGNitude] <numeric_value>	18-21
18.6.7.1.3	:PHASe <numeric_value>	18-21
18.6.7.2	:STATe <Boolean>	18-21
18.6.8	:OFFSet	18-21
18.6.8.1	[:MAGNitude] <numeric_value>	18-21
18.6.8.2	:PHASe <numeric_value>	18-21
18.6.8.3	:STATe <Boolean>	18-22
18.6.9	:RVELocity	18-22
18.6.9.1	:COAX <numeric_value>	18-22
18.6.9.2	:MEDium COAX WAVEguide	18-22
18.6.9.3	:STATe <Boolean>	18-22
18.6.9.4	:WAVEguide <numeric_value>	18-22
18.6.9.4.1	:FCUToff <numeric_value>	18-22
18.6.10	:SPOint	18-23
18.6.10.1	:ACQuire	18-23
18.6.10.2	:DTOLerance <numeric_value>	18-23
18.6.11	[:STATe] <Boolean>	18-23
18.6.12	:ZERO	18-23
18.6.12.1	:ACQuire	18-23
18.6.12.2	:DTOLerance <numeric_value>	18-23
18.7	CURRent Subsystem	18-25
18.7.1	:AC [:DC]	18-25
18.7.1.1	:APERture <numeric_value>	18-25
18.7.1.2	:NPLCycles <numeric_value>	18-26
18.7.1.3	:ATTenuation <numeric_value>	18-26
18.7.1.3.1	:AUTO <Boolean>	18-26
18.7.1.4	:PROTection	18-26
18.7.1.4.1	[:LEVel] <numeric_value>	18-26
18.7.1.4.2	:STATe <Boolean>	18-26
18.7.1.4.3	:TRIPped?	18-26
18.7.1.4.4	:CLEar	18-26
18.7.1.5	:RANGE	18-27
18.7.1.5.1	[:UPPer] <numeric_value>	18-27
18.7.1.5.2	:LOWer <numeric_value>	18-27
18.7.1.5.3	:AUTO <Boolean> ONCE	18-27
18.7.1.5.3.1	:DIREction UP DOWN EITHER	18-27
18.7.1.5.3.2	:LLIMit <numeric_value>	18-27

1999 SCPI Command Reference

18.7.1.5.3.3	:ULIMit <numeric_value>	18-27
18.7.1.5.4	:OFFSet <numeric_value>	18-28
18.7.1.5.5	:PTPeak <numeric_value>	18-28
18.7.1.6	:REFerence <numeric_value>	18-28
18.7.1.6.1	:STATe <Boolean>	18-28
18.7.1.7	:RESolution <numeric_value>	18-28
18.7.1.7.1	:AUTO <Boolean> ONCE	18-28
18.7.2	:DETector INTernal EXTernal	18-29
18.8	DETector Subsystem	18-30
18.8.1	:BANDwidth BWIDth	18-30
18.8.2	[:FUNction] <detector function>	18-30
18.8.2.1	:AUTO <Boolean> ONCE	18-31
18.8.3	:SHApe LINear LOGarithmic	18-31
18.9	DISTance Subsystem	18-32
18.9.1	:RESet	18-32
18.10	FILTer Subsystem	18-33
18.10.1	[:LPASs]	18-33
18.10.1.1	[:STATe] <Boolean>	18-33
18.10.1.2	:FREQuency <numeric_value>	18-33
18.10.2	:HPASs	18-33
18.10.2.1	[:STATe] <Boolean>	18-34
18.10.2.2	:FREQuency <numeric_value>	18-34
18.10.3	:DEMPhasis	18-34
18.10.3.1	[:STATe] <Boolean>	18-34
18.10.3.2	:TCONstant <numeric_value>	18-34
18.10.4	:CCITt	18-34
18.10.4.1	[:STATe] <Boolean>	18-34
18.10.5	:CMESsage	18-34
18.10.5.1	[:STATe] <Boolean>	18-34
18.10.6	:CCIR	18-35
18.10.6.1	[:STATe] <Boolean>	18-35
18.10.7	:CARM	18-35
18.10.7.1	[:STATe] <Boolean>	18-35
18.10.8	:AWEighting	18-35
18.10.8.1	[:STATe] <Boolean>	18-35
18.11	FM Subsystem	18-36
18.11.1	[:DEViation]	18-36
18.11.1.1	:RANGe	18-36
18.11.1.1.1	:AUTO <Boolean> ONCE	18-36
18.11.1.1.2	[:UPPer] <numeric_value>	18-36
18.11.1.1.3	:LOWer <numeric_value>	18-36
18.12	FREQuency Subsystem	18-37
18.12.1	:APERture <numeric_value>	18-37
18.12.2	:CENTer <numeric_value>	18-37
18.12.3	[:CW]:FIXed] <numeric_value>	18-38

1999 SCPI Command Reference

18.12.3.1	:AFC <Boolean> ONCE	18-38
18.12.3.2	:AUTO <Boolean> ONCE	18-38
18.12.4	:MANual <numeric_value>	18-38
18.12.5	:MODE CWIFIXed SWEep LIST SOURce	18-38
18.12.6	:MULTiplier <numeric_value>	18-39
18.12.7	:OFFSet <numeric_value>	18-39
18.12.8	:RANGe	18-39
18.12.8.1	[:UPPer] <numeric_value>	18-39
18.12.8.2	:LOWer <numeric_value>	18-39
18.12.8.3	:AUTO <Boolean> ONCE	18-39
18.12.9	:RESolution <numeric_value>	18-40
18.12.9.1	:AUTO <Boolean> ONCE	18-40
18.12.10	:SPAN <numeric_value>	18-40
18.12.10.1	:HOLD <Boolean>	18-40
18.12.10.2	:LINK CENTer STARt STOP	18-40
18.12.10.3	:FULL	18-40
18.12.11	:STARt <numeric_value>	18-41
18.12.12	:STOP <numeric_value>	18-41
18.13	FUNCTion & DATA Subsystem	18-42
18.13.1	DATA? [<data_handle>]	18-42
18.13.1.1	SENSe <data_handle>s	18-42
18.13.1.2	:PREAmble? [<data_handle>]	18-43
18.13.2	:FUNCTion	18-43
18.13.2.1	:CONCurent <Boolean>	18-43
18.13.2.2	:OFF <sensor_function>{,<sensor_function>}	18-44
18.13.2.2.1	:ALL	18-44
18.13.2.2.2	:COUNT?	18-44
18.13.2.3	[:ON] <sensor_function>{,<sensor_function>}	18-44
18.13.2.3.1	:ALL	18-45
18.13.2.3.2	:COUNT?	18-45
18.13.2.4	:STATe? <sensor_function>	18-45
18.13.2.5	<sensor_function>	18-45
18.13.2.6	<presentation_layer>	18-47
18.13.2.6.1	[XNONE:]	18-47
18.13.2.6.2	XTIME:	18-47
18.13.2.6.3	XFREquency:	18-47
18.13.2.6.4	XPOWER:	18-47
18.13.2.6.5	XVOLtage:	18-47
18.13.2.6.6	XCURrent:	18-47
18.13.2.7	<function_name>	18-48
18.13.2.8	<function>	18-49
18.13.2.8.1	ACCeleration	18-50
18.13.2.8.2	AM	18-50
18.13.2.8.2.1	[:DEPTh]	18-50
18.13.2.8.2.2	:DISTortion	18-50

1999 SCPI Command Reference

18.13.2.8.2.3	:FREQuency	18-51
18.13.2.8.2.4	:SNDRatio	18-51
18.13.2.8.2.5	:SNR	18-51
18.13.2.8.2.6	:THD	18-51
18.13.2.8.3	CONCetration	18-51
18.13.2.8.3.1	:RAW	18-51
18.13.2.8.3.2	:SDEViation	18-51
18.13.2.8.3.3	:TALign	18-51
18.13.2.8.4	CONDition	18-51
18.13.2.8.5	CURRent	18-51
18.13.2.8.5.1	[:DC]	18-51
18.13.2.8.5.2	:AC	18-51
18.13.2.8.6	DISTance	18-52
18.13.2.8.7	FM	18-52
18.13.2.8.7.1	:[DEViation]	18-52
18.13.2.8.7.2	:DISTortion	18-52
18.13.2.8.7.3	:FREQuency	18-52
18.13.2.8.7.4	:SNDRatio	18-52
18.13.2.8.7.5	:SNR	18-52
18.13.2.8.7.6	:THD	18-52
18.13.2.8.8	FERRor	18-52
18.13.2.8.9	FORCe	18-52
18.13.2.8.10	FREQuency	18-52
18.13.2.8.11	FRESistance	18-53
18.13.2.8.12	PERiod	18-53
18.13.2.8.13	PHASe	18-53
18.13.2.8.14	PM	18-53
18.13.2.8.14.1	:[DEViation]	18-53
18.13.2.8.14.2	:DISTortion	18-53
18.13.2.8.14.3	:FREQuency	18-53
18.13.2.8.14.4	:SNDRatio	18-53
18.13.2.8.14.5	:SNR	18-53
18.13.2.8.14.6	:THD	18-53
18.13.2.8.15	POWEr	18-53
18.13.2.8.15.1	:AC	18-54
18.13.2.8.15.2	:ACHannel	18-54
18.13.2.8.15.3	:COHerence	18-54
18.13.2.8.15.4	:CROSSs	18-54
18.13.2.8.15.5	[:DC]	18-54
18.13.2.8.15.6	:DISTortion	18-54
18.13.2.8.15.7	:PSDensity	18-54
18.13.2.8.15.8	:S11l:S12l:S22l:S21	18-54
18.13.2.8.15.9	:SNDRatio	18-55
18.13.2.8.15.10	:SNR	18-55
18.13.2.8.15.11	:THD	18-55

1999 SCPI Command Reference

18.13.2.8.16	PULM	18-55
18.13.2.8.17	RESistance	18-55
18.13.2.8.18	SPEed	18-55
18.13.2.8.18.1	:FRONt	18-55
18.13.2.8.18.2	[:REAR]	18-55
18.13.2.8.19	SSB	18-56
18.13.2.8.20	TEMPerature	18-56
18.13.2.8.21	TIMer	18-56
18.13.2.8.21.1	COUNt	18-56
18.13.2.8.22	TINTerval	18-56
18.13.2.8.23	TOTalize	18-56
18.13.2.8.24	TPLoss	18-56
18.13.2.8.25	VOLTage	18-56
18.13.2.8.25.1	:AC	18-56
18.13.2.8.25.2	:CDFunction	18-56
18.13.2.8.25.3	[:DC]	18-56
18.13.2.8.25.4	:HISTogram	18-56
18.13.2.8.25.5	:PDFunction	18-57
18.14	LIST Subsystem	18-58
18.14.1	:COUNT <numeric_value>	18-58
18.14.2	:DIRection UPIDOWN	18-58
18.14.3	:DWELl <numeric_value>{,<numeric_value>}	18-58
18.14.3.1	:POINTs?	18-58
18.14.4	:FREQuency <numeric_value>{,<numeric_value>}	18-59
18.14.4.1	:POINTs?	18-59
18.14.5	:SEQuence <numeric_value>{,<numeric_value>}	18-59
18.14.5.1	:AUTO <Boolean> ONCE	18-59
18.14.5.2	:POINTs?	18-59
18.15	MIXer Subsystem	18-60
18.15.1	:BIAS <numeric_value>	18-60
18.15.1.1	:AUTO <Boolean> ONCE	18-60
18.15.1.2	:LIMit <numeric_value>	18-60
18.15.2	:HARMonic <numeric_value>	18-60
18.15.2.1	:AUTO <Boolean> ONCE	18-60
18.15.3	:LOSS <numeric_value>	18-61
18.15.3.1	:AUTO <Boolean>	18-61
18.16	PM Subsystem	18-62
18.16.1	[:DEViation]	18-62
18.16.1.1	:RANGe	18-62
18.16.1.1.1	:AUTO <Boolean> ONCE	18-62
18.16.1.1.2	[:UPPer] <numeric_value>	18-62
18.16.1.1.3	:LOWer <numeric_value>	18-62
18.17	POWER Subsystem	18-63
18.17.1	:ACHannel	18-63
18.17.1.1	:SPACing	18-63

1999 SCPI Command Reference

18.17.1.1.1	:LOWer <numeric_value>	18-64
18.17.1.1.1.1	:AUTO <Boolean>	18-64
18.17.1.1.2	[:UPPer] <numeric_value>	18-64
18.17.2	:AC[:DC]	18-64
18.17.2.1	:APERture <numeric_value>	18-64
18.17.2.2	:NPLCycles <numeric_value>	18-64
18.17.2.3	:ATTenuation <numeric_value>	18-64
18.17.2.3.1	:AUTO <Boolean>	18-65
18.17.2.4	:PROTection	18-65
18.17.2.4.1	[:LEVel] <numeric_value>	18-65
18.17.2.4.2	:STATe <Boolean>	18-65
18.17.2.4.3	:TRIPped?	18-65
18.17.2.4.4	:CLEar	18-65
18.17.2.5	:RANGe	18-65
18.17.2.5.1	[:UPPer] <numeric_value>	18-65
18.17.2.5.2	:LOWer <numeric_value>	18-66
18.17.2.5.3	:AUTO <Boolean> ONCE	18-66
18.17.2.5.3.1	:DIRection UP DOWN EITHER	18-66
18.17.2.5.3.2	:LLIMit <numeric_value>	18-66
18.17.2.5.3.3	:ULIMit <numeric_value>	18-66
18.17.2.5.4	:OFFSet <numeric_value>	18-66
18.17.2.5.5	:PTPeak <numeric_value>	18-66
18.17.2.6	:REFerence <numeric_value>	18-67
18.17.2.6.1	:STATe <Boolean>	18-67
18.17.2.7	:RESolution <numeric_value>	18-67
18.17.2.7.1	:AUTO <Boolean> ONCE	18-67
18.17.3	:DETector INTernal EXTernal	18-67
18.18	RESistance FREsistance Subsystem	18-68
18.18.1	:APERture <numeric_value>	18-68
18.18.2	:NPLCycles <numeric_value>	18-68
18.18.3	:OCOMPensated <Boolean>	18-68
18.18.4	:RANGe	18-69
18.18.4.1	[:UPPer] <numeric_value>	18-69
18.18.4.2	:LOWer <numeric_value>	18-69
18.18.4.3	:AUTO <Boolean> ONCE	18-69
18.18.4.3.1	:DIRection UP DOWN EITHER	18-69
18.18.4.3.2	:LLIMit <numeric_value>	18-69
18.18.4.3.3	:ULIMit <numeric_value>	18-69
18.18.5	:REFerence <numeric_value>	18-70
18.18.5.1	:STATe <Boolean>	18-70
18.18.6	:RESolution <numeric_value>	18-70
18.18.6.1	:AUTO <Boolean> ONCE	18-70
18.19	ROSCillator Subsystem	18-71
18.19.1	[:INTernal]	18-71
18.19.1.1	:FREQuency <numeric_value>	18-71

1999 SCPI Command Reference

18.19.2	:EXTeRnal	18-71
18.19.2.1	:FREQuency <numeric_value>	18-71
18.19.3	:SOURce INTernal EXTeRnal NONE CLK10 CLK100	18-71
18.19.3.1	:AUTO <Boolean> ONCE	18-72
18.20	SMOothing Subsystem	18-73
18.20.1	[:STATe] <Boolean>	18-73
18.20.2	:APERTure <numeric_value>	18-73
18.20.3	:POINts <numeric_value>	18-73
18.21	SSB Subsystem	18-74
18.21.1	:TYPE USB LSB A1	18-74
18.22	STABILize Subsystem	18-75
18.22.1	:NTOLerance <numeric_value>	18-75
18.22.2	[:STATE] < Boolean >	18-75
18.22.3	:TIME<n> <numeric_value>	18-76
18.23	SWEEP Subsystem	18-77
18.23.1	:COUNT <numeric_value>	18-78
18.23.2	:DIRection UP DOWN	18-78
18.23.3	:DWELl <numeric_value>	18-78
18.23.3.1	:AUTO <Boolean> ONCE	18-79
18.23.4	:GENeration STEPped ANALog	18-79
18.23.5	:MODE AUTO MANual	18-79
18.23.6	:OFFSet	18-79
18.23.6.1	:POINts <numeric_value>	18-79
18.23.6.2	:TIME <numeric_value>	18-79
18.23.7	:OREFERENCE	18-80
18.23.7.1	:LOCation <numeric_value>	18-80
18.23.7.2	:POINts <numeric_value>	18-80
18.23.8	:POINts <numeric_value>	18-80
18.23.9	:REALtime	18-81
18.23.9.1	[:STATE] <Boolean>	18-81
18.23.10	:SPACing LINear LOGarithmic	18-81
18.23.11	:STEP <numeric_value>	18-82
18.23.12	:TIME <numeric_value>	18-82
18.23.12.1	:AUTO <Boolean> ONCE	18-82
18.23.12.2	:LLIMit <numeric_value>	18-83
18.23.13	:TINTerval <numeric_value>	18-83
18.24	VOLTage Subsystem	18-84
18.24.1	:AC [:DC]	18-84
18.24.1.1	:APERTure <numeric_value>	18-84
18.24.1.2	:NPLCycles <numeric_value>	18-85
18.24.1.3	:ATTenuation <numeric_value>	18-85
18.24.1.3.1	:AUTO <Boolean>	18-85
18.24.1.4	:PROTection	18-85
18.24.1.4.1	[:LEVel] <numeric_value>	18-85
18.24.1.4.2	:STATe <Boolean>	18-85

1999 SCPI Command Reference

18.24.1.4.3	:TRIPped?	18-85
18.24.1.4.4	:CLEar	18-85
18.24.1.5	:RANGe	18-86
18.24.1.5.1	[:UPPer] <numeric_value>	18-86
18.24.1.5.2	:LOWer <numeric_value>	18-86
18.24.1.5.3	:AUTO <Boolean> ONCE	18-86
18.24.1.5.3.1	:DIRection UP DOWN EITHER	18-86
18.24.1.5.3.2	:LLIMit <numeric_value>	18-86
18.24.1.5.3.3	:ULIMit <numeric_value>	18-86
18.24.1.5.4	:OFFSet <numeric_value>	18-87
18.24.1.5.5	:PTPeak <numeric_value>	18-87
18.24.1.6	:REFerence <numeric_value>	18-87
18.24.1.6.1	:STATe <Boolean>	18-87
18.24.1.7	:RESolution <numeric_value>	18-87
18.24.1.7.1	:AUTO <Boolean> ONCE	18-87
18.24.2	:DETector INTernal EXTernal	18-88
18.25	WINDow Subsystem	18-89
18.25.1	[:TYPE] RECTangular UNIFORM FLATtop HAMMING HANNing KBESsell FORCe EXPonential	18-89
18.25.1.1	:KBESsell <numeric_value>	18-89
18.25.1.2	:EXPonential <numeric_value>	18-89
18.25.1.3	:FORCe <numeric_value>	18-89

Chapter 19 SOURce Subsystem

19.1	ACCeleration Subsystem	19-5
19.1.1	[:LEVel] <numeric_value>	19-5
19.2	AM Subsystem	19-6
19.2.1	:COUPLing AC DC GROund	19-6
19.2.2	[:DEPT] <numeric_value>	19-6
19.2.3	:EXTernal	19-6
19.2.3.1	:COUPLing AC DC GROund	19-7
19.2.3.2	:IMPedance <numeric_value>	19-7
19.2.3.3	:POLarity NORMal INVerted	19-7
19.2.4	:INTernal	19-7
19.2.4.1	:FREQuency <numeric_value>	19-7
19.2.5	:MODE	19-7
19.2.6	:POLarity NORMal INVerted	19-8
19.2.7	:SENSitivity <numeric_value>	19-8
19.2.8	:SOURce EXTernal INTernal{,EXTernal,INTernal}	19-8
19.2.9	:STATe <Boolean>	19-8
19.2.10	:TYPE	19-9
19.3	COMBine Subsystem	19-10
19.3.1	:FEED <data_handle>	19-10
19.4	CORRection Subsystem	19-11

1999 SCPI Command Reference

19.4.1	[:STATe] <Boolean>	19-11
19.4.2	:COLLect	19-12
19.4.2.1	[:ACQUIre]	19-12
19.4.2.2	:METHod PMETer	19-12
19.4.2.3	:SAVE [<name>]	19-12
19.4.3	:CSET	19-12
19.4.3.1	[:SElect] <name>	19-12
19.4.3.2	STATe <Boolean>	19-12
19.4.4	:OFFSet	19-13
19.4.4.1	[:MAGNitude] <numeric_value>	19-13
19.4.4.2	:PHASe <numeric_value>	19-13
19.4.4.3	:STATe <Boolean>	19-13
19.4.5	:LOSSl:GAINl:SLOPe	19-13
19.4.5.1	:STATe <Boolean>	19-14
19.4.5.2	[:OUTPut]	19-14
19.4.5.2.1	[:MAGNitude] <numeric_value>	19-14
19.4.5.2.2	:PHASe <numeric_value>	19-14
19.4.6	:EDELay	19-14
19.4.6.1	[:TIME] <numeric_value>	19-14
19.4.6.2	:DISTance <numeric_value>	19-14
19.4.6.3	:STATe <Boolean>	19-15
19.4.7	:RVELocity	19-15
19.4.7.1	:MEDium COAXIWAVEguide	19-15
19.4.7.2	:COAX <numeric_value>	19-15
19.4.7.3	:WAVEguide <numeric_value>	19-15
19.4.7.3.1	:FCUToff <numeric_value>	19-15
19.4.7.4	:STATe <Boolean>	19-15
19.5	CURRent Subsystem	19-17
19.5.1	:ATTenuation <numeric_value>	19-18
19.5.1.1	:AUTO <Boolean>	19-18
19.5.2	:ALC	19-18
19.5.2.1	[:STATe] <Boolean>	19-18
19.5.2.2	:SEARch <Boolean> ONCE	19-18
19.5.2.3	:SOURce INTernallDIODElPMETERlMMHead	19-19
19.5.2.4	:BANDwidthl:BWIDth <numeric_value>	19-19
19.5.2.4.1	:AUTO <Boolean> ONCE	19-19
19.5.3	:CENTer <numeric_value>	19-19
19.5.4	[:LEVel]	19-19
19.5.4.1	[:IMMediate]	19-19
19.5.4.1.1	[:AMPLitude] <numeric_value>	19-20
19.5.4.1.1.1	:AUTO <Boolean> ONCE	19-20
19.5.4.1.2	:OFFSet <numeric_value>	19-20
19.5.4.1.3	:HIGH <numeric_value>	19-20
19.5.4.1.4	:LOW <numeric_value>	19-20
19.5.4.2	:TRIGgered	19-21

1999 SCPI Command Reference

19.5.4.2.1	[:AMPLitude] <numeric_value>	19-21
19.5.4.2.2	:OFFSet <numeric_value>	19-21
19.5.4.2.3	:HIGH <numeric_value>	19-21
19.5.4.2.4	:LOW <numeric_value>	19-21
19.5.5	:LIMit	19-21
19.5.5.1	[:AMPLitude] <numeric_value>	19-21
19.5.5.2	:OFFSet <numeric_value>	19-22
19.5.5.3	:HIGH <numeric_value>	19-22
19.5.5.4	:LOW <numeric_value>	19-22
19.5.5.5	:STATe <Boolean>	19-22
19.5.6	:MANual <numeric_value>	19-22
19.5.7	:MODE FIXed SWEEP LIST	19-22
19.5.8	:PROTection	19-23
19.5.8.1	[:LEVel] <numeric_value>	19-23
19.5.8.2	:STATe <Boolean>	19-23
19.5.8.3	:TRIPped?	19-23
19.5.8.4	:CLEar	19-23
19.5.9	:RANGe <numeric_value>	19-23
19.5.9.1	:AUTO <Boolean> ONCE	19-23
19.5.10	:REFeRence <numeric_value>	19-23
19.5.10.1	:STATe <Boolean>	19-24
19.5.11	:SLEW <numeric_value>	19-24
19.5.12	:SPAN <numeric_value>	19-24
19.5.12.1	:FULL	19-24
19.5.12.2	:HOLD <Boolean>	19-24
19.5.12.3	:LINK CENTer START STOP	19-24
19.5.13	:STARt <numeric_value>	19-24
19.5.14	:STOP <numeric_value>	19-25
19.6	DM Subsystem	19-26
19.6.1	:FORMat <modulation format>	19-27
19.6.2	:STATe <Boolean>	19-27
19.6.3	:SOURce EXTernal PRBS CALibrate	19-27
19.6.4	:FILTer	19-27
19.6.4.1	[:SOURce] INTernal EXTernal	19-27
19.6.4.2	:ICORrection <numeric_value>	19-27
19.6.4.3	:QCORrection <numeric_value>	19-27
19.6.5	:IQRatio	19-28
19.6.5.1	:STATe <Boolean>	19-28
19.6.5.2	[:MAGNitude] <numeric_value>	19-28
19.6.6	:LEAKage	19-28
19.6.6.1	:STATe <Boolean>	19-28
19.6.6.2	[:MAGNitude] <numeric_value>	19-28
19.6.6.3	:ANGLE <numeric_value>	19-28
19.6.7	:QUADrature	19-28
19.6.7.1	:STATe <Boolean>	19-29

1999 SCPI Command Reference

19.6.7.2	:ANGLE <numeric_value>	19-29
19.6.8	:COUPling	19-29
19.6.8.1	[:ALL] ACIDClGROund	19-29
19.6.8.2	:DATA ACIDClGROund	19-29
19.6.8.3	:CLOCK ACIDClGROund	19-29
19.6.9	:THReshold	19-29
19.6.9.1	[:ALL] <numeric_value>	19-29
19.6.9.2	:DATA <numeric_value>	19-29
19.6.9.3	:CLOCK <numeric_value>	19-29
19.6.10	:DMODE SERIalPARallel	19-30
19.6.11	:FRAMe	19-30
19.6.11.1	:SOURce INTernallEXTernal	19-30
19.6.12	:POLarity [:ALL] NORMallINVerted	19-30
19.6.12.1	:I<n> NORMallINVerted	19-30
19.6.12.2	:Q<n> NORMallINVerted	19-30
19.6.12.3	:ICLock NORMallINVerted	19-30
19.6.12.4	:QCLock NORMallINVerted	19-30
19.6.13	:CLOCK	19-30
19.6.13.1	:SOURce NONEINTernallEXTernal	19-31
19.7	FM Subsystem	19-32
19.7.1	:COUPling ACIDClGROund	19-32
19.7.2	[:DEViation] <numeric_value>	19-32
19.7.3	:EXTernal	19-32
19.7.3.1	:COUPling ACIDClGROund	19-33
19.7.3.2	:IMPedance <numeric_value>	19-33
19.7.3.3	:POLarity NORMallINVerted	19-33
19.7.4	:INTernal	19-33
19.7.4.1	:FREQuency <numeric_value>	19-33
19.7.5	:MODE LOCKedUNLocked	19-33
19.7.6	:POLarity NORMallINVerted	19-34
19.7.7	:SENSitivity <numeric_value>	19-34
19.7.8	:SOURce EXTernalINTernal{,EXTernal,INTernal}	19-34
19.7.9	:STATe <Boolean>	19-34
19.8	FORCe Subsystem	19-35
19.8.1	:CDOWn	19-35
19.8.1.1	:INITiate	19-36
19.8.1.2	:SOFFset <numeric_value>	19-36
19.8.1.3	:NRUNs <numeric_value>	19-37
19.8.1.4	:RLDerivation	19-37
19.8.1.4.1	:FACceptance <numeric_value>	19-37
19.8.1.4.2	:INITiate	19-37
19.8.1.4.3	:RMAXimum <numeric_value>	19-38
19.8.1.4.4	:RVERify <numeric_value>	19-38
19.8.2	:CONFigure	19-39
19.8.2.1	:ABRAke	19-39

1999 SCPI Command Reference

19.8.2.1.1	:GAIN <numeric_value>	19-39
19.8.2.1.2	[:STATe] <Boolean>	19-39
19.8.2.1.3	:THReshold <numeric_value>	19-39
19.8.2.2	:GRADE	19-39
19.8.2.2.1	:LEVel <numeric_value>	19-39
19.8.2.2.2	:SOURce <INTernallEXTernal>	19-39
19.8.2.2.3	[:STATe] <Boolean>	19-40
19.8.2.3	[:VEHicle]	19-40
19.8.2.3.1	:DCOefficient <numeric_value>,<numeric_value>,<numeric_value> [,<numeric_value>]	19-40
19.8.2.3.2	:DINertia <numeric_value>	19-40
19.8.2.3.3	[:STATe] <Boolean>	19-40
19.8.2.3.4	:TCOefficient <numeric_value>,<numeric_value>,<numeric_value> [,<numeric_value>]	19-41
19.8.2.3.5	:TINertia <numeric_value>	19-41
19.8.2.3.6	:WEIGHt <numeric_value>	19-41
19.8.3	:INITiate	19-41
19.8.4	[:LEVel] <numeric_value>	19-42
19.8.5	:RLSimulation	19-42
19.8.5.1	:INITiate	19-42
19.9	FREQuency Subsystem	19-43
19.9.1	:CENTer <numeric_value>	19-43
19.9.2	[:CW :FIXed] <numeric_value>	19-43
19.9.2.1	:AUTO <Boolean> ONCE	19-43
19.9.3	:MANual <numeric_value>	19-44
19.9.4	:MODE CW FIXed SWEep LIST SENSe	19-44
19.9.5	:MULTiplier <numeric_value>	19-44
19.9.6	:OFFSet <numeric_value>	19-44
19.9.7	:RESolution <numeric_value>	19-45
19.9.7.1	AUTO <Boolean> ONCE	19-45
19.9.8	:SPAN <numeric_value>	19-45
19.9.8.1	:FULL	19-45
19.9.8.2	:HOLD <Boolean>	19-45
19.9.8.3	:LINK CENTer START STOP	19-45
19.9.9	:START <numeric_value>	19-46
19.9.10	:STOP <numeric_value>	19-46
19.10	FUNCTION Subsystem	19-47
19.10.1	[:SHAPE] <source_shape>	19-47
19.10.2	:MODE <source_mode>	19-47
19.11	LIST Subsystem	19-48
19.11.1	:AM	19-49
19.11.1.1	:DEPTH <numeric_value>{,<numeric_value>}	19-49
19.11.1.1.1	:POINTs?	19-49
19.11.2	:APRobe <numeric_list>{,<numeric_list>}	19-49
19.11.2.1	:POINTs?	19-49

1999 SCPI Command Reference

19.11.3	:CONCurrent <numeric_value>{,<numeric_value>}	19-49
19.11.3.1	:AUTO <Boolean> ONCE	19-50
19.11.3.2	:POINTs?	19-50
19.11.4	:CONTRol	19-50
19.11.4.1	:APOWer <Boolean>{,<Boolean>}	19-50
19.11.4.1.1	:POINTs?	19-50
19.11.4.2	:BLOWer <Boolean>{,<Boolean>}	19-50
19.11.4.2.1	:POINTs?	19-50
19.11.4.3	:COMPressor <Boolean>{,<Boolean>}	19-50
19.11.4.3.1	:POINTs?	19-50
19.11.5	:COUNt <numeric_value>	19-50
19.11.6	:CURRent <numeric_value>{,<numeric_value>}	19-50
19.11.6.1	:POINTs?	19-50
19.11.7	:DIRection UPIDOWN	19-51
19.11.8	:DWELl <numeric_value>{,<numeric_value>}	19-51
19.11.8.1	:POINTs?	19-51
19.11.9	:FREQuency <numeric_value>{,<numeric_value>}	19-51
19.11.9.1	:POINTs?	19-51
19.11.10	:GENeration DSEQuence SEQuence DCONcurrent CONCurrent	19-51
19.11.11	:PULM	19-51
19.11.11.1	:STATe <Boolean>{,<Boolean>}	19-51
19.11.11.1.1	:POINTs?	19-51
19.11.12	:POWer <numeric_value>{,<numeric_value>}	19-52
19.11.12.1	:POINTs?	19-52
19.11.13	:RESistance <numeric_value>{,<numeric_value>}	19-52
19.11.13.1	:POINTs?	19-52
19.11.14	:RTIME <numeric_value>{,<numeric_value>}	19-52
19.11.14.1	:POINTs?	19-52
19.11.15	:SEQuence <numeric_value>{,<numeric_value>}	19-52
19.11.15.1	:AUTO <Boolean> ONCE	19-52
19.11.15.2	:POINTs?	19-52
19.11.16	:TEMPerature <numeric_value>{,<numeric_value>}	19-52
19.11.16.1	:POINTs?	19-52
19.11.17	:VOLTage <numeric_value>{,<numeric_value>}	19-53
19.11.17.1	:POINTs?	19-53
19.12	MARKer Subsystem	19-54
19.12.1	:AMPLitude <Boolean>	19-54
19.12.2	:AOFF	19-54
19.12.3	:FREQuency <numeric_value>	19-54
19.12.4	:MODE FREQuency POSition DELTA	19-54
19.12.5	:POINT <numeric_value>	19-54
19.12.6	:REFerence <numeric_value>	19-55
19.12.7	[[:STATe]] <Boolean>	19-55
19.13	PHASe Subsystem	19-56
19.13.1	[[:ADJust]] <numeric_value>	19-56

1999 SCPI Command Reference

19.13.1.1	:STEP <numeric_value>	19-56
19.13.2	:SOURce INTernal EXTernal	19-56
19.13.3	:REFerence	19-56
19.14	PM Subsystem	19-57
19.14.1	[:DEViation] <numeric_value>	19-57
19.14.2	:SENSitivity <numeric_value>	19-57
19.14.3	:MODE LOCKed UNLocked	19-57
19.14.4	:STATe <Boolean>	19-58
19.14.5	:SOURce EXTernal INTernal{,EXTernal,INTernal}	19-58
19.14.6	:COUPling AC DC GROund	19-58
19.14.7	:POLarity NORMal INVerted	19-58
19.14.8	:INTernal	19-59
19.14.8.1	:FREQuency <numeric_value>	19-59
19.14.9	:EXTernal	19-59
19.14.9.1	:IMPedance <numeric_value>	19-59
19.14.9.2	:COUPling AC DC GROund	19-59
19.14.9.3	:POLarity NORMal INVerted	19-59
19.15	POWer Subsystem	19-60
19.15.1	:ATTenuation <numeric_value>	19-61
19.15.1.1	:AUTO <Boolean>	19-61
19.15.2	:ALC	19-61
19.15.2.1	[:STATe] <Boolean>	19-61
19.15.2.2	:SEARch <Boolean> ONCE	19-61
19.15.2.3	:SOURce INTernal DIODE PMETER MMHead	19-62
19.15.2.4	:BANDwidth :BWIDth <numeric_value>	19-62
19.15.2.4.1	:AUTO <Boolean> ONCE	19-62
19.15.3	:CENTer <numeric_value>	19-62
19.15.4	[:LEVel]	19-62
19.15.4.1	[:IMMEDIATE]	19-62
19.15.4.1.1	[:AMPLitude] <numeric_value>	19-63
19.15.4.1.2	:OFFSet <numeric_value>	19-63
19.15.4.1.3	:HIGH <numeric_value>	19-63
19.15.4.1.4	:LOW <numeric_value>	19-63
19.15.4.2	:TRIGgered	19-63
19.15.4.2.1	[:AMPLitude] <numeric_value>	19-64
19.15.4.2.2	:OFFSet <numeric_value>	19-64
19.15.4.2.3	:HIGH <numeric_value>	19-64
19.15.4.2.4	:LOW <numeric_value>	19-64
19.15.5	:LIMit	19-64
19.15.5.1	[:AMPLitude] <numeric_value>	19-64
19.15.5.2	:OFFSet <numeric_value>	19-64
19.15.5.3	:HIGH <numeric_value>	19-65
19.15.5.4	:LOW <numeric_value>	19-65
19.15.5.5	:STATe <Boolean>	19-65
19.15.6	:MANual <numeric_value>	19-65

1999 SCPI Command Reference

19.15.7	:MODE FIXed SWEep LIST	19-65
19.15.8	:PROTection	19-65
19.15.8.1	[:LEVel] <numeric_value>	19-65
19.15.8.2	:STATe <Boolean>	19-66
19.15.8.3	:TRIPped?	19-66
19.15.8.4	:CLEar	19-66
19.15.9	:RANGe <numeric_value>	19-66
19.15.9.1	:AUTO <Boolean> ONCE	19-66
19.15.10	:REFerence <numeric_value>	19-66
19.15.10.1	:STATe <Boolean>	19-66
19.15.11	:SLEW <numeric_value>	19-66
19.15.12	:SPAN <numeric_value>	19-67
19.15.12.1	:HOLD <Boolean>	19-67
19.15.12.2	:LINK CENTer START STOP	19-67
19.15.12.3	:FULL	19-67
19.15.13	:START <numeric_value>	19-67
19.15.14	:STOP <numeric_value>	19-67
19.16	PULSe Modulation Subsystem	19-68
19.16.1	:EXTeRnal	19-68
19.16.1.1	:HYSTeresis <numeric_value>	19-68
19.16.1.2	:IMPedance <numeric_value>	19-68
19.16.1.3	:LEVel <numeric_value>	19-68
19.16.1.4	:POLarity NORMal INVerted	19-69
19.16.2	:INTeRnal	19-69
19.16.2.1	:FREQuency <numeric_value>	19-69
19.16.3	MODE	19-69
19.16.4	:POLarity NORMal INVerted	19-69
19.16.5	:SOURce EXTeRnal INTeRnal{,EXTeRnal,INTeRnal}	19-70
19.16.6	:STATe <Boolean>	19-70
19.17	PULSe Subsystem	19-71
19.17.1	:PERiod <numeric_value>	19-71
19.17.2	:WIDTh <numeric_value>	19-71
19.17.3	:DCYClE <numeric_value>	19-71
19.17.4	:HOLD WIDTh DCYClE	19-71
19.17.5	:DELay <numeric_value>	19-72
19.17.6	:DOUBle	19-72
19.17.6.1	[:STATE] <Boolean>	19-72
19.17.6.2	:DELay <numeric_value>	19-72
19.17.7	:TRANsition	19-72
19.17.7.1	:STATe <Boolean>	19-72
19.17.7.2	[:LEADing] <numeric_value>	19-72
19.17.7.3	:TRAiling <numeric_value>	19-73
19.17.7.3.1	:AUTO <Boolean> ONCE	19-73
19.17.8	:COUNT <numeric_value>	19-73
19.17.9	:POLarity NORMal COMPlEment INVerted	19-73

1999 SCPI Command Reference

19.18	RESistance Subsystem	19-74
19.18.1	[:LEVel]	19-75
19.18.1.1	[:IMMediate]	19-75
19.18.1.1.1	[:AMPLitude] <numeric_value>	19-75
19.18.1.1.2	:OFFSet <numeric_value>	19-75
19.18.1.1.3	:HIGH <numeric_value>	19-75
19.18.1.1.4	:LOW <numeric_value>	19-75
19.18.1.2	:TRIGgered	19-75
19.18.1.2.1	[:AMPLitude] <numeric_value>	19-76
19.18.1.2.2	:OFFSet <numeric_value>	19-76
19.18.1.2.3	:HIGH <numeric_value>	19-76
19.18.1.2.4	:LOW <numeric_value>	19-76
19.18.2	:LIMit	19-76
19.18.2.1	[:AMPLitude] <numeric_value>	19-76
19.18.2.2	:OFFSet <numeric_value>	19-76
19.18.2.3	:HIGH <numeric_value>	19-76
19.18.2.4	:LOW <numeric_value>	19-77
19.18.3	:PROTection	19-77
19.18.3.1	[:LEVel] <numeric_value>	19-77
19.18.3.2	:STATe <Boolean>	19-77
19.18.3.3	:TRIPped?	19-77
19.18.3.4	:CLEar	19-77
19.18.4	:SLEW <numeric_value>	19-77
19.18.5	:CENTer <numeric_value>	19-77
19.18.6	:SPAN <numeric_value>	19-77
19.18.6.1	:HOLD <Boolean>	19-78
19.18.6.2	:LINK CENTer START STOP	19-78
19.18.6.3	:FULL	19-78
19.18.7	:START <numeric_value>	19-78
19.18.8	:STOP <numeric_value>	19-78
19.18.9	:MANual <numeric_value>	19-78
19.18.10	:MODE FIXEd SWEep LIST	19-79
19.18.11	:REFerence <numeric_value>	19-79
19.18.11.1	:STATe <Boolean>	19-79
19.18.12	:RANGe <numeric_value>	19-79
19.18.12.1	:AUTO <Boolean> ONCE	19-79
19.19	ROSCillator Subsystem	19-80
19.19.1	[:INTernal]	19-80
19.19.1.1	:FREQuency <numeric_value>	19-80
19.19.2	:EXTernal	19-80
19.19.2.1	:FREQuency <numeric_value>	19-80
19.19.3	:SOURce INTernal EXTernal NONE	19-80
19.19.3.1	:AUTO <Boolean> ONCE	19-80
19.20	SPEed Subsystem	19-82
19.20.1	:INITiate	19-82

1999 SCPI Command Reference

19.20.2	:LEVel] <numeric_value>	19-82
19.20.3	:SSDLoss	19-82
19.20.3.1	:INITiate	19-83
19.20.3.2	:LATime <numeric_value>	19-83
19.20.3.3	:STime <numeric_value>	19-84
19.21	SWEEP Subsystem	19-85
19.21.1	:TIME <numeric_value>	19-85
19.21.1.1	:AUTO <Boolean> ONCE	19-85
19.21.1.2	:LLIMit <numeric_value>	19-85
19.21.2	:DWELL <numeric_value>	19-85
19.21.2.1	:AUTO <Boolean> ONCE	19-86
19.21.3	:DIRection UP DOWN	19-86
19.21.4	:MODE AUTO MANual	19-86
19.21.5	:SPACing LINear LOGarithmic	19-86
19.21.6	:GENeration STEPped ANALog	19-86
19.21.7	:STEP <numeric_value>	19-87
19.21.8	:POINTs <numeric_value>	19-87
19.21.9	:COUNT <numeric_value>	19-87
19.22	TEMPeratureSubsystem	19-88
19.22.1	:APRobe <numeric_list>	19-88
19.22.2	:DWELL <numeric_value>	19-88
19.22.3	:LCONstants	19-89
19.22.3.1	:DERivative <numeric_value>	19-89
19.22.3.2	:GAIN] <numeric_value>	19-89
19.22.3.3	:INTEgral <numeric_value>	19-89
19.22.4	:MODE FIXEd LIST PROGram	19-90
19.22.5	:PROTection	19-90
19.22.5.1	:HIGH]	19-90
19.22.5.1.1	:CLEar	19-90
19.22.5.1.2	:LEVel] <numeric_value>	19-90
19.22.5.1.3	:STATe <Boolean>	19-91
19.22.5.1.4	:TOUT <numeric_value>	19-91
19.22.5.1.5	:TRIPped?	19-91
19.22.5.2	:LOW	19-91
19.22.5.2.1	:CLEar	19-91
19.22.5.2.2	:LEVel] <numeric_value>	19-91
19.22.5.2.3	:STATe <Boolean>	19-91
19.22.5.2.4	:TOUT <numeric_value>	19-92
19.22.5.2.5	:TRIPped?	19-92
19.22.6	:RTIME <numeric_value>	19-92
19.22.7	:SPOint] <numeric_value>	19-92
19.23	VOLTage Subsystem	19-93
19.23.1	:ATTenuation <numeric_value>	19-94
19.23.1.1	:AUTO <Boolean>	19-94
19.23.2	:ALC	19-94

1999 SCPI Command Reference

19.23.2.1	:STATe <Boolean>	19-94
19.23.2.2	:SEARCh <Boolean> ONCE	19-94
19.23.2.3	:SOURce INTernallDIODElPMETerlMMHead	19-95
19.23.2.4	:BANDwidthl:BWIDth <numeric_value>	19-95
19.23.2.4.1	:AUTO <Boolean> ONCE	19-95
19.23.3	:CENTer <numeric_value>	19-95
19.23.4	:LEVel]	19-95
19.23.4.1	:IMMediate]	19-95
19.23.4.1.1	:AMPLitude] <numeric_value>	19-96
19.23.4.1.1.1	:AUTO <Boolean> ONCE	19-96
19.23.4.1.2	:OFFSet <numeric_value>	19-96
19.23.4.1.3	:HIGH <numeric_value>	19-96
19.23.4.1.4	:LOW <numeric_value>	19-96
19.23.4.2	:TRIGgered	19-97
19.23.4.2.1	:AMPLitude] <numeric_value>	19-97
19.23.4.2.2	:OFFSet <numeric_value>	19-97
19.23.4.2.3	:HIGH <numeric_value>	19-97
19.23.4.2.4	:LOW <numeric_value>	19-97
19.23.5	:LIMit	19-97
19.23.5.1	:AMPLitude] <numeric_value>	19-97
19.23.5.2	:OFFSet <numeric_value>	19-98
19.23.5.3	:HIGH <numeric_value>	19-98
19.23.5.4	:LOW <numeric_value>	19-98
19.23.5.5	:STATe <Boolean>	19-98
19.23.6	:MANual <numeric_value>	19-98
19.23.7	:MODE FIXedlSWEePlLIST	19-98
19.23.8	:PROTection	19-99
19.23.8.1	:LEVel] <numeric_value>	19-99
19.23.8.2	:STATe <Boolean>	19-99
19.23.8.3	:TRIPped?	19-99
19.23.8.4	:CLEar	19-99
19.23.9	:RANGe <numeric_value>	19-99
19.23.9.1	:AUTO <Boolean> ONCE	19-99
19.23.10	:REFerence <numeric_value>	19-99
19.23.10.1	:STATe <Boolean>	19-100
19.23.11	:SLEW <numeric_value>	19-100
19.23.12	:SPAN <numeric_value>	19-100
19.23.12.1	:HOLD <Boolean>	19-100
19.23.12.2	:LINK CENTerlSTARTlSTOP	19-100
19.23.12.3	:FULL	19-100
19.23.13	:START <numeric_value>	19-100
19.23.14	:STOP <numeric_value>	19-101

Chapter 20 STATus Subsystem

20.1	:OPERation	20-3
20.1.1	:BIT<n>	20-3
20.1.2	:CONDition?	20-3
20.1.3	:ENABle <NRf> <non-decimal numeric>	20-3
20.1.4	[:EVENTt]?	20-4
20.1.5	:MAP <NRf>,<NRf>	20-4
20.1.6	:NTRansition <NRf> <non-decimal numeric>	20-4
20.1.7	:PTRansition <NRf> <non-decimal numeric>	20-4
20.2	:PRESet	20-4
20.3	:QUEStionable	20-7
20.3.1	:BIT<n>	20-7
20.3.2	:CONDition?	20-7
20.3.3	:ENABle <NRf> <non-decimal numeric>	20-7
20.3.4	[:EVENTt]?	20-7
20.3.5	:MAP <NRf>,<NRf>	20-7
20.3.6	:NTRansition <NRf> <non-decimal numeric>	20-7
20.3.7	:PTRansition <NRf> <non-decimal numeric>	20-7

Chapter 21 SYSTem Subsystem

21.1	:ALTErnate <numeric_value>	21-3
21.1.1	:STATe <Boolean>	21-3
21.2	:BEEPPer	21-3
21.2.1	:FREQuency <numeric_value>	21-3
21.2.2	[:IMMediate][<frequency>[,<time>[,<volume>]]]	21-4
21.2.3	:STATe <Boolean>	21-4
21.2.4	:TIME <numeric_value>	21-4
21.2.5	:VOLume <numeric_value>	21-4
21.3	:CAPability?	21-4
21.4	:COMMunicate	21-4
21.4.1	:CENTronics	21-4
21.4.1.1	:FEED <data_handle>	21-5
21.4.2	:GPIB	21-5
21.4.2.1	:RDEvice	21-5
21.4.2.1.1	:ADDReSS <numeric_value>[,<numeric_value>]	21-5
21.4.2.1.2	:FEED <data_handle>	21-5
21.4.2.2	[:SELF]	21-5
21.4.2.2.1	:ADDReSS <numeric_value>[,<numeric_value>]	21-5
21.4.3	:SERial	21-6
21.4.3.1	:CONTrol	21-6
21.4.3.1.1	:DTR ON OFF STANdard IBFull	21-6
21.4.3.1.2	:RTS ON OFF STANdard IBFull RFR	21-6
21.4.3.2	:FEED <data_handle>	21-7
21.4.3.3	[:RECeive]	21-7

1999 SCPI Command Reference

21.4.3.3.1	:BAUD <numeric_value>	21-7
21.4.3.3.2	:BITS <numeric_value>	21-7
21.4.3.3.3	:PACE XON ACK NONE	21-7
21.4.3.3.3.1	:THReshold	21-8
21.4.3.3.4	:PARity	21-8
21.4.3.3.4.1	:CHECk <Boolean>	21-8
21.4.3.3.4.2	[:TYPE] EVEN ODD ZERO ONE NONE IGNore	21-8
21.4.3.3.5	:SBITs <numeric_value>	21-8
21.4.3.4	:TRANsmit	21-8
21.4.3.4.1	:AUTO <Boolean>	21-9
21.4.3.4.2	:BAUD <numeric_value>	21-9
21.4.3.4.3	:BITS <numeric_value>	21-9
21.4.3.4.4	:DELay <numeric_value>	21-9
21.4.3.4.5	:PACE XON ACK NONE	21-9
21.4.3.4.6	:PARity	21-9
21.4.3.4.6.1	[:TYPE] EVEN ODD ZERO ONE NONE	21-9
21.4.3.4.7	:SBITs <numeric_value>	21-9
21.4.4	:SOCKet <n>	21-10
21.4.4.1	:ADDRess <string>	21-10
21.4.4.2	:CONNect	21-10
21.4.4.3	:DISConnect	21-10
21.4.4.4	:FEED <n> <data_handle>{,<data_handle>}	21-10
21.4.4.4.1	:OCONdition <event_handle>	21-10
21.4.4.4.2	:SCONdition <event_handle>	21-10
21.4.4.5	:LISTen	21-10
21.4.4.6	:PORT <numeric_value>	21-10
21.4.4.7	:TYPE TCPIUDP	21-11
21.5	:CPON <card_destination> ALL	21-11
21.6	:CTYPE? <card_destination>	21-11
21.7	:DATE <year>,<month>,<day>	21-11
21.8	:ERRor Subsystem	21-11
21.8.1	The Error/Event Queue	21-13
21.8.2	Error/Event numbers	21-14
21.8.3	No Error	21-14
21.8.4	ALL?	21-14
21.8.5	CODE	21-14
21.8.5.1	ALL?	21-14
21.8.5.2	[NEXT]?	21-14
21.8.6	COUNT?	21-14
21.8.7	:ENABLe	21-15
21.8.7.1	:ADD <numeric list>	21-15
21.8.7.2	:DELete <numeric list>	21-15
21.8.7.3	[:LIST] <numeric list>	21-15
21.8.8	[NEXT]?	21-15
21.8.9	Command Error	21-15

1999 SCPI Command Reference

21.8.10	Execution Error	21-19
21.8.11	Device-Specific Error	21-25
21.8.12	Query Error	21-26
21.8.13	Power On Event	21-27
21.8.14	User Request Event	21-27
21.8.15	Request Control Event	21-28
21.8.16	Operation Complete Event	21-28
21.9	:HELP	21-28
21.9.1	:HEADers?	21-28
21.9.2	:SYNTax? <command_header>	21-31
21.10	:KEY <numeric_value>	21-32
21.10.1	:CATalog	21-32
21.10.2	:DEFine <numeric_value>,<block>[,<string>]	21-32
21.10.3	:DELete <numeric_value>	21-32
21.11	:KLOCK <Boolean>	21-33
21.12	:LANGuage <string>	21-33
21.13	:LFRequency<numeric_value>	21-33
21.13.1	:AUTO <Boolean> ONCE	21-33
21.14	:LOCK	21-34
21.14.1	:OWNer?	21-34
21.14.2	:RELease	21-34
21.14.3	:REQuest?	21-34
21.15	:PASSword	21-35
21.15.1	:CDISable <password>	21-35
21.15.2	[:CENable] <password>	21-35
21.15.2.1	:STATe?	21-35
21.15.3	:NEW <current password>,<new password>	21-35
21.16	:PRESet	21-35
21.17	:SECurity	21-36
21.17.1	:IMMediate	21-36
21.17.2	[:STATe] <Boolean>	21-36
21.18	:SET <block data>	21-36
21.19	:TIME <hour>,<minute>,<second>	21-37
21.19.1	:TIMer	21-38
21.19.1.1	:COUNT <numeric_value>	21-38
21.19.1.2	[:STATe] <Boolean>	21-38
21.20	:TZONe <hour> [,<minute>]	21-38
21.21	:VERSion?	21-38

Chapter 22 TEST Subsystem

Chapter 23 TRACe | DATA

23.1	:CATalog?	23-2
23.2	:COPY <trace_name>,(<trace_name> <data_handle>)	23-2

1999 SCPI Command Reference

23.3	[:DATA] <trace_name>,(<block> <dif_expression> (<numeric_value>{,<numeric_value>}))	23-2
23.3.1	:LINE <trace_name>,<numeric_value>,<numeric_value>, <numeric_value>,<numeric_value>	23-3
23.3.2	:PREAmble? <trace_name>	23-3
23.3.3	:VALue <trace_name>,<numeric_value>,<numeric_value>	23-3
23.4	:DEFine <trace_name>[,<numeric_value> <trace_name>)]	23-3
23.5	:DELete	23-4
23.5.1	[:NAME] <trace_name>	23-4
23.5.2	:ALL	23-4
23.6	:FEED <trace_name>,(<data_handle> NONE)	23-4
23.6.1	:CONTRol <trace_name>, ALWays OCONdition NEXT NEVer	23-4
23.6.2	:OCONdition <trace_name>,<condition_expr>	23-5
23.7	:FREE?	23-5
23.8	:POINTs <trace_name>[,<numeric_value>]	23-5
23.8.1	:AUTO <trace_name>,<Boolean> ONCE)	23-5

Chapter 24 TRIGger Subsystem

24.1	ARM-TRIGger Model	24-1
24.2	Model Layers	24-2
24.2.1	IDLE State	24-2
24.2.2	Initiated	24-2
24.2.3	Event Detection Layer	24-3
24.3	Sequence Event Use	24-4
24.4	Expanded Capability Trigger Model	24-5
24.4.1	LAYer Nomenclature	24-6
24.4.2	Standard SEQUences	24-7
24.4.3	Subservient Sequences	24-7
24.5	ABORt	24-12
24.6	ARM	24-12
24.6.1	[:SEQUence]	24-12
24.6.1.1	:DEFine <sequence_name>	24-12
24.6.1.1.1	MGRules <Boolean>	24-13
24.6.1.2	[:LAYer]	24-14
24.6.1.2.1	:COUNT <numeric_value>	24-14
24.6.1.2.2	:COUPLing ACIDC	24-14
24.6.1.2.3	:DELay <numeric_value>	24-14
24.6.1.2.3.1	:AUTO <Boolean> ONCE	24-14
24.6.1.2.4	:ECL	24-15
24.6.1.2.5	:ECOUNT <numeric_value>	24-15
24.6.1.2.6	:FILTer	24-15
24.6.1.2.6.1	:HPASs	24-15
24.6.1.2.6.2	[:LPASs]	24-15
24.6.1.2.7	:HYSTEResis <numeric_value>	24-16

1999 SCPI Command Reference

24.6.1.2.8	[:IMMediate]	24-16
24.6.1.2.9	:LEVel <numeric_value>	24-16
24.6.1.2.9.1	:AUTO <Boolean> ONCE	24-16
24.6.1.2.10	:LINK <event_handle>	24-17
24.6.1.2.11	PROToCol	24-17
24.6.1.2.11.1	VXI SYNChronous SSYNchronous ASYNchronous	24-17
24.6.1.2.12	:SIGNal	24-17
24.6.1.2.13	:SLOPe POSitive NEGative EITHER	24-18
24.6.1.2.14	:SOURce <parameter>	24-18
24.6.1.2.15	:TIMer <numeric_value>	24-19
24.6.1.2.16	:TTL	24-19
24.6.1.2.17	:TYPE EDGE VIDEo	24-19
24.6.1.2.18	:VIDEo	24-19
24.6.1.2.18.1	:FIEld	24-19
24.6.1.2.18.2	:FORMat	24-20
24.6.1.2.18.3	:LINE	24-20
24.6.1.2.18.4	:SSIGNal	24-20
24.7	INITiate	24-21
24.7.1	:CONTinuous <Boolean>	24-21
24.7.1.1	[:ALL] <Boolean>	24-22
24.7.1.2	:NAME <sequence_name>, <Boolean>	24-22
24.7.1.3	:SEQuence <Boolean>	24-22
24.7.2	[:IMMediate]	24-22
24.7.2.1	[:ALL]	24-22
24.7.2.2	:NAME <sequence_name>	24-23
24.7.2.3	:SEQuence	24-23
24.7.3	:POFLag INCLude EXCLude	24-23
24.8	TRIGger	24-23
24.8.1	[:SEQuence]	24-23
24.8.1.1	:ATRigger	24-23
24.8.1.1.1	[:STATe] <Boolean>	24-24
24.8.1.2	:COUNt <numeric_value>	24-24
24.8.1.3	:COUPling ACIDC	24-24
24.8.1.4	:DEFine <sequence_name>	24-24
24.8.1.4.1	MGRules <Boolean>	24-25
24.8.1.5	:DELay <numeric_value>	24-25
24.8.1.5.1	:AUTO <Boolean> ONCE	24-26
24.8.1.6	:ECL	24-26
24.8.1.7	:ECOUNt <numeric_value>	24-26
24.8.1.8	:FILTer	24-26
24.8.1.8.1	:HPASs	24-26
24.8.1.8.1.1	:FREQuency <numeric_value>	24-26
24.8.1.8.1.2	[:STATe] <Boolean>	24-26
24.8.1.8.2	[:LPASs]	24-26
24.8.1.8.2.1	:FREQuency <numeric_value>	24-27

1999 SCPI Command Reference

24.8.1.8.2.2	:STATe] <Boolean>	24-27
24.8.1.9	:HOLDoff <numeric_value>	24-27
24.8.1.10	:HYSTeresis <numeric_value>	24-27
24.8.1.11	:IMMediate]	24-27
24.8.1.12	:LEVel <numeric_value>	24-28
24.8.1.12.1	:AUTO <Boolean> ONCE	24-28
24.8.1.13	:LINK <event_handle>	24-28
24.8.1.14	:PROTOcol	24-28
24.8.1.14.1	:VXI SYNChronous SSYNchronous ASYNchronous	24-29
24.8.1.15	:SIGNal	24-29
24.8.1.16	:SLOPe POSitive NEGative EITHer	24-29
24.8.1.17	:SOURce <parameter>	24-29
24.8.1.18	:TIMer <numeric_value>	24-30
24.8.1.19	:TTL	24-30
24.8.1.20	:TYPE EDGE VIDEo	24-31
24.8.1.21	:VIDeo	24-31
24.8.1.21.1	:FIELd	24-31
24.8.1.21.1.1	: NUMBer] <numeric_value>	24-31
24.8.1.21.1.2	: SElect ODD EVEN ALL NUMBer	24-31
24.8.1.21.2	:FORMat	24-31
24.8.1.21.2.1	: LPFRame <numeric_variable>	24-31
24.8.1.21.3	:LINE	24-32
24.8.1.21.3.1	: NUMBer] <numeric_value>	24-32
24.8.1.21.3.2	: SElect ALL NUMBer	24-32
24.8.1.21.4	:SSIGNAL	24-32
24.8.1.21.4.1	: POLarity POSitive NEGative	24-32

Chapter 25 UNIT Subsystem

25.1	:ANGLE DEGI RAD	25-1
25.2	:CURRent, :POWER, and :VOLTage	25-1
25.3	:TEMPerature C CEL F FAR K	25-2
25.4	:TIME HOUR MINute SECond	25-2

Chapter 26 VXI Subsystem

26.1	:CONFigure	26-3
26.1.1	:DNUMber?	26-3
26.1.2	:HIERarchy?	26-3
26.1.2.1	:ALL?	26-4
26.1.2.2	:VERBose?	26-4
26.1.2.2.1	:ALL?	26-4
26.1.3	:INFormation?	26-4
26.1.3.1	:ALL?	26-6
26.1.3.2	:VERBose?	26-6
26.1.3.3	:ALL?	26-6

1999 SCPI Command Reference

26.1.4	:LADdress?	26-6
26.1.5	:NUMBer?	26-7
26.2	REGister	26-7
26.2.1	:READ? <register>	26-7
26.2.1.1	:VERBoSe? <register>	26-8
26.2.2	:WRITe (<numeric_value> <register>), <data>	26-8
26.3	:RESet?	26-8
26.3.1	:VERBoSe?	26-9
26.4	:SElect <logical_address>	26-9
26.5	:WSPRotocol	26-9
26.5.1	:COMMand	26-9
26.5.1.1	[:ANY] <data>	26-9
26.5.1.2	:AHLine <hand_id>,<line_number>	26-9
26.5.1.3	:AILine <int_id>,<line_number>	26-9
26.5.1.4	:AMControl <response_mask>	26-10
26.5.1.5	:ANO	26-10
26.5.1.6	:BAVailable <Boolean>,<byte>	26-10
26.5.1.7	:BNO <Boolean>	26-10
26.5.1.8	:BRQ	26-10
26.5.1.9	:CEVent <Boolean>,<event_number>	26-10
26.5.1.10	:CLR	26-10
26.5.1.11	:CLOCK	26-11
26.5.1.12	:CRESPonse <response_mask>	26-11
26.5.1.13	:ENO	26-11
26.5.1.14	:GDEVice <logical_address>	26-11
26.5.1.15	:ICOMmander <logical_address>	26-11
26.5.1.16	:RDEVice <logical_address>	26-11
26.5.1.17	:RHANDlers	26-11
26.5.1.18	:RHLine <hand_id>	26-12
26.5.1.19	:RILine <int_id>	26-12
26.5.1.20	:RINTerrupter	26-12
26.5.1.21	:RMODid	26-12
26.5.1.22	:RPERror	26-12
26.5.1.23	:RPRotocol	26-12
26.5.1.24	:RSTB	26-12
26.5.1.25	:RSARea	26-13
26.5.1.26	:SLModid <Boolean>,<MODID 6-0>	26-13
26.5.1.27	:SLOCK	26-13
26.5.1.28	:SUModid <Boolean>,<MODID 12-7>	26-13
26.5.1.29	:TRIGger	26-13
26.5.2	:MESSAge	26-13
26.5.2.1	:RECeive? <count> <terminator>	26-13
26.5.2.2	:SEND <message_string> [, (END NEND)]	26-13
26.5.3	:QUERy	26-14
26.5.3.1	[:ANY]? <data>	26-14

1999 SCPI Command Reference

26.5.3.2	:AHLine? <hand_id>,<line_number>	26-14
26.5.3.3	:AIlLine? <int_id>,<line_number>	26-14
26.5.3.4	:AMControl? <response_mask>	26-14
26.5.3.5	:ANO?	26-14
26.5.3.6	:BNO? <Boolean>	26-14
26.5.3.7	:BRQ?	26-15
26.5.3.8	:CEVent? <Boolean>,<event_number>	26-15
26.5.3.9	:CRESPonse? <response_mask>	26-15
26.5.3.10	:ENO?	26-15
26.5.3.11	:RDEvice? <logical_address>	26-15
26.5.3.12	:RHANDlers?	26-15
26.5.3.13	:RHLine? <hand_id>	26-15
26.5.3.14	:RILine? <int_id>	26-16
26.5.3.15	:RINTerrupter?	26-16
26.5.3.16	:RMODid?	26-16
26.5.3.17	:RPERror?	26-16
26.5.3.18	:RPRotocol?	26-16
26.5.3.19	:RSTB?	26-16
26.5.3.20	:RSARea?	26-16
26.5.3.21	:SLModid? <Boolean>,<MODID 6-0>	26-17
26.5.3.22	:SUModid? <Boolean>,<MODID 12-7>	26-17
26.5.4	:RESPonse?	26-17

1999 SCPI Command Reference

1 Introduction

1.1 Requirements

This volume of the SCPI standard, “Command Reference,” shall be used in conjunction with the “Syntax and Style” volume.

1.2 Organization

The first chapter in this volume of the SCPI standard describes the SCPI model that is used in defining the functional areas of SCPI. The subsequent chapters describe each functional area, known as a subsystem, and are in alphabetic order by subsystem name.

2 Instrument Model

A model is used within SCPI as a means of achieving compatibility. SCPI concerns itself with three types of compatibility. The first form of compatibility is called vertical compatibility. Vertical compatibility is where two instruments of the same type have identical controls (e.g. two oscilloscopes both have the same controls for their timebases, triggers, and voltage settings).

The second form of compatibility is called horizontal compatibility. Horizontal compatibility is where two instruments can make the same measurement, regardless of the actual measurement techniques used. To be horizontally compatible, both instruments would use the same commands to make this measurement. For example, both an oscilloscope and a counter can perform a risetime measurement on a pulse. The two instruments are said to be horizontally compatible if the same command is used in both instruments.

The third form of compatibility is called functional compatibility. Functional compatibility is where two instruments which perform the same function do so with the same commands. For example, a spectrum analyzer and an rf source may both sweep in frequency. If the same frequency and sweep commands are used in both instruments, they would be functionally compatible in this area.

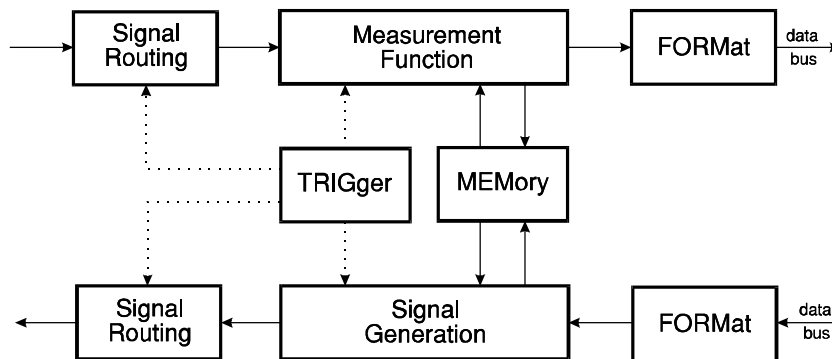


Figure 2-1 Model of a Programmable Instrument

Figure 2-1, “Model of a Programmable Instrument,” represents the way in which instrument functionality is viewed and categorized by SCPI. The purpose of this categorization is to provide organization and consistency between the various commands available in SCPI for all the different types of instrumentation. The model defines where elements of the language must be assigned in the SCPI hierarchy. Major areas of signal functionality are shown broken into blocks; each of these blocks are major command subtrees in SCPI. In this volume of the SCPI standard, each subtree is discussed in its own chapter.

The model describes the flow of measurement and applied signal data through the instrument. The administrative data flows associated with commands, queries, performing calibrations, mass memory accesses, and other related functions are not included in this model. The model does not define how an instrument handles or formats data internally. In

Figure 2-1, data flow is represented by solid arrows and control flow is represented by dashed arrows.

Individual instruments shall only implement those blocks which apply to their implementation. For example, a voltmeter which has a single input and no user addressable memory might include the measurement function, TRIGger, and FORMat blocks.

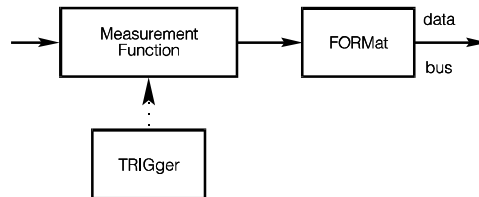


Figure 2-2 Simplified Model for a Sensor Instrument

Similarly, a non-triggered, single output power supply might include the signal generation and FORMat blocks.

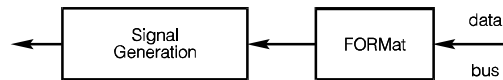


Figure 2-3 Simplified Model for a Source Instrument

An instrument with a particular block is not always required to implement a subtree or any commands associated with that block. This occurs when the block exists in a fixed configuration that does not conflict with related commands, and that have the *RST default values. For example, an instrument might have a fixed TTL level external trigger, with no adjustable TRIGger parameters. In this case no part of the TRIGger subsystem need be implemented. However, if trigger controls exist, those controls shall be implemented as defined in the TRIGger subsystem.

2.1 Signal Routing

The purpose of the signal routing block is to control the routing of signals between an instrument's signal ports and its internal signal functionality. Signal routing also controls the connection from signal port to signal port, where such capability exists. The commands which control this block are described in the SCPI tree under the ROUTe subsystem. The implementation of this subsystem is optional for those instruments that have fixed connections to the measurement function block or the signal generation block.

2.1.1 Setting the Destination for Data Flow

The FEED model works well when the user is focused on setting the source for the data flow into a block. Occasionally, the user's focus is on the destination of data flow from a block. When this view is appropriate, a DESTination <data_handle> command is used.

DESTination is an event; it has no query form. The command sets one or more FEED settings to known values. Assume the subsystem where the DESTination command is located could be accessed with a <data_handle> of <source_data_handle>. When DESTination <data_handle> is executed, any FEED command which is currently set to <source_data_handle> shall be set to “”. The FEED command under the <data_handle> subsystem shall be set to <source_data_handle>. Any subsystem which may appear in the <data_handle> shall contain a FEED command. An allowed value of the <data_handle> for that FEED command shall be <source_data_handle>.

2.2 Measurement Function

The measurement function block converts a physical signal into an internal data form that is available for formatting into bus data. It may perform the additional tasks of signal conditioning and post conversion calculation. The measurement function box is subdivided into three distinct parts: INPut, SENSE, and CALCulate. These are shown as solid boxes in Figure 2-4, “Expanded Measurement Function Model.” The dotted boxes shown represent blocks from Figure 2-1, “Model of a Programmable Instrument.”

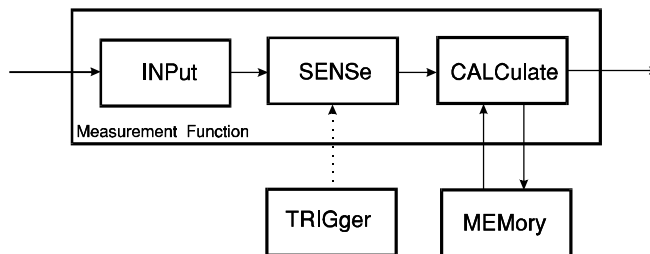


Figure 2-4 Expanded Measurement Function Model

The subdivisions INPut, SENSE, and CALCulate are not detailed in the “Model of a Programmable Instrument” figure because at this lower level of division, functionality across different instruments can no longer be considered horizontally compatible. For example, measuring the frequency of a signal is done in SENSE block on a counter, but is done in the CALCulate block in a spectrum analyzer. The two instruments are considered compatible since they both can perform the function. However, the horizontal compatibility only occurs at the high level MEASurement commands.

2.2.1 INPut

The purpose of the INPut block is to condition the incoming signal before it is converted into data by the SENSE block. INPut block functions include filtering, biasing, frequency conversion (such as a mixer or prescaler function), and attenuation. The INPut block appears in the SCPI tree under the INPut subsystem. The implementation of this subsystem is optional for those instruments that have no INPut block characteristics.

2.2.2 SENSE

The purpose of the SENSE block is to convert signal(s) into internal data that can be manipulated by normal computer techniques. The commands associated with the SENSE block control the various characteristics of the conversion process. Examples are range, resolution, gate time, normal mode rejection, etc. This block does not include any mathematical manipulation of the data after it has been converted.

The SENSE commands appear in the SCPI tree under the SENSE subsystem. The SENSE block is required for all sensor instruments, since the nature of a bus programmable instrument implies a conversion from an electrical signal to a “number.” However, the SENSE subsystem contains a large number of subservient subsystems, subdividing the different signal characteristics being sensed. An instrument is required to only implement the appropriate commands from those subservient subsystems that apply.

2.2.3 CALCulate

The purpose of the CALCulate block is to convert or derive sensed data into a form more useful to the application. Typical calculations include converting units, and postprocessing calculations (for example, calculation of rise time from a time domain waveform). The CALCulate commands are described in the CALCulate subsystem.

2.3 Signal Generation

The signal generation block is responsible for the conversion of data into physical signals. It may perform additional tasks of preconversion calculation and signal conditioning. The signal generation box is subdivided into three distinct parts: CALCulate, SOURce, and OUTPut. These are shown as solid boxes in Figure 2-5, “Expanded Signal Generation Model.” The dotted boxes shown represent blocks from Figure 2-1, “Model of a Programmable Instrument.”

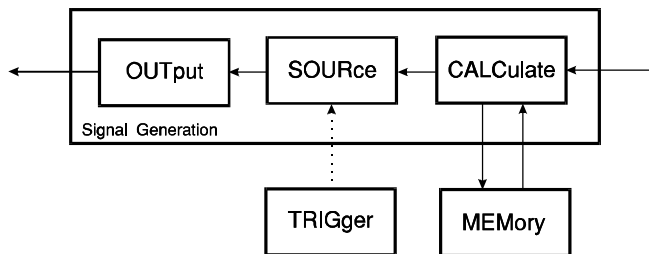


Figure 2-5 Expanded Signal Generation Model

The subdivisions CALCulate, SOURce, and OUTPut are not detailed in the “Model of a Programmable Instrument,” because at this lower level of division, functionality across different instruments can no longer be considered horizontally compatible.

2.3.1 OUTPut

The purpose of the OUTPut block is to condition the outgoing signal after it has been generated. The OUTPut block functions include filtering, biasing, frequency conversion (such as a mixer function), and attenuation. The OUTPut block appears in the SCPI tree

under the OUTPut subsystem. As with the INPut block in sensor-type instruments, this block appears in all source-type instruments, though the commands may not.

2.3.2 **SOURce**

The purpose of the SOURce block is to generate a signal based on specified characteristics and/or supplied data. The commands associated with this block describe the characteristics of the generated signal.

The SOURce commands are described in the SOURce subsystem. The SOURce block is required for all source instruments. The SOURce subsystem contains a large number of subservient subsystems, sub-dividing the different signal characteristics being sourced. An instrument is only required to implement the appropriate commands from those subservient subsystems that apply.

2.3.3 **CALCulate**

The purpose of the CALCulate block is to convert application data to account for anomalies in generating a signal, such as correcting for external effects, converting units and changing domains. The commands associated with the CALCulate block are defined in the CALCulate subsystem.

2.4 **TRIGger**

The purpose of the TRIGger block is to provide an instrument with synchronization capability with external events. The TRIGger block is described in chapter 22, and appears in the SCPI tree as TRIGger, ARM, INITiate, and ABORt subsystems. Two models exist in the TRIGger block, one complex and the other simple. Both models may be reduced to a level that corresponds to the sophistication of an instrument's trigger capabilities.

2.5 **MEMory**

The purpose of the MEMory block is to hold data inside the instrument. The memory may be implicit and inaccessible to the user (for example, internal calibration data), and may be fixed (for example, current measurement data) or may be allocated and user-addressed.

The allocation of memory appears in SCPI under the MEMory subsystem. The manipulation of memory is performed by commands throughout the SENSE, SOURce, FORMat, DISPlay, and other subsystems. While every programmable instrument contains memory, not all such instruments provide the user with explicit control of this memory. In such cases there is no requirement to implement the MEMory subsystem.

2.6 **FORMat**

The purpose of the format block is to convert between data representations, especially on the data that is transferred over the external interface. An example is conversion of internal data formats to ASCII.

Formatting appears in SCPI under the FORMat subsystem. All instruments are required to use the formats described in *IEEE 488.2*.

2.7 **Internal Routing**

The instrument model presented so far introduced the fundamental elements or blocks from which an instrument is constructed, and defined a simple or lamina arrangement of the connections between those blocks, known as the "Lamina Model." This model is adequate

for many instruments. In some instruments the connections may not be as shown, as multiple instances of certain individual blocks from the Lamina Model may exist in the instrument. In such instruments the connections between blocks may be programmable. Each instance of a block is differentiated by means of the numeric suffix used with the block name.

2.7.1 Data and Control Flow

All of the solid lines in the Lamina Model represent data flow between the blocks. The external signal (often electrical) can be considered as data, since it contains information. In a sensor type instrument using the Lamina Model, this data is routed through the ROUTe block into the INPut block, the data then flows into the SENSE block, then on to the CALCulate block, and finally into the DISPlay block or to the system controller over the bus. In a source type instrument using the Lamina Model, data from the controller flows over the bus, through the CALCulate block, then through the SOURce block. Finally, the data in the form of an external signal (often electrical) flows into the OUTPut block, and is then distributed through the ROUTe block.

Obviously, each block has points where data flow is absorbed and emitted. Many blocks have only one point where the data is absorbed and one point where data is emitted, however there is no restriction on how many of either of these points a block must have, further the number of points where data is absorbed and the number of points where data is emitted do not have to appear in equal numbers.

When describing the data flow, the SCPI model specifies for each point where data is absorbed where that data is obtained from. This selection of data flow into a block is made with the FEED command.

There are certain restrictions that apply to the data FEED for each block; these restrictions are enumerated under each block's command description for its FEED. That is, a particular block's FEED is only allowed access to certain points in the data stream, generally the FEED must select a block which is of the same type as the block that immediately precedes it in the instrument model described earlier. For example a SENSE block which deals with physical signals, is not allowed to FEED data from a CALCulate block, which is in an internal format, typically numeric data. Further, SENSE blocks are only allowed to FEED data from INPut blocks. Instruments that effectively have no INPut functionality, that is no form of signal conditioning, still are considered to have INPut blocks, so an INPut is available to be specified as the FEED of the SENSE block. A block, such as the INPut block just described, is called a null block.

Associated with the data flow is a control mechanism to ensure the coherent processing of the data. Each process has points where control flow is absorbed and emitted. There is no restriction on how many of either of these control flow points a block must have. A control flow carries messages or events. These messages or events are absorbed from a control flow by a process, to control the progress and direction of the process. In turn, each process emits messages or events that are used by other processes. The action of moving from one process to another is an event. Each process may be sub-divided into a series of steps, the action from moving from one step to another is also an event.

In a particular instrument, there will be a number of control flows established as part of the instrument's configuration. A typical example is found in the TRIGger block, where the final layer in a trigger sequence generates an event that causes the "device action" to be performed, and a subsequent event is returned by the block(s) performing the device action to indicate its completion. Another example, is one where the completion of a data acquisition by the SENSE block is an event that is used by the CALCulate block to automatically start the calculation process.

In certain cases some of the control flows may be programmable, allowing a user to select the controlling events. When describing the control flow, the SCPI model specifies for each point where control information flows in, where it obtains its control flow from. The selection of a control flow into a block is made with the LINK command.

2.7.2 Numeric Suffixes

Numeric suffixes are associated with each block to provide them with a unique identifier. The way in which a block with a particular suffix is related to other blocks in an instrument depends upon the model employed. The next section describes the Lamina Model and the closely related Cloned Model, also described is the Amorphous SENSE Model.

2.7.3 Lamina and Cloned Models

The Model of a Programmable instrument shows one instance of each of the major blocks (or subsystems) that are available in SCPI. In this case each block is given a numeric suffix of 1. This numeric suffix is not usually shown, as it may be defaulted. The relationship between the blocks is fixed in the order shown. For example, Figure 2-6 shows the Lamina Model for a sensor instrument where the name of each block is shown in its explicit form.

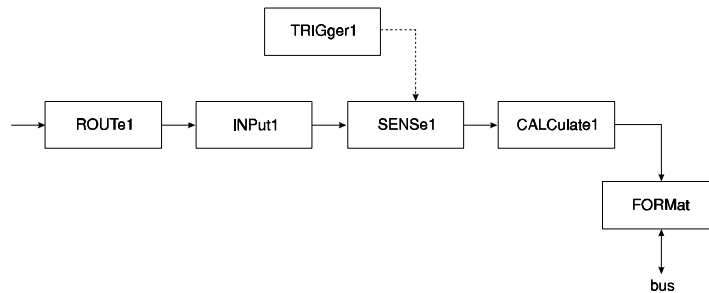


Figure 2-6 Lamina Model for a Sensor Instrument

The Cloned Model repeats the Lamina Model as many times as required. Each repetition of the Lamina Model is accompanied with a different numeric suffix, incremented by 1 from the last. The cloned model is suitable for instruments that have multiple independent capabilities. For example, shown in Figure 2-7 is the Cloned Model that could be used to represent a multiple independent capability source instrument.

2.7.3.1 FEED Exceptions

As mentioned earlier, a particular block's FEED is only allowed access to certain points in the data stream, generally the FEED must select a block which is of the same type as the block that immediately precedes it, in the Instrument Model.

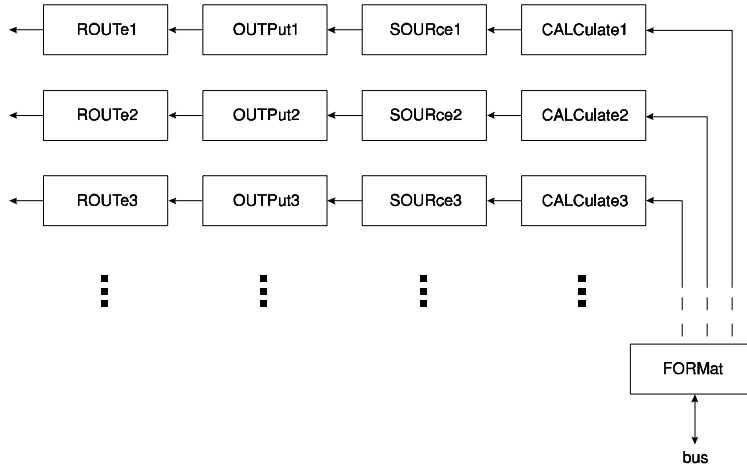


Figure 2-7 Cloned Model for a Multi-Channel Instrument

CALCulate typically follows this rule, obtaining data from the preceding block, which in the case of a sensor instrument would be the SENSE block. However, where an implementation permits, CALCulate blocks may be cascaded to provide complex functionality, it is permissible for a CALCulate block to use another CALCulate block as its FEED, as shown in Figure 2-8:

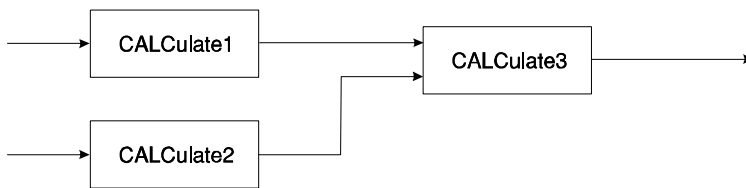


Figure 2-8 Cascaded CALCulate Blocks

DISPlay is also a special case, it may obtain data from any point where the data is in an internal format, typically numeric data. Examples include a CALCulate block or a SENSE block.

2.7.3.2 Amorphous SENSE Model

In certain sensor instruments there are SENSE blocks that produce parallel independent results, where certain parts of the SENSE subsystem are shared. A distinguishing feature of the Amorphous SENSE Model is that the SENSE block has more than one INPut block

attached to it. In this model the numeric suffix associated with the INPut block is associated with the subsystems of the SENSE block, not the SENSE itself. Thus INPut<n>, where <n> is a number, would relate to the subsystems of SENSE such as, SENSE:VOLTage<n>, SENSE:CORRection<n> or SENSE:DETEctor<n>. This is typical, unless a particular subsystem of SENSE has been explicitly configured differently, to provide different or additional combinations of functionality.

One example, is where a SENSE block with two power sensors, provides two POWER results and a cross-product result such as a POWER:RATio. Another example, is where a SENSE block with two voltage sensors and two independent time sweep capabilities can be arranged to provide four different results.

2.7.3.3 Other Subsystems

Only one instance of certain blocks (or subsystems) is permitted in an instrument, these are FORMat, INSTRument, MEMory, MMEMory, STATus, SYSTem and TRACe. In most instruments there is only one DISPlay, since it is rare to find instruments with two or more displays that are entirely separate and distinct.

2.7.4 FEED <data_handle>

The parameter to a FEED command is <data_handle>. All <data_handle>s are defined to be IEEE 488.2 STRING PROGRAM DATA. Along a data stream there are discrete points where data may be picked up. These points have names that are called <data_handle>s. For blocks with a single point where data is emitted, the block name with its numeric suffix is the <data_handle>. For example, the <data_handle> for CALCulate3 is the string "CALCulate3".

Note that the <data_handle> in the FEED command may consist of either long or short forms mnemonics. However the query shall return the short form and shall omit any default nodes. Thus, the query for the above example would return the string "CALC3".

The <data_handle>s for the different SENSE block found in each of the model, is described in the following sections.

2.7.4.1 Sensor Function Selection

Selecting a sensor's function effects the data flow inside of the SENSE block. Selecting a particular <sensor_function> enables the data flow associated with it. If a <sensor_function> is enabled then there must be a <data_handle> associated with the data flow generated by that function. A SENSE block that has only a single point where data is emitted has a single SENSE:FUNCTION selected. The selections are mutually exclusive of one another, that is enabling one function would disable another. In the case shown in Figure 2-10, the SENSE:FUNCTION selected is "RESistance", since it is the only data flow emitted from the block, SENSE1 is all that is needed to create a unique <data_handle> for this case.

In instruments which have amorphous SENSE blocks one or more functions can be enabled simultaneously within the same block, that is selecting a function adds to the list of currently enabled functions. Such a SENSE block has one or more points where data is emitted, thus the SENSE block and its suffix is no longer sufficient to distinguish the different data flows.

It is necessary for the <data_handle> for these data flows to include the SENSE:FUNCTION that generated the flow, along with the SENSE block and its numeric suffix.

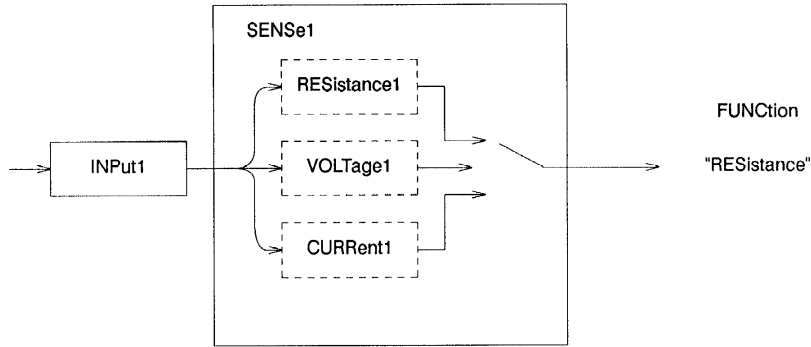


Figure 2-10 Single Function Active SENSE Block

In Figure 2-9, the sensed characteristic alone is not sufficient, to uniquely specify a function, and therefore cannot be used to uniquely identify a data flow. By including in the <sensor_function> the INPut block number(s) associated with a particular data flow, within the SENSE block, a unique specification for the <sensor_function> and thus a unique <data_handle> can be formed. Where the <sensor_function> is included in a <data_handle>, the SENSE node with the optional numeric suffix of 1, may be omitted altogether, since many instruments that employ an amorphous SENSE block, will have only one SENSE block, that is SENSE1.

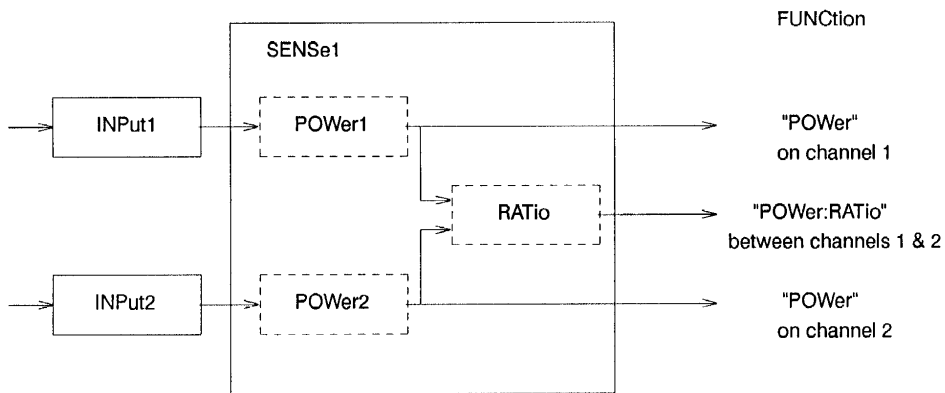


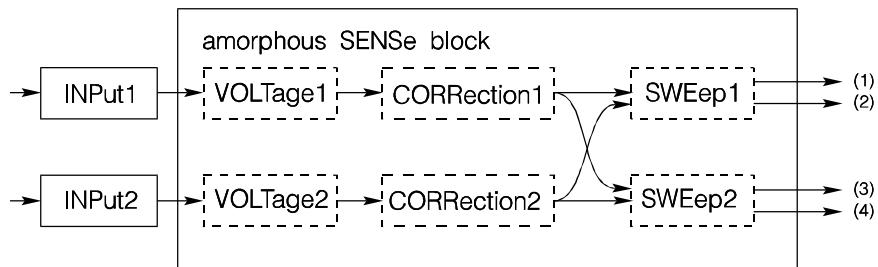
Figure 2-9 Amorphous SENSE Block with Cross-Product

A specification of a <sensor_function> shall consist of the function and optionally the INPut block number. The default INPut suffix is 1. For the cross-product POWER:RATio, both INPutS must be included. To define an INPut block in the specification, only the INPut suffix is included, and not the mnemonic INPut. For the example above the <sensor_function>s and corresponding <data_handle>s are, respectively from top to bottom:

DESCRIPTION	<sensor_function>	<data_handle>
The power function on INPut1	"POWER 1"	"SENSe:POWER 1"
The power ratio on INPut1 & INPut2	"POWER:RATio 1,2"	"SENSe:POWER:RATio 1,2"
The power function on INPut2	"POWER 2"	"SENSe:POWER 2"

The amorphous SENSE block has an additional level of complexity, where SENSE subsystems (or sub-blocks) are used in combination as shown in Figure 2-11.

Figure 2-11 shows an amorphous SENSE block with shared SWEep subsystems, and the corresponding <sensor_function>s. It should be noted that the function part of the specification indicates the physical characteristic being sensed, and not which subsystem of SENSE is used. The subsystems used are specified by association with the INPut block, in the example above VOLTage1 and CORRection1 are tied to the data flow associated with INPut1. Where the association is not as specified above, the subsystem with its numeric suffix is appended to the specification after the literal token 'ON'. Note how SWEep1 associated with INPut1 and SWEep2 associated with INPut2 can be defaulted, as they follow the default relationship of an amorphous SENSE block.



- (1) "VOLTage:AC 1 ON SWEep1" or "VOLTage:AC 1"
- (2) "VOLTage:AC 2 ON SWEep1"
- (3) "VOLTage:AC 1 ON SWEep2"
- (4) "VOLTage:AC 2 ON SWEep2" or "VOLTage:AC 2"

Figure 2-11 Amorphous SENSE Block with Shared SWEep

The <data_handle> is then the <sensor_function> preceded by the SENSE block and its suffix. Thus a FEED for a display trace from an amorphous SENSE block, as shown above, might look like:

```
DISPlay:WINDow:TRACe:FEED "SENSE1:VOLTage:AC 2 ON SWEEp1"
```

2.7.4.2 FEEDs from CALCulate Sub-Blocks

The CALCulate block has only one data stream through it. The CALCulate block is divided into smaller sub-blocks, which correspond to the subsystems found in the CALCulate subsystem.

Due to the discrete way in which each active sub-block exists, the data stream is accessible after each sub-block, where permitted by the implementation. The name of the CALCulate block and its numeric suffix is not sufficient to select the intermediate points. Therefore the <data_handle> to access these intermediate points is comprised of the CALCulate block name and numeric suffix followed by a colon (:) and the name of a CALCulate sub-block. For example, if a CALCulate consists of MATH1, TRANSform, followed by MATH2, then the <data_handle> to select the data flow after the TRANSform, but before MATH2, would be the string "CALCulate:TRANSform"

2.7.5 COMBine

The COMBine block acts to combine several data streams or signals. Each separate input to a COMBine block has a FEED command.

The output is the sum of the inputs. For example if the inputs are complex vectors representing waveforms with certain phases, then the output is the linear combination of the inputs which is also a complex vector.

Several different FEED commands might possibly specify the same <data_handle>, in effect routing a single data stream into multiple subsystems.

2.7.6 Memory Associated with Data Flow

In the Model of a Programmable Instrument a MEMory block is shown adjacent to the other blocks in the model. For the user area memory this appears to be the correct model, however, for "instrument data" this proves to be cumbersome. A better model for "instrument data" is obtained by imagining that memory is something that pervades the whole model, any point along the data flow potentially may have memory associated with it. In general terms memory exists at all points along this flow of data.

The examples in Figure 2-12 show how memory is associated with the simplest of blocks, it then shows how memory is associated with a CALCulate block consisting of three sub-blocks. This is a very simple model to present to the user; at every point along the data flow memory exists. The fact that it exists, does not mean that a particular implementation has to allow the user access to it, as the data may not be in a form useful to the user. If access is provided to these intermediate memory stores, certain restrictions may apply depending on the memory usage. Within this model there are two types of memory:

- TRANSIENT - memory and data are retained only long enough to be used by next step in data flow

- **STATIC** - memory is permanently allocated and data is retained, until next “acquisition”

If the memory is transient then it is not possible to obtain data from it after the data has flowed past this point. Thus, querying such a point will not return anything, until new data flows into the newly created transient memory. This would occur if such a query were sent in the same program message as an INITiate command.

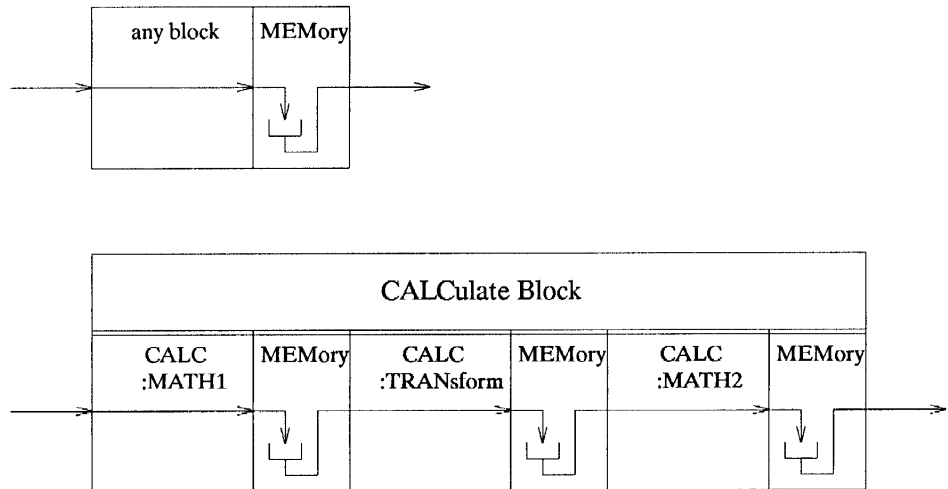


Figure 2-12 MEMory Association with a Generic and a CALCulate Block

Static memory is permanently allocated to the specified point in the data flow. Any data flowing past this point will fill this memory, and the data at that point may be queried at any time. However, should a new data flow start, for example with an INITiate, then the contents of the memory shall be replaced with the new data.

The names used in querying memory should reflect where they occur in the data flow. To best achieve this the “handles” given to these points for connecting blocks together are also appropriate in data query.

There is a third type of memory, user memory which is already defined in the TRACeDATA subsystem.

Given the FEED mechanism it is possible for a TRACe name, such as MY-TRACE, to tap the data flow. Every time that new data flows past the “tapped” point, MY-TRACE will be updated with the new data. While this may be useful, a significant benefit can be achieved by providing a control mechanism that selectively allows data into the TRACe, upon a specified condition of events. For example, it should be possible to set up the control so that MY-TRACE will only receive new data when a limit test fails.

It can be seen that a user can create a TRACe, such as MY- TRACE, that may track the “tapped” point, but MY-TRACE and the memory at the tapped point are distinct and separate. Note that the memory at that point will always (while it exists) reflect the latest data flow, while MY-TRACE may not since the data flow into it may be gated. It is not permitted to create TRACe | DATA names identical to any data handles.

2

2.7.7 Querying the Data Flow

SCPI allows the data flow to be queried, so that a system controller can obtain intermediate results. This is only possible where the data flow is in an internal format, typically numeric data. That is, if the data flow is some physical characteristic such as an electrical signal, then the data flow at such a point is not queryable.

The form of the query is generally the block name followed by :DATA?. In certain cases, this either does not provide a unique data flow, or more resolution is required than just the block level. The SENSE:DATA? query has an optional function name as a parameter to define the particular data flow required out of the SENSE block. An implementation may allow access to the data flow after each sub-block subsystem in the CALCulate block. Thus, the query becomes the heirarchical name of the sub-block followed by :DATA?, for example the data flow after MATH1 could be obtained with CALCulate:MATH1:DATA?. The returned data is affected by the format subsystem.

2.7.8 <event_handle>

The <event_handle> is the way events are described that are used by the LINK commands throughout SCPI. Events may exist in all blocks of the Instrument Model, although a particular implementation will dictate what events are available to be used with each LINK command implemented.

The general form of <event_handle> is:

to be determined

Events that are commonplace in instruments, are those found in the TRIGger subsystem; the action of exiting any layer generates an event. The <event_handle>s to reference these TRIGger generated events are of the form:

`“ARM[:SEQUENCE[:LAYER]]”` or `“TRIGger[:SEQUENCE]”`

Other common examples include completion of a process in a block or an instrument state transition.

3 Measurement Instructions

The purpose of the MEASure group of instructions is to acquire data using a set of high-level instructions. The MEASure group of instructions have a duality, in that they exhibit command and query characteristics. The exception to this is CONFigure, which has distinct query and command forms. These instructions are independent of the block diagram and refer to the characteristics of the signal being measured. These instructions are intended to be used with the measurement <functions> referenced later in this chapter.

The MEASure group of commands are structured to allow the user to trade off interchangeability with fine control of the measurement process. MEASure? provides a complete capability where the instrument is configured, a measurement taken, and results returned in one operation. Often, more precise control of the measurement is required. Therefore, MEASure? is complemented by providing two commands, CONFigure and READ?. CONFigure performs the configuration portion of the measurement and READ? performs the data acquisition, postprocessing, and data output portions of the measurement. This allows the user to perform a generic configuration of the measurement through CONFigure and then customize the measurement by changing particular instrument-dependent functions. The READ? then completes the measurement process.

READ?, in turn, is broken down into two additional commands INITiate[:IMMediate] and FETCh?. INITiate[:IMMediate] performs the data acquisition. This command is described in chapter 22. FETCh? performs the postprocessing function and returns the data. This allows the user to perform several different FETCh? functions on a single set of acquired data. For example, an oscilloscope can acquire measurement data that can yield many different signal characteristics such as frequency or AC and DC voltages. Thus, a transient signal may be captured once using a MEASure?, READ? or INITiate. A FETCh? may then be used to obtain each of the different signal characteristics without reacquiring a new measurement.

MEASure? provides the best compatibility between instruments because no knowledge of the instrument is required to perform the operation. CONFigure/READ? is less compatible if instrument reconfiguration is performed between the CONFigure and READ? operations. This is because the reconfiguration is, by necessity, instrument-specific. FETCh? is also less compatible because knowledge of the instrument is necessary to determine whether the necessary information has been captured by the instrument. For example, an oscilloscope can capture both rise time and pulse width in a single acquisition. Therefore, if pulse width was acquired through a MEASure? command, it would be possible to FETCh? rise time. However, if pulse width were measured with a counter, the rise time information might not be available without performing another data acquisition. Therefore FETCh? could not be used.

Changing certain parts of an instrument's configuration shall cause existing measurement data to be invalidated. Specifically, in figure 2-1, "Generalized Model of a Programmable Instrument" in chapter 2, "Instrument Model," any reconfiguration of signal routing, measurement function, signal generation and/or trigger shall cause existing readings to be invalidated. For example, the sequence:

```
INITiate;CONFigure:VOLTage;FETCh:VOLTage?
```

would cause an error to be generated as the data was invalidated by the CONFIGure command. Reconfiguring the display or format blocks shall not have this effect.

KEYWORD	PARAMETER FORM	NOTES
CONFIGure:<function>	<parameters>[,<source list>]	
FETCh[:<function>]?<parameters>[,<source list>]		[query only]
READ[:<function>]?<parameters>[,<source list>]		[query only]
MEASure:<function>?<parameters>[,<source list>]		[query only]

3

3.1

CONFIGure:<function> <parameters>[,<source list>]

Sets up the instrument in order to perform the measurement specified by the function. Individual functions are described in <functions> later in this chapter. The parameters of this command are described in the individual function descriptions. The execution of the CONFIGure command may affect the value of any other setting in the instrument. The state of the instrument is not specified after the execution of this command, except that a subsequent READ? QUERY operation would perform the specified function.

If parameters are omitted, they shall assume the values specified by the particular function description. Parameters may be defaulted from the right by omitting them, or anywhere by substituting the keyword DEFAULT. DEFAULT must be accepted as a parameter for this command.

The query form of the command, CONFIGure?, returns the setup configured by the last CONFIGure or MEASure? query. If the instrument state has changed through receiving commands other than CONFIGure or MEASure?, the information returned by a CONFIGure? may not reflect the actual measurement conditions. However, it is permissible for an instrument to attempt to track these changes and construct an accurate CONFIGure query response. The format of the response data is <STRING RESPONSE DATA> which is of the form:

```
"<function> <parameters>"
```

For example, if the previous CONFIGure command:

```
CONFIGURE:VOLTage:AC 5,.001
```

had been executed, the query CONFIGure? would return:

```
"VOLT:AC 5.0,0.001"
```

A special parameter type is the <source list>. The <source list> is syntactically a <channel list> as described in "Syntax and Style." The purpose of the <source list> is to specify the physical port which is the source of the measurement. For example, (@2) would specify that the measurement should occur on channel 2. Multiple channels can be specified by using multiple channels in the channel list. For example, (@1,3:5,9) specifies that measurements are to be taken on channels 1,3,4,5, and 9. SCPI does not specify if the acquisitions are simultaneous or sequential. A function which requires multiple channels, such as time interval, would specify each channel as a separate list. For example, (@1),(@2) would specify a two-channel measurement between channels 1 and 2. A multiple channel, multiple measurement is specified by using multiple channels in each of the lists. For example,

(@1,2),(@3,4) specifies a pair of two-channel measurements, the first between channels 1 and 3 and the second between channels 2 and 4.

At *RST, the configuration is set to a device-dependent value.

3.2 **FETCh[:<function>]? <parameters>[,<source list>]**

Retrieves the measurements taken by the INITiate command and places them into the device's output buffer. By specifying a function as part of a compound header, the device will retrieve the value of the function requested derived from the data taken by the latest INITiate command. For instance, even if a WAVEform was the original function when the measurement was taken, functions that can be calculated from the stored data can be FETChed. When <function> is omitted, the last <function> fetched, read, or measured shall be used.

The FETCh? query will return data any time that the last reading is valid. Data becomes invalid under the following conditions:

- When *RST is executed,
- When an INITiate is executed.
- When there is any reconfiguration of signal routing, measurement function, signal generation and/or trigger blocks.
- When the sensor begins acquisition of a new reading.

If data is invalid, the FETCh? query shall not be complete until all data is valid. The exceptions to this are, if the instrument is in the trigger IDLE state and the data is invalid, or the instrument has been reconfigured as defined above and no new measurement has been initiated. In such cases, the FETCh? routine shall generate an error -230 and no result shall be returned. A common cause for this error is receiving a FETCh? after a *RST.

Parameters are allowed in the FETCh? query. They may be defaulted from the right by omitting them, or in any other position by substituting the keyword DEFault. DEFault must be accepted as a parameter for this query. If a parameter is specified, it is used in the calculation if applicable. If the parameter is invalid because it conflicts with the acquired data, an error -221 shall be generated. If the instrument receives a parameter which is unexpected (too many parameters), it shall process the query and return the data. The unexpected parameter shall be ignored, and the "command warning" bit of the "data QUESTIONable" status register shall be set.

A special parameter type is the <source list>. The <source list> is syntactically a <channel list> as described in "Syntax and Style." The purpose of the <source list> is to specify the physical port which is the source of the measurement. For example, (@2) would specify that the measurement should occur on channel 2. Multiple channels can be specified by using multiple channels in the channel list. For example, (@1,3:5,9) specifies that measurements are to be taken on channels 1,3,4,5, and 9. SCPI does not specify if the acquisitions are simultaneous or sequential. A function which requires multiple channels, such as time interval, would specify each channel as a separate list. For example, (@1),(@2) would specify a two-channel measurement between channels 1 and 2. A multiple channel, multiple

measurement is specified by using multiple channels in each of the lists. For example, (@1,2),(@3,4) specifies a pair of two-channel measurements, the first between channels 1 and 3 and the second between channels 2 and 4.

The format of the returned data is determined by the FORMat command.

3.3 **READ[:<function>]? <parameters> [,<source list>]**

The READ? query is identical to:

```
ABORt;
INITiate;
FETCh[:<function>]? <parameters>[,<source list>]
```

The READ? function provides a method of performing a FETCh? operation on fresh data.

A common application is to use the READ? in conjunction with a CONFigure command in order to provide a MEASure? capability in which the application programmer is allowed to provide fine adjustments to the instrument's block diagram. For example,

```
CONFigure:VOLTage:RISE:TIME 10 PCT, 90 PCT, 0.001 S
SWEep:TIME .05 S
VOLTage:AC:RANGE 5 V
READ:VOLTage:RISE:TIME?
```

Allows the applications programmer to specify the amplitude sensitivity and sweep time in order to provide the exact measurement needed.

If parameters are omitted, they are assumed to be those currently in use. Parameters may be defaulted from the right by omitting them, or anywhere by substituting the keyword DEFault. DEFault shall be accepted as a parameter for this query. If the instrument receives a parameter which is unexpected (too many parameters), it shall process the query and return the data, ignoring the unexpected parameter and shall also set the "command warning" bit of the "data QUEStionable" status register.

A special parameter set is the <source list>. The <source list> is syntactically a <channel list> as described in "Syntax and Style." The purpose of the <source list> is to specify the physical port which is the source of the measurement. For example (@2) would specify that the measurement should occur on channel 2. Multiple channels can be specified by using multiple channels in the channel list. For example, (@1,3:5,9) specifies that measurements are to be taken on channels 1,3,4,5, and 9. SCPI does not specify if the acquisitions are simultaneous or sequential. A function which requires multiple channels, such as time interval, would specify each channel as a separate list. For example, (@1),(@2) would specify a two-channel measurement between channels 1 and 2. A multiple channel, multiple measurement is specified by using multiple channels in each of the lists. For example, (@1,2),(@3,4) specifies a pair of two channel measurements, the first between channels 1 and 3 and the second between channels 2 and 4.

The format of the returned data is determined by the FORMat command.

If TRIGger:SOURce BUS is selected, execution of this query shall cause an error -214 to be generated and no data shall be returned. If ARM:SOURce BUS is selected, the same action shall occur, except that the error returned shall be -215.

3.4 **MEASure:<function>? <parameters>[,<source list>]**

The MEASure? query is a query identical to:

```
ABORt;
CONFIgure:<function> <parameters>;
READ:<function>? <parameters>[,<source list>];
```

The MEASure? query provides a complete measurement sequence, including configuration and reading of the data. MEASure? is used when the generic measurement is acceptable and fine adjustment of the instrument block diagram is unnecessary.

If parameters are omitted, they shall assume the values specified by the particular function description. Parameters may be defaulted from the right by omitting them, or anywhere by substituting the keyword DEFault. DEFault must be accepted as a parameter for this query. If the instrument receives a parameter which is unexpected (too many parameters), it shall process the query and return the data, ignoring the unexpected parameter and shall also set the “command warning” bit of the “Data questionable” status register.

A special parameter set is the <source list>. The <source list> is syntactically a <channel list> as described in “Syntax and Style.” The purpose of the <source list> is to specify the physical port which is the source of the measurement. For example (@2) would specify that the measurement should occur on channel 2. Multiple channels can be specified by using multiple channels in the channel list. For example, (@1,3;5,9) specifies that measurements are to be taken on channels 1,3,4,5, and 9. SCPI does not specify if the acquisitions are simultaneous or sequential. A function which requires multiple channels, such as time interval, would specify each channel as a separate list. For example, (@1),(@2) would specify a two-channel measurement between channels 1 and 2. A multiple channel, multiple measurement is specified by using multiple channels in each of the lists. For example, (@1,2),(@3,4) specifies a pair of two-channel measurements, the first between channels 1 and 3 and the second between channels 2 and 4.

The format of the returned data is determined by the FORMat command.

If TRIGger:SOURce BUS is selected, execution of this query shall cause an error -214 to be generated and no data shall be returned. If ARM:SOURce BUS is selected, the same action shall occur, except that the error returned shall be -215.

3.5 **<function>**

Functions define the measurement operation to be used by the MEASure?, CONFIgure, READ?, and FETCh? instructions and are used directly in combination with these instructions. The functions used in conjunction with the MEASure? instructions are different than those described in the SENSE chapter, in that the MEASure? functions are signal-oriented and potentially independent of the hardware.

A function is described as a subtree under the CONFigure, READ?, FETCH?, and MEASure? instructions. The functions are layered into presentation, fundamental measurement, and measurement function layers as follows:

```
MEASure[:<presentation layer>]
      [:<fundamental measurement layer>]
      :<measurement function>?
```

For example, to measure an array of DC Voltage measurements, we specify the following:

```
MEASure:ARRAY:VOLTage:DC?
```

The presentation layer defaults to a simple scalar measurement. The fundamental measurement layer has a device dependent default. For example, an oscilloscope would default VOLTage as its <fundamental measurement layer>. Therefore to perform a rise time measurement on an oscilloscope, the instrument shall accept any and all of the following commands:

```
MEASure:SCALar:VOLTage:RISE:TIME?
MEASure:VOLTage:RISE:TIME?
MEASure:SCALar:RISE:TIME?
MEASure:RISE:TIME?
```

Each layer of the function can have a set of parameters associated with it. Within each set, parameters default from the right. Any parameter in a set may be optional unless the command description states that the parameter is required. Further, any parameter which is allowed to be defaulted may be defaulted by substituting the keyword DEFault for the parameter.

Each set of parameters is separated from the others by enclosing them in parentheses. The exception is that the rightmost parameter set may omit the parentheses. Therefore the syntax is as follows:

```
MEASure:<presentation layer>
      :<fundamental measurement layer>
      :<measurement function>?
      [[(<presentation parameters>)]
      ,(<fundamental measurement parameters>)]
      ,<measurement function parameters>]
      [,<source list>]
```

For example, if we wished to measure an array of voltage rise times, where we wished to acquire 100 readings, and the approximate voltage value was 5 V, and the expected 10-90 rise time was 5 ms, we could write the following query:

```
MEASure:ARRAY:VOLTage:RISE:TIME? (100),(5 V),10,90,5 MS
```

The parameter group is only allowed when the corresponding header is part of the command. For example, the command:

```
MEASure:RISE:TIME? (5 V),10,90,5 MS
```

would generate an error because the fundamental measurement parameters have been specified without specifying the fundamental measurement layer in the header. However, as

the syntax shows, it is permissible to default groups of parameters from the left. Therefore it is possible to default the fundamental measurement parameters while specifying the fundamental measurement layer as follows:

MEASure:VOLTage:RISE:TIME? 10,90,5 MS

A majority of the functions share two parameters, expected value and resolution. **Expected value** is a guess as to the actual value of the measurement. It is used instead of range because range is a device-dependent concept, describing the actual sensor characteristics, while expected value is signal-oriented, describing the expected signal.

Resolution describes the absolute resolution of the measurement. If the parameter is omitted, the instrument selects a resolution. If the instrument cannot provide a measurement at the desired resolution, an error -231 is generated, but an answer is still returned if possible.

3.6 Presentation Layer

The purpose of the presentation layer is to handle groups of data and present them in the way the user specifies. This area is still under discussion, and therefore most of the presentation types (for example, histograms, time and frequency domain waveforms, and so on) are not included in this release.

3.6.1 Presentation Command Summary

PRESENTATION KEYWORD	PARAMETER
:SCALar]	
:ARRay	<size>

3.6.2 [:SCALar]

A SCALar or single value of the desired measurement is taken. SCALar has no associated parameters, and is a default node at this level.

3.6.3 :ARRay <size>

An array of readings is specified. No domain information (time or frequency spacing) is implied in the record. There is one associated parameter, size. This specifies the number of readings to be taken. This parameter cannot be defaulted.

3.7 Fundamental Measurement Layer

The purpose of the fundamental measurement layer is to specify the fundamental signal characteristic being measured.

3.7.1 Fundamental Measurement Command Summary

KEYWORD	PARAMETER
VOLTage	[<expected_value>[,<resolution>]]
CURRent	[<expected_value>[,<resolution>]]
POWER	[<expected_value>[,<resolution>]]
RESistance	[<expected_value>[,<resolution>]]
FRESistance	[<expected_value>[,<resolution>]]
TEMPerature	[<transducer>[,<type>[,<expected value>[,<resolution>]]]]

3.7.2 :VOLTage [<expected_value>[,<resolution>]]

Measures the voltage characteristics of the signal. If resolution is not specified, the instrument chooses a resolution based on its own tradeoffs. If <expected_value> is not specified, either autoranging or a default range is selected.

3.7.3 :CURRent [<expected_value>[,<resolution>]]

Measures the current characteristics of the signal. If resolution is not specified, the instrument chooses a resolution based on its own tradeoffs. If <expected_value> is not specified, either autoranging or a default range is selected.

3.7.4 :POWer [<expected_value>[,<resolution>]]

Measures the power characteristics of the signal. If resolution is not specified, the instrument chooses a resolution based on its own tradeoffs. If <expected_value> is not specified, either autoranging or a default range is selected.

3.7.5 :RESistance [<expected_value>[,<resolution>]]

Measures the resistance at the input. If resolution is not specified, the instrument chooses a resolution based on its own tradeoffs. If <expected_value> is not specified, either autoranging or a default range is selected.

3.7.6 :FRESistance [<expected_value>[,<resolution>]]

Measures the four-wire resistance at the input. If resolution is not specified, the instrument chooses a resolution based on its own tradeoffs. If <expected_value> is not specified, either autoranging or a default range is selected.

3.7.7 :TEMPerature [<transducer>[,<type>[,<expected_value>[,<resolution>]]]]

Measures the temperature at the sensor. The <transducer> parameter is character data which can take on the following values:

KEYWORD	DESCRIPTION
TCouple	Thermocouple
THERmistor	Thermistor
FTHERmistor	Four-wire thermistor
RTD	Resistive temperature device
FRTD	Four-wire resistive temperature device

If a transducer is not specified, the instrument-dependent default is used.

The <type> parameter is either character data or <NRF> format data, depending on which transducer is specified. If TCouple is selected, the parameter is character data specifying the thermocouple type. Examples of valid values for thermocouple type are: J,K,R,S,T,B,E, and N14.

Thermistors and four-wire thermistors use the nominal resistance value for the type, as a numeric parameter. Typical values are 2250, 5000, and 10000.

RTDs and FRTDs use a <numeric_value> which is specified by the manufacturer. Typical values are 85 and 92.

If a type is not specified, an instrument-dependent default is used.

If resolution is not specified, the instrument chooses a resolution based on its own tradeoffs. If <expected value> is not specified, either autoranging or a default range is selected.

3.8 Measurement Function Layer

This layer presents the specific functions to be measured. This layer has been broken down into groups of measurements to provide clarity. This is only done for purposes of clarifying the presentation and there is no grouping provided in the tree.

A trailing :RATio node may be added to the Measurement Function layer, to request for a RATio measurement between the characteristics of two signals. The RATio taken, relates to the signal characteristics specified by the preceding nodes.

The <source_list> consists of two <channel_list> parameters if the RATio node is employed in the command. The first <channel_list> parameter specifies the numerator and the second one is the denominator. For example, MEASure:FREQuency:RATio? (@1),(@2) measures the ratio of the signal frequency of channel 1 to the signal frequency on channel 2.

The parameters, specified with a ratio measurement command, relate to the signal characteristics of the first <channel_list>; this is the signal which determines the numerator.

3.8.1 Simple Measurements

The simple measurements set the fundamental characteristics of steady state signals.

3.8.1.1 Simple Measurements Command Summary

KEYWORD	PARAMETER	
:AC		
[:DC]		
:FREQuency	[<expected_value>[,<resolution>]]	
:BURSt	[<expected_value>[,<resolution>]]	1992
:PRF	[<expected_value>[,<resolution>]]	1992
:PERiod	[<expected_value>[,<resolution>]]	
:PHAsE	[<expected_value>[,<resolution>]]	1992

3.8.1.2 :AC

Measures the AC RMS characteristic of the signal.

3.8.1.3 [:DC]

Measures the DC level of the signal.

3.8.1.4 :FREQuency[<expected_value>[,<resolution>]]

Measures the frequency of the signal. If resolution is not specified, the instrument chooses a resolution based on its own tradeoffs. If <expected_value> is not specified, either autoranging or a default range is selected.

3.8.1.4.1 :BURSt[<expected_value>[,<resolution>]]

Measures the frequency of an ON-OFF modulated signal during its ON period. Such signals are often referred to as burst signals. The characteristic to be measured is the frequency inside the burst as is shown in Figure 3-1. If <resolution> is not specified, the instrument chooses a resolution based on its own tradeoffs. If <expected_value> is not specified either auto ranging or a device dependent, particular range is selected.

3.8.1.4.2 :PRF [<expected_value>[,<resolution>]]

Measures the repetition frequency of the ON-OFF modulation of a signal that is periodically switched ON and OFF. This characteristic is often referred to as Pulsed Repetition Frequency (PRF) as shown in Figure 3-1. If <resolution> is not specified, the instrument chooses a resolution based on its own tradeoffs. If <expected_value> is not specified either auto ranging or a default range is selected.

3.8.1.5 :PERiod [<expected_value>[,<resolution>]]

The period (that is, 1/frequency) of a signal. If resolution is not specified, the instrument

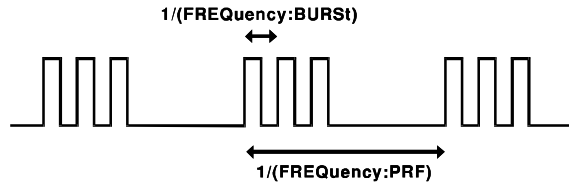


Figure 3-1 Burst Signal

chooses a resolution based on its own tradeoffs. If <expected_value> is not specified, either autoranging or a default range is selected.

3.8.1.6 :PHASe [<expected_value>[,<resolution>]]

Measures the PHASe. If <resolution> is not specified, the instrument chooses a resolution based on its own tradeoffs. If <expected_value> is not specified either auto ranging or a default range is selected.

If two channel lists are specified, the second shall be the reference.

3.8.2 Time Domain Waveform Measurements

These functions measure the time domain characteristics of pulsed signals.

Figure 3-2 is used to illustrate the terms being used to describe pulse signal characteristics.

3.8.2.1 Waveform Measurements Command Summary

KEYWORD	PARAMETER FORM
:AMPLitude	
:LOW	
:HIGH	
:RISE	
:TIME	[<lower reference>[,<upper reference> [,<expected_value>[,<resolution>]]]]
:OVERshoot	
:PREShoot	

KEYWORD

PARAMETER FORM

:RTIME	[<lower reference>[,<upper reference> [,<expected_value>[,<resolution>]]]]
:FALL	
:TIME	[<lower reference>[,<upper reference> [,<expected_value>[,<resolution>]]]]
:OVERshoot	
:PREShoot	
:FTIME	[<lower reference>[,<upper reference> [,<expected_value>[,<resolution>]]]]
:NWIDTH	[<reference>]
:PWIDTH	[<reference>]
:PDUTYcycle:DCYCLE	[<reference>]
:NDUTYcycle	[<reference>]
:TMAXimum	
:TMINimum	
:MINimum	
:MAXimum	
:PTPeak	

1992

3.8.2.2 :AMPLitude

Amplitude is the signal amplitude calculated by taking HIGH - LOW.

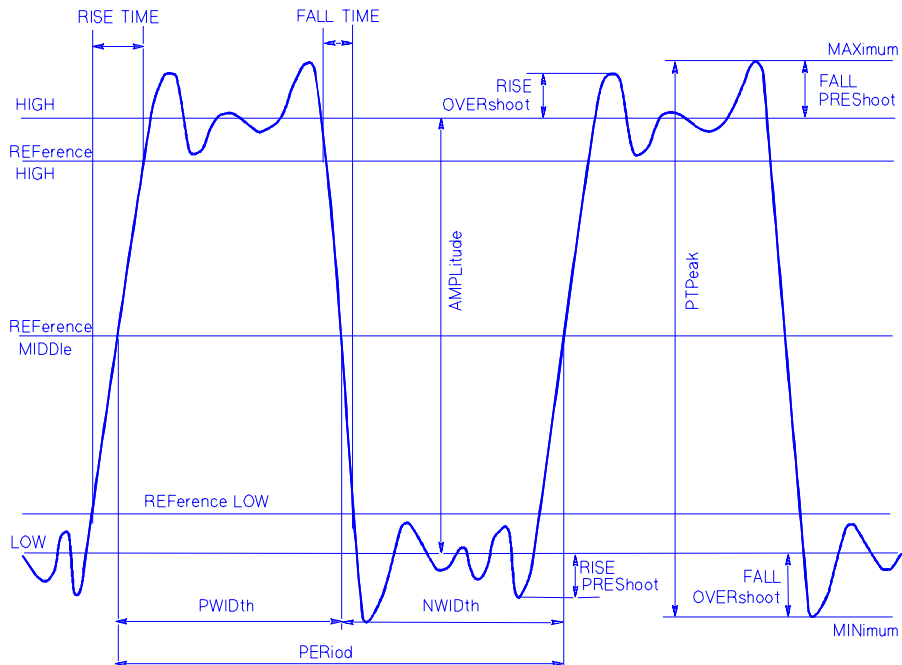


Figure 3-2 Pulse Measurement Terminology

3.8.2.3 :LOW

This command specifies the LOW signal level. LOW is defined as the less positive of the BASE and TOP signal levels as defined in IEEE Std 194 sections 3.2.1 and 3.2.2.

3.8.2.4 :HIGH

The HIGH signal level. HIGH is defined as the more positive of the BASE and TOP signal levels as defined in IEEE Std 194 sections 3.2.1 and 3.2.2.

3.8.2.5 :RISE

This node specifies measurements on a rising edge of a waveform. This node has no associated parameters. This node specifies the first rising edge of the waveform.

3.8.2.5.1 :TIME [<low reference>,<high reference>,<expected_value>,<resolution>]]]

The time interval during which the instantaneous amplitude of a pulse increases from the low reference to high reference of the normal pulse amplitude.

This function is an alias to RTIME. If RISE:TIME is implemented, RTIME shall also be implemented. It is recommended that RISE:TIME also be implemented.

Default low and high references are 10% and 90% of normal high-signal level. These defaults may be changed via the parameters. The reference parameters have a default unit of percent. An implementation may allow the references to be entered in absolute sensor units. For example, an implementation may choose to allow the following syntax:

```
MEASure:VOLTage:RISE:TIME 0.2V,4.5V,.005S
```

to measure the 0.2 to 4.5 Volt rise time of a voltage signal.

3.8.2.5.2 :OVERshoot

The difference between the HIGH and the peak amplitude to which the instantaneous pulse waveform initially rises. This function uses the first edge of the waveform, expressed as a percentage of the waveform amplitude.

3.8.2.5.3 :PREShoot

The difference between the LOW signal level and the negative peak signal value to which the waveform initially falls expressed as a percentage of waveform amplitude.

3.8.2.6 :RTIME [<low reference>,<high reference>,<expected_value>,<resolution>]]]

The time interval during which the instantaneous amplitude of a pulse increases from the low reference to high reference of the normal pulse amplitude.

RISE:TIME is an alias to this function. If RISE:TIME is implemented, RTIME shall also be implemented.

Default low and high references are 10% and 90% of normal high-signal level. These defaults may be changed via the parameters. The reference parameters have a default unit of percent. An implementation may allow the references to be entered in absolute sensor units. For example, an implementation may choose to allow the following syntax:

```
MEASure:VOLTage:RISE:TIME 0.2V,4.5V,.005S
```

to measure the 0.2 to 4.5 Volt rise time of a voltage signal.

3.8.2.7 :FALL

This node specifies measurements on a falling edge of a waveform. This node has no associated parameters. This node assumes the first falling edge of the waveform.

3.8.2.7.1 :TIME [<low reference>,<high reference>[,<expected_value>[,<resolution>]]]

The time interval during which the instantaneous amplitude of a pulse decreases from the high reference to low reference of the normal pulse amplitude.

This function is an alias to FTIME. If FALL:TIME is implemented, FTIME shall also be implemented. It is recommended that FALL:TIME also be implemented.

Default low and high references are 10% and 90% of normal high-signal level. These defaults may be changed via the parameters. The reference parameters have a default unit of percent. An implementation may allow the references to be entered in absolute sensor units. For example, an implementation may choose to allow the following syntax:

```
MEASure:VOLTage:FALL:TIME 0.2V,4.5V,.005S
```

to measure the 4.5 to 0.2 Volt fall time of a voltage signal.

3.8.2.7.2 :OVERshoot

The difference between the LOW and the negative peak amplitude to which the instantaneous pulse waveform initially falls, expressed as a percentage of the waveform amplitude.

3.8.2.7.3 :PREShoot

The difference between the HIGH signal level and the positive peak signal value to which the waveform initially rises expressed as a percentage of waveform amplitude.

3.8.2.8 :FTIME [<low reference>,<high reference>[,<expected_value>[,<resolution>]]]

The time interval during which the instantaneous amplitude of a pulse decreases from the high reference to low reference of the normal pulse amplitude.

FALL:TIME is an alias to this function. If FALL:TIME is implemented, FTIME shall also be implemented.

Default low and high references are 10% and 90% of normal high-signal level. These defaults may be changed via the parameters. The reference parameters have a default unit of percent. An implementation may allow the references to be entered in absolute sensor units. For example, an implementation may choose to allow the following syntax:

```
MEASure:VOLTage:FALL:TIME 0.2V,4.5V,.005S
```

to measure the 4.5 to 0.2 Volt fall time of a voltage signal.

3.8.2.9 :PWIDth [<reference>]

The positive width expressed in seconds from the first rising edge reference to the next falling edge reference. The same reference is used for the rising and falling edges. The

default reference is defined as midway between HIGH and LOW levels. Other positive references may be specified by the parameter.

3.8.2.10 :NWIDth [<reference>]

The negative width expressed in seconds from the first falling edge reference to the next rising edge reference. The same reference is used for the rising and falling edges. The default reference is midway between HIGH and LOW levels. Other references may be specified by the parameter.

3.8.2.11 :PDUTYcycle!DCYCLE [<reference>]

Positive dutycycle. The ratio of PWIDth to PERiod. The reference specifies the point on the pulse where the duty cycle is determined. This parameter defaults to 50%.

3.8.2.12 :NDUTYcycle [<reference>]

Negative dutycycle. The ratio of NWIDth to PERiod. The reference specifies the point on the pulse where the duty cycle is determined. This parameter defaults to 50%.

3.8.2.13 :TMAXimum Layer:Time Domain Waveform Measurements

The time at which the first occurrence of the maximum voltage occurs.

3.8.2.14 :TMINimum

The time at which the first occurrence of the minimum voltage occurs.

3.8.2.15 :MINimum

The MINimum command calculates the minimum value of the waveform.

3.8.2.16 :MAXimum

The MAXimum command calculates the maximum value of the waveform.

3.8.2.17 :PTPeak

This command measures the Peak To Peak value of the signal which is (MAXimum - MINimum).

4 CALCulate Subsystem

The CALCulate subsystem exists to perform postacquisition data processing. Functions in the SENSE subsystem are related to data acquisition, while the CALCulate subsystem operates on the data acquired by a SENSE function.

The CALCulate subsystem is logically between the SENSE subsystem and data output to either the bus or the display. When a measurement is triggered by a MEASure command, an INITiate command, or meeting the prevailing TRIGger conditions, the SENSE subsystem collects data. This data is transformed by CALCulate, as specified, and then passed on to the selected output. In effect, the collection of new data triggers the CALCulate subsystem. The CALCulate subsystem may also be directed by command to perform a transform, making it possible to change the configuration of CALCulate and consequently derive a different set of results from the same SENSEd data set without reacquiring sense data.

The CALCulate subsystem consists of a number of independent subsystems. Each of the subsystems is a sub-block of the CALCulate block. The data flows through the sub-blocks in a serial fashion, through the first sub-block, then onto the second sub-block and so on. The manner in which these sub-blocks are arranged is specified in the PATH command. It is permissible for a CALCulate block to have more than one instance of any of the sub-blocks. Instances of the same sub-block are differentiated by a numeric suffix. For example, two independent filters would exist as the :CALCulate:FILTer1 and the :CALCulate:FILTer2 subsystem sub-blocks.

Note: The notation X_n^* indicates the complex conjugate of a number. So:

$$XX^* = \text{MAG}^2(X) = |X|^2$$

KEYWORD	PARAMETER FORM	NOTES
CALCulate		
:AVERage		1992
:CLEar		[no query] 1992
:COUNT	<numeric_value>	1992
:AUTO	<Boolean> ONCE	1992
[:STATe]	<Boolean>	1992
:TCONtrol	EXPonential MOVing NORMal REPeat	1992
:TYPE	COMPLex ENVELOpe MAXimum MINimum RMS SCALar	1992
:CLIMits		1991
:FAIL?		[query only] 1991
:FLIMits		1991
[:DATA]?		[query only] 1991
:POINts?		[query only] 1991
:DATA?		[query only] 1995
:DATA		1995
:PREamble?		[query only] 1994
:DERivative		1993
:STATe	<Boolean>	1993

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
:POINTs	<numeric_value>	1993
:FEED	<data_handle>	1991
:FILTER		
[:GATE]		
:TIME		
:STATe	<Boolean>	1992
[:TYPE]	BPASs NOTCh	
:START	<numeric_value>	
:STOP	<numeric_value>	
:SPAN	<numeric_value>	
:CENTER	<numeric_value>	
:POINTs	<numeric_value>	
:AUTO	<Boolean> ONCE	
:WINDow	RECTangular UNIFORM FLATtop HAMMING HANNing KBESSel FORCE EXPonential	
:KBESSel	<numeric_value>	
:EXPonential	<numeric_value>	
:FORCE	<numeric_value>	
:FREQuency		
:STATe	<Boolean>	1992
[:TYPE]	BPASs NOTCh	
:START	<numeric_value>	
:STOP	<numeric_value>	
:SPAN	<numeric_value>	
:CENTER	<numeric_value>	
:POINTs	<numeric_value>	
:AUTO	<Boolean> ONCE	
:WINDow	RECTangular UNIFORM FLATtop HAMMING HANNing KBESSel FORCE EXPonential	
:KBESSel	<numeric_value>	
:EXPonential	<numeric_value>	
:FORCE	<numeric_value>	
:FORMat	NONE MLINear MLOGarithmic PHASe REAL IMAGinary SWR GDELay COMPLex NYQuist NICHols POLar UPHase	1991 1991
:UPHase		1991
:CREference	<numeric_value>	1991
:PREference	<numeric_value>	1991
:GDAPerture		
:SPAN	<numeric_value>	
:APERture	<numeric_value>	

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
:IMMediate		
:AUTO	<Boolean>	1992
:INTEgral		1993
:STATe	<Boolean>	1993
:TYPE	SCALar MOVing	1993
:LIMit		1991
:STATe	<Boolean>	1991,2
:CONTRol		1991
[:DATA]	<numeric_value>{,<numeric_value>}	1991
:POINts?		[query only] 1991
:UPPer		1991
[:DATA]	<numeric_value>{,<numeric_value>}	1991
:POINts?		[query only] 1991
:STATe	<Boolean>	1991
:LOWer		1991
[:DATA]	<numeric_value>{,<numeric_value>}	1991
:POINts?		[query only] 1991
:STATe	<Boolean>	1991
:FAIL?		[query only] 1991
:FCOunt?		[query only] 1991
:REPort		1991
[:DATA]?		[query only] 1991
:POINts?		[query only] 1991
:CLEar		1991
:AUTO	<Boolean>	1991
[:IMMediate]		[no query] 1991
:INTERpolate	<Boolean>	1991
:MATH		
[:EXPRession]	<numeric_expression>	1991
:CATalog?		<query only> 1993
[:DEFine]	<numeric_expression>	1993
:DELeTe		1993
[:SELeCted]	<expression_name>	1993
:ALL		1993
:NAME	<expression_name>	1993
:STATe	<Boolean>	1992
:SMOothing		
[:STATe]	<Boolean>	1992
:APERture	<numeric_value>	
:POINts	<numeric_value>	
:STATe	<Boolean>	1992
:TRANSform		
:HISTogram		1993
:COUNt	<numeric_value>	1993

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
:ORDinate	RATio PERCent PCT COUNt	1993
:POINts	<numeric_value>	1993
:RANGe		1993
:AUTO	<Boolean>	1993
:STATe	<Boolean>	1993
:TIME		
:STATe	<Boolean>	1992
[:TYPE]	LPASs BPASs	
:STIMulus	STEP IMPulse	
:START	<numeric_value>	
:STOP	<numeric_value>	
:SPAN	<numeric_value>	
:CENTer	<numeric_value>	
:POINts	<numeric_value>	
:AUTO	<Boolean> ONCE	
:WINDow	RECTangular UNIFORM FLATtop HAMMING HANNing FORCE EXPonential	
:KBESsel	<numeric_value>	
:EXPonential	<numeric_value>	
:FORCE	<numeric_value>	
:DISTance		
:STATe	<Boolean>	1992
[:TYPE]	LPASs BPASs	
:STIMulus	STEP IMPulse	
:START	<numeric_value>	
:STOP	<numeric_value>	
:SPAN	<numeric_value>	
:CENTer	<numeric_value>	
:POINts	<numeric_value>	
:AUTO	<Boolean> ONCE	
:WINDow	RECTangular UNIFORM FLATtop HAMMING HANNing FORCE EXPonential	
:KBESsel	<numeric_value>	
:EXPonential	<numeric_value>	
:FORCE	<numeric_value>	
:FREQuency		
:STATe	<Boolean>	1992
[:TYPE]	LPASs BPASs	
:STIMulus	STEP IMPulse	
:START	<numeric_value>	
:STOP	<numeric_value>	
:SPAN	<numeric_value>	

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
:CENTer	<numeric_value>	
:POINTs	<numeric_value>	
:AUTO	<Boolean> ONCE	
:WINDow	RECTangular UNIFORM FLATtop HAMMING HANNing KBESsel FORCe EXPonential	
:KBESsel	<numeric_value>	
:EXPonential	<numeric_value>	
:FORCe	<numeric_value>	
:PATH	(MATH TRANSform FILTer SMOothing FORMat LIMIT AVERage) {,(MATH TRANSform FILTer SMOothing FORMat LIMIT AVERage)}	[query only] 1991 1992 1991 1992

4.1 :AVERage subsystem

CALCulate:AVERage

This subsystem provides an analytical post-processing function. The AVERage subsystem combines successive measurements to produce a new composite result. The new result has the same number of points and control axis as the original measurement.

A :COUNT command is provided to specify the number of measurements to combine. When :COUNT averages are reached, the TerminalCONTROL switch specifies the operation of the average subsystem. The TerminalCONTROL switch can be configured to continue replacing old values with new values (MOVing), to exponentially weight new measurements (EXponential), to automatically repeat (REpeat), or to continue adding new measurements in the NORMal fashion, as specified by :TYPE.

The average operation differs from CALC:SMOothing in the fact that AVERage combines the j th point of each measurement with the j th point of preceeding measurements, whereas in SMOothing the j th point is processed with adjacent points ($\dots, j-2, j-1, j, j+1, j+2, \dots$) of the current measurement set.

A typical device action for CALC:AVERage

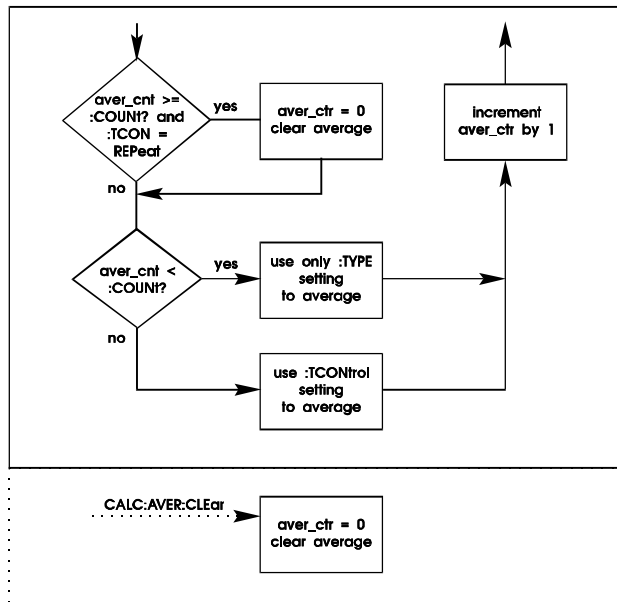


Figure 4-1 AVERage device action

4.1.1 :CLEar

CALCulate:AVERage:CLEar

This command causes the average data to be cleared and the average counter reset to zero.

4.1.2 :COUNT <numeric_value>

CALCulate:AVERage:COUNT

Specifies the number of measurements to combine using the :TYPE setting. After :COUNT measurements have been averaged, the operation of the AVERage subsystem is controlled by the setting of the :TCONtrol node.

The count for array based results is independent of the number of points in the result array. For example, an instrument which normally provides a measurement result which is a 401 point array would specify an average count of two to average two results, not 802.

When averaging is ON, some devices may automatically set :COUNT values in the TRIGger subsystem based on the AVER:COUNT value, such that the TRIGger subsystem provides enough triggers for the average.

At *RST, this value is device dependent.

4.1.2.1 :AUTO <Boolean> | ONCE

CALCulate:AVERage:COUNT:AUTO

AUTO ON causes the device to select a value for :COUNT which is appropriate for the current measurement. Selecting AUTO ONCE will have the effect of setting AUTO ON and then OFF.

At *RST, this value is set to OFF.

4.1.3 [:STATe] <Boolean>

CALCulate:AVERage:STATe

The STATe command is used to turn averaging ON and OFF.

At *RST, this value is set to OFF.

4.1.4 :TCONtrol EXPonential | MOVing | NORMal | REPeat

CALCulate:AVERage:TCONtrol

TerminalCONtrol specifies the action of the AVERage subsystem when more than AVERage:COUNT measurement results are generated.

The parameter has the following meanings:

EXPonential Continue the average with an exponential weighting applied to old values. The additional averages are weighted using:

For :TYPE SCALAr when X_n is real and :TYPE COMPLEX when X_n is either real or complex

$$AVG_n = \frac{1}{k} X_n + \frac{k-1}{k} AVG_{n-1}$$

For :TYPE SCALAr when X_n is complex

$$AVG_n = \frac{1}{k} MAG(X_n) + \frac{k-1}{k} AVG_{n-1}$$

For :TYPE RMS when X_n is real

$$AVG_n = \sqrt{\frac{1}{k}X_n^2 + \frac{k-1}{k}AVG_{n-1}^2}$$

For :TYPE RMS when X_n is complex

$$AVG_n = \sqrt{\frac{1}{k}X_nX_n^* + \frac{k-1}{k}AVG_{n-1}^2}$$

The exponential factor k is equal to the AVERage:COUNT value. Some instruments may use an approximation to true exponential.

Operation with EXponential :TCON is undefined with a :TYPE setting of MINimum, MAXimum, or ENvelope.

MOVing	As new data is added the oldest data is discarded.
REPeat	Clear average data and counter and restart the average process.
NORMal	Additional averages continue to be accumulated according to the :TYPE selection.

At *RST, this value is device dependent.

4.1.5 :TYPE COMplex | ENvelope | MAXimum | MINimum | RMS | SCALar CALCulate:AVERage:TYPE

This command selects the type of averaging, as follows:

COMPLex The points in the summation are treated as real/imaginary pairs.

$$AVG(n) = \frac{1}{n} \sum_{i=1}^n X_i$$

ENVELOpe Both the MIN and MAX values are retained.

If X_n is real

MAXimum $AVG(n) = MAX(X_1 \dots X_n)$

MINimum $AVG(n) = MIN(X_1 \dots X_n)$

RMS $AVG(n) = \sqrt{\frac{1}{n} \sum_{i=1}^n X_i^2}$

SCALar The scalar magnitude of each point is averaged.

$$AVG(n) = \frac{1}{n} \sum_{i=1}^n X_i$$

If X_n is complex

MAXimum $AVG(n) = MAX(MAG(X_1) \dots MAG(X_n))$

MINimum $AVG(n) = \text{MIN}(\text{MAG}(X_1) \dots \text{MAG}(X_n))$

RMS $AVG(n) = \sqrt{\frac{1}{n} \sum_{i=1}^n X_i X_i^*}$

SCALar The scalar magnitude of each point is averaged.

$$AVG(n) = \frac{1}{n} \sum_{i=1}^n |X_i|$$

At *RST, this value is device dependent.

4.2 :CLIMits

CALCulate:CLIMits

This node defines commands to obtain Composite LIMit results within the specified CALCulate. It is permissible for CALCulate to contain two or more LIMit tests. These tests may be active simultaneously. The commands under CLIMits provide a mechanism to determine a summary of the PASS/FAIL information, and also a way to directly ascertain which particular LIMit tests are failing.

4.2.1 :FAIL?

CALCulate:CLIMits:FAIL?

FAIL is a summary of all the individual LIMit:FAILs. If any of the individual LIMit:FAILs have a non-zero value, indicating a failure, the composite FAIL query shall return a 1. If there are no failures then a 0 shall be returned. When all the failed LIMits have been CLEared, then the composite FAIL shall also be cleared.

4.2.2 :FLIMits

CALCulate:CLIMits:FLIMits

This node provides the information to determine the Failed LIMits, that is, identify how many and which LIMit tests have failed. When all the failed LIMits have been CLEared, then FLIMits shall also be cleared.

4.2.2.1 [:DATA]?

CALCulate:CLIMits:FLIMits:DATA?

This query returns a number or list of numbers, each number corresponds to an instance of the LIMit subsystem within the CALCulate block. For example, if CALC:LIMit1, CALC:LIMit2 and CALC:LIMit3 existed, the query shall return 1,3, if LIMit1 and LIMit3 had failed. If there are no failures then NAN shall be returned.

4.2.2.2 :POINTs?

CALCulate:CLIMits:FLIMits:POINTs?

This query shall return the number of failed LIMits in DATA. If there are no failures then a 0 shall be returned.

4.3 :DATA?

CALCulate:DATA?

Provides access to the result of the CALCulate subsystem. The query form outputs the present CALCulate results.

This command is query-only and therefore has no *RST condition.

4.3.1 :PREamble?

CALCulate:DATA:PREamble?

PREamble? returns the preamble information supporting the DATA(CURVe(VALues)) but omits the following data:

- DIF blocks of waveform, measurement, and delta.
- DIF curve block keywords VALues and CSUM and their parameters.

The DIF difid block must contain the keyword SCOPE with the parameter of PREamble.

While the PREamble? query is designed to efficiently transmit data relating directly to the curve, the various notes and identification parameters may be sent if absolutely required.

4.4 :DERivative

CALCulate:DERivative

This subsystem calculates the derivative of the points in a data set. Each point in the differentiated data set is the derivative at the corresponding point in the original data set (note this requires some approximation at the end points). The units of the result of this process is the quotient of the currently selected amplitude units of the differentiated data and the units of the X-axis.

4.4.1 :STaTe <Boolean>

CALCulate:DERivative:STaTe

Determines whether the derivative function is enabled.

At *RST this function is OFF

4.4.2 :POINts <numeric_value>

CALCulate:DERivative:POINts

Specifies the number of points provided to the differential algorithm. They will be centered about the current data point.

At *RST this value is device dependent.

4.5 :FEED <data_handle>

CALCulate:FEED

Sets or queries the data flow to be fed into the CALCulate block.

At *RST, FEED shall be set to a device dependent value.

4.6 :FILTer

CALCulate:FILTer

This subsystem defines a filtering process to be performed on the SENSE data. A filter can operate in domains, which may or may not be the domain of the acquisition data.

4.6.1 [:GATE]

CALCulate:FILTer:GATE

Specifies the or modification of the data set according to the selected parameters in a specific domain.

4.6.1.1 :TIME

CALCulate:FILTer:GATE:TIME

This filter selection will modify the acquisition data to remove or modify information in a specified time region. Units of time are seconds.

4.6.1.1.1 :STATE <Boolean>

CALCulate:FILTer:GATE:TIME:STATE

Determines whether the time filter is enabled.

At *RST, this function is OFF.

4.6.1.1.2 [:TYPE] BPASS|NOTCh

CALCulate:FILTer:GATE:TIME:TYPE

Specifies what occurs to the data in the specific time region. The parameters have the following meanings:

- BPASS — Pass all information in specified time region and reject all else.
- NOTCh — Reject all information in specified time region and pass all else.

At *RST, this value is device-dependent.

4.6.1.1.3 :START <numeric_value>

CALCulate:FILTer:GATE:TIME:START

Specifies the start time of the filter. Range is instrument- and setup-dependent.

The couplings between START, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, START is set to MIN.

4.6.1.1.4 :STOP <numeric_value>

CALCulate:FILTer:GATE:TIME:STOP

Specifies the stop time of the filter. Range is instrument- and setup-dependent.

The couplings between START, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, STOP is set to MIN.

4.6.1.1.5 :SPAN <numeric_value>

CALCulate:FILTer:GATE:TIME:SPAN

Specifies the time span of the filter. Range is instrument- and setup-dependent.

The couplings between START, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, SPAN is set to MIN.

4.6.1.1.6 :CENTER <numeric_value>

CALCulate:FILTer:GATE:TIME:Center

Specifies the center time of the filter. Range is instrument- and setup-dependent.

The couplings between START, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, CENTER is set to MIN.

4.6.1.1.7 :POINTS <numeric_value>

CALCulate:FILTer:GATE:TIME:POINTS

Specifies the number of points output by the filter subsystem.

At *RST, this value is device-dependent.

4.6.1.1.7.1 :AUTO <Boolean>IONCE

CALCulate:FILTer:GATE:TIME:POINTS:AUTO

Setting AUTO ON allows POINTs to be set by device-dependent parameters (for example, the size of incoming SENSE data).

At *RST, AUTO is ON.

4.6.1.1.8 :WINDOW RECTangular|UNIFORM|FLATtop|HAMMING IHANNing|KBESSel|FORCe|EXPonential

CALCulate:FILTer:GATE:TIME:WINDOW

Specifies the type and parameter of data windowing (shaping) done prior to the filter.

Individual windowing algorithms are defined in “*On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform*,” F. J. Harris, Proc. of the IEEE, Vol. 66-1, January 1978, pp 51-83.

At *RST, this value is device-dependent.

4.6.1.1.9 :KBESSel <numeric_value>

CALCulate:FILTer:GATE:TIME:KBESSel

The Kaiser BESSel command sets the parametric window parameter for the Kaiser Bessel window.

At *RST, this value is device-dependent.

4.6.1.1.10 :EXPonential <numeric_value>

CALCulate:FILTer:GATE:TIME:EXPonential

Enters the exponential decay time constant which characterizes the EXPonential window.

At *RST, this value is device-dependent.

4.6.1.1.11 :FORCE <numeric_value>

CALCulate:FILTer:GATE:TIME:FORCE

Enters the time value parameter corresponding to the width of the gated portion of the input time record for FORCE windows.

At *RST, this value is device-dependent.

4.6.1.2 :FREQUENCY

CALCulate:FILTer:GATE:FREQUENCY

This filter selection will modify the acquisition data to remove or modify information in a specified frequency region.

Units are Hertz.

4.6.1.2.1 :STATE <Boolean>

CALCulate:FILTer:GATE:FREQUENCY:STATE

Determines whether the frequency filter is enabled.

At *RST, this function is OFF.

4.6.1.2.2 [:TYPE] BPASS|NOTCh

CALCulate:FILTer:GATE:FREQUENCY:TYPE

Specifies what occurs to the data in the specific frequency region. The parameters have the following meanings:

- BPASS — Pass all information in specified time region and reject all else.
- NOTCh — Reject all information in specified frequency region and pass all else.

At *RST, this value is device-dependent.

4.6.1.2.3 :START <numeric_value>

CALCulate:FILTer:GATE:FREQUENCY:START

Specifies the start frequency of the filter. Range is instrument- and setup-dependent.

The couplings between START, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, START is set to MIN.

4.6.1.2.4 :STOP <numeric_value>

CALCulate:FILTer:GATE:FREQUENCY:STOP

Specifies the stop frequency of the filter. Range is instrument- and setup-dependent.

The couplings between START, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, STOP is set to MIN.

4.6.1.2.5 :SPAN <numeric_value>

CALCulate:FILTer:GATE:FREQuency:SPAN

Specifies the frequency span of the filter. Range is instrument- and setup-dependent.

The couplings between START, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, SPAN is set to MIN.

4.6.1.2.6 :CENTER <numeric_value>

CALCulate:FILTer:GATE:FREQuency:CENTer

Specifies the center frequency of the filter. Range is instrument- and setup-dependent.

The couplings between START, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, CENTER is set to MIN.

4.6.1.2.7 :POINTS <numeric_value>

CALCulate:FILTer:GATE:FREQuency:POINts

Specifies the number of points output by the filter subsystem.

At *RST, this value is device-dependent.

4.6.1.2.7.1 :AUTO <Boolean>IONCE

CALCulate:FILTer:GATE:FREQuency:POINts:AUTO

Setting AUTO ON allows POINTs to be set by device-dependent parameters (for example, the size of incoming SENSE data).

At *RST, AUTO is ON.

4.6.1.2.8 :WINDOW RECTangular|UNIFORM|FLATtop|HAMMING |HANNING|KBESsel|FORCe|EXPonential

CALCulate:FILTer:GATE:FREQuency:WINDow

Specifies the type and parameter of data windowing (shaping) done prior to the filter.

Individual windowing algorithms are defined in “*On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform*,” F. J. Harris, Proc of the IEEE, Vol. 66-1, January 1978, pp 51-83.

At *RST, this value is device-dependent.

4.6.1.2.9 :KBESsel <numeric_value>

CALCulate:FILTer:GATE:FREQuency:KBESsel

The Kaiser BESsel command sets the parametric window parameter for the Kaiser Bessel window.

At *RST, this value is device-dependent.

4.6.1.2.10 :EXPonential <numeric_value>

CALCulate:FILTer:GATE:FREQuency:EXPonential

Enters the exponential decay time constant which characterizes the EXPonential window.

At *RST, this value is device-dependent.

4.6.1.2.11 :FORCe <numeric_value>

CALCulate:FILTer:GATE:FREQuency:FORCe

Enters the time value parameter corresponding to the width of the gated portion of the input time record for FORCe windows.

At *RST, this value is device-dependent.

4.7 :FORMat NONEIMLINearIMLOGarithmicIPHASeIREAL IIMAGinaryISWRIGDELay ICOMplexINYQuistINICHolsIPOLar IUPHase

CALCulate:FORMat

Determines a simple post-processing of SENSE data. This subsystem is limited to simple point-by-point post processing. SENSE data description is $x + jy$, and if no y data, assume $y=0$. Regardless of the unprocessed data form, the data shall be interpreted as $x + jy$ for the purposes of defining the formats in the table below.

NYQuist is an optional alias for COMPLEX. If NYQuist is implemented, COMPLEX shall also be implemented.

NAME	EQUATION	TYPE	
NONE	NONE	Unprocessed data	
MLINear	$\sqrt{x^2 + y^2}$	(scalar)	
MLOGarithmic	$10 \cdot \log(x^2 + y^2)$	(scalar, LOG is base 10)	
PHASe	$\arctan\left(\frac{y}{x}\right)$	(scalar in current angle units, ± 180 degrees)	
IMAGinary	y	(linear scalar)	
REAL	x	(linear scalar)	
COMPLEX NYQuist	pair (x,y)	(linear scalar)	1991
POLar	pair (r1, r2)		1991
	$r1 = \sqrt{x^2 + y^2}$		1991
	$r2 = \arctan(y/x)$		1991
NICHols	pair (r1, r2)		1991
	$r1 = 10 \cdot \log(x^2 + y^2)$		1991
	$r2 = \arctan(y/x)$		1991

1999 SCPI Command Reference

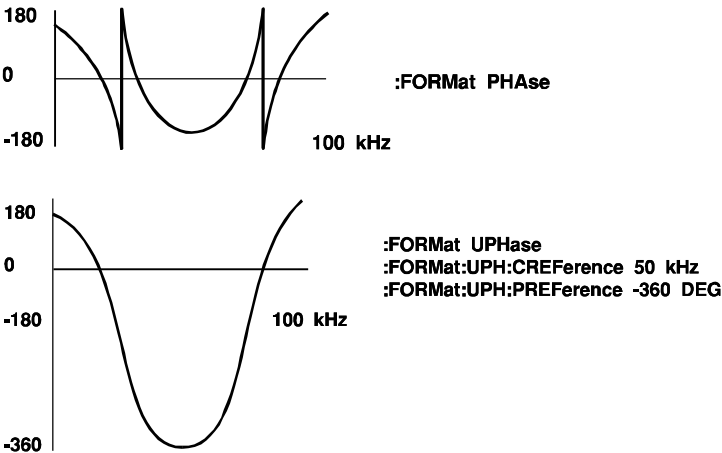
NAME	EQUATION	TYPE
SWR	$\frac{(1 + r)}{(1 - r)}$ WHERE $r = \text{MagLIN}(data)$	(scalar)
GDElay	$\frac{-d(\text{phase})}{360 * d(\text{frequency})}$	(Scalar; units are seconds. Phase specified in current angle units. If current angle units are radians, replace the constant 360 with $2 * \text{PI}$. The $d(\text{frequency})$ term is replaced with <code>CALCulate:GDAPerture:APERture</code> in practice.)
UPHase	<code>UNWRAP(arctan(y/x))</code>	(scalar in current angle units) 1991

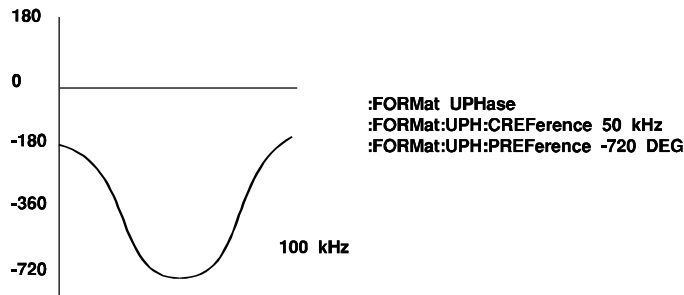
At *RST, this value is device-dependent.

4.7.1 :UPHase

`CALCulate:FORMat:UPHase`

The UPHase subsystem is used to specify how unwrapped phase is computed. A reference position is specified which is used as an estimate of what the unwrapped phase of calculated result should be. The calculated result is constrained to be within +/- 180 degrees of the reference position. Other points are calculated to produce a result which is compatible with the above constraint and does not “wrap” when cardinal points such as +/- 180 degrees are crossed. The reference position would typically be a well known point in a measurement such as the center of a band pass filter.





4.7.1.1 **:CREference <numeric_value>**

CALCulate:FORMat:UPHase:CREference

The Control REference command designates the reference position on the independent dimension, called the control axis (typically the X axis) where the phase unwrap reference point is located. This value in combination with the :PREF value identifies the phase unwrap reference point.

At *RST, this value is device dependent.

4.7.1.2 **:PREference <numeric value>**

CALCulate:FORMat:UPHase:PREference

The Phase REference command designates the phase of the phase unwrap reference point. This value in combination with the :CREference value identifies the reference point. The CALC result is constrained to be within +/- 180 degrees of this value at the point corresponding to the :CREference position.

At *RST, this value is device dependent.

4.8 **:GDAPerture**

CALCulate:GDAPerture

Group Delay Aperture is defined as the negative of the derivative of phase with respect to frequency. The group delay format requires specifying an aperture over which the measurement is made. Since the frequency span of the measurement is known, the frequency step over the entire span can be thought of as $d(\text{freq})$. This function permits the setting of that aperture. APERTure is tightly coupled to SPAN by the equation:

$$\text{APERTure} = \text{GDAPerture} \cdot \text{SPAN} / (\text{SENSe}|\text{SOURce}) : \text{FREQuency} : \text{SPAN}$$

4.8.1 **:SPAN <numeric_value>**

CALCulate:GDAPerture:SPAN

Specifies the aperture in Hertz. Minimum value is instrument-dependent.

At *RST, this function is set to MIN.

4.8.2 **:APERture <numeric_value>**

CALCulate:GDAPerture:APERture

Specifies the aperture as a ratio of desired aperture span/measured frequency span.

At *RST, this value is set to MIN.

4.9 :IMMediate

CALCulate:IMMediate

Causes the CALCulate subsystem to reprocess existing SENSE data without reacquiring the data. This permits the user to change a CALCulate subsystem, setting and output new results without initiating a data acquisition.

Note that CALC:IMMediate? is semantically equivalent to CALC:IMM;DATA?. The query form outputs the results of the new calculation.

This command describes an event and therefore has no associated *RST condition.

4

4.9.1 :AUTO <Boolean>

CALCulate:IMMediate:AUTO

The AUTO command is used to specify if the CALCulate subsystem continually transforms the data (AUTO ON) whenever any changes are made which would affect the CALCulate subsystem output. With CALC:IMM:AUTO set to OFF, CALCulate only produces new data when data is received or when the CALC:IMMediate command is received. Once CALC:IMM:AUTO is set to ON, the CALCulate Subsystem shall produce results when any CALC subsystem command is processed, even when new data is not being received. This allows the user to make configuration changes in the CALCulate subsystem and immediately have new CALC:DATA results on the same sensed data.

At *RST, this value is set to OFF.

4.10 :INTegral

CALCulate:INTegral

This subsystem provides a point to point integration of a data set. The result is an estimate of the integral of the input. The units of the result of this process is the product of the currently selected amplitude units of the integrated data and the units of the X- axis.

4.10.1 :STATe <Boolean>

CALCulate:INTegral:STATe

Determines whether the integrate function is enabled.

At *RST this function is OFF

4.10.2 :TYPE SCALar | MOVing

CALCulate:INTegral:TYPE

Specifies whether the result is a single value (SCALar) or a set of data points (MOVing). If SCALar is selected the result is an estimate of the definite integral over the entire data set. If MOVing is selected each point in the integrated data set is the integral up to the corresponding point in the original data set.

At *RST this function is set to a device dependent value.

4.11 :LIMit

CALCulate:LIMit

The LIMit node collects together the commands associated with controlling and getting reports from a single LIMit test. The LIMit test may defined as an UPPER limit, a LOWER

limit, or both. A LIMit test may be defined as being scalar, that is UPPer or LOWEr are provided a single value each.

Alternatively, UPPer or LOWEr may be defined as a series of points (a vector) representing a waveform limit. For such a vector limit test, each point in the series (vector) must be qualified by where it occurs with respect to the control (typically the X) axis. The CONTrol node is used to define the points along the control axis. UPPer and LOWEr represent the dependent (typically the Y) axis, and are given values corresponding to these control axis points. That is, the first value for UPPer and LOWEr corresponds to the first value of CONTrol, the second value for UPPer and LOWEr corresponds to the second value of CONTrol, and so on. UPPer, LOWEr and CONTrol shall contain the same number of points, or an error shall be generated when data processing commences. For scalar limit tests, as described earlier, the CONTrol subsystem is not used.

If the data and the limit are scalar (single values), then LIMit shall produce results for each piece of scalar data tested. If the data and the limit are both vectors, then LIMit shall produce results for each vector data tested. If the limit is scalar and the data is a vector, then LIMit shall perform a the limit test by treating the scalar limit as an equivalent vector (all points having the same value).

4.11.1 :STATe <Boolean>

CALCulate:LIMit:STATe

Is used to set or query if the LIMit test is active or not.

At *RST, this is set to OFF.

4.11.2 :CONTrol

CALCulate:LIMit:CONTrol

This node collects the commands and queries used to set up the control axis of the limit data, where the limit test is not based on a single point.

4.11.2.1 [:DATA] <numeric_value>{,<numeric_value>}

CALCulate:LIMit:CONTrol:DATA

Sets or queries the CONTrol axis data.

4.11.2.2 :POINTs?

CALCulate:LIMit:CONTrol:POINTs?

The POINTs? query returns the number of points currently in CONTrol:DATA.

4.11.3 :UPPer

CALCulate:LIMit:UPPer

UPPer is used to define the upper limit to be used in the LIMit test. When the data is greater than the value specified in UPPer, LIMit shall report a fail. That is, when the data is equal to the limit, a fail shall not be reported.

4.11.3.1 [:DATA] <numeric_value>{,<numeric_value>}

CALCulate:LIMit:UPPer:DATA

Sets or queries the UPPer axis data.

4.11.3.2 :POINTs?

CALCulate:LIMit:UPPer:POINTs?

The POINTs? query returns the number of points currently in UPPer:DATA.

4.11.3.3 :STATe <Boolean>

CALCulate:LIMit:UPPer:STATe

Sets or queries if the individual LIMit test is enabled.

At *RST, this is set to ON.

4.11.4 :LOWer

CALCulate:LIMit:LOWer

LOWer is used to define the lower limit to be used in the LIMit test. When the data is less than the value specified in LOWer, LIMit shall report a fail. That is, when the data is equal to the limit, a fail shall not be reported.

4.11.4.1 [:DATA]<numeric_value>{,<numeric_value>}

CALCulate:LIMit:LOWer:DATA

Sets or queries the LOWer axis data.

4.11.4.2 :POINTs?

CALCulate:LIMit:LOWer:POINTs?

The POINTs? query returns the number of points currently in LOWer:DATA.

4.11.4.3 :STATe <Boolean>

CALCulate:LIMit:LOWer:STATe

Sets or queries if the individual LIMit test is enabled.

At *RST, this is set to ON.

4.11.5 :FAIL?

CALCulate:LIMit:FAIL?

This query returns a 0 or 1, to indicate if the LIMit test has failed or not; 0 represents pass and 1 represents fail.

4.11.6 :FCOunt?

CALCulate:LIMit:FCOunt?

This Fail COunt query returns the number of times that the LIMit test has failed. With a 0 indicating no failures. With vector limits and vector data, only one failure shall be recorded with each (measurement) vector that is tested, even if many points are outside the limits.

4.11.7 :REPort

CALCulate:LIMit:REPort

This node collects together commands to get a more detailed information on vector limit tests.

4.11.7.1 [:DATA]?

CALCulate:LIMit:REPort:DATA?

This query returns a number or list of numbers, each number corresponds to a value of the CONTROL variable where either the UPPer or LOWer limit test has failed. Note that: When

INTERpolate ON has been selected, the returned DATA shall correspond to the points along the CONTrol axis where a failure has occurred, which may or may not coincide with points defined for CONTrol. If there are no failures then NAN shall be returned.

4.11.7.2 :POINTs?

CALCulate:LIMit:REPort:POINTs?

This query shall return the number of points in the DATA. If there are no failures then a 0 shall be returned.

4.11.8 :CLEar

CALCulate:LIMit:CLEar

The CLEar node collects together the commands that are used to clear the information in DATA, FAIL and REPort.

4.11.8.1 :AUTO <Boolean> | ONCE

CALCulate:LIMit:CLEar:AUTO

Sets or queries if the information is to be CLEared with each INITiate command, or more precisely when the TRIGger state machine exits the “IDLE state.” Such an INITiate shall cause the information in FAIL, FCOUNT and REPort to be cleared if AUTO is set to ON. If AUTO is set to OFF, no such effect shall occur.

At *RST, AUTO shall be set to ON.

4.11.8.2 [:IMMediate]

CALCulate:LIMit:CLEar:IMMediate

IMMediate shall cause the information in FAIL, FCOUNT and REPort to be cleared IMMEDIATEly.

4.11.9 :INTERpolate <Boolean>

CALCulate:LIMit:INTERpolate

Sets or queries whether or not limit tests will be performed at points between those specified by the CONTrol variable. The interpolation shall be straight line between points, however the algorithm determining the number of samples between points is device dependent.

At *RST, it shall be OFF.

4.12 :MATH

CALCulate:MATH

This subsystem permits processing of sense data in numerical expression format. The operators are +, -, *, / and use of constants and data arrays are permitted.

4.12.1 [:EXPReSSion] <numeric_expression>

CALCulate:MATH:EXPReSSion

This subsystem controls which expression is used by the CALCulate block.

4.12.1.1 :CATalog?

CALCulate:MATH:EXPReSSion:CATalog?

The CATalog query lists all the defined expressions. Response is a list of comma-separated strings. Each string contains the name of an expression. If no expressions are currently defined, the response is a null string ("").

4.12.1.2 **[[:DEFine] <numeric_expression>**

CALCulate:MATH:EXPRession:DEFine

Defines the expression used for the math operations. The <numeric_expression> may contain any of the following as operands and they may occur in any combination:

- A <trace_name>
- The reserved keyword IMPLied
- The <data_handle> associated with another block

The reserved keyword IMPLied, refers to the IMPLied output of the previous CALCulate sub-block. The ordering of the sub-blocks is specified by the PATH command. If the MATH subsystem appears first in the PATH, then the IMPLied keyword refers to the <data_handle> specified with the FEED command.

For example, to subtract a reference trace from the SENSE data, the following command would configure the MATH appropriately.

```
CALCulate:MATH:EXPRession (IMPLied - MY-TRACE)
```

This example assumes that the FEED command has been configured to select the appropriate SENSE <data_handle>. The IMPLied CALCulate data flow can be ignored and the SENSE data may be used directly in a <numeric_expression>:

```
CALCulate:MATH:EXPRession ( "SENSE1:POWER 3" - MY-TRACE)
```

At *RST, this function is undefined.

4.12.1.3 **:DELeTe**

CALCulate:MATH:EXPRession:DELeTe

The DELeTe subsystem deletes expressions.

4.12.1.3.1 **[[:SELeCted] <expression_name>**

CALCulate:MATH:EXPRession:DELeTe:SELeCted

This command causes the selected expression to be deleted.

4.12.1.3.2 **:ALL**

CALCulate:MATH:EXPRession:DELeTe:ALL

This command causes the all expressions to be deleted.

4.12.1.4 **:NAME <expression_name>**

CALCulate:MATH:EXPRession:NAME

The NAME command defines the name of the expression to be selected. If the expression name, <expression_name>, already exists, then that existing expression shall be selected. If the expression name does not exist, then the new name shall be selected, but no expression shall be defined by this selection. <expression_name> is character data.

At *RST the value of NAME is device dependent.

4.12.2 :STATe <Boolean>

CALCulate:MATH:STATe

Determines whether math processing is done.

At *RST, this function is OFF.

4.13 :SMOothing

CALCulate:SMOothing

This subsystem specifies point-to-point smoothing of a data set. A data point in a smoothed data set is the average of adjacent points from a single original data set. Smoothing differs from averaging, in which a given point is the average of corresponding points from multiple data sets or acquisitions.

4.13.1 [:STATe] <Boolean>

CALCulate:SMOothing:STATe

Determines whether the smoothing algorithm is enabled.

At *RST, this function is OFF.

4.13.2 :APERTure <numeric_value>

CALCulate:SMOothing:APERTure

Specifies the size of the smoothing APERTure as a ratio of smoothing aperture points/trace points.

At *RST, this function is device-dependent.

4.13.3 :POINTS <numeric_value>

CALCulate:SMOothing:POINTS

Controls the number of points to be included in the running average. POINTs is coupled to APERTure by the equation:

$$POINTs = APERTure * TRACE:POINTs$$

4.14 :STATe <Boolean>

CALCulate:STATe

Controls whether postprocessing is enabled. If disabled, this subsystem is effectively transparent.

At *RST, this is set ON.

4.15 :TRANSform

CALCulate:TRANSform

TRANSform is the subsystem that permits sense data to be transformed from one domain to another. Examples are from frequency domain to time domain or time domain to frequency domain. This is different from a SENSE:FUNC selection of domain because in SENSE, the data is actually collected in the selected domain, while in CALCulation:TRANSform, the data is computed, for example, by use of Fourier Transform.

4.15.1 :HISTogram

CALCulate:TRANSform:HISTogram

This subsystem transforms the data into a histogram. The histogram provides an amplitude distribution of the incoming signal. The points in a histogram specify the ratio between the number number of times a data point is within a particular amplitude belt and the total number of incoming data points that participate in the transformation. The ratio is expressed as a percentage.

The resolution of the histogram is specified by the number of points of the output signal. It determines the amplitude belt wherein the data points are counted, using the equation:

amplitude belt = amplitude range / histogram points

The units of the amplitude range are the currently selected amplitude units of the data to be transformed.

4.15.1.1 :COUNT <numeric_value>

CALCulate:TRANSform:HISTogram:COUNT

This specifies the number of measurements to include in the histogram.

For scalar measurements, COUNT is the number of incoming data points that are used to create the histogram.

The COUNT for array based results is independent of the number of points in the result array. For example, an instrument which normally provides a measurement result which is a 401 point array would specify a histogram count of two to create a histogram based on 802 data points.

At *RST this value is device dependent

4.15.1.2 :ORDinate RATio | PERCent | PCT | COUNT

CALCulate:TRANSform:HISTogram:ORDinate

When ORDinate is set to RATio, the output of the HISTogram function is an absolute ratio. When ORDinate is set to PERCent, the output of the HISTogram function is PERCent (RATio*100). When ORDinate is set to COUNT, the output of the HISTogram function is the number of points in that amplitude belt.

PCT is provided as a synonym for PERCent since IEEE 488.2 allows the use of PCT as a numeric suffix.

At *RST this value is device dependent.

4.15.1.3 :POINTs <numeric_value>

CALCulate:TRANSform:HISTogram:POINTs

Specifies the number of POINTs (amplitude belts) in the histogram.

At *RST this value is device dependent.

4.15.1.4 :RANGe

CALCulate:TRANSform:HISTogram:RANGe

This node collects the commands used to control the size of the amplitude belts.

4.15.1.4.1 :**AUTO** <Boolean>

CALCulate:TRANSform:HISTogram:RANGe:AUTO

With AUTO ON, the number of output points is determined by the amplitude of the incoming data via a device dependent algorithm.

At *RST this function is set to a device dependent value.

4.15.1.5 :**STaTe** <Boolean>

CALCulate:TRANSform:HISTogram:STaTe

Determines whether the histogram transformation is enabled.

At *RST this function is OFF

4.15.2 :**TIME**

CALCulate:TRANSform:TIME

Specifies a transformation of data into its equivalent time representation. The parameters specify the time region over which the output data is transformed. The units are seconds.

4.15.2.1 :**STaTe** <Boolean>

CALCulate:TRANSform:TIME:STaTe

Determines whether the time transform is enabled.

At *RST, this function is OFF.

4.15.2.2 [:**TYPE**] **LPASsIBPASs**

CALCulate:TRANSform:TIME:TYPE

Selects a particular method to be used in band limiting information or the manner in which windows should be applied.

When LPASs is selected, the windowing shall affect only the end of the data set. LPASs requires data to start from zero in the appropriate units, such as Hertz and seconds. If the data set does not extend to zero, then the instrument shall either compute by extrapolation the extra points needed to complete the data set, or shall generate an execution (-200) error. When BPASs is selected, the applied window shall affect both the start and end parts of the data set.

At *RST, this value is device-dependent.

4.15.2.3 :**STIMulus STEPIIMpulse**

CALCulate:TRANSform:TIME:STIMulus

Specifies the type of stimulus to be simulated in the transform process.

At *RST, this value is device-dependent.

4.15.2.4 :**STARt** <numeric_value>

CALCulate:TRANSform:TIME:STARt

Specifies the start time of the output data record. Range is instrument- and setup-dependent.

The couplings between STARt, STOP, CENTer, and SPAN are the same as described in the SENSE subsystem.

At *RST, STARt is set to MIN.

4.15.2.5 :STOP <numeric_value>

CALCulate:TRANSform:TIME:STOP

Specifies the stop time of the output data record. Range is instrument- and setup-dependent.

The couplings between START, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, STOP is set to MIN.

4.15.2.6 :SPAN <numeric_value>

CALCulate:TRANSform:TIME:SPAN

Specifies the time span of the output data record. Range is instrument- and setup-dependent.

The couplings between START, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, SPAN is set to MIN.

4.15.2.7 :CENTER <numeric_value>

CALCulate:TRANSform:TIME:CENTer

Specifies the center time of the output data record. Range is instrument- and setup-dependent.

The couplings between START, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, CENTER is set to MIN.

4.15.2.8 :POINTS <numeric_value>

CALCulate:TRANSform:TIME:POINTS

Specifies the number of points output by the transform subsystem.

At *RST, this value is device-dependent.

4.15.2.8.1 :AUTO <Boolean>IONCE

CALCulate:TRANSform:TIME:POINTS:AUTO

When AUTO is ON, the number of points is determined by the size of the incoming SENSE Data.

At *RST, AUTO is ON.

4.15.2.9 :WINDOW RECTangular|UNIFORM|FLATtop|HAMMING |HANNING|KBESsell|FORCe|EXPonential

CALCulate:TRANSform:TIME:WINDOW

Specifies the type and parameter of data windowing (shaping) done prior to the transformation.

At *RST, this value is device-dependent.

Individual windowing algorithms are defined in “*On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform*,” F. J. Harris, Proc. of the IEEE, Vol. 66-1, January 1978, pp 51-83.

4.15.2.10 :KBESsel <numeric_value>

CALCulate:TRANSform:TIME:KBESsel

The Kaiser BESsel command sets the parametric window parameter for the Kaiser Bessel window.

At *RST, this value is device-dependent.

4.15.2.11 :EXPonential <numeric_value>

CALCulate:TRANSform:TIME:EXPonential

Enters the exponential decay time constant which characterizes the EXPonential window.

At *RST, this value is device-dependent.

4.15.2.12 :FORCe <numeric_value>

CALCulate:TRANSform:TIME:FORCe

Enters the time value parameter corresponding to the width of the gated portion of the input time record for FORCe windows.

At *RST, this value is device-dependent.

4.15.3 :DISTance

CALCulate:TRANSform:DISTance

This subsystem specifies a transformation of data into its equivalent distance representation. The parameters specify the distance region over which the output data is transformed. The units are meters.

4.15.3.1 :STATe <Boolean>

CALCulate:TRANSform:DISTance:STATe

Determines whether the distance transform is enabled.

At *RST, this function is OFF.

4.15.3.2 [:TYPE] LPASsIBPASs

CALCulate:TRANSform:DISTance:TYPE

Selects a particular method to be used in limiting information or the manner in which windows shall be applied for the specified transform.

When LPASs is selected, the windowing shall affect only the end of the data set. LPASs requires data to start from zero in the appropriate units, such as Hertz and seconds. If the data set does not extend to zero, then the instrument shall either compute by extrapolation the extra points needed to complete the data set, or shall generate an execution (-200) error. When BPASs is selected, the applied window shall affect both the start and end parts of the data set.

At *RST, this value is device-dependent.

4.15.3.3 :STIMulus STEPIIMPulse

CALCulate:TRANSform:DISTance:STIMulus

Specifies the type of stimulus to be simulated in the transform process.

At *RST, this value is device-dependent.

4.15.3.4 :START <numeric_value>

CALCulate:TRANSform:DISTance:STARt

Specifies the start distance of the output data record. Range is instrument- and setup-dependent.

The couplings between STARt, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, STARt is set to MIN.

4.15.3.5 :STOP <numeric_value>

CALCulate:TRANSform:DISTance:STOP

Specifies the stop distance of the output data record. Range is instrument- and setup-dependent.

The couplings between STARt, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, STOP is set to MIN.

4.15.3.6 :SPAN <numeric_value>

CALCulate:TRANSform:DISTance:SPAN

Specifies the distance span of the output data record. Range is instrument- and setup-dependent.

The couplings between STARt, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, SPAN is set to MIN.

4.15.3.7 :CENTER <numeric_value>

CALCulate:TRANSform:DISTance:CENTER

Specifies the center distance of the output data record. Range is instrument- and setup-dependent.

The couplings between STARt, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, CENTER is set to MIN.

4.15.3.8 :POINTS <numeric_value>

CALCulate:TRANSform:DISTance:POINTS

Specifies the number of points output by the transform subsystem.

At *RST, this is set to a device-dependent value.

4.15.3.8.1 :AUTO <Boolean>IONCE

CALCulate:TRANSform:DISTance:POINTS:AUTO

When AUTO is ON, the number of points is determined by the size of the incoming SENSE data.

At *RST, AUTO is ON.

4.15.3.9 :WINDow RECTangular|UNIFORM|FLAT|top|HAMMING IHANNing|KBESsel|FORCe|EXPonential

CALCulate:TRANSform:DISTance:WINDow

Specifies the type and parameter of data windowing (shaping) done prior to the transformation.

At *RST, this value is device-dependent.

Individual windowing algorithms are defined in “*On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform*,” F. J. Harris, Proc. of the IEEE, Vol. 66-1, January 1978, pp 51-83.

4.15.3.10 :KBESsel <numeric_value>

CALCulate:TRANSform:DISTance:KBESsel

The Kaiser BESsel command sets the parametric window parameter for the Kaiser Bessel window.

At *RST, this value is device-dependent.

4.15.3.11 :EXPonential <numeric_value>

CALCulate:TRANSform:DISTance:EXPonential

Enters the exponential decay time constant which characterizes the EXPonential window.

At *RST, this value is device-dependent.

4.15.3.12 :FORCe <numeric_value>

CALCulate:TRANSform:DISTance:FORCe

Enters the time value parameter corresponding to the width of the gated portion of the input time record for FORCe windows.

At *RST, this value is device-dependent.

4.15.4 :FREQuency

CALCulate:TRANSform:FREQuency

This subsystem specifies a transformation of data into its equivalent frequency representation. The parameters specify the frequency region over which the output data is transformed. The units are Hertz.

4.15.4.1 :STATe <Boolean>

CALCulate:TRANSform:FREQuency:STATe

Determines whether the frequency transform is enabled.

At *RST, this function is OFF.

4.15.4.2 [:TYPE] LPASs|BPASs

CALCulate:TRANSform:FREQuency:TYPE

Selects a particular method to be used in limiting information or the manner in which windows shall be applied for the specified transform.

When LPASs is selected, the windowing shall affect only the end of the data set. LPASs requires data to start from zero in the appropriate units, such as Hertz and seconds. If the

data set does not extend to zero, then the instrument shall either compute by extrapolation the extra points needed to complete the data set, or shall generate an execution (-200) error. When BPASs is selected, the applied window shall affect both the start and end parts of the data set.

At *RST, this value is device-dependent.

4.15.4.3 :STIMulus STEPIIMPulse

CALCulate:TRANSform:FREQuency:STIMulus

Specifies the type of stimulus to be simulated in the transform process.

At *RST, this selection is device-dependent.

4.15.4.4 :STARt <numeric_value>

CALCulate:TRANSform:FREQuency:STARt

Specifies the start frequency of the output data record. Range is instrument- and setup-dependent.

The couplings between STARt, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, STARt is set to MIN.

4.15.4.5 :STOP <numeric_value>

CALCulate:TRANSform:FREQuency:STOP

Specifies the stop frequency of the output data record. Range is instrument- and setup-dependent.

The couplings between STARt, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, STOP is set to MIN.

4.15.4.6 :SPAN <numeric_value>

CALCulate:TRANSform:FREQuency:SPAN

Specifies the frequency span of the output data record. Range is instrument- and setup-dependent.

The couplings between STARt, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, SPAN is set to MIN.

4.15.4.7 :CENTER <numeric_value>

CALCulate:TRANSform:FREQuency:CENTER

Specifies the center frequency of the output data record. Range is instrument- and setup-dependent.

The couplings between STARt, STOP, CENTER, and SPAN are the same as described in the SENSE subsystem.

At *RST, CENTER is set to MIN.

4.15.4.8 :POINTs <numeric_value>

CALCulate:TRANSform:FREQuency:POINTs

Specifies the number of points output by the transform subsystem.

At *RST, this value is device-dependent.

4.15.4.8.1 :AUTO <Boolean>|ONCE

CALCulate:TRANSform:FREQuency:POINTs:AUTO

When AUTO is ON, the number of points is determined by the size of the incoming SENSE Data.

At *RST, AUTO is ON.

4.15.4.9 :WINDow RECTangular|UNIFORM|FLATtop|HAMMING IHANNING|KBESsel|FORCe|EXPonential

CALCulate:TRANSform:FREQuency:WINDow

This specifies the type and parameter of data windowing (shaping) done prior to the transformation.

At *RST, this value is device-dependent.

Individual windowing algorithms are defined in “*On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform*,” F. J. Harris, Proc. of the IEEE, Vol. 66-1, January 1978, pp 51-83.

4.15.4.10 :KBESsel <numeric_value>

CALCulate:TRANSform:FREQuency:KBESsel

The Kaiser BESsel command sets the parametric window parameter for the Kaiser Bessel window.

At *RST, this value is device-dependent.

4.15.4.11 :EXPonential <numeric_value>

CALCulate:TRANSform:FREQuency:EXPonential

Enters the exponential decay time constant which characterizes the EXPonential window.

At *RST, this value is device-dependent.

4.15.4.12 :FORCe <numeric_value>

CALCulate:TRANSform:FREQuency:FORCe

Enters the time value parameter corresponding to the width of the gated portion of the input time record for FORCe windows.

At *RST, this value is device-dependent.

4.16 :PATH(MATHITRANSform|FILTER|SMOothing|FORMat|LIMIT|AVERage) {,(MATHITRANSform|FILTER|SMOothing|FORMat|LIMIT|AVERage)}

CALCulate:PATH

This subsystem defines the order in which CALCulate subsystems are to be performed. In many instruments this will be fixed and instrument-dependent. If it is possible to order the operations, then PATH is the method.

1999 SCPI Command Reference

For example, if TRANSform is first, then FILTer, and finally MATH, the following would be sent:

```
CALCulate:PATH TRANSform,FILTer,MATH
```

Even if not settable, the query of PATH should return the actual order of operations.

The suffix of each subsystem used as a parameter in the PATH command corresponds to the instance of the specified CALCulate subsystem sub-block. For example, the command PATH FILTer1, FILTer2, sets up the data flow through the filter defined by CALCulate:FILTer1 and then through the filter defined by CALCulate:FILTer2.

At *RST, the PATH definition is set to a device-dependent setting.

5 CALibration Subsystem

This subsystem has the function of performing system calibration.

Instrument CALibration commands tend to be very class specific. Such commands properly belong in Volume 4 “Instrument Classes”. Therefore, expanded commands for controlling instrument calibration may not be found in Volume 2 “Command Reference”.

KEYWORD	PARAMETER FORM	NOTES
CALibration		
[:ALL]		[event; no query] 1994
[:ALL]?		[event; query]
:AUTO	<Boolean> ONCE	
:BINertia		1999
:AVERage?	<numeric_value>	[query only] 1999
:HSPeed	<numeric_value>	1999
:INITiate		[event; no query] 1999
:LSPeed	<numeric_value>	1999
:NRUNs	<numeric_value>	1999
:SDEviation?	<numeric_value>	[query only] 1999
:UPDate		[event; no query] 1999
:DATA	<arbitrary block program data>	
:PLOSs		1999
:APCoeffs	<numeric_value>,<numeric_value>,...	1999
:INITiate		[event; no query] 1999
:LATime	<numeric_value>	1999
:STIMe	<numeric_value>	1999
:UPDate		[event; no query] 1999
:SOURce	INTernal EXTernal	
:STATe	<Boolean>	
:VALue	<numeric_value>	
:WARMup		1999
:INITiate		[event; no query] 1999
:SPEed	<numeric_value>	1999
:TIMEout	<numeric_value>	1999
:ZERO		
:AUTO	<Boolean> ONCE	
:FSEnsor		1999
:INITiate		[event; no query] 1999
:LATime	<numeric_value>	1999
:LEVel?		[query only] 1999
:SPEed	<numeric_value>	1999
:STIM	<numeric_value>	1999
:UPDate		[event; no query] 1999

5.1

[[:ALL]

CALibration:ALL

The CALibrate:ALL command performs the same function as the CALibrate:ALL? except there is no response. An instrument shall report calibration errors through the status-reporting mechanism.

If this command is implemented as overlapped it shall be included in the No Operation Pending flag (see IEEE 488.2 chapter 12).

While this command is executing the CALibrating bit of the Operation Status Condition register shall be set (see 9.3 of Syntax & Style).

5

5.2

[[:ALL]]?

CALibration:ALL?

The ALL? query performs a full calibration of the instrument and responds with a <numeric_value> indicating the success of the calibration. A zero shall be returned if calibration is completed successfully; otherwise a nonzero value which represents the appropriate error number shall be returned. An instrument shall still report calibration errors through the status-reporting mechanism, even though an error is reported by the value of the query response.

5.3

:AUTO <Boolean>IONCE

CALibration:AUTO

AUTO sets whether or not the instrument should perform auto calibration at device-dependent intervals without user intervention.

At *RST, AUTO OFF is selected.

5.4

:BINertia

CALibration:BINertia

Base Inertia

For chassis dynamometers, this node describes the procedure to determine the base mechanical inertia of the dynamometer roll system (all rotating mechanical components outside of the control loop). This procedure consists of an acceleration and deceleration of the dynamometer rolls, from which a mechanical inertia calculation is made. The dynamometer is motored from an initial speed to a final speed at a given constant acceleration, and this interval is timed. Immediately, the dynamometer is motored back from the final speed to the initial speed at the same constant acceleration (inverted), and this interval is timed. For this pair of intervals, the force transducer output is continuously measured, and the average interval force calculated. The average acceleration is calculated using the interval initial and final speed values, and the interval time. This data is used to calculate the mechanical inertia ($Inertia = Force / acceleration$).

5.4.1

:AVERage?

CALibration:BINertia:AVERage?

This query returns the running average of all base inertia values acquired since the last CALibration:BINertia:INITiate was invoked.

5.4.2 :HSPeet <numeric_value>

CALibration:BINertia:HSPeet

High Speed

The higher magnitude speed (m/s) of the speed interval.

At *RST, this value is set to 16.

5.4.3 :INITiate

CALibration:BINertia:INITiate

Begin motoring the dynamometer through the ACCeleration/deceleration pairs, using :SOURce:ACCeleration as the acceleration/deceleration rate, from LSPeet to HSPeet and back, NRUNs times. All query commands must be accepted and responded to while this command is executing. This command describes an event and therefore has no associated *RST condition.

When this command is initiated for dynamometers, the dynamometer must:

- Check critical ancillaries, if any, for proper conditions: Brakes Off, Motor Contactor On, parameters within dynamometer safety limits. For conditions that prohibit operation of this procedure, issue appropriate warning messages, and execute the Idle Procedure.
- Set the DYNO operation condition register Base Inertia Procedure bit indicating the Base Inertia is executing. Note: The DYNO operation condition register will be further defined in Volume 4.
- Accelerate/decelerate the dynamometer through the speed intervals, recording time, average acceleration, average force for the specified number of runs. The data is placed in the BaseInertia table after each acceleration/deceleration is run. Update the :CALibration:BINertia:SDEVIation value based on this run. End the procedure when the specified runs are complete.
- Clear the DYNO operation condition register Base Inertia Procedure bit indicating the Base Inertia is not executing.
- Issue the process complete message.
- Execute the Idle Procedure.
- Update the on-line mechanical inertia value, if desired.

5.4.4 :LSPeet <numeric_value>

CALibration:BINertia:LSPeet

Low Speed

The lower magnitude speed (m/s) of the speed interval.

At *RST, this value is set to 6.

5.4.5 :NRUNs <numeric_value>

CALibration:BINertia:NRUNs

Number of Runs - NRUNs

1999 SCPI Command Reference

The number of ACCeleration/deceleration pairs to perform. NRUNs will equal the number of base inertia values acquired.

At *RST, this value is set to 10.

5.4.6 :SDEviation?

CALibration:BINertia:SDEviation?

Standard Deviation

This query returns the running standard deviation of all base inertia values acquired since the last CALibration:BINertia:INITiate was invoked.

5

5.4.7 :UPDate

CALibration:BINertia:UPDate

Place on-line the last determined average base mechanical inertia value.

This command describes an event and therefore has no associated *RST condition.

5.5 :DATA <arbitrary block program data>

CALibration:DATA

Transfers the calibration data as arbitrary block program data. The query form returns the current calibration data. The block is transferred into the current calibration data. If an error occurs during the transfer causing the calibration to be lost, an error -313 shall be generated in addition to the execution error.

5.6 :PLOSs

CALibration:PLOSs

Parasitic Loss

For chassis dynamometers this node describes the commands associated with the parasitic loss procedure. This procedure measures and, if requested, updates the parasitic loss corrections for the dynamometer. The parasitic losses are measured by operating the dynamometer at various selected speeds and measuring the force required to maintain the constant speed. This procedure uses the :MEMory table 'PARasitic' that defines the speed points at which parasitic loss measurements are taken. The parasitic loss coefficients are calculated at the end of the procedure and reside in the memory table PCoefficient.

5.6.1 :APCcoeff

<numeric_value>,<numeric_value>,<numeric_value>[,<numeric_value>]

:CALibration:PLOSs:APCcoeff

Active Parasitic Coefficients - APCoefficients

This is the presently active parasitic loss curve used by the dynamometer. The dynamometer will maintain coefficients for forward and reverse operations. The appropriate values must be placed in the APCoeff table, based on roll rotation.

5.6.2 :INITiate

:CALibration:PLOSs:INITiate

This command starts the parasitic loss procedure. This is an overlapped command. This command describes an event and therefore has no associated *RST condition.

When this command is initiated for dynamometers, the dynamometer must:

- Check critical ancillaries, if any, for proper conditions: Brakes Off, Motor Contactor On, parameters within dynamometer safety limits. For conditions that prohibit operation of this procedure, issue appropriate warning messages, and execute the Idle Procedure.
- Set the DYNO operation condition register Parasitic Loss Procedure bit indicating the Parasitic Loss procedure is executing. Note: The DYNO operation condition register will be further defined in Volume 4.
- Accelerate dynamometer to the first speed point defined in the PARasitic memory table. Stabilize at speed point for defined stabilization time.
- Average the force at the roll surface required to maintain the speed set point over the latency time. The average force value is placed in the PARasitic table for the specified speed point. Continue until all speed points have been executed in the PARasitic table. The first zero speed in the PARasitic table ends the procedure.
- Perform a curve fit on the PARasitic table data and place the coefficients in the memory table PCoeff.
- Clear the DYNO operation condition register Parasitic Loss Procedure bit indicating the parasitic loss procedure is not executing.
- Issue the process complete message.
- Execute the Idle Procedure.

5.6.3 :LATime

:CALibration:PLOSs:LATime

Loss Averaging Time - sec

The length of time in seconds to average the losses at a speed point.

At *RST, this value is set to 5.

5.6.4 :STIME <numeric_value>

:CALibration:PLOSs:STIME

Stabilization Time - sec

Time for stabilization prior to LATime.

At *RST, this value is set to 5.

5.6.5 :UPDate

:CALibration:PLOSs:UPDate

Update the dynamometer parasitic loss coefficients contained in the table APCoeff with the coefficients contained in memory table PCoeff.

This command is an event and has no associated *RST condition.

1999 SCPI Command Reference

5.7 :SOURce INTernal|EXTernal

CALibration:SOURce

Controls the source of the calibration signal. If set to INTernal, a reference internal to the instrument is selected. If SOURce is set to EXTernal, then an external reference shall be applied.

5.8 :STATe <Boolean>

CALibration:STATe

STATe is used to select if the calibration data is applied or not. If STATe is ON then the instrument uses the calibration data for correction. If STATe OFF is selected, then no correction using the calibration data shall be made.

5

At *RST, STATe is set to ON.

5.9 :VALue <numeric_value>

CALibration:VALue

Enters the value of the reference signal which is used in the calibration. For example, a voltmeter may calibrate its input to a reference signal which has a proven value of 4.99327513 Volts.

5.10 :WARMup

CALibration:WARMup

This node controls parameters relating to the warmup of the device.

For chassis dynamometers, this node describes the procedure to set up the conditions required to ready the dynamometer for testing. The warm-up consists of running the dynamometer at a fixed speed over a period of time. Warm-up completion is defined by either running the dynamometer for a fixed period of time, or by monitoring parasitic losses and comparing them to a target value tolerance.

5.10.1 :INITiate

CALibration:WARMup:INITiate

Start the warmup procedure.

For chassis dynamometers, motor the dynamometer roll to SPEed, and hold this speed for a TIMEout period of time. All query commands must be accepted and responded to while this command is executing. This command describes an event and therefore has no associated *RST condition.

When this command is initiated for dynamometers, the dynamometer must:

- Check critical ancillaries, if any, for proper conditions: Brakes Off, Motor Contactor On, parameters within dynamometer safety limits. For conditions that prohibit operation of this procedure, issue appropriate warning messages, and execute the Idle Procedure.
- Set the DYNO operation condition register Warm up Procedure bit indicating the Warm-up is executing. Note: The DYNO operation condition register will be further defined in Volume 4.
- Accelerate dynamometer to the speed point defined by :CALibration:WARMup:SPEed.

- Average the force at the roll surface required to maintain the speed set point over :CALibrate:WARMup:LATime. The average force can be monitored by :SENSe:FORCe?
- The procedure will end when the elapsed time is greater than :CALibration:WARMup:TIMEout.
- Clear the DYNO operation condition register Warm up Procedure bit indicating the Warm-up is not executing.
- Issue the process complete message.
- Execute the Idle Procedure.

5.10.2 :SPEed <numeric_value>

CALibration:WARMup:SPEed

For chassis dynamometers this is the fixed roll speed (m/s) at which the warm-up will be run.

At *RST, this value is set to 22.

5.10.3 :TIMEout <numeric_value>

CALibration:WARMup:TIMEout

For chassis dynamometers this is the length of time in seconds the warm-up will run before automatically returning to zero speed.

At *RST, this value is device dependent.

5.11 :ZERO

CALibration:ZERO

This subsystem controls the autozero calibration of the sensor.

For chassis dynamometers, this node describes the procedure to set up the conditions required and perform a mathematical zero correction of the roll force measurement system calibration. The correction is determined from a measurement of the roll force sensor at a given roll speed. The results are stored in the FSensorZero pre-defined table.

5.11.1 :AUTO <Boolean>IONCE

CALibration:ZERO:AUTO

Controls whether autozeroing calibration occurs.

At *RST, this value is set to OFF.

5.11.2 :FSENsor

CALibration:ZERO:FSENsor

Force Sensor

For chassis dynamometers this node sets up the conditions required and performs a mathematical zero correction of the roll force measurement system calibration. The correction is determined from a measurement of the roll force sensor at a given roll speed.

5.11.2.1 :INITiate

CALibration:ZERO:FSENSor:INITiate

This is an overlapped command. Perform the measurement of the roll force, acquiring for the length of time specified by :LATime. This command describes an event and therefore has no associated *RST condition.

When this command is initiated for dynamometers, the dynamometer must:

- Check critical ancillaries, if any, for proper conditions: Brakes Off, Motor Contactor On, parameters within dynamometer safety limits. For conditions that prohibit operation of this procedure, issue appropriate warning messages, and execute the Idle Procedure.
- Set the DYNO operation condition register Force Sensor Zero Procedure bit indicating the Zero is executing. Note: The DYNO operation condition register will be further defined in Volume 4.
- Accelerate dynamometer to the speed specified in :CALibration:ZERO:FSENSor:SPEed. Stabilize for :STIME.
- Average the force at the roll surface required to maintain the speed set point over the :LATime. The average force value is placed in :LEVel.
- Clear the DYNO operation condition register Force Sensor Zero Procedure bit indicating the Zero is not executing.
- Issue the process complete message.
- Execute the Idle Procedure.

5.11.2.2 :LATime <numeric_value>

CALibration:ZERO:FSENSor:LATime

Loss Averaging Time - sec

The length of time to average the measured zero reading.

At *RST, this value is set to 0.5.

5.11.2.3 :LEVel?

CALibration:ZERO:FSENSor:LEVel?

Returns the measured force value used to determine the mathematical zero correction. This value is in units of force at the roll surface. This is query only.

5.11.2.4 :SPEed <numeric_value>

CALibration:ZERO:FSENSor:SPEed

The roll speed set point for the force measurement period in m/s.

At *RST, this value is set to 0.

5.11.2.5 :STIME <numeric_value>

CALibration:ZERO:FSENSor:STIME

Stabilization Time - sec

Time for stabilization prior to LATime.

At *RST, this value is set to 5.

5.11.2.6 :UPDate

CALibration:ZERO:FSENSor:UPDate

Once the mathematical zero correction has been determined, apply the results to the on-line calibration. This is not a queryable command and therefore has no *RST associated with it.

6 CONTROL Subsystem

The CONTROL subsystem is used to turn on and off electromechanical devices or control some dynamometer states that are used in a procedure.

KEYWORD	PARAMETER FORM	NOTES
:CONTRol		1993
:APOWer		1993
[:STATe]	<Boolean>	1993
:BLOWer		1993
[:STATe]	<Boolean>	1993
:BRAKe		1999
[:STATe]	<Boolean>	1999
:COMPressor		1993
[:STATe]	<Boolean>	1993
:COVer		1999
[:ADJust]	OPEN CLOSe SOPen SCLOse	[event; no query] 1999
:POSition?		[query only] 1999
:EBENch		1999
[:CLEan]		1999
:INITiate		[event; no query] 1999
:DURation	<numeric_value>	1999
:IDLE		1999
:INITiate		[event; no query] 1999
:LIFT		1999
[:ADJust]	UP DOWN	[event; no query] 1999
:POSition?		[query only] 1999
:MCONTRol		1999
[:STATe]	<Boolean>	1999
:ROTation		1999
[:DIRection]	FORWard REVerse	1999
:VCDevice		1999
[:STATe]	<Boolean>	1999
:TDIameter	<numeric_value>	1999

6.1 :APOWer

CONTRol:APOWer

Controls the Auxiliary POWER. Auxiliary POWER are additional power outlets that can be used to power devices.

6.1.1 [:STATe] <Boolean>

CONTRol:APOWer[:STATe]

Turns the APOWer ON and OFF.

At *RST this value is set to OFF.

6.2 :BLOWer

CONTRol:BLOWer

Controls the BLOWer.

For example, in order to keep a uniform temperature inside the environment chamber, the air inside needs to be thoroughly mixed. This is usually accomplished using a blower.

6.2.1 [:STATe] <Boolean>

CONTRol:BLOWer[:STATe]

Turns the BLOWer ON and OFF.

At *RST this value is set to OFF.

6.3 :BRAKe

CONTRol:BRAKe

This node contains the commands that control the brake.

6.3.1 [:STATe] <Boolean>

:CONTRol:BRAKe[:STATe]

Sets or queries the state of the brake.

At *RST, this value is set to OFF.

6.4 :COMPressor

CONTRol:COMPressor

Controls the COMPressor.

For example in an environmental chamber, a compressor is often needed additional control of the heating and cooling.

6.4.1 [:STATe] <Boolean>

CONTRol:COMPressor[:STATe]

Turns the COMPressor ON and OFF.

At *RST this value is set to OFF.

6.5 :COVer

CONTRol:COVer

This node contains the commands which control the cover.

6.5.1 [:ADJust] OPENICLOSe|SOPEn|SCLOSe

CONTRol:COVer[:ADJust]

This command controls the cover. This command is an event and has no associated *RST condition

- OPEN - open covers
- CLOSe - close covers
- SCLOSe - step covers incrementally toward CLOSe
- SOPEn - step covers incrementally toward OPEN

6.5.2 :POSition?

CONTRol:COVer:POSition?

This command queries the position of the cover

- OPEN - covers are fully open
- CLOSe - covers are fully closed
- MID - covers are partially open

6.6 :EBENch

CONTRol:EBENch

This subsystem controls the state and activities of a gas analyzer emissions bench.

6.6.1 :CLEan

:CONTRol:EBENch:CLEan

This subsystem controls the bench's internal procedure to clean out its gas lines, commonly known as "purge" or "backflush".

6.6.1.1 :INITiate

:CONTRol:EBENch:CLEan:INITiate

Initiate the bench's procedure to clean out its gas lines.

This command is an event and therefore has no *RST value.

6.6.1.2 :DURation <numeric_value>

:CONTRol:EBENch:CLEan:DURation

This command sets or queries the duration in seconds of the "clean" procedure.

At *RST, this value is device-dependent.

6.7 :IDLE

CONTRol:IDLE

This node contains commands that control the idle state of the device.

6.7.1 :INITiate

:CONTRol:IDLE:INITiate

Returns the device to idle.

For chassis dynamometers, the dynamometer must decelerate to 0 m/s at the :SOURce:ACCEleration rate and wait for a command. The DYNO operation condition register Idle Procedure bit should be set indicating that :CONTRol:IDLE is executing. Note: The DYNO operation condition register will be further defined in Volume 4.

This command is an event and has no associated *RST condition.

6.8 :LIFT

CONTRol:LIFT

This node contains commands that control the lift.

1999 SCPI Command Reference

6.8.1 **[[:ADJust] UPIDOWN**

CONTRol:LIFT[:ADJust]

This command controls the lift. This command is an event and has no associated *RST condition.

- UP - raises lift
- DOWN - lowers lift

6.8.2 **:POSition?**

CONTRol:LIFT:POSition?

Queries the position of the lift.

- UP - lifts are up.
- DOWN - lifts are down.
- MID - lifts are positioned between UP and DOWN.

6.9 **:MCONTRol**

CONTRol:MCONTRol

Motor Control

Controls the motor.

6.9.1 **[[:STATe] <Boolean>**

:CONTRol:MCONTRol[:STATe]

Turns the power of motor ON or Off.

At *RST, this value is set to OFF.

6.10 **:ROTation**

CONTRol:ROTation

This node contains commands that control device rotation.

6.10.1 **[[:DIRection]**

CONTRol:ROTation[:DIRection]

For chassis dynamometers, configures the dynamometer rotation direction. The dynamometer determines FORWard or REVerse based on configuration.

At *RST, this value is set to FORWard.

6.11 **:VCDevice**

CONTRol:VCDevice

Vehicle Centering Device

For chassis dynamometers, this node contains commands that control the Vehicle Centering Device.

6.11.1 **[[:STATe] <Boolean>**

CONTRol:VCDevice[:STATe]

When the state is ON, the dynamometer is performing the centering function. When the state is OFF, centering is not being performed.

At *RST, this value is set to OFF.

6.11.2 **:TDiameter <numeric_value>**

CONTRol:VCDevice:TDiameter

Tire Diameter

Centering device uses tire diameter (m) as limit to center the vehicle on the dynamometer.

At *RST, this value is set to 0.7.

7 **DIAGnostic Subsystem**

The purpose of the DIAGnostic subsystem is to provide a tree node for all of the instrument service and diagnostic routines used in routine maintenance and repair. The actual commands in this subsystem are considered instrument-specific, and therefore are not included in this document. Instrument designers are free to add service and diagnostic commands to meet the needs of individual instruments.

8 DISPlay Subsystem

The DISPlay subsystem controls the selection and presentation of textual, graphical, and TRACe information. This information includes measurement data, user-interaction displays, and data presented to the instrument by the controller. DISPlay is independent of, and does not modify, how data is returned to the controller.

Multiple DISPlay subsystems are used to represent independent display medium. A front panel mounted display and an attached terminal used only for information display are examples of independent displays. Conversely, an instrument with many dedicated indicators is considered to have one general display because of the dependencies in operation that exist between the indicators.

Within a DISPlay, information may be separated into individual WINdows (this is always the case for instruments with dedicated indicators). Each window is considered to consist of three overlapped planes, one each for text, graphics, and trace data. Thus, text, graphics, and trace information may be presented at the same time in a given window.

Most of the *RST conditions in the DISPlay subsystem are device-dependent, to accommodate for typical display functions that exist for each of the various categories of instrumentation.

KEYWORD	PARAMETER FORM	NOTES
DISPlay		
:ANNotation		
[:ALL]	<Boolean>	
:AMPLitude	<Boolean>	
:FREQuency	<Boolean>	
:BRIGhtness	<numeric_value>	
:CMAp		
:DEFault	[event; no query]	
:COLor		1992
:HSL	<hue>,<sat>,<lum>	1992
:RGB	<red>,<green>,<blue>	1992
:CONTrast	<numeric_value>	
:ENABle	<Boolean>	
:MENU		
[:NAME]	<menu_name>	
:STATe	<Boolean>	1992
:KEY	<string>	
[:WINDow]		
:BACKground		
:COLor	<color number>	
:GEOMetry		
:LLEFt	<numeric_value>,<numeric_value>	
:SIZE	<numeric_value>,<numeric_value>	
:URIGHt	<numeric_value>,<numeric_value>	

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
:GRAPhics		
:CLEAr		[event; no query]
:COLor	<color number>	
:CSIZe	<numeric_value>	
[[:DRAW]	<numeric_value>,<numeric_value>	
:PCL	<block>	
:HPGL	<block>	
:IDRaw	<numeric_value>,<numeric_value>	
:IMOVe	<numeric_value>,<numeric_value>	
:LABel	<string>	
:LDIRection	<numeric_value>	
:LTYPe	<numeric_value>[,<numeric_value>]	
:MOVE	<numeric_value>,<numeric_value>	
:STATe	<Boolean>	
[[:STATe]	<Boolean>	
:TEXT		
:ATTRibutes	<Boolean>	
:CLEAr		[event; no query]
:COLor	<numeric_value>	
:CSIZe	<numeric_value>	
[[:DATA]	<string> <block>	
:FEED	<data_handle>	1991
:LOCate	<row>[,<col>]	
:PAGE	<numeric_value>	1991
:STATe	<Boolean>	
:TRACe		
:COLor	<numeric_value>	
:FEED	<data_handle>	1991
:GRATicule		
:AXIS		
[[:STATe]	<Boolean>	
:COLor	<numeric_value>	
:FRAMe		
[[:STATe]	<Boolean>	
:GRID		
:AUTO	<Boolean>	
[[:STATe]	<Boolean>	
:PERSistence	<numeric_value>	
:AUTO	<Boolean> ONCE	
[[:STATe]	<Boolean>	
:X		
:LABel	<string>	
[[:SCALE]		

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
:AUTO	<Boolean> ONCE	1991
:CENTer	<numeric_value>	1991
:LEFT	<numeric_value>	
:PDIVision	<numeric_value>	1991
:LINK	LEFT CENTer RIGHT	1991
:RIGHT	<numeric_value>	
:SPACing	LINEar LOGarithmic	
:Y		
:LABel	<string>	
:RLINe	<Boolean>	1991
[:SCALe]		
:AUTO	<Boolean> ONCE	1991
:BOTTom		
:PDIVision	<numeric_value>	1991
:RLEVel	<numeric_value>	1991
:AUTO	<Boolean>	1991
:RPOSition	<numeric_value>	1991
:TOP	<numeric_value>	
:SPACing	LINEar LOGarithmic	
:R		
:LABel	<string>	
[:SCALe]		
:AUTO	<Boolean> ONCE	1991
:CPOint	<numeric_value>	
:OEDGE	<numeric_value>	
:SPACing	LINEar LOGarithmic	

8.1 :ANNotation

DISPlay:ANNotation

This subsystem controls which annunciators are visible. ANNotation does not affect the visibility of GRATicule. If SYSTem:SECurity is enabled, annotation cannot be switched from OFF to ON.

The ANNotation parameters are set to ON at *RST, unless SYSTem:SECurity is enabled.

8.1.1 [:ALL] <Boolean>

DISPlay:ANNotation:ALL

Controls ALL of the annotation information.

8.1.2 :AMPLitude <Boolean>

DISPlay:ANNotation:AMPLitude

Controls the amplitude annotation information.

8.1.3 :FREQuency <Boolean>

DISPlay:ANNotation:FREQuency

Controls the frequency annotation information.

8.2 :BRIGhtness <numeric_value>

DISPlay:BRIGhtness

Controls the intensity of the display. The range of the parameter is 0 to 1, where 1 is full intensity and 0 is fully blanked.

8.3 :CMAP

DISPlay:CMAP

The CMAP subsystem controls the physical color associated with logical color numbers in the color map of the display. On a monochrome DISPlay, the CMAP may be used to represent different intensity levels of the same shade. When a color is redefined, all information associated with that color number may change immediately.

Two methods are available for defining a particular color, HSL and RGB models. If both are implemented in a device, then they shall be coupled such that redefining a color with one model shall be reflected in the values of the other model.

At *RST, COLor[1] shall represent “black,” and COLor2 shall represent “white” or the actual color of a monochrome display.

8.3.1 :DEFault

DISPlay:CMAP:DEFault

Sets the color map to the instrument’s default values for all colors, except that COLor[1] shall be represent “black,” and COLor2 shall represent “white” or the actual color of a monochrome display.

8.3.2 :COLor

DISPlay:CMAP:COLor

This subsystem controls the color map for the instrument.

8.3.2.1 :HSL <hue>,<sat>,<lum>

DISPlay:CMAP:COLor:HSL

The HSL command sets the instrument's color map based on the Hue/Saturation/Luminance levels color model. A query of HSL shall return three <NR2>s separated by commas.

Hue ranges from zero to one, circularly, with a value of zero resulting in the same hue as a value of one. The approximate color progression is (starting at zero): red, orange, yellow, green, cyan, blue, magenta, and back to red.

Saturation is the amount of pure color to be mixed with white. The saturation value ranges from zero to one, with zero specifying no color (only white or gray, depending on intensity) and one specifying no white.

Luminance specifies the brightness per unit area of the color. A luminance of zero results in black; a luminance of one results in the brightest color available.

At *RST, these parameters are set to the DEFault values, equivalent to executing DEFault command. The HSL command is coupled to the RGB command. Changing values in either will affect the values of the other.

8.3.2.2 :RGB <red>,<green>,<blue>

DISPlay:CMAP:COLor:RGB

The RGB command sets the instrument's color map based on the Red/Green/Blue color model. A query of RGB shall return three <NR2>s separated by commas.

Red ranges from zero to one, with zero indicating "no red" and one indicating "full intensity red."

Green ranges from zero to one, with zero indicating "no green" and one indicating "full intensity green."

Blue ranges from zero to one, with zero indicating "no blue" and one indicating "full intensity blue."

At *RST, these parameters are set to the DEFault values, equivalent to executing DEFault command. The RGB command is coupled to the HSL command. Changing values in either will affect the values of the other.

8.4 :CONTRast <numeric_value>

DISPlay:CONTRast

Determines the relative difference in brightness between "full" intensity and "no" intensity as displayed. The parameter ranges in value from 0 to 1; 0 indicates no difference between data and background, and 1 indicates maximum contrast.

8.5 :ENABLE <Boolean>

DISPlay:ENABLE

Controls whether the whole display is visible. If SYSTem:SECurity is enabled, DISPlay cannot be switched from OFF to ON.

ENABLE is set to ON at *RST, unless SYSTem:SECurity is enabled.

8.6 :MENU

DISPlay:MENU

Allows selection of predefined device-dependent menus for display. A menu is basically a softkey help screen, and it may consist of only key labels or more extensive information.

8.6.1 [:NAME] <menu_name>

DISPlay:MENU:NAME

The NAME command selects a menu by name from an instrument-predefined list of valid menu names.

8.6.2 :STATe <Boolean>

DISPlay:MENU:STATe

Turns the current menu page ON or OFF.

8.6.3 :KEY <string>

DISPlay:MENU:KEY

Assigns the soft key label <string> to key. This affects only the displayed key label; it does not change the key definition.

8

8.7 [:WINDow]

DISPlay:WINDow

The WINDow subtree contains commands to enable a window, to set its size and location, and control its contents. Multiple instances of WINDow may exist under a particular DISPlay.

8.7.1 :BACKground

DISPlay:WINDow:BACKground

This subsystem controls the characteristics of the window background.

8.7.1.1 :COLor <numeric_value>

DISPlay:WINDow:BACKground:COLor

Selects from the CMAP the value given by <numeric_value> to become the background color.

8.7.2 :GEOMetry

DISPlay:WINDow:GEOMetry

This subtree is used to define the position and size of the WINDow with respect to the display. A window may be specified by using any two of LLEFt, URIGHt or SIZE commands, since $SIZE = URIGHt - LLEFt$. If any one of these commands is implemented, then all three shall be implemented and the coupling between them shall be maintained.

For the GEOMetry subtree, the lower left corner of the whole display is defined to be (0,0); zero in the horizontal direction and zero in the vertical direction. The upper right corner is defined to be (1,1).

The effect of changing the size or aspect ratio of a WINDow that already is displaying information is device-dependent. The device may choose to rescale the information to fit the new WINDow or it may leave the information unchanged, clipping it where it exceeds the WINDow boundaries.

8.7.2.1 :LLEft<numeric_value>,<numeric_value>

DISPlay:WINDow:GEOMetry:LLEft

This specifies (or queries) the location of the lower left corner of the window in the range 0 to 1.

8.7.2.2 :SIZE<numeric_value>,<numeric_value>

DISPlay:WINDow:GEOMetry:SIZE

This specifies (or queries) the size of the display window. The first parameter represents the length in the horizontal direction, and the second parameter represents the length in the vertical direction.

The parameters may be set in the range greater than zero to 1. Where the existence of a window can be represented by an icon, the device shall accept SIZE 0,0 to indicate the window is to be replaced by an icon.

If a window is represented by an icon, any valid SIZE command with non-zero parameters shall reset the size of the window, and shall cause the icon to be replaced by the window.

8.7.2.3 :URIGht<numeric_value>,<numeric_value>

DISPlay:WINDow:GEOMetry:URIGht

This specifies (or queries) the location of the upper right corner of the display window in the range 0 to 1.

8.7.3 :GRAPhics

DISPlay:WINDow:GRAPhics

The GRAPhics subtree allows for the display of graphical information in the WINDow. For example, it allows direct control of the WINDow by the controller, using it as a plotter output device.

For the GRAPhics subtree, the lower left corner of the WINDow is defined to be (0,0); zero in the horizontal direction and zero in the vertical direction. The upper right corner is defined to be (1,1).

8.7.3.1 :CLEar

DISPlay:WINDow:GRAPhics:CLEar

Erases the graphics from the WINDow.

This command defines an event and therefore has no query form or *RST value.

8.7.3.2 :COLor<numeric_value>

DISPlay:WINDow:GRAPhics:COLor

Selects from the CMAP the value given by <numeric_value> to become the color for the next GRAPhics operation. It is similar in function to selecting a pen on a plotter.

8.7.3.3 :CSIZe<numeric_value>[,<numeric_value>]

DISPlay:WINDow:GRAPhics:CSIZe

Sets the size and optional aspect ratio (width/height) of the character cell used by the LABEL command. The actual physical size of the resulting character is device-dependent.

8.7.3.4 [:DRAW] <numeric_value>,<numeric_value>

DISPlay:WINDow:GRAPhics:DRAW

Draws a line from the current “pen” position to the specified X and Y coordinate (the first parameter is X, the second is Y; both parameters are required). The current line type and pen number are used.

8.7.3.5 :PCL <block>

DISPlay:WINDow:GRAPhics:PCL

Graphics and text data are sent to the instrument using Hewlett-Packard “<esc>*” terminal- and printer-style escape sequences. These are described in the Hewlett-Packard “Printer Command Language” document. Some escape sequences may interact with global parameters such as color selection. The effect of these changes is global in scope and not restricted to the :GRAPhics subsystem.

8.7.3.6 :HPGL <block>

DISPlay:WINDow:GRAPhics:HPGL

Graphics and text data are sent to the instrument using HP-GL (Hewlett-Packard-Graphics Language) plotter language. Some HP-GL sequences will interact with global parameters such as color selection; the effect of these changes is global in scope and not restricted to the :GRAPhics subsystem.

8.7.3.7 :IDRaw <numeric_value>,<numeric_value>

DISPlay:WINDow:GRAPhics:IDRaw

Draws a line from the current pen position to a new position determined by adding an X offset (first parameter) and a Y offset (second parameter) to the current coordinates.

8.7.3.8 :IMOVe <numeric_value>,<numeric_value>

DISPlay:WINDow:GRAPhics:IMOVe

Updates the current pen position by adding an X offset (first parameter) and a Y offset (second parameter) to the current coordinates. No line is drawn.

8.7.3.9 :LABel <string>

DISPlay:WINDow:GRAPhics:LABel

Places text on the graphics display at the current pen position, using the current pen, with the label direction set by LDIR.

8.7.3.10 :LDIRectiOn <numeric_value>

DISPlay:WINDow:GRAPhics:LDIRectiOn

Defines the angle (in radians) at which labels are drawn. An angle of 0 is normal horizontal printing. The angle increases counter-clockwise; $\pi/2$ is bottom-to-top.

8.7.3.11 :LTYPe <numeric_value>[,<numeric_value>]

DISPlay:WINDow:GRAPhics:LTYPe

Selects a line type and optional repeat length for all subsequent lines drawn by IDRaw or DRAW. Although the specific line types are device-dependent, line type 1 is by tradition a “plain” line (solid), while other types include dots and dashes or combinations of dots and dashes.

8.7.3.12 **:MOVE <numeric_value>,<numeric_value>**

DISPlay:WINDow:GRAPhics:MOVE

Updates the pen position without drawing a new line.

8.7.3.13 **:STaTe <Boolean>**

DISPlay:WINDow:GRAPhics:STaTe

Controls whether the GRAPhics is visible or not.

8.7.4 **[[:STaTe] <Boolean>**

DISPlay:WINDow:STaTe

Controls whether the WINDow is visible or not. The command DISPlay <Boolean> refers to this node. If SYSTem:SECurity is enabled, WINDow cannot be switched from OFF to ON.

At *RST, DISPlay[:WINDow][[:STaTe] is set to ON, unless SYSTem:SECurity is enabled.

8.7.5 **:TEXT**

DISPlay:WINDow:TEXT

The TEXT subtree allows for the display of textual information in the WINDow. For example, it allows direct control of the WINDow by the controller, using it as a VDT (Visual Display Terminal) output device.

For the TEXT subtree, the upper left corner of the WINDow is defined to be (1,1). The lower right corner is dependent on the WINDow:SIZE and the TEXT:Character SIZE selected.

8.7.5.1 **:ATTRibutes <Boolean>**

DISPlay:WINDow:TEXT:ATTRibutes

ATTRibutes allows the device to interpret the ANSI Standard Terminal escape sequences when displaying TEXT:DATA. Escape sequences that generate an inquiry are not permitted. The device may absorb escape sequences that it cannot interpret to prevent them from appearing on the display.

If ATTRibutes is ON, escape sequences embedded in the data are used to control the attributes of the string, such as blinking, or underline. ATTRibutes set to OFF disables the interpretation of display sequences, but does not affect the current display. A blinking display keeps on blinking until it is overwritten. The device may absorb the escape sequences to prevent them from appearing on the display.

8.7.5.2 **:CLEar**

DISPlay:WINDow:TEXT:CLEar

Erases the text from the WINDow.

This command defines an event, and therefore has no query form or *RST value.

8.7.5.3 **:COLor <numeric_value>**

DISPlay:WINDow:TEXT:COLor

Selects from the CMAP the value given by <numeric_value> to become the color for the next DATA stream.

1999 SCPI Command Reference

8.7.5.4 **:CSIZe<numeric_value>[,<numeric_value>]**

DISPlay:WINDow:TEXT:CSIZe

Sets the size and optional aspect ratio (width/height) of the character cell used by DATA. Actual physical size of the resulting character is device-dependent.

8.7.5.5 **:FEED <data_handle>**

DISPlay:WINDow:TEXT:FEED

Sets or queries what data flow is fed into the TEXT display window.

At *RST, the <data_handle> is set to a device-dependent value.

8.7.5.6 **[[:DATA] <string> | <block>**

DISPlay:WINDow:TEXT:DATA

The data that gets written to the text display area. Note that writing to the display overwrites any data that may have already been written. The result of overwriting a string with a shorter string is not defined; the excess characters may or may not be cleared.

8 8.7.5.7 **:LOCate<numeric_value>[,<numeric_value>**

DISPlay:WINDow:TEXT:LOCate

This command selects the display ROW and display COLUMN where the next LINE of TEXT DATA is to appear. Row number 1 is the top row of the instrument display; column number 1 is the left-most column.

8.7.5.8 **:PAGE<numeric_value>**

DISPlay:WINDow:TEXT:PAGE

The PAGE command sets the page to be displayed. This command may accept UP and DOWN as defined in <numeric_value>.

8.7.5.9 **:STATe<Boolean>**

DISPlay:WINDow:TEXT:STATe

Controls whether the TEXT is visible or not.

8.7.6 **:TRACe**

DISPlay:WINDow:TRACe

The TRACe subtree allows for the display of trace data in the WINDow; for example, directly from measurement data or from MEMory. The format of the data (and associated information) is used by the TRACe subsystem to define how commands are to be interpreted and whether on not they are valid.

8.7.6.1 **:COLor<numeric_value>**

DISPlay:WINDow:TRACe:COLor

Selects from the CMAP the value given by <numeric_value> to become the color for the TRACe.

8.7.6.2 **:FEED <data_handle>**

DISPlay:WINDow:TRACe:FEED

Sets or queries what data flow is fed into the TRACe display window.

At *RST, the <data_handle> is set to a device-dependent value.

8.7.6.3 :GRATicule

DISPlay:WINDow:TRACe:GRATicule

The GRATicule subtree is used to set the type of graticule displayed in relation to the TRACe.

8.7.6.3.1 :AXIS

DISPlay:WINDow:TRACe:GRATicule:AXIS

The axes are either X and Y AXIS with markings (divisions), or the R AXIS with markings (divisions) with respect to the TRACe.

8.7.6.3.1.1 [:STATe] <Boolean>

DISPlay:WINDow:TRACe:GRATicule:AXIS:STATe

Determines if the AXIS is visible or not.

8.7.6.3.2 :FRAMe

DISPlay:WINDow:TRACe:GRATicule:FRAMe

The FRAMe is the perimeter boundary and markings with respect to the TRACe.

8.7.6.3.2.1 [:STATe] <Boolean>

DISPlay:WINDow:TRACe:GRATicule:FRAMe:STATe

Determines if the FRAMe is visible or not.

8.7.6.3.3 :GRID

DISPlay:WINDow:TRACe:GRATicule:GRID

The GRID provides constant lines of X, Y or R with respect to the TRACe.

8.7.6.3.3.1 :AUTO <Boolean>

DISPlay:WINDow:TRACe:GRATicule:GRID:AUTO

AUTO couples the subtree of GRID to FRAMe. Turning ON FRAMe, with AUTO set to ON, shall cause GRID to turn on also.

8.7.6.3.3.2 [:STATe] <Boolean>

DISPlay:WINDow:TRACe:GRATicule:GRID:STATe

Determines if the GRID is visible or not.

8.7.6.4 :PERSistence <numeric_value>

DISPlay:WINDow:TRACe:PERSistence

Sets how long trace data written to the screen will remain visible. The parameter is in seconds.

8.7.6.4.1 :AUTO <Boolean>|ONCE

DISPlay:WINDow:TRACe:PERSistence:AUTO

When AUTO is set to ON, the persistence is determined by the device.

At *RST, this value is set to ON.

8.7.6.5 :STATe <Boolean>

DISPlay:WINDow:TRACe:STATe

Controls whether the TRACe and related information is visible or not.

8.7.6.6 :X

DISPlay:WINDow:TRACe:X

This subsystem controls the setting of the X axis. The X axis is displayed horizontally, and it is usually the independent variable of the data, such as time or frequency. The :X node is used to define the parameters associated with the X axis. It is only available when the format of the data is Cartesian.

8.7.6.6.1 :LAbel <string>

DISPlay:WINDow:TRACe:X:LAbel

Specifies custom axis labeling.

8.7.6.6.2 [:SCALE]

DISPlay:WINDow:TRACe:X:SCALe

The SCALE subtree is used to define which portion of the (TRACe) data is to be displayed, by defining the end points of the axis as they appear in the WINDow. The values are in the same UNITS as the UNIT of data for the x axis. The x scale can be defined in terms of CENTER, LEFT, RIGHT and/or PDIVision, where the following equations express the relationship of these parameters:

$$RIGHT - LEFT = (RIGHT - CENTER) * 2 = PDIVision * \# \text{ of graticule divisions.}$$

If an instrument implements x axis scaling, the commands RIGHT and LEFT must be implemented, while implementing PDIVision and CENTER are optional.

8.7.6.6.2.1 :AUTO <Boolean> | ONCE

DISPlay:WINDow:TRACe:X:SCALe:AUTO

The AUTO ON command sets the display to always configure the scaling on the particular axis to best display the data. This rescaling may affect the values of any of the parameters under the SCALE node. When AUTO is ON the display may be rescaled after each measurement or sweep. Setting AUTO to ONCE, turns auto ON, adjusting the scaling of the axis for one measurement and then turns auto OFF. Turning AUTO ON can change any of the parameters which can be set under the SCALE node.

At *RST AUTO is set to OFF.

8.7.6.6.2.2 :CENTer <numeric_value>

DISPlay:WINDow:TRACe:X:SCALe:CENTer

Set or queries the value represented by the center point of the x-axis. This value may be bounded by the range of the data. When a new CENTER value is entered, the PDIVision value is kept constant and the RIGHT and LEFT values are changed.

At *RST, this parameter value is device dependent.

8.7.6.6.2.3 :LEFT <numeric_value>

DISPlay:WINDow:TRACe:X:SCALe:LEFT

Set or queries the value represented by the minimum (left) edge of the x-axis. This value may be bounded by the range of the data. When a new LEFT value is entered, the PDIVision value is kept constant and the CENTER and RIGHT values are changed.

At *RST, this parameter value is device dependent.

8.7.6.6.2.4 :PDIVision <numeric_value>

DISPlay:WINDow:TRACe:X:SCALE:PDIVision

Set or query the value between two grid graticules (value “per division”). When a new PDIVision value is entered, the parameter set as the PDIVision:LINK is held constant and the other two are changed.

At *RST, this parameter value is device dependent.

8.7.6.6.2.4.1 :LINK LEFT | CENTER | RIGHT

DISPlay:WINDow:TRACe:X:SCALE:PDIVision:LINK

LINK selects the parameter, either CENTER, LEFT or RIGHT that shall not be changed when the PDIVision value is changed. For example, if LINK is set to LEFT, then changing PDIVision shall cause CENTER and RIGHT, not LEFT, to change.

At *RST the value of LINK is device dependent.

8.7.6.6.2.5 :RIGHT <numeric_value>

DISPlay:WINDow:TRACe:X:SCALE:RIGHT

Set or queries the value represented by the maximum (right) edge of the x-axis. This value may be bounded by the range of the data. When a new RIGHT value is entered, the PDIVision value is kept constant and the CENTER and LEFT values are changed.

At *RST, this parameter value is device dependent.

8.7.6.7 :Y

DISPlay:WINDow:TRACe:Y

This subsystem controls the setting of the Y axis. The :Y node is displayed vertically, and it is usually the dependent variable of the data, such as power or voltage. The :Y node is used to define the parameters associated with the Y axis. It is only available when the format of the data is Cartesian.

8.7.6.7.1 :LABel <string>

DISPlay:WINDow:TRACe:Y:LABel

Specifies custom axis labeling.

8.7.6.7.2 :RLIne <Boolean>

DISPlay:WINDow:TRACe:Y:RLIne

Turn on/off a line which is positioned on the graticule at the current Reference POSition.

At *RST RLIne is off.

8.7.6.7.3 [:SCALE]

DISPlay:WINDow:TRACe:Y:SCALE

The SCALE subtree is used to define which portion of the (TRACe) data is to be displayed, by defining the end points of the axis as they appear in the WINDow. The values are in the same UNITS as the UNIT of data for the y axis. The y scale can be defined in terms of TOP,

BOTTOM, PDIVision or Reference LEVEL and Reference POSition. These values are coupled together in the following manner:

$TOP - BOTTOM = PDIVision * \# \text{ of graticule divisions}$

$TOP = \text{Reference LEVEL}$ when Reference POSition is 100 %

$BOTTOM = \text{Reference LEVEL}$ when Reference POSition is 0 %

If an instrument implements y axis scaling, the commands TOP and BOTTOM must be implemented, while implementing the RLEVEL and RPOSITION and PDIVision commands are optional.

8.7.6.7.3.1 :AUTO <Boolean> | ONCE

DISPlay:WINDow:TRACe:Y:SCALE:AUTO

The AUTO ON command sets the display to always configure the scaling on the particular axis to best display the data. This rescaling may affect the values of any of the parameters under the SCALE node. When AUTO is ON the display may be rescaled after each measurement or sweep. Setting AUTO to ONCE, turns auto ON, adjusting the scaling of the axis for one measurement and then turns auto OFF. Turning AUTO ON can change any of the parameters which can be set under the SCALE node.

At *RST AUTO is set to OFF.

8.7.6.7.3.2 :BOTTOM <numeric_value>

DISPlay:WINDow:TRACe:Y:SCALE:BOTTOM

Sets or queries the value represented by the minimum (bottom) edge of the display. The value may be bounded by the range of the data.

8.7.6.7.3.3 :PDIVision <numeric_value>

DISPlay:WINDow:TRACe:Y:SCALE:PDIVision

Set or query the value between two grid graticules (value “per division”). When a new PDIVision value is entered, the current Reference LEVEL is kept the same, while adjusting the top and bottom scaling for the new PDIVision value.

At *RST, this parameter value is device dependent.

8.7.6.7.3.4 :RLEVEL <numeric_value>

DISPlay:WINDow:TRACe:Y:SCALE:RLEVEL

Set or query the value represented at the designated Reference POSition on the y-axis. Setting a new Reference LEVEL does not affect the value of PDIVision.

At *RST, this parameter value is device dependent.

8.7.6.7.3.4.1 :AUTO <Boolean>

DISPlay:WINDow:TRACe:Y:SCALE:RLEVEL:AUTO

Setting RLEVEL:AUTO to ON causes the display to automatically choose a reference level to best display the particular data. When AUTO is ON, a new reference level may be selected each time new data is received.

At *RST AUTO is off.

8.7.6.7.3.5 :RPOsition <numeric_value>

DISPlay:WINDow:TRACe:Y:SCALe:RPOsition

Set or query the point on the y-axis to be used as the reference position as a percentage of the length of the y- axis. The top of the y axis is defined to have a Reference POSition of 100%, while the bottom of the y axis is defined to have a Reference POSition of 0%. The Reference POSition is the point on the y-axis which should equal the Reference LEVel.

At *RST, this parameter value is device dependent.

8.7.6.7.3.6 :TOP <numeric_value>

DISPlay:WINDow:TRACe:Y:SCALe:TOP

Sets or queries the value represented by the top edge of the display. The value may be bounded by the range of the data.

8.7.6.7.4 :SPACing LOGarithmic I LInear

DISPlay:WINDow:TRACe:Y:SPACing

The SPACing command is used to set the scale to either linear or log.

8.7.6.8 :R

DISPlay:WINDow:TRACe:R

This subsystem controls the R axis. The :R axis is displayed radially from the center point and usually represents magnitude of complex data. It is only available when the format of the data is in polar coordinates.

8.7.6.8.1 :LABel <string>

DISPlay:WINDow:TRACe:R:LABel

Specifies custom axis labeling.

8.7.6.8.2 [:SCALe]

DISPlay:WINDow:TRACe:R:SCALe

The SCALe subtree is used to define which portion of the (TRACe) data is to be displayed by defining the end points of the axis as they appear in the WINDow. The values are in the same UNITs as the UNIT of the data for that AXIS.

8.7.6.8.2.1 :AUTO <Boolean> I ONCE

DISPlay:WINDow:TRACe:R:SCALe:AUTO

The AUTO ON command sets the display to always configure the scaling on the particular axis to best display the data. This rescaling may affect the values of any of the parameters under the SCALe node. When AUTO is ON the display may be rescaled after each measurement or sweep. Setting AUTO to ONCE, turns auto ON, adjusting the scaling of the axis for one measurement and then turns auto OFF. Turning AUTO ON can change any of the parameters which can be set under the SCALe node.

At *RST AUTO is set to OFF.

8.7.6.8.2.2 :CPOint <numeric_value>

DISPlay:WINDow:TRACe:R:SCALe:CPOint

Sets or queries the value represented by the minimum (Center POint) of the circular display. The value may be bounded by the range of the data.

1999 SCPI Command Reference

8.7.6.8.2.3 :OEDGE <numeric_value>

DISPlay:WINDow:TRACe:R:SCALe:OEDGE

Sets or queries the value represented by the Outside EDGE of the circular display. The value may be bounded by the range of the data.

8.7.6.8.3 :SPACing LOGarithmic | LInear

DISPlay:WINDow:TRACe:R:SPACing

The SPACing command is used to set the scale to either linear or log.

9 FORMat Subsystem

The FORMat subsystem sets a data format for transferring numeric and array information. This data format is used for both command and response data by those commands that are specifically designated to be affected by the FORMat subsystem. The designation is either given as part of a command description, or in the definition of block or array data used by a command. The data format for command data may override the definition of FORMat if the data received is self typing (indicates its type), for the duration of that data transfer.

KEYWORD	PARAMETER FORM	NOTES
FORMat		
:BORDer	NORMallSWAPped	
[:DATA]	<type> [,<length>]	
:DINTerchange	<Boolean>	
:SREGister	ASCii BINary HEXadecimal OCTal	1992

9.1 :BORDer NORMallSWAPped

FORMat:READings:BORDer

Byte ORDER. Controls whether binary data is transferred in normal or swapped byte order.

At *RST, this value is set to NORMal.

9.2 [:DATA] <type>[,<length>]

FORMat:READings:DATA

The DATA command selects the data format; the <type> and type <length>. Valid types are ASCii, INTeger, UINteger, REAL, HEXadecimal, OCTal, BINary, and PACKed. The <length> parameter is optional for all types; its meaning is dependent on the type selected.

The <length> parameter is an <NRf>. In the query response <length> is returned as <NR1>.

If the type INTeger, UINteger, REAL, or PACKed is selected the data is transferred in a block. The types ASCii, HEXadecimal, OCTal, and BINary are self typing. That is, the syntax designates the type. In these cases, the FORMat subsystem is only necessary to determine the output format.

At *RST, ASCii is selected as the default type, the type length is device-dependent.

The various types are defined as:

<type>	Interpretation:
ASCii	Numeric data is transferred as ASCii bytes in <NR1>, <NR2> or <NR3> format, as appropriate. The numbers are separated by commas as specified in <i>IEEE 488.2</i> . Forgiving listening allows ASCii input to always be valid. The optional <length> parameter specifies the number of significant digits to be returned. A <length> value of zero indicates that the device selects the number of significant digits to be returned. When a <length> of zero has been

1999 SCPI Command Reference

specified, the FORMat[:DATA]? query shall return zero as its second parameter.

BINary	Data is encoded as a non-decimal numeric, base 2, preceded by “#B” as specified in <i>IEEE 488.2</i> . The <length> parameter is the number of binary digits in each number (not including the “#B”).
HEXadecimal	Data is encoded as a non-decimal numeric, base 16, preceded by “#H” as specified in <i>IEEE 488.2</i> . The <length> parameter is the number of hexadecimal digits in each number (not including the “#H”).
INTeger	Data is transferred in a definite length block as signed integers of the length specified (in bits; default is 16). Scaling and offset, if required for interpretation of measurement data, are determined by measurement parameters set in some other subsystem. Valid values of <length> for INTeger data are 8, 16 and 32.
OCTal	Data is encoded as a non-decimal numeric, base 8, preceded by “#Q” as specified in <i>IEEE 488.2</i> . The <length> parameter is the number of octal digits in each number (not including the “#Q”).
PACKed	Data is transferred in a definite length block, in a manner specified in the device documentation. The <length> parameter may be used to convey additional information about the device-dependent format.
REAL	Data is transferred in a definite length block as IEEE 754 floating point numbers of the specified length (in bits; default is 64). Valid values of <length> for REAL data are 32 and 64.
UINTeger	Data is transferred in a definite length block as unsigned integers of the length specified (in bits; default is 16). Scaling and offset, if required for interpretation of measurement data, are determined by measurement parameters set in some other subsystem. Valid values of <length> for UINTeger data are 8, 16, 32 and 64.

9.3 :DINTerchange <Boolean>

FORMat:READings:DINTerchange

Determines whether measurement data is formatted as a <dif_expression>. See Data Interchange Format, Volume 3. When DINTerchange ON is selected, returned data is encapsulated in this structure.

The commands which are affected by this switch are a device dependent subset of those affected by the FORMat:DATA command, but at a minimum shall include SENSE:DATA?, CALCulate:DATA?, TRACe:DATA?, MEASure?, READ?, and FETCH?

The DIF encoding format for its DATA(CURVe(VALues)) block follows that specified in FORMat[:DATA] and FORMat:BORDER.

At *RST this function is set to OFF.

9.4 :SREGister ASCII | BINary | HEXadecimal | OCTal

FORMat:SREGister

This command selects the data type of the response to queries for any CONDition, EVEnt and ENABle register and for PTRansition and NTRansition filters that belong to the status subsystem. Note that this includes the IEEE 488.2 status register queries.

ASCIi	The data is transferred as ASCIi bytes in <NR1> format. This argument is required if the command is implemented.
BINary	The data is encoded as non-decimal numeric, base 2, preceded by ‘#B’ as specified in IEEE 488.2
HEXadecimal	The data is encoded as non-decimal numeric, base 16, preceded by ‘#H’ as specified in IEEE 488.2
OCTal	The data is encoded as non-decimal numeric, base 8, preceded by ‘#Q’ as specified in IEEE 488.2

At *RST, ASCIi is selected as the default response data type.

10 HCOPy

The Hard COPY subsystem controls the setup of plotting and printing to an external device. The Hard COPY subsystem does not perform any data formatting, that is it does not convert the data from one representation to another, such as COMPLEX to POLAR. Instead HCOPY adds the necessary page formatting (dependent upon the hard copy device language) to turn the data into an acceptable form for the hard copy device.

The HCOPY subsystem may FEED data from SENSE or CALCulate, and this data may be printed or plotted. A FEED may be established from a DISPLAY:WINDOW. In this case the HCOPY output shall reflect what is currently displayed in that WINDOW. In particular, any scaling and offset that applies shall be employed.

The selection of the interface and any address information is accomplished in the SYSTEM:COMMunicate subsystem. DATA may be streamed to an external device without sending it through the HCOPY subsystem, where no page formatting commands are needed.

The HCOPY subsystem gives a user several ways to select and initiate a plot or print. A user may request all aspects of the data be plotted or printed, or the user may select a subset that is of interest.

The ITEM subsystem provides the means to select the desired aspects of the data in one of two ways. First, the ITEM of interest can be directly plotted or printed. For example, the TRACE data can be plotted or printed immediately by using the HCOPY:ITEM:...:IMMEDIATE command or the HCOPY:ITEM:...:DATA? query. Alternatively, the user may set the STATE to ON for all the ITEMS of interest, and then use the HCOPY:IMMEDIATE command or the HCOPY:DATA? query to plot or print the selected items in one transaction with the hard copy device.

If the user wishes to plot or print all the data, pointed to by the FEED, and does not want to individually initiate each ITEM or change each ITEM's STATE to ON, the user can send the HCOPY:ITEM:ALL[:IMMEDIATE] command or the HCOPY:ITEM:ALL:DATA? query.

The :HCOPY:SDUMP:IMMEDIATE command and the :HCOPY:SDUMP:DATA? query allow the user to do a Screen DUMP of the whole DISPLAY, mimicking the traditional role of a camera recorder, and provide a simple hard copy scheme for instruments with a DISPLAY.

The HCOPY subsystem provides two schemes, IMMEDIATE and DATA?, for getting hard copy data out of an instrument. First, all of the IMMEDIATE events can be implemented and used where the instrument is capable of directly controlling the interface. For example, RS-232, Centronics, or GPIB where the instrument can talk directly to the hard copy device by becoming the active controller. Any instrument which requires control of the system interface to execute the IMMEDIATE command shall also implement the *PCB common command, see IEEE 488.2 10.21.

IMMEDIATE can also be used on GPIB when the active controller addresses the instrument to talk and the hard copy device to listen while monitoring the data flow by shadow handshaking. The format of the data sent in response to an IMMEDIATE command is not necessarily an IEEE 488.2 format. This behavior is allowed by IEEE 488.2 section 8.1.

WARNING:

The IEEE 488.2 message exchange protocol requirements are violated when the IMMEDIATE command generates a response that is sent when the instrument is addressed to talk. The instrument shall revert to the normal message exchange protocol if it receives a device clear or any data on the system interface. Designers should use the control passing method rather than shadow handshaking method to avoid this escape of the IEEE 488.2 message exchange protocol.

Second, the DATA? query exists for use on buses, such as GPIB, where the data flow from the instrument to the hard copy device by way of a separate controller is acceptable or necessary. It is also useful when the controller wants to store formatted data for later use. The instrument returns the formatted data in an <INDEFINITE LENGTH ARBITRARY RESPONSE DATA> element, see IEEE 488.2 section 8.7.10. This response data element has a leading #0 and a trailing NL^END. The controller could strip the header and trailer and send the remainder to an appropriate printer or plotter. The indefinite form is used because the block's size or nature may prevent the instrument from computing the block's length until after it is sent.

A device may implement IMMEDIATE or DATA? or both of these forms to plot or print results.

The result from HCOpy can also be fed to the Mass MEMory subsystem. The user could then retrieve the results from the stored files at a more convenient time.

KEYWORD	PARAMETER FORM	NOTES
HCOpy		1993
:ABORt		[event;no query] 1993
:DATA?		1993
:DESTination	<data_handle>	[event;no query] 1993
:DEvice		1993
:CMAP		1993
:COLor		1993
:HSL	<hue>,<sat>,<lum>	1993
:RGB	<red>,<green>,<blue>	1993
:DEFault		[event;no query] 1993
:COLor	<Boolean>	1993
:LANGuage	PCL[<n>] HPGL[<n>] POSTscript[<n>]	1993
:MODE	TABLE GRAPh	1993
:RESolution	<numeric_value>	1993
:UNIT	<SUFFIX PROGRAM DATA>	1993
:SPEd	<numeric_value>	1993
:UNIT	<SUFFIX PROGRAM DATA>	1993
:FEED	<data_handle>	1993
[:IMMediate]		1993
:ITEM		1993

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
:ALL		1993
:DATA?		[query only] 1993
[:IMMEDIATE]		[event;no query] 1993
:ANNOtation		1993
:COLor	<numeric_value>	1993
:DATA?		[query only] 1993
[:IMMEDIATE]		[event;no query] 1993
:STATE	<Boolean>	1993
:CUT		1993
:DATA?		[query only] 1993
[:IMMEDIATE]		[event;no query] 1993
:STATE	<Boolean>	1993
:FFEd		1993
:DATA?		[query only] 1993
[:IMMEDIATE]		[event;no query] 1993
:STATE	<Boolean>	1993
:LABel		1993
:COLor	<numeric_value>	1993
:DATA?		[query only] 1993
[:IMMEDIATE]		[event;no query] 1993
:STATE	<Boolean>	1993
:TEXT	<string>	1993
:MENU		1993
:COLor	<numeric_value>	1993
:DATA?		[query only] 1993
[:IMMEDIATE]		[event;no query] 1993
:STATE	<Boolean>	1993
:TDSamp		1993
:COLor	<numeric_value>	1993
:DATA?		[query only] 1993
[:IMMEDIATE]		[event;no query] 1993
:STATE	<Boolean>	1993
[:WINDow]		1993
:DATA?		[query only] 1993
[:IMMEDIATE]		[event;no query] 1993
:STATE	<Boolean>	1993
:TEXT		1993
:COLor	<numeric_value>	1993
:DATA?		[query only] 1993
[:IMMEDIATE]		[event;no query] 1993
:STATE	<Boolean>	1993
:TRACe		1993
:COLor	<numeric_value>	1993
:DATA?		[query only] 1993

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
:GRATicule		1993
:COLor	<numeric_value>	1993
:DATA?		[query only] 1993
[:IMMediate]		[event;no query] 1993
:STATe	<Boolean>	1993
[:IMMediate]		[event;no query] 1993
:LTYPe	SOLid DOTTed DASHed STYLe<n>	1993
:STATe	<Boolean>	1993
:PAGE		1993
:DIMensions		1993
:AUTO	<Boolean>	1993
:LLEFt	<numeric_value>,<numeric_value>	1993
:QUADrant<n>		[event;no query] 1993
:URIGHt	<numeric_value>,<numeric_value>	1993
:LENGth	<numeric_value>	1993
:ORientation	LANDscape PORTrait	1993
:SCALe	<numeric_value>	1993
:SIZE	CUSTom A B C D E A0 A1 A2 A3 A4 B0 B1 B2 B3 B4 B5	1993 1993 1993
:UNIT	<SUFFIX PROGRAM DATA>	1993
:WIDTH	<numeric_value>	1993
:SDUMp		1993
:DATA?		[query only] 1993
[:IMMediate]		[event;no query] 1993

10.1 :ABORt

Hard COPY:ABORt

Abort the current hard copy operation. A new program message automatically aborts any incomplete response from an :HCOPY:....:DATA? query as that is an INTERRUPTED condition.

10.2 :DATA?

Hard COPY:DATA?

This query initiates the plot or print according to the current Hard COPY setup parameters. All of the items under the ITEM node which are turned ON (STATe ON) are encapsulated in an <INDEFINITE LENGTH ARBITRARY RESPONSE DATA> element.

10.3 :DESTination<data_handle>

Hard COPY:DESTination

An event which sets all :FEED connections which had been set to “HCOPY” to “”. The :FEED connection pointed to by <data handle> is set to “HCOPY”. Any <data_handle> which can be used as a parameter to :DESTination shall have a :FEED command under it which can be set to “HCOPY”.

10.4 :DEvice

Hard COPY:DEvice

This subsystem is used to set and query characteristics relating to the hard copy device and its output.

10.4.1 :CMAP

Hard COPY:DEvice:CMAP

The CMAP subsystem controls the physical color associated with logical color numbers in the color map of the display. On a monochrome DISPlay, the CMAP may be used to represent different intensity levels of the same shade. When a color is redefined, all information associated with that color number may change immediately.

Two methods are available for defining a particular color, HSL and RGB models. If both are implemented in a device, then they shall be coupled such that redefining a color with one model shall be reflected in the values of the other model.

At *RST, COLor[1] shall represent “black,” and COLor2 shall represent “white” or the actual color of a monochrome display.

10.4.1.1 :COLor

Hard COPY:DEvice:CMAP:COLor

This subsystem controls the color map for the instrument.

10.4.1.1.1 :HSL <hue>,<sat>,<lum>

Hard COPY:DEvice:CMAP:COLor:HSL

The HSL command sets the instrument’s color map based on the Hue/Saturation/Luminance levels color model. A query of HSL shall return three <NR2>s separated by commas.

Hue ranges from zero to one, circularly, with a value of zero resulting in the same hue as a value of one. The approximate color progression is (starting at zero): red, orange, yellow, green, cyan, blue, magenta, and back to red.

Saturation is the amount of pure color to be mixed with white. The saturation value ranges from zero to one, with zero specifying no color (only white or gray, depending on intensity) and one specifying no white.

Luminance specifies the brightness per unit area of the color. A luminance of zero results in black; a luminance of one results in the brightest color available.

At *RST, these parameters are set to the DEFault values, equivalent to executing DEFault command. The HSL command is coupled to the RGB command. Changing values in either will affect the values of the other.

10.4.1.1.2 :RGB <red>,<green>,<blue>

Hard COPY:DEvice:CMAP:COLor:RGB

The RGB command sets the instrument’s color map based on the Red/Green/Blue color model. A query of RGB shall return three <NR2>s separated by commas.

Red ranges from zero to one, with zero indicating “no red” and one indicating “full intensity red.”

Green ranges from zero to one, with zero indicating “no green” and one indicating “full intensity green.”

Blue ranges from zero to one, with zero indicating “no blue” and one indicating “full intensity blue.”

At *RST, these parameters are set to the DEFault values, equivalent to executing DEFault command. The RGB command is coupled to the HSL command. Changing values in either will affect the values of the other.

10.4.1.2 :DEFault

Hard COPY:DEvice:CMAP:DEFault

Sets the color map to the instrument’s default values for all colors, except that COLor[1] shall be represent “black,” and COLor2 shall represent “white” or the actual color of a monochrome display.

10.4.2 :COLor <Boolean>

Hard COPY:DEvice:COLor

Sets or queries whether color information should be sent as part of the plot or print information, in those languages that support both color and monochrome (black and white).

At *RST, the value of this parameter is OFF.

10.4.3 :LANGUage PCL[<n>] | HPGL[<n>] | POSTscript[<n>]

Hard COPY:DEvice:LANGUage

Selects the control language or data format to be used when sending out plot or print information.

PCL Hewlett-Packard’s Printer Control Language.

HPGL Hewlett-Packard’s Graphics Language.

Information about PCL and HPGL is available by contacting:

HP Peripheral Developer Program
16399 West Bernardo Drive
San Diego, CA 92127-1899
(619) 487-4100

POSTscript Adobe Systems printer language.

Information about PostScript is available by contacting:

Adobe Systems Incorporated
1585 Charleston Road
Mountain View, CA 94039-7900
(415) 961-4400

The optional number, [<n>], following the languages, refers to a specific version of that language.

At *RST, the value of this parameter is device dependent.

10.4.4 :MODE TABLE | GRAPH

Hard COPY:DEvice:MODE

Sets or queries how the data is to be represented in the hard copy. TABLE implies the data is formatted into a table rather than a pictorial form. Only characters appear on the output. GRAPH implies the format is a graph or picture. The graph may be generated using vectors, raster, or other available technology.

At *RST, the value of this setting is device dependent.

10.4.5 :RESolution <numeric_value>

Hard COPY:DEvice:RESolution

Sets or queries the resolution of the result on the hard copy device. This setting is generally used to affect the quality of the output from a raster printer.

At *RST, the value of this parameter is device dependent.

10.4.5.1 :UNIT <SUFFIX PROGRAM DATA>

Hard COPY:DEvice:RESolution:UNIT

Sets or queries the units of the RESolution setting.

At *RST, the value of this parameter is device dependent.

10.4.6 :SPEed <numeric_value>

Hard COPY:DEvice:SPEed

Sets or queries the speed at which vectors are drawn. This setting is generally used to affect the quality of the output from a pen plotter.

At *RST, the value of this parameter is device dependent.

10.4.6.1 :UNIT <SUFFIX PROGRAM DATA>

Hard COPY:DEvice:SPEed:UNIT

Sets or queries the units of the SPEed setting.

At *RST, the value of this parameter is cm/sec.

10.5 :FEED <data_handle>

Hard COPY:FEED

Sets or queries the data flow to be fed into the Hard COPY block.

At *RST, the value of this parameter is device dependent.

10.6 [:IMMediate]

Hard COPY:IMMediate

This command immediately initiates the plot or print according to the current Hard COPY setup parameters. All of the items under the ITEM node which are turned ON (STATe ON) are plotted or printed.

10.7 :ITEM

Hard COPY:ITEM

The :ITEM subsystem collects together commands to configure what items of data will be plotted or printed, and how that data shall appear.

Most data item nodes have :COLor nodes to allow the user to set the color of the specific data item. The parameter to the COLor node selects either the actual pen number of the hardcopy device, or the color map number, whichever value the hardcopy device expects. Some devices may elect not to support some or all of the COLor nodes. For example, a device which has a color DISPlay may elect to use just the existing colors in the DISPlay when plotting or printing a color hardcopy.

10.7.1 :ALL

Hard COPY:ITEM:ALL

This subsystem is used to plot or print all ITEMS.

10.7.1.1 :DATA?

Hard COPY:ITEM:ALL:DATA?

All ITEMS, regardless of their individual states, are encapsulated in an <INDEFINITE LENGTH ARBITRARY RESPONSE DATA> element.

10.7.1.2 [:IMMediate]

Hard COPY:ITEM:ALL:IMMediate

Immediately plot or print all ITEMS, regardless of their individual states.

10

10.7.2 :ANNotation

Hard COPY:ITEM:ANNotation

Configures the plotting or printing of any display annotation.

10.7.2.1 :COLor <numeric_value>

Hard COPY:ITEM:ANNotation:COLor

Sets or queries the color to be used for plotting or printing the display annotation.

At *RST, the value of this parameter is device dependent.

10.7.2.2 :DATA?

Hard COPY:ITEM:ANNotation:DATA?

Returns the display annotation encapsulated in an <INDEFINITE LENGTH ARBITRARY RESPONSE DATA> element.

10.7.2.3 [:IMMediate]

Hard COPY:ITEM:ANNotation:IMMediate

Immediately plot or print the display annotation.

10.7.2.4 :STATe <Boolean>

Hard COPY:ITEM:ANNotation:STATe

Sets or queries whether ANNotation should be plotted or printed when the HCOPY:IMMediate command or HCOPY:DATA? query is sent.

At *RST, the value of this parameter is device dependent.

10.7.3 :CUT

Hard COPY:ITEM:CUT

Configures cutting the page after the plot or print is complete.

10.7.3.1 :DATA?

Hard COPY:ITEM:CUT:DATA?

Returns what would be sent to the hard copy device to cut the page encapsulated in an <INDEFINITE LENGTH ARBITRARY RESPONSE DATA> element.

10.7.3.2 [:IMMediate]

Hard COPY:ITEM:CUT:IMMediate

Immediately cut the page.

10.7.3.3 :STATe <Boolean>

Hard COPY:ITEM:CUT:STATe

Sets or queries whether a page cut should be performed as part of the HCOPY:IMMediate command or HCOPY:DATA? query.

At *RST, the value of this parameter is device dependent.

10.7.4 :FFEd

Hard COPY:ITEM:FFEd

Configures form-feeding after the plot or print is complete.

10.7.4.1 :DATA?

Hard COPY:ITEM:FFEd:DATA?

Returns what would be sent to the hard copy device to do a form feed encapsulated in an <INDEFINITE LENGTH ARBITRARY RESPONSE DATA> element.

10.7.4.2 [:IMMediate]

Hard COPY:ITEM:FFEd:IMMediate

Immediately form-feed the page.

10.7.4.3 :STATe <Boolean>

Hard COPY:ITEM:FFEd:STATe

Sets or queries whether a form feed should be performed as part of the HCOPY:IMMediate command or HCOPY:DATA? query.

At *RST, the value of this parameter is device dependent.

10.7.5 :LABel

Hard COPY:ITEM:LABel

Configures the plotting or printing of the user entered label or title. The text for LABel is not supplied by the source of the FEED connection. The TEXT is supplied by a setting in this subsystem since it could only be generated by the user. It might reflect specific operating conditions or refer to a specific part being tested. Where on the physical page the LABel:TEXT is actually plotted/printed is device dependent.

10.7.5.1 :COLor <numeric_value>

Hard COPY:ITEM:LABel:COLor

Sets or queries the color to be used for plotting or printing the label.

At *RST, the value of this parameter is device dependent.

10.7.5.2 **:DATA?**

Hard COPY:ITEM:LABel:DATA?

Returns the label encapsulated in an <INDEFINITE LENGTH ARBITRARY RESPONSE DATA> element.

10.7.5.3 **[[:IMMediate]**

Hard COPY:ITEM:LABel:IMMediate

Immediately plot or print the label.

10.7.5.4 **:STATe <Boolean>**

Hard COPY:ITEM:LABel:STATe

Sets or queries whether the label should be plotted or printed when the HCOPY:IMMediate command or HCOPY:DATA? query is sent.

At *RST, the value of this parameter is device dependent.

10.7.5.5 **:TEXT <string>**

Hard COPY:ITEM:LABel:TEXT

Sets or queries the contents of the user label. The contents of the <string> is placed in an appropriate place on the hard copy output. The string may contain formatting characters. A textual label containing the empty string ("") shall be plotted/printed as a single blank line.

At *RST, all textual labels are set to the empty string.

10.7.6 **:MENU**

Hard COPY:ITEM:MENU

Configures the plotting or printing of the device menu.

10.7.6.1 **:COLor <numeric_value>**

Hard COPY:ITEM:MENU:COLor

Sets or queries the color to be used for plotting or printing the menu.

At *RST, the value of this parameter is device dependent.

10.7.6.2 **:DATA?**

Hard COPY:ITEM:MENU:DATA?

Returns the menu encapsulated in an <INDEFINITE LENGTH ARBITRARY RESPONSE DATA> element.

10.7.6.3 **[[:IMMediate]**

Hard COPY:ITEM:MENU:IMMediate

Immediately plot or print the menu.

10.7.6.4 **:STATe <Boolean>**

Hard COPY:ITEM:MENU:STATe

Sets or queries whether the menu should be plotted or printed when the HCOPY:IMMediate command or HCOPY:DATA? query is sent.

At *RST, the value of this parameter is device dependent.

10.7.7 :TDSTamp

Hard COPY:ITEM:TDSTamp

Configures the plotting or printing of a time and date stamp.

10.7.7.1 :COLor <numeric_value>

Hard COPY:ITEM:TDSTamp:COLor

Sets or queries the color to be used for plotting or printing the time and date stamp.

At *RST, the value of this parameter is device dependent.

10.7.7.2 :DATA?

Hard COPY:ITEM:TDSTamp:DATA?

Returns the time and date stamp encapsulated in an <INDEFINITE LENGTH ARBITRARY RESPONSE DATA> element.

10.7.7.3 [:IMMediate]

Hard COPY:ITEM:TDSTamp:IMMediate

Immediately plot or print the time and date stamp.

10.7.7.4 :STATe <Boolean>

Hard COPY:ITEM:TDSTamp:STATe

Sets or queries whether the time and date stamp should be plotted or printed when the HCOPY:IMMediate command or HCOPY:DATA? query is sent.

At *RST, the value of this parameter is device dependent.

10.7.8 [:WINDow]

Hard COPY:ITEM:WINDow

Configures the plotting or printing of windows.

10.7.8.1 :DATA?

Hard COPY:ITEM:WINDow:DATA?

Returns the window encapsulated in an <INDEFINITE LENGTH ARBITRARY RESPONSE DATA> element.

10.7.8.2 [:IMMediate]

Hard COPY:ITEM:WINDow:IMMediate

Immediately plot or print the window.

10.7.8.3 :STATe <Boolean>

Hard COPY:ITEM:WINDow:STATe

Sets or queries whether the window should be plotted or printed when the HCOPY:IMMediate command or HCOPY:DATA? query is sent.

At *RST, the value of this parameter is device dependent.

10.7.8.4 :TEXT

Hard COPY:ITEM:WINDow:TEXT

Configures the plotting or printing of text blocks or textual labels.

10.7.8.4.1 **:COLor <numeric_value>**

Hard COPY:ITEM:WINDow:TEXT:COLor

Sets or queries the color to be used for plotting or printing TEXT.

At *RST, the value of this parameter is device dependent.

10.7.8.4.2 **:DATA?**

Hard COPY:ITEM:WINDow:TEXT:DATA?

Returns the menu encapsulated in an <INDEFINITE LENGTH ARBITRARY RESPONSE DATA> element.

10.7.8.4.3 **[[:IMMediate]**

Hard COPY:ITEM:WINDow:TEXT:IMMediate

Immediately plot or print the text blocks or textual labels.

10.7.8.4.4 **:STATe <Boolean>**

Hard COPY:ITEM:WINDow:TEXT:STATe

Sets or queries whether the text blocks or textual labels should be plotted or printed when the HCOPY:IMMediate command or HCOPY:DATA? query is sent.

At *RST, the value of this parameter is device dependent.

10.7.8.5 **:TRACe**

Hard COPY:ITEM:WINDow:TRACe

Configures the plotting or printing of graphical trace data.

10.7.8.5.1 **:COLor <numeric_value>**

Hard COPY:ITEM:WINDow:TRACe:COLor

Sets or queries the color to be used for plotting or printing the trace.

At *RST, the value of this parameter is device dependent.

10.7.8.5.2 **:DATA?**

Hard COPY:ITEM:WINDow:TRACe:DATA?

Returns the trace encapsulated in an <INDEFINITE LENGTH ARBITRARY RESPONSE DATA> element.

10.7.8.5.3 **:GRATicule**

Hard COPY:ITEM:WINDow:TRACe:GRATicule

Configures the plotting or printing of the graticule.

10.7.8.5.3.1 **:COLor <numeric_value>**

Hard COPY:ITEM:WINDow:TRACe:GRATicule:COLor

Sets or queries the color to be used for plotting or printing the graticule.

At *RST, the value of this parameter is device dependent.

10.7.8.5.3.2 **:DATA?**

Hard COPY:ITEM:WINDow:TRACe:GRATicule:DATA?

Returns the graticule encapsulated in an <INDEFINITE LENGTH ARBITRARY RESPONSE DATA> element.

10.7.8.5.3.3 **[[:IMMediate]**

Hard COPY:ITEM:WINDow:TRACe:GRATicule:IMMediate

Immediately plot or print the graticule.

10.7.8.5.3.4 **:STATe <Boolean>**

Hard COPY:ITEM:WINDow:TRACe:GRATicule:STATe

Sets or queries whether the graticule should be plotted or printed when the HCOPY:IMMediate command or HCOPY:DATA? query is sent.

At *RST, the value of this parameter is device dependent.

10.7.8.5.4 **[[:IMMediate]**

Hard COPY:ITEM:WINDow:TRACe:IMMediate

Immediately plot or print the trace.

10.7.8.5.5 **:LTYPe SOLid I DOTTed I DASHed I STYLe<n>**

Hard COPY:ITEM:WINDow:TRACe:LTYPe

Sets or queries the line type to be used for plotting or printing the trace.

For each value of <n> in the STYLe parameter, the appearance of the line type shall be different. The STYLe line types should be chosen so they appear different on the specific hard copy device being used. The actual appearance is device dependent.

At *RST, the value of this parameter is device dependent.

10.7.8.5.6 **:STATe <Boolean>**

Hard COPY:ITEM:WINDow:TRACe:STATe

Sets or queries whether the trace should be plotted or printed when the HCOPY:IMMediate command or HCOPY:DATA? query is sent.

At *RST, the value of this parameter is device dependent.

10.8 **:PAGE**

Hard COPY:PAGE

This subsystem contains commands to set up parameters relating to the medium of the plot or print output.

10.8.1 **:DIMensions**

Hard COPY:PAGE:DIMensions

Sets the plot or print dimensions. Different plotters or printers allow the user to set the page dimensions in different ways, such as length, width dimensions of the page, selecting a standard page size, or setting “corners” of the page in terms of plotter units (that is p1 and p2). A device will likely support the setting of plot or print dimensions in a subset of these ways, depending on which plotter or printer(s) it supports. The interaction of the various dimension commands is device dependent.

10.8.1.1 **:AUTO <Boolean>**

Hard COPY:PAGE:DIMensions:AUTO

When AUTO is on the device selects the default page dimensions. When AUTO is OFF the page dimensions are set by the other setting in this subsystem.

At *RST, the value of this parameter is ON.

10.8.1.2 :LLEft<numeric_value>,<numeric_value>

Hard COPY:PAGE:DIMensions:LLEft

Sets or queries the x,y position of lower left corner of the page. Units are percent of the width and length of the page.

At *RST, the value of this parameter is device dependent.

10.8.1.3 :QUADrant[<n>]

Hard COPY:PAGE:DIMensions:QUADrant

Sets dimensions so that the plot/print occupies one quadrant of the page.

<n>	quadrant		
1	upper right	2	1
2	upper left		
3	lower left		
4	lower right	3	4

This command is an event only.

10.8.1.4 :URIGht<numeric_value>,<numeric_value>

Hard COPY:PAGE:DIMensions:URIGht

Specifies the x,y position of upper right corner of the page. Units are percent of the width and length of the page.

At *RST, the value of this parameter is device dependent.

10.8.2 :LENGth<numeric_value>

Hard COPY:PAGE:LENGth

Sets or queries the length of the page to be plotted or printed.

At *RST, the value of this parameter is device dependent.

10.8.3 :ORIENTATION LANDscape I PORTrait

Hard COPY:PAGE:ORientation

Sets or queries the orientation of the plot or print. Switching between LANDscape and PORTrait rotates the hardcopy result by 90 degrees. No other settings are changed.

At *RST, the value of this parameter is device dependent.

10.8.4 :SCALE<numeric_value>

Hard COPY:PAGE:SCALE

Sets or queries a scaling factor which is applied to the hard copy output. This factor applies to both dimensions. The output from HCOPY shall not attempt to place anything on the page outside the limits set under :DIMensions even when :SCALE is greater than one.

At *RST, the value of this setting is one.

10.8.5 **:SIZE CUSTomIAIBICIDIEIA0IA1IA2IA3IA4IB0IB1IB2IB3IB4IB5**

Hard COPY:PAGE:SIZE

Sets or queries the size of the paper in terms of a standard paper size. If LENGth or WIDth are set, the value of this setting becomes CUSTom.

At *RST, the value of this parameter is device dependent.

10.8.6 **:UNIT <SUFFIX PROGRAM DATA>**

Hard COPY:PAGE:UNIT

Sets or queries the units for LENGth and WIDTh under DIMensions.

At *RST, the value of this parameter is device dependent.

10.8.7 **:WIDTh <numeric_value>**

Hard COPY:PAGE:WIDTh

Sets or queries the width of the page to be plotted or printed.

At *RST, the value of this parameter is device dependent.

10.9 **:SDUMp**

Hard COPY:SDUMp

The Screen DUMp subsystem obtains its FEED from the whole DISPlay subsystem, and shall only be used where a DISPlay subsystem is present in an instrument.

10.9.1 **:DATA?**

Hard COPY:SDUMp:DATA?

Returns the whole DISPlay encapsulated in an <INDEFINITE LENGTH ARBITRARY RESPONSE DATA> element.

10.9.2 **[[:IMMediate]**

Hard COPY:SDUMp:IMMediate

This event shall cause the whole DISPlay to be plotted or printed.

11 INPut Subsystem

The INPut subsystem controls the characteristics of a sensor's input ports.

KEYWORD	PARAMETER FORM	NOTES
INPut		
:ATTenuation	<numeric_value>	
:AUTO	<Boolean> ONCE	
:STATe	<Boolean>	1994
:BIAS		1991
:CURRent		1991
:AC	<numeric_value>	1991
[:DC]	<numeric_value>	1991
[:STATe]	<Boolean>	1991
:TYPE	CURRent VOLTage	1991
:VOLTage		1991
:AC	<numeric_value>	1991
[:DC]	<numeric_value>	1991
:COUPling	ACIDC GROund	1991
:FILTer		
:AWEighting		1991
[:STATe]	<Boolean>	1991
:HPASs		
:FREQuency	<numeric_value>	
[:STATe]	<Boolean>	
[:LPASs]		
:FREQuency	<numeric_value>	
[:STATe]	<Boolean>	
:GAIN	<numeric_value>	
:AUTO	<Boolean> ONCE	
:STATe	<Boolean>	1994
:GUARd	LOW FLOat	
:IMPedance	<numeric_value>	
:LOW	FLOat GROund	
:OFFSet	<numeric_value>	1991
:STATe	<Boolean>	1991
:POLarity	NORMall INVerted	1994
:POLarization	<numeric_value>	1994
:HORizontal		[no query] 1994
:VERTical		[no query] 1994
:POSition		1994
[:X]		1994
:ANGLE		1994
:DIRection	UP DOWN	1994

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
[:IMMediate]	<numeric_value>	1994
:LIMit		1994
:HIGH	<numeric_value>	1994
:LOW	<numeric_value>	1994
[:STATe]	<Boolean>	1994
:OFFSet	<numeric_value>	1994
:VELocity	<numeric_value>	1994
[:DISTance]		1994
:DIRection	UPIDOWN	1994
[:IMMediate]	<numeric_value>	1994
:LIMit		1994
:HIGH	<numeric_value>	1994
:LOW	<numeric_value>	1994
[:STATe]	<Boolean>	1994
:OFFSet	<numeric_value>	1994
:VELocity	<numeric_value>	1994
:Y		1994
:ANGLE		1994
:DIRection	UPIDOWN	1994
[:IMMediate]	<numeric_value>	1994
:LIMit		1994
:HIGH	<numeric_value>	1994
:LOW	<numeric_value>	1994
[:STATe]	<Boolean>	1994
:OFFSet	<numeric_value>	1994
:VELocity	<numeric_value>	1994
[:DISTance]		1994
:DIRection	UPIDOWN	1994
[:IMMediate]	<numeric_value>	1994
:LIMit		1994
:HIGH	<numeric_value>	1994
:LOW	<numeric_value>	1994
[:STATe]	<Boolean>	1994
:OFFSet	<numeric_value>	1994
:VELocity	<numeric_value>	1994
:Z		1994
:ANGLE		1994
:DIRection	UPIDOWN	1994
[:IMMediate]	<numeric_value>	1994
:LIMit		1994
:HIGH	<numeric_value>	1994
:LOW	<numeric_value>	1994
[:STATe]	<Boolean>	1994
:OFFSet	<numeric_value>	1994

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
:VELOCITY	<numeric_value>	1994
[:DISTance]		1994
:DIRection	UPIDOWN	1994
[:IMMediate]	<numeric_value>	1994
:LIMit		1994
:HIGH	<numeric_value>	1994
:LOW	<numeric_value>	1994
[:STATe]	<Boolean>	1994
:OFFSet	<numeric_value>	1994
:VELOCITY	<numeric_value>	1994
[:STATe]	<Boolean>	
:TYPE	<character data>	1992

11.1 :ATTenuation <numeric_value>

INPut:ATTenuation

Sets the input ATTenuation. The default units are in the current relative amplitude units.

This should only be used where the setting is not reflected within the SENSE system. For example, setting the input attenuation to 10 dB will cause the measured signal to decrease in amplitude by 10 dB.

This setting is not coupled to INPut:GAIN. If both ATTenuation and GAIN were set to 10 dB, the net result would be a signal of the original amplitude.

At *RST, this value is device-dependent.

11.1.1 :AUTO <Boolean>IONCE

INPut:ATTenuation:AUTO

AUTO ON changes the ATTenuation to achieve maximum sensitivity without overloading the input channel. Programming a specified ATTenuation sets AUTO OFF.

At *RST, this value is set to ON.

11.1.2 :STATe <Boolean>

INPut:ATTenuation:STATe

Turns the input attenuator ON and OFF. OFF means, that any attenuation circuits are taken out of the signal path, thus, no distortion to the signal will happen.

At *RST, this value is device dependent.

11.2 :BIAS

INPut:BIAS

This subsystem is used to control the bias applied to an external transducer/detector.

11.2.1 :CURRent

INPut:BIAS:CURRent

Specifies the specific bias current being generated.

11.2.1.1 **:AC <numeric_value>**

INPut:BIAS:CURRent:AC

Specifies the AC component of the bias current. The parameter is specified in Amperes.

At *RST, this value is device-dependent.

11.2.1.2 **[:DC] <numeric_value>**

INPut:BIAS:CURRent:DC

Specifies the DC component of the bias current. The parameter is specified in Amperes.

At *RST, this value is device-dependent.

11.2.2 **[:STATe] <Boolean>**

INPut:BIAS:STATe

Sets or queries whether or not input biasing is enabled.

At *RST, this value is device-dependent.

11.2.3 **:TYPE CURRent | VOLTage**

INPut:BIAS:TYPE CURRent

Specifies the type of bias to be generated. CURRent specifies a current bias and VOLTage specifies a voltage bias.

At *RST, this value is device-dependent.

11.2.4 **:VOLTage**

INPut:BIAS:VOLTage

Specifies the specific bias voltage being generated.

11.2.4.1 **:AC <numeric_value>**

INPut:BIAS:VOLTage:AC

Specifies the AC component of the bias voltage. The parameter is specified in Volts.

At *RST, this value is device-dependent.

11.2.4.2 **[:DC] <numeric_value>**

INPut:BIAS:VOLTage:DC

Specifies the DC component of the bias voltage. The parameter is specified in Volts.

At *RST, this value is device-dependent.

11.3 **:COUPling ACIDCIGROund**

INPut:COUPling

Selects AC or DC coupling for the specified signal. DC coupling allows the measurement signal's AC and DC component to pass. AC coupling passes only the measurement signal's AC component. GROund coupling allows no signal to pass at all and grounds the input amplifier without grounding the input signal.

At *RST, this value is device-dependent.

11.4 :FILTer

INPut:FILTer

The FILTer function allows a filter to be inserted in the path of the measurement signal. The FREQuency function allows the programming of a specific break frequency. The STATE command enables the filter.

11.4.1 :AWEighting

INPut:FILTer:AWEighting

Controls the “A” weighting input filter. This is a human hearing compensation filter described in the *IEC Recommendation 179* and *ANSI S1.4*.

11.4.1.1 [:STATe] <Boolean>

INPut:FILTer:AWEighting:STATe

Turns the “A” weighting input filter on and off.

At *RST this value is device-dependent.

11.4.2 :HPASSs

INPut:FILTer:HPASSs

Controls the high pass filter.

11.4.2.1 :FREQuency <numeric_value>

INPut:FILTer:HPASSs:FREQuency

Determines the cutoff frequency of the high pass filter. Default units are Hertz.

At *RST, this value is device-dependent.

11.4.2.2 [:STATe] <Boolean>

INPut:FILTer:HPASSs:STATe

Turns the high pass filter ON and OFF.

At *RST, this value is device-dependent.

11.4.3 [:LPASSs]

INPut:FILTer:LPASSs

Controls the low pass filter.

11.4.3.1 :FREQuency <numeric_value>

INPut:FILTer:LPASSs:FREQuency

Determines the cutoff frequency of the low pass filter. Default units are Hertz.

At *RST, this value is device-dependent.

11.4.3.2 [:STATe] <Boolean>

INPut:FILTer:LPASSs:STATe

Turns the low pass filter ON and OFF.

At *RST, this value is device-dependent.

11.5 :GAIN <numeric_value>

INPut:GAIN

Sets the input GAIN. The default units are in the current relative amplitude units.

This should only be used where the setting is not reflected within the SENSE system. For example, setting the input gain to 10 dB will cause the measured signal to increase in amplitude by 10 dB.

This setting is not coupled to INPut:ATTenuation. If both ATTenuation and GAIN were set to 10 dB, the net result would be a signal of the original amplitude.

At *RST, this value is device-dependent.

11.5.1 :AUTO <Boolean>IONCE

INPut:GAIN:AUTO

AUTO ON changes the GAIN to achieve maximum sensitivity without overloading the input channel.

AT *RST, this value is set to ON.

11.5.2 :STATe <Boolean>

INPut:GAIN:STATe

Turns the input gain (preamplifier) ON and OFF. OFF means, that any preamplifier circuits are taken out of the signal path, thus, no distortion to the signal will happen. OFF is not the same as INPut:GAIN 0 dB, because an isolation preamplifier will have that gain.

At *RST, this value is device dependent.

11.6 :GUARd LOWIFLOat

INPut:GUARd

Connects guard to signal low terminal or allows guard to float.

At *RST, this value is device-dependent.

11.7 :IMPedance <numeric_value>

INPut:IMPedance

Termination input impedance for the input signal specified in Ohms.

At *RST, this value is device-dependent.

11.8 :LOW FLOatIGROund

INPut:LOW

Connects the low signal terminal to ground or allows it to float.

At *RST, this value is device-dependent.

11.9 :OFFSet <numeric_value>

INPut:OFFSet

Sets or queries the input OFFSet. OFFSet may take on positive and negative values. The OFFSet shall be subtracted from the input, thus making OFFSet more positive shall make the signal more negative. The default units are in the current amplitude units

Introducing an OFFSet in the input has the effect of offsetting the input signal, prior to detection in the SENSE block. Any offset introduced in the input block shall not be corrected for in the sensor and subsequently the sensor results. For example, if the current amplitude units are in volts, a value of 5 shall decrease the input signal by 5 volts.

At *RST, the value of OFFSet is device-dependent.

11.9.1 :STATE <Boolean>

INPut:OFFSet:STATe

Turns the effect of OFFSet ON or OFF.

At *RST, STATe shall be OFF.

11.10 :POLarity NORMal|INVerted

INPut:POLarity

Sets or queries the input polarity. NORMal causes the input signal to pass through the INPut block without inversion. INVerted causes the polarity of the input signal to be reversed. Any inversion introduced in the INPut block shall not be corrected for in the sensor and subsequently the sensor results.

At *RST, this value is NORMal.

11.11 :POLarization <numeric_value>

INPut:POLarization

Sets the polarization of the input. The default units are radian.

At *RST, this value is device-dependent.

11.11.1 :HORizontal

INPut:POLarization:HORizontal

Sets the polarization of the input to zero. This command is an event. Therefore it has no associated query or meaning at *RST.

11.11.2 :VERTical

INPut:POLarization:VERTical

Sets the polarization of the input to $\pi/2$ radians. This command is an event. Therefore it has no associated query or meaning at *RST.

11.12 :POSition

INPut:POSition

The characteristics of an incoming signal can be affected by the input's spatial position and orientation relative to the origin. The definition of the origin is application specific. This subsystem uses a righthand coordinate system and has only one origin.

In this subsystem only the IMMEDIATE commands initiate movement.

In this subsystem velocity is always positive.

11.12.1 **[[:X]**

INPut:POSiTion:X

Controls the settings concerning the X-axis.

11.12.1.1 **:ANGLE**

INPut:POSiTion:X:ANGLE

Specifies the angle of rotation around the X-axis.

11.12.1.1.1 **:DIRection UPIDOWN**

INPut:POSiTion:X:ANGLE:DIRection

Controls the direction of the movement when started with INITiate.

At *RST, this is set to UP

11.12.1.1.2 **[[:IMMediate] <numeric_value>**

INPut:POSiTion:X:ANGLE:IMMediate

This command indicates that the input is moved to the specified position without waiting for further commands.

*RST has no effect on the IMMediate value except that movement shall stop.

11

11.12.1.1.3 **:LIMit**

INPut:POSiTion:X:ANGLE:LIMit

Sets the maximum bounds on the physical POSition.

11.12.1.1.3.1 **:HIGH <numeric_value>**

INPut:POSiTion:X:ANGLE:LIMit:HIGH

Sets the maximum for the POSition.

At *RST, this value is set to the clockwise physical limit.

11.12.1.1.3.2 **:LOW <numeric_value>**

INPut:POSiTion:X:ANGLE:LIMit:LOW

Sets the minimum for the POSition.

At *RST, this value is set to the counterclockwise physical limit.

11.12.1.1.3.3 **:STATe <Boolean>**

INPut:POSiTion:X:ANGLE:LIMit:STATe

Controls whether the LIMit is enabled.

At *RST, this value is set to ON.

11.12.1.1.4 **:OFFSet <numeric_value>**

INPut:POSiTion:X:ANGLE:OFFSet

This defines a single value that is subtracted from the physical POSition.

At *RST, this value is device-dependent.

11.12.1.1.5 :VELocity <numeric_value>

INPut:POStion:X:ANGLe:VELocity

Controls the velocity of the angular rotation around the x-axis.

At *RST, this value is device dependent.

11.12.1.2 [:DISTance]

INPut:POStion:X:DISTance

Controls the settings in linear direction of the X-axis.

11.12.1.2.1 :DIRection UPIDOWN

INPut:POStion:X:DISTance:DIRection

Controls the direction of the movement when started with INITiate.

At *RST, this is set to UP

11.12.1.2.2 [:IMMediate] <numeric_value>

INPut:POStion:X:DISTance:IMMediate

This command indicates that the input is moved to the specified position without waiting for further commands.

*RST has no effect on the IMMediate value except that movement shall stop.

11.12.1.2.3 :LIMit

INPut:POStion:X:DISTance:LIMit

Sets the maximum bounds on the physical POSition.

11.12.1.2.3.1 :HIGH <numeric_value>

INPut:POStion:X:DISTance:LIMit:HIGH

Sets the maximum for the POSition.

At *RST, this value is set to the upper physical limit.

11.12.1.2.3.2 :LOW <numeric_value>

INPut:POStion:X:DISTance:LIMit:LOW

Sets the minimum for the POSition.

At *RST, this value is set to the lower physical limit.

11.12.1.2.3.3 :STATe <Boolean>

INPut:POStion:X:DISTance:LIMit:STATe

Controls whether the LIMit is enabled.

At *RST, this value is set to ON.

11.12.1.2.4 :OFFSet <numeric_value>

INPut:POStion:X:DISTance:OFFSet

This defines a single value that is subtracted from the physical POSition.

At *RST, this value is device-dependent.

11.12.1.2.5 :VELocity <numeric_value>

INPut:POStion:X:DISTance:VELocity

Controls the velocity of the displacement on the x-axis.

At *RST, this value is device dependent.

11.12.2 :Y

INPut:POStion:Y

Controls the settings concerning the Y-axis.

11.12.2.1 :ANGLE

INPut:POStion:Y:ANGLE

Specifies the angle of rotation around the Y-axis.

11.12.2.1.1 :DIRection UPIDOWN

INPut:POStion:Y:ANGLE:DIRection

Controls the direction of the movement when started with INITiate.

At *RST, this is set to UP

11.12.2.1.2 [:IMMediate] <numeric_value>

INPut:POStion:Y:ANGLE:IMMediate

This command indicates that the input is moved to the specified position without waiting for further commands.

*RST has no effect on the IMMediate value except that movement shall stop.

11.12.2.1.3 :LIMit

INPut:POStion:Y:ANGLE:LIMit

Sets the maximum bounds on the physical POSition.

11.12.2.1.3.1 :HIGH <numeric_value>

INPut:POStion:Y:ANGLE:LIMit:HIGH

Sets the maximum for the POSition.

At *RST, this value is set to the clockwise physical limit.

11.12.2.1.3.2 :LOW <numeric_value>

INPut:POStion:Y:ANGLE:LIMit:LOW

Sets the minimum for the POSition.

At *RST, this value is set to the counterclockwise physical limit.

11.12.2.1.3.3 :STATe <Boolean>

INPut:POStion:Y:ANGLE:LIMit:STATe

Controls whether the LIMit is enabled.

At *RST, this value is set to ON.

11.12.2.1.4 :OFFSet <numeric_value>

INPut:POStion:Y:ANGLE:OFFSet

This defines a single value that is subtracted from the physical POSition.

At *RST, this value is device-dependent.

11.12.2.1.5 :VELocity <numeric_value>

INPut:POSition:Y:ANGLE:VELocity

Controls the velocity of the angular rotation around the y-axis.

At *RST, this value is device dependent.

11.12.2.2 [:DISTance]

INPut:POSition:Y:DISTance

Controls the settings in linear direction of the Y-axis.

11.12.2.2.1 :DIRection UPIDOWN

INPut:POSition:Y:DISTance:DIRection

Controls the direction of the movement when started with INITiate.

At *RST, this is set to UP

11.12.2.2.2 [:IMMediate] <numeric_value>

INPut:POSition:Y:DISTance:IMMediate

This command indicates that the input is moved to the specified position without waiting for further commands.

*RST has no effect on the IMMediate value except that movement shall stop.

11.12.2.2.3 :LIMit

INPut:POSition:Y:DISTance:LIMit

Sets the maximum bounds on the physical POSition.

11.12.2.2.3.1 :HIGH <numeric_value>

INPut:POSition:Y:DISTance:LIMit:HIGH

Sets the maximum for the POSition.

At *RST, this value is set to the upper physical limit.

11.12.2.2.3.2 :LOW <numeric_value>

INPut:POSition:Y:DISTance:LIMit:LOW

Sets the minimum for the POSition.

At *RST, this value is set to the lower physical limit.

11.12.2.2.3.3 :STATe <Boolean>

INPut:POSition:Y:DISTance:LIMit:STATe

Controls whether the LIMit is enabled.

At *RST, this value is set to ON.

11.12.2.2.4 :OFFSet <numeric_value>

INPut:POSition:Y:DISTance:OFFSet

This defines a single value that is subtracted from the physical POSition.

At *RST, this value is device-dependent.

11.12.2.2.5 :VELocity <numeric_value>

INPut:POSition:Y:DISTance:VELocity

Controls the velocity of the displacement on the y-axis.

At *RST, this value is device dependent.

11.12.3 :Z

INPut:POSition:Z

Controls the settings concerning the Z-axis

11.12.3.1 :ANGLE

INPut:POSition:Z:ANGLE

Specifies the angle of rotation around the Z-axis.

11.12.3.1.1 :DIRection UPIDOWN

INPut:POSition:Z:ANGLE:DIRection

Controls the direction of the movement when started with INITiate.

At *RST, this is set to UP

11.12.3.1.2 [:IMMediate] <numeric_value>

INPut:POSition:Z:ANGLE:IMMediate

This command indicates that the input is moved to the specified position without waiting for further commands.

*RST has no effect on the IMMediate value except that movement shall stop.

11.12.3.1.3 :LIMit

INPut:POSition:Z:ANGLE:LIMit

Sets the maximum bounds on the physical POSition.

11.12.3.1.3.1 :HIGH <numeric_value>

INPut:POSition:Z:ANGLE:LIMit:HIGH

Sets the maximum for the POSition.

At *RST, this value is set to the clockwise physical limit.

11.12.3.1.3.2 :LOW <numeric_value>

INPut:POSition:Z:ANGLE:LIMit:LOW

Sets the minimum for the POSition.

At *RST, this value is set to the counterclockwise physical limit.

11.12.3.1.3.3 :STATe <Boolean>

INPut:POSition:Z:ANGLE:LIMit:STATe

Controls whether the LIMit is enabled.

At *RST, this value is set to ON.

11.12.3.1.4 :OFFSet <numeric_value>

INPut:POSition:Z:ANGLE:OFFSet

This defines a single value that is subtracted from the physical POSition.

At *RST, this value is device-dependent.

11.12.3.1.5 :VELocity <numeric_value>

INPut:POStion:Z:ANGLE:VELocity

Controls the velocity of the angular rotation around the z-axis.

At *RST, this value is device dependent.

11.12.3.2 [:DISTance]

INPut:POStion:Z:DISTance

Controls the settings in linear direction of the Z-axis.

11.12.3.2.1 :DIRection UPIDOWN

INPut:POStion:Z:DISTance:DIRection

Controls the direction of the movement when started with INITiate.

At *RST, this is set to UP

11.12.3.2.2 [:IMMediate]<numeric_value>

INPut:POStion:Z:DISTance:IMMediate

This command indicates that the input is moved to the specified position without waiting for further commands.

*RST has no effect on the IMMediate value except that movement shall stop.

11.12.3.2.3 :LIMit

INPut:POStion:Z:DISTance:LIMit

Sets the maximum bounds on the physical POSition.

11.12.3.2.3.1 :HIGH <numeric_value>

INPut:POStion:Z:DISTance:LIMit:HIGH

Sets the maximum for the POSition.

At *RST, this value is set to the upper physical limit.

11.12.3.2.3.2 :LOW <numeric_value>

INPut:POStion:Z:DISTance:LIMit:LOW

Sets the minimum for the POSition.

At *RST, this value is set to the lower physical limit.

11.12.3.2.3.3 :STATe <Boolean>

INPut:POStion:Z:DISTance:LIMit:STATe

Controls whether the LIMit is enabled.

At *RST, this value is set to ON.

11.12.3.2.4 :OFFSet <numeric_value>

INPut:POStion:Z:DISTance:OFFSet

This defines a single value that is subtracted from the physical POSition.

At *RST, this value is device-dependent.

11.12.3.2.5 :VELocity <numeric_value>

INPut:POStion:Z:DISTance:VELocity

Controls the velocity of the displacement on the z-axis.

At *RST, this value is device dependent.

11.13 [:STATe] <Boolean>

INPut:STATe

The STATe command connects the input terminal to the measurement signal path when ON. It can be turned OFF to allow maximum isolation of the measurement signal. Note that this does not mean ground the input signal.

At *RST, STATe ON shall be selected.

11.14 :TYPE <character data>

INPut:TYPE

Selects the input characteristic to be used if more than one is available.

The following characteristics are currently defined for instruments:

- BALanced – balanced or differential input.
- UNBalanced – unbalanced with respect to ground.

At *RST the TYPE is device-dependent.

12 INSTRument Subsystem

An instrument may support multiple logical instruments. A dual channel power supply for example is considered as two logical instruments. The logical instruments are not required to have identical functionality, nor does the functionality have to be available simultaneously. The INSTRument subsystem provides a mechanism to identify and select logical instruments by either name or number. In this way a particular logical instrument could be selected and it would respond to commands, such as MEASure, in the same manner as a dedicated instrument having the same functionality as the logical instrument. The INSTRument identifiers have no fixed correspondence to the numeric suffixes allowed with command headers.

KEYWORD	PARAMETER FORM	NOTES
INSTRument		
:CATalog?		[query only] 1994
:FULL?		[query only] 1994
:COUPle		
[:<subsystem>]	ALL NONE <list>	
:DEFine		1999
:GROup	<identifier>,<identifier_list>	1999
[:NAME]	<identifier>,<numeric_value>	1999
:DELete		1999
[:NAME]	<identifier>	[event; no query] 1994
:ALL		[event; no query] 1994
:NSElect	<numeric_value>	
[:SElect]	<identifier>	1999
:STATe	<Boolean>	

12

12.1 :CATalog?

INSTRument:CATalog?

The CATalog query returns a comma-separated list of strings which contains the names of all logical instruments and groups. If no logical instruments are defined, a single null string is returned.

Some instruments may have one or more intrinsic (permanent) <identifier>'s. These are listed along with any user-named <identifier>'s.

12.1.1 :FULL?

INSTRument:CATalog?:FULL?

CATalog:FULL? query returns a list of string - number pairs. The string contains the name of the logical instrument. The immediately following NR1-formatted number is its associated logical instrument number. All response data elements are comma separated. If no logical instrument is defined, a null string followed by a zero is returned.

Intrinsic <identifier>'s associated with logical instrument numbers are listed along with any user-named <identifier>'s and numbers.

12.2 :COUPle[:<subsystem>] ALL|NONE|<list>

INSTrument:COUPle

Defines a coupling between various logical instruments or between various channels within an instrument. The COUPle command consists of an optional subsystem node followed by a single parameter.

The subsystem node denotes a SCPI subsection which can either affect each instrument individually, or affect all instruments/channels collectively. This node is used only when the user can turn this coupling ON or OFF. Predefined couplings need not use this node. If no node follows the COUPle command, then the entire instrument is assumed to be coupled. Coupling a node lower in the tree overrides any couplings which are assigned at a higher level.

The parameter indicates to which logical instruments the specified coupling is to apply. ALL indicates that specified coupling is to apply to all logical instruments. NONE indicates that specified coupling is to be removed. A list of instruments specifies a particular set of logical instruments to be coupled.

Within SCPI certain subsystems, such as SYSTem, STATus, MEMory, MMEMory, TEST and DIAGnostic, are considered common to all logical instruments. That is, they appear coupled across all logical instruments, and they cannot be uncoupled. The IEEE common commands are assumed to always operate on all logical instruments.

At *RST, all subsystems are uncoupled as far as the instrument hardware allows.

12.3 :DEFine

INSTrument:DEFine

12.3.1 :GROup <identifier>,<identifier_list>

INSTrument:DEFine:GROup

Defines or queries a group of instruments. The <identifier> is character data. The <identifier_list> is a comma-separated list of strings which contains the names of logical instruments. The defined instruments can be queried using :INSTrument:DEFine:GROup? <identifier>. No instrument selection is performed by this command. Any instrument implementing the :GROup command shall define which commands are affected by groups.

When groups are defined and used, and data is set or queried, the following rules shall apply unless otherwise noted:

- When setting a list of values, each value will apply to the corresponding instrument in the selected group. If the number of values does not correspond to the number of instruments selected, no action is taken and error -115 is reported.

- When a set of values is required by a command, one set of values may be sent for each instrument in the group. For instance,

```
FOO a,b;
```

becomes

```
FOO a1,b1,a2,b2,...
```

- If a single parameter value is supplied for a command that expects one value per group member, the value is applied to every instrument in the currently selected group.
- Note that commands that take an arbitrary number of optional parameters need to have some way to determine how many parameters apply to each member of the group.
- When queried, the command shall return the value(s) for each instrument in the currently selected group.

When groups are defined and used, events will be sent to all instruments in the currently selected group.

12.3.2 **[[:NAME] <identifier>, <numeric_value>**

INSTrument:DEfine[:NAME]

The command form defines a logical instrument name, <identifier>, and associates it with a logical instrument number, <numeric_value>. The <identifier> is in character data format.

The query form takes only <identifier> as a parameter and returns the logical instrument number.

12.4 **:DELeTe**

INSTrument:DELeTe

The DELeTe subsystem disassociates <identifier>'s from logical instrument numbers or groups of instrument <identifier>'s. After the :DELeTe command executes, the <identifier> is no longer accessible until it is redefined. The instrument behaves as if the <identifier> never existed.

12.4.1 **:ALL**

INSTrument:DELeTe:ALL

This command is an event that disassociates all identifiers from their logical instrument numbers or identifier list, except for the currently selected identifier and any intrinsic identifier definitions. There is no query form.

The situation of “nothing to delete” does not cause an execution error.

This command is an event, and therefore has no *RST value associated with it.

*RST has no effect.

12.4.2 **[[:NAME] <identifier>**

INSTrument:DELeTe:NAME

This command is an event, which disassociates the named <identifier> from its logical instrument number or identifier list. There is no query form.

Attempting to delete the currently selected logical instrument or any intrinsic (permanent) <identifier> - number associations results in an -221, (settings conflict) execution error.

This command is an event, and therefore has no *RST value associated with it.

12.5 :NSElect <numeric_value>

INSTrument:NSElect

This command is used in conjunction with the SElect command. It serves the same purpose, except that it uses a numeric value instead of the identifier used in the SElect command. When queried it shall return the logical instrument number. Note that the numbering used for logical instruments shall directly correspond to the numbers used in status reporting for multiple instruments; specifically the STATus:QUEStionable:INSTrument and STATus:OPERation:INSTrument commands. By selecting an instrument with either SElect or NSElect and querying the selected instrument with the other command, the equivalence between the instrument number and identifier can be determined. For example, by selecting a logical instrument using the SElect command with its name identifier and then querying which logical instrument is selected with the NSElect command, the corresponding numeric value shall be returned.

Assigning numeric values to logical instrument identifiers is not appropriate for instrument groups. Therefore, NSElect returns an error if a group has been selected.

12.6 [:SElect] <identifier>

INSTrument:SElect

This command selects a path or entire instrument or group as the default. When a logical instrument or a group is selected, all other logical instruments or groups are unavailable for programming until selected. If group logical identifiers are not used, and either :SElect or :NSElect commands are implemented, then both of the commands must be implemented.

Character data names may be predefined by the instrument.

At *RST, a device-dependent instrument is selected.

12.7 :STATe <Boolean>

INSTrument:STATe

Turns the selected logical instrument ON or OFF. A logical instrument does not have to be turned OFF before another logical instrument is selected. That is, several logical instruments may be active, while only one may be selected. When an instrument is active, measurements occur and signals are generated. When inactive, measurements do not occur and signals are not generated. When a logical instrument is selected, yet is inactive, all commands shall be processed so that the logical instrument shall reflect the state changes requested when the logical instrument is turned on. The exception to this are commands which would normally cause trigger events in an active logical instrument. Any of these commands shall cause an execution error (-200) to be generated.

At *RST, this value is device-dependent.

13 MEMory Subsystem

The purpose of the memory subsystem is to manage instrument memory. This specifically excludes memory used for mass storage, which is defined in the MMEMory Subsystem. Instrument memory may be used to store several data types (e.g. ASCii, BINary, instrument STATe, TRACe or MACRo.) The data types available for a given instrument are device-dependent.

An instrument may support either fixed or dynamic allocation of instrument memory, between the different data types. With fixed allocation a certain amount of memory is dedicated to each of the supported data types. The dedicated memory is unavailable for use by any other data type. Alternatively, with dynamic allocation, instrument memory is shared between the various data types supported on a demand basis.

In the MEMory:TABLE subsystems, many fundamental measurement are defined as subnodes below :TABLE, such as MEMory:TABLE:FREQuency, :VOLTage, :POWer, etc. These nodes define standardized entries in the table (note that these entries can be conceptualized as either a column or a row). While it makes sense to define widely used entries in the SCPI Volume 2 standard, there is sometimes a desire to standardize very instrument class specific names. Where instrument class specific names require standardization, and little or no overlap is expected with other instrument classes, these instrument class specific MEMory:TABLE nodes are defined in the Volume 4, Instrument classes chapter to which they apply.

As an example, the SCPI commands for Chassis Dynamometers :FORCe :SPEed and TIME are defined in Volume 2 under :MEMory:TABLE. These are fundamental measurements, and might be used by many instruments. However Deceleration RATE is deemed to be specific to dynamometers and is defined in Volume 4.

KEYWORD	PARAMETER FORM	NOTES
MEMory		
:CATalog		
[:ALL]?		[query only]
:ASCii?		[query only]
:BINary?		[query only]
:MACRo?		[query only]
:STATe?		[query only]
:TABLE?		[query only] 1992
:CLEar		1993
[:NAME]	<name>	[event; no query] 1993
:TABLE	<name>	[event; no query] 1993
:COPY		1993
[:NAME]	<name>,<name>	[event; no query] 1993
:TABLE	<name>	[event; no query] 1993
:DATA	<name>,<data>	1994
:DELeTe		
:ALL		[event; no query]

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
[:NAME]	<name>	[event; no query]
:EXCHange		1993
[:NAME]	<name>,<name>	[event; no query] 1993
:TABLe	<name>	[event; no query] 1993
:FREE		
[:ALL]?		[query only]
:ASCii?		[query only]
:BINary?		[query only]
:MACRo?		[query only]
:STATe?		[query only]
:TABLe?		[query only] 1992
:NSTates?		[query only]
:STATe		1994
:CATalog?		[query only] 1994
:DEFine	<name>,<register_number>	1994
:DEFine?	<name>	1994
:TYPE?	<name>	[query only]
:TABLe		1992
:BNUMber<n>	<numeric_value>{,<numeric_value>}	1999*
POINTs		[query only] 1999*
:CCURve	<numeric_value>{,<numeric_value>}	1999*
POINTs		[query only] 1999*
:CONCetration	<numeric_value>{,<numeric_value>}	1999*
POINTs		[query only] 1999*
:CONDition		1993
[:MAGNitude]	<Boolean> {,<Boolean>}	1993
:POINTs?		[query only] 1993
:CPOint	<numeric_value>{,<numeric_value>}	1999*
POINTs		[query only] 1999*
:CURRent		1992
[:MAGNitude]	<numeric_value>{,<numeric_value>}	1992
:POINTs?		[query only] 1992
:PHASe	<numeric_value>{,<numeric_value>}	1992
:POINTs?		[query only] 1992
:DEFine	<identifier>[,<identifier>]	1999
:DFACTory	<numeric_value>{,<numeric_value>}	1999*
POINTs		[query only] 1999*
:DLASt	<numeric_value>{,<numeric_value>}	1999*
POINTs		[query only] 1999*
:DLINearize	<numeric_value>{,<numeric_value>}	1999*
POINTs		[query only] 1999*
:EXPEcted	<numeric_value>{,<numeric_value>}	1999*
POINTs		[query only] 1999*
:FORCe		1999

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES	
[[:MAGNitude]	<numeric_value> {,<numeric_value>}		1999
:POINts?		[query only]	1999
:FREQuency	<numeric_value>{,<numeric_value>}		1992
:POINts?		[query only]	1992
:LABel	<string>{,<string>}		1999*
POINts		[query only]	1999*
:LLIMit	<numeric_value>{,<numeric_value>}		1999
:POINts?		[query only]	1999
:LOG	<string>{,<string>}		1999*
POINts		[query only]	1999*
:LOSS			1992
[[:MAGNitude]	<numeric_value>{,<numeric_value>}		1992
:POINts?		[query only]	1992
:PHASe	<numeric_value>{,<numeric_value>}		1992
:POINts?		[query only]	1992
:NCURve	<numeric_value>{,<numeric_value>}		1999*
POINts		[query only]	1999*
:POWer			1992
[[:MAGNitude]	<numeric_value>{,<numeric_value>}		1992
:POINts?		[query only]	1992
:RAW	<numeric_value>{,<numeric_value>}		1999*
POINts		[query only]	1999*
:RESistance			1992
[[:MAGNitude]	<numeric_value>{,<numeric_value>}		1992
:POINts?		[query only]	1992
:SElect	<identifier>		1999
:SPEed			1999
[[:MAGNitude]	<numeric_value> {,<numeric_value>}		1999
:POINts?		[query only]	1999
:TIME			1999
[[:MAGNitude]	<numeric_value> {,<numeric_value>}		1999
:POINts?		[query only]	1999
:TOLerance	<numeric_value>{,<numeric_value>}		1999*
POINts		[query only]	1999*
:ULIMit	<numeric_value>{,<numeric_value>}		1999
:POINts?		[query only]	1999
:VOLTage			1992
[[:MAGNitude]	<numeric_value>{,<numeric_value>}		1992
:POINts?		[query only]	1992
:PHASe	<numeric_value>{,<numeric_value>}		1992
:POINts?		[query only]	1992
:WFACTor	<numeric_value>{,<numeric_value>}		1999*
POINts		[query only]	1999*

1999 SCPI Command Reference

* **NOTE:** The commands marked with an asterisk query or set values inside of predefined emissions bench memory tables. The tables involved are as follows:

Table	Column Name(s)	Description
CPIxxRyy	CPOint,TOLerance,WFACTORWT	Linearization input cut point tables
GasBottle	BNUmber,CONCenTration,LABel	Gas bottle table
LSIxxRyy	CPOint,EXPeCted,RAW,CCURve,NCURve	Linearization results tables
VERify	BNUmber1,BNUmber2,...BNUmber	Verify gas table
QLIxxRyy	LOG	Linearization log tables
QNIxxRyy	LOG	NOx efficiency log tables
SDRift	DFACTORy,DLAST,DLINEarize	Span Drift results table
SpanGas	BNUmber1,BNUmber2,...BNUmber	Span Gas table
TopGas	BNUmber1,BNUmber2,...BNUmber	Top Gas table
ZDRift	DFACTORy,DLAST,DLINEarize	Zero Drift results table
ZeroGas	BNUmber1,BNUmber2,...BNUmber	Zero Gas table

The <name> parameter is used to define a user label, which shall be <character program data>. This label can then be used by the application in place of the specified data defined here or in other subsystems. The <name> parameter is sent as <character data>, rather than as a string, because that is the simplest *IEEE 488.2* type that meets the requirements. When a name is returned, for example, in response to a CATalog? query, it must be sent as <string response data>, so that a null string (“”) can indicate that nothing is defined.

The MEMory capabilities of an instrument are not part of the instrument state and are not affected by *RST or *RCL. They may survive a power cycle, but this capability is optional. If the instrument implements SYSTem:SECure mode, MEMory:DELeTe:ALL must be executed when SYSTem:SEC OFF is executed.

13.1 :CATalog

MEMory:CATalog

The CATalog queries return information on the current contents and state of the instrument's memory. The returned data shall be in the following form:

<numeric_value>,<numeric_value>{,<string>}

The instrument shall return two numeric parameters and as many strings as there are items in the directory list. The first parameter shall indicate the total amount of storage currently used, in bytes. The second parameter shall indicate the total amount of storage available, also in bytes. Each string parameter returned shall indicate the name, type and size of one in the directory list, thus:

<name>,<type>,<size>

The <name>, <type> and <size> are all character data. The <name> is the exact name of a file as it appears in the directory list, including an extension if present. The <size> provides the size of the file in bytes. The <type> is indicated by the name of the major subsystem or one of the following:

- ASCii — An ASCII text file

- BINary — A binary file
- MACRo — An instrument macro
- STATE — An instrument state
- TABLE — A table of data

The qualified CATalog query for TRACe memory is specified in the TRACe subsystem.

13.1.1 **[[:ALL]]?**

MEMory:CATalog:ALL?

If the CATalog query is qualified by the optional ALL node then the query shall return a directory list and memory sizes related to all allocatable items in instrument memory. Note that this includes all allocatable objects, such as trace, etc.

13.1.2 **:ASCii?**

MEMory:CATalog:ASCii?

If the CATalog query is qualified by the ASCii node then the query shall return the directory list related only to ASCii items in instrument memory. The memory usage and availability parameters shall reflect values that correspond to values for ASCii data. In an instrument that has dedicated memory for ASCii data types, values reported shall be different from the total given by a CATalog of ALL types.

13.1.3 **:BINary?**

MEMory:CATalog:BINary?

If the CATalog query is qualified by the BINary node then the query shall return the directory list related only to BINary items in instrument memory. The memory usage and availability parameters shall reflect values that correspond to values for BINary data. In an instrument that has dedicated memory for BINary data types, values reported shall be different from the total given by a CATalog of ALL types.

13.1.4 **:MACRo?**

MEMory:CATalog:MACRo?

If the CATalog query is qualified by the MACRo node then the query shall return the directory list related only to MACRo items in instrument memory. The memory usage and availability parameters shall reflect values that correspond to values for MACRo data. In an instrument that has dedicated memory for MACRo data types, values reported shall be different from the total given by a CATalog of ALL types.

13.1.5 **:STATE?**

MEMory:CATalog:STATE?

If the CATalog query is qualified by the STATE node then the query shall return the directory list related only to STATE items in instrument memory. The memory usage and availability parameters shall reflect values that correspond to values for STATE data. In an instrument that has dedicated memory for STATE data types, values reported shall be different from the total given by a CATalog of ALL types.

13.1.6 :TABLE?

MEMory:CATalog:TABLE?

If the CATalog query is qualified by the TABLE node then the query shall return the directory list related only to TABLE items in instrument memory. The memory usage and availability of parameters shall reflect values that correspond to values for TABLE data. In an instrument that has dedicated memory for TABLE data types, the values reported shall be different from the total given by a CATalog of ALL types.

13.2 :CLEAr

MEMory:CLEAr

The commands from this subsystem remove the data that is stored in instrument memory. This subsystem differs from the DELEte subsystem in that CLEAr removes the data contents but does not affect the memory definitions such as memory type and structure, available space, and the name of the associated instrument memory.

13.2.1 [:NAME] <name>

MEMory:CLEAr:NAME

This command is a default node and removes all data from the instrument memory associated with <name>. The NAME command does not affect the related memory definitions.

This command is an event with no associated query form and *RST condition.

13.2.2 :TABLE

MEMory:CLEAr:TABLE

The CLEAr:TABLE command removes the data values from all element lists in the SELEcted TABLE. This command does not affect the space and structure of the table as specified with the table's DEFIne command. Executing a POINts? query for any table element after the CLEAr:TABLE command, would return 0.

This command is an event with no associated query form and *RST condition.

13.3 :COPI

MEMory:COPI

The COPY subsystem copies the data from an existing instrument memory item into another, named memory item.

13.3.1 [:NAME] <name>,<name>

MEMory:COPI:NAME

This command is a default node that copies the data contents from the memory item associated with the <name> specified in the first parameter to a memory item associated with the <name> specified by the second parameter.

When the memory item to which data is copied is of a different type, or has an incompatible structure, an execution error - 294 (Incompatible type) is generated. When the specified destination name does not exist, the device may create a new memory item of the same type and structure as the source. In that case, the destination name is to be associated with the new created memory item.

This command is an event with no associated query form and *RST condition.

13.3.2 :TABLE <name>

MEMory:COPY:TABLE

The COPY:TABLE command copies the data values of all element lists from the SElected TABLE to the memory table associated with the specified <name>.

When the memory item to which data is copied is of a different type, or has an incompatible structure, an execution error -294 (Incompatible type) is generated. When the TABLE name does not exist, the device may define a new memeory table with the same structure as the selected table. In that case, the specified TABLE name is associated with the new created memory table.

This command is an event with no associated query form and *RST condition.

13.4 :DATA<name>,<data>

MEMory:DATA

The command form is MEMory:DATA <name>,<data>. The command loads <data> into the memory location <name>. <data> is in 488.2 block format. <name> is character data.

The query form is MEMory:DATA? <name> with the response being the associated <data> in block format.

MEMory:DATA? response is not affected by the FORMat subsystem.

13.5 :DELeTe

MEMory:DELeTe

The DELeTe command purges the specified definition from instrument memory and makes it available for reuse.

13.5.1 :ALL

MEMory:DELeTe:ALL

The ALL command removes all memory names, key definitions, and data, and it returns memory to “available.”

13.5.2 [:NAME] <name>

MEMory:DELeTe:NAME

The NAME command is an optional node and deletes the definition associated with <name>.

13.6 :EXCHange

MEMory:EXCHange

The EXCHange subsystem allows for swapping the data contents between two memory items.

13.6.1 [:NAME] <name>,<name>

MEMory:EXCHange:NAME

This command is a default node which swaps the data contents between the memory items associated with the <name> parameters.

When the memory items between which the data is swapped are of different type, or have an incompatible structure, an execution error -294 (Incompatible type) is generated.

This command is an event with no associated query form and *RST condition.

13.6.2 :TABLE <name>

MEMory:EXCHange:TABLE

This command swaps the data contents between the SELEcted TABLE and the memory table that is associated with the <name> parameter.

When the memory item associated with the specified <name> parameter is not a TABLE, or the tables between which the data is swapped have an incompatible structure, an execution error -294 (Incompatible type) is generated.

This command is an event with no associated query form and *RST condition.

13.7 :FREE

MEMory:FREE

Returns information on the user memory space as follows:

<bytes available>,<bytes in use>

where the returned data for both shall be in <NR1> format.

The qualified FREE query for TRACe only memory usage is specified in the TRACe subsystem.

13.7.1 [:ALL]?

MEMory:FREE:ALL?

If the FREE query is qualified by the optional ALL node then the query shall return memory sizes related to all items in instrument memory.

13.7.2 :ASCIi?

MEMory:FREE:ASCIi?

If the FREE query is qualified by the ASCIi node then the memory usage and availability shall reflect the values that correspond to values for ASCIi data. In an instrument that has dedicated memory for ASCIi data types, these values reported shall be different from the total given by a FREE query of ALL types.

13.7.3 :BINary?

MEMory:FREE:BINary?

If the FREE query is qualified by the BINary node then the memory usage and availability shall reflect the values that correspond to values for BINary data. In an instrument that has dedicated memory for BINary data types, these values reported shall be different from the total given by a FREE query of ALL types.

13.7.4 :MACRo?

MEMory:FREE:MACRo?

If the FREE query is qualified by the MACRo node then the memory usage and availability shall reflect the values that correspond to values for MACRo data. In an instrument that has dedicated memory for MACRo data types, these values reported shall be different from the total given by a FREE query of ALL types.

13.7.5 :STATe?

MEMory:FREE:STATe?

If the FREE query is qualified by the STATe node then the memory usage and availability shall reflect the values that correspond to values for STATe data. In an instrument that has dedicated memory for STATe data types, these values reported shall be different from the total given by a FREE query of ALL types.

13.7.6 :TABLE?

MEMory:FREE:TABLE?

If the FREE query is qualified by the TABLE node then the memory usage and availability shall reflect the values that correspond to values for TABLE data. In an instrument that has dedicated memory for TABLE types, the values reported shall be different from the total given by a FREE query of ALL types.

13.8 :NSTates?

MEMory:NSTates?

The Number of States query requests that the instrument return the number of *SAV/*RCL states available in the instrument. The instrument shall return a <NR1> value which is one greater than the maximum that can be sent as a parameter to the *SAV and *RCL commands.

13.9 :STATe

MEMory:STATe

Collects commands relating to memory STATe data.

13.9.1 :CATalog?

MEMory:STATe:CATalog?

This query requests a list of defined names in the MEMory:STATe subsystem. The instrument returns a list of defined <name>'s in a comma separated list. Each <name> is returned in a quoted string. If no names are defined, a single null string is returned. There is no command form.

13.9.2 :DEFine <name> , <register_number>

MEMory:STATe:DEFine

:DEfine associates a <name> with a *SAV/*RCL register number where <name> is in character data format. <register_number> is in <NRf> format.

The query form of this command, MEMory:STATe:DEFine? <name> returns the associated <register_number> in <NR1> format.

*RST has no effect on this command.

13.10 :TYPE? <name>

MEMory:TYPE?

The TYPE query requests information about a defined name. The instrument must return a string indicating the subsystem where the name was defined. If the name is not defined, a null string is returned. For example, if VAT3 was previously defined by:

ROUTe:MODule:DEFine VAT3,(@20000(0:9))

then the query:

MEMory:TYPE? VAT3

requests a return of

“ROUT:MOD:”

13.11 :TABLE

MEMory:TABLE

The MEMory:TABLE command allows the user to DEFine a TABLE and to write data to and read data from it.

13.11.1 :BNUMber <numeric_value> {,<numeric_value>}

:MEMory:TABLE:BNUMber

BNUMber is the gas bottle number. This command sets the gas bottle number, which is a positive integer.

NOTE: For Emissions benches, BNUMber<n> are bottle ID number columns in the GasBottle, ZeroGas, SpanGas, Verify, and TopGas tables. In the ZeroGas, SpanGas, Verify, and TopGas tables, there are columns BNUMber<n>, where n corresponds to the analyzer range number. In GasBottle, there is a single BNUMber column.

13.11.1.1 :POINTs?

:MEMory:TABLE:BNUMber:POINTs?

This command returns the number of points in column BNUMber, for the currently selected table.

13.11.2 :CCURve <numeric_value> {,<numeric_value>}

:MEMory:TABLE:CCURve

CCURve sets the value of concentration in ppm using the current curve in linearization tables. This command sets the values of CCURve.

13.11.2.1 :POINTs?

:MEMory:TABLE:CCURve:POINTs?

This command returns the number of points in column CCURve, for the currently selected table.

13.11.3 :CONCetration <numeric_value> {,<numeric_value>}

:MEMory:TABLE:CONCetration

CONCetration is the gas concentration in ppm. This command sets the values of CONCetration.

13.11.3.1 :POINTs?

:MEMory:TABLE:CONCetration:POINTs?

This command returns the number of points in column CONCetration, for the currently selected table.

13.11.4 :CONDition

MEMory:TABLE:CONDition

13.11.4.1 [:MAGNitude]<Boolean>{,<Boolean>}

MEMory:TABLE:CONDition:MAGNitude

Specifies the CONDition[:MAGNitude] points of the TABLE. If there are more <Boolean>'s than in the associated DEFine statement, execution error -223 (Too much data) will be generated. This item list is included in the table DEFine command as CONDition[:MAGNitude]

At *RST, the contents of this list is device dependent.

13.11.4.1.1 :POINTs?

MEMory:TABLE:CONDition:MAGNitude:POINTs?

Returns the number of points in the CONDition[:MAGNitude] list of the SELEcted TABLE. If no CONDition[:MAGNitude] value have been sent, POINTs? returns a 0. If no TABLE has been selected, POINTs? returns a NAN.

13.11.5 :CPOint <numeric_value>{,<numeric_value>}

:MEMory:TABLE:CPOint

CPOint is the linearization gas dilution percentage or "cut point". This command sets the values of CPOint.

13.11.5.1 :POINTs?

:MEMory:TABLE:CPOint:POINTs?

This command returns the number of points in column CPOint, for the currently selected table.

13.11.6 :CURRent

MEMory:TABLE:CURRent

13.11.6.1 [:MAGNitude]<numeric_value>{,<numeric_value>}

MEMory:TABLE:CURRent:MAGNitude

Specifies the CURRent[:MAGNitude] points of the TABLE. <numeric_value>'s may include current units as listed in the UNIT subsystem. The default unit is specified by UNIT:CURRent. If there are more <numeric_value>'s than in the associated DEFine statement, execution error -223 (Too much data) will be generated. This item list is included in the table DEFine command as CURRent[:MAGNitude].

At *RST, the contents of this list item is device dependent.

13.11.6.1.1 :POINTs?

MEMory:TABLE:CURRent:MAGNitude:POINTs?

Returns the number of points in the CURRent[:MAGNitude] list of the SELEcted TABLE. If no CURRent[:MAGNitude] values have been sent POINTs? returns 0. If no TABLE has been selected, POINTs? returns NAN.

13.11.6.2 :PHASe<numeric_value>{,<numeric_value>}

MEMory:TABLE:CURRent:PHASe

Specifies the CURRent:PHASe points of the TABLE. <numeric_value>'s may include angle units as listed in the UNIT subsystem. The default unit is specified by UNIT:ANGLE. If there are more <numeric_value>'s than in the associated DEFine statement, execution error

-223 (Too much data) will be generated. This item list is included in the table DEFine command as CURRent:PHASe.

At *RST, the contents of this list item is device dependent.

13.11.6.2.1 :POINTS?

MEMory:TABLE:CURRent:PHASe:POINTS?

Returns the number of points in the CURRent:PHASe list of the SELEcted TABLE. If no CURRent:PHASe values have been sent POINTs? returns 0. If no TABLE has been selected, POINTs? returns NAN.

13.11.7 :DFACTory <numeric_value> {,<numeric_value>}

:MEMory:TABLE:DFACTory

DFACTory is the maximum allowable drift from factory settings.

For emissions benches, this is the zero or span factory setting in the tables ZDRift and SDRift respectively. This command sets the values of DFACTory. Drift from factory setting is calculated as follows:

Zero Drift
$$ZD_f = \frac{(Z^* - Z_f)}{f_s} * 100$$

Span Drift
$$SD_f = \frac{(S^* - S_f)}{f_s} * 100$$

where Z^* and S^* are the currently read zero and span values in ppm
 Z_f and S_f are the factory settings stored in DFACTory in ppm
 f_s is the full scale CONCentration in ppm
 ZD_f is the zero drift in PCT
 SD_f is the span drift in PCT

Note: a ratio of 1.0 = 100 PCT

13.11.7.1 :POINTS?

:MEMory:TABLE:DFACTory:POINTS?

This command returns the number of points in column DFACTory, for the currently selected table.

13.11.8 :DLASt <numeric_value> {,<numeric_value>}

:MEMory:TABLE:DLASt

DLASt is the maximum allowable drift from the most recently saved zero and set point.

For emissions benches, this is the most recently saved zero or span setting in the tables ZDRift and SDRift respectively. This command sets the values of DLASt. Drift from most recently saved zero or span (SPoint) setting is calculated as follows:

Zero Drift
$$ZD_0 = \frac{(Z^* - Z_0)}{f_s} * 100$$

Span Drift

$$SD_0 = \frac{(S^* - S_0)}{f_s} * 100$$

where Z^* and S^* are the currently read zero and span values in ppm
 Z_0 and S_0 are the most recently saved settings stored in DLAST in ppm
 f_s is the full scale CONcEntration in ppm
 ZD_0 is the zero drift in PCT
 SD_0 is the span drift in PCT

Note: a ratio of 1.0 = 100 PCT

13.11.8.1 :POINTs?

:MEMory:TABLE:DLAST:POINTs?

This command returns the number of points, for the currently selected table.

13.11.9 :DLINearize <numeric_value> {,<numeric_value>}

:MEMory:TABLE:DLINearize

DLINearize is the maximum allowable drift from the most recent linearization.

For emissions benches, this is the zero or span setting saved at the time of the most recent linearization in the tables ZDRift and SDRift respectively. This command sets the values of DLINearize. Drift from most recent linearization is calculated as follows:

Zero Drift

$$ZD_l = \frac{(Z^* - Z_l)}{f_s} * 100$$

Span Drift

$$SD_l = \frac{(S^* - S_l)}{f_s} * 100$$

where Z^* and S^* are the currently read zero and span values in ppm
 Z_l and S_l are the settings from the most recent linearization stored in DLINearize in ppm
 f_s is the full scale CONcEntration in ppm
 ZD_l is the zero drift in PCT
 SD_l is the span drift in PCT

Note: a ratio of 1.0 = 100 PCT

13.11.9.1 :POINTs?

:MEMory:TABLE:DLINearize:POINTs?

This command returns the number of points in column DLINearize, for the currently selected table.

13.11.10 :EXPeCted <numeric_value> {,<numeric_value>}

:MEMory:TABLE:EXPeCted

EXPeCted is the expected value.

For emissions benches, EXPeCted is the expected gas concentration in ppm. This command sets the values of EXPeCted.

13.11.10.1 :POINTs?

MEMory:TABLE:EXPEcted:POINTs?

This command returns the number of points in column EXPEcted , for the currently selected table.

13.11.11 :DEfINE<structure_string>[,<numeric_value>]

MEMory:TABLE:DEfINE

The DEfINE subsystem is used to allocate space and define the structure of TABLE in the instrument memory. The <name> of the TABLE is specified by SElect.

The string parameter <structure_string> specifies the structure of the TABLE being defined, it contains a comma separated list of the elements. Any item lists under MEMory:TABLE, such as FREQuency or VOLTage:PHASe, may be specified as elements in the table structure. The item lists maybe specified in long or short form, optional nodes if sent shall be accepted without error. The item lists available and the legal combinations of these item lists are device dependent. All lists are empty (0 length) when DEfined.

The second parameter, is the size of the TABLE. This is the maximum number of POINTs in each of the lists. Computations based on device dependent internal data formats will be made to determine the number of bytes. For example, if the TABLE will have two lists and the internal storage requires 8 bytes for a number, then an entry of 10 will allocate 160 bytes. If there is insufficient room, execution error -225 (Out of memory) will be generated. The size of the table parameter is optional. The algorithm to allocate memory is device dependent when this parameter is omitted.

The DEfINE query returns both the <structure_string> and a numeric value for the size (in points) of memory allocated for the table. The elements of the structure string are returned in the short form, with any optional nodes omitted.

13.11.12 :FORCe

MEMory:TABLE:FORCe

Specifies the FORCe points of the TABLE.

13.11.12.1 [:MAGNitude] <numeric_value>{,<numeric_value>}

MEMory:TABLE:FORCe:MAGNitude

Specifies the FORCe[:MAGNitude] points of the TABLE. The default units of this command are newtons.

At *RST, the contents of this list item is device dependent.

13.11.12.1.1 :POINTs?

MEMory:TABLE:FORCe:MAGNitude:POINTs?

Returns the number of points in the FORCe[:MAGNitude] list of the selected TABLE. If no FORCe[:MAGNitude] values have been sent POINTs? returns 0. If no TABLE has been selected, POINTs? returns NAN.

13.11.13 :FREQuency <numeric_value>{,<numeric_value>}

MEMory:TABLE:FREQuency

Specifies the FREQuency points of the TABLE. If there are more <numeric_value>'s than in the associated DEFine statement, execution error -223 (Too much data) will be generated. This item list is included in the table DEFine command as FREQuency.

At *RST, the contents of this list item is device dependent.

13.11.13.1 :POINTs?

MEMory:TABLE:FREQuency:POINTs?

Returns the number of points in the FREQuency list of the SELEcted TABLE. If no FREQuency values have been sent POINTs? returns 0. If no TABLE has been selected, POINTs? returns NAN.

13.11.14 :LABel <string> {,<string>}

:MEMory:TABLE:LABel

LABel is the label name in a table in character string format. This command sets the values of LABel.

For emissions benches, LABel is the label name in the GasBottle table.

13.11.14.1 :POINTs?

:MEMory:TABLE:LABel:POINTs?

This command returns the number of points in column LABel, for the currently selected table.

13.11.15 :LLIMit <numeric_value> {,<numeric_value>}

:MEMory:TABLE:LLIMit

LLIMit is the lower limit value. This command sets the values of LLIMit. If there are more <numeric_value>'s than in the associated DEFine statement, execution error -223 (Too much data) will be generated. This item list is included in the table DEFine command as LLIMit.

At *RST, the contents of this list item is device dependent.

13.11.15.1 :POINTs?

:MEMory:TABLE:LLIMit:POINTs?

This command returns the number of points in column LLIMit, for the currently selected table. If no LLIMit values have been sent POINTs? returns 0. If no TABLE has been selected, POINTs? returns NAN.

13.11.16 :LOG <string> {,<string>}

:MEMory:TABLE:LOG

LOG is a series of string data elements.

For emissions benches, LOG is a collection of audit-related strings. This command sets the values of LOG.

13.11.16.1 :POINTS?

:MEMory:TABLE:LOG:POINTS?

This command returns the number of points in column LOG, for the currently selected table.

13.11.17 :LOSS

MEMory:TABLE:LOSS

13.11.17.1 [:MAGNitude]<numeric_value>{,<numeric_value>}

MEMory:TABLE:LOSS:MAGNitude

Specifies the LOSS[:MAGNitude] points of the TABLE. The default unit is DB. If there are more <numeric_value>'s than in the associated DEFine statement, execution error -223 (Too much data) will be generated. This item list is included in the table DEFine command as LOSS[:MAGNitude].

At *RST, the contents of this list item is device dependent.

13.11.17.1.1 :POINTS?

MEMory:TABLE:LOSS:MAGNitude:POINTS?

Returns the number of points in the LOSS[:MAGNitude] list of the SELEcted TABLE. If no LOSS[:MAGNitude] values have been sent POINTs? returns 0. If no TABLE has been selected, POINTs? returns NAN.

13.11.17.2 :PHASe<numeric_value>{,<numeric_value>}

MEMory:TABLE:LOSS:PHASe

Specifies the LOSS:PHASe points of the TABLE. <numeric_value>'s may include angle units as listed in the UNIT subsystem. The default unit is specified by UNIT:ANGLE. If there are more <numeric_value>'s than in the associated DEFine statement, execution error -223 (Too much data) will be generated. This item list is included in the table DEFine command as LOSS:PHASe.

At *RST, the contents of this list item is device dependent.

13.11.17.2.1 :POINTS?

MEMory:TABLE:LOSS:PHASe:POINTS?

Returns the number of points in the LOSS:PHASe list of the SELEcted TABLE. If no LOSS:PHASe values have been sent POINTs? returns 0. If no TABLE has been selected, POINTs? returns NAN.

13.11.18 :NCURve <numeric_value> {,<numeric_value>}

:MEMory:TABLE:NCURve

NCURve sets the value of concentration in ppm using the newly calculated curve (New CURve) in linearization tables. This command sets the values of NCURve.

13.11.18.1 :POINTS?

:MEMory:TABLE:NCURve:POINTS?

This command returns the number of points in column NCURve, for the currently selected table.

13.11.19 :Power

MEMory:TABLE:POWer

13.11.19.1 [:MAGNitude]<numeric_value>{,<numeric_value>}

MEMory:TABLE:POWer:MAGNitude

Specifies the POWER[:MAGNitude] points of the TABLE. <numeric_value>'s may include POWER units as listed in the UNIT subsystem. The default unit is specified by UNIT:POWER. If there are more <numeric_value>'s than in the associated DEFINE statement, execution error -223 (Too much data) will be generated. This item list is included in the table DEFINE command as POWER[:MAGNitude].

At *RST, the contents of this list item is device dependent.

13.11.19.1.1 :POINTS?

MEMory:TABLE:POWer:MAGNitude:POINTS?

Returns the number of points in the POWER[:MAGNitude] list of the SElected TABLE. If no POWER[:MAGNitude] values have been sent POINTS? returns 0. If no TABLE has been selected, POINTS? returns NAN.

13.11.20 :RAW <numeric_value> {,<numeric_value>}

:MEMory:TABLE:RAW

RAW is a collection of points in instrument internal units.

For emissions benches, it is the raw gas concentration in bench internal units from a linearization. This command sets the values of RAW.

13.11.20.1 :POINTS?

:MEMory:TABLE:RAW:POINTS?

This command returns the number of points in column RAW, for the currently selected table.

13.11.21 :RESistance

MEMory:TABLE:RESistance

13.11.21.1 [:MAGNitude]<numeric_value>{,<numeric_value>}

MEMory:TABLE:RESistance:MAGNitude

Specifies the RESistance[:MAGNitude] points of the TABLE. If there are more <numeric_value>'s than in the associated DEFINE statement, execution error -223 (Too much data) will be generated. This item list is included in the table DEFINE command as RESistance[:MAGNitude].

At *RST, the contents of this list item is device dependent.

13.11.21.1.1 :POINTS?

MEMory:TABLE:RESistance:MAGNitude:POINTS?

Returns the number of points in the RESistance[:MAGNitude] list of the SElected TABLE. If no RESistance[:MAGNitude] values have been sent POINTS? returns 0. If no TABLE has been selected, POINTS? returns NAN.

13.11.22 :SElect <name>

MEMory:TABLE:SElect

Sets or queries the name of the TABLE selected. If the TABLE <name> already exists, then the existing TABLE shall be selected. If the TABLE <name> does not exist, then the new name shall be selected, but no TABLE shall be defined by this selection. The TABLE <name> is character data.

At *RST the <name> select shall be device dependent.

13.11.23 :SPEed

MEMory:TABLE:SPEed

Specifies the SPEed points of the TABLE.

13.11.23.1 [:MAGNitude] <numeric_value>{,<numeric_value>}

MEMory:TABLE:SPEed:MAGNitude

Specifies the SPEed[:MAGNitude] points of the TABLE. The default units of this command are meters per second.

At *RST, the contents of this list item is device dependent.

13.11.23.1.1 :POINts?

MEMory:TABLE:SPEed:MAGNitude:POINts?

Returns the number of points in the SPEed[:MAGNitude] list of the selected TABLE. If no SPEed[:MAGNitude] values have been sent POINts? returns 0. If no TABLE has been selected, POINts? returns NAN.

13.11.24 :TIME

MEMory:TABLE:TIME

Specifies the TIME points of the TABLE.

13.11.24.1 [:MAGNitude] <numeric_value>{,<numeric_value>}

MEMory:TABLE:TIME:MAGNitude

Specifies the TIME[:MAGNitude] points of the TABLE. The default units of this command are seconds.

At *RST, the contents of this list item is device dependent.

13.11.24.1.1 :POINts?

MEMory:TABLE:TIME:MAGNitude:POINts?

Returns the number of points in the TIME[:MAGNitude] list of the selected TABLE. If no TIME[:MAGNitude] values have been sent POINts? returns 0. If no TABLE has been selected, POINts? returns NAN.

13.11.25 :TOLerance <numeric_value> {,<numeric_value>}

:MEMory:TABLE:TOLerance

TOLerance is a series of tolerance values.

For emissions benches, it is the allowable deviation in gas concentration in percentage of full scale (%FS) between expected and current (or new curve) values stored from a linearization.

The bench will set an error bit in the status register if the difference between expected and actual values exceed the tolerance. This command sets the values of TOLerance.

13.11.25.1 :POINTs?

:MEMory:TABLE:TOLerance:POINTs?

This command returns the number of points in column TOLerance, for the currently selected table.

13.11.26 :ULIMit <numeric_value>{,<numeric_value>}

:MEMory:TABLE:ULIMit

ULIMit is the upper limit value. This command sets the values of ULIMit. If there are more <numeric_value>'s than in the associated DEFine statement, execution error -223 (Too much data) will be generated. This item list is included in the table DEFine command as ULIMit.

At *RST, the contents of this list item is device dependent.

13.11.26.1 :POINTs?

:MEMory:TABLE:ULIMit:POINTs?

This command returns the number of points in column ULIMit, for the currently selected table. If no ULIMit values have been sent POINTs? returns 0. If no TABLE has been selected, POINTs? returns NAN.

13.11.27 :VOLTage

MEMory:TABLE:VOLTage

13.11.27.1 [:MAGNitude]<numeric_value>{,<numeric_value>}

MEMory:TABLE:VOLTage:MAGNitude

Specifies the VOLTage[:MAGNitude] points of the TABLE. <numeric_value>'s may include voltage units as listed in the UNIT subsystem. The default unit is specified by UNIT:VOLTage. If there are more <numeric_value>'s than in the associated DEFine statement, execution error -223 (Too much data) will be generated. This item list is included in the table DEFine command as VOLTage[:MAGNitude].

At *RST, the contents of this list item is device dependent.

13.11.27.1.1 :POINTs?

MEMory:TABLE:VOLTage:MAGNitude:POINTs?

Returns the number of points in the VOLTage[:MAGNitude] list of the SElected TABLE. If no VOLTage[:MAGNitude] values have been sent POINTs? returns 0. If no TABLE has been selected, POINTs? returns NAN.

13.11.27.2 :PHASe <numeric_value>{,<numeric_value>}

MEMory:TABLE:VOLTage:PHASe

Specifies the VOLTage:PHASe points of the TABLE. <numeric_value>'s may include angle units as listed in the UNIT subsystem. The default unit is specified by UNIT:ANGLE. If there are more <numeric_value>'s than in the associated DEFine statement, execution error -223 (Too much data) will be generated. This item list is included in the table DEFine command as VOLTage:PHASe.

1999 SCPI Command Reference

At *RST, the contents of this list item is device dependent.

13.11.27.2.1 :POINTs?

MEMory:TABLE:VOLTage:PHASe:POINTs?

Returns the number of points in the VOLTage:PHASe list of the SELEcted TABLE. If no VOLTage:PHASe values have been sent POINTs? returns 0. If no TABLE has been selected, POINTs? returns NAN.

13.11.28 :WFACTor <numeric_value> {,<numeric_value>}

:MEMory:TABLE:WFACTor

Values in column WFACTor are weighting factors.

For emissions benches, a value in the WFACTor column is assigned to each cut point in a linearization. If weighting is not desired, all values are set to the same non-zero number. WFACTor is a column in linearization tables. This command sets the values of WFACTor.

13.11.28.1 :POINTs?

:MEMory:TABLE:WFACTor:POINTs?

This command returns the number of points in column WFACTor, for the currently selected table.

14 MMEMory Subsystem

The Mass MEMory subsystem provides mass storage capabilities for instruments. All mass memory device commands are contained in the MMEMory subsystem. The mass storage may be either internal or external to the instrument. If the mass memory device is external, then the instrument must have controller capability on the mass memory device bus.

The CLOSe, FEED, NAME, and OPEN commands permit the streaming of data from anywhere in the data flow into a file; this is particularly useful for saving HCOPY output.

Mass storage media may be formatted in any one of a number of standard formats.

Selecting Mass Memory Devices

When an instrument has multiple attached mass storage devices, the desired mass storage unit is selected with the mass storage unit specifier <msus>. The syntax of the <msus> string is instrument specific. Each instrument's documentation shall describe the syntax applicable to that instrument. For example:

```
"C:"
":700,2"
"fileserv"
```

Note that some file systems do not make use of the <msus> concept.

File Names

The <file_name> parameter described in the commands in this subsystem is a string. The contents of the string are dependent on the needs of the format of the mass storage media. In particular, the file name may contain characters for specifying subdirectories (that is, \ for DOS, / for HFS) and the separator for extensions in DOS (that is, period).

The following commands are provided for mass memory device operations.

KEYWORD	PARAMETER FORM	NOTES
MMEMory		
:CATalog?	[<msus>]	[query only]
:CDIRectory	[<directory_name>]	
:CLOSe		[event; no query] 1993
:COPY	<file_name>,<file_name> <file_name>,<msus>,<file_name>,<msus>	
:DATA	<file_name>,<data>	1995
:DELete	<file_name>[,<msus>]	[no query]
:FEED	<data_handle>	1993
:INITialize	[<msus>[(LIF DOS HFS) [,<numeric_value>]]]	[no query]
:LOAD		
:DINTerchange	<label>,<file_name>[,<msus>]	[no query]
:TRACe	<label>,<file_name>[,<msus>]	[no query] 1992
:MACRO	<label>,<file_name>[,<msus>]	[no query]
:STATe	<numeric_value>,<file_name>[,<msus>]	[no query]

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES	
:TABLE	<label>,<file_name>[,<msus>]	[no query]	1994
:TRACe	<label>,<file_name>[,<msus>]	[no query]	
:MSIS	[<msus>]		
:MOVE	(<src_file>,<dest_file>)	[no query]	1991
	l(<src_file>,<src_msus>,<dest_file>,<dest_msus>)		1991
:NAME	<file_name>[,<msus>]		1993
:OPEN		[event; no query]	1993
:PACK	<msus>	[no query]	1991
:STORe			
:DINTerchange	<label>,<file_name>[,<msus>]	[no query]	
:TRACe	<label>,<file_name>[,<msus>]	[no query]	1992
:MACRo	<label>,<file_name>[,<msus>]	[no query]	
:STATe	<numeric_value>,<file_name>[,<msus>]	[no query]	
:TABLE	<label>,<file_name>[,<msus>]	[no query]	1994
:TRACe	<label>,<file_name>[,<msus>]	[no query]	

14.1 :CATalog? [<msus>]

MMEMory:CATalog?

The CATalog command is query-only and returns information on the current contents and state of the mass storage media. Upon a CATalog? query, the instrument shall read the specified mass memory device and returns its directory information in the following format:

<numeric_value>,<numeric_value>{,<file_entry>}

The instrument shall return two numeric parameters and as many strings as there are files in the directory list. The first parameter shall indicate the total amount of storage currently used in bytes. The second parameter shall indicate the total amount of storage available, also in bytes. The <file_entry> is a string. Each <file_entry> shall indicate the name, type, and size of one file in the directory list:

<file_name>,<file_type>,<file_size>

Note that this syntax places some restrictions on the <file_name> (for example, commas are not allowed.) The <file_name> is the exact name of a file as it appears in the directory list, including an extension if present. The <file_size> provides the size of the file in bytes. The <file_type> is indicated by one of the following:

- ASCii — An ASCii text file
- BINary — A binary file
- STATe — Instrument (setting) state
- TRACe — Trace (display) data
- MACRo — An instrument macro

For media which cannot determine the file type from the directory information (i.e. DOS, HFS) this field shall be left blank.

14.2 **:CDIRectory [<directory_name>]**

MMEMory:CDIRectory

Changes the default directory for a mass memory file system. The <directory_name> parameter is a string. The contents of the <directory_name> parameter are dependent on the file system being accessed. If no parameter is specified, the directory is set to the *RST value.

At *RST, this value is set to the root for DOS file systems and to a device-dependent value for other file systems such as HFS.

14.3 **:CLOSe**

MMEMory:CLOSe

The CLOSe event causes the selected file specified in NAME to be closed. An attempt to CLOSe a file that was not open shall cause an error -256, (File name not found) to be generated.

14.4 **:COPY <file_source>,<file_destination>**

MMEMory:COPY

The COPY command copies an existing file to a new file.

Two forms of parameters are allowed. The first form has two parameters. In this form, the first parameter specifies the source file, and the second parameter specifies the destination file.

The second form has four parameters. In this form, the first and third parameter specify the file_names. The second and fourth parameters specify the <msus>. The first pair of parameters specifies the source file. The second pair specifies the destination file. An error is generated if the source file doesn't exist. An error may be generated if the destination file already exists.

This command defines an event and thus has no *RST state or query form.

14.5 **:DATA <file_name>,<data>**

MMEMory:DATA

The command form is MMEMory:DATA <file_name>,<data>. The command loads <data> into the file <file_name>. <data> is in 488.2 block format. <file_name> is string data.

The query form is MMEMory:DATA? <file_name> with the response being the associated <data> in block format.

MMEMory:DATA? response is not affected by the FORMat subsystem.

14.6 **:DELeTe <file_name>[,<msus>]**

MMEMory:DELeTe

The DELeTe command removes a file from the specified mass storage device. The <file_name> parameter specifies the file_name to be removed.

This command defines an event, and it has no *RST condition or query form.

14.7 :FEED <data_handle>

MMEMory:FEED

Sets or queries the <data_handle> to be used to feed data into the file specified by NAME. New data arriving from <data_handle> always overwrites the contents of the file specified by NAME. If the <data_handle> generates new data and the file is not OPEN an error -256, (File name not found) shall be generated.

At *RST, the value of FEED is device dependent.

14.8 :INITialize [<msus>[(LIFIDOSIHFS)],<numeric_value>]]

MMEMory:INITialize

The INITialize command is used to initialize the specified mass storage media. If the <msus> is not specified, the default is used. The default is the <msus> selected at *RST, not the currently selected <msus>. The second parameter specifies the type of media to be formatted. The <numeric_value> parameter is used to specify the format, interleave, sector sizes, and other media-dependent information. Execution of the command attempts to initialize the media. Previous data on the media, if any, is destroyed.

This command is an event and does not have a query form or a *RST condition.

14.9 :LOAD and :STORE

MMEMory:LOAD and :STORE

The LOAD and STORE subsystems transfer information between the mass memory device and the instrument's memory. The LOAD commands transfer from mass memory device to internal memory, and STORE transfers from internal memory to the mass memory device.

When information is loaded, the instrument uses the file type information to determine whether the data is in internal (BINary) or transportable (ASCii) form.

The <file_name> parameter is a quoted string and the <label> refers to an internal identifier for the instrument trace or macro.

The LOAD and STORE commands do not have query forms.

14.9.1 :DINTerchange <label>,<file_name>[,<msus>]

MMEMory:LOAD:DINTerchange

Specifies that the device transfer DINTerchange formatted data.

14.9.1.1 :TRACe <label>,<file_name>[,<msus>]

MMEMory:LOAD:DINTerchange:TRACe

The specified trace or array is loaded from or stored to the mass memory device in DINTerchange format.

14.9.2 :MACRo <label>,<file_name>[,<msus>]

MMEMory:LOAD:MACRo

The specified macro is loaded from or stored to the mass memory device.

14.9.3 :STATe <numeric_value>,<file_name>[,<msus>]

MMEMory:LOAD:STATe

The specified instrument state is stored or loaded. The <numeric_value> parameter refers to the state number, which corresponds to the numbers found in *SAV and *RCL.

14.9.4 :TABLE <label>,<file_name>[,<msus>]

MMEMory:LOAD:TABLE

The MEMory:TABLE specified by <label> is loaded from or stored to the mass memory device.

14.9.5 :TRACe <label>,<file_name>[,<msus>]

MMEMory:LOAD:TRACe

The specified trace or array is loaded from or stored to the mass memory device.

14.10 :MSIS [<msus>]

MMEMory:MSIS

The “Mass Storage IS” command selects a default mass storage device which is used by all MMEMory commands except INITialize. If the parameter is omitted, the device-dependent setting for default mass storage device is selected.

At *RST, the value of this parameter is device-dependent.

14.11 :MOVE (<src_file>,<dest_file>) I(<src_file>,<src_msus>,<dest_file>,<dest_msus>)

MMEMory:MOVE

The MOVE command moves an existing file to another file name.

Two forms of the parameter are allowed. The first form has two parameters. In this form the first parameter, <src_file>, specifies the file to be renamed, and the second parameter, <dest_file>, specifies the new name of the file.

The move operation is performed on the default mass storage device.

The second form has four parameters. The first two parameters, <src_file> and <src_msus>, specify the file name and storage device of the source file and the second two parameters, <dest_file> and <dest_msus>, specify the new file name and new storage device for the source file.

Some instruments may not support the moving of a file across mass storage devices and thus shall generate error -250, “Mass Storage Error”, when the msus parameters are not the same.

If the <src_file> does not exist, the instrument shall generate error -256, “File name not found”. If the <dest_file> already exists, the instrument shall generate error -257, “File name error”.

This command is an event-only command and thus has no *RST value, and no query form.

14.12 :NAME <file_name>[,<msus>]

MMEMory:NAME

Sets or queries the name of the file specification used by the CLOSe and OPEN commands.

At *RST, the value of this parameter is device dependent.

1999 SCPI Command Reference

14.13 :OPEN

MMEMory:OPEN

The OPEN event causes the selected file specified in NAME to be opened. An attempt to OPEN a file that is already opened shall cause an error -257, (File name error) to be generated. If the selected file is non-existent, it is created.

14.14 :PACK [<msus>]

MMEMory:PACK

This command causes the mass storage device to be packed, such that unused memory between files is recovered and packed together. If an instrument, which supports file packing (ie: implements the MMEM:PACK command), receives the MMEM:PACK command but the current disk format does not support file packing, the instrument shall generate an error -250 (Mass Storage Error). If the <msus> is not specified the default is used.

This command defines an event and thus has no *RST state or query form.

15 OUTPut Subsystem

The OUTPut subsystem controls the characteristics of the source's output port.

A source which is sourcing impedance (programmable load) is allowed to alias this subsystem as INPut, providing that the OUTPut keyword is also recognized and the INPut subsystem is not used for its intended purpose within the instrument.

KEYWORD	PARAMETER FORM	NOTES
OUTPut INPut		See text
:ATTenuation	<numeric_value>	
:COUPling	AC DC	
:FILTer		
:AUTO	<Boolean> ONCE	1992
:EXTernal		1992
[:STATe]	<Boolean>	1992
:HPASs		
:FREQuency	<numeric_value>	1991
[:STATe]	<Boolean>	
:TYPE	BESSel CHEByshev	1992
[:LPASs]		
:FREQuency	<numeric_value>	1991
[:STATe]	<Boolean>	
:TYPE	BESSel CHEByshev	1992
:IMPedance	<numeric_value>	
:LOW	FLOat GROund	1991
:POLarity	NORMal INVerted	1992
:POLarization	<numeric_value>	1994
:HORizontal		[no query] 1994
:VERTical		[no query] 1994
:POSition		1994
[:X]		1994
:ANGLE		1994
:DIRection	UP DOWN	1994
[:IMMEDIATE]	<numeric_value>	1994
:LIMit		1994
:HIGH	<numeric_value>	1994
:LOW	<numeric_value>	1994
[:STATe]	<Boolean>	1994
:OFFSet	<numeric_value>	1994
:VELocity	<numeric_value>	1994
[:DISTance]		1994
:DIRection	UP DOWN	1994
[:IMMEDIATE]	<numeric_value>	1994
:LIMit		1994
:HIGH	<numeric_value>	1994

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
:LOW	<numeric_value>	1994
[:STATe]	<Boolean>	1994
:OFFSet	<numeric_value>	1994
:VELocity	<numeric_value>	1994
:Y		1994
:ANGLE		1994
:DIRection	UPIDOWN	1994
[:IMMediate]	<numeric_value>	1994
:LIMit		1994
:HIGH	<numeric_value>	1994
:LOW	<numeric_value>	1994
[:STATe]	<Boolean>	1994
:OFFSet	<numeric_value>	1994
:VELocity	<numeric_value>	1994
[:DISTance]		1994
:DIRection	UPIDOWN	1994
[:IMMediate]	<numeric_value>	1994
:LIMit		1994
:HIGH	<numeric_value>	1994
:LOW	<numeric_value>	1994
[:STATe]	<Boolean>	1994
:OFFSet	<numeric_value>	1994
:VELocity	<numeric_value>	1994
:Z		1994
:ANGLE		1994
:DIRection	UPIDOWN	1994
[:IMMediate]	<numeric_value>	1994
:LIMit		1994
:HIGH	<numeric_value>	1994
:LOW	<numeric_value>	1994
[:STATe]	<Boolean>	1994
:OFFSet	<numeric_value>	1994
:VELocity	<numeric_value>	1994
[:DISTance]		1994
:DIRection	UPIDOWN	1994
[:IMMediate]	<numeric_value>	1994
:LIMit		1994
:HIGH	<numeric_value>	1994
:LOW	<numeric_value>	1994
[:STATe]	<Boolean>	1994
:OFFSet	<numeric_value>	1994
:VELocity	<numeric_value>	1994
:PROTection		
:DELay	<numeric_value>	

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
[[:STATe]	<Boolean>	
:TRIPped?		[query only]
:CLEar		[no query]
:ROSCillator		1991
[[:STATe]	<Boolean>	1991
:TTLTrg<n>		
l:ECLTrg<n>		1991
:IMMediate		1991
:LEVel	<Boolean>	1991
:POLarity	NORMallINVerted	1994
:PROTocol	SYNChronous SSYNchronous ASYNchronous	1991
:WIDTh	<numeric_value>	1991
[[:STATe]	<Boolean>	
:SOURce	<character data>	
[[:STATe]	<Boolean>	
:TYPE	<character data>	1992

15.1 :ATTenuation <numeric_value>

OUTPut:ATTenuation

Sets the output ATTenuation level. The default units are in the current relative amplitude units.

This should only be used where the setting is not reflected within the SOURce subsystem. For example, setting the output attenuation to 10 dB will cause the generated signal to decrease in amplitude by 10 dB. However, the power level setting will be unaffected by changing the output attenuation.

At *RST, this value is device-dependent.

15.2 :COUPling ACIDC

OUTPut:COUPling

Controls whether the signal is AC or DC coupled to the output port.

At *RST, this value is device-dependent.

15.3 :FILTer

OUTPut:FILTer

The FILTer function allows filters to be inserted in the path of the signal.

15.3.1 :AUTO <Boolean> | ONCE

OUTPut:FILTer:AUTO

Allows the system to determine the best filter characteristic and cutoff frequency. Explicitly selecting any output filter or output filter characteristic shall set AUTO OFF.

At *RST, this value is device dependent.

15.3.2 :EXTernal

OUTPut:FILTer:EXTernal

Controls an external, user provided, filter.

15.3.2.1 **[[:STATe] <Boolean>**

OUTPut:FILTer:EXTeRnal:STATe

Selects a user provided filter. When STATe is ON the filter is placed in the signal path.

At *RST, this value is OFF.

15.3.3 **:HPASs**

OUTPut:FILTer:HPASs

Controls the high pass filter.

15.3.3.1 **:FREQuency <numeric_value>**

OUTPut:FILTer:HPASs:FREQuency

Determines the cutoff frequency of the high pass filter.

At *RST, this value is device-dependent.

15.3.3.2 **[[:STATe] <Boolean>**

OUTPut:FILTer:HPASs:STATe

Turns the high pass filter ON and OFF.

At *RST, this value is device-dependent.

15.3.3.3 **:TYPE BESSEl | CHEByshev**

OUTPut:FILTer:HPASs:TYPE

Determines the characteristic of the high pass filter.

At *RST, this value is device dependent.

15.3.4 **[[:LPASs]**

OUTPut:FILTer:LPASs

Controls the low pass filter.

15.3.4.1 **:FREQuency <numeric_value>**

OUTPut:FILTer:LPASs:FREQuency

Determines the cutoff frequency of the low pass filter.

At *RST, this value is device-dependent.

15.3.4.2 **[[:STATe] <Boolean>**

OUTPut:FILTer:LPASs:STATe

Turns the low pass filter ON and OFF.

At *RST, this value is device-dependent.

15.3.4.3 **:TYPE BESSEl | CHEByshev**

OUTPut:FILTer:LPASs:TYPE

Determines the characteristic of the low pass filter.

At *RST, this value is device dependent.

- 15.4 :IMPedance <numeric_value>**
 OUTPut:IMPedance
 OUTPut source impedance for the signal specified in Ohms.
 At *RST, this value is device-dependent.
- 15.5 :LOW FLOatIGROund**
 OUTPut:LOW
 Connects the low signal terminal to ground or allows it to float.
 This parameter is set to a device-dependent value at *RST.
- 15.6 :POLarity NORMal I INVerted**
 OUTPut:POLarity
 Sets or queries the output polarity. NORMal causes the source block's signal to appear at the output with the same polarity as it was generated. INVerted causes the polarity of the source block's signal to be reversed when it appears at the output terminals.
 At *RST, this value is NORMal.
- 15.7 :POLarization <numeric_value>**
 OUTPut:POLarization
 Sets the polarization of the output. The default units are radian.
 At *RST, this value is device-dependent.
- 15.7.1 :HORizontal**
 OUTPut:POLarization:HORizontal
 Sets the polarization of the output to zero. This command is an event. Therefore it has no associated query or meaning at *RST.
- 15.7.2 :VERTical**
 OUTPut:POLarization:VERTical
 Sets the polarization of the output to $\pi/2$ radians. This command is an event. Therefore it has no associated query or meaning at *RST.
- 15.8 :POSition**
 OUTPut:POSition
 The characteristics of an incoming signal can be affected by the output's spatial position and orientation relative to the origin. The definition of the origin is application specific. This subsystem uses a righthand coordinate system and has only one origin.
 In this subsystem only the IMMEDIATE commands initiate movement.
 In this subsystem velocity is always positive.
- 15.8.1 [:X]**
 OUTPut:POSition:X
 Controls the settings concerning the X-axis.

15.8.1.1 :ANGLE

OUTPut:POSition:X:ANGLE

Specifies the angle of rotation around the X-axis.

15.8.1.1.1 :DIRection UPIDOWN

OUTPut:POSition:X:ANGLE:DIRection

Controls the direction of the movement when started with INITiate.

At *RST, this is set to UP

15.8.1.1.2 [:IMMediate] <numeric_value>

OUTPut:POSition:X:ANGLE:IMMediate

This command indicates that the output is moved to the specified position without waiting for further commands.

*RST has no effect on the IMMediate value except that movement shall stop.

15.8.1.1.3 :LIMit

OUTPut:POSition:X:ANGLE:LIMit

Sets the maximum bounds on the physical POSition.

15.8.1.1.3.1 :HIGH <numeric_value>

OUTPut:POSition:X:ANGLE:LIMit:HIGH

Sets the maximum for the POSition.

At *RST, this value is set to the clockwise physical limit.

15.8.1.1.3.2 :LOW <numeric_value>

OUTPut:POSition:X:ANGLE:LIMit:LOW

Sets the minimum for the POSition.

At *RST, this value is set to the counterclockwise physical limit.

15.8.1.1.3.3 :STATe <Boolean>

OUTPut:POSition:X:ANGLE:LIMit:STATe

Controls whether the LIMit is enabled.

At *RST, this value is set to ON.

15.8.1.1.4 :OFFSet <numeric_value>

OUTPut:POSition:X:ANGLE:OFFSet

This defines a single value that is subtracted from the physical POSition.

At *RST, this value is device-dependent.

15.8.1.1.5 :VELocity <numeric_value>

OUTPut:POSition:X:ANGLE:VELocity

Controls the velocity of the angular rotation around the x-axis.

At *RST, this value is device dependent.

15.8.1.2 **[[:DISTance]**

OUTPut:POSition:X:DISTanCe

Controls the settings in linear direction of the X-axis.

15.8.1.2.1 **:DIRection UPIDOWN**

OUTPut:POSition:X:DISTanCe:DIRection

Controls the direction of the movement when started with INITiate.

At *RST, this is set to UP

15.8.1.2.2 **[[:IMMediate] <numeric_value>**

OUTPut:POSition:X:DISTanCe:IMMediate

This command indicates that the output is moved to the specified position without waiting for further commands.

*RST has no effect on the IMMediate value except that movement shall stop.

15.8.1.2.3 **:LIMit**

OUTPut:POSition:X:DISTanCe:LIMit

Sets the maximum bounds on the physical POSition.

15.8.1.2.3.1 **:HIGH <numeric_value>**

OUTPut:POSition:X:DISTanCe:LIMit:HIGH

Sets the maximum for the POSition.

At *RST, this value is set to the upper physical limit.

15.8.1.2.3.2 **:LOW <numeric_value>**

OUTPut:POSition:X:DISTanCe:LIMit:LOW

Sets the minimum for the POSition.

At *RST, this value is set to the lower physical limit.

15.8.1.2.3.3 **:STATe <Boolean>**

OUTPut:POSition:X:DISTanCe:LIMit:STATe

Controls whether the LIMit is enabled.

At *RST, this value is set to ON.

15.8.1.2.4 **:OFFSet <numeric_value>**

OUTPut:POSition:X:DISTanCe:OFFSet

This defines a single value that is subtracted from the physical POSition.

At *RST, this value is device-dependent.

15.8.1.2.5 **:VELocity <numeric_value>**

OUTPut:POSition:X:DISTanCe:VELocity

Controls the velocity of the displacement on the x-axis.

At *RST, this value is device dependent.

15.8.2 :Y

OUTPut:POSition:Y

Controls the settings concerning the Y-axis.

15.8.2.1 :ANGLE

OUTPut:POSition:Y:ANGLE

Specifies the angle of rotation around the Y-axis.

15.8.2.1.1 :DIRection UPIDOWN

OUTPut:POSition:Y:ANGLE:DIRection

Controls the direction of the movement when started with INITiate.

At *RST, this is set to UP

15.8.2.1.2 [:IMMediate] <numeric_value>

OUTPut:POSition:Y:ANGLE:IMMediate

This command indicates that the output is moved to the specified position without waiting for further commands.

*RST has no effect on the IMMediate value except that movement shall stop.

15.8.2.1.3 :LIMit

OUTPut:POSition:Y:ANGLE:LIMit

Sets the maximum bounds on the physical POSition.

15.8.2.1.3.1 :HIGH <numeric_value>

OUTPut:POSition:Y:ANGLE:LIMit:HIGH

Sets the maximum for the POSition.

At *RST, this value is set to the clockwise physical limit.

15.8.2.1.3.2 :LOW <numeric_value>

OUTPut:POSition:Y:ANGLE:LIMit:LOW

Sets the minimum for the POSition.

At *RST, this value is set to the counterclockwise physical limit.

15.8.2.1.3.3 :STATe <Boolean>

OUTPut:POSition:Y:ANGLE:LIMit:STATe

Controls whether the LIMit is enabled.

At *RST, this value is set to ON.

15.8.2.1.4 :OFFSet <numeric_value>

OUTPut:POSition:Y:ANGLE:OFFSet

This defines a single value that is subtracted from the physical POSition.

At *RST, this value is device-dependent.

15.8.2.1.5 :VELocity <numeric_value>

OUTPut:POSition:Y:ANGLE:VELocity

Controls the velocity of the angular rotation around the y-axis.

At *RST, this value is device dependent.

15.8.2.2 **[[:DISTance]**

OUTPut:POStion:Y:DiSTance

Controls the settings in linear direction of the Y-axis.

15.8.2.2.1 **:DiRection UPIDOWN**

OUTPut:POStion:Y:DiSTance:DiRection

Controls the direction of the movement when started with INITiate.

At *RST, this is set to UP

15.8.2.2.2 **[[:IMMediate] <numeric_value>**

OUTPut:POStion:Y:DiSTance:IMMediate

This command indicates that the output is moved to the specified position without waiting for further commands.

*RST has no effect on the IMMediate value except that movement shall stop.

15.8.2.2.3 **:LiMiT**

OUTPut:POStion:Y:DiSTance:LiMiT

Sets the maximum bounds on the physical POStion.

15.8.2.2.3.1 **:HiGH <numeric_value>**

OUTPut:POStion:Y:DiSTance:LiMiT:HiGH

Sets the maximum for the POStion.

At *RST, this value is set to the upper physical limit.

15.8.2.2.3.2 **:LoW <numeric_value>**

OUTPut:POStion:Y:DiSTance:LiMiT:LoW

Sets the minimum for the POStion.

At *RST, this value is set to the lower physical limit.

15.8.2.2.3.3 **:STATe <Boolean>**

OUTPut:POStion:Y:DiSTance:LiMiT:STATe

Controls whether the LiMiT is enabled.

At *RST, this value is set to ON.

15.8.2.2.4 **:OFFSet <numeric_value>**

OUTPut:POStion:Y:DiSTance:OFFSet

This defines a single value that is subtracted from the physical POStion.

At *RST, this value is device-dependent.

15.8.2.2.5 **:VELOCITY <numeric_value>**

OUTPut:POStion:Y:DiSTance:VELOCITY

Controls the velocity of the displacement on the y-axis.

At *RST, this value is device dependent.

15.8.3 :Z

OUTPut:POSition:Z

Controls the settings concerning the Z-axis

15.8.3.1 :ANGLE

OUTPut:POSition:Z:ANGLE

Specifies the angle of rotation around the Z-axis.

15.8.3.1.1 :DIRection UPIDOWN

OUTPut:POSition:Z:ANGLE:DIRection

Controls the direction of the movement when started with INITiate.

At *RST, this is set to UP

15.8.3.1.2 [:IMMediate] <numeric_value>

OUTPut:POSition:Z:ANGLE:IMMediate

This command indicates that the output is moved to the specified position without waiting for further commands.

*RST has no effect on the IMMediate value except that movement shall stop.

15.8.3.1.3 :LIMit

OUTPut:POSition:Z:ANGLE:LIMit

Sets the maximum bounds on the physical POSition.

15.8.3.1.3.1 :HIGH <numeric_value>

OUTPut:POSition:Z:ANGLE:LIMit:HIGH

Sets the maximum for the POSition.

At *RST, this value is set to the clockwise physical limit.

15.8.3.1.3.2 :LOW <numeric_value>

OUTPut:POSition:Z:ANGLE:LIMit:LOW

Sets the minimum for the POSition.

At *RST, this value is set to the counterclockwise physical limit.

15.8.3.1.3.3 :STATe <Boolean>

OUTPut:POSition:Z:ANGLE:LIMit:STATe

Controls whether the LIMit is enabled.

At *RST, this value is set to ON.

15.8.3.1.4 :OFFSet <numeric_value>

OUTPut:POSition:Z:ANGLE:OFFSet

This defines a single value that is subtracted from the physical POSition.

At *RST, this value is device-dependent.

15.8.3.1.5 :VELocity <numeric_value>

OUTPut:POSition:Z:ANGLE:VELocity

Controls the velocity of the angular rotation around the z-axis.

At *RST, this value is device dependent.

15.8.3.2 **[[:DISTance]**

OUTPut:POStion:Z:DISTance

Controls the settings in linear direction of the Z-axis.

15.8.3.2.1 **:DIRection UPIDOWN**

OUTPut:POStion:Z:DISTance:DIRection

Controls the direction of the movement when started with INITiate.

At *RST, this is set to UP

15.8.3.2.2 **[[:IMMediate] <numeric_value>**

OUTPut:POStion:Z:DISTance:IMMediate

This command indicates that the output is moved to the specified position without waiting for further commands.

*RST has no effect on the IMMediate value except that movement shall stop.

15.8.3.2.3 **:LIMit**

OUTPut:POStion:Z:DISTance:LIMit

Sets the maximum bounds on the physical POStion.

15.8.3.2.3.1 **:HIGH <numeric_value>**

OUTPut:POStion:Z:DISTance:LIMit:HIGH

Sets the maximum for the POStion.

At *RST, this value is set to the upper physical limit.

15.8.3.2.3.2 **:LOW <numeric_value>**

OUTPut:POStion:Z:DISTance:LIMit:LOW

Sets the minimum for the POStion.

At *RST, this value is set to the lower physical limit.

15.8.3.2.3.3 **:STATe <Boolean>**

OUTPut:POStion:Z:DISTance:LIMit:STATe

Controls whether the LIMit is enabled.

At *RST, this value is set to ON.

15.8.3.2.4 **:OFFSet <numeric_value>**

OUTPut:POStion:Z:DISTance:OFFSet

This defines a single value that is subtracted from the physical POStion.

At *RST, this value is device-dependent.

15.8.3.2.5 **:VELocity <numeric_value>**

OUTPut:POStion:Z:DISTance:VELocity

Controls the velocity of the displacement on the z-axis.

At *RST, this value is device dependent.

1999 SCPI Command Reference

- 15.9 :PROTection**
OUTPut:PROTection
Controls the characteristics of the output protection or foldback circuits.
- 15.9.1 :DELaY <numeric_value>**
OUTPut:PROTection:DELaY
Controls the amount of time the protection trip condition must be true before the subsequent protection is taken (for example, tripping a foldback circuit or reporting an error). The units for the parameter are seconds.

At *RST, this value is device-dependent.
- 15.9.2 [:STATe] <Boolean>**
OUTPut:PROTection:STATe
Controls whether the output protection circuit is enabled.

At *RST, this value is device-dependent.
- 15.9.3 :TRIPped?**
OUTPut:PROTection:TRIPped?
This query returns a 1 if the protection circuit is tripped and a 0 if it is untripped. TRIPped only has a query form and thus has no associated *RST condition.
- 15.9.4 :CLEar**
OUTPut:PROTection:CLEar
Causes the protection circuit to be cleared.

This command is an event and has no associated *RST condition.
- 15.10 :ROSCillator**
OUTPut:ROSCillator
The ROSCillator subtree controls the output on the ROSCillator OUTPut port.
- 15.10.1 [:STATe] <Boolean>**
OUTPut:ROSCillator:STATe
The STATe command selects whether or not the device outputs a signal on its ROSCillator port. When ON is selected, the device outputs a signal; when OFF, it does not.

At *RST, STATe OFF shall be selected.
- 15.11 :TTLTrg<n>|:ECLTrg<n>**
OUTPut:TTLTrgl:ECLTrg
The TTLTrg and ECLTrg subtrees control the driving of the VXI backplane TTL and ECL trigger buses. A numeric suffix is required, indicating the particular line in the bus. Valid lines are TTLTrg0 through TTLTrg7, and ECLTrg0 through ECLTrg5.
- 15.11.1 :IMMediate**
OUTPut:TTLTrgl:ECLTrg:IMMediate
The IMMediate command causes a pulse to appear on the specified trigger line, regardless of the level.

15.11.2 :LEVel <Boolean>

OUTPut:TTLTrgl:ECLTrg:LEVel

The LEVel command sets the selected TTLTrg or ECLTrg line to a logical level 0 or 1. For TTL, level 0 is 5 volts and level 1 is 0 volts. For ECL level 0 is -2 volts and level 1 is 0 volts.

At *RST, LEVel is set to OFF.

15.11.3 :POLarity NORMAl|INVerted

OUTPut:TTLTrgl:ECLTrg:POLarity

This command sets and queries the polarity of the specified VXI TTL or ECL trigger line.

At *RST, the value of this parameter is NORMAl.

15.11.4 :PROTOcol SYNChronous|SSYNchronous|ASYNchronous

OUTPut:TTLTrgl:ECLTrg:PROTOcol

This command selects the trigger protocol for the VXI TTLTrg or ECLTrg trigger lines. The protocols are specified in the VXIbus System Specification Revision 1.4 under sections B.6.2.3 and B.6.2.4.

- SYNChronous selects the synchronous trigger protocol.
- SSYNchronous selects the semi-synchronous trigger protocol.
- ASYNchronous selects the asynchronous trigger protocol.

At *RST, the SYNChronous protocol is selected for all trigger lines.

15.11.5 :WIDTh <numeric_value>

OUTPut:TTLTrgl:ECLTrg:WIDTh

Selects width of the pulse being generated by the OUTPut:TTLTrg<n>:IMMEDIATE or OUTPut:ECLTrg<n>:IMMEDIATE command. The default units are seconds.

The *RST value is device dependent.

15.11.6 [:STATe] <Boolean>

OUTPut:TTLTrgl:ECLTrg:STATe

The STATe command selects whether or not the VXI module drives the VXI backplane trigger line. When ON is selected, the module shall drive the trigger line; when OFF, it shall not.

At *RST, STATe OFF shall be selected.

15.11.7 :SOURce <character data>

OUTPut:TTLTrgl:ECLTrg:SOURce

Controls which signal from the VXI module shall drive the VXI backplane trigger line. Source names are character data, such as INTernal or EXTernal.

At *RST, the value of this parameter is device dependent.

15.12 **[[:STATe] <Boolean>**

OUTPut:STATe

The STATe function controls whether the output terminals are open or closed. When the function is OFF, the terminals are at maximum isolation from the signal.

At *RST, this function shall be set to OFF.

15.13 **:TYPE <character data>**

OUTPut:TYPE

Selects the output characteristic to be used if more than one is available.

The following characteristics are currently defined for instruments:

- BALanced – balanced or differential output.
- UNBalanced – unbalanced with respect to ground.

At *RST the TYPE is device-dependent.

16 PROGram Subsystem

The purpose of the PROGram subsystem is to provide the administrative features needed to generate and control one or more user-programmed tasks resident in an instrument. Two distinct methods of accessing a particular program are provided in the PROGram subsystem. One method employs EXPLicit reference for each command, allowing access to all programs, but requires the program name to be always specified. The other method allows a single program to be SELEcted at a time, and the related commands access only the currently selected program. If the EXPLicit method is implemented, then the SELEcted method shall also be implemented.

A *RST received from a remote controller, via the device interface, shall cause all programs to be stopped.

A *RST generated by a program running in the device shall cause the same effect as a *RST received from a remote controller, with the exception that the program that generated the *RST shall not be forced to stop. It is necessary for a *RST to behave in this manner so that a program, when run on either a remote controller or the device, shall have the same effect.

KEYWORD	PARAMETER FORM	NOTES
PROGram		
:CATalog?		[query only]
[:SELEcted]		
:DEFine	<program>	
:DELete		[no query]
[:SELEcted]		
:ALL		
:EXECute	<program_command>	[no query]
:MALLocate	<nbytes> DEFAULT	
:NAME	<prognam>	
:NUMBer	<varname>{,<nvalues>}	
:STATe	(RUN PAUSE STOP CONTinue)	
:STRing	<varname>{,<svalues>}	
:WAIT		1991
:EXPLicit		
:DEFine	<prognam>,<program>	
:DEFine?	<prognam>	[query only]
:DELete	<prognam>	
:EXECute	<prognam>,<program_command>	[no query]
:MALLocate	<prognam>,<nbytes> DEFAULT	
:NUMBer	<prognam>,<varname>{,<nvalues>}	
:STATe	<prognam>,(RUN PAUSE STOP CONTinue)	
:STRing	<prognam>,<varname>{,<svalues>}	
:WAIT	<prognam>	1991

16.1 :CATalog?

PROGram:CATalog?

The CATalog query commands lists all the defined programs. Response is a list of comma-separated strings. Each string contains the name of a program. If no programs are currently defined, then response is a null string ("").

16.2 [:SElected]

PROGram:SElected

All the commands under this node access only the program that has been selected by the NAME command.

16.2.1 :DEFine <program>

PROGram:SElected:DEFine

The DEFine command is used to create and download programs. The DEFine query is used to upload programs. The program name used for the definition is the currently selected program name.

In the DEFine command, the selected program name shall represent a unique name. That is, to download and overwrite an existing program with the same program name, the existing program must be DELeted first. If such an attempt is made without deleting first, an "Illegal program name" error (-282) shall be generated.

The <program> shall be block data. The contents are device-dependent. When overflow occurs a program error (-28X) shall be generated.

In the DEFine query, the selected program shall be returned. The <program> shall be uploaded as definite length arbitrary block response data.

16.2.2 :DElete

PROGram:SElected:DElete

The DElete command deletes downloaded programs.

16.2.2.1 [:SElected]

PROGram:SElected:DElete:SElected

This command causes the selected program to be deleted.

16.2.2.2 :ALL

PROGram:SElected:DElete:ALL

ALL is used to specify that all programs in the device are to be deleted. If any of the programs are in the RUN state, a "Program currently running" error (-284) shall be generated and no programs shall be deleted.

16.2.3 :EXECute <program_command>

PROGram:SElected:EXECute

The EXECute command executes the <program_command> in the selected program environment. <program_command> is string data representing any legal program command. If the string data is not legal, then a "Program syntax error" (-285) shall be generated. The selected program shall be in either the PAUSed or STOPped state before the EXECute

command shall be allowed. If the program is in the RUN state, a “Program currently running” error (-284) shall be generated.

16.2.4 :MALLocate <nbytes>IDEFault

PROGram:SELEcted:MALLocate

The Memory ALLocate command is used to reserve memory space in the device for use by the selected program. If DEFault is specified, then the device shall calculate the amount of memory needed. <nbytes> is numerical data which specifies the required memory in bytes. If the required memory is greater than the space available, then a parameter error “Data out of range” (-222) shall be generated.

*RST has no effect on the value of MALLocate.

16.2.5 :NAME <progrname>

PROGram:SELEcted:NAME

The NAME command defines the name of the program to be selected. If the program name <progrname> already exists, then that existing program shall be selected. If the program name does not exist, then the new name shall be selected, but no program shall be defined by this selection. <progrname> is character data.

*RST causes the selected name to become PROG.

16.2.6 :NUMBer <varname>{,<nvalues>}

PROGram:SELEcted:NUMBer

The NUMBer command is used to set and query the contents of numeric program variables and arrays in the currently selected program. The currently selected program must be a DEFined program, otherwise an “Illegal program name” error (-282) shall be generated. The variable specified in <varname> shall be the name of an existing variable in the selected program, otherwise an “Illegal variable name” error (-283) shall be generated. <varname> can be either character data or string data. <nvalues> is a list of comma separated <numeric_values> which are used to set <varname>. If the specified variable cannot hold all of the specified <numeric_values> then a “Parameter not allowed” error (-108) shall be generated.

The query form NUMBer? <varname> returns the contents of the variable as a comma separated list

16.2.7 :STATe RUNIPAUSelSTOPICONTinue

PROGram:SELEcted:STATe

The STATe command is used to either set the state or query the state of a selected program. The following matrix defines the effect of setting the STATe to the desired value from each of the possible current states. In certain cases a parameter error “Settings conflict” (-221) shall be generated.

The states are described as follows:

RUNNING:	The program is currently executing
PAUSED:	The program has reached a break in execution but can be continued.
STOPPED:	Execution has been terminated.

1999 SCPI Command Reference

state requested	current state		
	RUNNING	PAUSED	STOPPED
RUN	error (-221)	RUNNING	RUNNING
CONT	error (-221)	RUNNING	error (-221)
PAUSE	PAUSED	PAUSED	STOPPED

16.2.8 **:STRing <varname>{,<svalues>}**

PROGram:SElected:STRing

The STRing command is used to set and query the contents of string program variables and arrays in the currently selected program. The currently selected program must be a DEFINed program, otherwise an “Illegal program name” error (-282) shall be generated. The variable specified in <varname> shall be the name of an existing variable in the selected program, otherwise an “Illegal variable name” error (-283) shall be generated. <varname> can be either character data or string data. <svalues> is a list of comma-separated strings which are used to set <varname>. If the specified variable cannot hold all of the specified string values then a “Parameter not allowed” error (-108) shall be generated. If a string value is too long then it will be truncated when stored in the programs variable.

The query form STRing? <varname> returns the contents of the variable as a comma separated list

16.2.9 **:WAIT**

PROGram:SElected:WAIT

The WAIT command and its query form shall cause no further commands or queries to be executed, until the selected program exits from the RUN state, that is either STOPped or PAUSed. For the query, a 1 is returned in NR1 format at this time, when the program is either STOPped or PAUSed.

16.3 **:EXPLicit**

PROGram:EXPLicit

All the commands under the EXPLicit node directly reference the desired program by name, thus allowing access to a program without having to change the selected program. The <progrname> parameter is required for these commands.

16.3.1 **:DEFine <progrname>,<program>**

PROGram:EXPLicit:DEFine

The DEFine command is used to create and download programs. The DEFine query is used to upload programs.

In the DEFine command, the specified program name shall represent a unique name. That is, to download and overwrite an existing program with the same program name, the existing program must be DELEted first. If such an attempt is made, without deleting first, an “Illegal program name” error (-282) shall be generated.

The <program> shall be block data. The contents are device-dependent. When overflow occurs a program error (-28X) shall be generated.

In the DEFine query, the specified program name shall be the name of an existing program; otherwise an “Illegal program name” error (-282) shall be generated. The <program> shall be uploaded as definite length arbitrary block response data.

16.3.2 :DELeTe <progrname>

PROGram:EXPLicit:DELeTe

The delete command deletes the specified downloaded program.

16.3.3 :EXECute <progrname>,<program_command>

PROGram:EXPLicit:EXECute

The EXECute command executes the <program_command> in the specified program environment. <program_command> is string data representing any legal program command. If the string data is not legal, then a “Program syntax error” (-285) shall be generated. The specified program shall be in either the PAUSed or STOPped state before the EXECute command shall be allowed. If the program is in the RUN state, a “Program currently running” error (-284) shall be generated.

16.3.4 :MALLocate <progrname>,<(nbytes>IDEFault)

PROGram:EXPLicit:MALLocate

The Memory ALLocate command is used to reserve memory space in the device for use by the specified program. The specified program shall be the name of an existing program, otherwise an “Illegal program name” error (-282) shall be generated. If DEFault is specified, then the device shall calculate the amount of memory needed. <nbytes> is numerical data which specifies the required memory in bytes. If the required memory is greater than the space available, then a parameter error “Data out of range” (-222) shall be generated.

*RST has no effect on the value of MALLocate.

16.3.5 :NUMBer <progrname>,<varname>{,<nvalues>}

PROGram:EXPLicit:NUMBer

The NUMBer command is used to set and query the contents of numeric program variables and arrays in the specified program. The specified program must be a DEFined program, otherwise an “Illegal program name” error (-282) shall be generated. The variable specified in <varname> shall be the name of an existing variable in the specified program, otherwise an “Illegal variable name” error (- 283) shall be generated. <varname> can be either character data or string data. <nvalues> is a list of comma-separated <numeric_values> which are used to set <varname>. If the specified variable cannot hold all of the specified <numeric_values>, then a “Parameter not allowed” error (-108) shall be generated.

The query form NUMBer? <varname> returns the contents of the variable as a comma separated list

16.3.6 :STATe <progrname>,<(RUNIPAUSeISTOPICONTinue)

PROGram:EXPLicit:STATe

The STATe command is used to either set the state or query the state of the specified program. The matrix below defines the effect of setting the STATe to the desired value from

1999 SCPI Command Reference

each of the possible current states. In certain cases a parameter error “Settings conflict” (-221) shall be generated.

desired state	current state		
	RUNNING	PAUSED	STOPPED
RUN	error (-221)	RUNNING	RUNNING
CONT	error (-221)	RUNNING	error (-221)
PAUSE	PAUSED	PAUSED	STOPPED
STOP	STOPPED	STOPPED	STOPPED

The states are described as follows:

RUNNING: The program is currently executing
PAUSED: The program has reached a break in execution but can be continued.
STOPPED: Execution has been terminated.

16.3.7 :STRing <progrname>,<varname>{,<svalues>}

PROGram:EXPLicit:STRing

The STRing command is used to set and query the contents of string program variables and arrays in the specified program. The specified program must be a DEFINed program, otherwise an “Illegal program name” error (-282) shall be generated. The variable specified in <varname> shall be the name of an existing variable in the specified program, otherwise an “Illegal variable name” error (-283) shall be generated. <varname> can be either character data or string data. <svalues> is a list of comma-separated strings which are used to set <varname>. If the specified variable cannot hold all of the specified string values, then a “Parameter not allowed” error (-108) shall be generated. If a string value is too long then it will be truncated when stored in the programs variable.

The query form STRing? <varname> returns the contents of the variable as a comma separated list

16.3.8 :WAIT <progrname>

PROGram:EXPLicit:WAIT

The WAIT command and its query form shall cause no further commands or queries to be executed, until the defined program exits from the RUN state, that is either STOPped or PAUSEd. For the query, a 1 is returned in NR1 format at this time, when the program is either STOPped or PAUSEd.

17 ROUTe Subsystem

In the SCPI instrument model, the block where user accessibility to the actual signals occurs is called “signal routing.” In some instruments this function may be trivial and thus commands for this block would not be required. In such cases the ports associated with the INPut or OUTput blocks are available directly to the user, typically on the instrument’s front panel. The ROUTe commands apply equally to instruments whose primary purpose is to provide signal routing and to instruments that provide some routing capability “in front of” the INPut and/or OUTput blocks.

The ROUTe node may be optional in a particular instrument. This capability is intended for instruments whose primary function is signal routing. The ROUTe node cannot be optional if either SENSE or SOURCE is optional, since only one node at a given level may be optional.

KEYWORD	PARAMETER FORM	NOTES
:ROUTe		
:CLOSE	<channel_list>	1992
:STATe?		[query only]
:MODule		
:CATalog?		[query only]
:DEFine	<module_name>,<module_address>	
:DEFine?	<module_name>	
:DELete		
:ALL		[event; no query]
[:NAME]	<module_name>	[event; no query]
:OPEN	<channel_list>	1992
:ALL		[event; no query] 1993
:PATH		
:CATalog?		[query only]
:DEFine	<path_name>,<channel_list>	
:DEFine?	<path_name>	
:DELete		
:ALL		[event; no query]
[:NAME]	<path_name>	[event; no query]
:SAMPLE		1999
:CATalog?		[query only] 1999
[:OPEN]	BAG DILute PRE POST MID ICEFFiciency NONE ZERO SPAN VERify MANifold	1999
:SCAN	<channel_list>	1992,4
:TERMinals	FRONT REAR BOTH NONE	1992

17.1 :CLOSE <channel_list>

ROUTe:CLOSE

The CLOSE command allows specific individual channels to be closed or queried. If all the specified channels cannot be closed, an execution error is reported.

1999 SCPI Command Reference

For example, suppose a module contains a coaxial multiplexor, and only one channel on a bank is able to be closed at a time. If a command tries to close two channels on the same bank, the instrument should report an execution error.

CLOSE may be an overlapped command. The *OPC mechanism may be used to determine when switches are actually closed or opened. The sequence in which channels are closed is not guaranteed.

The CLOSE? query allows the condition of individual switches to be queried. The instrument returns a 1 or 0 for each channel in the list, in the same order that the list is specified. A response of 1 means the channel is closed and a 0 means the channel is open.

At *RST, this value is device-dependent.

17.1.1 :STATe?

ROUTe:CLOSe:STATe?

The ROUTe:CLOSe:STATe? query, which has no parameters, returns an IEEE 488.2 definite length block which contains a <channel_list> of all the closed switches in the entire instrument.

17.2 :MODule

ROUTe:MODule

The MODule subsystem controls the assignment of <module_name>.

17.2.1 :CATalog?

ROUTe:MODule:CATalog?

The ROUTe:MODule:CATalog? query returns a list of all currently defined <module_names>.

Response is a list of comma-separated strings. Each string contains a <module_name>. If no <module_name> elements are currently defined, then response is a null string ("").

17.2.2 [:DEFine]<module_name>,<module_address>

ROUTe:MODule:DEFine

Name a module <module_address> as <module_name>. This command assigns a user-defined name to a number representing a card address in hardware-dependent form (for example, a VME address).

The query form of this command, "ROUTe:MODule:DEFine? <module_name>" returns an <NR1> response showing what <module_address> is bound to <module_name>.

17.2.3 :DELeTe

ROUTe:MODule:DELeTe

17.2.3.1 :ALL

ROUTe:MODule:DELeTe:ALL

The ROUTe:MODule:DELeTe:ALL command deletes all module name bindings, and frees all names created by the MODule command.

17.2.3.2 **[[:NAME] <module_name>**

ROUTe:MODUle:DELeTe:NAME

The ROUTe:MODUle:DELeTe[:NAME] command removes a module name created by a MODUle command.

17.3 **:OPEN <channel_list>**

ROUTe:OPEN

The OPEN command allows specific channels to be opened or queried.

OPEN may be an overlapped command. The *OPC mechanism may be used to determine when switches are actually closed or opened. The sequence in which channels are opened is not guaranteed.

The query OPEN? allows the condition of switches to be queried. The instrument returns a 1 or 0 for each channel in the list, in the same order that the list is specified. A response of 0 means the channel is closed and a 1 means the channel is open.

At *RST, this value is device-dependent.

17.3.1 **:ALL**

ROUTe:OPEN:ALL

The OPEN:ALL command opens all the channels.

OPEN:ALL may be an overlapped command. The *OPC mechanism may be used to determine when switches are actually closed or opened. The sequence in which channels are opened is not guaranteed.

17.4 **:PATH**

ROUTe:PATH

The PATH subsystem controls the binding of <channel_list> elements.

17.4.1 **:CATalog?**

ROUTe:PATH:CATalog?

The ROUTe:PATH:CATalog? query returns a list of all currently defined <path_names>.

Response is a list of comma-separated strings. Each string contains a <path_name>. If no <path_name> elements are currently defined, then response is a null string ("").

17.4.2 **[[:DEFine] <path_name>,<channel_list>**

ROUTe:PATH:DEFine

The ROUTe:PATH:DEFine command assigns <path_name> as a user-specified way of referring to <channel_list>. This command allows the user to define a list of relays that can be opened and closed with the use of a <path_name>.

The query form of this command, "ROUTe:PATH:DEFine? <path_name>" returns a block containing the <channel_list> bound to <path_name>.

17.4.3 :DELeTe

ROUTe:PATH:DELeTe

17.4.3.1 :ALL

ROUTe:PATH:DELeTe:ALL

The ROUTe:PATH:DELeTe:ALL command deletes all path name bindings, and frees all names created by the PATH command.

ROUTe:SAMPlE

17.4.3.2 [[:NAME] <path_name>

ROUTe:PATH:DELeTe:NAME

The ROUTe:PATH:DELeTe[:NAME] command removes a path name created by a PATH command.

17.5 :SAMPlE

ROUTe:SAMPlE

This subsystem sets up the route for the sample gas to flow.

17.5.1 :CATalog?

:ROUTe:SAMPlE:CATalog?

Returns a comma-separated list of strings containing available sample points used for :ROUTe:SAMPlE:OPEN.

17.5.2 [[:OPEN] BAGIDILute|PREIPOSTIMIDICEFFiciency|NONE|ZERO|SPAN IVERify|MANifold

:ROUTe:SAMPlE[:OPEN]

This command causes flow from the specified sample point to the selected instrument(s). An execution error -200 shall occur if the <sample_point> references a configuration that cannot be achieved by the device. This command can also query the currently selected <sample_point>. The query response is a string. Each string contains a <sample_point>. If a <sample_point> is not currently selected, then the response is NONE.

At *RST the selected path goes to the idle state (NONE).

For emissions benches, the :ROUTe:SAMPlE command is used to route gases from the source to the measurement instrument. This command causes gases to flow to the instruments from the specified sample point. No measurement sequence is executed, however subsequent SENSE:DATA? queries will reflect the current values of the gases flowing through the analyzers. Gas routing may be implied during predefined procedures. On *RST the selected path goes to the idle state (NONE).

When the following <sample_point>s are implemented they shall conform to the following naming conventions:

- **BAG** — Causes gases from the specified bag sampling system to flow through the instruments. (Note: an appropriate CVS command is needed to complete this function.)
- **DILute** — Gases from the Sampler after dilution with ambient air, for example, the tunnel of a diesel sampling system

1999 SCPI Command Reference

- **PRE** — Gasses from between the engine and the catalytic converter.
- **POST** — Gasses from the vehicle exhaust after the catalyst but before dilution in the sampling system.
- **MID** — Gases from between the oxidizing and reducing catalysts.
- **CEfficiency** — Directs gases from each of the necessary sample points to each of the necessary sample manifolds in order to perform a catalyst efficiency test.
- **NONE** — Places the flow of gases in an idle state.
- **ZERO** — Directs zero gases to the selected instruments.
- **SPAN** — Directs span gases to the selected instruments for the selected ranges.
- **VERify** — Flows verification gases to all selected instruments
- **MANifold** — Enables Manifold as locally configured at the bench.

Examples:

```
:INSTRUMENT:DEFINE:GROUP MyGroup, "CONC1", "CONC2", "CONC3"  
:INSTRUMENT:SELECT MyGroup  
:ROUTE:SAMPLE ZERO
```

Routes zero gas to all analyzers defined by the group "MyGroup".

```
:INSTRUMENT:DEFINE:GROUP MyGroup, "CONC1", "CONC3", "CONC4"  
:INSTRUMENT:SELECT MyGroup  
:ROUTE:SAMPLE NONE
```

Discontinues any flows in progress and places the flow of gases into an idle state, for the group of instruments defined by "MyGroup".

```
:INSTRUMENT:DEFINE:GROUP MyGroup, "CONC2", "CONC3"  
:INSTRUMENT:SELECT MYGROUP  
:ROUTE:SAMPLE MANifold1
```

Flows engine exhaust from manifold setting 1 for the group of instruments defined by "MyGroup".

```
:INSTRUMENT:DEFINE:GROUP MyGroup, "CONC1", "CONC2", "CONC3"  
:INSTRUMENT:SELECT MyGroup  
:ROUTE:SAMPLE?
```

Returns: MANifold1

17.6 :SCAN <channel_list>

ROUTe:SCAN

The SCAN command specifies or queries a list of channels for the instrument to sequence through. The instrument will automatically step through the list of channels provided when the trigger subsystem device action occurs. The device action is to open the channel previously closed, then to close the next channel in the list. Whether this action is break-before-make is device-dependent.

The INITiate[:IMMediate] command must be sent in order to initiate scanning.

Execution of the SCAN command, and the resulting scan operation, may or may not open any channels which are already closed.

If the <channel_list> parameter contains a <channel_range>, the instrument will expand the range into its individual channels, and scan each channel individually. For example, (@1,3:5,9) will be scanned identically to (@1,3,4,5,9).

The query, which has no parameter, returns the scan list. The format of the response is an IEEE 488.2 definite length block which contains a <channel_list>. The instrument may return the scan list exactly as it was received, or it may return an equivalent list with paths or ranges expanded.

Note: The query form was added in 1994.

17.7 TERMinals FRONT|REAR|BOTH|NONE

ROUTe:TERMinals

FRONT opens the rear terminals and closes the front terminals (break before make). REAR opens the front terminals and closes the rear terminals (also break before make). BOTH closes both the front and rear terminals in a device-dependent order. NONE opens both the front and rear terminals in a device-dependent order.

At *RST, TERM is set to FRONT.

18 SENSE subsystem

The SENSE setup commands are divided into several sections. Each section or subsystem deals with controls that directly affect device-specific settings of the device and not those related to the signal-oriented characteristics. These commands are referenced through the SENSE node.

The SENSE node may be optional in a particular device. The reason that the SENSE is optional is to allow devices which are primarily sensors to accept shorter commands. The SENSE node cannot be optional if either SOURCE or ROUTE is optional, since only one node at a given level may be optional. For example, a typical counter may elect to make the SENSE node optional, since the most frequently used commands would be under the SENSE subsystem. If the counter also contained either source or routing functionality, these would be controlled through their respective nodes; however, the SOURCE or ROUTE nodes would be required. An optional node, such as SENSE, implies that the device shall accept and process commands with or without the optional node and have the same result. That is, the device is required to accept the optional node, if sent, without error.

In some instances, a sensor may contain subservient source or sensor functions. In such cases, the additional subservient functionality shall be placed as SENSE or SOURCE subnodes under the SENSE subsystem. Further, no optional nodes (SENSE, SOURCE or ROUTE) are permitted if such subservient functionality exists in a device. Otherwise conflicts in interpreting commands would occur, for example, between SENSE and [SOURCE:]SENSE if the SOURCE were allowed to be optional.

The FREQUENCY subsystem contains several sets of commands which have complex couplings. This includes the swept commands START, STOP, CENTER, and SPAN as one set. Sending any one of a set singly will affect two others of the set. However, if two are sent in a single message unit, then these two should be set to the values specified and the other two changed. This is in accordance with the style guidelines and *IEEE 488.2*. If any command in the set is implemented then all the commands in the set shall be implemented.

If the requested setting is unachievable given the other instrument settings, then an error must be generated (-221 "Settings conflict"). The device may resolve the conflict in a device-dependent manner (for example, change START, STOP, CENTER, and/or SPAN) to resolve the error. Note that when more than one of the four sweep settings are received in the same program message unit, the sweep will be determined by the last two received.

The coupled commands may define commands as subnodes which alter these couplings. The command description will define how the couplings are altered. However, any command which alters couplings must default to the couplings described in this section as the *RST couplings.

When CENTER or SPAN is changed the following couplings apply:

SPAN	=	SPAN
CENTER	=	CENTER
STOP	=	CENTER + (SPAN/2)
START	=	CENTER - (SPAN/2)

1999 SCPI Command Reference

When START or STOP changed the following couplings apply:

$$\begin{array}{llll} \text{START} & = & \text{START} & \text{CENTer} = (\text{START} + \text{STOP})/2 \\ \text{STOP} & = & \text{STOP} & \text{SPAN} = \text{STOP} - \text{START} \end{array}$$

The CURRENT, VOLTage and POWER subsystems contain several commands which have complex couplings. These commands exist under RANGE: they are UPPER, LOWER, OFFSET, and PTPeak. UPPER and LOWER are considered as one set, with OFFSET and PTPeak as another set. Sending any one of a set singly shall affect the two values of those commands in the other set. However, if two are sent in a single message unit, then these two should be set to the values specified and the unspecified two values shall be changed. This is in accordance with the style guidelines and IEEE 488.2. If either OFFSET or PTPeak is implemented, then all four commands, UPPER, LOWER, OFFSET, and PTPeak shall also be implemented.

When UPPER or LOWER is changed the following couplings apply:

$$\begin{array}{llll} \text{UPPER} & = & \text{UPPER} & \text{OFFSET} = (\text{UPPER} + \text{LOWER})/2 \\ \text{LOWER} & = & \text{LOWER} & \text{PTPeak} = \text{UPPER} - \text{LOWER} \end{array}$$

When OFFSET or PTPeak is changed the following couplings apply:

$$\begin{array}{llll} \text{OFFSET} & = & \text{OFFSET} & \text{UPPER} = \text{OFFSET} + (\text{PTPeak}/2) \\ \text{PTPeak} & = & \text{PTPeak} & \text{LOWER} = \text{OFFSET} - (\text{PTPeak}/2) \end{array}$$

The SENSE subsystem for the dynamometer contains the commands needed to sense real-time measurements taken from the dynamometer system. This includes force, speed, frictional losses, acceleration rate, distance, and simulation error. These commands are used in conjunction with the TRIGer Subsystem for collecting data in blocks. See SCPI 1995 Volume 2 Command Reference chapter 18 section 11 for further detail on the DATA and FUNCTION subsystems.

First Level SENSE Keywords

[[:SENSe]

:AM	1991
:AVERage	1992
:BANDwidth[:BWIDth	
:CONCetration	1999
:CONDition	
:CORRection	
:CURRent	
:DATA?	1999
:DETector	
:DISTance	1999
:FILTer	1991
:FM	1991
:FREQuency	
:FUNCtion	
:LIST	
:MIXer	
:PM	1991
:POWer	
:RESistance[:FRESistance	
:ROSCillator	
:SMOothing	
:SSB	
:STABilize	1999
:SWEep	
:VOLTage	
:WINDow	

18.1 **AM Subsystem**

SENSe:AM

The Amplitude Modulation subsystem is used to set up the modulation controls and parameters associated with sensing and demodulating amplitude modulated signals.

KEYWORD	PARAMETER FORM	NOTES
:AM		1991
[:DEPTh]		1991
:RANGe		1991
:AUTO	<Boolean> ONCE	1991
[:UPPer]	<numeric_value>	1991
:LOWer	<numeric_value>	1991
:TYPE	LINear LOGarithmic	1994

18.1.1 **[:DEPTh]**

SENSe:AM:DEPTh

This node is used to collect together the depth parameters.

18.1.1.1 **:RANGe**

SENSe:AM:DEPTh:RANGe

Sets the range for the sensor function.

18.1.1.1.1 **:AUTO <Boolean> | ONCE**

SENSe:AM:DEPTh:RANGe:AUTO

Sets the RANGe to the value determined to give the most dynamic range without overloading. The actual algorithm is device-dependent. Selecting AUTO ONCE will have the effect of setting AUTO to ON and then OFF.

At *RST, this value is set to ON.

18.1.1.1.2 **[:UPPer] <numeric_value>**

SENSe:AM:DEPTh:RANGe:UPPer

Specifies the maximum signal level expected for the sensor input.

At *RST, this value is device-dependent.

18.1.1.1.3 **:LOWer <numeric_value>**

SENSe:AM:DEPTh:RANGe:LOWer

Specifies the smallest signal level expected for the sensor input. This allows setting for the best dynamic range. This function is coupled to UPPer.

At *RST, this value is device-dependent.

18.1.2 **:TYPE LINear|LOGarithmic**

SENSe:AM:TYPE

SENSe:AM:TYPE sets or queries the type of amplitude demodulation technique the demodulator uses. LINear means the demodulated signal is derived directly from the amplitude of the signal in a linear way. LOGarithmic means the demodulated signal is derived from the amplitude of the signal in a logarithmic way. The amplitude of the demodulated signal is proportional to the log-value of the amplitude of the signal.

1999 SCPI Command Reference

At *RST, this value is set to LINear.

18.2 AVERage Subsystem

SENSe:AVERage

The purpose of this subsystem is to control averaging used to improve the measurement. For example to improve the accuracy or to acquire maximum/minimum measurements. Analytical post-acquisition statistical functions are controlled within the CALCulate block. The AVERage subsystem combines successive measurements to produce a new composite result. The new result has the same number of points and control axis as the original measurement.

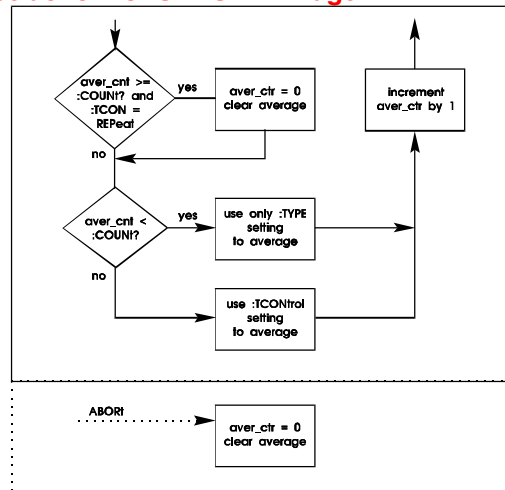
A :COUNT command is provided to specify the number of measurements to combine. When :COUNT averages are reached, the TerminalControl switch specifies the operation of the average subsystem. The TerminalControl switch can be configured to continue replacing old values with new values (MOVing), to exponentially weight new measurements (EXponential), to automatically repeat (REPeat), or to continue adding new measurements in the NORMal fashion, as specified by :TYPE.

The average operation differs from CALC:SMOothing in the fact that AVERage combines the *j*th point of each measurement with the *j*th point of preceeding measurements, whereas in SMOothing the *j*th point is processed with adjacent points (... , *j*-2, *j*-1, *j*, *j*+1, *j*+2, ...) of the current measurement set.

Note: The notation X_n^* indicates the complex conjugate of a number. So:

$$XX^* = \text{MAG}^2(X) = |X|^2$$

18.2.1 A typical device action for SENSe:AVERage



1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
:AVERage		1992
:COUNT	<numeric_value>	1992
:AUTO	<Boolean> ONCE	1992
[:STATe]	<Boolean>	1992
:TCONtrol	EXPonential MOVing NORMal REPeat	1992
:TYPE	COMPLex ENVELOpe MAXimum	1992
	MINimum RMS SCALar	1992

18.2.2 :COUNT <numeric_value>

SENSe:AVERage:COUNT

Specifies the number of measurements to combine using the :TYPE setting. After :COUNT measurements have been averaged, the operation of the AVERage subsystem is controlled by the setting of the :TCONtrol node.

The count for array based results is independent of the number of points in the result array. For example, an instrument which normally provides a measurement result which is a 401 point array would specify an average count of two to average two results, not 802.

When averaging is ON, some devices may automatically set :COUNT values in the TRIGger subsystem based on the AVER:COUNT value, such that the TRIGger subsystem provides enough triggers for the average.

At *RST, this value is device dependent.

18.2.2.1 :AUTO <Boolean> | ONCE

SENSe:AVERage:COUNT:AUTO

AUTO ON causes the device to select a value for :COUNT which is appropriate for the current measurement. Selecting AUTO ONCE will have the effect of setting AUTO ON and then OFF.

At *RST, this value is set to OFF.

18.2.3 [:STATe] <Boolean>

SENSe:AVERage:STATE

The STATe command is used to turn averaging ON and OFF.

At *RST, this value is set to OFF.

18.2.4 :TCONtrol EXPonential | MOVing | NORMal | REPeat

SENSe:AVERage:TCONtrol

TerminalCONtrol specifies the action of the AVERage subsystem when more than AVERage:COUNT measurement results are generated.

The parameter has the following meanings:

EXPonential Continue the average with an exponential weighting applied to old values.
The additional averages are weighted using:

1999 SCPI Command Reference

For :TYPE SCALar when X_n is real and :TYPE COMPlax when X_n is either real or complex

$$AVG_n = \frac{1}{k} X_n + \frac{k-1}{k} AVG_{n-1}$$

For :TYPE SCALar when X_n is complex

$$AVG_n = \frac{1}{k} MAG(X_n) + \frac{k-1}{k} AVG_{n-1}$$

For :TYPE RMS when X_n is real

$$AVG_n = \sqrt{\frac{1}{k} X_n^2 + \frac{k-1}{k} AVG_{n-1}^2}$$

For :TYPE RMS when X_n is complex

$$AVG_n = \sqrt{\frac{1}{k} X_n X_n^* + \frac{k-1}{k} AVG_{n-1}^2}$$

The exponential factor k is equal to the AVERAge:COUNT value. Some instruments may use an approximation to true exponential.

Operation with EXPOntial :TCON is undefined with a :TYPE setting of MINimum, MAXimum, or ENVELOpe.

MOVing	As new data is added the oldest data is discarded.
REPeat	Clear average data and counter and restart the average process.
NORMal	Additional averages continue to be accumulated according to the :TYPE selection.

At *RST, this value is device dependent.

18.2.5 :TYPE COMPlax | ENVELOpe | MAXimum | MINimum | RMS | SCALar SENSe:AVERAge:TYPE

This command selects the type of averaging, as follows:

COMPlax	The points in the summation are treated as real/imaginary pairs.

$$AVG(n) = \frac{1}{n} \sum_{i=1}^n X_i$$

ENVELOpe	Both the MIN and MAX values are retained.

If X_n is real

MAXimum	$AVG(n) = MAX(X_1 \dots X_n)$

MINimum	$AVG(n) = MIN(X_1 \dots X_n)$

RMS
$$AVG(n) = \sqrt{\frac{1}{n} \sum_{i=1}^n X_i^2}$$

SCALar The scalar magnitude of each point is averaged.

$$AVG(n) = \frac{1}{n} \sum_{i=1}^n X_i$$

If X_n is complex

MAXimum $AVG(n) = \text{MAX}(\text{MAG}(X_1) \dots \text{MAG}(X_n))$

MINimum $AVG(n) = \text{MIN}(\text{MAG}(X_1) \dots \text{MAG}(X_n))$

RMS
$$AVG(n) = \sqrt{\frac{1}{n} \sum_{i=1}^n X_i X_i^*}$$

SCALar The scalar magnitude of each point is averaged.

$$AVG(n) = \frac{1}{n} \sum_{i=1}^n |X_i|$$

At *RST, this value is device dependent.

18.3 BANDwidth|BWIDth Subsystem

SENSe:BANDwidth

This subsystem controls the characteristics of the filter bandwidth of the instrument. Instruments must accept both forms, if they accept either.

KEYWORD	PARAMETER FORM	NOTES
:BANDwidth :BWIDth		
[:RESolution]	<numeric_value>	
:AUTO	<Boolean> ONCE	
:RATio	<numeric_value>	
:TRACK	<Boolean>	
:VIDeo	<numeric_value>	
:AUTO	<Boolean> ONCE	
:RATio	<numeric_value>	

18.3.1 [:RESolution] <numeric_value>

SENSe:BANDwidth:RESolution

Controls the resolution bandwidth of the instrument. Resolution Bandwidth is defined as the bandwidth in Hz of the predetection bandpass filtering. This command has units of Hz.

At *RST, the value of this function is device-dependent.

18.3.1.1 :AUTO <Boolean>|ONCE

SENSe:BANDwidth:RESolution:AUTO

When enabled, the value of resolution bandwidth is coupled to other parameters of the measurement. The exact coupling equation is machine-dependent.

At *RST, this function is set to ON.

18.3.1.2 :RATio <numeric_value>

SENSe:BANDwidth:RESolution:RATio

Controls the ratio of resolution bandwidth to span when the two are coupled. Coupled is enabled when AUTO is ON. The <numeric_value> is RESolution(Hz)/SPAN(Hz).

At *RST, the value of this function is instrument-dependent.

18.3.1.3 :TRACK <Boolean>

SENSe:BANDwidth:RESolution:TRACK

This function allows the resolution bandwidth to dynamically change during a logarithmic frequency sweep. Logarithmic sweep is enabled via the SENSE:SWEep:SPACing command.

At *RST, the value of this function is ON.

18.3.2 :VIDeo <numeric_value>

SENSe:BANDwidth:VIDeo

Controls the video filtering, a particular type of post-detection filtering. This has units of Hertz.

At *RST, the value of this function is instrument-dependent.

18.3.2.1 :**AUTO** <Boolean>**IONCE**

SENSe:BANDwidth:VIDeo:AUTO

Couples the value of video bandwidth to instrument-determined values.

At *RST, the value of this setting is ON.

18.3.2.2 :**RATio** <numeric_value>

SENSe:BANDwidth:VIDeo:RATio

Controls the ratio of video bandwidth to resolution bandwidth when the two are coupled.

Coupled is enabled when AUTO is ON. The <numeric_value> is
VIDeo(Hz)/RESolution(Hz).

At *RST, the value of this function is instrument-dependent.

18.4 CONCentration Subsystem

SENSe:CONCentration

This subsystem configures the CONCentration measurement functions and parameters. An instance of this subsystem exists for each range in an instrument. The data defined and queried and the events will only be performed on the currently selected range of the currently selected instrument(s).

KEYWORD	PARAMETER FORM	COMMENTS
:CONCentration		1999
:CSET	<numeric_value>,<numeric_value>	1999
:LOWer	<numeric_value>	1999
:LSET	POLYnomial<n> SRATional<n>,<numeric_value>	1999
:RANGe		1999
:AUTO		1999
:LOWer	<numeric_value>	1999
[:STATe]	<Boolean>	1999
:UPPer	<numeric_value>	1999
[:FIXed]	<numeric_value>	1999
:TALign	<numeric_value>	1999
:UPPer	<numeric_value>	1999

18.4.1 :CSET <numeric_value>,<numeric_value>

SENSe:CONCentration:CSET

This command specifies or queries the correction set(s) used for the current range(s). The parameters are the slope and the Y intercept used in the latest correction calculation.

At *RST these values are not changed.

18.4.2 :LOWer <numeric_value>

SENSe:CONCentration:LOWer

This command specifies or queries the lowest measurable concentration for the currently-selected range for the selected instruments.

At *RST these values are not changed.

18.4.3 :LSET POLYnomial<n> | SRATional<n>,<numeric_value>{,<numeric_value>}

SENSe:CONCentration:LSET

This command sets or returns the curve fit type and coefficients that are currently in use. Coefficients are listed starting with the 0th order term. The first parameter is the curve type, which consists of either POLYnomial for a polynomial curve fit or SRATional for a simple rational polynomial curve fit, followed by a positive integer <n>, which is the curve order. The additional parameters are the curve coefficients. These are listed starting with the 0th order term. Note that the number of coefficients depends on the order of the curve - there are exactly n+1 coefficients for a curve of order n. The parameter list for LSET includes a curve type and associated coefficients for the currently selected range of each currently selected analyzer. See :CALibration:LINearize:CURVe:FIT for examples.

Note that the :CALibration:LINearize:ACcept command sets the parameter values.

At *RST these values are device dependent.

18.4.4 :RANGe

:SENSe:CONCenTration:RANGe

This node allows the setting of a specific range or turns off/on auto-ranging mode for the pre-selected instrument(s).

18.4.4.1 :AUTO

:SENSe:CONCenTration:RANGe:AUTO

The AUTO command controls the automatic range switching feature.

18.4.4.1.1 :LOWer <numeric_value>

:SENSe:CONCenTration:RANGe:AUTO:LOWer

This command specifies or queries the lower CONCenTration(s) in ppm for automatic ranging for the selected instrument(s). When a CONCenTration is decreasing, this value is the switch point for this range.

At *RST, these values are not changed.

18.4.4.1.2 [:STATe] <Boolean>

:SENSe:CONCenTration:RANGe:AUTO[:STATe]

The STATe command turns the automatic range switching feature on or off. In automatic range mode, the instrument will switch ranges to keep the measured values within the limits defined in SENSe:CONCenTration:RANGe:AUTO:LOWer and :UPPer. Turning AUTO off will keep the device at the current range.

At *RST, the STATe is device-dependent.

18.4.4.1.3 :UPPer <numeric_value>

:SENSe:CONCenTration:RANGe:AUTO:UPPer

This command specifies or queries the upper CONCenTration(s) in ppm for automatic ranging for the selected instrument(s). When a CONCenTration is increasing, this value is the switch point for this range.

At *RST, these values are not changed.

18.4.4.2 [:FIXed] <numeric_value>

:SENSe:CONCenTration:RANGe:FIXed

This command specifies or queries the range number of the already-selected instrument(s). The instrument(s) will continue to measure in this range until a subsequent range command changes the range(s). If auto-ranging is enabled, it will be disabled upon specifying range(s) with this command, but will remain unchanged if the range(s) are queried with this command.

At *RST, the settings of :FIXed are device-dependent.

18.4.5 :TALign <numeric_value>

:SENSe:CONCenTration:TALign <numeric_value>

Sets (or queries) the time delay in seconds for the Time Alignment of the continuously measured data.

At *RST these values are not changed.

18.4.6 :UPPer <numeric_value>

:SENSe:CONCetration:UPPer

This command specifies or queries the full scale concentration for the currently-selected range for the selected instrument(s).

At *RST these values are not changed.

18.5 **CONDition Subsystem**

SENSe:CONDition

This subsystem controls the electrical characteristics of a sensor that determines the ON or OFF condition of a digital signal.

KEYWORD	PARAMETER FORM	COMMENTS
:CONDition		1993
LEVel	<numeric_value> TTL ECL	1993

18.5.1 **:LEVel <numeric_value> | TTL ECL**

SENSe:CONDition:LEVel

This command specifies the characteristics that causes the transition between an ON and OFF condition of the sensed signal. When the value of the input signal exceeds the specified level, the condition is qualified as ON or 1 (True) for positive logic type, and qualified as OFF or 0 (False) when it is associated with negative types of logic. For positive logic, the condition is sensed as OFF or 0 (False) when the input value is below the level; for negative logic this value is qualified as an ON or 1 (True) condition.

The default units are the amplitude units associated with the currently sensed signal.

At *RST this value is device dependent.

18.6 **CORRection Subsystem**

SENSe:CORRection

The correction subsystem provides for known external losses or gains. In contrast to CALibration, these losses usually are dependent on individual measurement setups.

Correction data may be entered and stored in several ways. A mechanism is provided to automatically measure a reference standard(s) and store the resulting “correction set” in a trace. The trace can be selected at any time as the active “correction set”. This trace could also be generated by an external device and transferred to the instrument over a computer interface. Commands are also provided to enter individual losses, gains, offsets, delays, etc. that the instrument should use in correcting the data. An instrument is not required to implement all of the methods.

The commands in this subsystem are designed to correct for losses, gains, and offsets in external probes or transducers. For instance, they can be used to inform the sensor that a 10X probe has been attached to the instrument. This way, the data and measurement results will be corrected so that they are not 1/10th their actual value. In instruments that automatically detect the attachment of a probe, these commands may not be necessary.

KEYWORD	PARAMETER FORM	COMMENTS
:CORRection		
:AUTO		[event] 1999
:CALCulate		[event] 1999
:COLlect		
[:ACQuire]	STANdard	
:METHod	TPORt	
:SAVE	<trace_name>	
:CSET		
[:SElect]	<name>	1992
:STATe	<Boolean>	
:EDELay		
:DISTance	<numeric_value>	
:STATe	<Boolean>	
[:TIME]	<numeric_value>	
:IMPedance		1993
[:INPut]::OUTPut		1993
[:MAGNitude]	<numeric_value>	1993
:STATe	<Boolean>	1993
:LOSS[:GAIN]::SLOPe		
[:INPut]::OUTPut		1992
:AUTO	ONIOFF	
[:MAGNitude]	<numeric_value>	
:PHASe	<numeric_value>	
:STATe	<Boolean>	
:OFFSet		
[:MAGNitude]	<numeric_value>	

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	COMMENTS
:PHASe	<numeric_value>	
:STATe	<Boolean>	
:RVELocity		
:COAX	<numeric_value>	
:MEDium	COAXIWAVEguide	
:STATe	<Boolean>	
:WAVEguide	<numeric_value>	
:FCUToff	<numeric_value>	
:SPOint		1999
:ACQuire		[event] 1999
:DTOLerance	<numeric_value>	1999
[:STATe]	<Boolean>	
:ZERO		1999
:ACQuire		[event] 1999
:DTOLerance	<numeric_value>	1999

18.6.1 :**AUTO**

SENSe:CORRection:AUTO

AUTO performs an automatic correction of the selected instruments. The procedure shall be defined according to the manufacturer's recommended procedure. This command is an event and therefore has no *RST value associated with it.

18.6.2 :**CALCulate**

SENSe:CORRection:CALCulate

If a calculation is needed for the CORRection subsystem, this command initiates the calculation.

This command is an event, and has no *RST value.

18.6.3 :**COLlect**

SENSe:CORRection:COLlect

Deals with the collection of the correction data. This data is collected directly by the device by measurements of standards.

18.6.3.1 **[:ACQuire] STANDard**

SENSe:CORRection:COLlect:ACQuire

A measurement will be performed and saved as the data for standard of the chosen correction method. The number of standards required to be measured for the correction is determined by the method chosen. If a standard is specified that does not exist in the chosen method, then an error is generated and no measurement is taken. This measurement will not affect any of the users settings for regular measurements.

For complex corrections, several standards will be measured and then followed by a SAVE command. The parameter on the save command will be the name of the correction set. The standards themselves are device-specific.

This command is an event, does not have a *RST value, and is not queryable.

18.6.3.2 :METHod TPOrt

SENSe:CORRection:COLlect:METHod

Selects the correction method to be used for the correction that is about to be performed. If only one correction method exists in the instrument then this command is not required.

The various settings of the parameters have the following meanings:

TPOrt Two Port

At *RST, this value is device-dependent.

18.6.3.3 :SAVE [<trace_name>]

SENSe:CORRection:COLlect:SAVE

Calculates the correction data using the current method selection and then saves the correction data. If only one correction array exists then it is saved there. Otherwise it is saved in the set specified by <trace_name>. Note that the parameter <trace_name> must be defined and exist in the instrument. This command will not create a trace. Trace definition is done in the trace subsystem. Note that the correction array may be a single value.

Depending on the complexity of the correction method chosen, a calculation may or may not be performed on the measured arrays to determine the correction data. For example, if the chosen method has only one standard, then no calculation will be performed.

This command is an event, does not have a *RST value, and is not queryable.

18.6.4 :CSET

SENSe:CORRection:CSET

The Correction SET subsystem is used to select the active correction set.

18.6.4.1 [:SElect] <name>

SENSe:CORRection:CSET:SElect

Specifies the active CORRection set. This is the set that is used when CORR:CSET:STATe ON is specified. <name> may be either a <trace_name>, or a <table_name> which must be defined and exist in the instrument. This command will not create either <trace_name>, or a <table_name>. Trace definition is done in the TRACe subsystem. Table definition is done in the MEMory subsystem.

If <table_name> specifies a MEMory:TABLE the various lists must have an equal number of points, or execution error -226 (Lists not same length) will be generated.

Those lists in a TABLE which are defined in the CORRection subsystem will be applied when CORRection:STATe is ON. Other items in a TABLE will be ignored without generating an error. This allows the same TABLE to be shared with other subsystems.

At *RST this value is device dependent.

18.6.4.2 :STATe <Boolean>

SENSe:CORRection:CSET:STATe

Determines whether the correction data defined in the selected set is applied to the measurement.

At *RST, this function is OFF. Off is chosen because all of the correction sets may be empty. The instrument must collect data before correction can be applied.

18.6.5 :EDELay

SENSe:CORRection:EDELay

The Electrical DELay is used to compensate for delays in the signal path.

This command is used to correct external factors that the instrument cannot measure or detect.

TIME and DISTance are coupled by the equation:

$$TIME = RVELocity * DISTance$$

RVELocity is not coupled to either TIME or DISTance.

18.6.5.1 :DISTance <numeric_value>

SENSe:CORRection:EDELay:DISTance

This command will set the electrical delay with the distance parameter given. The effective delay will be computed using this parameter and the relative velocity set in the instrument. The units for this command are meters. A value accepted here will modify the time setting because they are the same parameter in different forms. Implementation of this command requires implementation of CORRection:EDELay[:TIME].

At *RST, the value of this setting is zero.

18.6.5.2 :STATe <Boolean>

SENSe:CORRection:EDELay:STATe

This function is used to enable or disable the electrical delay correction.

At *RST, this function is set to OFF.

18.6.5.3 [:TIME] <numeric_value>

SENSe:CORRection:EDELay:TIME

This command will set the electrical delay with the time parameter given. The units for this command are seconds. A value accepted here will modify the distance setting because they are the same parameter in different forms.

At *RST, the value of this setting is zero.

18.6.6 :IMPedance

SENSe:CORRection:IMPedance

This is the termination impedance provided external to the device.

18.6.6.1 [:INPut]I:OUTPut

SENSe:CORRection:IMPedance:INPut

The INPut and OUTPut nodes are used to reference measurements made at the instrument's input and output, respectively. Most sensor instruments are only capable of making measurements at its input ports; however, some instruments are capable of making measurements on output ports that are sourcing a signal.

18.6.6.1.1 **[[:MAGNitude] <numeric_value>**

SENSe:CORRection:IMPedance:INPut:MAGNitude

Sets the magnitude of the impedance. The default units are Ohms

At *RST, this value is device-dependant.

18.6.6.2 **:STATe <Boolean>**

SENSe:CORRection:IMPedance:STATe

The STATe command is used to enable or disable external IMPedance correction factors.

At *RST, this function is set to OFF.

18.6.7 **:LOSSI:GAINI:SLOPe**

SENSe:CORRection:LOSS

These commands are used either to provide a mechanism to apply simple correction or to correct external factors that have not been accounted for in the correction set. These correction factors are typically applied to measurements made at the device input; however, in cases where measurements can be at the device output a separate set of correction factors may be applied.

Loss, gain and slope effect the measurement values as a multiplier for linear units such as VOLT and as additive (or subtractive) value for logarithmic units such as DB.

:LOSS This command is used to set the anticipated loss in the system, excluding those factors covered by the active correction set. The device then corrects every measurement by this factor to compensate for the loss. LOSS is coupled to GAIN by the equation $LOSS = 1/GAIN$ when the default unit is linear and $LOSS = -GAIN$ when the default unit is logarithmic.

:GAIN This command is used to set the anticipated gain in the system, excluding those factors covered by the active correction set. The device then corrects every measurement by this factor to compensate for the gain. GAIN is coupled to LOSS by the equation $GAIN = 1/LOSS$ when the default unit is linear and $GAIN = -LOSS$ when the default unit is logarithmic.

:SLOPe This node defines an increasing or decreasing value that is added to the signal. This correction value is zero at a zero value of the x axis and increase or decreases from there.

18.6.7.1 **[[:INPut]]:OUTPut**

SENSe:CORRection:LOSS:INPut

The INPut and OUTPut nodes are used to reference measurements made at the instrument's input and output, respectively. Most sensor instruments are only capable of making measurements at its input ports; however, some instruments are capable of making measurements on output ports that are sourcing a signal.

18.6.7.1.1 **:AUTO ONIOFF**

SENSe:CORRection:LOSS:INPut:AUTO

When AUTO is ON the OUTPut correction values must track those of the INPut.

At *RST, AUTO is set to ON.

18.6.7.1.2 [:MAGNitude] <numeric_value>

SENSe:CORRection:LOSS:INPut:MAGNitude

This is the magnitude value of the correction data for LOSS, GAIN or SLOPe. The units associated with LOSS and GAIN are the current relative amplitude unit. The unit associated with SLOPe is:

<current realtive amplitude unit>/Hz

After a *RST, this value should no longer affect the data being measured. Thus its value is set to zero at *RST.

18.6.7.1.3 :PHASe <numeric_value>

SENSe:CORRection:LOSS:INPut:PHASe

This is the phase value of the correction data for LOSS, GAIN or SLOPe. The unit associated with SLOPe is current angle units/Hertz. The units associated with LOSS and GAIN are current angle units.

After a *RST, this value should no longer affect the data being measured. Thus its value is set to zero at *RST.

18.6.7.2 :STATe <Boolean>

SENSe:CORRection:LOSS:STATe

The STATe command is used to enable or disable the LOSS, GAIN or SLOPe correction factors.

At *RST, this function is set to OFF.

18.6.8 :OFFSet

SENSe:CORRection:OFFSet

This node defines a single value that is added to every point measured by the instrument. If the instrument is a trace-based machine, then this value is added to every point in the trace. If the instrument measures a single value then that value is modified by this setting. This command shall be used with the currently selected units. It is used to correct external factors that the instrument cannot measure or detect.

18.6.8.1 [:MAGNitude] <numeric_value>

SENSe:CORRection:OFFSet:MAGNitude

This is the magnitude value of the correction offset data.

After a *RST, this value should no longer affect the data being measured. Its value is set to zero at *RST.

18.6.8.2 :PHASe <numeric_value>

SENSe:CORRection:OFFSet:PHASe

This is the phase value of the correction offset data. The units associated with this are the currently selected angle units.

18.6.8.3 :STaTe <Boolean>

SENSe:CORRection:OFFSet:STaTe

This function is used to enable or disable the offset correction.

At *RST, this function is set to OFF.

After a *RST, this value should no longer affect the data being measured. Its value is set to zero at *RST .

18.6.9 :RVELOCITY

SENSe:CORRection:RVELOCITY

This command corrects for the relative velocity of the medium of the Device Under Test. This factor will be used in any delay calculation performed in the instrument.

This command is used to correct external factors that the instrument cannot measure or detect.

TIME and DISTance are coupled by the equation:

$$TIME = RVELOCITY * DISTance$$

RVELOCITY is not coupled to either TIME or DISTance.

18.6.9.1 :COAX <numeric_value>

SENSe:CORRection:RVELOCITY:COAX

This command sets the relative velocity factor for coaxial lines. The parameter is unitless.

At *RST, the value of this setting is one.

18.6.9.2 :MEDium COAXIWAVEguide

SENSe:CORRection:RVELOCITY:MEDium

This command selects the correction algorithm for the media through which the detected signal is transmitted. It will determine the method of calculation for any operation that uses relative velocity. For example, for COAX a linear phase shift would be computed if a delay was added, but for WAVEguide (a dispersive medium) a different phase shift is calculated.

At *RST, this value is device-dependent.

18.6.9.3 :STaTe <Boolean>

SENSe:CORRection:RVELOCITY:STaTe

This function is used to enable or disable the relative velocity correction.

At *RST, this function is set to OFF.

18.6.9.4 :WAVEguide <numeric_value>

SENSe:CORRection:RVELOCITY:WAVEguide

This command sets the relative velocity factor for waveguide. The parameter is unitless.

At *RST, the value of this setting is one.

18.6.9.4.1 :FCUToff <numeric_value>

SENSe:CORRection:RVELOCITY:WAVEguide:FCUToff

This command specifies the frequency cutoff of the waveguide medium. The units are Hertz.

At *RST, this value is device-dependent.

18.6.10 :SPOint

:SENSe:CORRection:SPOint

This subsystem controls the set point correction of the sensor.

For Emissions benches, this subsystem references the span gas of the instrument.

This command is an event and therefore has no *RST value associated with it.

18.6.10.1 :ACQuire

:SENSe:CORRection:SPOint:ACQuire

This command supplies a known signal to the selected instruments and waits for the stabilization criteria to be met..

This command is an event, and has no *RST value.

18.6.10.2 :DTOLerance <numeric_value>

:SENSe:CORRection:SPOint:DTOLerance

This command specifies or queries the drift tolerance for a set point correction procedure for the current range of the selected instrument(s).

For emission benches, this tolerance is the difference between the specified concentration referenced in the SpanGas and GasBottle tables and the measured concentration. Drift tolerance is in units of percent of full scale value.

The data defined and queried will only apply to the currently selected range of the currently selected instrument(s).

At *RST, this value is not changed.

18.6.11 [:STATe] <Boolean>

SENSe:CORRection:STATe

Determines whether the correction data defined in this section is applied to the measurement.

At *RST, this function is OFF.

18.6.12 :ZERO

:SENSe:CORRection:ZERO

This subsystem controls the zero correction of the sensor.

18.6.12.1 :ACQuire

:SENSe:CORRection:ZERO:ACQuire

This command sends a zero signal to the selected instruments and waits for the stabilization criteria to be met.

This command is an event, and has no *RST value

18.6.12.2 :DTOLerance <numeric_value>

:SENSe:CORRection:ZERO:DTOLerance

This command specifies the drift tolerance for a zero procedure for the selected instrument(s).

1999 SCPI Command Reference

At *RST, this value is not changed.

18.7 CURRent Subsystem

SENSe:CURRent

This subsection controls the current amplitude characteristics of the sensor.

KEYWORD	PARAMETER FORM	NOTES
:CURRent		
:ACI[:DC]		
:APERture	<numeric_value>	
:NPLCycles	<numeric_value>	1991
:ATTenuation	<numeric_value>	
:AUTO	<Boolean>	
:PROTection		
[:LEVel]	<numeric_value>	
:STATe		
:TRIPped?		[query only]
:CLEar		[no query]
:RANGe		
[:UPPer]	<numeric_value>	
:LOWer	<numeric_value>	
:AUTO	<Boolean> ONCE	
:DIRection	UPI DOWN EITHer	
:LLIMit	<numeric_value>	1996
:ULIMit	<numeric_value>	1996
:OFFSet		1991
:PTPeak		1991
:REFerence	<numeric_value>	
:STATe	<Boolean>	
:RESolution	<numeric_value>	
:AUTO	<Boolean> ONCE	
:DETector	INTernal EXTernal	1991

18.7.1 :ACI[:DC]

SENSe:CURRent:AC

This subsystem selects the alternating current or the direct current sensor. The default units are amperes.

18.7.1.1 :APERture <numeric_value>

SENSe:CURRent:AC:APERture

Specifies the acquisition/sampling/gate time for a single measurement point. The parameter has a unit of seconds.

At *RST, this value is device-dependent.

18.7.1.2 :NPLCycles <numeric_value>

SENSe:CURRent:AC:NPLCycles

Specifies the acquisition/sampling/gate time for a single measurement point in terms of the number of power line cycles. The parameter is unitless. If this command is implemented, the APERTure command must also be implemented.

The value is coupled to APERTure by the equation:

$$APERTure = NPLCycles / selected\ line\ frequency$$

At *RST, this value is device-dependent.

18.7.1.3 :ATTenuation <numeric_value>

SENSe:CURRent:AC:ATTenuation

Sets the ATTenuation level. Note that when setting the level to 10 dB the magnitude of the incoming signal will be decreased by 10 dB. Changing the ATTenuation changes RANGE by the same amount.

Default units are as determined in the UNITS system.

18.7.1.3.1 :AUTO <Boolean>

SENSe:CURRent:AC:ATTenuation:AUTO

Couples the attenuator to RANGE such that maximum dynamic range of the incoming signal is assured without overloading.

Programming a specified attenuation sets AUTO OFF.

At *RST, AUTO is set to ON.

18.7.1.4 :PROtection

SENSe:CURRent:AC:PROtection

Controls the protection circuits.

18.7.1.4.1 [:LEVel] <numeric_value>

SENSe:CURRent:AC:PROtection:LEVel

This command sets the input level at which the input protection circuit will trip.

18.7.1.4.2 :STATE <Boolean>

SENSe:CURRent:AC:PROtection:STATe

Controls whether the input protection circuit is enabled.

At *RST, this value is set to ON.

18.7.1.4.3 :TRIPped?

SENSe:CURRent:AC:PROtection:TRIPped?

This query returns a 1 if the protection circuit is tripped and a 0 if it is untripped. TRIPped only has a query form and thus has no associated *RST condition.

18.7.1.4.4 :CLEar

SENSe:CURRent:AC:PROtection:CLEar

Causes the protection circuit to be cleared.

This command is an event and has no associated *RST condition.

18.7.1.5 :RANGe

SENSe:CURRent:AC:RANGe

Sets the range for the amplitude sensor function. The RANGe endpoints may be specified in terms of the UPPer endpoint and the LOWer endpoint, alternatively the midpoint of the RANGe (OFFSet) and the span of the RANGe (PTPeak) may be used. A non-symmetrical range around zero may cause some instruments to introduce an actual offset to the signal entering the SENSe block.

18.7.1.5.1 [:UPPer] <numeric_value>

SENSe:CURRent:AC:RANGe:UPPer

Specifies the most positive signal level expected for the sensor input.

At *RST, this value is device-dependent.

18.7.1.5.2 :LOWer <numeric_value>

SENSe:CURRent:AC:RANGe:LOWer

Specifies the most negative signal level expected for the sensor input. This allows setting for the best dynamic range. This function is coupled to UPPer in a device-dependent manner.

At *RST, this value is device-dependent.

18.7.1.5.3 :AUTO <Boolean>IONCE

SENSe:CURRent:AC:RANGe:AUTO

Sets the RANGe to the value determined to give the most dynamic range without overloading. The actual algorithm is device-dependent.

At *RST, AUTO is set to ON.

18.7.1.5.3.1 :DIREction UPIDOWNIEITHer

SENSe:CURRent:AC:RANGe:AUTO:DIREction

The DIREction command defines the manner in which AUTO works. When UP is selected, the instrument shall only change the range automatically to ranges larger than the current range, as the input signal dictates. When DOWN is selected, the instrument shall only change the range automatically to ranges smaller than the current range, as the input signal dictates. When EITHER is selected the instrument may range in either direction.

At *RST, DIREction shall be set to EITHer.

18.7.1.5.3.2 :LLIMit <numeric_value>

SENSe:CURRent:AC:RANGe:AUTO:LLIMit

The Lower LIMit command sets the smallest range to which the instrument will go while autoranging. When AUTO is ON and the present RANGe is outside the new limits, the instrument may change the RANGe to a value inside the limits or wait until the next time the autoranging algorithm is performed.

At *RST, LLIMit shall be set to MINimum.

18.7.1.5.3.3 :ULIMit <numeric_value>

SENSe:CURRent:AC:RANGe:AUTO:ULIMit

The Upper LIMit command sets the largest range to which the instrument will go while

autoranging. When AUTO is ON and the present RANGE is outside the new limits, the instrument may change the RANGE to a value inside the limits or wait until the next time the autoranging algorithm is performed.

At *RST, ULIMit shall be set to MAXimum.

18.7.1.5.4 :OFFSet <numeric_value>

SENSe:CURRent:AC:RANGe:OFFSet

OFFSet determines the midpoint of the range.

At *RST, this value is device-dependent.

18.7.1.5.5 :PTPeak <numeric_value>

SENSe:CURRent:AC:RANGe:PTPeak

Peak To Peak specifies the dynamic range required for the sensor.

At *RST, this value is device-dependent.

18.7.1.6 :REFeRence <numeric_value>

SENSe:CURRent:AC:REFeRence

Sets a reference amplitude for sensor instruments. This command differs in intent from the reference correction in that this is intended to be a measurement offset where a user wishes to reference a measurement to some known value.

At *RST, this value is device-dependent, but STATe is set to OFF.

18.7.1.6.1 :STATe <Boolean>

SENSe:CURRent:AC:REFeRence:STATe

Determines whether amplitude is measured in absolute or relative mode. If STATe is ON, then amplitude is referenced to the value set in REFeRence.

At *RST, this value is set to OFF.

18.7.1.7 :RESolution <numeric_value>

SENSe:CURRent:AC:RESolution

Specifies the absolute resolution of the measurement. For example, if a measurement of 10 Volts is specified with a resolution of .01 Volts, the measurement is returned with a resolution of .01 Volts (that is, 10.00).

RANGe is not coupled to RESolution. Setting RANGe shall not cause RESolution to change. Setting RESolution shall not cause RANGe to change.

At *RST, this value is device-dependent.

18.7.1.7.1 :AUTO <Boolean>|ONCE

SENSe:CURRent:AC:RESolution:AUTO

Allows the system to determine the best resolution for the other measurement conditions.

Explicitly setting a value for :RESolution will turn AUTO OFF.

At *RST, this value is set to ON.

18.7.2 :DETECTOR INTERNAL | EXTERNAL

SENSe:CURRent:DETECTOR

The DETECTOR can be specified as internal or external.

At *RST, this value is set to INT.

18.8 DETector Subsystem

SENSe:DETEctor

This subsystem controls the acquisition of data points, including how to choose points and the linear characteristics of the sensor.

KEYWORD	PARAMETER FORM	NOTES
:DETEctor		
:BANDwidth	<numeric_value>	1991
l:BWIDth		1991
[[:FUNCTion]	GROund NEGative POSitive SAMPle SIGNal PAVerage AAVerage RMS	1991
:AUTO	<Boolean> ONCE	
:SHAPE	LINEar LOGarithmic	

18.8.1 :BANDwidth | BWIDth

<numeric_value>

SENSe:DETEctor:BANDwidth

The BANDwidth function controls the bandwidth after an intermediate signal has been processed within the SENSe block. Instruments must accept both forms, if they accept either.

18.8.2 [:FUNCTion] <detector function>

SENSe:DETEctor:FUNCTion

Specifies the detector sampling characteristics. The various settings have the following meanings:

- GROund: The input detector is grounded. Thus no signal is acquired.
- NEGative: The minimum signal detected during the acquisition of the particular point is chosen.
- POSitive: The maximum signal detected during the acquisition of the particular point is chosen.
- SAMPle: A single arbitrary point detected is chosen.
- SIGNal: The detector chooses points so that envelope noise is maximized and positive signals above some threshold are also maximized.
- PAVerage: PeakAVerage is the average of the minimum and maximum excursions of the signal. $PAV = (\text{absolute minimum peak} + \text{absolute maximum peak}) / 2$
- AAVerage: AbsoluteAVerage averages the absolute values of a sine wave input and processes the value to read RMS values.
- RMS: True Root Mean Square signal detector.

At *RST, this value is device-dependent.

18.8.2.1 **:AUTO <Boolean>|ONCE**

SENSe:DETEctor:FUNCTion:AUTO

Couples the detector function to several settings within the instrument. When AUTO ON is selected, the detector type is determined by the system in order to give the “best” quality of data.

Explicitly selecting a detector will set AUTO OFF.

At *RST, this parameter is set to ON.

18.8.3 **:SHAPE LINear|LOGarithmic**

SENSe:DETEctor:SHAPE

Controls the linearity of the detector. This setting is coupled to DETector:FUNCTion in a device-dependent manner.

At *RST, this value is device-dependent.

18.9 **DISTance Subsystem**

SENSe:DISTance

This node is used to set the parameters associated with sensing distance.

KEYWORD	PARAMETER FORM	NOTES
:DISTance		1999
:RESet		[event; no query]1999

18.9.1 **:RESet**

:SENSe:DISTance:RESet

This command will set DISTance to 0. This command is not queryable.

At *RST the DISTance is set to zero.

18.10 FILTER Subsystem

SENSe:FILTer

The FILTer function allows a filter to be inserted after an intermediate signal has been processed within the SENSE block. This FILTer subsystem is not interchangeable with the INPut:FILTer subsystem.

KEYWORD	PARAMETER FORM	NOTES
:FILTer		1991
[:LPASs]		1991
[:STATe]	<Boolean>	1991
:FREQuency	<numeric_value>	1991
:HPASs		1991
[:STATe]	<Boolean>	1991
:FREQuency	<numeric_value>	1991
:DEMPhasis		1991
[:STATe]	<Boolean>	1991
:TCONstant	<numeric_value>	1991
:CCITt		1991
[:STATe]	<Boolean>	1991
:CMESsage		1991
[:STATe]	<Boolean>	1991
:CCIR		1991
[:STATe]	<Boolean>	1991
:CARM		1991
[:STATe]	<Boolean>	1991
:AWEighting		1991
[:STATe]	<Boolean>	1991

18.10.1 [:LPASs]

SENSe:FILTer:LPASs

Controls the sensor low pass filter.

18.10.1.1 [:STATe] <Boolean>

SENSe:FILTer:LPASs:STATe

Turns the low pass filter on and off.

At *RST, this value is device-dependent.

18.10.1.2 :FREQuency <numeric_value>

SENSe:FILTer:LPASs:FREQuency

Determines the cutoff frequency of the low pass filter.

At *RST, this value is device-dependent.

18.10.2 :HPASs

SENSe:FILTer:HPASs

Controls the sensor high pass filter.

18.10.2.1 **[[:STATe] <Boolean>**

SENSe:FILTer:HPASs:STATe

Turns the high pass filter on and off.

At *RST, this value is device-dependent.

18.10.2.2 **:FREQuency <numeric_value>**

SENSe:FILTer:HPASs:FREQuency

Determines the cutoff frequency of the high pass filter.

At *RST, this value is device-dependent.

18.10.3 **:DEMPhasis**

SENSe:FILTer:DEMPhasis

Controls the sensor FM De-EMphasis filter. De-emphasis filters compensate for pre-emphasis on FM signals.

18.10.3.1 **[[:STATe] <Boolean>**

SENSe:FILTer:DEMPhasis:STATe

Turns the FM de-emphasis filter on and off.

At *RST, this value is device-dependent.

18.10.3.2 **:TCONstant <numeric_value>**

SENSe:FILTer:DEMPhasis:TCONstant

Determines the Time CONstant of the FM de-emphasis filter. The default units are in seconds.

At *RST, this value is device-dependent.

18.10.4 **:CCITt**

SENSe:FILTer:CCITt

Controls the CCITT sensor filter. This standard filter is described in "CCITT Recommendation P53". It is a voice bandwidth filter used in radio or telephone applications.

18.10.4.1 **[[:STATe] <Boolean>**

SENSe:FILTer:CCITt:STATe

Turns the CCITT filter on and off.

At *RST, this value is device-dependent.

18.10.5 **:CMESsage**

SENSe:FILTer:CMESsage

Controls the C-Message sensor filter. This Bell Telephone standard filter is described in "Per BTSM 41004". It is a voice bandwidth filter used in radio or telephone applications.

18.10.5.1 **[[:STATe] <Boolean>**

SENSe:FILTer:CMESsage:STATe

Turns the C-Message filter on and off.

At *RST, this value is device-dependent.

18.10.6 **:CCIR**

SENSe:FILTer:CCIR

Controls the CCIR weighting sensor filter. This standard filter is described in “CCIR Recommendation 468-2”. It is a music bandwidth filter used in applications such as broadcasting and FM receivers.

18.10.6.1 **[[:STATe]] <Boolean>**

SENSe:FILTer:CCIR:STATe

Turns the CCIR weighting filter on and off.

At *RST, this value is device-dependent.

18.10.7 **:CARM**

SENSe:FILTer:CARM

Controls the CCIR/ARM weighting sensor filter. This is a standard music bandwidth filter described in “CCIR Recommendation 468-2” and “Dolby Labs Bulletin No. 19/4”.

18.10.7.1 **[[:STATe]] <Boolean>**

SENSe:FILTer:CARM:STATe

Turns the CCIR/ARM weighting filter on and off.

At *RST, this value is device-dependent.

18.10.8 **:AWEighting**

SENSe:FILTer:AWEighting

Controls the “A” weighting sensor filter. This is a human hearing compensation filter described in “IEC Recommendation 179” and “ANSI S1.4”.

18.10.8.1 **[[:STATe]] <Boolean>**

SENSe:FILTer:AWEighting:STATe

Turns the “A” weighting filter on and off.

At *RST, this value is device-dependent.

18.11 FM Subsystem

SENSe:FM

The Frequency Modulation subsystem is used to set up the modulation controls and parameters associated with sensing and demodulating frequency modulated signals.

KEYWORD	PARAMETER FORM	NOTES
:FM		1991
[:DEViation]		1991
:RANGe		1991
:AUTO	<Boolean> ONCE	1991
[:UPPer]	<numeric_value>	1991
:LOWer	<numeric_value>	1991

18.11.1 [:DEViation]

SENSe:FM:DEViation

This node is used to collect together the deviation parameters.

18.11.1.1 :RANGe

SENSe:FM:DEViation:RANGe

Sets the range for the sensor function.

18.11.1.1.1 :AUTO <Boolean>| ONCE

SENSe:FM:DEViation:RANGe:AUTO

Sets the RANGe to the value determined to give the most dynamic range without overloading. The actual algorithm is device-dependent. Selecting AUTO ONCE will have the effect of setting AUTO to ON and then OFF.

At *RST, this value is set to ON.

18.11.1.1.2 [:UPPer] <numeric_value>

SENSe:FM:DEViation:RANGe:UPPer

Specifies the maximum signal level expected for the sensor input.

At *RST, this value is device-dependent.

18.11.1.1.3 :LOWer <numeric_value>

SENSe:FM:DEViation:RANGe:LOWer

Specifies the smallest signal level expected for the sensor input. This allows setting for the best dynamic range. This function is coupled to UPPer.

At *RST, this value is device-dependent.

18.12 FREQuency Subsystem

SENSe:FREQuency

The FREQuency subsystem controls the frequency characteristics of the instrument. Two types of controls are included in this subsystem. The first is frequency sensor controls, such as RANGE. The second is frequency sweep controls, such as START, or tuning controls such as CW.

Unless otherwise noted, the default unit for all <numeric_values> in this subsystem are in Hz.

KEYWORD	PARAMETER FORM	NOTES
:FREQuency	<numeric_value>	
:APERture	<numeric_value>	
:CENTer	<numeric_value>	
[:CW]:FIXed]	<numeric_value>	
:AFC	<Boolean> ONCE	1994
:AUTO	<Boolean> ONCE	
:MANual	<numeric_value>	
:MODE	CW FIXed SWEep LIST SOURce	
:MULTiplier	<numeric_value>	
:OFFSet	<numeric_value>	
:RANGe		
[:UPPer]	<numeric_value>	
:LOWer	<numeric_value>	
:AUTO	<Boolean> ONCE	
:RESolution	<numeric_value>	
:AUTO	<Boolean> ONCE	
:SPAN	<numeric_value>	
:HOLD	<Boolean>	
:LINK	CENTer START STOP	
:FULL		[no query]
:START	<numeric_value>	
:STOP	<numeric_value>	

18.12.1 :APERture <numeric_value>

SENSe:FREQuency:APERture

Specifies the acquisition/sampling/gate time for a single measurement point. The parameter has a unit of seconds.

At *RST, this value is device-dependent.

18.12.2 :CENTer <numeric_value>

SENSe:FREQuency:CENTer

Set the center frequency of the sweep or measurement.

At *RST, this function will be set to (MINimum+MAX)/2.

18.12.3 [:CWI:FIXed] <numeric_value>

SENSe:FREQuency:CW

The Continuous Wave or FIXed node is used to select a frequency of a non-swept signal. This node is optional. An optional node, such as [:CWI:FIXed], implies that the device shall accept and process commands with or without the optional node and have the same result. That is, the device is required to accept the optional node, if sent, without error. A device must accept both :CW and :FIXed if it implements this node.

At *RST, this value is device-dependent.

18.12.3.1 :AFC <Boolean>IONCE

SENSe:FREQuency:CW:AFC

If AFC is ON the sensor is coupled to the frequency of the signal. The default frequency is determined by SENSe:FREQuency[:CW].

The actual value of FREQuency[:CW] is adjusted to match the frequency of the signal. The query FREQuency[:CW]? returns the adjusted value, not the value sent with the last FREQuency[:CW] command.

At *RST, this value is set to OFF.

18.12.3.2 :AUTO <Boolean>IONCE

SENSe:FREQuency:CW:AUTO

Couples the CW frequency to center frequency. Explicitly setting FREQuency:CW will set AUTO OFF.

At *RST, this value is set to ON.

18.12.4 :MANual <numeric_value>

SENSe:FREQuency:MANual

Same as CW except limited by START and STOP. This command only affects the frequency if SENSe:SWEep:MODE MANual and SENSe:FREQuency:MODE SWEep are selected. If the sweep limits are changed such that manual frequency would be outside the sweep limits, then the manual frequency will be set to the nearest end point of the sweep.

At *RST, this value is device-dependent.

18.12.5 :MODE CWIFIXedISWEepLISTISOURce

SENSe:FREQuency:MODE

Determines which set of commands control the frequency. If CW or FIXed is selected, the frequency is determined by FREQuency:CW. CW and FIXed are aliases. Both must be implemented if either is implemented. If SWEep is selected, the sensor is in the swept mode and frequency is determined by FREQuency:START, STOP, CENT, SPAN, and MAN. If LIST is selected, the frequencies are determined by LIST:FREQuency.

If SOURce is selected, the sensor is coupled to the source. All frequency controls are determined by the SOURCe:FREQuency block, except for OFFSet and MULTiplier. These two commands remain active under SENSe to allow for offset and harmonic tracking.

At *RST, this value is set to CW.

18.12.6 :MULTiplier <numeric_value>

SENSe:FREQuency:MULTiplier

Sets a reference multiplier for all other frequency settings in the instrument. All entered and displayed frequencies are affected by this command. The coupling equation is:

$$\text{Entered/Displayed frequency} = (\text{Hardware frequency} * \text{multiplier}) + \text{offset}.$$

This command affects the values of all functions which affect sensing of the signal. This includes such things as CW frequency and the sweep frequency controls. It does not affect things like input filter bandwidths. It does affect the settings of relative frequency parameters such as delta markers or FM deviation. Also note that this command does not affect the hardware settings of the instrument, but only the entered and displayed frequencies.

At *RST, this value is set to 1.

18.12.7 :OFFSet <numeric_value>

SENSe:FREQuency:OFFSet

Sets a reference frequency for all other absolute frequency settings in the instrument. This command affects the values of all functions which affect sensing of the signal. This includes such things as CW frequency and the sweep frequency controls. It does not affect things like input filter bandwidths. It also does not affect the settings of relative frequency parameters such as delta markers or SPAN. Also note that this command does not affect the hardware settings of the instrument, but only the entered and displayed frequencies. The coupling equation is:

$$\text{Entered/Displayed frequency} = (\text{Hardware frequency} * \text{multiplier}) + \text{offset}.$$

At *RST, this value is set to 0.

18.12.8 :RANGe

SENSe:FREQuency:RANGe

Sets the frequency range for the sensor function.

18.12.8.1 [:UPPer] <numeric_value>

SENSe:FREQuency:RANGe:UPPer

Specifies the maximum value expected for the sensor input.

At *RST, this value is device-dependent.

18.12.8.2 :LOWer <numeric_value>

SENSe:FREQuency:RANGe:LOWer

Specifies the lowest value expected for sensor input. This function is coupled to UPPer in a device-dependent manner.

At *RST, this value is device-dependent.

18.12.8.3 :AUTO <Boolean>IONCE

SENSe:FREQuency:RANGe:AUTO

Sets the RANGe to the value determined by the instrument.

At *RST, AUTO is set to ON.

18.12.9 :RESolution <numeric_value>

SENSe:FREQuency:RESolution

Specifies the absolute resolution of the frequency measurement. For example, if a measurement of 10 MHz is made with a resolution of .01, the measurement is returned with a resolution of .01 Hz (that is, 10000000.00Hz)

At *RST, this value is device-dependent.

18.12.9.1 :AUTO <Boolean>|ONCE

SENSe:FREQuency:RESolution:AUTO

Allows the system to determine the best resolution for the other measurement conditions.

Explicitly selecting a value for FREQuency:RESolution will set AUTO OFF.

At *RST, this value is set to ON.

18.12.10 :SPAN <numeric_value>

SENSe:FREQuency:SPAN

Sets the frequency span.

At *RST, this function will be set to MAXimum.

18.12.10.1 :HOLD <Boolean>

SENSe:FREQuency:SPAN:HOLD

Provides a mechanism to prevent the SPAN from being changed implicitly by the defined coupling between START, STOP, CENTER and SPAN. When HOLD is set to ON, SPAN shall only be changed by explicitly sending the SPAN command.

At *RST, this value is set to OFF.

18.12.10.2 :LINK CENTER|START|STOP

SENSe:FREQuency:SPAN:LINK

Allows the default couplings for SPAN to be overridden. LINK selects the parameter, either CENTER, START or STOP, that shall not be changed when SPAN's value is changed. For example, if LINK is set to START, then changing SPAN shall cause CENTER and STOP, not START, to change.

At *RST, this value is set to CENTER, to be compatible with the default definition for the couplings.

18.12.10.3 :FULL

SENSe:FREQuency:SPAN:FULL

When this command is received, start frequency is set to its minimum value, and stop frequency is set to its maximum value. Center frequency and span are set to their coupled values.

This command is an event. Therefore it has no associated query, or meaning at *RST.

18.12.11 **:START <numeric_value>**

SENSe:FREQuency:START

Sets the starting frequency for a sweep or measurement. It also sets one limit of a manual frequency adjustment. If the instrument does not support negative frequency span, having the start frequency greater than the stop frequency will cause an error to be generated.

At *RST, the parameter will be set to MINimum.

18.12.12 **:STOP <numeric_value>**

SENSe:FREQuency:STOP

Sets the stop frequency of a sweep or measurement. It also sets the other limit of a manual sweep adjustment. If the instrument does not support negative frequency span, having the stop frequency less than the start frequency will cause an error to be generated.

At *RST, this parameter will be set to MAXimum.

18.13 FUNCTION & DATA Subsystem

SENSe:FUNCTION & DATA

The FUNCTION subsystem selects the <sensor_function>(s) to be sensed by the instrument. These functions differ from the signal-oriented MEASure instructions in that a <sensor_function> is something that the sensor can be configured to directly detect.

The DATA subsystem is used to query data from the SENSe block.

KEYWORD	PARAMETER FORM	NOTES	
:DATA?	[<data_handle>]	[query only]	1991
:DATA			1995
:PREAmble?	[<data_handle>]	[query only]	1994
:FUNCTION			
:CONCurrent	<Boolean>		1991
:OFF	<sensor_function>{,<sensor_function>}		1991
:ALL		[no query]	1991
:COUNT?		[query only]	1991
[:ON]	<sensor_function>{,<sensor_function>}		1991
:ALL		[no query]	1991
:COUNT?		[query only]	1991
:STATe?	<sensor_function>	[query only]	1991

18.13.1 DATA? [<data_handle>]

SENSe:DATA?

Provides access to the result(s) of the SENSe block. This query retrieves the current SENSe result(s).

The <data_handle> parameter to this query is optional. Instruments that implement this query are required to accept the the optional parameter <data_handle> if sent. If a sensor function is specified, and that function is currently being sensed the result is returned. If the function is not being sensed an error -221 (Settings Conflict) shall be generated.

If a <data_handle> is not specified, then the result(s) of all functions being sensed are returned. If CONCurrent is OFF there will be only one FUNCTION result. If CONCurrent is ON there will be one or more results, depending on how many functions are currently on. The order of the function results returned by the DATA? query is device-dependent, although the order of these results shall be the same as the order of the functions returned by the FUNCTION[:ON]? query. If no functions are currently being sensed an error -221 (Settings Conflict) shall be generated. The data returned by the DATA? query is affected by the FORMat subsystem.

This command is query-only and therefore has no *RST condition.

18.13.1.1 SENSe <data_handle>s

If the SENSOR block only emits one data flow, then the <data_handle> is just the string "SENSe<n>", where <n> is a numeric suffix. However, for the case where the SENSe block outputs more than one result, each data result needs to be assigned a unique <data_handle>, symbolic of the data flow. Since the SENSe:FUNCTION command is already defined to take a

well-defined string as a parameter, these <sensor_function> strings are used as part of the <data_handle>. If more than one SENSE block in the instrument outputs the same <sensor_function> the SENSE block must also be included in the data handle. The syntax for the SENSE block <data_handle> is:

<data_handle> := “[SENSe:]<sensor_function>” | “SENSe[:<sensor_function>]”

18.13.1.2 :PREAmble? [<data_handle>]

SENSe:DATA:PREAmble?

PREAmble? returns the preamble information supporting the DATA(CURVe(VALues)) but omits the following data:

- DIF blocks of waveform, measurement, and delta.
- DIF curve block keywords VALues and CSUM and their parameters.

The DIF difid block must contain the keyword SCOPE with the parameter of PREAmble.

While the PREAmble? query is designed to efficiently transmit data relating directly to the curve, the various notes and identification parameters may be sent if absolutely required.

The optional <data_handle> parameter functions the same as it does in SENSE:DATA?.

18.13.2 :FUNCTION

SENSe:FUNCTION

The FUNCTION subsystem selects the <sensor_function>(s) to be sensed by the instrument. For compatibility, instruments that support CONCurrent ON are required to support CONCurrent OFF mode. This is so some level of compatibility can be maintained with non-concurrent instruments which expect data results for only one FUNCTION to be returned by the SENSE:DATA? query.

18.13.2.1 :CONCurrent <Boolean>

SENSe:FUNCTION:CONCurrent

The CONCurrent command indicates whether the SENSor block should be configured to SENSE one function at a time or SENSE more than one function at a time (concurrently).

If CONCurrent is OFF, the FUNCTION[:ON] command acts as a one-of-n switch selecting the indicated function to be the only sensed function. If CONCurrent is ON, the function(s) specified as parameter(s) to the FUNCTION[:ON] command are turned on, while the state of other functions are not affected.

When CONCurrent is turned OFF, the instrument shall select one function to be turned on, and turn off any other functions that were on. The function which is selected to be on shall always be the same one, even if that function is not currently ON.

When CONCurrent is turned ON, the states of individual functions shall not be changed.

At *RST the value of CONCurrent is device-dependent. However if an instrument implements CONCurrent ON, then it must implement CONCurrent OFF also for purposes of instrument compatibility.

18.13.2.2 :OFF<sensor_function>{,<sensor_function>}

SENSe:FUNCTION:OFF

The FUNCTION:OFF command selects the <sensor_function>s to be turned off. This command only needs to be implemented by instruments which allow users to turn OFF individual functions while leaving others on. The state of any other functions are not affected by turning the specified functions OFF.

The query response of FUNCTION:OFF? shall return a comma separated list of functions which are currently OFF, each of which are <STRING RESPONSE DATA>. If no functions are OFF, a null string is returned. The query shall return the short form mnemonics and shall omit any default nodes in the <sensor_function>.

At *RST, this value is device-dependent.

18.13.2.2.1 :ALL

SENSe:FUNCTION:OFF:ALL

This command turns OFF all of the <sensor_function>s which the instrument can concurrently sense. There is no query for this command.

18.13.2.2.2 :COUNT?

SENSe:FUNCTION:OFF:COUNT?

This query returns the number of <sensor_function>s which are OFF. This command must be implemented if CONCurrent ON and FUNCTION:OFF are implemented.

This command is query-only and therefore has no *RST condition.

18.13.2.3 [:ON]<sensor_function>{,<sensor_function>}

SENSe:FUNCTION:ON

The FUNCTION[:ON] command selects the <sensor_function>(s) to be SENSEd by the instrument. This command only needs to be implemented by instruments which allow users to select what functions will be SENSEd. The <sensor_function> is specified as a quoted string. For example:

```
FUNCTION "VOLTage:AC"
```

If CONCurrent is OFF, a single <sensor_function> is sent as the parameter. This selects this function to be sensed. If more than one function is sent an error -108 (Parameter not allowed) shall be generated.

If CONCurrent is ON, a comma separated list of <sensor_function> may be sent as parameters. These functions are turned on, while the state of other functions are not affected.

The query response of FUNCTION[:ON]? shall return a comma separated list of functions which are on, each of which are <STRING RESPONSE DATA>. If no functions are ON, a null string is returned. The query shall return the short form mnemonics and shall omit any default nodes in the <sensor_function>.

For example FUNCTION[:ON]? returns the quoted string "VOLT:AC".

At *RST, this value is device-dependent.

18.13.2.3.1 :ALL

SENSe:FUNCTION:ON:ALL

This command turns ON all of the <sensor_function>s which the instrument can concurrently sense. There is no query for this command.

18.13.2.3.2 :COUNT?

SENSe:FUNCTION:ON:COUNT?

This query returns the number of <sensor_function>s which are ON. This command must be implemented if CONCurrent ON and FUNCTION[:ON] are implemented.

This command is query-only and therefore has no *RST condition.

18.13.2.4 :STATE? <sensor_function>

SENSe:FUNCTION:STATE?

Returns a Boolean indicating whether the specified <sensor_function> is currently ON or OFF. This command is query only.

At *RST, the state of each <sensor_function> is device- dependent.

18.13.2.5 <sensor_function>

SENSe:FUNCTION

The FUNCTION subsystem uses the <sensor_function> to select the function on which the FUNCTION commands operate on, <sensor_function> has the following syntax:

*<sensor_function> ::= "<presentation_layer><function_name>
[<input_block>{,<input_block>}] [ON <subnode>{,<subnode>}]"*

The <sensor_function> is STRING PROGRAM DATA.

The names inside angle brackets (<) and (>) are called productions and the commas (,) and 'ON' are literals. White space is defined in section 7.4.1.2 of IEEE 488.2 as one or more white space characters.

For a <sensor_function> sent to an instrument, white space is not allowed inside any production or literal. White space is not allowed between the <presentation_layer> and <function_name> productions. At least one white space character is required before and after the literal ON. At least one white space character is required before the first <input_block>. White space is permitted, but not required, everywhere else in the <sensor_function>. The FUNCTION commands shall accept the long and short forms of <sensor_function> mnemonics. Instruments shall accept both upper and lowercase characters without distinguishing between cases.

When a <sensor_function> is returned in response to a query, it shall contain no white space except for a single space before the first <input_block> and single spaces around the literal ON if either is present. The mnemonics in the query response shall use short form with default nodes omitted. All alpha characters in the response shall be in uppercase.

<input_block> elements are omitted in the query response only if all have been set to their default values. If any <input_block> element has been set to anything other than its default value, then all <input_block> elements are returned in the query response.

The <function_name> is a well-defined hierarchical sensor function indicating the type of function the sensor will be configured to directly detect.

The optional <input_block> parameters are numeric values indicating which INPut blocks shall be “fed” into the sensor function. The numeric value corresponds to the same-numbered INPut block. Specifying an input block in a sensor function makes an implicit internal routing connection such that the output of the designated INPut block will flow into the sensor. If no <input_block> parameters are specified the instrument defined default data flow is used.

For example, SENSE:FUNCTION “VOLTage:AC 2” would configure the sensor to sense the AC Voltage of the data flowing out of INPut2. The SENSE:FUNCTION? query would then return the quoted string “VOLT:AC 2”.

The optional <subnode> parameter is used when the SENSE block contains user configurable selections of subnodes to be used in computation of the SENSE FUNCTION. For example if a single SENSE block contained two SWEep subblocks and the sensor could use either SWEep block to measure the AC Voltage of INPut1. Currently SWEep and AVERage are the only subnodes which are allowed after the ON keyword.

18.13.2.6 <presentation_layer>

1991

PRESENTATION LAYER

[XNONE:]	1991
XTime:	1991
XFrequency:	1991
XPower:	1991
XVOLTage:	1991
XCURrent:	1991

18.13.2.6.1 [XNONE:]

SENSe:FUNCTION "XNONE:<function>"

This specifies that the X-axis has no specific units or number of points. No knowledge can be associated with the X-axis. XNONE is only scalar.

18.13.2.6.2 XTime:

SENSe:FUNCTION "XTime:<function>"

This specifies that the sensor function results are acquired with respect to time. The X-axis is in units of time.

18.13.2.6.3 XFrequency:

SENSe:FUNCTION "XFrequency:<function>"

This specifies that the sensor function results are acquired with respect to frequency. The X-axis is in units of frequency.

18.13.2.6.4 XPower:

SENSe:FUNCTION "XPower:<function>"

This specifies that the sensor function results are acquired with respect to power. The X-axis is in units of power.

18.13.2.6.5 XVOLTage:

SENSe:FUNCTION "XVOLTage:<function>"

This specifies that the sensor function results are acquired with respect to voltage. The X-axis is in units of voltage.

18.13.2.6.6 XCURrent:

SENSe:FUNCTION "XCURrent:<function>"

This specifies that the sensor function results are acquired with respect to current. The X-axis is in units of current.

18.13.2.7 **<function_name>**

SENSe:FUNCTion

A SENSe:FUNCTion <function_name> is a well-defined hierarchical sensor function indicating the type of function the sensor will be configured to directly detect. The <function_name> comprises the <function> and is optionally followed by a :<function_suffix>.

<function_name> := <function>[:<function_suffix>]

Where: <function_suffix> is of type <CHARACTER PROGRAM DATA> with the following values defined:

<function_suffix> := RATio | SUM

When the RATio suffix is included, then a ratio between two input blocks shall be selected.

When the SUM suffix is included, then the sum of two or more input blocks shall be selected.

Excluding the <function_suffix> shall cause the sensor function to be selected.

For example:

:SENSe:FUNCTion "FREQuency"

Selects frequency sensing.

:SENSe:FUNCTion "FREQuency:RATio"

Selects a ratio of two frequencies as a function. If the user specifies input blocks after the sensor function, then the first input block specified is the numerator and the second input block is the denominator.

:SENSe:FUNCTion "VOLTage:DC:SUM 1,2"

Selects the sum of DC voltages on inputs 1 and 2 as a function.

18.13.2.8 <function>

The following is the sensor <function> tree and descriptions:

FUNCTION	DESCRIPTION	
ACCeleration	Acceleration	1999
AM		1991
[:DEPTh]	AM Depth	1991
:DISToRtion	AM Distortion	1991
:FREQuency	AM Frequency	1991
:SNDRatio	AM Signal, Noise & Distortion Ratio (SINAD)	1991
:SNR	AM Signal to Noise Ratio	1991
:THD	AM Total Harmonic Distortion	1991
CONCentration	Non-time-aligned concentration	1999
:RAW	Raw concentration value in device dependent units	1999
:SDEViation	Standard Deviation corresponding to data point CONCentration	1999
:TALign	Time-aligned analyzer for analyzer number (1,2,...n) in device-dependent units	1999
CONDition	True/False condition of a signal	1993
CURRent		
[:DC]	DC Current	
:AC	AC Current	
DISTance	Distance	1999
FERRor	Force Error	1999
FM		1991
[:DEViation]	FM Deviation	1991
:DISToRtion	FM Distortion	1991
:FREQuency	FM Frequency	1991
:SNDRatio	FM Signal, Noise & Distortion Ratio (SINAD)	1991
:SNR	FM Signal to Noise Ratio	1991
:THD	FM Total Harmonic Distortion	1991
FORCe	Force	1999
FREQuency	Direct sensing of frequency	
FRESistance	Four-wire resistance	1991
PERiod	Inverse of frequency	
PHASe	Direct sensing of phase	1992
PM		1991
[:DEViation]	PM Deviation	1991
:DISToRtion	PM Distortion	1991
:FREQuency	PM Frequency	1991
:SNDRatio	PM Signal, Noise & Distortion Ratio (SINAD)	1991
:SNR	PM Signal to Noise Ratio	1991
:THD	PM Total Harmonic Distortion	1991
POWer		
:AC	AC Power	

1999 SCPI Command Reference

FUNCTION	DESCRIPTION	
:ACHannel	Adjacent Channel Power	1991
:LOWer	Lower Adjacent Channel Power	1991
[:UPPer]	Upper Adjacent Channel Power	1991
:COHerence		1992
:CROSSs	Cross Product	1992
[:DC]	DC Power	
:DISTortion	Distortion	1991
:PSDensity	Power Spectral Density	1992
:S1I S2I S2I S12	Scattering Parameter Function	1992
:SNDRatio	Signal, Noise & Distortion Ratio (SINAD)	1991
:SNR	Signal to Noise Ratio	1991
:THD	Total Harmonic Distortion	1991
PULM	Pulse Modulation	1994
RESistance	Two wire resistance	
SPEed	Speed	1999
:FRONt	Front Roll Speed	1999
[:REAR]	Rear Roll Speed	1999
SSB	Single Sideband	1994
TEMPerature	Temperature	1999
TIMer	Value of the timer (see :SYSTem:TIME:TIMer)	1999
TINTerval	Time Interval	1991
TOTALize	Totalize Events	1991
TPLoss	Target Parasitic Loss	1999
VOLTage		
:AC	AC Voltage	
:CDFunction	Cumulative Density Function	1992
[:DC]	DC Voltage	
:HISTogram	Histogram	1992
:PDFunction	Probability Density Function	1992

18

18.13.2.8.1 ACCeleration

SENSe:FUNction "ACCeleration"
Acceleration.

18.13.2.8.2 AM

SENSe:FUNction "AM"
Amplitude modulation.

18.13.2.8.2.1 [:DEPTH]

SENSe:FUNction "AM:DEPTH"
AM depth.

18.13.2.8.2.2 :DISTortion

SENSe:FUNction "AM:DISTortion"
DISTortion is the ratio of noise plus distortion to the signal plus noise plus distortion.

18.13.2.8.2.3 :FREQuency

SENSe:FUNction "AM:FREQuency"

AM frequency.

18.13.2.8.2.4 :SNDRatio

SENSe:FUNction "AM:SNDRatio"

Signal, Noise and Distortion Ratio (SINAD or SNDRatio) is the ratio of the unfiltered signal to the signal with its fundamental removed (ratio of signal plus noise plus distortion to the noise plus distortion).

$$\text{Signal to Noise Ratio (SNR)} = (\text{signal} + \text{noise}) / \text{noise}.$$

18.13.2.8.2.5 :SNR

SENSe:FUNction "AM:SNR"

AM signal to noise ratio.

18.13.2.8.2.6 :THD

SENSe:FUNction "AM:THD"

Total Harmonic Distortion (THD) is the ratio of the signal with its fundamental removed to the unfiltered signal.

18.13.2.8.3 CONCentration

SENSe:FUNction "CONCentration"

Non-time-aligned concentration.

18.13.2.8.3.1 :RAW

SENSe:FUNction "CONCentration:RAW"

Raw concentration value in device dependent units.

18.13.2.8.3.2 :SDEviation

SENSe:FUNction "CONCentration:SDEviation"

Standard Deviation corresponding to data point CONCentration.

18.13.2.8.3.3 :TALign

SENSe:FUNction "CONCentration:TALign"

Time-aligned analyzer for analyzer number (1,2,...n) in device-dependent units.

18.13.2.8.4 CONdition

SENSe:FUNction "CONdition"

True/False condition of a signal.

18.13.2.8.5 CURRent

SENSe:FUNction "CURRent"

18.13.2.8.5.1 [:DC]

SENSe:FUNction "CURRent:DC"

DC current.

18.13.2.8.5.2 :AC

SENSe:FUNction "CURRent:AC"

AC current.

18.13.2.8.6 **DISTance**

SENSe:FUNction "DISTance"

Distance. For chassis dynamometers this is the distance accumulated by the dynamometer.

18.13.2.8.7 **FM**

SENSe:FUNction "FM"

Frequency modulation.

18.13.2.8.7.1 **:[DEViation]**

SENSe:FUNction "FM:DEViation"

FM Deviation.

18.13.2.8.7.2 **:DISTortion**

SENSe:FUNction "FM:DISTortion"

DISTortion is the ratio of noise plus distortion to the signal plus noise plus distortion.

18.13.2.8.7.3 **:FREQuency**

SENSe:FUNction "FM:FREQuency"

FM frequency.

18.13.2.8.7.4 **:SNDRatio**

SENSe:FUNction "FM:SNDRatio"

Signal, Noise and Distortion Ratio (SINAD or SNDRatio) is the ratio of the unfiltered signal to the signal with its fundamental removed (ratio of signal plus noise plus distortion to the noise plus distortion).

$$\text{Signal to Noise Ratio (SNR)} = (\text{signal} + \text{noise}) / \text{noise}.$$

18.13.2.8.7.5 **:SNR**

SENSe:FUNction "FM:SNR"

FM signal to noise ratio.

18.13.2.8.7.6 **:THD**

SENSe:FUNction "FM:THD"

Total Harmonic Distortion (THD) is the ratio of the signal with its fundamental removed to the unfiltered signal.

18.13.2.8.8 **FERRor**

SENSe:FUNction "FERRor"

Force Error. For chassis dynamometers this specifies the error inherent in the measurement of the real-time force.

18.13.2.8.9 **FORCE**

SENSe:FUNction "FORCE"

Force. For chassis dynamometers this is the force at the roll surface.

18.13.2.8.10 **FREQuency**

SENSe:FUNction "FREQuency"

Direct sensing of frequency.

18.13.2.8.11 FRESistance

SENSe:FUNCTion "FRESistance"

Four-wire resistance.

18.13.2.8.12 PERiod

SENSe:FUNCTion "PERiod"

Inverse of frequency.

18.13.2.8.13 PHASe

SENSe:FUNCTion "PHASe"

Direct sensing of phase.

18.13.2.8.14 PM

SENSe:FUNCTion "PM"

Pulse modulation.

18.13.2.8.14.1 [:DEViation]

SENSe:FUNCTion "PM:DEViation"

PM deviation.

18.13.2.8.14.2 :DISTortion

SENSe:FUNCTion "PM:DISTortion"

DISTortion is the ratio of noise plus distortion to the signal plus noise plus distortion.

18.13.2.8.14.3 :FREQuency

SENSe:FUNCTion "PM:FREQuency"

PM Frequency.

18.13.2.8.14.4 :SNDRatio

SENSe:FUNCTion "PM:SNDRatio"

Signal, Noise and Distortion Ratio (SINAD or SNDRatio) is the ratio of the unfiltered signal to the signal with its fundamental removed (ratio of signal plus noise plus distortion to the noise plus distortion).

$$\text{Signal to Noise Ratio (SNR)} = (\text{signal} + \text{noise}) / \text{noise}.$$

18.13.2.8.14.5 :SNR

SENSe:FUNCTion "PM:SNR"

PM signal to noise ratio.

18.13.2.8.14.6 :THD

SENSe:FUNCTion "PM:THD"

PM Total Harmonic Distortion (THD) is the ratio of the signal with its fundamental removed to the unfiltered signal.

18.13.2.8.15 POWer

SENSe:FUNCTion "POWer"

18.13.2.8.15.1 :AC

SENSe:FUNcTION "POWer:AC"

AC Power.

18.13.2.8.15.2 :ACHannel

SENSe:FUNcTION "POWer:ACHannel"

Adjacent Channel Power measurements involve measuring the power at a specified carrier for a given bandwidth and then ratio it to the power detected at the upper (or lower) adjacent channel.

18.13.2.8.15.2.1 :LOWer

SENSe:FUNcTION "POWer:ACHannel:LOWer"

Lower Adjacent Channel Power.

18.13.2.8.15.2.2 [:UPPer]

SENSe:FUNcTION "POWer:ACHannel:UPPer"

Upper Adjacent Channel Power.

18.13.2.8.15.3 :COHerence

SENSe:FUNcTION "POWer:COHerence"

COHerence is the portion of the first <input_block> spectrum related to the second <input_block> spectrum. It is a measure of the statistical validity of a frequency response measurement.

18.13.2.8.15.4 :CROSSs

SENSe:FUNcTION "POWer:CROSSs"

This is the Cross Product of two input_blocks. It is computed by multiplying the complex conjugate of the first <input_block> by the second <input_block>.

18.13.2.8.15.5 [:DC]

SENSe:FUNcTION "POWer:DC"

DC Power.

18.13.2.8.15.6 :DISTortion

SENSe:FUNcTION "POWer:DISTortion"

DISTortion is the ratio of noise plus distortion to the signal plus noise plus distortion.

18.13.2.8.15.7 :PSDensity

SENSe:FUNcTION "POWer:PSDensity"

This is the Power Spectral Density. It is the power spectrum normalized to a 1 Hz bandwidth.

18.13.2.8.15.8 :S11:S12I:S22I:S21

SENSe:FUNcTION "POWer:S11"

These are the Scattering Parameter Functions.

Scattering parameters or S parameters are a tool used in frequency characterization of devices to determine how the device modifies signal flow. An S parameter is a ratio of two energy values and is defined as:

$$S_{xy} = \frac{\text{energy emerging from port } x}{\text{energy entering port } y}$$

with all ports other than port Y terminated in a Z0 load.

By convention, a devices ports are numbered (1,2,...n). An n port device has n^2 S parameters labelled S_{xy} , where $\{x,y\} = \{1,2,...,n\}$.

For more information please refer to “Scattering and Transmission Coefficients” chapter in “Fields and Waves in Communication Electronics”, Ramo, Whinnery and Van Duzer.

18.13.2.8.15.9 :SNDRatio

SENSe:FUNCTION “POWER:SNDRatio”

Signal, Noise and Distortion Ratio (SINAD or SNDRatio) is the ratio of the unfiltered signal to the signal with its fundamental removed (ratio of signal plus noise plus distortion to the noise plus distortion).

$$\text{Signal to Noise Ratio (SNR)} = (\text{signal} + \text{noise}) / \text{noise}.$$

18.13.2.8.15.10 :SNR

SENSe:FUNCTION “POWER:SNR”

Signal to Noise Ratio.

18.13.2.8.15.11 :THD

SENSe:FUNCTION “POWER:THD”

Total Harmonic Distortion (THD) is the ratio of the signal with its fundamental removed to the unfiltered signal.

18.13.2.8.16 PULM

SENSe:FUNCTION “PULM”

Pulse modulation.

18.13.2.8.17 RESistance

SENSe:FUNCTION “RESistance”

Two wire resistance.

18.13.2.8.18 SPEed

SENSe:FUNCTION “SPEed”

Speed.

18.13.2.8.18.1 :FRONT

SENSe:FUNCTION “SPEed:FRONT”

Front Roll Speed. For chassis dynamometers this is the roll speed. On twin roll systems, this is the front roll speed.

18.13.2.8.18.2 [:REAR]

SENSe:FUNCTION “SPEed[:REAR]”

Rear Roll Speed. For chassis dynamometers this is the roll speed. On twin roll systems, this is the rear roll speed.

18.13.2.8.19 **SSB**

SENSe:FUNcTION "SSB"

Single sideband modulation.

18.13.2.8.20 **TEMPerature**

SENSe:FUNcTION "TEMPerature"

Temperature.

18.13.2.8.21 **TIMer**

SENSe:FUNcTION "TIMer"

Value of the timer (see :SYSTem:TIME:TIMer).

18.13.2.8.21.1 **COUNT**

SENSe:FUNcTION "TIMer:COUNT"

Returns the current timer count.

18.13.2.8.22 **TINTerval**

SENSe:FUNcTION "TINTerval"

Time Interval.

18.13.2.8.23 **TOTALize**

SENSe:FUNcTION "TOTALize"

Totalize events.

18.13.2.8.24 **TPLoss**

SENSe:FUNcTION "TPLoss"

Target Parasitic Loss. For chassis dynamometers this is the value of frictional loss at the present speed based on the active parasitic loss curve.

18.13.2.8.25 **VOLTage**

SENSe:FUNcTION "VOLTage"

18.13.2.8.25.1 **:AC**

SENSe:FUNcTION "VOLTage:AC"

AC Voltage.

18.13.2.8.25.2 **:CDFunction**

SENSe:FUNcTION "VOLTage:CDFunction"

The Cumulative Density Function is the integral of a normalized histogram.

18.13.2.8.25.3 **[:DC]**

SENSe:FUNcTION "VOLTage:DC"

DC Voltage.

18.13.2.8.25.4 **:HISTogram**

SENSe:FUNcTION "VOLTage:HISTogram"

The histogram measures how the amplitude of the input signal is distributed between it's maximum and minimum values.

18.13.2.8.25.5 :PDFFunction

SENSe:FUNCtion "VOLTage:PDFFunction"

The Probability Density Function is computed by normalizing a histogram and is a statistical measure of the probability that a specific level occurred.

18.14 LIST Subsystem

SENSe:LIST

The list subsystem controls automatic sequencing through associated lists of frequency, and dwell time values. The various values are specified in the LIST:FREQuency, LIST:DWELl commands. Points in each list correspond to the associated point in all other lists.

The LIST subsystem controls a characteristic of the instrument when that subsystem's mode is set to LIST (for example, FREQuency:MODE LIST).

When employing multiple lists, all lists must be of equal length. If they are not of the same length, an error is generated at sequencing time. An exception is made for the case where the shorter list is of length 1. In this case, the particular list is treated as though it were a list of equal length, with all values being the same.

Lists are not affected by *RST.

KEYWORD	PARAMETER FORM	NOTES
:LIST		
:COUNT	<numeric_value>	
:DIRection	UPIDOWN	
:DWELl	<numeric_value>{,<numeric_value>}	
:POINTs?		[query only]
:FREQuency	<numeric_value>{,<numeric_value>}	
:POINTs?		[query only]
:SEQuence	<numeric_value>{,<numeric_value>}	
:AUTO	<Boolean> ONCE	
:POINTs?		[query only]

18.14.1 :COUNT <numeric_value>

SENSe:LIST:COUNT

Controls the number of times the sequence list is scanned when a trigger is received.

18.14.2 :DIRection UPIDOWN

SENSe:LIST:DIRection

Specifies the direction that the sequence list is scanned. If UP is selected, the list is scanned from the first to the last item in the sequence list. If DOWN is selected, the list is scanned from the last item to the first item in the sequence list.

18.14.3 :DWELl <numeric_value>{,<numeric_value>}

SENSe:LIST:DWELl

Specifies the dwell time points of the lists. The units are in seconds.

This specifies the amount of time spent at each point.

18.14.3.1 :POINTs?

SENSe:LIST:DWELl:POINTs?

Returns the number of points currently in the DWELl list.

18.14.4 **:FREQuency<numeric_value>{,<numeric_value>}**

SENSe:LIST:FREQuency

Specifies the frequency points of the list set. The units are Hz.

18.14.4.1 **:POINts?**

SENSe:LIST:FREQuency:POINts?

This query returns the number of points currently in the FREQuency list.

18.14.5 **:SEQuence<numeric_value>{,<numeric_value>}**

SENSe:LIST:SEQuence

Defines a sequence for stepping through the list. Individual points may be specified as many times as desired in a single sequence. The points specified are indexes into the lists. For example, if 3 was selected, the third point in the frequency and dwell lists would be sequenced.

18.14.5.1 **:AUTO<Boolean>IONCE**

SENSe:LIST:SEQuence:AUTO

When on, the sequence list is set to 1 through N, where N is the longest list.

18.14.5.2 **:POINts?**

SENSe:LIST:SEQuence:POINts?

This query returns the number of points currently in the SEQuence list.

18.15 MIXer Subsystem

SENSe:MIXer

This subsystem controls external mixers for tuned receivers. It also provides controls to compensate for known fixed losses in external mixers.

KEYWORD	PARAMETER FORM	NOTES
:MIXer		
:BIAS	<numeric_value>	
:AUTO	<Boolean> ONCE	
:LIMit	<numeric_value>	
:HARMonic	<numeric_value>	
:AUTO	<Boolean> ONCE	
:LOSS	<numeric_value>	
:AUTO	<Boolean>	1991

18.15.1 :BIAS <numeric_value>

SENSe:MIXer:BIAS

Controls the mixer bias.

At *RST, this value is device-dependent.

18.15.1.1 :AUTO <Boolean>|ONCE

SENSe:MIXer:BIAS:AUTO

Allows the mixer bias to be automatically set by other parameters in the system. Explicitly setting a value for BIAS shall cause AUTO to be set to OFF.

At *RST, this value shall be set to ON.

18.15.1.2 :LIMit <numeric_value>

SENSe:MIXer:BIAS:LIMit

Controls the maximum mixer bias level which can be set by the BIAS command or by other parameters when AUTO is ON.

The limits and *RST conditions are instrument-dependent.

18.15.2 :HARMonic <numeric_value>

SENSe:MIXer:HARMonic

Selects which harmonic of the local oscillator will be used for mixing.

At *RST, this value is device-dependent.

18.15.2.1 :AUTO <Boolean>|ONCE

SENSe:MIXer:HARMonic:AUTO

When AUTO is ON, the harmonic is chosen by the system or instrument, and may change during the sweep. When OFF, the harmonic is locked to the selected value. Setting a value for MIXer:HARMonic sets AUTO to OFF.

At *RST, this value is set to ON.

18.15.3 :LOSS <numeric_value>

SENSe:MIXer:LOSS

Compensates for losses external to the instrument. This parameter is added to all readings. The unit for this command is the current relative amplitude unit.

At *RST, this value is set to 0.

18.15.3.1 :AUTO <Boolean>

SENSe:MIXer:LOSS:AUTO

Sets the LOSS to the value appropriate to the connected external mixer.

At *RST, this value is device-dependent.

18.16 PM Subsystem

SENSe:PM

The Phase Modulation subsystem is used to set the modulation controls and parameters associated with sensing phase modulated signals.

KEYWORD	PARAMETER FORM	NOTES
:PM		1991
[:DEViation]		1991
:RANGe		1991
:AUTO	<Boolean> ONCE	1991
[:UPPer]	<numeric_value>	1991
:LOWer	<numeric_value>	1991

18.16.1 [:DEViation]

SENSe:PM:DEViation

This node is used to collect together the deviation parameters.

18.16.1.1 :RANGe

SENSe:PM:DEViation:RANGe

Sets the range for the sensor function.

18.16.1.1.1 :AUTO <Boolean> | ONCE

SENSe:PM:DEViation:RANGe:AUTO

Sets the RANGe to the value determined to give the most dynamic range without overloading. The actual algorithm is device-dependent. Selecting AUTO ONCE will have the effect of setting AUTO to ON and then OFF.

At *RST, this value is set to ON.

18.16.1.1.2 [:UPPer] <numeric_value>

SENSe:PM:DEViation:RANGe:UPPer

Specifies the maximum signal level expected for the sensor input.

At *RST, this value is device-dependent.

18.16.1.1.3 :LOWer <numeric_value>

SENSe:PM:DEViation:RANGe:LOWer

Specifies the smallest signal level expected for the sensor input. This allows setting for the best dynamic range. This function is coupled to UPPer.

At *RST, this value is device-dependent.

18.17 Power Subsystem

SENSe:POWer

This subsection controls the power amplitude characteristics of the sensor.

KEYWORD	PARAMETER FORM	NOTES
:POWer		
:ACHannel		1991
:SPACing		1991
:LOWer	<numeric_value>	1991
:AUTO	<Boolean>	1991
[[:UPPer]	<numeric_value>	1991
:AC[:DC]		
:APERture	<numeric_value>	
:NPLCycles	<numeric_value>	1991
:ATTenuation	<numeric_value>	
:AUTO	<Boolean>	
:PROTection		
[[:LEVel]	<numeric_value>	
:STATe		
:TRIPped?		[query only]
:CLEar		[no query]
:RANGe		
[[:UPPer]	<numeric_value>	
:LOWer	<numeric_value>	
:AUTO	<Boolean> ONCE	
:DIRection	UP DOWN EITHer	
:LLIMit	<numeric_value>	1996
:ULIMit	<numeric_value>	1996
:OFFSet		1991
:PTPeak		1991
:REFerence	<numeric_value>	
:STATe	<Boolean>	
:RESolution	<numeric_value>	
:AUTO	<Boolean> ONCE	
:DETector	INTernal EXTernal	1991

18.17.1 :ACHannel

SENSe:POWer:ACHannel

Adjacent Channel Power Measurements involve measuring the power at a specified carrier for a given bandwidth and then ratio it to the power detected at the upper (or lower) adjacent channel.

18.17.1.1 :SPACing

SENSe:POWer:ACHannel:SPACing

Controls the channel SPACing which is the distance from the modulated carrier signal to the upper or lower Adjacent Channel.

18.17.1.1.1 :LOWer <numeric_value>

SENSe:POWer:ACHannel:SPACing:LOWer

Controls the channel SPACing to the lower Adjacent Channel.

At *RST, this value is device-dependent.

18.17.1.1.1.1 :AUTO <Boolean>

SENSe:POWer:ACHannel:SPACing:LOWer:AUTO

When AUTO is ON, the value of LOWER is coupled to the value of UPPER.

18.17.1.1.2 [:UPPer] <numeric_value>

SENSe:POWer:ACHannel:SPACing:UPPer

Controls the channel SPACing which is the distance from the modulated carrier signal to the upper Adjacent Channel.

At *RST, this value is device-dependent.

18.17.2 :ACI[:DC]

SENSe:POWer:AC

This subsystem selects the alternating current or the direct current sensor. The default units are watts.

18.17.2.1 :APERture <numeric_value>

SENSe:POWer:AC:APERture

Specifies the acquisition/sampling/gate time for a single measurement point. The parameter has a unit of seconds.

At *RST, this value is device-dependent.

18.17.2.2 :NPLCycles <numeric_value>

SENSe:POWer:AC:NPLCycles

Specifies the acquisition/sampling/gate time for a single measurement point in terms of the number of power line cycles. The parameter is unitless. If this command is implemented, the APERture command must also be implemented.

The value is coupled to APERture by the equation:

$$APERture = NPLCycles / selected\ line\ frequency$$

At *RST, this value is device-dependent.

18.17.2.3 :ATTenuation <numeric_value>

SENSe:POWer:AC:ATTenuation

Sets the ATTenuation level. Note that when setting the level to 10 dB the magnitude of the incoming signal will be decreased by 10 dB. Changing the ATTenuation changes RANGE by the same amount.

Default units are as determined in the UNITS system.

18.17.2.3.1 :**AUTO** <Boolean>

SENSe:POWer:AC:ATTenuation:AUTO

Couples the attenuator to RANGE such that maximum dynamic range of the incoming signal is assured without overloading.

Programming a specified attenuation sets AUTO OFF.

At *RST, AUTO is set to ON.

18.17.2.4 :**PROTection**

SENSe:POWer:AC:PROTection

Controls the protection circuits.

18.17.2.4.1 [:**LEVel**] <numeric_value>

SENSe:POWer:AC:PROTection:LEVel

This command sets the input level at which the input protection circuit will trip.

18.17.2.4.2 :**STATe** <Boolean>

SENSe:POWer:AC:PROTection:STATe

Controls whether the input protection circuit is enabled.

At *RST, this value is set to ON.

18.17.2.4.3 :**TRIPped?**

SENSe:POWer:AC:PROTection:TRIPped?

This query returns a 1 if the protection circuit is tripped and a 0 if it is untripped. TRIPped only has a query form and thus has no associated *RST condition.

18.17.2.4.4 :**CLEAr**

SENSe:POWer:AC:PROTection:CLEAr

Causes the protection circuit to be cleared.

This command is an event and has no associated *RST condition.

18.17.2.5 :**RANGe**

SENSe:POWer:AC:RANGe

Sets the range for the amplitude sensor function. The RANGe endpoints may be specified in terms of the UPPer endpoint and the LOWEr endpoint, alternatively the midpoint of the RANGe (OFFSet) and the span of the RANGe (PTPeak) may be used. A non-symmetrical range around zero may cause some instruments to introduce an actual offset to the signal entering the SENSe block.

18.17.2.5.1 [:**UPPer**] <numeric_value>

SENSe:POWer:AC:RANGe:UPPer

Specifies the most positive signal level expected for the sensor input.

At *RST, this value is device-dependent.

18.17.2.5.2 :LOWer <numeric_value>

SENSe:POWer:AC:RANGe:LOWer

Specifies the most negative signal level expected for the sensor input. This allows setting for the best dynamic range. This function is coupled to UPPER in a device-dependent manner.

At *RST, this value is device-dependent.

18.17.2.5.3 :AUTO <Boolean>IONCE

SENSe:POWer:AC:RANGe:AUTO

Sets the RANGe to the value determined to give the most dynamic range without overloading. The actual algorithm is device-dependent.

At *RST, AUTO is set to ON.

18.17.2.5.3.1 :DIRection UPIDOWNIEITHer

SENSe:POWer:AC:RANGe:AUTO:DIRection

The DIRection command defines the manner in which AUTO works. When UP is selected, the instrument shall only change the range automatically to ranges larger than the current range, as the input signal dictates. When DOWN is selected, the instrument shall only change the range automatically to ranges smaller than the current range, as the input signal dictates. When EITHER is selected the instrument may range in either direction.

At *RST, DIRection shall be set to EITHER.

18.17.2.5.3.2 :LLIMit <numeric_value>

SENSe:POWer:AC:RANGe:AUTO:LLIMit

The Lower LIMit command sets the smallest range to which the instrument will go while autoranging. When AUTO is ON and the present RANGe is outside the new limits, the instrument may change the RANGe to a value inside the limits or wait until the next time the autoranging algorithm is performed.

At *RST, LLIMit shall be set to MINimum.

18.17.2.5.3.3 :ULIMit <numeric_value>

SENSe:POWer:AC:RANGe:AUTO:ULIMit

The Upper LIMit command sets the largest range to which the instrument will go while autoranging. When AUTO is ON and the present RANGe is outside the new limits, the instrument may change the RANGe to a value inside the limits or wait until the next time the autoranging algorithm is performed.

At *RST, ULIMit shall be set to MAXimum.

18.17.2.5.4 :OFFSet <numeric_value>

SENSe:POWer:AC:RANGe:OFFSet

OFFSet determines the midpoint of the range.

At *RST, this value is device-dependent.

18.17.2.5.5 :PTPeak <numeric_value>

SENSe:POWer:AC:RANGe:PTPeak

Peak To Peak specifies the dynamic range required for the sensor.

At *RST, this value is device-dependent.

18.17.2.6 :REFeRence <numeric_value>

SENSe:POWer:AC:REFeRence

Sets a reference amplitude for sensor instruments. This command differs in intent from the reference correction in that this is intended to be a measurement offset where a user wishes to reference a measurement to some known value.

At *RST, this value is device-dependent, but STATE is set to OFF.

18.17.2.6.1 :STATE <Boolean>

SENSe:POWer:AC:REFeRence:STATE

Determines whether amplitude is measured in absolute or relative mode. If STATE is ON, then amplitude is referenced to the value set in REFeRence.

At *RST, this value is set to OFF.

18.17.2.7 :RESolution <numeric_value>

SENSe:POWer:AC:RESolution

Specifies the absolute resolution of the measurement. For example, if a measurement of 10 Volts is specified with a resolution of .01 Volts, the measurement is returned with a resolution of .01 Volts (that is, 10.00).

RANGE is not coupled to RESolution. Setting RANGE shall not cause RESolution to change. Setting RESolution shall not cause RANGE to change.

At *RST, this value is device-dependent.

18.17.2.7.1 :AUTO <Boolean>IONCE

SENSe:POWer:AC:RESolution:AUTO

Allows the system to determine the best resolution for the other measurement conditions.

Explicitly setting a value for :RESolution will turn AUTO OFF.

At *RST, this value is set to ON.

18.17.3 :DETector INTERNAL | EXTERNAL

SENSe:POWer:DETector

The DETector can be specified as internal or external.

At *RST, this value is set to INT.

18.18 RESistanceIFRESistance Subsystem

SENSe:RESistance

This subsection controls the signal characteristics of a resistance sensor. RESistance should be used for a two-wire measurement. FRESistance (four-wire resistance) should be used for a four-wire measurement. The default units are ohms.

KEYWORD	PARAMETER FORM	NOTES
:RESistance		
l:FRESistance		1991
:APERTure	<numeric_value>	
:NPLCycles	<numeric_value>	1991
:OCOMpensated	<Boolean>	1991
:RANGe		
[:UPPer]	<numeric_value>	
:LOWer	<numeric_value>	
:AUTO	<Boolean> ONCE	
:DIRection	UPIDOWN EITHer	
:LLIMit	<numeric_value>	1996
:ULIMit	<numeric_value>	1996
:REFerence	<numeric_value>	
:STATe	<Boolean>	
:RESolution	<numeric_value>	
:AUTO	<Boolean> ONCE	

18.18.1 :APERTure <numeric_value>

SENSe:RESistance:APERTure

Specifies the acquisition/sampling/gate time for a single measurement point. The parameter has a unit of seconds.

At *RST, this value is device-dependent.

18.18.2 :NPLCycles <numeric_value>

SENSe:RESistance:NPLCycles

Specifies the acquisition/sampling/gate time for a single measurement point in terms of the number of power line cycles. The parameter is unitless. If this command is implemented, the APERTure command must also be implemented.

The value is coupled to APERTure by the equation:

$$APERTure = NPLCycles / \text{selected line frequency}$$

At *RST, this value is device-dependent.

18.18.3 :OCOMpensated <Boolean>

SENSe:RESistance:OCOMpensated

Enables or disables the offset compensation when measuring resistance. (For example, offset compensation first measures the voltage on the input terminals before turning on the current used to measure resistance.)

At *RST, OCOMpensated shall be set to OFF.

18.18.4 :RANGe

SENSe:RESistance:RANGe

Sets the range for the resistance sensor.

18.18.4.1 [:UPPer] <numeric_value>

SENSe:RESistance:RANGe:UPPer

Specifies the maximum resistance expected for the sensor input.

At *RST, this value is device-dependent.

18.18.4.2 :LOWer <numeric_value>

SENSe:RESistance:RANGe:LOWer

Specifies the smallest resistance expected for the sensor input. This allows setting for the best dynamic range. This function is coupled to the UPPer command in a device-dependent manner.

At *RST, this value is device-dependent.

18.18.4.3 :AUTO <Boolean>IONCE

SENSe:RESistance:RANGe:AUTO

Sets the RANGe to the value determined to give the most dynamic range without over loading. The actual algorithm is device-dependent.

At *RST, AUTO is set to ON.

18.18.4.3.1 :DIRection UPIDOWNIEITHer

SENSe:RESistance:RANGe:AUTO:DIRection

The DIRection command defines the manner in which AUTO works. When UP is selected, the instrument shall only change the range automatically to ranges larger than the current range, as the input signal dictates. When DOWN is selected, the instrument shall only change the range automatically to ranges smaller than the current range, as the input resistance dictates. When EITHER is selected, the instrument may range in either direction.

AT *RST, DIRection shall be set to EITHER.

18.18.4.3.2 :LLIMit <numeric_value>

SENSe:RESistance:RANGe:AUTO:LLIMit

The Lower LIMit command sets the smallest range to which the instrument will go while autoranging. When AUTO is ON and the present RANGe is outside the new limits, the instrument may change the RANGe to a value inside the limits or wait until the next time the autoranging algorithm is performed.

At *RST, LLIMit shall be set to MINimum.

18.18.4.3.3 :ULIMit <numeric_value>

SENSe:RESistance:RANGe:AUTO:ULIMit

The Upper LIMit command sets the largest range to which the instrument will go while autoranging. When AUTO is ON and the present RANGe is outside the new limits, the instrument may change the RANGe to a value inside the limits or wait until the next time the autoranging algorithm is performed.

At *RST, ULIMit shall be set to MAXimum.

18.18.5 :REFerence <numeric_value>

SENSe:RESistance:REFerence

Sets a reference resistance for sensor instruments. This command differs in intent from the reference correction in that this is intended to be a measurement offset where a user wishes to reference a measurement to some known value.

At *RST, this value is device-dependent, but STATe is set to OFF.

18.18.5.1 :STATe <Boolean>

SENSe:RESistance:REFerence:STATe

Determines whether resistance is measured in absolute or relative mode. If STATe is ON, then resistance is referenced to the value set in REFerence.

At *RST, this value is set to OFF.

18.18.6 :RESolution <numeric_value>

SENSe:RESistance:RESolution

Specifies the absolute resolution of the measurement. For example, if a measurement of 10 Ohms is made with a resolution of .01, the measurement is returned with a resolution of .01 Ohms (that is, 10.00).

At *RST, this value is device-dependent.

18.18.6.1 :AUTO <Boolean>|ONCE

SENSe:RESistance:RESolution:AUTO

Allows the system to determine the best resolution for the other measurement conditions.

Explicitly setting a value for RESolution will turn AUTO OFF.

At *RST, this value is set to ON.

18.19 ROSeillator Subsystem

SENSe:ROSeillator

This subsystem controls the frequency reference oscillator.

KEYWORD	PARAMETER FORM	NOTES
:ROSeillator		
[:INTErnal]		1992
:FREQuency	<numeric_value>	
:EXTErnal		1992
:FREQuency	<numeric_value>	1992
:SOURce	INTErnal EXTErnal NONE	
	ICLK10 CLK100	1991
:AUTO	<Boolean> ONCE	

18.19.1 [:INTErnal]

SENSe:ROSeillator:INTErnal

This subsystem configures the internal reference oscillator(s).

18.19.1.1 :FREQuency <numeric_value>

SENSe:ROSeillator:INTErnal:FREQuency

Specifies the frequency of the internal reference oscillator. The default units are Hz.

At *RST, this value is device-dependent.

18.19.2 :EXTErnal

SENSe:ROSeillator:EXTErnal

This subsystem configures the external reference oscillator(s).

18.19.2.1 :FREQuency <numeric_value>

SENSe:ROSeillator:EXTErnal:FREQuency

Specifies the frequency of the external reference oscillator. The default units are Hz.

At *RST, this value is device-dependent.

18.19.3 :SOURce INTErnal|EXTErnal|NONE|ICLK10|CLK100

SENSe:ROSeillator:SOURce

Controls the selection of the reference oscillator source. This typically refers to a precision, stabilized time base. The parameter values have the following meanings:

- INTErnal — The reference frequency is derived from an internal precision oscillator.
- EXTErnal — The reference frequency is derived from an external signal supplied to the device.
- NONE — The reference frequency is not derived from any precision oscillator.
- CLK10 — The reference frequency is derived from VXIbus signal CLK10.
- CLK100 — The reference frequency is derived from VXIbus signal CLK100.

At *RST, this value is device-dependent.

18.19.3.1 :**AUTO** <Boolean>**!ONCE**

SENSe:ROSCillator:SOURce:AUTO

When AUTO is ON, the system automatically selects which oscillator will be used by the instrument.

Explicitly selecting a reference oscillator turns AUTO OFF.

At *RST, AUTO is set to ON.

18.20 SMOothing Subsystem

SENSe:SMOothing

This subsystem controls the point-to-point smoothing of the signal. Smoothing differs from averaging in that with smoothing, adjacent points of one set of data are averaged, while in averaging the average is taken over several acquisitions (that is, smoothing is a moving average). This subsystem controls smoothing done to compensate for measurement artifacts. Analytical post-acquisition smoothing is controlled with CALCulate:SMOothing.

KEYWORD	PARAMETER FORM	NOTES
SMOothing		
[[:STATe]	<Boolean>	
:APERture	<numeric_value>	
:POINts	<numeric_value>	

18.20.1 [[:STATe] <Boolean>

SENSe:SMOothing:STATe

Determines whether the smoothing algorithm is enabled.

At *RST, this function is set to OFF.

18.20.2 :APERture <numeric_value>

SENSe:SMOothing:APERture

Specifies the size of the smoothing APERTure as a ratio of smoothing window points/trace points. POINts is coupled to APERTure by the equation:

$$POINts = APERTure * SWEep:POINts$$

For example, if an APERTure of .01 were chosen for a 1000-point sweep, then the smoothing window would be 10 points wide.

At *RST, this value is device-dependent.

18.20.3 :POINts <numeric_value>

SENSe:SMOothing:POINts

Controls the number of points to be included in the running average. POINts is coupled to APERTure by the equation:

$$POINts = APERTure * SWEep:POINts$$

At *RST, this value is device-dependent.

18.21 **SSB Subsystem**

SENSe:SSB

The single sideband modulation subsystem is used to set up the modulation controls and parameters associated with sensing and demodulating single sideband modulated signals.

18.21.1 **:TYPE USBLSBIA1**

SENSe:SSB:TYPE

SENSe:SSB:TYPE sets or queries the type of SSB demodulation technique the demodulator uses. USB means the demodulator is configured to upper sideband signals. LSB means the demodulator is configured to lower sideband signals. A1 means the demodulator is configured to A1 signals.

At *RST, this value is device-dependent.

18.22 **STABilize Subsystem**

SENSe:STABilize

This subsystem contains the commands that control the stability parameters for the designated instrument and range.

KEYWORD	PARAMETER FORM	NOTES
:STABilize		1999
:NTOLerance	<numeric_value>	1999
[:STATe]	<Boolean>	1999
:TIME<n>	<numeric_value>	1999

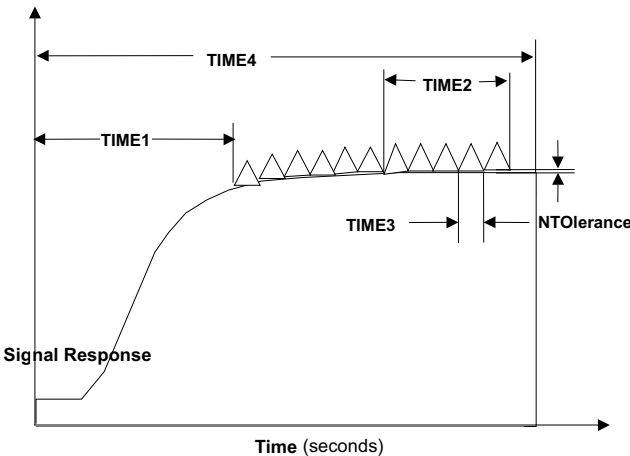


Figure 18.1 Stabilization Use

18.22.1 **:NTOLerance <numeric_value>**

:SENSe:STABilize:NTOLerance

This command specifies or returns the allowable tolerance between averaged readings for a stabilized read, in percentage of full scale (%FS).

At *RST these values are not changed.

18.22.2 **[:STATe] < Boolean >**

:SENSe:STABilize[:STATe]

This command sets or queries the stabilization state for all sensors. If stabilization is enabled, a procedure involving use of the stabilization times TIME1, TIME2, TIME3, and TIME4 and the stabilization tolerance is invoked in response to commands such as :SENSe:DATA?, and :SENSe:CORRection:ZERO:ACQuire. An error in stabilization shall result in error #-365.

At *RST, stabilization is OFF.

18.22.3 :TIME<n> <numeric_value>

:SENSe:STABilize:TIME<n>

This command sets the nth time parameter for the stabilization procedure, for the selected instrument(s).

The numeric suffix of :TIME, which is NOT optional, is defined as follows:

TIME1: Stabilized read initial rise or delay time.

TIME2: Stabilized read period, consisting of a number of TIME3's, over which stabilization is considered.

TIME3: Stabilized read duration of a single averaging period.

TIME4: Stabilized read maximum time of stabilization procedure.

At *RST these values are not changed.

18.23 SWEep Subsystem

SENSe:SWEep

This subsystem controls the timebase and/or sweep generator in instruments that acquire data either directly or indirectly as a function of time.

Examples of instruments that acquire data directly as a function of time (non-swept instruments) are digitizers and logic analyzers. The X axis has units of time. The <sensor_function> presentation layer is :XTIME.

Examples of instruments that acquire data indirectly as a function of time (swept instruments) are spectrum analyzers. In this case, the X axis has units of frequency, but the frequency is swept with relationship to time. The <sensor_function> presentation layer is :XFREquency. The parameter to be swept is specified by setting the :MODE node in the associated subsystem to SWEEP (FREQuency:MODE SWEep).

Some commands in this subsystem are applicable to both swept and non-swept instruments. Others may apply to only one of these classes.

Commands in this subsystem may be coupled to those in implementor-defined trigger sequences such as TRIGger:SAMple. The TRIGger sequences can provide additional flexibility by allowing the user more precise control over the acquisition.

The commands in the :SWEep subsystem specify the timing and spacing of the values returned in :DATA?, FETCh?, and other measurement queries. The commands in the TRIGger subsystem specify when actual measurements are acquired. The precise relationship between the parameters in the two subsystems is device dependent. For example, the :SWEep:TINTerval command specifies the timing period between two successive samples of the SENSe:DATA? response. The TRIGger:SAMple:TIMER command specifies the timing period between two acquisition samples that are taken by the trigger subsystem.

The :SWEep subsystem contains several sets of commands which have complex couplings. These include the commands :TIME, :POINTs, :DWELL, and TINTerval. Sending any one of a set singly will affect others of the set. However, if two are sent in a single program message, then these two shall set the values specified and the other changed. This is in accordance with the style guidelines and IEEE 488.2. See Syntax and Style section 7.4 for additional rules.

Default units in this subsystem are seconds unless otherwise specified.

KEYWORD	PARAMETER FORM	NOTES
:SWEep		
:COUNT	<numeric_value>	
:DIRection	UPIDOWN	
:DWELL	<numeric_value>	
:AUTO	<Boolean> ONCE	
:GENeration	STEPped ANALog	
:MODE	AUTO MANual	
:OFFSet		

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
:POINTs	<numeric_value>	1994
:TIME	<numeric_value>	1994
:OREference		1994
:LOCation	<numeric_value>	1994
:POINTs	<numeric_value>	1994
:POINTs	<numeric_value>	
:REALtime		1993
[:STATe]	<Boolean>	1993
:SPACing	LINEar LOGarithmic	
:STEP	<numeric_value>	
:TIME	<numeric_value>	
:AUTO	<Boolean> ONCE	
:LLIMit	<numeric_value>	
:TINTerval	<numeric_value>	1994

18.23.1 :COUNT <numeric_value>

SENSe:SWEEp:COUNT

This command determines the number of sweeps which are initiated or vectors (arrays of samples, waveforms, etc) which are acquired by a single trigger event.

At *RST, this value is set to 1.

18.23.2 :DIRection UP|DOWN

SENSe:SWEEp:DIRection

Controls the direction of the sweep. If UP is selected, the sweep is carried out from start to stop. If DOWN is selected, the sweep is carried out from stop to start. This command only applies to swept instruments.

At *RST, this value is set to UP.

18.23.3 :DWELl <numeric_value>

SENSe:SWEEp:DWELl

Controls the amount of time spent at each point during a sweep. This command only applies to swept instruments. Explicitly setting a value for SWEEp:DWELl sets SWEEp:DWELl:AUTO OFF.

The following coupling equation applies.

$$SWEEp:TINTerval = SWEEp:DWELl + <stepping\_time>$$

Changing DWELl will cause either TINTerval or <stepping_time> to change in a device dependent manner. See also introductory comments in the SWEEp subsystem.

<stepping_time> is the device dependent time required for the sweep generator to move from one point (swept entity) to the next.

Note that DWELl cannot exceed TINTerval. Setting DWELl to a greater value will cause error -222, "Data out of range" to be generated.

At *RST, this value is device-dependent.

18.23.3.1 :**AUTO** <Boolean>**IONCE**

SENSe:SWEep:DWELl:AUTO

When ON is selected, the dwell time is calculated internally when SWEep:TINterval is changed. Explicitly setting a value for SWEep:DWELl sets SWEep:DWELl:AUTO OFF.

At *RST, AUTO is set to ON.

18.23.4 :**GENeration STEPped**ANALog

SENSe:SWEep:GENeration

Determines if the sweep or acquisition is stepped or analog.

- STEPped: A stepped sweep or acquisition is selected. The number of points is set by the SWEep:POINts command.
- ANALog: The sweep or acquisition is controlled by an analog signal. SWEep:POINts is not applicable.

At *RST, this value is device-dependent.

18.23.5 :**MODE AUTO**IMANual

SENSe:SWEep:MODE

Determines whether the sweep is performed automatically, or if the actual swept entity is controlled by the MANual command in the appropriate subsystem (for example, FREQuency:MANual). This command only applies to swept instruments.

At *RST, this value is set to AUTO.

18.23.6 :**OFFSet**

SENSe:SWEep:OFFSet

This subsystem sets the offset between the offset reference point (SWEep:OREference) and the trigger point.

18.23.6.1 :**POINts** <numeric_value>

SENSe:SWEep:OFFSet:POINts

Sets the offset in terms of points. This value may be positive or negative. A positive value specifies that the offset reference point occurs after the trigger point.

The following coupling equation applies.

$$SWEep:OFFSet:POINts = SWEep:OFFSet:TIME / SWEep:TINterval$$

Changing SWEep:OFFSet:POINts will cause SWEep:OFFSet:TIME to change, but not SWEep:TINterval. See also introductory comments in the SWEep subsystem.

At *RST, this value is set to 0.

18.23.6.2 :**TIME** <numeric_value>

SENSe:SWEep:OFFSet:TIME

Sets the offset in units of time. This value may be positive or negative. A positive value specifies that the offset reference point occurs after the trigger point.

The following coupling equation applies.

$$SWEep:OFFSet:POINts = SWEep:OFFSet:TIME / SWEep:TINterval$$

Changing `SWEEP:OFFSET:TIME` will cause `SWEEP:OFFSET:POINTS` to change, but not `SWEEP:TINTerval`. See also introductory comments in the `SWEEP` subsystem.

At `*RST`, this value is set to 0.

18.23.7 :OREference

`SENSe:SWEEP:OREference`

This subsystem sets the location of the offset reference point in the sweep or acquisition. The offset reference is used by the `SWEEP:OFFSET` commands.

18.23.7.1 :LOCation <numeric_value>

`SENSe:SWEEP:OREference:LOCation`

Sets the offset reference by specifying a relative location. `SWEEP:OREference:LOCation` accepts a range of values from 0 to 1. A value of 0 selects the first point of acquisition as the reference. A value of 1 selects the last point.

The following coupling equation applies.

$$SWEEP:OREference:LOCation = (SWEEP:OREference:POINTS - 1) / (SWEEP:POINTS - 1)$$

Changing `SWEEP:OREference:LOCation` will cause `SWEEP:OREference:POINTS` to change, but not `SWEEP:POINTS`. See also introductory comments in the `SWEEP` subsystem.

At `*RST`, this value is device-dependent.

18.23.7.2 :POINTS <numeric_value>

`SENSe:SWEEP:OREference:POINTS`

Sets the offset reference by specifying a number of points. `SWEEP:OREference:POINTS` accepts values from 1 to N, where N is the maximum value of `SWEEP:POINTS`. A value of 1 selects the first point of acquisition as the reference.

The following coupling equation applies.

$$SWEEP:OREference:LOCation = (SWEEP:OREference:POINTS - 1) / (SWEEP:POINTS - 1)$$

Changing `SWEEP:OREference:POINTS` will cause `SWEEP:OREference:LOCation` to change, but not `SWEEP:POINTS`. See also introductory comments in the `SWEEP` subsystem.

At `*RST`, this value is device-dependent.

18.23.8 :POINTS <numeric_value>

`SENSe:SWEEP:POINTS`

Sets the number of points in a stepped sweep or acquisition. `POINTS` is ignored when `GENeration` is `ANALog`.

The following coupling equations apply.

$$SWEEP:TINTerval = SWEEP:TIME / (SWEEP:POINTS - 1)$$

Changing `SWEEP:POINTS` will cause either `SWEEP:TINTerval` or `SWEEP:TIME` to change in a device dependent manner. See also introductory comments in the `SWEEP` subsystem.

$$\text{SWEep:OREference:LOCation} = (\text{SWEep:OREference:POINTs} - 1) / (\text{SWEep:POINTs} - 1)$$

Changing SWEep:POINTs will cause either SWEep:OREference:LOCation or SWEep:OREference:POINTs to change in a device dependent manner. See also introductory comments in the SWEep subsystem.

When SPACing is LINear:

$$\text{SWEep:STEP} = \langle \text{swept\_subsystem} \rangle : \text{SPAN} / (\text{SWEep:POINTs} - 1)$$

Changing SWEep:POINTs will cause SWEep:STEP to change, but not $\langle \text{swept\_subsystem} \rangle : \text{SPAN}$.

When SPACing is LOGarithmic:

$$\langle \text{points\_per\_decade} \rangle = (\text{SWEep:POINTs} - 1) / \langle \text{decades\_in\_span} \rangle$$

Changing SWEep:POINTs will cause $\langle \text{points\_per\_decade} \rangle$ to change, but not $\langle \text{decades\_in\_span} \rangle$. $\langle \text{decades\_in\_span} \rangle$ is calculated internally from $\langle \text{swept\_subsystem} \rangle : \text{START}$, STOP , CENTER , and SPAN . Normally, $\langle \text{decades\_in\_span} \rangle$ equals $\log_{10}(\text{STOP} / \text{START})$, when $\text{STOP} > \text{START}$. See also introductory comments in the SWEep subsystem.

At *RST, this value is device-dependent.

18.23.9 :REALtime

SENSe:SWEep:REALtime

Instruments that acquire data directly as function of time, may use several data acquisition methods to sense the data points of the sweep.

There are techniques where the points in the requested sweep are collected from a number of successive acquisitions. In this case the result is a reconstruction of the original signal out of several acquisitions; this is not real time.

An acquisition is known as real time if no reconstruction is done; that is, the points in the resulting data set are in the order in which they were acquired.

18.23.9.1 [[:STATe]] <Boolean>

SENSe:SWEep:REALtime:STATe

When STATe is ON, the instrument is required to collect data in real time. When STATe is OFF, the instrument is allowed to use reconstruction techniques.

At *RST this function is OFF

18.23.10 :SPACing LINear|LOGarithmic

SENSe:SWEep:SPACing

Determines the time vs. swept entity characteristics of the sweep. The various settings have the following meanings:

- LINear: The swept entity is incremented (decremented) by the stepsize until the sweep limit is reached if the sweep generation is stepped. If the sweep is analog,

selecting this parameter specifies a ramp.

- LOGarithmic: Step size is determined by a logarithmic curve fitted between the start and stop value of the swept entity. Stepping is determined by the number of steps function.

At *RST, this value is set to LINear.

18.23.11 :STEP <numeric_value>

SENSe:SWEep:STEP

Controls the swept entity step size for a stepped linear sweep. STEP is ignored when GENeration is ANALog or when SPACing is LOGarithmic.

The following coupling equation applies.

$$SWEep:STEP = \langle swept\_subsystem \rangle :SPAN / (SWEep:POINts - 1)$$

Changing SWEep:STEP will cause SWEep:POINts to change, but not $\langle swept\_subsystem \rangle :SPAN$. See also introductory comments in the SWEep subsystem.

The default units are those of the associated $\langle swept\_subsystem \rangle :SPAN$ command.

At *RST, this value is device-dependent.

18.23.12 :TIME <numeric_value>

SENSe:SWEep:TIME

Sets the duration of the sweep or acquisition. Note that this does not turn sweeping or acquisition on. Explicitly setting a value for SWEep:TINterval or SWEep:TIME sets SWEep:TIME:AUTO OFF.

The following coupling equation applies.

$$SWEep:TINterval = SWEep:TIME / (SWEep:POINts - 1)$$

Changing SWEep:TIME will cause SWEep:TINterval to change, but not SWEep:POINts. See also introductory comments in the SWEep subsystem.

At *RST, this value is device-dependent.

18.23.12.1 :AUTO <Boolean>IONCE

SENSe:SWEep:TIME:AUTO

When ON is selected, SWEep:TIME and SWEep:TINterval are calculated internally when related settings outside the time domain coupling equations are changed. Explicitly setting a value for SWEep:TIME or SWEep:TINterval sets SWEep:TIME:AUTO OFF.

This command only applies to swept instruments, where explicitly setting SWEep:TIME and SWEep:TINterval is the exception. For instance, in a spectrum analyzer, a user normally sets the frequency parameters (FREQuency:SPAN and :CENTER) and expects the instrument to calculate the appropriate sweep time and interval to produce a reasonable resolution bandwidth. The :AUTO node should not be implemented in non-swept instruments that normally expect the user to explicitly set SWEep:TIME or SWEep:TINterval.

At *RST, AUTO is set to ON.

18.23.12.2 :LLIMit <numeric_value>

SENSe:SWEEp:TIME:LLIMit

Defines a lower limit for sweep time. This lower limit restricts the sweep time value set either explicitly or automatically.

At *RST, this value is device-dependent.

18.23.13 :TINterval <numeric_value>

SENSe:SWEEp:TINterval

Sets the time interval between points of the sweep or acquisition. Explicitly setting a value for SWEEp:TINterval or SWEEp:TIME sets SWEEp:TIME:AUTO OFF.

The following coupling equations apply.

$$SWEEp:TINterval = SWEEp:TIME / (SWEEp:POINts - 1)$$

Changing SWEEp:TINterval will cause SWEEp:TIME to change, but not SWEEp:POINts. See also introductory comments in the SWEEp subsystem.

$$SWEEp:OFFSet:POINts = SWEEp:OFFSet:TIME / SWEEp:TINterval$$

Changing SWEEp:TINterval will cause either SWEEp:OFFSet:POINts or SWEEp:OFFSet:TIME to change in a device dependent manner. See also introductory comments in the SWEEp subsystem.

At *RST, this value is device-dependent.

18.24 VOLTage Subsystem

SENSe:VOLTage

This subsection controls the voltage amplitude characteristics of the sensor.

KEYWORD	PARAMETER FORM	NOTES
:VOLTage		
:AC[:DC]		
:APERTure	<numeric_value>	
:NPLCycles	<numeric_value>	1991
:ATTenuation	<numeric_value>	
:AUTO	<Boolean>	
:PROTection		
[:LEVel]	<numeric_value>	
:STATe		
:TRIPped?		[query only]
:CLEar		[no query]
:RANGe		
[:UPPer]	<numeric_value>	
:LOWer	<numeric_value>	
:AUTO	<Boolean> ONCE	
:DIRection	UPIDOWN EITHer	
:LLIMit	<numeric_value>	1996
:ULIMit	<numeric_value>	1996
:OFFSet	<numeric_value>	1991,2
:PTPeak	<numeric_value>	1991,2
:REFeRence	<numeric_value>	
:STATe	<Boolean>	
:RESolution	<numeric_value>	
:AUTO	<Boolean> ONCE	
:DETeCTOR	INTernal EXTernal	1991

18 18.24.1 :ACI[:DC]

SENSe:VOLTage:AC

This subsystem selects the alternating current or the direct current sensor. The default units are volts.

18.24.1.1 :APERTure <numeric_value>

SENSe:VOLTage:AC:APERTure

Specifies the acquisition/sampling/gate time for a single measurement point. The parameter has a unit of seconds.

At *RST, this value is device-dependent.

18.24.1.2 :NPLCycles <numeric_value>

SENSe:VOLTage:AC:NPLCycles

Specifies the acquisition/sampling/gate time for a single measurement point in terms of the number of power line cycles. The parameter is unitless. If this command is implemented, the APERTure command must also be implemented.

The value is coupled to APERTure by the equation:

$$APERTure = NPLCycles / selected\ line\ frequency$$

At *RST, this value is device-dependent.

18.24.1.3 :ATTenuation <numeric_value>

SENSe:VOLTage:AC:ATTenuation

Sets the ATTenuation level. Note that when setting the level to 10 dB the magnitude of the incoming signal will be decreased by 10 dB. Changing the ATTenuation changes RANGE by the same amount.

Default units are as determined in the UNITS system.

18.24.1.3.1 :AUTO <Boolean>

SENSe:VOLTage:AC:ATTenuation:AUTO

Couples the attenuator to RANGE such that maximum dynamic range of the incoming signal is assured without overloading.

Programming a specified attenuation sets AUTO OFF.

At *RST, AUTO is set to ON.

18.24.1.4 :PROTection

SENSe:VOLTage:AC:PROTection

Controls the protection circuits.

18.24.1.4.1 [:LEVel] <numeric_value>

SENSe:VOLTage:AC:PROTection:LEVel

This command sets the input level at which the input protection circuit will trip.

18.24.1.4.2 :STATe <Boolean>

SENSe:VOLTage:AC:PROTection:STATe

Controls whether the input protection circuit is enabled.

At *RST, this value is set to ON.

18.24.1.4.3 :TRIPped?

SENSe:VOLTage:AC:PROTection:TRIPped?

This query returns a 1 if the protection circuit is tripped and a 0 if it is untripped. TRIPped only has a query form and thus has no associated *RST condition.

18.24.1.4.4 :CLEAr

SENSe:VOLTage:AC:PROTection:CLEAr

Causes the protection circuit to be cleared.

This command is an event and has no associated *RST condition.

18.24.1.5 :RANGe

SENSe:VOLTage:AC:RANGe

Sets the range for the amplitude sensor function. The RANGe endpoints may be specified in terms of the UPPer endpoint and the LOWer endpoint, alternatively the midpoint of the RANGe (OFFSet) and the span of the RANGe (PTPeak) may be used. A non-symmetrical range around zero may cause some instruments to introduce an actual offset to the signal entering the SENSe block.

18.24.1.5.1 [:UPPer] <numeric_value>

SENSe:VOLTage:AC:RANGe:UPPer

Specifies the most positive signal level expected for the sensor input.

At *RST, this value is device-dependent.

18.24.1.5.2 :LOWer <numeric_value>

SENSe:VOLTage:AC:RANGe:LOWer

Specifies the most negative signal level expected for the sensor input. This allows setting for the best dynamic range. This function is coupled to UPPer in a device-dependent manner.

At *RST, this value is device-dependent.

18.24.1.5.3 :AUTO <Boolean>IONCE

SENSe:VOLTage:AC:RANGe:AUTO

Sets the RANGe to the value determined to give the most dynamic range without overloading. The actual algorithm is device-dependent.

At *RST, AUTO is set to ON.

18.24.1.5.3.1 :DIRection UPIDOWNIEITHer

SENSe:VOLTage:AC:RANGe:AUTO:DIRection

The DIRection command defines the manner in which AUTO works. When UP is selected, the instrument shall only change the range automatically to ranges larger than the current range, as the input signal dictates. When DOWN is selected, the instrument shall only change the range automatically to ranges smaller than the current range, as the input signal dictates. When EITHER is selected the instrument may range in either direction.

At *RST, DIRection shall be set to EITHer.

18.24.1.5.3.2 :LLIMit <numeric_value>

SENSe:VOLTage:AC:RANGe:AUTO:LLIMit

The Lower LIMit command sets the smallest range to which the instrument will go while autoranging. When AUTO is ON and the present RANGe is outside the new limits, the instrument may change the RANGe to a value inside the limits or wait until the next time the autoranging algorithm is performed.

At *RST, LLIMit shall be set to MINimum.

18.24.1.5.3.3 :ULIMit <numeric_value>

SENSe:VOLTage:AC:RANGe:AUTO:ULIMit

The Upper LIMit command sets the largest range to which the instrument will go while

autoranging. When AUTO is ON and the present RANGE is outside the new limits, the instrument may change the RANGE to a value inside the limits or wait until the next time the autoranging algorithm is performed.

At *RST, ULIMit shall be set to MAXimum.

18.24.1.5.4 :OFFSet <numeric_value>

SENSe:VOLTage:AC:RANGe:OFFSet

OFFSet determines the midpoint of the range.

At *RST, this value is device-dependent.

18.24.1.5.5 :PTPeak <numeric_value>

SENSe:VOLTage:AC:RANGe:PTPeak

Peak To Peak specifies the dynamic range required for the sensor.

At *RST, this value is device-dependent.

18.24.1.6 :REference <numeric_value>

SENSe:VOLTage:AC:REference

Sets a reference amplitude for sensor instruments. This command differs in intent from the reference correction in that this is intended to be a measurement offset where a user wishes to reference a measurement to some known value.

At *RST, this value is device-dependent, but STATE is set to OFF.

18.24.1.6.1 :STATE <Boolean>

SENSe:VOLTage:AC:REference:STATE

Determines whether amplitude is measured in absolute or relative mode. If STATE is ON, then amplitude is referenced to the value set in REference.

At *RST, this value is set to OFF.

18.24.1.7 :RESolution <numeric_value>

SENSe:VOLTage:AC:RESolution

Specifies the absolute resolution of the measurement. For example, if a measurement of 10 Volts is specified with a resolution of .01 Volts, the measurement is returned with a resolution of .01 Volts (that is, 10.00).

RANGe is not coupled to RESolution. Setting RANGe shall not cause RESolution to change. Setting RESolution shall not cause RANGe to change.

At *RST, this value is device-dependent.

18.24.1.7.1 :AUTO <Boolean>|ONCE

SENSe:VOLTage:AC:RESolution:AUTO

Allows the system to determine the best resolution for the other measurement conditions.

Explicitly setting a value for :RESolution will turn AUTO OFF.

At *RST, this value is set to ON.

18.24.2 :DETector INTernal I EXTernal

SENSe:VOLTage:DETector

The DETector can be specified as internal or external.

At *RST, this value is set to INT.

18.25 WINDOW Subsystem

SENSe:WINDow

The window subsystem selects digital signal processing windowing parameters for use by FFT analyzers. Windows are time-domain weighting functions applied to input time records to condition the results of a time-domain to frequency-domain transformation. This subsystem controls windowing done to compensate for measurement artifacts. Analytical post-acquisition windowing is controlled with CALCulate:TRANSform.

KEYWORD	PARAMETER FORM	NOTES
:WINDow		
[:TYPE]	RECTangular subsystem UNIFORM FLATtop subsystem HAMMING HANNING KBESsel FORCE EXPonential	
:KBESsel	<numeric_value>	
:EXPonential	<numeric_value>	
:FORCE	<numeric_value>	

**18.25.1 [:TYPE] RECTangular|UNIFORM|FLATtop|HAMMING|HANNING
|KBESsel|FORCE|EXPonential**

SENSe:WINDow:TYPE

TYPE allows user selection from among several well-known or standard window types. Each window has characteristics that make it particularly suitable for certain kinds of measurements.

Individual windowing algorithms are defined in “*On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform*,” F. J. Harris, Proc of the IEEE, Vol. 66-1, January 1978, pp 51-83.

At *RST, this value is device-dependent.

18.25.1.1 :KBESsel <numeric_value>

SENSe:WINDow:TYPE:KBESsel

Enters the exponential decay time constant which characterizes the KBESsel window. The parameter to this command has units of seconds.

At *RST, this value is device-dependent.

18.25.1.2 :EXPonential <numeric_value>

SENSe:WINDow:TYPE:EXPonential

Enters the exponential decay time constant which characterizes the EXPonential window. The parameter to this command has units of seconds.

At *RST, this value is device-dependent.

18.25.1.3 :FORCE <numeric_value>

SENSe:WINDow:TYPE:FORCE

Enters the time value parameter corresponding to the width of the gated portion of the input time record for FORCE windows. The parameter of this command has units of seconds.

At *RST, this value is device-dependent.

19 SOURCE subsystem

The SOURCE setup commands are divided into several sections. Each section or subsystem deals with controls that directly affect device-specific settings of the device, and not those related to the signal-oriented characteristics. These commands are referenced through the SOURCE node.

The SOURCE node may be optional in a particular device. The reason that the SOURCE is optional is to allow devices which are primarily sources to accept shorter commands. The SOURCE node cannot be optional if either SENSE or ROUTe is optional, since only one node at a given level may be optional. For example, a typical power supply may elect to make the SOURCE node optional, since the most frequently used commands would be under the SOURCE subsystem. If the power supply also contained either source or routing functionality, these would be controlled through their respective nodes; however, the SENSE or ROUTe nodes would be required. An optional node, such as SOURCE, implies that the device shall accept and process commands with or without the optional node and have the same result. That is, the device is required to accept the optional node, if sent, without error.

In some instances, a source may contain subservient source or sensor functions. In such cases, the additional subservient functionality shall be placed as SENSE or SOURCE subnodes under the SOURCE subsystem. Further, no optional nodes (SENSE, SOURCE or ROUTe) are permitted if such subservient functionality exists in a device. Otherwise conflicts in interpreting commands would occur, for example, between SOURCE and [SENSE:]SOURCE if the SENSE were allowed to be optional.

CURRENT, POWER, VOLTage, FREQuency, and RESistance contain several sets of commands which have complex couplings. This includes the sweep commands START, STOP, CENTER, and SPAN as one set, and AMPLitude, OFFSet, HIGH, and LOW as another. Sending any one of a set singly will affect two others of the set. However, if two are sent in a single message unit, then these two should be set to the values specified and the other two changed. This is in accordance with the style guidelines and *IEEE 488.2*. START, STOP, CENTER, and SPAN must be implemented as a set. That is, if any one is implemented, then all must be implemented.

If the requested setting is unachievable given the other instrument settings, then an error must be generated (-211 “settings conflict”). The device may resolve the conflict in a device-dependent manner (for example, change START, STOP, CENTER, and/or SPAN) to resolve the error. Note that when more than one of the four sweep settings are received in the same program message unit, the sweep will be determined by the last two received.

The coupled commands may define commands as subnodes which alter these couplings. The command description will define how the couplings are altered. However, any command which alters couplings must default to the couplings described in this section as the *RST couplings.

When CENTER or SPAN is changed the following couplings apply:

SPAN	=	SPAN
CENTER	=	CENTER

1999 SCPI Command Reference

$$\begin{aligned}\text{STOP} &= \text{CENTer} + (\text{SPAN}/2) \\ \text{STARt} &= \text{CENTer} - (\text{SPAN}/2)\end{aligned}$$

When STARt or STOP is changed the following couplings apply:

$$\begin{aligned}\text{STARt} &= \text{STARt} \\ \text{STOP} &= \text{STOP} \\ \text{CENTer} &= (\text{STARt} + \text{STOP})/2 \\ \text{SPAN} &= \text{STOP} - \text{STARt}\end{aligned}$$

The couplings above assume linear units in the sweep specified.

HIGH, LOW, AMPLitude, and OFFSet have the same interactions described above for STARt, STOP, CENTer, and SPAN.

When AMPLitude or OFFSet is changed the following couplings apply:

$$\begin{aligned}\text{OFFSet} &= \text{OFFSet} \\ \text{AMPLitude} &= \text{AMPLitude} \\ \text{HIGH} &= \text{OFFSet} + (\text{AMPLitude}/2) \\ \text{LOW} &= \text{OFFSet} - (\text{AMPLitude}/2)\end{aligned}$$

When HIGH or LOW is changed the following couplings apply:

$$\begin{aligned}\text{HIGH} &= \text{HIGH} \\ \text{LOW} &= \text{LOW} \\ \text{OFFSet} &= (\text{HIGH} + \text{LOW})/2 \\ \text{AMPLitude} &= \text{HIGH} - \text{LOW}\end{aligned}$$

For CURRENT, POWER, and VOLTage when ATTenuation:AUTO is ON a coupling exists between ATTenuation and LEVel. This coupling is needed in sources which use an attenuator to extend the dynamic range of the source's level. Fig 19-1 illustrates this situation.

When ATTenuator:AUTO is ON, the full dynamic range of the instrument is available using the LEVel command. The instrument uses an algorithm which adjusts both the limited dynamic range signal and the attenuator to supply the requested level.

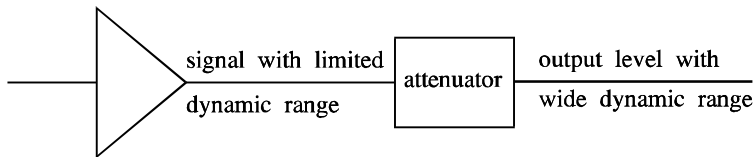


Figure 19-1 ATTenuation - LEVel Coupling Model

When ATTenuation:AUTO is OFF, the amount of attenuation is fixed and the range of allowed values for LEVel is reduced. The MINimum and MAXimum values for LEVel depend on the value of ATTenuation.

For example, assume an instrument has the block diagram illustrated in Fig 19-2.

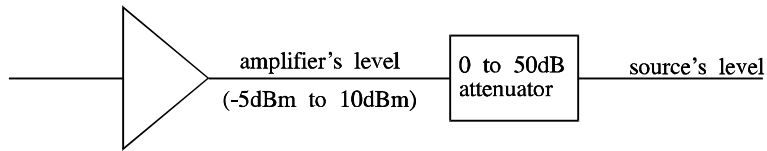


Figure 19-2 ATTenuation - LEVel Coupling Example

Say the output of the amplifier can vary from -5dBm to 10dBm and the attenuator can range from 0 to 50dB in steps of 10dB. The source's level can thus vary from -55dBm to 10dBm.

After *RST, POWER:ATTenuation:AUTO is ON allowing POWER:ATTenuation to automatically change depending on POWER[:LEVel]. The allowable range for POWER[:LEVel] is from -55dBm (MINimum) to 10dBm (MAXimum). If POWER[:LEVel] is set to -48dBm, POWER:ATTenuation is set to 50dB and the power level at the amplifier's output is 2dBm. If POWER[:LEVel] is -43dBm, the instrument could be in one of two states:

POWER:ATTenuation is -40dB and amplifier's level is -3dBm or

POWER:ATTenuation is -50dB and amplifier's level is 7dBm

Which state is chosen may depend on the previous state of POWER[:LEVe]] and the algorithm employed by the instrument.

The user now wants to prevent the attenuator from changing values. POWER:ATTenuation -40 is sent. Receiving a value for POWER:ATTenuation has the side effect of turning POWER:ATTenuation:AUTO OFF, see 7.4 in Syntax & Style. Because the attenuator cannot automatically change, the allowable range for POWER[:LEVel] is limited to the range -45dBm (MINimum) to -30dBm (MAXimum). Trying to program a value outside this range causes an Execution Error. The permissible range for POWER[:LEVel] can be changed by sending a new value for POWER:ATTenuation.

The actual algorithm employed in the instrument and the range of values depend on the underlying hardware implementation. SCPI does require MAXimum/MINimum queries which a user could use to discover how the bounds of the LEVel command change.

First Level SOURce Keywords

[:SOURce]		
:ACCeleration		1999
:AM		
:COMBine		1994
:CORRection		1992
:CURRent		
:DM		
:FM		
:FREQuency		
:FORCe		1999
:FUNction		
:LIST		
:MARKer		
:PHASe		
:PM		
:POWer		
:PULM		
:PULSe		
:RESistance		
:ROSCillator		
:SPEed		1999
:SWEep		
:TEMPerature		1993
:VOLTage		

19.1 **ACCEleration Subsystem**

SOURce:ACCEleration

This node controls the acceleration of the device.

KEYWORD	PARAMETER FORM	COMMENTS
:ACCEleration		1999
[:LEVel]	<numeric_value>	1999

19.1.1 **[:LEVel] <numeric_value>**

:SOURce:ACCEleration[:LEVel]

Sets the acceleration in m/s².

At *RST, this value is device dependent.

19.2 **AM Subsystem**

SOURce:AM

The Amplitude Modulation subsystem is used to set the modulation controls and the parameters associated with the modulating signal(s). The keywords to describe the carrier signal exist in other subsystems (such as, FREQuency and PHASE).

KEYWORD	PARAMETER FORM	COMMENTS
:AM		
:COUPling	AC DC GROund	
[:DEPT]h	<numeric_value>	
:EXTernal		
:COUPling	AC DC GROund	
:IMPedance	<numeric_value>	
:POLarity	NORMall INVerted	
:INTernal		
:FREQuency	<numeric_value>	
:MODE	FIXed LIST	1993
:POLarity	NORMall INVerted	
:SENSitivity	<numeric_value>	
:SOURce	EXTernal INTernal{,EXTernal ,INTernal}	
:STATe	<Boolean>	
:TYPE	LINear LOGarithmic EXPonential	1991

19.2.1 **:COUPling ACIDCIGROund**

SOURce:AM:COUPling

COUPling, at this level, is used to set the coupling between the modulator and the modulating signal. The modulating signal may be the sum of several signals, either internal or external sources. If COUPling is set to DC, then both AC and DC components of the signal pass. AC coupling passes only the AC component of the signal. The GROund parameter allows Amplitude Modulation to be turned ON without the modulating signal connected.

At *RST, this value is device-dependent.

19.2.2 **[:DEPT]h <numeric_value>**

SOURce:AM:DEPT]h

Sets the modulation DEPT]h of an AM signal. The unit for DEPT]h is percent (%). This command is used where the DEPT]h can be controlled independently from the modulating signal voltage level. Alternately, it is used to set a device capability that can indicate when the desired DEPT]h has been achieved.

At *RST, this value is device-dependent.

19.2.3 **:EXTernal**

SOURce:AM:EXTernal

The EXTernal subsystem is used to control the specified external signal source. Where multiple external modulation sources are available, the EXTernal keyword may have a

numeric suffix signifying the particular external signal source. If no number is given, a 1 is assumed.

19.2.3.1 :COUPling ACIDCIGROund

SOURce:AM:EXTernal:COUPling

COUPling, at this level, sets the coupling for only the specified external signal source. If COUPling is set to DC, then both AC Modulation and DC components of the signal pass. AC coupling passes only the AC component of the signal. The GROund parameter allows the specified external source to be turned ON without being connected.

At *RST, this value is device-dependent.

19.2.3.2 :IMPedance <numeric_value>

SOURce:AM:EXTernal:IMPedance

Sets the impedance of the specified external signal source. The unit for IMPedance is Ohms.

At *RST, this value is device-dependent.

19.2.3.3 :POLarity NORMallINVerted

SOURce:AM:EXTernal:POLarity

POLarity, at this level, sets the POLarity for only the specified external signal source. POLarity sets the relationship between the polarity of the applied modulation voltage and the resulting direction of modulation. The definition of “normal” polarity for AM modulated signal is: a positive voltage which causes the modulation envelope or the instantaneous carrier amplitude to increase.

At *RST, this value must be set to NORMal.

19.2.4 :INTernal

SOURce:AM:INTernal

The INTernal subsystem is used to control the specified internal signal source. Where multiple internal modulation sources are available, the INTernal keyword may have a numeric suffix signifying the particular internal signal source. If no number is given, a 1 is assumed.

19.2.4.1 :FREQuency <numeric_value>

SOURce:AM:INTernal:FREQuency

Sets the frequency of the specified internal signal source. Some instruments may also allow the internal signal source(s) to be controlled as a subsystem(s). The unit of FREQuency is Hz.

At *RST, this value is device-dependent.

19.2.5 :MODE

SOURce:AM:MODE

Determines which set of commands currently control the AM subsystem. The settings have the following meanings:

FIXed: Settings of individual commands are as last set by the user.

LIST: Settings of individual commands are controlled by the LIST subsystem.

At *RST the value is FIXed.

19.2.6 :POLarity NORMallINVerted

SOURce:AM:POLarity

POLarity, at this level, is used to set the polarity between the modulator and the modulating signal. The modulating signal may be the sum of several signals, either internal or external sources. POLarity sets the relationship between the polarity of the applied modulation voltage and the resulting direction of modulation. The definition of “normal” polarity for AM modulated signal is a positive voltage, causing the modulation envelope or the instantaneous carrier amplitude to increase.

At *RST, this value must be set to NORMal.

19.2.7 :SENSitivity <numeric_value>

SOURce:AM:SENSitivity

Controls the modulation depth by setting a sensitivity Modulation to the modulation signal voltage level. The unit for SENSitivity is percent/Volt (%/V).

At *RST, this value is device-dependent.

19.2.8 :SOURce EXTernallINTERNAL{,EXTernall,INTERNAL}

SOURce:AM:SOURce

Selects the source for the modulating signal. It may specify a single source or a number of sources. The specified sources, in the SOURce command, are all selected and turned on. Any sources from a previous selection that are not part of the current selection list are deselected and turned off. If multiple sources are selected and the device does not support the combination requested in the SOURce command, an execution error -221 will be generated.

Where multiple internal modulation sources are available, individual INTernal sources must be distinguished by a unique number suffix. If no number suffix is provided in a command, a 1 must be assumed.

Where multiple external modulation sources are available, individual EXTernal sources must be distinguished by a unique number suffix. If no number suffix is provided in a command, a 1 must be assumed.

At *RST, this value is device-dependent.

19.2.9 :STATe <Boolean>

SOURce:AM:STATe

This command is used to turn Amplitude Modulation ON or OFF. Turning AM modulation ON will not automatically turn OFF any other types of modulation. Turning any or all modulations types ON or OFF must be done explicitly. Where a type of modulation is active, turning ON another type of modulation will add to the set of active modulations if that combination of capabilities are legal in the device; otherwise, an execution error -221 is generated, and the instrument state is device-dependent.

At *RST, this value must be set to OFF.

19.2.10 :TYPE

SOURce:AM:TYPE

Controls the type of AM shaping

- LINear – The output voltage is directly proportional to Vmod.
- LOGarithmic – The output voltage is proportional to $\log(V_{\text{mod}})$.
- EXPonential – The output voltage is proportional to $10^{(V_{\text{mod}})}$

At *RST the selection will be LINear.

19.3 COMBine Subsystem

SOURce:COMBine

This subsystem is used to combine two or more signals or data streams into one.

KEYWORD	PARAMETER FORM	COMMENTS
:COMBine		1994
:FEED	<data_handle>	1994

19.3.1 :FEED <data_handle>

Sets or queries the data flow into the COMBine block. The number of FEED in a :COMBine block is device dependent.

At *RST, FEED is device dependent.

19.4 **CORRection Subsystem**

SOURce:CORRection

The correction subsystem provides for known external losses or gains. In contrast to CALibration, these losses usually are dependent on individual setups.

Correction data may be entered and stored in several ways. A mechanism is provided to automatically measure the signal and store the resulting “correction set”. This data can be selected at any time as the active “correction set”. The data could also be generated by an external device and transferred to the instrument over a computer interface. Commands are also provided to enter individual losses, gains, offsets, delays, etc. that the instrument should use in correcting its operation. An instrument is not required to implement all of the methods.

KEYWORD	PARAMETER FORM	COMMENTS
:CORRection		1992
[:STATe]	<Boolean>	1992
:COLLect		1992
[:ACQuire]		1992
:METHod	PMETer	1992
:SAVE	<name>	1992
:CSET		1992
[:SELEct]	<name>	1992
:STATe	<Boolean>	1992
:OFFSet		1992
[:MAGNitude]	<numeric_value>	1992
:PHASe	<numeric_value>	1992
:STATe	<Boolean>	1992
:LOSSIGAINISLOPe		1992
:STATe	<Boolean>	1992
[:OUTPut]		1992
[:MAGNitude]	<numeric_value>	1992
:PHASe	<numeric_value>	1992
:EDELay		1992
[:TIME]	<numeric_value>	1992
:DISTance	<numeric_value>	1992
:STATe	<Boolean>	1992
:RVELocity		1992
:MEDium	COAX WAVeguide	1992
:COAX	<numeric_value>	1992
:WAVeguide	<numeric_value>	1992
:FCUToff	<numeric_value>	1992
:STATe	<Boolean>	1992

19.4.1 **[:STATe] <Boolean>**

SOURce:CORRection:STATe

Determines whether the correction data defined in this section is applied to the signal.

At *RST, this function is OFF.

1999 SCPI Command Reference

19.4.2 :COLlect

SOURce:CORRection:COLlect

Deals with the collection of the correction data by the device. Details may be device specific.

19.4.2.1 [:ACQuire]

SOURce:CORRection:COLlect:ACQuire

A signal will be generated, measured, and the results saved as data for the chosen correction method.

19.4.2.2 :METHod PMETer

SOURce:CORRection:COLlect:METHod

Selects the correction method to be used for the correction that is about to be performed. If only one correction method exists in the instrument then this command is not required.

The various settings of the parameters have the following meanings:

- PMETer Use external Power METer to take readings.

At *RST, this value is device dependent.

19.4.2.3 :SAVE [<name>]

SOURce:CORRection:COLlect:SAVE

Calculates the correction data using the current method selection and then saves the correction data. If only one correction array exists then it is saved there. Otherwise it is saved in the set specified by <name>.

<name> may be either a <trace_name>, or a <table_name> which must be defined and exist in the instrument. Note this command will not create either <trace_name>, or a <table_name>. Trace definition is done in the TRACe subsystem. Table definition is done in the MEMory subsystem.

19.4.3 :CSET

SOURce:CORRection:CSET

The Correction SET subsystem is used to select the active correction set.

19.4.3.1 [:SElect] <name>

SOURce:CORRection:CSET:SElect

Specifies the active correction set. This is the set that is used when CORR:CSET:STAT ON is specified.

Note that the parameter is a <name> that must be defined and exist in the instrument. <trace_name>, or a <table_name> which must be defined and exist in the instrument. Note this command will not create either <trace_name>, or a <table_name>. Trace definition is done in the TRACe subsystem. Table definition is done in the MEMory subsystem.

19.4.3.2 STATe <Boolean>

SOURce:CORRection:CSET:STATe

Determines whether the correction data defined in the selected set is applied to the output.

At *RST, this function is OFF. OFF is chosen because all of the correction sets may be empty. The instrument may need to be supplied with data before correction can be applied.

19.4.4 :OFFSet

SOURce:CORRection:OFFSet

This node defines a single value that is added to the signal supplied by the instrument. This command shall be used with the currently selected units. Offset effects the signal values as additive (or subtractive) for linear units such as VOLT.

19.4.4.1 [:MAGNitude] <numeric_value>

SOURce:CORRection:OFFSet:MAGNitude

This is the magnitude value of the correction offset data. Magnitude is always added linearly, as in volts.

At *RST, this value is set to zero.

19.4.4.2 :PHASe <numeric_value>

SOURce:CORRection:OFFSet:PHASe

This is the phase value of the correction offset data. The units associated with this are the currently selected angle units.

At *RST, this value is set to zero.

19.4.4.3 :STATe <Boolean>

SOURce:CORRection:OFFSet:STATe

This function is used to enable or disable the offset correction.

At *RST, this function is set to OFF.

19.4.5 :LOSSI:GAINI:SLOPe

SOURce:CORRection:LOSS

These commands are used either to provide a mechanism to apply simple correction or to correct external factors that have not been accounted for in the correction set.

Loss, gain and slope affect the signal amplitude as a multiplier for linear units such as VOLT and as additive (or subtractive) for logarithmic units such as dB.

:LOSS This command is used to set the anticipated loss in the system, excluding those factors covered by the active correction set. The device then adjusts the signal by this factor to compensate for the loss. Loss is coupled to GAIN by the equation $LOSS = 1/GAIN$ when the default unit is linear and $LOSS = -GAIN$ when the default unit is logarithmic.

:GAIN This command is used to set the anticipated gain in the system, excluding those factors covered by the active correction set. The device then adjusts the signal by this factor to compensate for the gain. Gain is coupled to LOSS by the equation $GAIN = 1/LOSS$ when the default unit is linear and $GAIN = -LOSS$ when the default unit is logarithmic.

:SLOPe This node defines an increasing or decreasing value that is added to the signal. This correction value is zero at a zero value of the x axis and increase or decreases from there.

19.4.5.1 :STaTe <Boolean>

SOURce:CORRection:LOSS:STaTe

This function is used to enable or disable the correction for LOSS or GAIN.

At *RST, this function is set to OFF.

19.4.5.2 [:OUTPut]

SOURce:CORRection:LOSS:OUTPut

The OUTPut node is used to reference measurements made at the instruments output.

19.4.5.2.1 [:MAGNitude] <numeric_value>

SOURce:CORRection:LOSS:OUTPut:MAGNitude

This is the magnitude value of the correction for LOSS, GAIN. The units associated with LOSS and GAIN are the current relative amplitude units.

At *RST, this value is set to zero.

19.4.5.2.2 :PHASe <numeric_value>

SOURce:CORRection:LOSS:OUTPut:PHASe

This is the phase value of the correction for LOSS or GAIN. The units associated with this are the currently selected angle units.

At *RST, this value is set to zero.

19.4.6 :EDELay

SOURce:CORRection:EDELay

The Electrical DELay is used to compensate for delays in the signal path.

This command is used to correct external factors that have not been accounted for in the correction set.

TIME and DISTance are coupled by the equation:

$$TIME = RVELOCITY * DISTANCE$$

RVELOCITY is not coupled to either TIME or DISTance.

19.4.6.1 [:TIME] <numeric_value>

SOURce:CORRection:EDELay:TIME

This command will set the electrical delay with the time parameter given. The units for this command are seconds. A value accepted here will modify the DISTance setting because they are the same parameter in different forms.

At *RST, the value of this setting is zero.

19.4.6.2 :DISTance <numeric_value>

SOURce:CORRection:EDELay:DISTance

This command will set the electrical delay with the distance parameter given. The effective delay will be computed using this parameter and the relative velocity set in the instrument. The units for this command are meters. A value accepted here will modify the time setting because they are the same parameter in different forms. Implementation of this command requires implementation of CORRection:EDELay[:TIME].

At *RST, the value of this setting is zero.

19.4.6.3 :STATe <Boolean>

SOURce:CORRection:EDELay:STATe

This function is used to enable or disable the correction for electrical delay.

At *RST, this function is set to OFF.

19.4.7 :RVELocity

SOURce:CORRection:RVELocity

This command corrects for the relative velocity of the medium connecting to the Device Under Test. This factor will be used in any delay corrections performed by the instrument.

This command is used to correct external factors that that have not been accounted for in the correction set.

TIME and DISTance are coupled by the equation:

$$\text{TIME} = \text{RVELocity} * \text{DISTance}$$

RVELocity is not coupled to either TIME or DISTance.

19.4.7.1 :MEDium COAXIWAVeguide

SOURce:CORRection:RVELocity:MEDium

This command selects the method of correction for any operation that uses relative velocity. For example, for COAX a linear phase shift will be applied, but for WAVeguide (a dispersive medium) a different phase shift is calculated.

At *RST, this value is device dependent.

19.4.7.2 :COAX <numeric_value>

SOURce:CORRection:RVELocity:COAX

This command sets the relative velocity for coaxial transmission lines. The parameter is unitless.

At *RST, the value of this setting is one.

19.4.7.3 :WAVeguide <numeric_value>

SOURce:CORRection:RVELocity:WAVeguide

This command sets the relative velocity factor for waveguide. The parameter is unitless.

At *RST, the value of this setting is one.

19.4.7.3.1 :FCUToff <numeric_value>

SOURce:CORRection:RVELocity:WAVeguide:FCUToff

This command specifies the frequency cutoff of the waveguide medium. The units are Hertz.

At *RST, the value of this setting is device dependent

19.4.7.4 :STATe <Boolean>

SOURce:CORRection:RVELocity:STATe

This function is used to enable or disable the relative velocity correction.

1999 SCPI Command Reference

At *RST, the value of this setting is OFF.

19.5 **CURRent Subsystem**

SOURce:CURRent

This subsection controls the signal amplitude characteristics of the source.

There is only one subsystem for each of the main amplitude characteristics of the signal. Thus, AC voltage and DC voltage are both placed under the VOLTage subsystem. A device which implemented both simultaneously would implement them as two separate sources, one AC and one DC.

KEYWORD	PARAMETER FORM	COMMENTS
SOURce:CURRent		
:ATTenuation	<numeric_value>	
:AUTO	<Boolean> ONCE	
:ALC		
[:STATe]	<Boolean>	
:SEARch	<Boolean> ONCE	
:SOURce	INTernal DIODE PMETer MMHead	
:BANDwidth	<numeric_value>	
!BWIDth		
:AUTO	<Boolean> ONCE	
:CENTer	<numeric_value>	
[:LEVel]		
[:IMMediate]		
[:AMPLitude]	<numeric_value>	
:AUTO	<Boolean> ONCE	1991
:OFFSet	<numeric_value>	
:HIGH	<numeric_value>	
:LOW	<numeric_value>	
:TRIGgered		
[:AMPLitude]	<numeric_value>	
:OFFSet	<numeric_value>	
:HIGH	<numeric_value>	
:LOW	<numeric_value>	
:LIMit		
[:AMPLitude]	<numeric_value>	
:OFFSet	<numeric_value>	
:HIGH	<numeric_value>	
:LOW	<numeric_value>	
:STATe	<Boolean>	
:MANual	<numeric_value>	
:MODE	FIXed SWEep LIST	
:PROTection		
[:LEVel]	<numeric_value>	
:STATe		
:TRIPped?		[query only]

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	COMMENTS
:CLEar		[no query]
:RANGe	<numeric_value>	
:AUTO	<Boolean> ONCE	
:REFerence	<numeric_value>	
:STATe	<Boolean>	
:SLEW	<numeric_value>	
:SPAN	<numeric_value>	
:HOLD	<Boolean>	
:LINK	CENTer START STOP	
:FULL		[no query]
:STARt	<numeric_value>	
:STOP	<numeric_value>	

19.5.1 :ATTenuation <numeric_value>

SOURce:CURRent:ATTenuation

Sets the ATTenuation level. Note that when increasing the level by 10 dB the magnitude of the outgoing signal as well as the LEVel will be decreased by 10 dB.

Default units are as determined in the UNIT system. This is coupled to LEVel.

19.5.1.1 :AUTO <Boolean>

SOURce:CURRent:ATTenuation:AUTO

Couples the attenuator to LEVel.

Programming a specified attenuation sets AUTO OFF.

See SOURce subsystem introduction for additional explanation.

At *RST, AUTO is set to ON.

19.5.2 :ALC

SOURce:CURRent:ALC

This subsystem command controls the automatic leveling control of the source.

19.5.2.1 [:STATe] <Boolean>

SOURce:CURRent:ALC:STATe

Controls whether the ALC loop controls the output level.

At *RST, this value is device-dependent.

19.5.2.2 :SEARCh <Boolean>|ONCE

SOURce:CURRent:ALC:SEARCh

SEARCh enables a form of leveling where the output level is calibrated by momentarily closing the leveling loop. Once the correct amount of level adjustment has been determined to produce the desired output level, the leveling loop is opened.

When ON, the search routine will be done anytime the output level requires change (for example, when LEVel is changed).

1999 SCPI Command Reference

Selecting SEARCH ONCE will have the effect of setting SEARCH to ON and then OFF. A modulator setting is determined by the system which will not be changed until the output level is explicitly reprogrammed.

At *RST, this value is set to OFF.

19.5.2.3 :SOURce INTernalDIODElPMETERlMMHead

SOURce:CURRent:ALC:SOURce

Selects the source of the feedback signal for ALC. The parameters have the following meanings:

- INTernal — The ALC feedback signal is measured from a point inside the source.
- DIODE — The ALC feedback is being fed from an external diode detector.
- PMETER — The ALC signal is coming from a power meter.
- MMHead — A millimeter head, with wuilt in leveling, is being used to supply the ALC signal.

At *RST, this value is INTernal.

19.5.2.4 :BANDwidthl:BWIDth <numeric_value>

SOURce:CURRent:ALC:BANDwidth

Controls the bandwidth of the ALC feedback signal. This parameter is in units of Hz.

At *RST, this value is device-dependent.

19.5.2.4.1 :AUTO <Boolean>IONCE

SOURce:CURRent:ALC:BANDwidth:AUTO

Couples the bandwidth of the ALC feedback signal to instrument-dependent parameters. When AUTO is ON, the best bandwidth is chosen to reflect the current instrument state.

Explicitly selecting a value for ALC:BANDwidth sets AUTO OFF.

At *RST, this value is set to ON.

19.5.3 :CENTER <numeric_value>

SOURce:CURRent:CENTer

Sets the center amplitude.

At *RST, this value is set to MINimum.

19.5.4 [:LEVel]

SOURce:CURRent:LEVel

This subsystem is used to control the signal amplitude when the source is operating in a continuous or FIXed MODE.

19.5.4.1 [:IMMediate]

SOURce:CURRent:LEVel:IMMediate

Indicates that the subsequent specification of a new signal amplitude is to be processed by the device without waiting for further commands.

19.5.4.1.1 **[[:AMPLitude] <numeric_value>**

SOURce:CURRent:LEVel:IMMediate:AMPLitude

Sets the actual magnitude of the unswept output signal in terms of the current operating units. The units are set to the default value, or alternately to a different value under the UNIT subsystem. AMPLitude may be used to specify the level for either a time varying or non-time varying signal.

Note the optional nodes exist to enable CURRent to be directly followed by a <numeric_value>. This is useful for programming simple devices in a straightforward manner.

At *RST, the signal being sourced should be set to a “safe” condition. This is generally achieved by setting the amplitude to its minimum value in conjunction with OUTPut:STATe subsystem.

19.5.4.1.1.1 **:AUTO <Boolean>IONCE**

SOURce:CURRent:LEVel:IMMediate:AMPLitude:AUTO

If AUTO ON is selected, the voltage setting will be adjusted automatically when a new current setting (with the existing voltage setting) exceeds the maximum power limit. If AUTO OFF is selected a new current setting will cause an execution error -221 (Settings conflict), and the instrument state is device dependent when the maximum power limit is exceeded.

*RST condition: AUTO OFF

19.5.4.1.2 **:OFFSet <numeric_value>**

SOURce:CURRent:LEVel:IMMediate:OFFSet

Sets the non-time varying component of the signal that is added to the time varying signal specified in AMPLitude, in terms of the current operating units. The units are set to the default value, or alternately to a different value under the UNIT subsystem. When SOURce:FUNCTION is DC, the effect of the OFFSet is device-dependent.

At *RST, the signal being sourced should be set to a “safe” condition. This is generally achieved by setting the offset to its minimum value in conjunction with OUTPut:STATe subsystem.

19.5.4.1.3 **:HIGH <numeric_value>**

SOURce:CURRent:LEVel:IMMediate:HIGH

Sets the more positive peak of a time varying signal. It is used in conjunction with LOW.

At *RST, the signal being sourced should be set to a “safe” condition. This is achieved by setting this to a minimum value in conjunction with OUTPut:STATe subsystem.

19.5.4.1.4 **:LOW <numeric_value>**

SOURce:CURRent:LEVel:IMMediate:LOW

Sets the more negative peak of a time varying signal, it is used in conjunction with HIGH.

At *RST, the signal being sourced should be set to a “safe” condition. This is achieved by setting this to a minimum value in conjunction with OUTPut:STATe subsystem.

19.5.4.2 :TRIGgered

SOURce:CURRent:LEVel:TRIGgered

Indicates that subsequent specification of a new signal amplitude is to be transferred to the IMMEDIATE value upon receipt of a trigger signal.

At *RST, after the receipt of an ABORT command or upon being triggered, the value in the TRIGgered subsystem tracks the values in the IMMEDIATE subsystem until a command in the TRIGgered subsystem is received. The purpose of tracking is so that spurious triggers do not change the value of LEVel unexpectedly.

19.5.4.2.1 [:AMPLitude] <numeric_value>

SOURce:CURRent:LEVel:TRIGgered:AMPLitude

Sets the actual magnitude of the unswept output signal in terms of the current operating units. The units are set to the default value, or alternately to a different value under the UNIT subsystem. AMPLitude may be used to specify the level for either a time varying or non-time varying signal.

19.5.4.2.2 :OFFSet <numeric_value>

SOURce:CURRent:LEVel:TRIGgered:OFFSet

Sets the non-time varying component of the signal that is added to the time varying signal specified in AMPLitude. The units are set to the default value, or to a different value under the UNIT subsystem. When SOURce:FUNCTION is DC, the effect of OFFSet is device-dependent.

19.5.4.2.3 :HIGH <numeric_value>

SOURce:CURRent:LEVel:TRIGgered:HIGH

This command is used to set the more positive peak of a time varying signal. It is used in conjunction with LOW.

19.5.4.2.4 :LOW <numeric_value>

SOURce:CURRent:LEVel:TRIGgered:LOW

This command is used to set the more negative peak of a time varying signal. It is used in conjunction with HIGH.

19.5.5 :LIMit

SOURce:CURRent:LIMit

Sets the maximum bounds on the output value. Setting a larger value will cause the output level to be clamped to the LIMit value.

19.5.5.1 [:AMPLitude] <numeric_value>

SOURce:CURRent:LIMit:AMPLitude

Sets the limit on the actual magnitude of the unswept output signal in terms of the current operating units. The units are set to the default value, or alternately to a different value under the UNIT subsystem. AMPLitude may be used to specify the level for either a time varying or non-time varying signal. OFFSet may only be used with AMPLitude, where AMPLitude is used to specify a time varying load.

1999 SCPI Command Reference

Explicit reference to the AMPLitude node is optional. Any device that implements this subsystem must accept and process commands, with or without the AMPLitude node, and respond to them in the same way.

At *RST, this value is device-dependent.

19.5.5.2 :OFFSet <numeric_value>

SOURce:CURRent:LIMit:OFFSet

Sets a non-time varying component limit of signal that is added to the time varying signal specified in AMPLitude. The units are set to the default value, alternately to a different value under the UNIT subsystem.

At *RST, this value is device-dependent.

19.5.5.3 :HIGH <numeric_value>

SOURce:CURRent:LIMit:HIGH

Sets the more positive peak limit of a time varying signal, it is used in conjunction with LOW.

At *RST, this value is device-dependent.

19.5.5.4 :LOW <numeric_value>

SOURce:CURRent:LIMit:LOW

Sets the more negative peak limit of a time varying signal, it is used in conjunction with HIGH.

At *RST, this value is device-dependent.

19.5.5.5 :STATe <Boolean>

SOURce:CURRent:LIMit:STATe

Controls whether the LIMit is enabled.

At *RST, this value is set to ON.

19.5.6 :MANual <numeric_value>

SOURce:CURRent:MANual

Allows manual adjustment of the amplitude between the sweep limits. The actual amplitude level is determined by the parameter only if SWEep:MODE is set to MANual and the amplitude MODE is set to SWEep. If the sweep limits are changed such that the manual amplitude value would be outside the limits, the manual amplitude value will be set to the nearest limit.

At *RST, this value is device-dependent.

19.5.7 :MODE FIXed|SWEep|LIST

SOURce:CURRent:MODE

Determines which set of commands control the amplitude subsystem. If FIXed is selected, the amplitude is determined by the LEVEL command under the appropriate subsystem. If SWEep is selected, the source is in the swept mode and amplitude is determined by START, STOP, CENTER, SPAN, and MANual commands under the appropriate subsystem. If LIST is

1999 SCPI Command Reference

selected, the amplitude values are determined by the appropriate amplitude list (such as LIST:).

At *RST, this value is set to FIXed.

19.5.8 :PROTection

SOURce:CURRent:PROTection

Controls the protection circuits.

19.5.8.1 [:LEVel] <numeric_value>

SOURce:CURRent:PROTection:LEVel

This command sets the output level at which the output protection circuit will trip.

19.5.8.2 :STATe <Boolean>

SOURce:CURRent:PROTection:STATe

Controls whether the output protection circuit is enabled.

At *RST, this value is set to ON.

19.5.8.3 :TRIPped?

SOURce:CURRent:PROTection:TRIPped?

This query returns a 1 if the protection circuit is tripped and a 0 if it is untripped.

TRIPped only has a query form and thus has no associated *RST condition.

19.5.8.4 :CLEAr

SOURce:CURRent:PROTection:CLEAr

Causes the protection circuit to be cleared.

This command is an event and has no associated *RST condition.

19.5.9 :RANGe <numeric_value>

SOURce:CURRent:RANGe

Sets a range for the output amplitude. This command is very hardware-specific.

19.5.9.1 :AUTO <Boolean>|ONCE

SOURce:CURRent:RANGe:AUTO

Couples the RANGe to an instrument-determined value. When AUTO is ON, the best range is chosen to reflect the current instrument state.

Explicitly selecting a value for RANGe sets AUTO OFF.

At *RST, this value is set to ON.

19.5.10 :REFerence <numeric_value>

SOURce:CURRent:REFerence

Sets a reference value which, if STATe is ON, allows all amplitude parameters to be queried/set as relative to the reference value.

At *RST, this value is device-dependent.

19.5.10.1 :STATe <Boolean>

SOURce:CURRent:REfERENCE:STATe

Determines whether amplitude is measured/output in absolute or relative mode. If STATe is ON, then amplitude is referenced to the value set in REfERENCE.

At *RST, this value is set to OFF.

19.5.11 :SLEW <numeric_value>

SOURce:CURRent:SLEW

Sets the slew rate of the output change when a new output level is programmed. The units are in (the currently active) amplitude unit/sec.

At *RST, this value is device-dependent.

19.5.12 :SPAN <numeric_value>

SOURce:CURRent:SPAN

Sets the amplitude span. If the current amplitude unit is logarithmic (dBm, dBuV, etc), then the unit of SPAN is dB. Otherwise SPAN is programmed in the current amplitude unit.

At *RST, this value is set to 0.

19.5.12.1 :FULL

SOURce:CURRent:SPAN:FULL

When this command is received, START amplitude is set to its minimum value and STOP amplitude is set to its maximum value. CENTER amplitude and SPAN are set to their coupled values.

This command is an event rather than a state. Therefore it has no associated query, or meaning at *RST.

19.5.12.2 :HOLD <Boolean>

SOURce:CURRent:SPAN:HOLD

Provides a mechanism to prevent the SPAN from being changed implicitly by the defined coupling between START, STOP, CENTER and SPAN. When HOLD is set to ON, SPAN shall only be changed by explicitly sending the SPAN command.

At *RST, this value is set to OFF.

19.5.12.3 :LINK CENTER|START|STOP

SOURce:CURRent:SPAN:LINK

Allows the default couplings for SPAN to be overridden. LINK selects the parameter, either CENTER, START or STOP, that shall not be changed when SPAN's value is changed. For example, if LINK is set to START then changing SPAN shall cause CENTER and STOP, not START to change.

At *RST, this value is set to CENTER, to be compatible with the default definition for the couplings.

19.5.13 :START <numeric_value>

SOURce:CURRent:START

Sets START amplitude.

At *RST, this value is set to MINimum.

19.5.14 **:STOP <numeric_value>**

SOURce:CURRent:STOP

Sets STOP amplitude.

At *RST, this value is set to MINimum.

19.6 **DM Subsystem**

SOURce:DM

The Digital Modulation subsystem is used to set the modulation controls and the parameters associated with the modulating signal(s). This modulation embraces a number of digital modulation techniques, which are described in the general terms of I and Q components. The keywords to describe the carrier signal exist in other subsystems (such as, FREQuency and PHASe).

KEYWORD	PARAMETER FORM	COMMENTS
:DM		
:FORMat	BPSK PSK2 QPSK PSK4 PSK8 QAM16 QAM64 QAM256 QAM1024 PRS9 PRS25 PRS49 PRS81	
:STATe	<Boolean>	
:SOURce	EXTeRnal PRBSICALibrate	
:FILTer		
[:SOURce]	INTernal EXTeRnal	
:ICORrection	<numeric_value>	
:QCORrection	<numeric_value>	
:IQRatio		
:STATe	<Boolean>	
[:MAGNitude]	<numeric_value>	
:LEAKage		
:STATe	<Boolean>	
[:MAGNitude]	<numeric_value>	
:ANGLE	<numeric_value>	
:QUADrature		
:STATe	<Boolean>	
:ANGLE	<numeric_value>	
:COUPling		
[:ALL]	AC DC GROund	
:DATA	AC DC GROund	
:CLOCK	AC DC GROund	
:THReshold		
[:ALL]	<numeric_value>	
:DATA	<numeric_value>	
:CLOCK	<numeric_value>	
:DMODE	SERial PARallel	
:FRAME		
:SOURce	INTernal EXTeRnal	
:POLarity		
[:ALL]	NORMal INVerted	
:I<n>	NORMal INVerted	
:Q<n>	NORMal INVerted	
:ICLock	NORMal INVerted	
:QCLock	NORMal INVerted	

1999 SCPI Command Reference

	KEYWORD	PARAMETER FORM	COMMENTS
	:CLOCK		
	:SOURce	NONE EXTeRnal	
19.6.1	:FORMat <modulation format>		
	SOURce:DM:FORMat		
			Sets the format type to be employed. It is expandable to any PRS<n>, QAM<n>, QPR<n> or PSK<n>. QPSK is an alias for PSK4 and BPSK is an alias for PSK2.
			At *RST, this value is device-dependent.
19.6.2	:STATe <Boolean>		
	SOURce:DM:STATe		
			Turns the digital modulation subsystem ON or OFF. Turning digital modulation ON will not automatically turn OFF any other types of modulation. Turning any or all modulations types ON or OFF must be done explicitly. Where a type of modulation is active, turning ON another type of modulation will add to the set of active modulations if that combination of capabilities are legal in the device. Otherwise, an execution error -221 is generated and the instrument state is left undefined.
			At *RST, this value must be set to OFF.
19.6.3	:SOURce EXTeRnal PRBSICALibrate		
	SOURce:DM:SOURce		
			Sets the source of digital modulation data. EXTeRnal means the digital data comes in externally. The data can be also generated internally with PRBS (a pseudo-random bit sequencer). CALibrate is a service feature for calibrating the baseband of the device.
			At *RST, this value must be set to EXTeRnal.
19.6.4	:FILTer		
	SOURce:DM:FILTer		
			This subsystem is used to control the I and Q filters.
19.6.4.1	[[:SOURce] INTeRnal EXTeRnal		
	SOURce:DM:FILTer:SOURce		
			This mnemonic allows a choice between INTeRnal or EXTeRnal I and Q filters.
			At *RST, this value must be set to INTeRnal.
19.6.4.2	:ICORrection <numeric_value>		
	SOURce:DM:FILTer:ICORrection		
			This mnemonic allows the entry of gain information about the EXTeRnal I filter. This has no effect on the signal when INTeRnal filter source is selected.
			AT *RST, this value must be set to 0.0DB.
19.6.4.3	:QCORrection <numeric_value>		
	SOURce:DM:FILTer:QCORrection		
			This mnemonic allows the entry of gain information about the EXTeRnal Q filter. This has no effect on the signal when INTeRnal filter source is selected.

At *RST, this value must be set to 0.0DB.

19.6.5 :IQRatio

SOURce:DM:IQRatio

This subsystem controls the gain imbalance between the I and Q channels.

19.6.5.1 :STATe <Boolean>

SOURce:DM:IQRatio:STATe

Turns ON and OFF the gain imbalance correction.

At *RST, this value must be set to OFF.

19.6.5.2 [:MAGNitude] <numeric_value>

SOURce:DM:IQRatio:MAGNitude

Specifies the correction for the gain imbalance between the I and Q channels. This value is equal to I/Q.

At *RST, this value must be set to 0.0DB.

19.6.6 :LEAKage

SOURce:DM:LEAKage

Carrier leakage is an offset added to the modulation pattern. Carrier leakage is measured as the magnitude relative to the full scale of the original signal. Hence a carrier leakage magnitude of 0dB would mean that the offset in the signal would be 1 full scale. This subsystem allows specification of the carrier leakage in angle-magnitude notation.

19.6.6.1 :STATe <Boolean>

SOURce:DM:LEAKage:STATe

This command turns carrier leakage ON and off.

At *RST, this value must be set to OFF.

19.6.6.2 [:MAGNitude] <numeric_value>

SOURce:DM:LEAKage:MAGNitude

Specifies the magnitude of carrier leakage relative to the full scale of the signal. 0.0dB means add 1 full scale of carrier leakage. 40dB would mean add (full scale - 40) amount of dBm.

At *RST, this value is device-dependent.

19.6.6.3 :ANGLE <numeric_value>

SOURce:DM:LEAKage:ANGLE

Specifies the angle of carrier leakage.

At *RST, this value is device-dependent.

19.6.7 :QUADrature

SOURce:DM:QUADrature

This subsystem controls quadrature correction.

19.6.7.1 **:STATe <Boolean>**

SOURce:DM:QUADrature:STATe

Turns ON or OFF the quadrature angle correction.

At *RST, this value must be set to OFF.

19.6.7.2 **:ANGLE <numeric_value>**

SOURce:DM:QUADrature:ANGLE

Specifies the angle of quadrature correction.

At *RST, this value must be set to 0.0DEG.

19.6.8 **:COUPling**

SOURce:DM:COUPling

This subsystem controls the input coupling for the digital modulation inputs.

19.6.8.1 **[[:ALL]] ACIDCIGROund**

SOURce:DM:COUPling:ALL

Sets the coupling of both the clock and data inputs to the instrument. This command is an event, and therefore does not have an associated *RST condition

19.6.8.2 **:DATA ACIDCIGROund**

SOURce:DM:COUPling:DATA

Sets the coupling of the data inputs to the instrument.

At *RST, this value is device-dependent.

19.6.8.3 **:CLOCK ACIDCIGROund**

SOURce:DM:COUPling:CLOCK

Sets the coupling of the clock inputs to the instrument.

At *RST, this value is device-dependent.

19.6.9 **:THReshold**

SOURce:DM:THReshold

This subsystem controls the detector thresholds for the digital modulation inputs.

19.6.9.1 **[[:ALL]] <numeric_value>**

SOURce:DM:THReshold:ALL

Sets the threshold of both the clock and data inputs to the instrument.

19.6.9.2 **:DATA <numeric_value>**

SOURce:DM:THReshold:DATA

Sets the threshold of the data inputs to the instrument.

At *RST, this value is device-dependent.

19.6.9.3 **:CLOCK <numeric_value>**

SOURce:DM:THReshold:CLOCK

Sets the threshold of the clock inputs to the instrument.

At *RST, this value is device-dependent.

19.6.10 :DMODE SERIALPARallel

SOURCE:DM:DMODE

This command sets the data transfer mode. SERIAL means data is transferred serially and PARallel means the data is transferred in a parallel fashion.

At *RST, this value is device-dependent.

19.6.11 :FRAME

SOURCE:DM:FRAME

This subsystem controls the framing for serialized digital modulation.

19.6.11.1 :SOURCE INTERNALEXTERNAL

SOURCE:DM:FRAME:SOURCE

Selects the source frame when data is transferred serially. INTERNAL selects autoframing, while EXTERNAL selects user-supplied framing.

19.6.12 :POLarity [:ALL] NORMallINVerted

SOURCE:DM:POLarity

Sets the polarity of all the clock and data inputs to the instrument.

19.6.12.1 :I<n> NORMallINVerted

SOURCE:DM:POLarity NORMal:I<n>

Selects the polarity of data inputs for the I channel. <n> selects which data input where n can be any number from 0-1 (largest DM format log4). For example, the range for n is 0-4 when the most complex mod format allowed is QAM1024.

At *RST, all I polarities must be set to NORM.

19.6.12.2 :Q<n> NORMallINVerted

SOURCE:DM:POLarity NORMal:Q<n>

Selects the polarity of data inputs for the Q channel. <n> selects which data input where n can be any number from 0-1 (largest DM format log4). For example, the range for n is 0-4 when the most complex mod format allowed is QAM1024.

At *RST, all Q polarities must be set to NORM.

19.6.12.3 :ICLock NORMallINVerted

SOURCE:DM:POLarity NORMal:ICLock

Selects the polarity of the I channel clock input.

At *RST, ICLock polarity must be set to NORM.

19.6.12.4 :QCLock NORMallINVerted

SOURCE:DM:POLarity NORMal:QCLock

Selects the polarity of the Q channel clock input.

At *RST, QCLock polarity must be set to NORM.

19.6.13 :CLOCK

SOURCE:DM:CLOCK

This subsystem controls the clock for digital modulation.

19.6.13.1 **:SOURce NONE|INTERNAL|EXTERNAL**

SOURce:DM:CLOCK:SOURce

Selects either no clock (asynchronous operation) or external clocking.

At *RST, this value is device-dependent.

19.7 FM Subsystem

SOURce:FM

The Frequency Modulation subsystem is used to set the modulation controls and the parameters associated with the modulating signal(s). The keywords to describe the carrier signal exist in other subsystems (such as FREQuency and PHASe).

KEYWORD	PARAMETER FORM	COMMENTS
:FM		
:COUPling	AC DC GROund	
[:DEViation]	<numeric_value>	
:EXTernal		
:COUPling	AC DC GROund	
:IMPedance	<numeric_value>	
:POLarity	NORMall INVerted	
:INTernal		
:FREQuency	<numeric_value>	
:MODE	LOCKed UNLocked	
:POLarity	NORMall INVerted	
:SENSitivity	<numeric_value>	
:SOURce	EXTernal INTernal{,EXTernal ,INTernal}	
:STATe	<Boolean>	

19.7.1 :COUPling AC|DC|GROund

SOURce:FM:COUPling

COUPling at this level is used to set the coupling between the modulator and the modulating signal. The modulating signal may be the sum of several signals, either as internal or external sources. If COUPling is set to DC, then both AC and DC components of the signal pass. AC coupling passes only the AC component of the signal. The GROund parameter allows the modulation to be turned ON without the modulating signal connected.

At *RST, this value is device-dependent.

19.7.2 [:DEViation] <numeric_value>

SOURce:FM:DEViation

Sets the modulation DEViation of an FM signal. The unit for DEViation is Hertz (Hz). This command is used where the DEViation can be controlled independently from the modulating signal voltage level. Alternatively, it is used to set a device capability that can indicate when the desired DEViation has been achieved.

At *RST, this value is device-dependent.

19.7.3 :EXTernal

SOURce:FMwEXTernal

Controls the specified external signal source. Where multiple external modulation sources are available, the EXTernal keyword may have a numeric suffix signifying the particular external signal source. If no number is given, a 1 is assumed.

19.7.3.1 :COUPling ACIDCIGROund

SOURce:FM:EXTernal:COUPling

COUPling at this level sets the coupling for only the specified external signal source. If COUPling is set to DC, then both AC and DC Modulation components of the signal pass. AC coupling passes only the AC component of the signal. The GROund parameter allows the specified external source to be turned ON without being connected.

At *RST, this value is device-dependent.

19.7.3.2 :IMPedance <numeric_value>

SOURce:FM:EXTernal:IMPedance

Sets the impedance of the specified external signal source. The units for IMPedance is Ohms.

At *RST, this value is device-dependent.

19.7.3.3 :POLarity NORMalINVerted

SOURce:FM:EXTernal:POLarity

POLarity at this level sets the POLarity for only the specified external signal source.

POLarity sets the relationship between the polarity of the applied modulation voltage and the resulting “direction” of modulation. The definition of “normal” polarity for FM modulated signal is a positive voltage which causes the instantaneous output frequency to increase.

At *RST, this value must be set to NORMal on.

19.7.4 :INTERNAL

SOURce:FM:INTERNAL

The INTERNAL subsystem is used to control the specified internal signal source. Where multiple internal modulation sources are available, the INTERNAL keyword may have a numeric suffix signifying the particular internal signal source. If no number is given, a 1 is assumed.

19.7.4.1 :FREQuency <numeric_value>

SOURce:FM:INTERNAL:FREQuency

Sets the frequency of the specified internal signal source. Some instruments may also allow the internal signal source(s) to be controlled as a subsystem(s). The unit of FREQuency is Hertz (Hz).

At *RST, this value is device-dependent.

19.7.5 :MODE LOCKedIUNLocked

SOURce:FM:MODE

Sets the synthesis mode employed in generating the FM signal. If MODE is set to LOCKed then a PLL is used, in synthesizing the FM signal, to prevent frequency drift. If UNLocked the phase locked loop is inactive.

At *RST, this value is device-dependent.

19.7.6 :POLarity NORMal|INVerted

SOURce:FM:POLarity

POLarity, at this level, is used to set the polarity between the modulator and the modulating signal. The modulating signal may be the sum of several signals, either as internal or external sources. POLarity sets the relationship between the polarity of the applied modulation voltage and the resulting “direction” of modulation. The definition of “normal” polarity for FM modulated signal is that a positive voltage causes the instantaneous output frequency to increase.

At *RST, this value must be set to NORMal.

19.7.7 :SENSitivity <numeric_value>

SOURce:FM:SENSitivity

Controls the modulation deviation by setting a sensitivity Modulation to the modulation signal voltage level. The unit for SENSitivity is Hertz/Volt (Hz/V).

At *RST, this value is device-dependent.

19.7.8 :SOURce EXTernal|INTernal{,EXTernal|,INTernal}

SOURce:FM:SOURce

Selects the source for the modulating signal, and it may specify a single source or a number of sources. The specified sources, in the SOURce command, are all selected and turned on. Any sources from a previous selection that are not in the current selection list are deselected and turned off. If multiple sources are selected and the device does not support the combination requested in the SOURce command, an execution error -221 must be generated.

Where multiple internal modulation sources are available, individual INTernal sources must be distinguished by a unique number suffix. If no number suffix is provided in a command, a 1 must be assumed.

Where multiple external modulation sources are available, individual EXTernal sources must be distinguished by a unique number suffix. If no number suffix is provided in a command, a 1 must be assumed.

At *RST, this value is device-dependent.

19.7.9 :STATe <Boolean>

SOURce:FM:STATe

Turns frequency modulation ON or OFF. Turning FM ON will not automatically turn OFF any other types of modulation. Turning any or all modulations types ON or OFF must be done explicitly. Where a type of modulation is active turning ON another type of modulation, will add to the set of active modulations if that combination of capabilities are legal in the device. Otherwise, an execution error -221 is generated, and the instrument state is device-dependent.

At *RST, this value must be set to OFF.

19.8 **FORCe Subsystem**

SOURce:FORCe

This node controls the force of the device.

KEYWORD	PARAMETER FORM	COMMENTS
:FORCe		1999
:CDOWn		1999
:INITiate		[event; no query]1999
:NRUNs	<numeric_value>	1999
:RLDeviatiOn		1999
:FACCeptance	<numeric_value>	1999
:INITiate		[event; no query]1999
:RMAXimum	<numeric_value>	1999
:RVERify	<numeric_value>	1999
:SOFFset	<numeric_value>	1999
:CONFigure		1999
:ABRake		1999
:GAIN	<numeric_value>	1999
[:STATe]	<Boolean>	1999
:THREshold	<numeric_value>	1999
:GRADe		1999
:LEVel	<numeric_value>	1999
:SOURce	INTernallEXTernal	1999
[:STATe]	<Boolean>	1999
[:VEHicle]		1999
:DCOefficient	<numeric_value>{,<numeric_value>}	1999
:DINertia	<numeric_value>	1999
[:STATe]	<Boolean>	1999
:TCOefficient	<numeric_value>{,<numeric_value>}	1999
:TINertia	<numeric_value>	1999
:WEIGHt	<numeric_value>	1999
:INITiate		[event; no query]1999
[:LEVel]	<numeric_value>	1999
:RLSimulation		[event; no query]1999
:INITiate		[event; no query]1999

19.8.1 **:CDOWn**

:SOURce:FORCe:CDOWn

Coast Down

This node contains the commands required to run a Dynamometer Coast Down procedure. A coast down measures the force, power, and elapsed time over multiple speed intervals as a vehicle or dynamometer roll coasts in road load simulation from a starting speed to a final speed. This procedure uses the values set in the :SOURce:FORCe:CONFigure node and the CDownSpeed table. The procedure results are stored in the CDownResult table. Measured coefficients are placed in the memory table CDownMCoeff and are determined after each

coast down run. CDownDCoeff should be calculated after coast down run according to SAE J2264, but do not adjust the dyno-setting coefficients.

19.8.1.1 :INITiate

:SOURce:FORCE:CDown:INITiate

This is an overlapped command. This command initiates the coast down using the current parameters from the :CONFigure node and CDownSpeed Table. This command describes an event and therefore has no associated *RST condition.

When this command is initiated for dynamometers, the dynamometer must:

- Check critical ancillaries, if any, for proper conditions: Brakes Off, Motor Contactor On, parameters within dynamometer safety limits. For conditions that prohibit operation of this procedure, issue appropriate warning messages, and execute the Idle Procedure.
- Set the DYNO operation condition register Coast Down Procedure bit indicating that a Coast down is executing. Note: The DYNO operation condition register will be further defined in Volume 4.
- Enter the theoretical data for the defined speed interval and configured parameters in CDownResult table.
- Accelerate dynamometer to :SOURce:FORCE:CDown:SOFFset +(plus) the starting speed for the first coast down interval, at the rate in :SOURce:ACceleration . If the dynamometer determines this speed is not appropriate for the defined coast down, set the DYNO questionable condition register Coastdown Speed bit and issue an error message, but continue on with the procedure.
- Coast down over the speed intervals defined in the CDownSpeed table. The first interval that has 0 for start and stop speeds ends the measurement. For each interval defined in CDownSpeed, place the measured force, power, and time CDownResult table for the coast down being executed. Remaining entries should be set to 0.
- Perform a curve fit on the CDownResult data and place the coefficients in the memory table CDownMCoeff.
- Repeat the coast down sequence for :SOURce:FORCE:CDown:NRUNs
- After the last measurement, issue the process complete message and wait for a command.
- When commanded out of this procedure, clear the DYNO operation condition register Coastdown Procedure bit indicating that the Coast down is not executing.

19.8.1.2 :SOFFset <numeric_value>

:SOURce:FORCE:CDown:SOFFset

Speed Offset - m/s

This command sets how much speed offset above the top speed in the CDownSpeed Table to which the dynamometer should be accelerated before beginning the coast down. This offset allows the dynamometer response to settle before coasting into the first speed segment.

At *RST, this value is set to 5.

19.8.1.3 :NRUNs <numeric_value>

:SOURce:FORCE:CDOWn:NRUNs

This command sets the number of coast downs to run.

At *RST, this value is set to 3.

19.8.1.4 :RLDerivation

:SOURce:FORCE:CDOWn:RLDerivation

Road Load Derivation

This node contains the commands required to run a Dynamometer Road Load Derivation. This procedure will run a series of coast downs with the vehicle on the dyno to determine a dyno-setting that will match the results from the track. The coast down procedure is defined by the SAE publication J2264 (See Appendix A - References for information on this publication.). Additional input for this procedure comes from the

:SOURce:FORCE:CONFigure node, CDownSpeed table, CDownDCoeff, CDownMCoeff, and CDownResult table. The initial dynamometer coefficients (“best guess”) will be in the first row of CDownDCoeff prior to starting this procedure.

19.8.1.4.1 :FACceptance <numeric_value>

:SOURce:FORCE:CDOWn:RLDerivation:FACceptance

Force Acceptance - N

This command sets the maximum acceptable force difference between the force vs. speed curve derived from the dynamometer target and measured coefficients. This is the maximum difference found at any of the specified coast down speed points.

At *RST, this value is set to 10.

19.8.1.4.2 :INITiate

:SOURce:FORCE:CDOWn:RLDerivation:INITiate

This is an overlapped command. This command initiates the coast down simulation using the current parameters in the :CONFigure node and CDownSpeed Table. This command describes an event and therefore has no associated *RST condition.

When this command is initiated for dynamometers, the dynamometer must:

- Check critical ancillaries, if any, for proper conditions: Brakes Off, Motor Contactor On, parameters within dynamometer safety limits. For conditions that prohibit operation of this procedure, issue appropriate warning messages, and execute the Idle Procedure.
- Set the DYNO operation condition register Road Load Simulation Procedure bit indicating that a Road Load Derivation is executing. Note: The DYNO operation condition register will be further defined in Volume 4.
- Enter the theoretical data for the defined speed interval and configured parameters in CDownResult table based on the target values in :SOURce:FORCE:CONFigure:VEHicle:TCoefficient and :TINertia.
- Accelerate dynamometer to :SOURce:FORCE:CDOWn:SOFFset +(plus) the starting speed for the first coast down interval, at the rate in :SOURce:ACCeleration . If the

1999 SCPI Command Reference

dynamometer determines this speed is not appropriate for the defined coast down, set the DYNO questionable condition register Coast down Trigger Speed bit and issue an error message, but continue on with the procedure.

- Coast down over the speed intervals defined in the CDownSpeed table. The first interval that has 0 for start and stop speeds ends the measurement for this coast down.
- For each coast down, calculate the dyno setting coefficients and measured coefficients. Measured coefficients are placed in CDownMCoeff, and dyno setting coefficients are placed in CDownDCoeff. Also update :SOURCE:FORCE:CONF:VEH:DCoefficient.
- For the present coast down, if the absolute difference between the measured and target force limit is less than the value :SOURCE:FORCE:CDOWN:RLDerivation:FACceptance, begin verification coast downs. If not, adjust the dyno set coefficients as outlined in SAE J2264, and perform another coast down. For verification coast downs, perform up to :SOURCE:FORCE:CDOWN:RLDerivation:RVERification coast downs.
- If the number of coast downs performed is equal to :SOURCE:FORCE:CDOWN:RLDerivation:RMAXimum, or any of the verification runs is not within :SOURCE:FORCE:CDOWN:RLDerivation:FACceptance, the procedure failed to converge. If the procedure fails to converge set the DYNO questionable condition register bit 8 and issue an error message.
- After the last coast down measurement, issue the process complete message. Clear the DYNO operation condition register bit 8 indicating that the Road Load Derivation is not executing.
- Execute the Idle Procedure.

19.8.1.4.3 :RMAXimum <numeric_value>

:SOURCE:FORCE:CDOWN:RLDerivation:RMAXimum

Runs Maximum

This command sets the maximum number of coast down runs the Road Load Derivation test will run without convergence.

At *RST, this value is set to 15.

19.8.1.4.4 :RVERify <numeric_value>

:SOURCE:FORCE:CDOWN:RLDerivation:RVERify

Runs to Verify

This command sets the maximum number of verification coast down runs after convergence of the Road Load Derivation procedure.

At *RST, this value is set to 2.

19.8.2 :CONFigure

:SOURce:FORCe:CONFigure

This node configures the force for various procedures. For dynamometers, these procedures may be Road Load Simulation, Coast downs, etc.

19.8.2.1 :ABRake

:SOURce:FORCe:CONFigure:ABRake

Augmented Brake

This node contains the commands to setup the Augmented Brake. This system utilizes the dynamometer to use regenerative braking of the rolls to simulate the brakes of the non-driven wheels.

19.8.2.1.1 :GAIN <numeric_value>

:SOURce:FORCe:CONFigure:ABRake:GAIN

The augmented brake gain is used to adjust how much simulated braking assistance force is required.

At *RST, this value is set to 0.

19.8.2.1.2 [:STATe] <Boolean>

:SOURce:FORCe:CONFigure:ABRake[:STATe]

This command sets the augmented brake simulation state.

At *RST, this value is set to OFF.

19.8.2.1.3 :THReshold <numeric_value>

:SOURce:FORCe:CONFigure:ABRake:THReshold

This command sets the vehicle force absorption level in N to initiate a simulated braking force for the non-rotating wheel brakes.

At *RST, this value is set to 500.

19.8.2.2 :GRADE

:SOURce:FORCe:CONFigure:GRADE

This node contains the commands to setup the grade input for the Road Load Simulation and [if desired] for Coast downs.

19.8.2.2.1 :LEVel <numeric_value>

:SOURce:FORCe:CONFigure:GRADE:LEVel

This command sets the grade force in units of Percent Grade. For example, a 45 degree slope is a 100% grade. The resulting force is given by:

$$\text{FORCe} = \text{WEIGHt} * \sin(\arctan(\text{GRADE}/100))$$

At *RST, this value is set to 0.

19.8.2.2.2 :SOURce <INTernal|EXTernal>

:SOURce:FORCe:CONFigure:GRADE:SOURce

This command sets the source for grade. The grade set by

:SOURce:FORCe:CONFigure:GRADE:LEVel is used when the source is “INTernal”. The

host computer can vary the grade at any time during the simulation by sending a new LEV command. If set to “EXTernal”, then the simulation will include the grade from an external analog input (Full Scale volts = 100% = 45V). “EXTernal” is used when a dynamometer is equipped with an analog input to control the grade.

At *RST, this value is set to INTernal.

19.8.2.2.3 [:STATe] <Boolean>

:SOURce:FORCe:CONFigure:GRADE[:STATe]

This command sets the grade simulation state. If the state is OFF, the simulation does not include a grade force. If the state is ON, the simulation includes the grade force from an internal or external source.

At *RST, this value is set to OFF.

19.8.2.3 [:VEHicle]

:SOURce:FORCe:CONFigure[:VEHicle]

This node configures the dynamometer vehicle parameters. These parameters are the dynamometer operating set-points [e.g., inertia, road-load, etc.].

19.8.2.3.1 :DCOefficient<numeric_value>,<numeric_value>,<numeric_value>[,<numeric_value>]

:SOURce:FORCe:CONFigure:VEHicle:DCOefficient

Dynamometer-setting Coefficient(s) - N, N/(m/s), N/(m/s)² [,N/(m/s)³]

This command sets the coefficients of the road-load polynomial for the force to be simulated. The query form of this command should return the active dynamometer coefficients, which may have been determined from a Road Load Derivation procedure. When the fourth parameter is omitted, it is assumed 0.

At *RST, this value is set to 0,0,0,0

19.8.2.3.2 :DINertia <numeric_value>

:SOURce:FORCe:CONFigure:VEHicle:DINertia

Dynamometer-setting Inertia - kg

This command sets the “Inertia” (in kg) to be simulated by the dynamometer. This is the total vehicle inertia (vehicle weight plus “inertia weight” of the rotating components) minus the inertia of the driven wheels and drive train (which will be spinning when the vehicle is driven on the dynamometer). When default is specified the dynamometer-setting inertia will be the mechanical base inertia.

At *RST, this value is set to the Mechanical Base Inertia.

19.8.2.3.3 [:STATe] <Boolean>

:SOURce:FORCe:CONFigure:VEHicle[:STATe]

This command tells the dynamometer that a vehicle is on the rolls.

At *RST, this value is set to OFF.

19.8.2.3.4 **TCOEfficient<numeric_value>,<numeric_value>,<numeric_value>[,<numeric_value>]**

:SOURce:FORCe:CONFigure:VEHicle:TCOEfficient

Target-setting Coefficient(s) - N, N/(m/s), N/(m/s)² [,N/(m/s)³]

This command sets the Target Road-Load Coefficients for the road load derivation or coast down procedure. These are derived from test track data for a vehicle. The host will update these values prior to executing a coast down or road load derivation. Target coefficients are never used to control the dynamometer. When the fourth parameter is omitted, it is assumed 0.

At *RST, this value is set to 0,0,0,0

For road load derivation, TCOefficient are used to:

- calculate theoretical data in CDownResult memory table
- compare against measured coast down coefficients as defined in J2264

For coast downs with or without a vehicle, TCOefficient are used to:

- calculate theoretical data in CDownResult memory table

19.8.2.3.5 **:TINertia <numeric_value>**

:SOURce:FORCe:CONFigure:VEHicle:TINertia

Target-setting Inertia - kg

This command sets the target inertia (kg) for a road load derivation procedure. Target inertia is defined in SAE J2264.

At *RST, this value is set to 0.

19.8.2.3.6 **:WEIGHT <numeric_value>**

:SOURce:FORCe:CONFigure:VEHicle:WEIGHT

Weight - kg

This command sets the (kg) weight of vehicle, not including the “inertia weight” of the wheels and drive-train. The value is used to calculate grade and augmented braking weight ratio (when simulating braking force of the non-driven wheels).

At *RST, this value is set to 0.

19.8.3 **:INITiate**

:SOURce:FORCe:INITiate

This command initiates force. This is an overlapped command. This command describes an event and therefore has no associated *RST condition.

When this command is initiated for dynamometers, the dynamometer must:

- Check critical ancillaries, if any, for proper conditions: Brakes Off, Motor Contactor On, parameters within dynamometer safety limits. For conditions that prohibit operation of this procedure, issue appropriate warning messages, and execute the Idle Procedure.

1999 SCPI Command Reference

- Set the DYNO operation condition register Go To Force Procedure bit indicating that Go To Force Procedure is executing. Note: The DYNO operation condition register will be further defined in Volume 4.
- Apply the force set in :SOURce:FORCE:LEVel.
- Maintain the force and wait for a command.
- When commanded out of this procedure, clear the DYNO operation condition register Go To Force Procedure bit indicating the Go To Force Procedure is not executing.
- Issue the process complete message.

19.8.4 **[[:LEVel] <numeric_value>**

:SOURce:FORCE[:LEVel]

Sets the force set-point in N.

At *RST, this value is device dependent.

19.8.5 **:RLSimulation**

:SOURce:FORCE:RLSimulation

Road Load Simulation

This node sets the dynamometer into the procedure to apply a load to simulate vehicle operation on the road. These commands use the values set in the :SOURce:FORCE:CONFigure node.

19.8.5.1 **:INITiate**

:SOURce:FORCE:RLSimulation:INITiate

This is an overlapped command. This command initiates the Road Load Simulation using the present parameters in the :CONFigure node. This command describes an event and therefore has no associated *RST condition.

When this command is initiated for dynamometers, the dynamometer must:

- Check critical ancillaries, if any, for proper conditions: Brakes Off, Motor Contactor On, parameters within dynamometer safety limits. For conditions that prohibit operation of this procedure, issue appropriate warning messages, and execute the Idle Procedure.
- Set the DYNO operation condition register Road Load Simulation Procedure bit indicating that Road Load Simulation is executing. Note: The DYNO operation condition register will be further defined in Volume 4.
- Begin simulating the load defined by :SOURce:FORCE:CONFigure:VEHicle:DINertia;DCOefficient;GRADE;ABRAke
- When commanded out of this procedure, clear the DYNO operation condition register bit 4 indicating that Road Load Simulation is not executing.
- Issue the process complete message.

19.9 FREQUENCY Subsystem

SOURce:FREQuency

The FREQUENCY subsystem controls the frequency characteristics of the instrument. Two types of controls are included in this subsystem. The first is frequency sensor controls, such as RANGE. The second is frequency sweep controls such as START, or tuning controls such as CW.

KEYWORD	PARAMETER FORM	COMMENTS
:FREQuency		
:CENTer	<numeric_value>	
[:CWI:FIXed]	<numeric_value>	
:AUTO	<Boolean> ONCE	
:MANual	<numeric_value>	
:MODE	CWIFIXed SWEeplLISTISENSE	
:MULTiplier	<numeric_value>	
:OFFSet	<numeric_value>	
:RESolution	<numeric_value>	1992
:AUTO	<Boolean> ONCE	1992
:SPAN	<numeric_value>	
:FULL		[no query]
:HOLD	<Boolean>	
:LINK	CENTER START STOP	
:START	<numeric_value>	
:STOP	<numeric_value>	

19.9.1 :CENTER <numeric_value>

SOURce:FREQuency:CENTer

Sets the CENTer frequency.

At *RST, this function will be set to (MINimum+MAX)/2.

19.9.2 [:CWI:FIXed] <numeric_value>

SOURce:FREQuency:CW

The Continuous Wave or FIXed node is used to select a frequency of a non-swept signal. This node is optional. An optional node, such as [:CWI:FIXed], implies that the device shall accept and process commands with or without the optional node and have the same result. That is, the device is required to accept the optional node, if sent, without error. A device must accept both :CW and :FIXed if it implements this node.

At *RST, this value is device-dependent.

19.9.2.1 :AUTO <Boolean>|ONCE

SOURce:FREQuency:CW:AUTO

Couples the CW frequency to center frequency. Explicitly setting a value for FREQuency:CW sets AUTO OFF.

At *RST, this value is set to ON.

19.9.3 :MANual <numeric_value>

SOURce:FREQuency:MANual

Same as CW except limited by START and STOP. Receipt of this command will only change the frequency if SWEep:MODE MANual subsystem and FREQuency:MODE SWEep is selected. If the sweep limits are changed such that manual frequency would be outside the sweep limits, then the manual frequency will be set to the nearest end point of the sweep.

At *RST, FREQuency:MAN is device-dependent.

19.9.4 :MODE CWIFIXedISWEepILISTISENSE

SOURce:FREQuency:MODE

Determines which set of commands control the frequency subsystem. The settings have the following meanings:

- CWIFIXed: The frequency is determined by FREQuency[:CW] or FREQuency[:FIXed]. CW and FIXed are aliases. Both must be implemented, if either is implemented.
- SWEep: The source is in the swept mode and frequency is determined by FREQuency:START, STOP, CENT, SPAN, and MAN.
- LIST: The source is in LIST sequence mode and frequency is determined by LIST:FREQuency.
- SENSE: The source is coupled to the sensor. All frequency controls are determined by the SENSE:FREQuency block, except for OFFSET and MULTIplier. These two commands remain active under SOURce to allow for offset and harmonic tracking.

At *RST, this value is set to CW.

19.9.5 :MULTIplier <numeric_value>

SOURce:FREQuency:MULTIplier

Sets a reference multiplier for all other frequency settings in the instrument. All entered and displayed frequencies are affected by this command. The coupling equation is:

$$\text{Entered/Displayed frequency} = (\text{Hardware frequency} * \text{multiplier}) + \text{offset}$$

This command affects the values of all functions which affect generation of the signal. This includes such things as CW frequency and the sweep frequency controls. It does not affect things like input filter bandwidths. It does affect the settings of relative frequency parameters such as delta markers or FM deviation. Also note that this command does not affect the hardware settings of the instrument, but only the entered and displayed frequencies.

At *RST, this value is set to 1.

19.9.6 :OFFSet <numeric_value>

SOURce:FREQuency:OFFSet

Sets a reference frequency for all other absolute frequency settings in the instrument. This command affects the values of all functions which affect generation of the signal. This includes such things as CW frequency and the sweep frequency controls. It does not affect

1999 SCPI Command Reference

things like input filter bandwidths. It also does not affect the settings of relative frequency parameters such as delta markers or FM deviation. Also note that this command does not affect the hardware settings of the instrument, but only the entered and displayed frequencies. The coupling equation is:

$$\text{Entered/Displayed frequency} = (\text{Hardware frequency} * \text{multiplier}) + \text{offset}$$

At *RST, this value is set to 0.

19.9.7 :RESolution <numeric_value>

SOURce:FREQuency:RESolution

RESolution sets or queries the absolute increment/decrement, at which the frequency of the output signal can be changed. The default unit for <numeric_value> is Hertz.

At *RST, this value is device-dependent.

19.9.7.1 AUTO <Boolean> | ONCE

SOURce:FREQuency:RESolution:AUTO

AUTO couples the RESolution to an instrument-determined value. When AUTO is ON, the best resolution is chosen to reflect the current instrument state.

Explicitly selecting a value for RESolution will set AUTO OFF.

At *RST, this value is set to ON.

19.9.8 :SPAN <numeric_value>

SOURce:FREQuency:SPAN

Sets the frequency SPAN.

At *RST, this function will be set to MAXimum.

19.9.8.1 :FULL

SOURce:FREQuency:SPAN:FULL

When this command is received, START frequency is set to its minimum value, and STOP frequency is set to its maximum value. CENTER frequency and SPAN are set to their coupled values.

This command is an event, rather than a state. Therefore it has no associated query, or meaning at *RST.

19.9.8.2 :HOLD <Boolean>

SOURce:FREQuency:SPAN:HOLD

The HOLD command provides a mechanism to prevent the SPAN from being changed implicitly by the defined coupling between START, STOP, CENTER, and SPAN. When HOLD is set to ON, SPAN shall only be changed by explicitly sending the SPAN command.

At *RST, this value is set to OFF.

19.9.8.3 :LINK CENTER|START|STOP

SOURce:FREQuency:SPAN:LINK

This command allows the default couplings for SPAN to be overridden. LINK selects the parameter, either CENTER subsystem, START or STOP, that shall not be changed when

1999 SCPI Command Reference

SPAN's value is changed. For example, if LINK is set to START then changing SPAN shall cause CENTER and STOP, not START to change.

At *RST, this value is set to CENTER, to be compatible with the default definition for the couplings.

19.9.9 :START <numeric_value>

SOURce:FREQuency:START

Sets the STARTing frequency. It also sets the lower limit of a manual sweep adjustment. If the instrument does not support negative frequency spans, having the START frequency greater than the STOP frequency will cause an error to be generated.

At *RST, this value will be set to MINimum.

19.9.10 :STOP <numeric_value>

SOURce:FREQuency:STOP

Sets the STOP frequency. It also sets the upper limit of a manual sweep adjustment. If the instrument does not support negative frequency spans, having the STOP frequency less than the START frequency will cause an error to be generated.

At *RST, this value will be set to MAXimum.

19.10 FUNCTION Subsystem

SOURce:FUNCTION

The function subsystem controls the shape and attributes of the output signal. The switch settings provided by this function are not horizontally compatible and represent what the source can be configured to generate directly.

KEYWORD	PARAMETER FORM	COMMENTS
:FUNCTION		
[:SHAPE]	<source_shape>	
:MODE	<source_mode>	

19.10.1 [:SHAPE] <source_shape>

SOURce:FUNCTION:SHAPE

Selects the shape of the output signal.

At *RST, the value of FUNCTION is device-dependent.

The following SOURce functions are available for instruments:

- DC — An unvarying signal with respect to time.
- IMPulse — IMPulse is an approximation of a zero width pulse
- PCHirp — PeriodicCHirp supplies a sine sweep over the current frequency span which periodically repeats.
- PRNoise — PeriodicRandomNoise provides a randomly varying signal with respect to time which periodically repeats.
- PULSe — A pulse, or string of pulses is generated.
- RANDom — provides a randomly varying signal with respect to time.
- SINusoid — A sinusoidal signal is generated.
- SQUare — A square wave signal is generated.
- TRIangle — A triangular wave signal is generated.
- USER — The USER function allows the user to specify a periodic or arbitrary waveform definition.

19.10.2 :MODE <source_mode>

SOURce:FUNCTION:MODE

Determines which signal characteristic is being controlled.

At *RST, the value of this function is device-dependent.

The function may be set to the following values:

- VOLTage.
- CURRent.
- POWer.

19.11 LIST Subsystem

SOURce:LIST

The list subsystem controls automatic sequencing through associated lists of specified signal values. It is also used to specify concurrent or simultaneous operation of the specified signal in the associated lists. The various values are specified in the LIST subsystem commands. The individual points in each of the list configuration commands are combined to make one signal configuration.

The LIST subsystem controls a characteristic of the instrument when that subsystem's mode is set to LIST (such as :FREQuency:MODE LIST). Note that is permissible to have one characteristic controlled by the list subsystem, and the others controlled by a different subsystem (such as :FREQuency:MODE LIST or :POWer:MODE SWEEp).

When employing multiple lists, all lists must be of the same length. If they are not all of the same length, an error is generated at sequencing time. An exception is made for the case where the shorter list is of length 1. In this case, the particular list is treated as though it were a list of equal length, with all values being the same.

Lists are not affected by *RST.

KEYWORD	PARAMETER FORM	COMMENTS
:LIST		
:AM		1993
[:DEPTh]	<numeric_value>{,<numeric_value>}	1993
:POINts?		[query only] 1993
:APRObe	<numeric_list>{,<numeric_list>}	1993
:POINts?		[query only] 1993
:CONCurent	<numeric_value>{,<numeric_value>}	
:AUTO	<Boolean> ONCE	
:POINts?		[query only]
:CONTRol		1993
:APOWer	<Boolean>{,<Boolean>}	1993
:POINts?		[query only] 1993
:BLOWer	<Boolean>{,<Boolean>}	1993
:POINts?		[query only] 1993
:COMPressor	<Boolean>{,<Boolean>}	1993
:POINts?		[query only] 1993
:COUNT	<numeric_value>	
:CURRent	<numeric_value>{,<numeric_value>}	
:POINts?		[query only]
:DIRection	UPIDOWN	
:DWELl	<numeric_value>{,<numeric_value>}	
:POINts?		[query only]
:FREQuency	<numeric_value>{,<numeric_value>}	
:POINts?		[query only]
:GENeration	DSEQUence SEQUence DCONcurrent CONCurent	

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	COMMENTS
:POWer	<numeric_value>{,<numeric_value>}	
:POINts?		[query only]
:PULM		1993
[:STATe]	<Boolean>{,<Boolean>}	1993
:POINts?		[query only] 1993
:RESistance	<numeric_value>{,<numeric_value>}	
:POINts?		[query only]
:RTIME	<numeric_value>{,<numeric_value>}	1993
:POINts?		[query only] 1993
:SEQuence	<numeric_value>{,<numeric_value>}	
:AUTO	<Boolean> ONCE	
:POINts?		[query only]
:TEMPerature	<numeric_value>{,<numeric_value>}	1993
:POINts?		[query only] 1993
:VOLTage	<numeric_value>{,<numeric_value>}	
:POINts?		[query only]

19.11.1 :AM

SOURce:LIST:AM

Subsystem for setting up the AM list values.

19.11.1.1 :DEPTH <numeric_value>{,<numeric_value>}

SOURce:LIST:AM:DEPTH

Specifies the AM DEPTH points of the list.

19.11.1.1.1 :POINts?

SOURce:LIST:AM:DEPTH:POINts?

Returns the number of points currently in the AM DEPTH list.

19.11.2 :APRrobe <numeric_list>{,<numeric_list>}

SOURce:LIST:APRrobe

APRrobe selects the specified probe/s to control the temperature.

19.11.2.1 :POINts?

SOURce:LIST:APRrobe:POINts?

Returns the number of points currently in the APRrobe list.

19.11.3 :CONCurent <numeric_value>{,<numeric_value>}

SOURce:LIST:CONCurent

Defines a list which indicated those elements of the signal list which shall be active, when CONCurent operation is selected. The points specified are indexes into the lists. For example, if 3 and 5 were selected, the third and fifth points in the signal list would be generated simultaneously. All entries in the CONCurent list must be unique; otherwise a parameter error (-220) will be generated. The concurrent list is separate from the other lists and is not sorted, nor are the individual points associated with the other arrays except as described above.

19.11.3.1 :**AUTO** <Boolean>|ONCE

SOURce:LIST:CONCurent:AUTO

When on, the CONCurent list is set to 1-N, where N is the longest list.

At *RST, AUTO ON is selected.

19.11.3.2 :**POINTs?**

SOURce:LIST:CONCurent:POINTs?

Returns the number of points currently in the CONCurent list.

19.11.4 :**CONTRol**

SOURce:LIST:CONTRol

Allows to turn on/off the devices

19.11.4.1 :**APOWER** <Boolean>{,<Boolean>}

SOURce:LIST:CONTRol:APOWER

Turns the APOWER on or off.

19.11.4.1.1 :**POINTs?**

SOURce:LIST:CONTRol:APOWER:POINTs?

Returns the number of points currently in the APOWER list.

19.11.4.2 :**BLOWer** <Boolean>{,<Boolean>}

SOURce:LIST:CONTRol:BLOWer

Turns the BLOWer on or off.

19.11.4.2.1 :**POINTs?**

SOURce:LIST:CONTRol:BLOWer:POINTs?

Returns the number of points currently in the BLOWer list.

19.11.4.3 :**COMPressor** <Boolean>{,<Boolean>}

SOURce:LIST:CONTRol:COMPressor

Turns the COMPressor on or off.

19.11.4.3.1 :**POINTs?**

SOURce:LIST:CONTRol:COMPressor:POINTs?

Returns the number of points currently in the COMPressor list.

19.11.5 :**COUNT** <numeric_value>

SOURce:LIST:COUNt

Controls the number of times the list is sequenced when a trigger is received.

At *RST, this value is set to 1.

19.11.6 :**CURRent** <numeric_value>{,<numeric_value>}

SOURce:LIST:CURRent

Specifies the current points of the lists.

19.11.6.1 :**POINTs?**

SOURce:LIST:CURRent:POINTs?

Returns the number of points currently in the CURRent list.

19.11.7 :DIRection UPIDOWN

SOURce:LIST:DIRection

Specifies the direction that the sequence list is scanned. If UP is selected, the list is scanned from the first to the last item in the sequence list. If DOWN is selected, the list is scanned from the last item to the first item in the sequence list.

At *RST, this value is set to UP.

19.11.8 :DWELl<numeric_value>{,<numeric_value>}

SOURce:LIST:DWELl

Specifies the dwell time points of the lists.

19.11.8.1 :POINTs?

SOURce:LIST:DWELl:POINTs?

Returns the number of points currently in the DWELl list.

19.11.9 :FREQuency<numeric_value>{,<numeric_value>}

SOURce:LIST:FREQuency

Specifies the frequency points of the list set.

19.11.9.1 :POINTs?

SOURce:LIST:FREQuency:POINTs?

Returns the number of points currently in the FREQuency list.

Since this is only a query, there is no associated *RST state.

19.11.10 :GENeration DSEQuence|SEQuence|CONcurrent|CONCurrent

SOURce:LIST:GENeration

The GENeration command is used to select how the defined lists are applied in a particular instrument. An instrument may support one or more of the generation modes. Selecting Default SEQuence (DSEQuence) causes the instrument to cycle sequentially through the complete list in order. When SEQuence is selected, the sequence in which the list is used is specified by the list given in the SEQuence command. Selecting Default CONCurrent causes the device to simultaneously generate all items in the list. When CONCurrent is selected, the list specified by the CONCurrent command defines which elements in the list are active simultaneously.

At *RST, GENeration shall be set to DSEQuence.

19.11.11 :PULM

SOURce:LIST:PULM

Subsystem for setting up the PULM list values.

19.11.11.1 :STATe <Boolean>{,<Boolean>}

SOURce:LIST:PULM:STATe

Specifies the PULM STATe points of the list.

19.11.11.1.1 :POINTs?

SOURce:LIST:PULM:STATe:POINTs?

Returns the number of points currently in the PULM STATe list.

19.11.12 :POWER <numeric_value>{,<numeric_value>}

SOURce:LIST:POWer

Specifies the power/amplitude points of the lists.

19.11.12.1 :POINTS?

SOURce:LIST:POWer:POINts?

Returns the number of points currently in the POWer list.

19.11.13 :RESistance <numeric_value>{,<numeric_value>}

SOURce:LIST:RESistance

Specifies the resistance points of the lists.

19.11.13.1 :POINTS?

SOURce:LIST:RESistance:POINts?

Returns the number of points currently in the RESistance list.

19.11.14 :RTIME <numeric_value>{,<numeric_value>}

SOURce:LIST:RTIME

Ramp TIME controls the time required to reach the target temperature.

19.11.14.1 :POINTS?

SOURce:LIST:RTIME:POINts?

Returns the number of points currently in the RTIME list.

19.11.15 :SEquence <numeric_value>{,<numeric_value>}

SOURce:LIST:SEquence

Defines a sequence for stepping through the list, when SEquence operation is selected. Individual points may be specified as many times as desired in a single sequence. The points specified are indexes into the lists. For example, if 3 was selected, the third point in the frequency, dwell, and power/amplitude lists would be sequenced.

The sequence list is separate from the other lists and is not sorted, nor are the individual points associated with the other arrays except as described above.

19.11.15.1 :AUTO <Boolean>IONCE

SOURce:LIST:SEquence:AUTO

When on, the sequence list is set to 1 through N, where N is the longest list.

At *RST, AUTO ON is selected.

19.11.15.2 :POINTS?

SOURce:LIST:SEquence:POINts?

Returns the number of points currently in the SEquence list.

19.11.16 :TEMPerature <numeric_value>{,<numeric_value>}

SOURce:LIST:TEMPerature

TEMPerature is the target temperature.

19.11.16.1 :POINTS?

SOURce:LIST:TEMPerature:POINts?

Returns the number of points currently in the TEMPerature list.

19.11.17 **:VOLTage<numeric_value>{,<numeric_value>}**

SOURce:LIST:VOLTage

Specifies the voltage points of the lists.

19.11.17.1 **:POINTS?**

SOURce:LIST:VOLTage:POINTS?

Returns the number of points currently in the VOLT list.

19.12 MARKer Subsystem

SOURce:MARKer

Selects between different markers, and adjusts the marker settings. The suffix with MARKer selects the marker with the same number associated with it. If the suffix is omitted, then marker #1 is assumed.

KEYWORD	PARAMETER FORM	COMMENTS
:MARKer		
:AMPLitude	ON OFF	
:AOFF		[no query]
:FREQuency	<numeric_value>	
:MODE	FREQuency POSition DELTA	
:POINT	<numeric_value>	
:REFerence	<numeric_value>	
[:STATe]	<Boolean>	
:TIME	<numeric_value>	

19.12.1 :AMPLitude <Boolean>

SOURce:MARKer:AMPLitude

Controls whether the marker affects the signal. If ON, the signal is modified when the marker frequency equals the output frequency. If OFF, only the marker output port is affected.

At *RST, this value is set to OFF.

19.12.2 :AOFF

SOURce:MARKer:AOFF

Turns OFF all markers. This command is an event, and therefore has no query form. All markers shall be turned OFF at *RST.

19.12.3 :FREQuency <numeric_value>

SOURce:MARKer:FREQuency

Controls the absolute frequency of the specified marker when MARKer:MODE is FREQuency. When MARKer:MODE is DELTA, this command controls relative frequency.

At *RST, this value is set to MINimum.

19.12.4 :MODE FREQuency|POSition|DELTA

SOURce:MARKer:MODE

This command controls whether the marker is tied to a frequency, referenced to an absolute trace point (position), or referenced to another marker (delta). If delta is selected, MARKer:REFerence determines which marker the current marker is referenced to.

At *RST, this value is FREQuency.

19.12.5 :POINT <numeric_value>

SOURce:MARKer:POINT

Sets the marker to the specified sweep point.

At *RST, this value is set to MINimum.

1999 SCPI Command Reference

19.12.6 **:REFeRence <numeric_value>**

SOURce:MARKer:REFeRence

MARK:REFeRence establishes a reference marker for delta markers.

At *RST, this value is device-dependent.

19.12.7 **[[:STATe]] <Boolean>**

SOURce:MARKer:STATe

Turns ON and OFF the marker specified by the MARKer command header suffix.

At *RST, this value is set to OFF.

19.13 PHASe Subsystem

SOURce:PHASe

This subsystem allows control of the phase of the output signal against some reference.

KEYWORD	PARAMETER FORM	COMMENTS
:PHASe		
[:ADJust]	<numeric_value>	
:STEP	<numeric_value>	
:SOURce	INTernal[<n>] EXTernal[<n>]	
:REFerence		[no query]

19.13.1 [:ADJust] <numeric_value>

SOURce:PHASe:ADJust

Controls the phase offset value relative to the reference. A negative value for ADJust shall cause the output signal to lag the reference.

The parameter has units of radians. The command shall accept a DEGREE suffix.

At *RST, this value is 0.

19.13.1.1 :STEP <numeric_value>

SOURce:PHASe:ADJust:STEP

Controls the step size in radians. The command shall accept a DEGREE suffix.

At *RST, this value is device-dependent.

19.13.2 :SOURce INTernal|EXTernal

SOURce:PHASe:SOURce

Determines whether the reference source is an external signal or if it is the internal signal itself marked at some reference time.

At *RST, this setting is device-dependent.

19.13.3 :REFerence

SOURce:PHASe:REFerence

This is an event which sets the current phase to be the reference for future phase adjustments. This function is nonqueryable.

19.14 PM Subsystem

SOURce:PM

The Phase Modulation subsystem is used to set the modulation controls and the parameters associated with the modulating signal(s). The keywords to describe the carrier signal exist in other subsystems (such as FREQuency and PHASe).

KEYWORD	PARAMETER FORM	COMMENTS
:PM		
[:DEViation]	<numeric_value>	
:SENSitivity	<numeric_value>	
:MODE	LOCKed UNLocked	
:STATE	<Boolean>	
:SOURce	EXTernal INTernal{,EXTernal,INTernal}	
:COUPling	AC DC GROund	
:POLarity	NORMal INVerted	
:INTernal		
:FREQuency	<numeric_value>	
:EXTernal		
:IMPedance	<numeric_value>	
:COUPling	AC DC GROund	
:POLarity	NORMal INVerted	

19.14.1 [:DEViation] <numeric_value>

SOURce:PM:DEViation

Sets the modulation DEViation of a PM signal. The unit for DEViation is radians. Where PM:DEViation is implemented, degree units must also be accepted. Radians must always be returned on a query. This command is used where the DEViation can be controlled independently from the modulating signal voltage level. Alternatively, it is used to set a device capability that can indicate when the desired DEViation has been achieved.

At *RST, this value is device-dependent.

19.14.2 :SENSitivity <numeric_value>

SOURce:PM:SENSitivity

Controls the modulation deviation by setting a sensitivity Modulation to the modulation signal voltage level. The unit for SENSitivity is radians/Volt (radians/V). Where PM:SENSitivity is implemented, degree/Volt (degree/V) units must also be accepted. Radians/Volt must always be returned on a query.

At *RST, this value is device-dependent.

19.14.3 :MODE LOCKed|UNLocked

SOURce:PM:MODE

Sets the synthesis mode employed in generating the PM signal. If MODE is set to LOCKed then a PLL is used in synthesizing the PM signal, to prevent frequency drift. If UNLocked the phase locked loop is inactive.

At *RST, this value is device-dependent.

19.14.4 :STATe <Boolean>

SOURce:PM:STATe

Turns phase modulation ON or OFF. Turning PM ON will not automatically turn OFF any other types of modulation. Turning any or all modulations types ON or OFF must be done explicitly. Where a type of modulation is active, turning ON another type of modulation will add to the set of active modulations if that combination of capabilities are legal in the device. Otherwise, an execution error -221 is generated and the instrument state is left undefined.

At *RST, this value must be set to OFF.

19.14.5 :SOURceEXTernal|INTERNAL{,EXTernal|,INTERNAL}

SOURce:PM:SOURce

Selects the source for the modulating signal, and it may specify a single source or a number of sources. The specified sources in the SOURce command are all selected and turned on. Any sources from a previous selection that are not part of the current selection list are deselected and turned off. If multiple sources are selected and the device does not support the combination requested in the SOURce command, an execution error -221 must be generated.

Where multiple internal modulation sources are available, individual INTERNAL sources must be distinguished by a unique number suffix. If no number suffix is provided in a command, a 1 must be assumed.

Where multiple external modulation sources are available, individual EXTERNAL sources must be distinguished by a unique number suffix. If no number suffix is provided in a command, a 1 must be assumed.

At *RST, this value is device-dependent.

19.14.6 :COUPling AC|DC|GROund

SOURce:PM:COUPling

COUPling at this level is used to set the coupling between the modulator and the modulating signal. The modulating signal may be the sum of several signals, either internal or external sources. If COUPling is set to DC, then both AC and DC components of the signal pass. AC coupling passes only the AC component of the signal. The GROund parameter allows the modulation to be turned ON without the modulating signal connected.

At *RST, this value is device-dependent.

19.14.7 :POLarity NORMal|INVerted

SOURce:PM:POLarity

POLarity, at this level, is used to set the polarity between the modulator and the modulating signal. The modulating signal may be the sum of several signals, either internal or external sources. POLarity sets the relationship between the polarity of the applied modulation voltage and the resulting “direction” of modulation. The definition of “normal” polarity for PM modulated signal is that a positive voltage causes the instantaneous output phase to increase.

At *RST, this value must be set to NORMal.

19.14.8 :INternal

SOURce:PM:INternal

The INternal subsystem is used to control the specified internal signal source. Where multiple internal modulation sources are available, the INternal keyword may have a numeric suffix signifying the particular internal signal source. If no number is given, a 1 is assumed.

19.14.8.1 :FREquency <numeric_value>

SOURce:PM:INternal:FREquency

Sets the frequency of the specified internal signal source. Some instruments may also allow the internal signal source(s) to be controlled as a subsystem(s). The unit of FREquency is Hertz (Hz).

At *RST, this value is device-dependent.

19.14.9 :EXternal

SOURce:PM:EXternal

The EXternal subsystem is used to control the specified external signal source. Where multiple external modulation sources are available, the EXternal keyword may have a numeric suffix signifying the particular external signal source. If no number is given, a 1 is assumed.

19.14.9.1 :IMPedance <numeric_value>

SOURce:PM:EXternal:IMPedance

Sets the impedance of the specified external signal source. The units for IMPedance is Ohms.

At *RST, this value is device-dependent.

19.14.9.2 :COUPling ACIDCIGROund

SOURce:PM:EXternal:COUPling

COUPling, at this level, sets the coupling for only the specified external signal source. If COUPling is set to DC, then both AC and DC components of the signal pass. AC coupling passes only the AC component of the signal. The GROund parameter allows the specified external source to be turned ON without being connected.

At *RST, this value is device-dependent.

19.14.9.3 :POLarity NORMallINVerted

SOURce:PM:EXternal:POLarity

POLarity, at this level, sets the POLarity for only the specified external signal source. POLarity sets the relationship between the polarity of the applied modulation voltage and the resulting “direction” of modulation. The definition of “normal” polarity for PM modulated signal is a positive voltage that causes the instantaneous output phase to increase.

At *RST, this value must be set to NORMal.

19.15 **Power Subsystem**

SOURce:POWer

This subsection controls the signal amplitude characteristics of the source.

There is only one subsystem for each of the main amplitude characteristics of the signal. Thus, AC voltage and DC voltage are both placed under the VOLTage subsystem. A device which implemented both simultaneously would implement them as two separate sources, one AC and one DC.

KEYWORD	PARAMETER FORM	COMMENTS
:POWer		
:ATTenuation	<numeric_value>	
:AUTO	<Boolean> ONCE	
:ALC		
[:STATe]	<Boolean>	
:SEARch	<Boolean> ONCE	
:SOURce	INTernal DIODE PMETer MMHead	
:BANDwidth	<numeric_value>	
!BWIDth		
:AUTO	<Boolean> ONCE	
:CENTer	<numeric_value>	
[:LEVel]		
[:IMMediate]		
[:AMPLitude]	<numeric_value>	
:OFFSet	<numeric_value>	
:HIGH	<numeric_value>	
:LOW	<numeric_value>	
:TRIGgered		
[:AMPLitude]	<numeric_value>	
:OFFSet	<numeric_value>	
:HIGH	<numeric_value>	
:LOW	<numeric_value>	
:LIMit		
[:AMPLitude]	<numeric_value>	
:OFFSet	<numeric_value>	
:HIGH	<numeric_value>	
:LOW	<numeric_value>	
:STATe	<Boolean>	
:MANual	<numeric_value>	
:MODE	FIXed SWEep LIST	
:PROTecton		
[:LEVel]	<numeric_value>	
:STATe		
:TRIPped?		[query only]
:CLEar		[no query]

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	COMMENTS
:RANGe	<numeric_value>	
:AUTo	<Boolean> ONCE	
:REFerence	<numeric_value>	
:STATe	<Boolean>	
:SLEW	<numeric_value>	
:SPAN	<numeric_value>	
:HOLD	<Boolean>	
:LiNK	CENTer STARt STOP	
:FULL		[no query]
:STARt	<numeric_value>	
:STOP	<numeric_value>	

19.15.1 :ATTenuation <numeric_value>

SOURce:POWer:ATTenuation

Sets the ATTenuation level. Note that when increasing the level by 10 dB the magnitude of the outgoing signal as well as the LEVel will be decreased by 10 dB.

Default units are as determined in the UNIT system. This is coupled to LEVel.

19.15.1.1 :AUTo <Boolean>

SOURce:POWer:ATTenuation:AUTO

Couples the attenuator to LEVel.

Programming a specified attenuation sets AUTO OFF.

See SOURce subsystem introduction for additional explanation.

At *RST, AUTO is set to ON.

19.15.2 :ALC

SOURce:POWer:ALC

This subsystem command controls the automatic leveling control of the source.

19.15.2.1 [:STATe] <Boolean>

SOURce:POWer:ALC:STATe

Controls whether the ALC loop controls the output level.

At *RST, this value is device-dependent.

19.15.2.2 :SEARCh <Boolean>|ONCE

SOURce:POWer:ALC:SEARCh

SEARCh enables a form of leveling where the output level is calibrated by momentarily closing the leveling loop. Once the correct amount of level adjustment has been determined to produce the desired output level, the leveling loop is opened.

When ON, the search routine will be done anytime the output level requires change (for example, when LEVel is changed).

1999 SCPI Command Reference

Selecting SEARCH ONCE will have the effect of setting SEARCH to ON and then OFF. A modulator setting is determined by the system which will not be changed until the output level is explicitly reprogrammed.

At *RST, this value is set to OFF.

19.15.2.3 :SOURce INTERNALDIODEIPMETERIMMHead

SOURce:POWer:ALC:SOURce

Selects the source of the feedback signal for ALC. The parameters have the following meanings:

- INTERNAL — The ALC feedback signal is measured from a point inside the source.
- DIODE — The ALC feedback is being fed from an external diode detector.
- PMETER — The ALC signal is coming from a power meter.
- MMHead — A millimeter head, with built in leveling, is being used to supply the ALC signal.

At *RST, this value is INTERNAL.

19.15.2.4 :BANDwidth:BWIDTH <numeric_value>

SOURce:POWer:ALC:BANDwidth

Controls the bandwidth of the ALC feedback signal. This parameter is in units of Hz.

At *RST, this value is device-dependent.

19.15.2.4.1 :AUTO <Boolean>IONCE

SOURce:POWer:ALC:BANDwidth:AUTO

Couples the bandwidth of the ALC feedback signal to instrument-dependent parameters. When AUTO is ON, the best bandwidth is chosen to reflect the current instrument state.

Explicitly selecting a value for ALC:BANDwidth sets AUTO OFF.

At *RST, this value is set to ON.

19.15.3 :CENTER <numeric_value>

SOURce:POWer:CENTer

Sets the center amplitude.

At *RST, this value is set to MINimum.

19.15.4 [:LEVel]

SOURce:POWer:LEVel

This subsystem is used to control the signal amplitude when the source is operating in a continuous or FIXED MODE.

19.15.4.1 [:IMMediate]

SOURce:POWer:LEVel:IMMediate

Indicates that the subsequent specification of a new signal amplitude is to be processed by the device without waiting for further commands.

19.15.4.1.1 **[[:AMPLitude] <numeric_value>**

SOURce:POWER:LEVel:IMMediate:AMPLitude

Sets the actual magnitude of the unswept output signal in terms of the current operating units. The units are set to the default value, or alternately to a different value under the UNIT subsystem. AMPLitude may be used to specify the level for either a time varying or non-time varying signal.

Note the optional nodes exist to enable POWER to be directly followed by a <numeric_value>. This is useful for programming simple devices in a straightforward manner.

At *RST, the signal being sourced should be set to a “safe” condition. This is generally achieved by setting the amplitude to its minimum value in conjunction with OUTPUT:STATe subsystem.

19.15.4.1.2 **:OFFSet <numeric_value>**

SOURce:POWER:LEVel:IMMediate:OFFSet

Sets the non-time varying component of the signal that is added to the time varying signal specified in AMPLitude, in terms of the current operating units. The units are set to the default value, or alternately to a different value under the UNIT subsystem. When SOURce:FUNCTION is DC, the effect of the OFFSet is device-dependent.

At *RST, the signal being sourced should be set to a “safe” condition. This is generally achieved by setting the offset to its minimum value in conjunction with OUTPUT:STATe subsystem.

19.15.4.1.3 **:HIGH <numeric_value>**

SOURce:POWER:LEVel:IMMediate:HIGH

Sets the more positive peak of a time varying signal. It is used in conjunction with LOW.

At *RST, the signal being sourced should be set to a “safe” condition. This is achieved by setting this to a minimum value in conjunction with OUTPUT:STATe subsystem.

19.15.4.1.4 **:LOW <numeric_value>**

SOURce:POWER:LEVel:IMMediate:LOW

Sets the more negative peak of a time varying signal, it is used in conjunction with HIGH.

At *RST, the signal being sourced should be set to a “safe” condition. This is achieved by setting this to a minimum value in conjunction with OUTPUT:STATe subsystem.

19.15.4.2 **:TRIGgered**

SOURce:POWER:LEVel:TRIGgered

Indicates that subsequent specification of a new signal amplitude is to be transferred to the IMMediate value upon receipt of a trigger signal.

At *RST, after the receipt of an ABORT command or upon being triggered, the value in the TRIGgered subsystem tracks the values in the IMMediate subsystem until a command in the TRIGgered subsystem is received. The purpose of tracking is so that spurious triggers do not change the value of LEVel unexpectedly.

19.15.4.2.1 **[[:AMPLitude] <numeric_value>**

SOURce:POWer:LEVel:TRIGgered:AMPLitude

Sets the actual magnitude of the unswept output signal in terms of the current operating units. The units are set to the default value, or alternately to a different value under the UNIT subsystem. AMPLitude may be used to specify the level for either a time varying or non-time varying signal.

19.15.4.2.2 **:OFFSet <numeric_value>**

SOURce:POWer:LEVel:TRIGgered:OFFSet

Sets the non-time varying component of the signal that is added to the time varying signal specified in AMPLitude. The units are set to the default value, or to a different value under the UNIT subsystem. When SOURce:FUNCTion is DC, the effect of OFFSet is device-dependent.

19.15.4.2.3 **:HIGH <numeric_value>**

SOURce:POWer:LEVel:TRIGgered:HIGH

This command is used to set the more positive peak of a time varying signal. It is used in conjunction with LOW.

19.15.4.2.4 **:LOW <numeric_value>**

SOURce:POWer:LEVel:TRIGgered:LOW

This command is used to set the more negative peak of a time varying signal. It is used in conjunction with HIGH.

19.15.5 **:LIMit**

SOURce:POWer:LIMit

Sets the maximum bounds on the output value. Setting a larger value will cause the output level to be clamped to the LIMit value.

19.15.5.1 **[[:AMPLitude] <numeric_value>**

SOURce:POWer:LIMit:AMPLitude

Sets the limit on the actual magnitude of the unswept output signal in terms of the current operating units. The units are set to the default value, or alternately to a different value under the UNIT subsystem. AMPLitude may be used to specify the level for either a time varying or non-time varying signal. OFFSet may only be used with AMPLitude, where AMPLitude is used to specify a time varying load.

Explicit reference to the AMPLitude node is optional. Any device that implements this subsystem must accept and process commands, with or without the AMPLitude node, and respond to them in the same way.

At *RST, this value is device-dependent.

19.15.5.2 **:OFFSet <numeric_value>**

SOURce:POWer:LIMit:OFFSet

Sets a non-time varying component limit of signal that is added to the time varying signal specified in AMPLitude. The units are set to the default value, alternately to a different value under the UNIT subsystem.

At *RST, this value is device-dependent.

19.15.5.3 :HIGH <numeric_value>

SOURce:POWer:LIMit:HIGH

Sets the more positive peak limit of a time varying signal, it is used in conjunction with LOW.

At *RST, this value is device-dependent.

19.15.5.4 :LOW <numeric_value>

SOURce:POWer:LIMit:LOW

Sets the more negative peak limit of a time varying signal, it is used in conjunction with HIGH.

At *RST, this value is device-dependent.

19.15.5.5 :STATe <Boolean>

SOURce:POWer:LIMit:STATe

Controls whether the LIMit is enabled.

At *RST, this value is set to ON.

19.15.6 :MANual <numeric_value>

SOURce:POWer:MANual

Allows manual adjustment of the amplitude between the sweep limits. The actual amplitude level is determined by the parameter only if SWEep:MODE is set to MANual and the amplitude MODE is set to SWEep. If the sweep limits are changed such that the manual amplitude value would be outside the limits, the manual amplitude value will be set to the nearest limit.

At *RST, this value is device-dependent.

19.15.7 :MODE FIXed|SWEep|LIST

SOURce:POWer:MODE

Determines which set of commands control the amplitude subsystem. If FIXed is selected, the amplitude is determined by the LEVel command under the appropriate subsystem. If SWEep is selected, the source is in the swept mode and amplitude is determined by START, STOP, CENTER, SPAN, and MANual commands under the appropriate subsystem. If LIST is selected, the amplitude values are determined by the appropriate amplitude list (such as LIST:).

At *RST, this value is set to FIXed.

19.15.8 :PROTection

SOURce:POWer:PROTection

Controls the protection circuits.

19.15.8.1 [:LEVel] <numeric_value>

SOURce:POWer:PROTection:LEVel

This command sets the output level at which the output protection circuit will trip.

19.15.8.2 :STATe <Boolean>

SOURce:POWer:PROTection:STATe

Controls whether the output protection circuit is enabled.

At *RST, this value is set to ON.

19.15.8.3 :TRIPped?

SOURce:POWer:PROTection:TRIPped?

This query returns a 1 if the protection circuit is tripped and a 0 if it is untripped.

TRIPped only has a query form and thus has no associated *RST condition.

19.15.8.4 :CLEAr

SOURce:POWer:PROTection:CLEAr

Causes the protection circuit to be cleared.

This command is an event and has no associated *RST condition.

19.15.9 :RANGe <numeric_value>

SOURce:POWer:RANGe

Sets a range for the output amplitude. This command is very hardware-specific.

19.15.9.1 :AUTO <Boolean>|ONCE

SOURce:POWer:RANGe:AUTO

Couples the RANGe to an instrument-determined value. When AUTO is ON, the best range is chosen to reflect the current instrument state.

Explicitly selecting a value for RANGe sets AUTO OFF.

At *RST, this value is set to ON.

19.15.10 :REFeRence <numeric_value>

SOURce:POWer:REFeRence

Sets a reference value which, if STATe is ON, allows all amplitude parameters to be queried/set as relative to the reference value.

At *RST, this value is device-dependent.

19.15.10.1 :STATe <Boolean>

SOURce:POWer:REFeRence:STATe

Determines whether amplitude is measured/output in absolute or relative mode. If STATe is ON, then amplitude is referenced to the value set in REFeRence.

At *RST, this value is set to OFF.

19.15.11 :SLEW <numeric_value>

SOURce:POWer:SLEW

Sets the slew rate of the output change when a new output level is programmed. The units are in (the currently active) amplitude unit/sec.

At *RST, this value is device-dependent.

19.15.12 :SPAN <numeric_value>

SOURce:POWer:SPAN

Sets the amplitude span. If the current amplitude unit is logarithmic (dBm, dBuV, etc), then the unit of SPAN is dB. Otherwise SPAN is programmed in the current amplitude unit.

At *RST, this value is set to 0.

19.15.12.1 :HOLD <Boolean>

SOURce:POWer:SPAN:HOLD

Provides a mechanism to prevent the SPAN from being changed implicitly by the defined coupling between START, STOP, CENTER and SPAN. When HOLD is set to ON, SPAN shall only be changed by explicitly sending the SPAN command.

At *RST, this value is set to OFF.

19.15.12.2 :LINK CENTER|START|STOP

SOURce:POWer:SPAN:LINK

Allows the default couplings for SPAN to be overridden. LINK selects the parameter, either CENTER, START or STOP, that shall not be changed when SPAN's value is changed. For example, if LINK is set to START then changing SPAN shall cause CENTER and STOP, not START to change.

At *RST, this value is set to CENTER, to be compatible with the default definition for the couplings.

19.15.12.3 :FULL

SOURce:POWer:SPAN:FULL

When this command is received, START amplitude is set to its minimum value and STOP amplitude is set to its maximum value. CENTER amplitude and SPAN are set to their coupled values.

This command is an event rather than a state. Therefore it has no associated query, or meaning at *RST.

19.15.13 :START <numeric_value>

SOURce:POWer:START

Sets START amplitude.

At *RST, this value is set to MINimum.

19.15.14 :STOP <numeric_value>

SOURce:POWer:STOP

Sets STOP amplitude.

At *RST, this value is set to MINimum.

19.16 PULse Modulation Subsystem

SOURce:PULM

The PULse Modulation subsystem is used to set the modulation controls and the parameters associated with the modulating signal(s). The keywords to describe the carrier signal exist in other subsystems (such as, FREQuency and PHASe).

KEYWORD	PARAMETER FORM	COMMENTS
:PULM		
:EXTErnal		
:HYSTEResis	<numeric_value>	
:IMPedance	<numeric_value>	
:LEVEl	<numeric_value>	
:POLarity	NORMallINVerted	
:INTernAl		
:FREQuency	<numeric_value>	
:MODE	FIXedLIST	1993
:POLarity	NORMallINVerted	
:SOURce	EXTernallINTernAl{,EXTernall,INTernAl}	
:STATe	<Boolean>	

19.16.1 :EXTErnal

SOURce:PULM:EXTErnal

The EXTErnal subsystem is used to control the specified external signal source. Where multiple external modulation sources are available, the EXTErnal keyword may have a numeric suffix signifying the particular external signal source. If no number is given, a 1 is assumed.

19.16.1.1 :HYSTEResis <numeric_value>

SOURce:PULM:EXTErnal:HYSTEResis

Sets, for the specified signal source, how far the signal voltage must rise above (alternatively, fall below) the threshold LEVEl before the carrier is turned ON from the OFF state (or turned OFF from the ON state). The unit for HYSTEResis is Volts.

At *RST, this defaults to the TTL preset value.

19.16.1.2 :IMPedance <numeric_value>

SOURce:PULM:EXTErnal:IMPedance

Sets the impedance of the specified external signal source. The unit for IMPedance is Ohms.

At *RST, this value is device-dependent.

19.16.1.3 :LEVEl <numeric_value>

SOURce:PULM:EXTErnal:LEVEl

For the specified signal source, this sets the threshold voltage level used to turn ON the carrier. The unit for LEVEl is Volts.

At *RST, this defaults to the TTL preset value.

19.16.1.4 :POLarity NORMalINVerted

SOURce:PULM:EXTernal:POLarity

POLarity, at this level, sets the POLarity for only the specified external signal source. POLarity sets the relationship between the polarity of the applied modulation voltage and the resulting “direction” of modulation. The definition of “normal” polarity for PULSe modulated signal is, a voltage more positive than some threshold (after applying hysteresis effects) and causes the carrier to turn on.

At *RST, this value must be set to NORMal.

19.16.2 :INTernal

SOURce:PULM:INTernal

The INTernal subsystem is used to control the specified internal signal source. Where multiple internal modulation sources are available, the INTernal keyword may have a numeric suffix signifying the particular internal signal source. If no number is given, a 1 is assumed.

19.16.2.1 :FREQuency <numeric_value>

SOURce:PULM:INTernal:FREQuency

Sets the frequency of the specified internal signal source. Some instruments may also allow the internal signal source(s) to be controlled as a subsystem(s). The unit of FREQuency is Hertz (Hz).

At *RST, this value is device-dependent.

19.16.3 MODE

SOURce:PULM:MODE

Determines which set of commands currently control the PULM subsystem. The settings have the following meanings:

FIXed: Settings of individual commands are as last set by the user.

LIST: Settings of individual commands are controlled by the LIST subsystem.

At *RST the value is FIXed.

19.16.4 :POLarity NORMalINVerted

SOURce:PULM:POLarity

POLarity, at this level, is used to set the polarity between the modulator and the modulating signal. The modulating signal may be the sum of several signals, either internal or external sources. POLarity sets the relationship between the polarity of the applied modulation voltage and the resulting “direction” of modulation. The definition of “normal” polarity for PULSe modulated signal is a voltage more positive than some threshold (after applying hysteresis effects) and causes the carrier to turn on.

At *RST, this value must be set to NORMal.

19.16.5 :SOURceEXTernal|INTERNAL{,EXTernal,INTERNAL}

SOURce:PULM:SOURce

Selects the source for the modulating signal, and it may specify a single source or a number of sources. The specified sources in the SOURce command are all selected and turned on. Any sources from a previous selection are ignored. If multiple sources are selected and the device does not support the combination requested in the SOURce command, an execution error -221 must be generated.

Where multiple internal modulation sources are available, individual INTERNAL sources may be distinguished by a unique numeric suffix. If no numeric suffix is provided in a command, a 1 is assumed.

Where multiple external modulation sources are available, individual EXTERNAL sources must be distinguished by a unique number suffix. If no number suffix is provided in a command, a 1 must be assumed.

At *RST, this value is device-dependent.

19.16.6 :STATe <Boolean>

SOURce:PULM:STATe

Turns Pulse Modulation ON or off. Turning ON PULse Modulation Modulation will not automatically turn OFF any other types of modulation. Turning any or all modulations types ON or OFF must be done explicitly. Where a type of modulation is active, turning ON another type of modulation will add to the set of active modulations if that combination of capabilities are legal in the device. Otherwise, an execution error -221 is generated and the instrument state is left undefined.

At *RST, this value must be set to OFF.

19.17 PULSe Subsystem

SOURce:PULSe

This subsystem collects together the commands used to control pulse generation.

KEYWORD	PARAMETER FORM	COMMENTS
:PULSe		
:PERiod	<numeric_value>	
:WIDTh	<numeric_value>	
:DCYClE	<numeric_value>	
:HOLD	WIDTh DCYClE	
:DELay	<numeric_value>	
:DOUBle		
[:STATe]	<Boolean>	
:DELay	<numeric_value>	
:TRANsition		
:STATe	<Boolean>	
[:LEADing]	<numeric_value>	
:TRAIling	<numeric_value>	
:AUTO	<Boolean> ONCE	
:COUNT	<numeric_value>	
:POLarity	NORMal COMPlEment INVerted	

19.17.1 :PERiod <numeric_value>

SOURce:PULSe:PERiod

Sets the period of a pulsed waveform. The fundamental units for PERiod is seconds.

At *RST, this value is device-dependent.

19.17.2 :WIDTh <numeric_value>

SOURce:PULSe:WIDTh

Sets the width or duration of the pulse. The fundamental units for WIDTh is seconds.

At *RST, this value is device-dependent.

19.17.3 :DCYClE <numeric_value>

SOURce:PULSe:DCYClE

Sets the duty cycle of a repetitive pulsed waveform. The fundamental units for DCYClE is percent (%).

At *RST, this value is device-dependent.

19.17.4 :HOLD WIDTh|DCYClE

SOURce:PULSe:HOLD

Sets, for a pulsed waveform, the parameter to be held constant when the period changes.

At *RST, this value is device-dependent.

19.17.5 :DElay <numeric_value>

SOURce:PULSe:DElay

Sets the time from the start of the period to the first edge of the pulse. The fundamental units for DElay is seconds.

At *RST, the value of this parameter is set to either 0 seconds or the nearest specified value.

19.17.6 :DOUble

SOURce:PULSe:DOUble

This subsystem contains the commands necessary to control the parameters associated with double pulse. When a double pulse is enabled, a second pulse which is identical to the primary pulse is generated at some point during the period of the pulse. This second pulse has the same characteristics as the first pulse, including width. A side effect is that since a second pulse has been added, the actual duty cycle of the signal is twice that specified in PULSe:DCYClE. Attempting to enable double pulse mode when the pulse width is greater than:

$$(PULSE:PERiod-PULSe:DOUble:DElay)/2$$

will cause an error -221, "Settings Conflict," to be generated.

19.17.6.1 [:STATE] <Boolean>

SOURce:PULSe:DOUble:STATE

This command sets the double pulse mode to ON or OFF.

At *RST, this parameter is set to OFF.

19.17.6.2 :DElay <numeric_value>

SOURce:PULSe:DOUble:DElay

Sets the time from the start of the period to the first edge of the second pulse. The fundamental units for DElay is seconds.

At *RST, this value is device-dependent.

19.17.7 :TRANsition

SOURce:PULSe:TRANsition

Collects together the commands to control the parameters associated with characteristics of the leading and trailing edges of the pulse.

19.17.7.1 :STATe <Boolean>

SOURce:PULSe:TRANsition:STATe

Sets the transition mode to ON or OFF. If STATe is OFF, then the LEADIng and TRAILing edges must be set to the minimum transition times available in the device. If STATe is ON, then the pulse transition times are specified by values for LEADIng and TRAILing.

At *RST, this parameter is set to OFF.

19.17.7.2 [:LEADIng] <numeric_value>

SOURce:PULSe:TRANsition:LEADIng

This command sets the transition time for the LEADIng edge. The fundamental units for LEADIng is seconds.

At *RST, this value is device-dependent.

19.17.7.3 :TRailing <numeric_value>

SOURce:PULSe:TRANSition:TRAIling

Sets the transition time for the TRAIling edge. The fundamental units for TRAIling is seconds.

At *RST, this value device-dependent.

19.17.7.3.1 :AUTO <Boolean>:ONCE

SOURce:PULSe:TRANSition:TRAIling:AUTO

Couples the value of TRAIling to LEADIng, when AUTO is set to ON.

At *RST, this parameter is set to ON.

19.17.8 :COUNT <numeric_value>

SOURce:PULSe:COUNT

Sets the number of times pulse (or double pulse) must be repeated for a single trigger event.

At *RST, the value of this parameter is set to 1.

19.17.9 :POLarity NORMalCOMPLEMENTINVerted

SOURce:PULSe:POLarity

Sets the polarity of the pulse. For NORMal operation the second nominal state is more positive than the first. COMPLEMENT and INVerted are aliases. For either position, the second nominal transition is more negative than the first.

At *RST, this parameter is set to NORMal.

19.18 **RESistance Subsystem**

SOURce:RESistance

This subsystem controls the characteristics of a resistance source or load.

KEYWORD	PARAMETER FORM	COMMENTS
:RESistance		
[:LEVel]		
[:IMMediate]		
[:AMPLitude]	<numeric_value>	
:OFFSet	<numeric_value>	
:HIGH	<numeric_value>	
:LOW	<numeric_value>	
:TRIGgered		
[:AMPLitude]	<numeric_value>	
:OFFSet	<numeric_value>	
:HIGH	<numeric_value>	
:LOW	<numeric_value>	
:LIMit		
[:AMPLitude]	<numeric_value>	
:OFFSet	<numeric_value>	
:HIGH	<numeric_value>	
:LOW	<numeric_value>	
:PROTection		
[:LEVel]	<numeric_value>	
:STATe	<Boolean>	
:TRIPped?		[query only]
:CLEar		[no query]
:SLEW	<numeric_value>	
:CENTer	<numeric_value>	
:SPAN	<numeric_value>	
:HOLD	<Boolean>	
:LINK	CENTER START STOP	
:FULL		[no query]
:START	<numeric_value>	
:STOP	<numeric_value>	
:MANual	<numeric_value>	
:MODE	FIXed SWEep LIST	
:REFerence	<numeric_value>	
:STATe	<Boolean>	
:RANGe	<numeric_value>	
:AUTO	<boolean> ONCE	

19.18.1 **[[:LEVel]**

SOURce:RESistance:LEVel

This subsystem is used to control the resistance when the source is operating in a continuous or FIXed MODE. That is, as a constant load.

19.18.1.1 **[[:IMMediate]**

SOURce:RESistance:LEVel:IMMediate

Indicates that the subsequent specification of a new resistance is to be processed by the device without waiting for further commands.

19.18.1.1.1 **[[:AMPLitude] <numeric_value>**

SOURce:RESistance:LEVel:IMMediate:AMPLitude

Sets the actual magnitude of the unswept resistance. AMPLitude may be used to specify the resistance level for either a time varying or non-time varying load.

At *RST, this value is device-dependent.

19.18.1.1.2 **:OFFSet <numeric_value>**

SOURce:RESistance:LEVel:IMMediate:OFFSet

Sets the non-time varying component of the resistance that is added to the time varying resistance signal specified in AMPLitude.

At *RST, this value is device-dependent.

19.18.1.1.3 **:HIGH <numeric_value>**

SOURce:RESistance:LEVel:IMMediate:HIGH

Sets the more positive peak of a time varying resistance signal, it is used in conjunction with LOW.

At *RST, this value is device-dependent.

19.18.1.1.4 **:LOW <numeric_value>**

SOURce:RESistance:LEVel:IMMediate:LOW

Sets the more negative peak of a time varying resistance signal, it is used in conjunction with HIGH.

At *RST, this value is device-dependent.

19.18.1.2 **:TRIGgered**

SOURce:RESistance:LEVel:TRIGgered

Indicates that subsequent specification of a new resistance settings are to be transferred to the IMMediate value upon receipt of a trigger signal.

At *RST, after the receipt of an ABORt command, or upon being triggered, the values in the TRIGgered subsystems track the values in the IMMediate subsystem until a command in the TRIGgered subsystem is received. The purpose of tracking is so that spurious triggers do not change the value of LEVel unexpectedly.

19.18.1.2.1 **[[:AMPLitude] <numeric_value>**

SOURce:RESistance:LEVel:TRIGgered:AMPLitude

Sets the actual magnitude of the unswept output signal. AMPLitude may be used to specify the resistance level for either a time varying or non-time varying load.

19.18.1.2.2 **:OFFSet <numeric_value>**

SOURce:RESistance:LEVel:TRIGgered:OFFSet

Sets the non-time varying component of the resistance that is added to the time varying resistance signal specified in AMPLitude.

19.18.1.2.3 **:HIGH <numeric_value>**

SOURce:RESistance:LEVel:TRIGgered:HIGH

Sets the more positive peak of a time varying resistance signal, and it is used in conjunction with LOW.

19.18.1.2.4 **:LOW <numeric_value>**

SOURce:RESistance:LEVel:TRIGgered:LOW

Sets the more negative peak of a time varying resistance signal, and it is used in conjunction with HIGH.

19.18.2 **:LIMit**

SOURce:RESistance:LIMit

Sets the bounds on the resistance value. Setting a larger value will cause the resistance to be clamped to the LIMit value.

19.18.2.1 **[[:AMPLitude] <numeric_value>**

SOURce:RESistance:LIMit:AMPLitude

Sets the limit on the actual magnitude of the unswept resistance. AMPLitude may be used to specify the resistance level for either a time varying or non-time varying load. OFFSet may only be used with AMPLitude, where AMPLitude is used to specify a time varying load.

Explicit reference to the AMPLitude node is optional. Any device that implements this subsystem shall accept and process commands, with or without the AMPLitude node, and respond to them in the same way.

At *RST, this value is device-dependent.

19.18.2.2 **:OFFSet <numeric_value>**

SOURce:RESistance:LIMit:OFFSet

Sets the limit for the non-time varying component of resistance that is added to the time varying resistance signal specified in AMPLitude.

At *RST, this value is device-dependent.

19.18.2.3 **:HIGH <numeric_value>**

SOURce:RESistance:LIMit:HIGH

Sets the more positive peak limit of a time varying resistance signal, it is used in conjunction with LOW.

At *RST, this value is device-dependent.

19.18.2.4 :LOW <numeric_value>

SOURce:RESistance:LIMit:LOW

Sets the more negative peak limit of a time varying resistance signal, and it is used in conjunction with HIGH.

At *RST, this value is device-dependent.

19.18.3 :PROTection

SOURce:RESistance:PROTection

Controls the resistance protection circuit

19.18.3.1 [:LEVel] <numeric_value>

SOURce:RESistance:PROTection:LEVel

Sets the output resistance level at which the output protection circuit will trip.

19.18.3.2 :STATe <Boolean>

SOURce:RESistance:PROTection:STATe

Controls whether the output protection circuit is enabled.

At *RST, this value is set to ON.

19.18.3.3 :TRIPped?

SOURce:RESistance:PROTection:TRIPped?

Returns a 1 if the protection circuit is tripped and a 0 if it is untripped. TRIPped only has a query form and thus has no associated *RST condition.

19.18.3.4 :CLEar

SOURce:RESistance:PROTection:CLEar

Causes the protection circuit to be cleared. This command is an event and has no associated *RST condition.

19.18.4 :SLEW <numeric_value>

SOURce:RESistance:SLEW

Sets the slew rate of the resistance change when a new resistance level is programmed. The units are in Ohm/sec.

At *RST, this value is device-dependent.

19.18.5 :CENTer <numeric_value>

SOURce:RESistance:CENTer

Sets the CENTer RESistance. At *RST, this value is set to MINimum.

19.18.6 :SPAN <numeric_value>

SOURce:RESistance:SPAN

Sets the resistance SPAN.

At *RST, this value is set to 0.

19.18.6.1 :HOLD <Boolean>

SOURce:RESistance:SPAN:HOLD

Provides a mechanism to prevent the SPAN from being changed implicitly by the defined coupling between START, STOP, CENTER and SPAN. When HOLD is set to ON, SPAN shall only be changed by explicitly sending the SPAN command.

At *RST, this value is set to OFF.

19.18.6.2 :LINK CENTER|START|STOP

SOURce:RESistance:SPAN:LINK

Allows the default couplings for SPAN to be overridden. LINK selects the parameter, either CENTER, START or STOP, that shall not be changed when SPAN's value is changed. For example, if LINK is set to START then changing SPAN shall cause CENTER and STOP, not START to change.

At *RST, this value is set to CENTER, to be compatible with the default definition for the couplings.

19.18.6.3 :FULL

SOURce:RESistance:SPAN:FULL

When this command is received, START RESistance is set to MINimum, and STOP RESistance is set to MAXimum. CENTER RESistance and SPAN are set to their coupled values.

This command is an event, rather than a state. Therefore it has no associated query, or meaning at *RST.

19.18.7 :START <numeric_value>

SOURce:RESistance:START

Sets START resistance.

At *RST, this value is set to MINimum.

19.18.8 :STOP <numeric_value>

SOURce:RESistance:STOP

Sets STOP resistance.

At *RST, this value is set to MINimum.

19.18.9 :MANual <numeric_value>

SOURce:RESistance:MANual

Allows manual adjustment of the RESistance between the sweep limits. The actual RESistance level is determined by the parameter only if SWEep:MODE is set to MANual and RESistance:MODE is set to SWEep. If the sweep limits are changed such that the manual RESistance value would be outside the limits, the manual RESistance value will be set to the nearest limit.

At *RST, this value is device-dependent.

19.18.10 :MODE FIXedISWEepLIST

SOURce:RESistance:MODE

Determines which set of commands control the resistance subsystem. If FIXed is selected, the RESistance is determined by the LEVel command. If SWEep is selected, the source is in the swept mode and RESistance is determined by START, STOP, CENTer, SPAN, and MANual commands. If LIST is selected, the RESistance values are determined by LIST:RESistance.

At *RST, this value is set to FIXed.

19.18.11 :REFeRence <numeric_value>

SOURce:RESistance:REFeRence

Sets a reference value which, if STATe is ON, allows all RESistance parameters to be queried/set as a relative to the reference value.

At *RST, this value is device-dependent.

19.18.11.1 :STATe <Boolean>

SOURce:RESistance:REFeRence:STATe

Determines whether RESistance is in absolute or relative mode. If STATe is ON, then RESistance is referenced to the value set in REFeRence.

At *RST, this value is set to OFF.

19.18.12 :RANGe <numeric_value>

SOURce:RESistance:RANGe

Sets a range for the output RESistance.

19.18.12.1 :AUTO <Boolean>IONCE

SOURce:RESistance:RANGe:AUTO

Couples the RANGe to an instrument-determined value. When AUTO POWER and VOLTage is ON, the best range is chosen to reflect the current instrument state.

Explicitly selecting a value for RANGe sets AUTO OFF.

At *RST, this value is set to ON.

19.19 **ROSCillator Subsystem**

SOURce:ROSCillator

This subsystem controls the reference oscillator.

KEYWORD	PARAMETER FORM	NOTES
:ROSCillator		
[:INTernal]		1992
:FREQuency	<numeric_value>	
:EXTernal		1992
:FREQuency	<numeric_value>	1992
:SOURce	INTernal EXTernal NONE	
:AUTO	<Boolean> ONCE	

19.19.1 **[:INTernal]**

SOURce:ROSCillator:INTernal

This subsystem configures the internal reference oscillator(s).

19.19.1.1 **:FREQuency <numeric_value>**

SOURce:ROSCillator:INTernal:FREQuency

Specifies the frequency of the internal reference oscillator. The default units are Hz.

At *RST, this value is device-dependent.

19.19.2 **:EXTernal**

SOURce:ROSCillator:EXTernal

This subsystem configures the external reference oscillator(s).

19.19.2.1 **:FREQuency <numeric_value>**

SOURce:ROSCillator:EXTernal:FREQuency

Specifies the frequency of the external reference oscillator. The default units are Hz.

At *RST, this value is device-dependent.

19.19.3 **:SOURce INTernal|EXTernal|NONE**

SOURce:ROSCillator:SOURce

Controls the selection of the reference oscillator source. This typically refers to a precision, stabilized time base. The parameter values have the following meanings:

- INTernal — The reference frequency is derived from an internal precision oscillator.
- EXTernal — The reference frequency is derived from an external signal supplied to the device.
- NONE — The reference frequency is not derived from any precision oscillator.

At *RST, this value is device-dependent.

19.19.3.1 **:AUTO <Boolean>|ONCE**

SOURce:ROSCillator:SOURce:AUTO

When AUTO is ON, the system automatically selects which oscillator will be used by the instrument.

Explicitly selecting a reference oscillator turns AUTO OFF.

1999 SCPI Command Reference

At *RST, AUTO is set to ON.

1999 SCPI Command Reference

19.20 SPEed Subsystem

SOURce:SPEed

This node controls the speed of the device.

KEYWORD	PARAMETER FORM	COMMENTS
:SPEed		1999
:INITiate		[event; no query]1999
[:LEVel]	<numeric_value>	1999
:SSDLoss		1999
:INITiate		[event; no query]1999
:LAtime	<numeric_value>	1999
:STIME	<numeric_value>	1999

19.20.1 :INITiate

:SOURce:SPEed:INITiate

This command initiates speed. This is an overlapped command. This command describes an event and therefore has no associated *RST condition.

When this command is initiated for chassis dynamometers, the dynamometer must:

- Check critical ancillaries, if any, for proper conditions: Brakes Off, Motor Contactor On, parameters within dynamometer safety limits. For conditions that prohibit operation of this procedure, issue appropriate warning messages, and execute the Idle Procedure.
- Set the DYNO operation condition register Go To Speed Procedure bit indicating Go To Speed Procedure is executing. Note: The DYNO operation condition register will be further defined in Volume 4.
- Accelerate dynamometer to the :SOURce:SPEed:LEVel speed point at the rate in :SOURce:ACCeleration. The dynamometer determines when the speed point has stabilized.
- Maintain speed and wait for a command.
- When commanded out of this procedure, clear the DYNO operation condition register Go To Speed Procedure bit indicating the Go To Speed Procedure is not executing.
- Issue the process complete message.

19.20.2 [:LEVel] <numeric_value>

:SOURce:SPEed:LEVel

Set the desired speed set-point in m/s.

At *RST, this value is device dependent.

19.20.3 :SSDLoss

SOURce:SPEed:SSDLoss

Steady State Driveline Loss

This node describes the procedure to measure the driveline loss forces as a function of speed. The Steady State Driveline Procedure measures the vehicle losses due to vehicle drive train

components. The vehicle losses are measured by operating the dynamometer at various selected speeds and measuring the force required to maintain the constant speed over a specified time interval. The results are stored in the DrvLLoss pre-defined table. This table has two columns, SPEed, the speeds at which the losses are to be measured (input) and LOSS, the measured losses at each speed (result). Coefficients for vehicle driveline losses are determined at the end of the procedure and reside in DrvLCoeff.

19.20.3.1 :INITiate

:SOURce:SPEed:SSDLoss:INITiate

This is an overlapped command. The command uses the speed points defined in the :MEMory:TABLE:SElect DrvLLoss. This command describes an event and therefore has no associated *RST condition.

When this procedure is initiated for dynamometers, the dynamometer must:

- Check critical ancillaries, if any, for proper conditions: Brakes Off, Motor Contactor On, parameters within dynamometer safety limits. For conditions that prohibit operation of this command, issue appropriate warning messages, and execute the Idle Procedure.
- Set the DYNO operation condition register Driveline Loss Procedure bit indicating the driveline loss is executing. Note: The DYNO operation condition register will be further defined in Volume 4.
- Accelerate dynamometer to the first speed point defined in the DrvLLoss memory table at the rate in :SOURce:ACceleration.
- Stabilize at this speed for :SOURce:SPEed:SSDLoss:STIME. If the speed does not stabilize, set Failed to Stabilize bit of the DYNO questionable condition register and continue with the procedure.
- Average the force at the roll surface required to maintain the speed set point over the loss average time :SOURce:SPEed:SSDLoss:LATime. The average force value is placed in the DrvLLoss memory table for the specified speed point. Repeat steps 3 to 5 for each point in the DrvLLoss table. The first zero speed in the DrvLLoss table ends the procedure.
- Perform a curve fit on the DrvLLoss data and place the coefficients in the memory table DrvLCoeff.
- Clear the DYNO operation condition register bit 7 indicating the driveline loss is not executing.
- Issue the process complete message.
- Execute the Idle Procedure.

19.20.3.2 :LATime <numeric_value>

:SOURce:SPEed:SSDLoss:LATime

Loss Averaging Time - sec

1999 SCPI Command Reference

This command sets the time after stabilization that the losses are to be averaged before moving to the next speed point.

At *RST, this value is set to 5.

19.20.3.3 :STIMe <numeric_value>

:SOURce:SPEed:SSDLoss:STIMe

Stabilization Time - sec

Time for stabilization prior to LATime after a change in speed.

At *RST, this value is set to 5.

19.21 SWEep Subsystem

SOURce:SWEep

This subsystem controls the sweep for those instruments which generate signals as a function of time. Other parameters may also be swept by specifying SWEep as their MODE (for example, FREQuency:MODE SWEep). Units for this subsystem are seconds unless otherwise noted.

KEYWORD	PARAMETER FORM	COMMENTS
:SWEep		
:TIME	<numeric_value>	
:AUTO	<Boolean> ONCE	
:LLIMit	<numeric_value>	
:DWELl	<numeric_value>	
:AUTO	<Boolean> ONCE	
:DIRection	UP DOWN	
:MODE	AUTO MANual	
:SPACing	LINEar LOGarithmic	
:GENeration	STEPped ANALog	
:STEP	<numeric_value>	
:POINts	<numeric_value>	
:COUNT	<numeric_value>	

19.21.1 :TIME <numeric_value>

SOURce:SWEep:TIME

Sets the duration of the sweep. Note that this does not turn sweeping on. Setting this value automatically turns TIME:AUTO OFF.

19.21.1.1 :AUTO <Boolean>|ONCE

SOURce:SWEep:TIME:AUTO

When enabled, the sweep time is calculated internally and is dependent on the span of the sweep.

At *RST, AUTO ON is selected.

19.21.1.2 :LLIMit <numeric_value>

SOURce:SWEep:TIME:LLIMit

Defines a lower limit for sweep time. This lower limit restricts the sweep time value set either explicitly or automatically.

At *RST, LLIMit is set to a device-dependent value.

19.21.2 :DWELl <numeric_value>

SOURce:SWEep:DWELl

Controls the amount of time spent at each point during a sweep. Note that DWELl cannot exceed TIME/POINts. Trying to set it to a greater value will cause an error to result. The points will not change, but time may be changed as a device-dependent decision.

At *RST, this value is device-dependent.

19.21.2.1 :**AUTO** <Boolean>|ONCE

SOURce:SWEep:DWELL:AUTO

When AUTO ON is selected, the dwell time is coupled to the sweep time and number of points. The coupling equation is:

$$DWELL=(TIME/POINTS)-(device-dependent\ stepping\ time)$$

Setting a value for dwell turns AUTO OFF.

At *RST, AUTO is set to ON.

19.21.3 :**DIRection** UP|DOWN

SOURce:SWEep:DIRection

Controls the direction of the sweep. If UP is selected, the sweep is carried out in ascending order, from START to STOP. If DOWN is selected, the sweep is carried out in descending order, from STOP to START.

At *RST, the value of this function is UP.

19.21.4 :**MODE** AUTO|MANual

SOURce:SWEep:MODE

Selects the sweep conditions for the sweep. The various conditions are:

- AUTO: The sweep is controlled by the internal sweep generator.
- MANual: The sweep is controlled by front panel controls or by the manual command, such as, FREQuency:MANual. This setting is valid only if the :MANual capability has been implemented.

At *RST, this function is set to AUTO.

19.21.5 :**SPACing** LINear|LOGarithmic

SOURce:SWEep:SPACing

Determines the swept entity versus time characteristics of the sweep. The various settings have the following meanings:

- LINear: For stepped sweeps, the swept entity is incremented (decremented) by the step size until the sweep limit is reached. For analog sweeps, a linear ramp is generated.
- LOGarithmic: For non-linear sweeps, step size is determined by a logarithmic curve fitted between the start and stop frequency. Stepping is determined by the SWEep:POINTs. For analog sweeps, a logarithmic ramp is generated.

At *RST, this value is set to LINear.

19.21.6 :**GENeration** STEPped|ANALog

SOURce:SWEep:GENeration

Selects between an analog or stepped sweep.

- STEPped: A stepped sweep is selected, and the points are controlled by the SWEep:POINTs and SWEep:STEP settings.
- ANALog: The sweep is controlled by an analog signal.

At *RST, this value is device-dependent.

19.21.7 **:STEP <numeric_value>**

SOURce:SWEep:STEP

Controls the swept entity step size for a stepped linear sweep. This parameter is not used if GENeration is ANALog or if SPACing is LOGarithmic.

This value is hard coupled to SPAN and POINTs by the equation:

$$STEP = SPAN / (POINTs - 1) \quad 1991$$

Changing the value of STEP will change POINTs, but not SPAN.

The value is instrument-dependent at *RST, but is coupled to the reset value of POINTs and SPAN.

19.21.8 **:POINTs <numeric_value>**

SOURce:SWEep:POINTs

The number of points in a stepped sweep. This parameter is not used if GENeration is ANALog. In a linear sweep, this value is coupled to sweep step by the equation:

$$STEP = SPAN / (POINTs - 1) \quad 1991$$

If POINTs are changed, STEP will also be changed, but not SPAN. In a logarithmic sweep, POINTs will determine the number of points/decade of sweep by the equation:

$$POINTs / DECADE = (POINTs - 1) / SPAN \text{ (in decades)} \quad 1991$$

Note that style rules on resolution do not apply to this command. If the exact number of points specified is not available, an error is generated, and the value remains unchanged.

At *RST, this value is set to MAXimum.

19.21.9 **:COUNT <numeric_value>**

SOURce:SWEep:COUNT

This command determines the number of sweeps which are enabled by a single trigger event.

At *RST, this function is set to 1.

19.22 TEMPeratureSubsystem

SOURce:TEMPerature

This subsystem controls the temperature of an environmental chamber.

KEYWORD	PARAMETER FORM	COMMENTS
:TEMPerature		1993
:APRobe	<numeric_list>	1993
:DWELl	<numeric_value>	1993
:LCONstants		1993
:DERivative	<numeric_value>	1993
[:GAIN]	<numeric_value>	1993
:INTegral	<numeric_value>	1993
:MODE	FIXed LIST PROGram	1993
:PROTection		1993
[:HIGH]		1993
:CLEar		[no query] 1993
[:LEVel]	<numeric_value>	1993
:STATe	<Boolean>	1993
:TOUT	<numeric_value>	1993
:TRIPped?		[query only] 1993
:LOW		1993
:CLEar		[no query] 1993
[:LEVel]	<numeric_value>	1993
:STATe	<Boolean>	1993
:TOUT	<numeric_value>	1993
:TRIPped?		[query only] 1993
:RTIME	<numeric_value>	1993
[:SPOint]	<numeric_value>	1993

19.22.1 :APRobe <numeric_list>

SOURce:TEMPerature:APRobe

Active PRobe selects one or more probes to become part of the feedback loop to control the temperature.

A temperature controller can have one or more probes to measure temperature at various places. If more than one probe is active, the signals from the probes are averaged to control the temperature.

AT *RST which probes are active is device dependent.

19.22.2 :DWELl <numeric_value>

SOURce:TEMPerature:DWELl

The DWELl command specifies the amount of time the temperature chamber will stay at the target temperature. At the end of the DWELl period the temperature will return to the ambient temperature. The DWELl time does not include the Ramp TIME.

AT *RST the value is device dependent.

19.22.3 :LCONstants

SOURCE:TEMPerature:LCONstants

A temperature chamber is a closed loop thermal system. The Loop CONstants of such a system determine its performance. These loop constants are popularly known as pid.

The p is the proportional or GAIN control and can be imagined as a coarse control. The i is the integral (lead) compensator and d is the derivative (lag) compensator. These two compensators are additional controls to fine tune the system. The INTEGRal value improves the steady-state error and the DERivative value improves the transient response of a feedback control system.

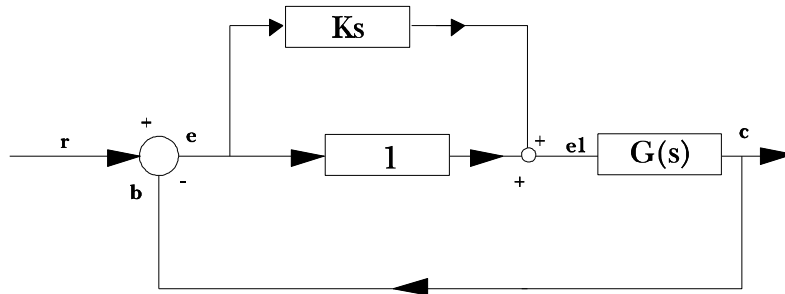
For further information on feedback control systems please refer to Vol 1: Syntax and Style - Chapter 2, Tuning a Control Loop Performance

19.22.3.1 :DERivative <numeric_value>

SOURCE:TEMPerature:LCONstants:DERivative

The DERivative, which is also called a lead compensator, is used to improve the transient response of the system. The unit for DERivative is (currently selected units) degrees per second.

With both GAIN and DERivative are specified, the output (heat/cold) is proportional to the input signal plus its derivative. This is shown in the figure below using Laplace transform notation.



At *RST the value is device dependent.

19.22.3.2 [:GAIN] <numeric_value>

SOURCE:TEMPerature:LCONstants:GAIN

The value of :GAIN determines the time required to reach the target temperature. The higher the value of gain, the sooner the target temperature is achieved. The GAIN is a unitless quantity.

At *RST the value is device dependent.

19.22.3.3 :INTEgral <numeric_value>

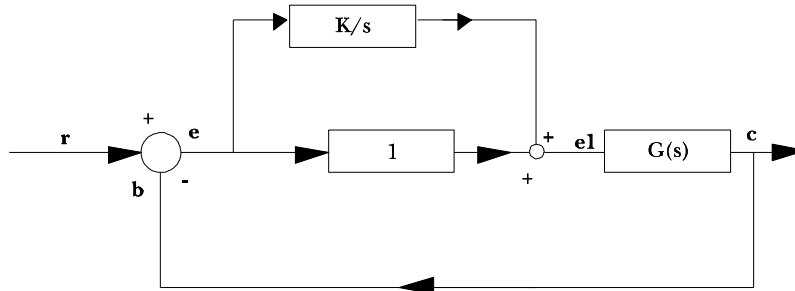
SOURCE:TEMPerature:LCONstants:INTEgral

The INTEgral, which is also called a lag compensator, is used to minimize the steady-state error.

1999 SCPI Command Reference

When the steady-state error is too large, it is possible to minimize this value by adjusting the value of INTEgral. The units for INTEgral are degree*seconds.

With both GAIN and INTEgral are specified, the output (heat/cold) is proportional to the input signal plus its integral. This is shown in the figure below using Laplace transform notation.



At *RST the value is device dependent.

19.22.4 :MODE FIXEdLISTIPROGram

SOURce:TEMPerature:MODE

Determines which set of commands control the temperature subsystem. If FIXEd is selected, the temperature is determined by the SPOint command. If LIST is selected, the temperature values are determined by the appropriate temperature list (such as LIST:). If PROGram is selected the temperature is controlled by the currently selected program specified under PROGram subsystem.

At *RST this value is set to FIXEd.

19.22.5 :PROTection

SOURce:TEMPerature:PROTection

Protects the temperature chamber as well as the unit under test.

19.22.5.1 [:HIGH]

SOURce:TEMPerature:PROTection:HIGH

The :HIGH subsystem controls the protection. The high value must be exceeded for the protection action to be taken.

19.22.5.1.1 :CLEAr

SOURce:TEMPerature:PROTection:HIGH:CLEAr

Causes the protection circuit to be cleared.

This command is an event and has no associated *RST condition.

19.22.5.1.2 [:LEVel] <numeric_value>

SOURce:TEMPerature:PROTection:HIGH:LEVel

Specifies the high temperature trip level. The device will check the current value of the temperature. If the temperature exceeds the programmed protection level, the chamber will trip the protection mechanism.

At *RST the value is device dependent.

19.22.5.1.3 :STATe <Boolean>

SOURce:TEMPerature:PROTection:HIGH:STATe

Enables or disables the protection mechanism.

At *RST this value is set to ON.

19.22.5.1.4 :TOUT <numeric_value>

SOURce:TEMPerature:PROTection:HIGH:TOUT

Specifies the time out in the currently specified unit of time. The device will start the time out counter anytime the difference between the controlled temperature and the SetPoint is greater than a predetermined quantity (for example 2 degrees C). If the temperature has not reached the target value and if the time out has occurred the chamber will trip the protection mechanism.

The time out counter will be reset as soon as the temperature has reached the target value within the predetermined tolerance limits.

AT *RST the value is device dependent.

19.22.5.1.5 :TRIPped?

SOURce:TEMPerature:PROTection:HIGH:TRIPped?

This query returns a 1 if the protection circuit is tripped and a 0 if it is untripped.

19.22.5.2 :LOW

SOURce:TEMPerature:PROTection:LOW

The :LOW subsystem controls the protection. The temperature must fall below the lowest level for the protection action to be taken.

19.22.5.2.1 :CLEar

SOURce:TEMPerature:PROTection:LOW:CLEar

Causes the protection circuit to be cleared.

This command is an event and has no associated *RST condition.

19.22.5.2.2 [:LEVel] <numeric_value>

SOURce:TEMPerature:PROTection:LOW:LEVel

Specifies the low temperature trip level. The device will check the current value of the temperature. If the temperature falls below the programmed protection level, the chamber will trip the protection mechanism.

AT *RST the value is device dependent.

19.22.5.2.3 :STATe <Boolean>

SOURce:TEMPerature:PROTection:LOW:STATe

Enables or disables the protection mechanism.

At *RST this value is set to ON.

19.22.5.2.4 :TOUT <numeric_value>

SOURce:TEMPerature:PROTection:LOW:TOUT

Specifies the time out in the currently specified unit of time. The device will start the time out counter anytime the difference between the controlled temperature and the SetPOint is greater than a predetermined quantity (for example 2 degrees C). If the temperature has not reached the target value and if the time out has occurred the chamber will trip the protection mechanism.

The time out counter will be reset as soon as the temperature has reached the target value within the predetermined tolerance limits.

AT *RST the value is device dependent.

19.22.5.2.5 :TRIPped?

SOURce:TEMPerature:PROTection:LOW:TRIPped?

This query returns a 1 if the protection circuit is tripped and a 0 if it is untripped.

19.22.6 :RTIME <numeric_value>

SOURce:TEMPerature:RTIME

The Ramp TIME command instructs the environmental chamber the amount of time it should take to reach the target temperature.

The unit for RTIME time is based on the currently selected unit.

AT *RST the value is device dependent.

19.22.7 [:SPOint] <numeric_value>

SOURce:TEMPerature:SPOint

The SetPOint command sets the target temperature of the environmental chamber. The units are the current temperature units.

19.23 **VOLTage Subsystem**

SOURce:VOLTage

This subsection controls the signal amplitude characteristics of the source.

There is only one subsystem for each of the main amplitude characteristics of the signal. Thus, AC voltage and DC voltage are both placed under the VOLTage subsystem. A device which implemented both simultaneously would implement them as two separate sources, one AC and one DC.

KEYWORD	PARAMETER FORM	COMMENTS
:VOLTage		
:ATTenuation	<numeric_value>	
:AUTO	<Boolean> ONCE	
:ALC		
[:STATe]	<Boolean>	
:SEARch	<Boolean> ONCE	
:SOURce	INTernal DIODE PMETER MMHead	
:BANDwidth	<numeric_value>	
!BWIDth		
:AUTO	<Boolean> ONCE	
:CENTer	<numeric_value>	
[:LEVel]		
[:IMMediate]		
[:AMPLitude]	<numeric_value>	
:AUTO	<Boolean> ONCE	1991
:OFFSet	<numeric_value>	
:HIGH	<numeric_value>	
:LOW	<numeric_value>	
:TRIGgered		
[:AMPLitude]	<numeric_value>	
:OFFSet	<numeric_value>	
:HIGH	<numeric_value>	
:LOW	<numeric_value>	
:LIMit		
[:AMPLitude]	<numeric_value>	
:OFFSet	<numeric_value>	
:HIGH	<numeric_value>	
:LOW	<numeric_value>	
:STATe	<Boolean>	
:MANual	<numeric_value>	
:MODE	FIXed SWEep LIST	
:PROTection		
[:LEVel]	<numeric_value>	
:STATe		
:TRIPped?		[query only]
:CLEar		[no query]

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	COMMENTS
:RANge	<numeric_value>	
:AUTO	<Boolean> ONCE	
:REFerence	<numeric_value>	
:STATe	<Boolean>	
:SLEW	<numeric_value>	
:SPAN	<numeric_value>	
:HOLD	<Boolean>	
:LINK	CENter STARt STOP	
:FULL		[no query]
:STARt	<numeric_value>	
:STOP	<numeric_value>	

19.23.1 :ATTenuation <numeric_value>

SOURce:VOLTage:ATTenuation

Sets the ATTenuation level. Note that when increasing the level by 10 dB the magnitude of the outgoing signal as well as the LEVel will be decreased by 10 dB.

Default units are as determined in the UNIT system. This is coupled to LEVel.

19.23.1.1 :AUTO <Boolean>

SOURce:VOLTage:ATTenuation:AUTO

Couples the attenuator to LEVel.

Programming a specified attenuation sets AUTO OFF.

See SOURce subsystem introduction for additional explanation.

At *RST, AUTO is set to ON.

19.23.2 :ALC

SOURce:VOLTage:ALC

This subsystem command controls the automatic leveling control of the source.

19.23.2.1 [:STATe] <Boolean>

SOURce:VOLTage:ALC:STATe

Controls whether the ALC loop controls the output level.

At *RST, this value is device-dependent.

19.23.2.2 :SEARch <Boolean>|ONCE

SOURce:VOLTage:ALC:SEARch

SEARch enables a form of leveling where the output level is calibrated by momentarily closing the leveling loop. Once the correct amount of level adjustment has been determined to produce the desired output level, the leveling loop is opened.

When ON, the search routine will be done anytime the output level requires change (for example, when LEVel is changed).

1999 SCPI Command Reference

Selecting SEARCH ONCE will have the effect of setting SEARCH to ON and then OFF. A modulator setting is determined by the system which will not be changed until the output level is explicitly reprogrammed.

At *RST, this value is set to OFF.

19.23.2.3 :SOURce INTernalDIODElPMETERlMMHead

SOURce:VOLTage:ALC:SOURce

Selects the source of the feedback signal for ALC. The parameters have the following meanings:

- INTernal — The ALC feedback signal is measured from a point inside the source.
- DIODE — The ALC feedback is being fed from an external diode detector.
- PMETER — The ALC signal is coming from a power meter.
- MMHead — A millimeter head, with built in leveling, is being used to supply the ALC signal.

At *RST, this value is INTernal.

19.23.2.4 :BANDwidthl:BWIDth <numeric_value>

SOURce:VOLTage:ALC:BANDwidth

Controls the bandwidth of the ALC feedback signal. This parameter is in units of Hz.

At *RST, this value is device-dependent.

19.23.2.4.1 :AUTO <Boolean>IONCE

SOURce:VOLTage:ALC:BANDwidth:AUTO

Couples the bandwidth of the ALC feedback signal to instrument-dependent parameters. When AUTO is ON, the best bandwidth is chosen to reflect the current instrument state.

Explicitly selecting a value for ALC:BANDwidth sets AUTO OFF.

At *RST, this value is set to ON.

19.23.3 :CENTER <numeric_value>

SOURce:VOLTage:CENTER

Sets the center amplitude.

At *RST, this value is set to MINimum.

19.23.4 [:LEVel]

SOURce:VOLTage:LEVel

This subsystem is used to control the signal amplitude when the source is operating in a continuous or FIXEd MODE.

19.23.4.1 [:IMMediate]

SOURce:VOLTage:LEVel:IMMediate

Indicates that the subsequent specification of a new signal amplitude is to be processed by the device without waiting for further commands.

19.23.4.1.1 [:AMPLitude] <numeric_value>

SOURce:VOLTage:LEVel:IMMediate:AMPLitude

Sets the actual magnitude of the unswept output signal in terms of the current operating units. The units are set to the default value, or alternately to a different value under the UNIT subsystem. AMPLitude may be used to specify the level for either a time varying or non-time varying signal.

Note the optional nodes exist to enable VOLTage to be directly followed by a <numeric_value>. This is useful for programming simple devices in a straightforward manner.

At *RST, the signal being sourced should be set to a “safe” condition. This is generally achieved by setting the amplitude to its minimum value in conjunction with OUTPut:STATe subsystem.

19.23.4.1.1.1 :AUTO <Boolean>IONCE

SOURce:VOLTage:LEVel:IMMediate:AMPLitude:AUTO

If AUTO ON is selected, the current setting will be adjusted automatically when a new voltage setting (with the existing current setting) exceeds the maximum power limit. If AUTO OFF is selected a new voltage setting will cause an execution error -221 (Settings conflict), and the instrument state is device dependent when the maximum power limit is exceeded.

*RST condition: AUTO OFF

19.23.4.1.2 :OFFSet <numeric_value>

SOURce:VOLTage:LEVel:IMMediate:OFFSet

Sets the non-time varying component of the signal that is added to the time varying signal specified in AMPLitude, in terms of the current operating units. The units are set to the default value, or alternately to a different value under the UNIT subsystem. When SOURce:FUNCTION is DC, the effect of the OFFSet is device-dependent.

At *RST, the signal being sourced should be set to a “safe” condition. This is generally achieved by setting the offset to its minimum value in conjunction with OUTPut:STATe subsystem.

19.23.4.1.3 :HIGH <numeric_value>

SOURce:VOLTage:LEVel:IMMediate:HIGH

Sets the more positive peak of a time varying signal. It is used in conjunction with LOW.

At *RST, the signal being sourced should be set to a “safe” condition. This is achieved by setting this to a minimum value in conjunction with OUTPut:STATe subsystem.

19.23.4.1.4 :LOW <numeric_value>

SOURce:VOLTage:LEVel:IMMediate:LOW

Sets the more negative peak of a time varying signal, it is used in conjunction with HIGH.

At *RST, the signal being sourced should be set to a “safe” condition. This is achieved by setting this to a minimum value in conjunction with OUTPut:STATe subsystem.

19.23.4.2 :TRIGgered

SOURce:VOLTage:LEVel:TRIGgered

Indicates that subsequent specification of a new signal amplitude is to be transferred to the IMMEDIATE value upon receipt of a trigger signal.

At *RST, after the receipt of an ABORT command or upon being triggered, the value in the TRIGgered subsystem tracks the values in the IMMEDIATE subsystem until a command in the TRIGgered subsystem is received. The purpose of tracking is so that spurious triggers do not change the value of LEVel unexpectedly.

19.23.4.2.1 [:AMPLitude] <numeric_value>

SOURce:VOLTage:LEVel:TRIGgered:AMPLitude

Sets the actual magnitude of the unswept output signal in terms of the current operating units. The units are set to the default value, or alternately to a different value under the UNIT subsystem. AMPLitude may be used to specify the level for either a time varying or non-time varying signal.

19.23.4.2.2 :OFFSet <numeric_value>

SOURce:VOLTage:LEVel:TRIGgered:OFFSet

Sets the non-time varying component of the signal that is added to the time varying signal specified in AMPLitude. The units are set to the default value, or to a different value under the UNIT subsystem. When SOURce:FUNCTION is DC, the effect of OFFSet is device-dependent.

19.23.4.2.3 :HIGH <numeric_value>

SOURce:VOLTage:LEVel:TRIGgered:HIGH

This command is used to set the more positive peak of a time varying signal. It is used in conjunction with LOW.

19.23.4.2.4 :LOW <numeric_value>

SOURce:VOLTage:LEVel:TRIGgered:LOW

This command is used to set the more negative peak of a time varying signal. It is used in conjunction with HIGH.

19.23.5 :LIMit

SOURce:VOLTage:LIMit

Sets the maximum bounds on the output value. Setting a larger value will cause the output level to be clamped to the LIMit value.

19.23.5.1 [:AMPLitude] <numeric_value>

SOURce:VOLTage:LIMit:AMPLitude

Sets the limit on the actual magnitude of the unswept output signal in terms of the current operating units. The units are set to the default value, or alternately to a different value under the UNIT subsystem. AMPLitude may be used to specify the level for either a time varying or non-time varying signal. OFFSet may only be used with AMPLitude, where AMPLitude is used to specify a time varying load.

1999 SCPI Command Reference

Explicit reference to the AMPLitude node is optional. Any device that implements this subsystem must accept and process commands, with or without the AMPLitude node, and respond to them in the same way.

At *RST, this value is device-dependent.

19.23.5.2 :OFFSet <numeric_value>

SOURce:VOLTage:LIMit:OFFSet

Sets a non-time varying component limit of signal that is added to the time varying signal specified in AMPLitude. The units are set to the default value, alternately to a different value under the UNIT subsystem.

At *RST, this value is device-dependent.

19.23.5.3 :HIGH <numeric_value>

SOURce:VOLTage:LIMit:HIGH

Sets the more positive peak limit of a time varying signal, it is used in conjunction with LOW.

At *RST, this value is device-dependent.

19.23.5.4 :LOW <numeric_value>

SOURce:VOLTage:LIMit:LOW

Sets the more negative peak limit of a time varying signal, it is used in conjunction with HIGH.

At *RST, this value is device-dependent.

19.23.5.5 :STATe <Boolean>

SOURce:VOLTage:LIMit:STATe

Controls whether the LIMit is enabled.

At *RST, this value is set to ON.

19.23.6 :MANual <numeric_value>

SOURce:VOLTage:MANual

Allows manual adjustment of the amplitude between the sweep limits. The actual amplitude level is determined by the parameter only if SWEep:MODE is set to MANual and the amplitude MODE is set to SWEep. If the sweep limits are changed such that the manual amplitude value would be outside the limits, the manual amplitude value will be set to the nearest limit.

At *RST, this value is device-dependent.

19.23.7 :MODE FIXEd|SWEep|LIST

SOURce:VOLTage:MODE

Determines which set of commands control the amplitude subsystem. If FIXEd is selected, the amplitude is determined by the LEVEL command under the appropriate subsystem. If SWEep is selected, the source is in the swept mode and amplitude is determined by START, STOP, CENTER, SPAN, and MANual commands under the appropriate subsystem. If LIST is

1999 SCPI Command Reference

selected, the amplitude values are determined by the appropriate amplitude list (such as LIST:).

At *RST, this value is set to FIXed.

19.23.8 :PROTection

SOURce:VOLTage:PROTection

Controls the protection circuits.

19.23.8.1 [:LEVel] <numeric_value>

SOURce:VOLTage:PROTection:LEVel

This command sets the output level at which the output protection circuit will trip.

19.23.8.2 :STATe <Boolean>

SOURce:VOLTage:PROTection:STATe

Controls whether the output protection circuit is enabled.

At *RST, this value is set to ON.

19.23.8.3 :TRIPped?

SOURce:VOLTage:PROTection:TRIPped?

This query returns a 1 if the protection circuit is tripped and a 0 if it is untripped.

TRIPped only has a query form and thus has no associated *RST condition.

19.23.8.4 :CLEAr

SOURce:VOLTage:PROTection:CLEAr

Causes the protection circuit to be cleared.

This command is an event and has no associated *RST condition.

19.23.9 :RANGe <numeric_value>

SOURce:VOLTage:RANGe

Sets a range for the output amplitude. This command is very hardware-specific.

19.23.9.1 :AUTO <Boolean>|ONCE

SOURce:VOLTage:RANGe:AUTO

Couples the RANGe to an instrument-determined value. When AUTO is ON, the best range is chosen to reflect the current instrument state.

Explicitly selecting a value for RANGe sets AUTO OFF.

At *RST, this value is set to ON.

19.23.10 :REFerence <numeric_value>

SOURce:VOLTage:REFerence

Sets a reference value which, if STATe is ON, allows all amplitude parameters to be queried/set as relative to the reference value.

At *RST, this value is device-dependent.

19.23.10.1 :STATe <Boolean>

SOURce:VOLTage:REfERENCE:STATe

Determines whether amplitude is measured/output in absolute or relative mode. If STATe is ON, then amplitude is referenced to the value set in REfERENCE.

At *RST, this value is set to OFF.

19.23.11 :SLEW <numeric_value>

SOURce:VOLTage:SLEW

Sets the slew rate of the output change when a new output level is programmed. The units are in (the currently active) amplitude unit/sec.

At *RST, this value is device-dependent.

19.23.12 :SPAN <numeric_value>

SOURce:VOLTage:SPAN

Sets the amplitude span. If the current amplitude unit is logarithmic (dBm, dBuV, etc), then the unit of SPAN is dB. Otherwise SPAN is programmed in the current amplitude unit.

At *RST, this value is set to 0.

19.23.12.1 :HOLD <Boolean>

SOURce:VOLTage:SPAN:HOLD

Provides a mechanism to prevent the SPAN from being changed implicitly by the defined coupling between START, STOP, CENTER and SPAN. When HOLD is set to ON, SPAN shall only be changed by explicitly sending the SPAN command.

At *RST, this value is set to OFF.

19.23.12.2 :LINK CENTER|START|STOP

SOURce:VOLTage:SPAN:LINK

Allows the default couplings for SPAN to be overridden. LINK selects the parameter, either CENTER, START or STOP, that shall not be changed when SPAN's value is changed. For example, if LINK is set to START then changing SPAN shall cause CENTER and STOP, not START to change.

At *RST, this value is set to CENTER, to be compatible with the default definition for the couplings.

19.23.12.3 :FULL

SOURce:VOLTage:SPAN:FULL

When this command is received, START amplitude is set to its minimum value and STOP amplitude is set to its maximum value. CENTER amplitude and SPAN are set to their coupled values.

This command is an event rather than a state. Therefore it has no associated query, or meaning at *RST.

19.23.13 :START <numeric_value>

SOURce:VOLTage:START

Sets START amplitude.

1999 SCPI Command Reference

At *RST, this value is set to MINimum.

19.23.14 **:STOP <numeric_value>**

SOURce:VOLTage:STOP

Sets STOP amplitude.

At *RST, this value is set to MINimum.

20 STATus Subsystem

This subsystem controls the SCPI-defined status-reporting structures. SCPI defines, in addition to those in *IEEE 488.2*, QUEStionable, OPERation, INSTRument SUMmary and INSTRument registers. These registers conform to the *IEEE 488.2* specification and each may be comprised of a condition register, an event register, an enable register, and negative and positive transition filters. The purpose and definition of the SCPI-defined registers is described in “Volume 1: Syntax and Style”.

SCPI also defines an *IEEE 488.2* queue for status. The queue provides a human readable record of instrument events. The application programmer may individually enable events into the queue. STATus:PRESet enables errors and disables all other events. If the summary of the queue is reported, it shall be reported in bit 2 of the status byte register. A subset of error/event numbers is defined by SCPI. Additional error/event numbers will be defined at a later date.

KEYWORD	PARAMETER FORM	NOTES
STATus		
:OPERation		
:BIT<n>	n=8-12	1999
:CONDition?		[query only] 1999
:ENABLE	<NR> <non-decimal_numeric>	1999
[:EVENT]?		[query only] 1999
:NTRansition	<NR> <non-decimal_numeric>	1999
:PTRansition	<NR> <non-decimal_numeric>	1999
:CONDition?		[query only]
:ENABLE	<NRf> <non-decimal numeric>	1995
[:EVENT]?		[query only]
:INSTRument		
:CONDition?		[query only]
:ENABLE	<NRf> <non-decimal numeric>	1995
[:EVENT]?		[query only]
:ISUMmary<n>		
:CONDition?		[query only]
:ENABLE	<NRf>	
[:EVENT]?		[query only]
:NTRansition	<NRf>	
:PTRansition	<NRf>	
:NTRansition	<NRf> <non-decimal numeric>	1995
:PTRansition	<NRf> <non-decimal numeric>	1995
:MAP	<NRf>,<NRf>	
:NTRansition	<NRf> <non-decimal numeric>	1995
:PTRansition	<NRf> <non-decimal numeric>	1995
:PRESet		[no query]
:QUEStionable		
:BIT<n>	n=9-12	1999

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
:CONDition?		[query only] 1999
:ENABle	<NRt> <non-decimal_numeric>	1999
[:EVENTt]?		[query only] 1999
:NTRansition	<NRt> <non-decimal_numeric>	1999
:PTRansition	<NRt> <non-decimal_numeric>	1999
:CONDition?		[query only]
:ENABle	<NRf> <non-decimal numeric>	1995
[:EVENTt]?		[query only]
:INSTrument		1991
:CONDition?		[query only] 1991
:ENABle	<NRf> <non-decimal numeric>	1991,5
[:EVENTt]?		[query only] 1991
:ISUMmary<n>		1991
:CONDition?		[query only] 1991
:ENABle	<NRf>	1991
[:EVENTt]?		[query only] 1991
:NTRansition	<NRf>	1991
:PTRansition	<NRf>	1991
:NTRansition	<NRf> <non-decimal numeric>	1991,5
:PTRansition	<NRf> <non-decimal numeric>	1991,5
:MAP	<NRf>,<NRf>	
:NTRansition	<NRf> <non-decimal numeric>	1995
:PTRansition	<NRf> <non-decimal numeric>	1995
:VOLTage		
!:CURRent		
!:TIMe		
!:POWer		
!:TEMPerature		
!:FREQuency		
!:PHASe		
!:MODulation		
!:CALibration		
:CONDition?		[query only]
:ENABle	<NRf> <non-decimal numeric>	1995
[:EVENTt]?		[query only]
:NTRansition	<NRf> <non-decimal numeric>	1995
:PTRansition	<NRf> <non-decimal numeric>	1995

The status reporting commands are orthogonal. Status register bits are classified as either terminal (reporting a single class of events) or summary (reporting several classes of events). Summary bits have the same register structure behind them. This is called register **fan-out**. The commands to access each register are always the same and are described below. The name of each subsystem follows the name of the associated bit in the next higher status

1999 SCPI Command Reference

structure. Therefore, the event register associated with the frequency error bit of the questionable signal/data register set would be accessed by the command:

STATus:QUESTionable:FREQuency:EVENT?

Certain commands in this subsystem are required to be implemented by all SCPI instruments. These commands are described in section 4.2.1 of “Volume 1: Syntax and Style.”

The parameters to the commands are described as <NRf> as defined in *IEEE 488.2* and not as SCPI’s <numeric_value>. This means that UP, DOWN, MINimum, and MAXimum are not accepted for these commands. When a STATus command is queried, the return form shall always be an <NR1> value.

The following commands can be applied to all SCPI registers by prefixing the command with the node(s) that represent the particular register to be controlled.

20.1 :OPERation

STATus:OPERation

The OPERation status register contains conditions which are part of the instrument’s normal operation.

See Volume 1 Section 9.3 for more detailed information.

20.1.1 :BIT<n>

STATus:OPERation:BIT<n>

This command accesses the user-definable bits in the OPERation register set. The value of <n> is restricted from 8 to 12 and represents bits 8 through 12 in the :STATus:OPERation status register. This allows standardized access to user-definable terminal and summary bits.

20.1.2 :CONDition?

STATus:OPERation:CONDition?

Returns the contents of the condition register associated with the status structure defined in the command. Reading the condition register is nondestructive.

The response is (NR1 NUMERIC RESPONSE DATA) (range: 0 through 32767) unless changed by the :FORMat:SREGister command.

20.1.3 :ENABle <NRf> | <non-decimal numeric>

STATus:OPERation:ENABle

Sets the enable mask which allows true conditions in the event register to be reported in the summary bit. If a bit is 1 in the enable register and its associated event bit transitions to true, a positive transition will occur in the associated summary bit.

The command accepts parameter values of either format in the range 0 through 65535 (decimal) without error.

The query response format is <NR1> unless changed by the :FORMat:SREGister command. Note that 32767 is the maximum value returned as the most-significant bit of the register cannot be set true.

20.1.4 **[[:EVENT]?]**

STATus:OPERation:EVENT?

This query returns the contents of the event register associated with the status structure defined in the command.

The response is (NR1 NUMERIC RESPONSE DATA) (range: 0 through 32767) unless changed by the :FORMat:SREGister command.

Note that reading the event register clears it.

20.1.5 **:MAP <NRf>,<NRf>**

STATus:OPERation:MAP

Maps an error from the list of possible events the instrument can generate into a specified bit in the OPERation or QUEStionable register. The first parameter is the specified bit location in the OPERation or QUEStionable register. The second parameter is the event number. For example, to map event number 678 into bit 9 of the QUEStionable status register, the programmer would send:

```
STATus:QUEStionable:MAP 9,678
```

20.1.6 **:NTRansition <NRf> I <non-decimal numeric>**

STATus:OPERation:NTRansition

Sets the negative transition filter. Setting a bit in the negative transition filter shall cause a 1 to 0 transition in the corresponding bit of the associated condition register to cause a 1 to be written in the associated bit of the corresponding event register.

The command accepts parameter values of either format in the range 0 through 65535 (decimal) without error.

The query response format is <NR1> unless changed by the :FORMat:SREGister command. Note that 32767 is the maximum value returned as the most-significant bit of the register cannot be set true.

20.1.7 **:PTRansition <NRf> I <non-decimal numeric>**

STATus:OPERation:PTRansition

Sets the positive transition filter. Setting a bit in the positive transition filter shall cause a 0 to 1 transition in the corresponding bit of the associated condition register to cause a 1 to be written in the associated bit of the corresponding event register.

The command accepts parameter values of either format in the range 0 through 65535 (decimal) without error.

The query response format is <NR1> unless changed by the :FORMat:SREGister command. Note that 32767 is the maximum value returned as the most-significant bit of the register cannot be set true.

20.2 **:PRESet**

STATus:PRESet

The PRESet command is an event that configures the SCPI and device-dependent status data structures such that device-dependent events are reported at a higher level through the

mandatory part of the status-reporting mechanism. Device-dependent events are summarized in the mandatory structures. The mandatory structure is defined in part by *IEEE 488.2*; SCPI-required structures compose the rest. The mandatory part of the status-reporting mechanism provides a device-independent interface for determining the gross status of a device.

The PRESet command affects only the enable register, the transition filter registers, and queue enabling for the SCPI-mandated and device-dependent status data structures. PRESet does not affect either the “status byte” or the “standard event status” as defined by *IEEE 488.2*. PRESet does not clear any of the event registers or any item from the error/event queue. The *CLS command is used to clear all event registers and queues in the device status-reporting mechanism.

For the device-dependent status data structures, the PRESet commands shall set the enable register to all 1's and the transition filter register to a device-dependent value. The device-dependent value shall be either PTR, NTR, or both, such that the device-dependent event shall be propagated to the structure that summarizes that event. For the SCPI mandatory status data structures, the PRESet command shall set the transition filter registers to recognize only positive transitions and set the enable register to 0's. The PRESet command shall also set error/event queue enabling to report only errors.

An application may select the events that shall cause a service request at the mandatory SCPI status data structure level, rather than excluding those that are of no interest to the application. Further, there is no requirement on the application to set up the device-dependent structures for normal device operation.

For this concept to be extended to encompass the *IEEE 488.2*-mandated status data structures, the following sequence of commands is required when initializing the device:

- *CLS – Clears all event status registers and queues
- *SRE 0 – Clears the *IEEE 488.2*-mandated service request enable register
- *ESE 0 – Clears the *IEEE 488.2*-mandated standard event status enable register
- STATus:PRESet – Presets all other registers and queues

1999 SCPI Command Reference

The table below indicates the effects of various commands upon the status data structures in a device.

	SCPI Transition Filters	SCPI Enable Registers	SCPI Event Registers	SCPI Error/Event Queue Enable	SCPI Error/Event Queue	IEEE 488.2 Registers ESE SRE	IEEE 488.2 Registers SESR STB
*RST	none	none	none	none	none	none	none
*CLS	none	none	clear	none	clear	none	clear
power-on	preset [#]	preset [#]	clear [#]	preset [#]	clear [#]	clear [#]	clear [#]
STATus:PRESet	preset	preset	none	preset	none	none	none

[#]Occurs if the power-on state clear flag is true. Effect changes to none if the power-on state clear flag is false.

The following table defines the effect of STATus:PRESet

Register	Filter/enable	PRESet value =
OPERational	ENABle	0's
	PTR	1's
	NTR	0's
QUESTionable	ENABle	0's
	PTR	1's
	NTR	0's
ISUMmary	ENABle	0's
	PTR	1's
	NTR	0's
INSTrument	ENABle	1's
	PTR	device-dependent [#]
	NTR	device-dependent [#]
All others	ENABle	1's
	PTR	device-dependent [#]
	NTR	device-dependent [#]

[#] Requires either PTR, NTR, or both to be set.

20.3 :QUESTionable

STATus:QUESTionable?

The QUESTionable status register set contains bits which give an indication of the quality of various aspects of the signal.

A bit set in the condition register indicates that the data currently being acquired or generated is of questionable quality due to some condition affecting the parameter associated with that bit. For example, if the FREQ bit were set, this would mean that the frequency accuracy of the signal was of questionable quality.

See Volume 1 Section 9.4 for more detailed information.

20.3.1 :BIT<n>

:STATus:QUESTionable:BIT<n>

This command accesses the user-definable bits in the QUESTionable register set. The value of <n> is restricted from 9 to 12 and represents bits 9 through 12 in the :STATus:QUESTionable status register. This allows standardized access to user-definable terminal and summary bits.

Note that this definition is different from that of :STATus:OPERation:BIT in that the bit number is restricted to the range of 9 through 12, while the range for :STATus:OPERation:BIT is restricted to 8 through 12.

20.3.2 :CONDition?

STATus:QUESTionable:CONDition?

Defined the same as STATus:OPERation:CONDition. See Section 20.1.2 for details.

20.3.3 :ENABLE <NRf> I <non-decimal numeric>

STATus:QUESTionable:ENABle

Defined the same as STATus:OPERation:ENABle. See Section 20.1.3 for details.

20.3.4 [:EVENT]?

STATus:QUESTionable:EVENT?

Defined the same as STATus:OPERation:EVENT. See Section 20.1.4 for details.

20.3.5 :MAP <NRf>, <NRf>

STATus:QUESTionable:MAP

Defined the same as STATus:OPERation:MAP. See Section 20.1.5 for details.

20.3.6 :NTRansition <NRf> I <non-decimal numeric>

STATus:QUESTionable:NTRansition

Defined the same as STATus:OPERation:NTRansition. See Section 20.1.6 for details.

20.3.7 :PTRansition <NRf> I <non-decimal numeric>

STATus:QUESTionable:PTRansition

Defined the same as STATus:OPERation:PTRansition. See Section 20.1.7 for details.

21 SYSTem Subsystem

The SYSTem subsystem collects the functions that are not related to instrument performance. Examples include functions for performing general housekeeping and functions related to setting global configurations, such as TIME or SECURITY.

KEYWORD	PARAMETER FORM	NOTES
SYSTem		
:ALTernate	<numeric_value>	
:STATe	<Boolean>	
:BEEPer		
:FREQuency	<numeric_value>	
[:IMMediate]	[<frequency>[,<time>[,<volume>]]]	
:STATe	<Boolean>	
:TIME	<numeric_value>	
:VOLume	<numeric_value>	
:CAPability?		[query only] 1994
:COMMunicate		
:CENTronics		1993
:FEED	<data_handle>	1993
:GPIB		
RDEvice		1993
:ADDRes	<numeric_value>[,<numeric_value>]	1993
:FEED	<data_handle>	1993
[:SELF]		1993
:ADDRes	<numeric_value>[,<numeric_value>]	1993
:SERial		
:CONTrol		1991
:DTR	ON OFF STANdard IBFull	1991
:RTS	ON OFF STANdard IBFull RFR	1991
:FEED	<data_handle>	1993
[:RECeive]		
:BAUD	<numeric_value>	
:BITS	<numeric_value>	
:PACE	XON ACK NONE	
:THReshold		1991
:STARt	<numeric_value>	1991
:STOP	<numeric_value>	1991
:PARity		
:CHECK	<Boolean>	
[:TYPE]	EVEN ODD ZERO ONE NONE	
:SBITs	<numeric_value>	
:TRANsmit		
:AUTO	<Boolean>	
:BAUD	<numeric_value>	

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
:BITS	<numeric_value>	
:DElay	<numeric_value>	1993
:PACE	XON ACK NONE	
:PARity		
[:TYPE]	EVEN ODD ZERO ONE NONE	
:SBITs	<numeric_value>	
:SOCKet	<n>	1999
:ADDRess	<string> (IPaddress Host name)	1999
:CONNect		1999
:DISConnect		1999
:FEED	<data_handle>{,<data_handle>}	1999
:OCONdition	<event_handle>	1999
:SCONdition	<event_handle>	1999
:LISTen		1999
:PORT	<numeric_value> (port #)	1999
:TYPE	TCP UDP	1999
:CPON	<card_destination> ALL	[event; no query]
:CTYPe?	<card_destination>	[query only]
:DATE	<numeric_value>,<numeric_value>,<numeric_value>	
:ERRor		
:ALL?		[query only] 1996
:CODE		1996
:ALL?		[query only] 1996
[:NEXT]?		[query only] 1996
:COUNT?		[query only] 1996
:ENABle		1999
:ADD	<numeric_list>	[no query] 1999
:DELeTe	<numeric_list>	[no query] 1999
[:LIST]	<numeric_list>	1999
[:NEXT]?		[query only] 1996
:HELP		1992
:HEADers?		[query only] 1995
:SYNTax?	<command_header>	[query only] 1992
:KEY	<numeric_value>	
:CATalog		
:DEFine	<numeric_value>,<block>[,<string>]	
:DELeTe		
:KLOCK	<Boolean>	
:LANGuage	<string>	
:LFRequency	<numeric_value>	1992
:AUTO	<Boolean> ONCE	1992
:LOCK		1999
:OWNer?		[query only] 1999
:RELease		[no query] 1999

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
:REQuest?		[event; query only]1999
:PASSword		1994
:CDISable	<password>	[event; no query]1994
[:CENable]	<password>	[event; no query]1994
:STATe?		[query only] 1994
:NEW	<current_password>,<new_password>	[event; no query]1994
:PRESet		[event; no query]
:SECurity		
:IMMediate		[event; no query]1994
[:STATe]	<Boolean>	
:SET	<block data>	
:TIME	<numeric_value>,<numeric_value>,<numeric_value>	
:TIMer		1999
:COUNt	<numeric_value>	1999
[:STATe]	<Boolean>	1999
:TZONe	<numeric_value>[,<numeric_value>]	
:VERSion?		

21.1 :ALTErnate <numeric_value>

SYSTem:ALTErnate

Selects an alternate instrument state, which is swapped with the current state at the end of some specified action. The <numeric_value> selects which of the states will be used. The states are created with the *SAV command. The action is generally sweep, list sequencing or some programmable delay, depending on the device configuration.

At *RST, this value is device-dependent.

21.1.1 :STATe <Boolean>

SYSTem:ALTErnate:STATe

Controls whether the alternate state mode is enabled. While it is ON, the device shall swap the active state with the one stored in the register specified in ALTErnate.

At *RST, this parameter is set to OFF.

21.2 :BEEPer

SYSTem:BEEPer

This subsystem controls the audible beeper of the instrument.

21.2.1 :FREQuency <numeric_value>

SYSTem:BEEPer:FREQuency

Programs the frequency of audible tones. It also sets the default frequency for the [:IMMediate] command. The parameter is specified in Hz. The range of valid parameters is instrument-dependent.

At *RST, this value is device-dependent.

21.2.2 **[[:IMMediate][<frequency>[,<time>[,<volume>]]]**

SYSTem:BEEPer:IMMediate

The receipt of this command causes an audible tone to be generated by the instrument. Note that this command generates an event and therefore it has no associated *RST state or query form.

The allowable values of the parameters <frequency>, <time> and <volume> are instrument-dependent. The frequency parameter is specified in Hz. The duration time parameter is specified in seconds. The volume parameter range is from 0 to 1, with 1 indicating the maximum volume that the instrument can produce and 0 indicating minimum volume.

21.2.3 **:STATe <Boolean>**

SYSTem:BEEPer:STATe

Enables/disables the beeper. When STATe OFF is selected, no instrument condition, except the :SYSTem:BEEPer:IMMediate command, shall cause an audible beep to be emitted.

At *RST, this value is device-dependent.

21.2.4 **:TIME <numeric_value>**

SYSTem:BEEPer:TIME

Programs the duration of audible tones. It also sets the default duration for the [:IMMediate] command. The parameter is specified in seconds. The range of valid parameters is instrument-dependent.

At *RST, this value is device-dependent.

21.2.5 **:VOLume <numeric_value>**

SYSTem:BEEPer:VOLume

Programs the volume of audible tones. It also sets the default volume for the [:IMMediate] command. The parameter range is from 0 to 1, with 1 indicating the maximum volume that the instrument can produce and 0 indicating minimum volume. Actual sound pressure levels generated by any particular volume setting are instrument-dependent.

At *RST, this value is device-dependent.

21.3 **:CAPability?**

SYSTem:CAPability?

This query returns an <instrument_specifier>. See the Compliance section in the introduction to Instrument Class Applications.

21.4 **:COMMunicate**

SYSTem:COMMunicate

The COMMunicate subsystem collects together the configuration of various control/communication interfaces.

21.4.1 **:CENTronics**

SYSTem:COMMunicate:CENTronics

The CENTronics subsystem sets the physical configuration of a centronics port.

21.4.1.1 :FEED <data_handle>

SYSTem:COMMunicate:CENTronics:FEED

Sets or queries what data is used to FEED the remote device. Data may be streamed to an external device without sending it through the HCOPy subsystem, for example by directly feeding SENSE or CALCulate data, where no page formatting commands are needed. That is, the SYSTem:COMMunicate subsystem shall not add any page formatting to the output stream, where this is required HCOPy should be used.

At *RST, FEED is device dependent.

21.4.2 :GPIB

SYSTem:COMMunicate:GPIB

The GPIB subsystem controls the physical configuration of a GPIB (*IEEE 488*) port.

21.4.2.1 :RDEvice

SYSTem:COMMunicate:GPIB:RDEvice

The Remote DEvice subsystem configures how the instrument uses a peripheral device, such as a printer or plotter, connected to a GPIB port.

21.4.2.1.1 :ADDRESS <numeric_value>[,<numeric_value>]

SYSTem:COMMunicate:GPIB:RDEvice:ADDRESS

Sets or queries the bus address of the peripheral device. The first parameter is the primary address and the optional second parameter is the secondary address. Changing the address with this does not affect the address of the peripheral device. It does affect the address to which data is sent by the instrument.

At *RST, the value of this parameter is device dependent.

21.4.2.1.2 :FEED <data_handle>

SYSTem:COMMunicate:GPIB:RDEvice:FEED

Sets or queries what data is used to FEED the peripheral device. Data may be streamed to an external device without sending it through the HCOPy subsystem, for example by directly feeding SENSE or CALCulate data, where no page formatting commands are needed. That is, the SYSTem:COMMunicate subsystem shall not add any page formatting to the output stream. Where page formatting is required, HCOPy should be used.

At *RST, FEED is device dependent.

21.4.2.2 [:SELF]

SYSTem:COMMunicate:GPIB[:SELF]

The SELF subsystem sets the configuration relating to the GPIB port on the instrument itself.

21.4.2.2.1 :ADDRESS <numeric_value>[,<numeric_value>]

SYSTem:COMMunicate:GPIB[:SELF]:ADDRESS

Sets the GPIB address of the instrument. The first parameter is the primary address and the optional second parameter is the secondary address.

*RST has no effect on the value of these parameters.

21.4.3 :SErial

SYSTem:COMMunicate:SErial

Controls the physical configuration of a serial port, such as RS-232-D, RS-422, or EIA/TIA-562. Any command that changes the configuration of the port takes effect immediately upon receipt of the Program Message Terminator.

*RST has no effect on the value of these parameters.

21.4.3.1 :CONTRol

SYSTem:COMMunicate:SErial:CONTRol

Add command which independently sets the mode in which DTR and RTS pacing operate. Since hardware handshaking can be configured to pertain to either transmit, receive, or both, a CONTRol branch separate from the transmit and receive branches was added.

21.4.3.1.1 :DTR ONIOFFISTANdardIIBFull

SYSTem:COMMunicate:SErial:CONTRol:DTR

Sets the hardware pacing scheme.

The ON parameter indicates that the DTR line should always be asserted (held in the ON condition).

The OFF parameter indicates that the DTR line should always be unasserted (held in the OFF condition).

When the STANdard option is set for DTR, the DTR line behaves as specified in the EIA standard RS-232-D section 4.4. As stated in the standard:

“Signals on this circuit are used to control the switching of the DCE to the communication channel. The ON condition prepares the DCE to be connected to the communication channel and maintains the connection established by external means.”

A common use of the DTR line that has evolved is that of pacing. The IIBFull parameter sets the DTR line to indicate when the device is ready to receive. When the number of received bytes in the input buffer of the device reaches the STOP threshold the device will de-assert the DTR line. When the number of bytes has been reduced to the STARt threshold, the device will assert DTR indicating that it can receive input again. The device will also monitor the state of DSR and CTS and will stop transmission if either of those lines becomes de-asserted.

*RST has no effect on the value of this parameter.

21.4.3.1.2 :RTS ONIOFFISTANdardIIBFullIRFR

SYSTem:COMMunicate:SErial:CONTRol:RTS

Sets the hardware pacing (hand-shaking) scheme.

The ON parameter indicates that the RTS line should always be asserted.

The OFF parameter indicates that the RTS line should always be unasserted.

When the STANdard option is set for RTS, the RTS line behaves as specified in the EIA standard RS-232-D section 4.4. As stated in the standard:

“This circuit is used to condition the local DCE for data transmission and on a half-duplex channel, to control the direction of data transmission of the local DCE.”

A common use of the RTS line that has evolved is that of pacing. Many high speed asynchronous modems use this line (paired with CTS) as receive/transmit pacing. The IBFull or its alias RFR (Ready For Receiving) parameter sets the RTS line to indicate when the device is ready to receive. When the number of received bytes in the input buffer of the device reaches the STOP threshold the device will de-assert the RTS line. When the number of bytes has been reduced to the START threshold, the device will assert RTS, indicating that it can receive input again. The device will also monitor the state of CTS and will stop transmission if that line becomes de-asserted.

The IBFull (RFR) mode follows the CCITT Recommendations V.42, Section 7.3.1, “Flow Control across the DTE/DCE interface, Option a) Using Circuit 133 and 106.”

*RST has no effect on the value of this parameter.

21.4.3.2 **:FEED <data_handle>**

SYSTem:COMMunicate:SErIAL:FEED

Sets or queries what data is used to FEED the peripheral device connected to the serial port. Data may be streamed to an external device without sending it through the HCOPY subsystem, for example by directly feeding SENSE or CALCulate data, where no page formatting commands are needed. That is, the SYSTem:COMMunicate subsystem shall not add any page formatting to the output stream, where this is required HCOPY should be used.

At *RST, FEED is device dependent.

21.4.3.3 **[[:RECeive]**

SYSTem:COMMunicate:SErIAL:RECeive

The optional node RECeive is used to affect the parameters associated with reception.

21.4.3.3.1 **:BAUD <numeric_value>**

SYSTem:COMMunicate:SErIAL:RECeive:BAUD

Sets the baud rate.

*RST has no effect on the value of this parameter.

21.4.3.3.2 **:BITS <numeric_value>**

SYSTem:COMMunicate:SErIAL:RECeive:BITS

Sets the number of data bits, typically 7 or 8. The value shall be rounded to the nearest integer.

*RST has no effect on the value of this parameter.

21.4.3.3.3 **:PACE XONIACKINONE**

SYSTem:COMMunicate:SErIAL:RECeive:PACE

This command sets the software pacing scheme.

The XON pacing mode follows the CCITT Recommendations V.42, Section 7.3.1, “Flow Control across the DTE/DCE interface, Option b) Using DC1/DC3 characters (XON/XOFF functions).”

*RST has no effect on the value of this parameter.

21.4.3.3.3.1 :THReshold

SYSTem:COMMunicate:SERial:RECeive:PACe:THReshold

This command sets the receive pacing threshold.

21.4.3.3.3.1.1 :STARt <numeric_value>

SYSTem:COMMunicate:SERial:RECeive:PACe:THReshold:STARt

This command indicates that when the number of characters in the device's input buffer goes to or below the indicated amount and some form of pacing has been set, the device should indicate that it is ready to receive.

*RST has no effect on the value of this parameter.

21.4.3.3.3.1.2 :STOP <numeric_value>

SYSTem:COMMunicate:SERial:RECeive:PACe:THReshold:STOP

This command indicates that when the number of characters in the device's input buffer goes to or above the indicated amount and some form of pacing has been set, the device should indicate that it is not ready to receive.

*RST has no effect on the value of this parameter.

21.4.3.3.4 :PARity

SYSTem:COMMunicate:SERial:RECeive:PARity

This subsystem controls the parity of the receiving channel

21.4.3.3.4.1 :CHECK <Boolean>

SYSTem:COMMunicate:SERial:RECeive:PARity:CHECK

When CHECK ON is selected, the parity shall be checked in accordance with the selected PARity:TYPE.

When CHECK is OFF, the parity TYPE setting (NONE or "other than NONE") still determines the expectation of not receiving or receiving a parity bit.

*RST has no effect on the value of this parameter.

21.4.3.3.4.2 [:TYPE] EVEN|ODD|ZERO|ONE|NONE|IGNore

SYSTem:COMMunicate:SERial:RECeive:PARity:TYPE

Sets the parity scheme that is to be employed for reception.

*RST has no effect on the value of this parameter.

21.4.3.3.5 :SBITs <numeric_value>

SYSTem:COMMunicate:SERial:RECeive:SBITs

Sets the number of stop bits, typically 1, 1.5 or 2 bits.

*RST has no effect on the value of this parameter.

21.4.3.4 :TRANsmit

SYSTem:COMMunicate:SERial:TRANsmit

Affects the parameters associated with transmission.

21.4.3.4.1 :**AUTO** <Boolean>

SYSTem:COMMunicate:SERIal:TRANsmit:AUTO

Couples the TRANsmit parameter values to the RECEive parameter values when AUTO ON is selected.

At *RST, AUTO ON is selected.

21.4.3.4.2 :**BAUD** <numeric_value>

SYSTem:COMMunicate:SERIal:TRANsmit:BAUD

This command sets the baud rate.

*RST has no effect on the value of this parameter.

21.4.3.4.3 :**BITS** <numeric_value>

SYSTem:COMMunicate:SERIal:TRANsmit:BITS

Sets the number of data bits, typically 7 or 8. The value shall be rounded to the nearest integer.

*RST has no effect on the value of this parameter.

21.4.3.4.4 :**DELAY** <numeric_value>

SYSTem:COMMunicate:SERIal:TRANsmit:DELAY

Sets the minimum delay in seconds that the device must wait after receipt of a Program Message Terminator before issuing a response message. The delay does not apply to software pacing or to other low level protocols.

*RST has no effect on the value of this parameter.

21.4.3.4.5 :**PACE XONIACKINONE**

SYSTem:COMMunicate:SERIal:TRANsmit:PACE

Sets the software pacing scheme.

The XON pacing mode follows the CCITT Recommendations V.42, Section 7.3.1, “Flow Control across the DTE/DCE interface, Option b) Using DC1/DC3 characters (XON/XOFF functions).”

*RST has no effect on the value of this parameter.

21.4.3.4.6 :**PARity**

SYSTem:COMMunicate:SERIal:TRANsmit:PARity

This subsystem controls the parity of the transmitting channel

21.4.3.4.6.1 [:**TYPE**] **EVEN|ODD|ZERO|ONE|NONE**

SYSTem:COMMunicate:SERIal:TRANsmit:PARity:TYPE

Sets the parity scheme that is to be employed for transmission.

*RST has no effect on the value of this parameter.

21.4.3.4.7 :**SBITS** <numeric_value>

SYSTem:COMMunicate:SERIal:TRANsmit:SBITS

Sets the number of stop bits, typically 1, 1.5, or 2 bits.

*RST has no effect on the value of this parameter.

21.4.4 :SOCKet <n>

:SYSTem:COMMunicate:SOCKet<n>

This command identifies the socket to be opened. The syntax of the numeric suffix is as specified in 6.2.5.2 in Volume 1, Syntax and Style.

21.4.4.1 :ADDRess <string>

:SYSTem:COMMunicate:SOCKet:ADDRess

This command allows the user to enter the IP address or host name of the remote device. If it is an IP address, it is specified using IP dot notation (e.g. 123.45.67.89). If it is a host name, it is specified by using the host name string (e.g., my_host).

21.4.4.2 :CONNect

:SYSTem:COMMunicate:SOCKet:CONNect

This command implements the socket connection.

21.4.4.3 :DISConnect

:SYSTem:COMMunicate:SOCKet:DISConnect

The Disconnect command breaks the socket connection.

21.4.4.4 :FEED <n> <data_handle>{,<data_handle>}

:SYSTem:COMMunicate:SOCKet:FEED

Sets or queries what data is used to FEED the communications socket.

Note: There can be multiple FEEDs for the same socket.

21.4.4.4.1 :OCONdition <event_handle>

:SYSTem:COMMunicate:SOCKet:FEED:OCONdition

The On Condition command, :OCONdition, defines the rate at which data will be collected and sent to the socket FEED.

21.4.4.4.2 :SCONdition <event_handle>

:SYSTem:COMMunicate:SOCKet:FEED:SCONdition

The Send Condition, SCONdition, defines the rate at which the data will be transmitted on the socket.

21.4.4.5 :LISTen

:SYSTem:COMMunicate:SOCKet:LISTen

The LISTen command instructs the device (instrument) to listen for a connection on the specified socket. This will allow data channels from one instrument to another instrument to be opened.

21.4.4.6 :PORT <numeric_value>

:SYSTem:COMMunicate:SOCKet:PORT

This command allows the user to enter the port number. This parameter is a 16 bit integer.

21.4.4.7 :TYPE TCPIUDP

:SYSTem:COMMunicate:SOCKet :TYPE

This command allows the user to select the type of Ethernet port to be opened, either TCP or UDP. The query form will return either TCP or UDP.

21.5 :CPON <card_destination>|ALL

SYSTem:CPON

Card Power ON resets a card within a cardcage to its power-on state. The parameter can either be a module number or a user-defined card name. If the parameter is ALL, then all addressable cards are reset to their power-on state.

SYSTem:CPON is an event; no query is allowed.

21.6 :CTYPE? <card_destination>

SYSTem:CTYPE?

The Card TYPE query identifies a card within a cardcage. The response is identical in form to that defined for the “*IDN?” query in *IEEE 488.2*.

21.7 :DATE <year>,<month>,<day>

SYSTem:DATE

Any instrument which has an internal calendar shall implement the following command to set the calendar:

<year>: Must be <numeric_value>. It is rounded to the nearest integer. Its range is limited by the capability of the instrument. The year shall be entered as a four-digit number, including century and millennium information.

<month>: Must be <numeric_value>. It is rounded to the nearest integer. Its range is 1 to 12 inclusive. The number 1 corresponds to the month January, 2 to February, and so on.

<day>: Must be <numeric_value>. It is rounded to the nearest integer. Its range is 1 to number of days in the month from the previous parameter. A device implementing this command shall keep track of the number of days in each month, accounting for leap years through the range of years that it accepts.

The response message shall consist of three fields separated by commas:

<year>,<month>,<day>

<year>: NR1 format. Range is device-dependent.

<month>: NR1 format. Range is 1 to 12.

<day>: NR1 format. Range is 1 to 31.

This shall not be affected by a *RST command.

21.8 :ERROR Subsystem

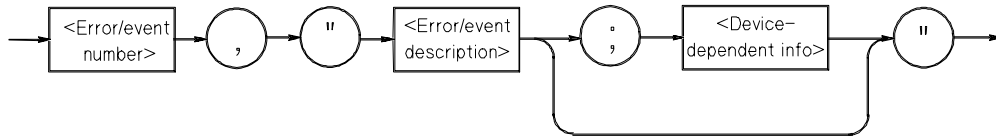
SYSTem:ERROR

This subsystem collects commands and queries relating to the reading and control of the

1999 SCPI Command Reference

error/event queue.

The error/event queue contains items that include a numerical and textual description of the error or event. Querying for the full queue item (for example, with `SYSTem:ERRor[:NEXT]?()`) will return a response with the following syntax:



The `<Error/event_number>` is a unique integer in the range `[-32768, 32767]`. All positive numbers are instrument-dependent. All negative numbers are reserved by the SCPI standard with certain standard error/event codes described in this document. The value, zero, is also reserved to indicate that no error or event has occurred.

The second parameter of the full response is a quoted string containing an `<Error/event_description>` followed by optional `<Device-dependent info>` text. Each `<Error/event_number>` has a unique and fixed `<Error/event_description>` associated with it.

The maximum string length of `<Error/event_description>` plus `<Device-dependent_info>` is 255 characters. For standard defined error/event codes, the `<Error/event_description>` shall be sent exactly as indicated in this document including case.

The `<Device-dependent_info>` text is optional but may be useful in some instruments to provide additional information allowing the user to determine the exact error or event and its context. If used, it is set off from the `<Error/event_description>` portion of the string with a semicolon. For example:

```
-131,"Invalid suffix;FREQuency:CENT 2.0E+5 dBuV"
```

Device-dependent information may be defined in greater detail for specific instrument classes in volume 4.

Where Date and Time are desired to be returned with the error message, they should be appended to the `<device_dependent_information>` separated by a semi-colon using the following format:

`<date><space><time>`

date - `yyyy/mm/dd` - year/month/day

time - `hh:mm:ss.sss` - 24 hour time

Example:

```
-200, "Execution error;Torque Safety Limit  
Exceeded;1996/08/15 13:22:51.01"
```

21.8.1 The Error/Event Queue

As errors and events are detected, they are placed in a queue. This queue is first in, first out. If the queue overflows, the last error/event in the queue is replaced with error

-350, "Queue overflow"

Any time the queue overflows, the least recent errors/events remain in the queue, and the most recent error/event is discarded. The minimum length of the error/event queue is 2, one position for the first error/event, and one for the "Queue overflow" message. Reading an error/event from the head of the queue removes that error/event from the queue, and opens a position at the tail of the queue for a new error/event, if one is subsequently detected.

When all errors/events have been read from the queue, further error/event queries shall return

0, "No error"

After a STATUS:PRESet, the masking shall be changed so that only errors are reported. Below is an overview of how the ERRor system works.

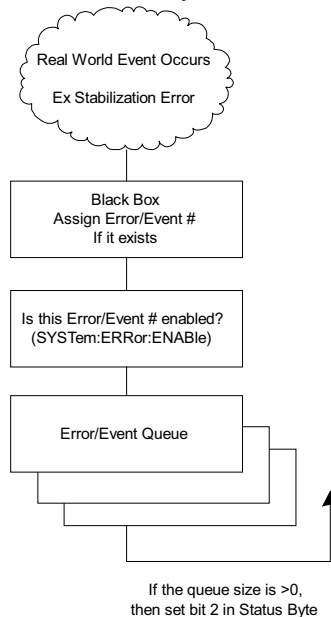


Figure 21.1 System Error Format

The error/event queue shall be cleared when any of the following occur (*IEEE 488.2*, section 11.4.3.4):

- Upon receipt of a *CLS command.
- Upon reading the last item from the queue.

21.8.2 Error/Event numbers

The system-defined error/event numbers are chosen on an enumerated (“1 of N”) basis. The SCPI-defined error/event numbers and the <error/event_description> portions of the full queue item are listed here. The first error/event described in each class (for example, -100, -200, -300, -400) is a “generic” error. In selecting the proper Error/event number to report, more specific error/event codes are preferred, and the generic error/event is used only if the others are inappropriate.

Note the organization of the following tables. A “simple-minded” parser might implement only the XX0 errors, and a smarter one might implement all of them. A “smart and friendly” parser might use the instrument-dependent part of the error/event message string to point out the offending part of the command.

21.8.3 No Error

This message indicates that the device has no errors or events to report.

Error Error Description [description/explanation/examples]

Number

0	No error [The queue is completely empty. Every error/event in the queue has been read or the queue was purposely cleared by power-on, *CLS, etc.]

21.8.4 ALL?

SYSTem:ERRor:ALL? queries the error/event queue for all the unread items and removes them from the queue. The response is a comma separated list of number, string pairs in FIFO order. If the queue is empty, the response is 0,"No error"

Note: If the queue is not empty, the 0,"No error" is not sent.

21.8.5 CODE

21.8.5.1 ALL?

SYSTem:ERRor:CODE:ALL? queries the error/event queue for all the unread items and removes them from the queue. The response returns a comma separated list of only the error/event code numbers in FIFO order. If the queue is empty, the response is 0.

Note: If the queue is not empty, the 0 is not sent.

21.8.5.2 [NEXT]?

SYSTem:ERRor:CODE[:NEXT]? queries the error/event queue for the next item and removes it from the queue. The response returns only the error/event code number omitting the string. Except for the shortened response, the query operates identically to SYSTem:ERRor[:NEXT]?

21.8.6 COUNT?

SYSTem:ERRor:COUNt? queries the error/event queue for the number of unread items. As errors and events may occur at any time, more items may be present in the queue at the time it is actually read.

Note: If the queue is empty, the response is 0.

21.8.7 :ENABLE

SYSTem:ERRor:ENABle

The ENABLE subsystem sets the errors and events that will be allowed into the queue.

At *RST, this list is not changed.

21.8.7.1 :ADD <numeric list>

SYSTem:ERRor:ENABle:ADD

The events included in the numeric list are added to the previous ENABLE list. The syntax of <numeric list> is the same as SYSTem:ERRor:ENABle:LIST.

This command is an event and cannot be queried.

21.8.7.2 :DELeTe <numeric list>

SYSTem:ERRor:ENABle:DELeTe

The events included in the numeric list are deleted from the previous ENABLE. The syntax of <numeric list> is the same as SYSTem:ERRor:ENABle:LIST.

This command is an event and cannot be queried.

21.8.7.3 [:LIST] <numeric list>

SYSTem:ERRor:ENABle[:LIST]

The list indicates which errors/events are allowed to enter the queue. The list allows a comma separated list or a range of numbers. In the case of a range, the end points are inclusive. For example, SYSTem:ERRor:ENABle (-123,225:227) will allow events numbered -123, 225, 226, and 227 to enter the queue.

At *RST, this list is not changed.

21.8.8 [NEXT]?

SYSTem:ERRor[:NEXT]

SYSTem:ERRor[:NEXT]? queries the error/event queue for the next item and removes it from the queue. The response returns the full queue item consisting of an integer and a string as described in the introduction to the SYSTem:ERRor subsystem.

This command is required of all SCPI implementations (See Syntax and Style, 4.2.1.

21.8.9 Command Error

An <error/event number> in the range [-199 , -100] indicates that an *IEEE 488.2* syntax error has been detected by the instrument's parser. The occurrence of any error in this class shall cause the command error bit (bit 5) in the event status register (*IEEE 488.2*, section 11.5.1) to be set. One of the following events has occurred:

- An *IEEE 488.2* syntax error has been detected by the parser. That is, a controller-to-device message was received which is in violation of the *IEEE 488.2* standard. Possible violations include a data element which violates the device listening formats or whose type is unacceptable to the device.
- An unrecognized header was received. Unrecognized headers include incorrect device-specific headers and incorrect or unimplemented *IEEE 488.2* common commands.

1999 SCPI Command Reference

- A Group Execute Trigger (GET) was entered into the input buffer inside of an *IEEE 488.2* <PROGRAM MESSAGE>.

Events that generate command errors shall not generate execution errors, device-specific errors, or query errors; see the other error definitions in this chapter.

Error Number	Error Description [description/explanation/examples]
-100	Command error [This is the generic syntax error for devices that cannot detect more specific errors. This code indicates only that a Command Error as defined in <i>IEEE 488.2</i> , 11.5.1.1.4 has occurred.]
-101	Invalid character [A syntactic element contains a character which is invalid for that type; for example, a header containing an ampersand, SETUP&. This error might be used in place of errors -114, -121, -141, and perhaps some others.]
-102	Syntax error [An unrecognized command or data type was encountered; for example, a string was received when the device does not accept strings.]
-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, *EMC 1:CH1:VOLTS 5.]
-104	Data type error [The parser recognized a data element different than one allowed; for example, numeric or string data was expected but block data was encountered.]
-105	GET not allowed [A Group Execute Trigger was received within a program message (see <i>IEEE 488.2</i> , 7.7).]
-108	Parameter not allowed [More parameters were received than expected for the header; for example, the *EMC common command only accepts one parameter, so receiving *EMC 0,1 is not allowed.]
-109	Missing parameter [Fewer parameters were received than required for the header; for example, the *EMC common command requires one parameter, so receiving *EMC is not allowed.]
-110	Command header error [An error was detected in the header. This error message should be used when the device cannot detect the more specific errors described for errors -111 through -119.]

1999 SCPI Command Reference

- 111 Header separator error
[A character which is not a legal header separator was encountered while parsing the header; for example, no white space followed the header, thus *GMC"MACRO" 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 this specific **device**; for example, *XYZ is not defined for any **device**.]
- 114 Header suffix out of range
[The value of a numeric suffix attached to a program mnemonic, see Syntax and Style section 6.2.5.2, makes the header invalid.]
- 115 Unexpected number of parameters
[The number of parameters received does not correspond to the number of parameters expected. This is typically due to an inconsistency with the number of instruments in the selected group (see section on INSTRUMENT:DEFine:GROup).]
- 120 Numeric data error
[This error, as well as errors -121 through -129, are generated when parsing a data element which appears to be numeric, including the nondecimal numeric types. This particular error message should be used if the **device** 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 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 contained 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 **device** does not accept one in this position for the header.]
- 130 Suffix error
[This error, as well as errors -131 through -139, are generated when parsing a suffix. This particular error message should be used if the **device** cannot detect a more specific error.]

1999 SCPI Command Reference

- 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 this **device**.]
- 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 which does not allow suffixes.]
- 140 Character data error
[This error, as well as errors -141 through -149, are generated when parsing a character data element. This particular error message should be used if the **device** 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 **device**.]
- 150 String data error
[This error, as well as errors -151 through -159, are generated when parsing a string data element. This particular error message should be used if the **device** 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 END 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 **device** at this point in parsing.]
- 160 Block data error
[This error, as well as errors -161 through -169, are generated when parsing a block data element. This particular error message should be used if the **device** 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.]

1999 SCPI Command Reference

- 168 Block data not allowed
[A legal block data element was encountered but was not allowed by the **device** at this point in parsing.]
- 170 Expression error
[This error, as well as errors -171 through -179, are generated when parsing an expression data element. This particular error message should be used if the **device** cannot detect a more specific error.]
- 171 Invalid expression
[The expression data element was invalid (see *IEEE 488.2*, 7.7.7.2); for example, unmatched parentheses or an illegal character.]
- 178 Expression data not allowed
[A legal expression data was encountered but was not allowed by the **device** at this point in parsing.]
- 180 Macro error
[This error, as well as errors -181 through -189, are generated when defining a macro or executing a macro. This particular error message should be used if the **device** cannot detect a more specific error.]
- 181 Invalid outside macro definition
[Indicates that a macro parameter placeholder (\$<number>) was encountered outside of a macro definition.]
- 183 Invalid inside macro definition
[Indicates that the program message unit sequence, sent with a *DDT or *DMC command, is syntactically invalid (see *IEEE 488.2*, 10.7.6.3).]
- 184 Macro parameter error
[Indicates that a command inside the macro definition had the wrong number or type of parameters.]

21.8.10 Execution Error

An <error/event number> in the range [-299 , -200] indicates that an error has been detected by the instrument's execution control block. The occurrence of any error in this class shall cause the execution error bit (bit 4) in the event status register (*IEEE 488.2*, section 11.5.1) to be set. One of the following events has occurred:

- A <PROGRAM DATA> element following a header was evaluated by the device as outside of its legal input range or is otherwise inconsistent with the device's capabilities.
- A valid program message could not be properly executed due to some device condition.

Execution errors shall be reported by the device after rounding and expression evaluation operations have taken place. Rounding a numeric data element, for example, shall not be reported as an execution error. Events that generate execution errors shall not generate

1999 SCPI Command Reference

Command Errors, device-specific errors, or Query Errors; see the other error definitions in this section.

Error Number	Error String [description/explanation/examples]
-200	Execution error [This is the generic syntax error for devices that cannot detect more specific errors. This code indicates only that an Execution Error as defined in <i>IEEE 488.2</i> , 11.5.1.1.5 has occurred.]
-201	Invalid while in local [Indicates that a command is not executable while the device is in local due to a hard local control (see <i>IEEE 488.2</i> , 5.6.1.5); for example, a device with a rotary switch receives a message which would change the switches state, but the device is in local so the message can not be executed.]
-202	Settings lost due to rtl [Indicates that a setting associated with a hard local control (see <i>IEEE 488.2</i> , 5.6.1.5) was lost when the device changed to LOCS from REMS or to LWLS from RWLS.]
-203	Command protected [Indicates that a legal password-protected program command or query could not be executed because the command was disabled.]
-210	Trigger error
-211	Trigger ignored [Indicates that a GET, *TRG, or triggering signal was received and recognized by the device but was ignored because of device timing considerations; for example, the device was not ready to respond. Note: a DT0 device always ignores GET and treats *TRG as a Command Error.]
-212	Arm ignored [Indicates that an arming signal was received and recognized by the device but was ignored.]
-213	Init ignored [Indicates that a request for a measurement initiation was ignored as another measurement was already in progress.]
-214	Trigger deadlock [Indicates that the trigger source for the initiation of a measurement is set to GET and subsequent measurement query is received. The measurement cannot be started until a GET is received, but the GET would cause an INTERRUPTED error.]
-215	Arm deadlock [Indicates that the arm source for the initiation of a measurement is set to GET and subsequent measurement query is received. The measurement cannot be

1999 SCPI Command Reference

started until a GET is received, but the GET would cause an INTERRUPTED error.]

- 220 Parameter error
[Indicates that a program data element related error occurred. This error message should be used when the **device** cannot detect the more specific errors described for errors -221 through -229.]
- 221 Settings conflict
[Indicates that 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
[Indicates that 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 **device** (see *IEEE 488.2*, 11.5.1.1.5.)]
- 223 Too much data
[Indicates that a legal program data element of block, expression, or string type was received that contained more data than the device could handle due to memory or related device-specific requirements.]
- 224 Illegal parameter value
[Used where exact value, from a list of possibles, was expected.]
- 225 Out of memory. [The device has insufficient memory to perform the requested operation.]
- 226 Lists not same length. [Attempted to use LIST structure having individual LIST's of unequal lengths.]
- 230 Data corrupt or stale
[Possibly invalid data; new reading started but not completed since last access.]
- 231 Data questionable
[Indicates that measurement accuracy is suspect.]
- 232 Invalid format
[Indicates that a legal program data element was parsed but could not be executed because the data format or structure is inappropriate. For example when loading memory tables or when sending a SYSTem:SET parameter from an unknown instrument.]
- 233 Invalid version
[Indicates that a legal program data element was parsed but could not be executed because the version of the data is incorrect to the device. This particular error should be used when file or block data formats are recognized by the instrument but cannot be executed for reasons of version incompatibility. For example, a not supported file version, a not supported instrument version]

1999 SCPI Command Reference

- 240 Hardware error
[Indicates that a legal program command or query could not be executed because of a hardware problem in the **device**. Definition of what constitutes a hardware problem is completely device-specific. This error message should be used when the **device** cannot detect the more specific errors described for errors -241 through -249.]
- 241 Hardware missing
[Indicates that a legal program command or query could not be executed because of missing **device** hardware; for example, an option was not installed. Definition of what constitutes missing hardware is completely device-specific.]
- 250 Mass storage error
[Indicates that a mass storage error occurred. This error message should be used when the **device** cannot detect the more specific errors described for errors -251 through -259.]
- 251 Missing mass storage
[Indicates that a legal program command or query could not be executed because of missing mass storage; for example, an option that was not installed. Definition of what constitutes missing mass storage is device-specific.]
- 252 Missing media
[Indicates that a legal program command or query could not be executed because of a missing media; for example, no disk. The definition of what constitutes missing media is device-specific.]
- 253 Corrupt media
[Indicates that a legal program command or query could not be executed because of corrupt media; for example, bad disk or wrong format. The definition of what constitutes corrupt media is device-specific.]
- 254 Media full
[Indicates that a legal program command or query could not be executed because the media was full; for example, there is no room on the disk. The definition of what constitutes a full media is device-specific.]
- 255 Directory full
[Indicates that a legal program command or query could not be executed because the media directory was full. The definition of what constitutes a full media directory is device-specific.]
- 256 File name not found
[Indicates that a legal program command or query 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. The definition of what constitutes a file not being found is device-specific.]

1999 SCPI Command Reference

- 257 File name error
[Indicates that 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.]
- 258 Media protected
[Indicates that a legal program command or query could not be executed because the media was protected; for example, the write-protect tab on a disk was present. The definition of what constitutes protected media is device-specific.]
- 260 Expression error
[Indicates that an expression program data element related error occurred. This error message should be used when the **device** cannot detect the more specific errors described for errors -261 through -269.]
- 261 Math error in expression
[Indicates that a syntactically legal expression program data element could not be executed due to a math error; for example, a divide-by-zero was attempted. The definition of math error is device-specific.]
- 270 Macro error
[Indicates that a macro-related execution error occurred. This error message should be used when the **device** cannot detect the more specific errors described for errors -271 through -279.]
- 271 Macro syntax error
[Indicates that that a syntactically legal macro program data sequence, according to *IEEE 488.2*, 10.7.2, could not be executed due to a syntax error within the macro definition (see *IEEE 488.2*, 10.7.6.3.)]
- 272 Macro execution error
[Indicates that a syntactically legal macro program data sequence could not be executed due to some error in the macro definition (see *IEEE 488.2*, 10.7.6.3.)]
- 273 Illegal macro label
[Indicates that the macro label defined in the *DMC command was a legal string syntax, but could not be accepted by the **device** (see *IEEE 488.2*, 10.7.3 and 10.7.6.2); for example, the label was too long, the same as a common command header, or contained invalid header syntax.]
- 274 Macro parameter error
[Indicates that the macro definition improperly used a macro parameter placeholder (see *IEEE 488.2*, 10.7.3.)]
- 275 Macro definition too long
[Indicates that a syntactically legal macro program data sequence could not be executed because the string or block contents were too long for the device to handle (see *IEEE 488.2*, 10.7.6.1).]

1999 SCPI Command Reference

- 276 Macro recursion error
[Indicates that a syntactically legal macro program data sequence could not be executed because the device found it to be recursive (see *IEEE 488.2*, 10.7.6.6).]
- 277 Macro redefinition not allowed
[Indicates that a syntactically legal macro label in the *DMC command could not be executed because the macro label was already defined (see *IEEE 488.2*, 10.7.6.4).]
- 278 Macro header not found
[Indicates that a syntactically legal macro label in the *GMC? query could not be executed because the header was not previously defined.]
- 280 Program error
[Indicates that a downloaded program-related execution error occurred. This error message should be used when the **device** cannot detect the more specific errors described for errors -281 through -289. A downloaded program is used to add algorithmic capability to a **device**. The syntax used in the program and the mechanism for downloading a program is device-specific.]
- 281 Cannot create program
[Indicates that 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
[Indicates that a syntax error appears in a downloaded program. The syntax used when parsing the downloaded program is device-specific.]
- 286 Program runtime error
- 290 Memory use error
[Indicates that a user request has directly or indirectly caused an error related to memory or <data_handle>s, this is not the same as “bad” memory.]
- 291 Out of memory
- 292 Referenced name does not exist
- 293 Referenced name already exists

-294 Incompatible type
[Indicates that the type or structure of a memory item is inadequate]

21.8.11 Device-Specific Error

An <error/event number> in the range [-399 , -300] or [1 , 32767] indicates that the instrument has detected an error which is not a command error, a query error, or an execution error; some device operations did not properly complete, possibly due to an abnormal hardware or firmware condition. These codes are also used for self-test response errors. The occurrence of any error in this class should cause the device-specific error bit (bit 3) in the event status register (*IEEE 488.2*, section 11.5.1) to be set. The meaning of positive error codes is device-dependent and may be enumerated or bit mapped; the <error message> string for positive error codes is not defined by SCPI and available to the device designer. Note that the string is not optional; if the designer does not wish to implement a string for a particular error, the null string should be sent (for example, 42,""). The occurrence of any error in this class should cause the device-specific error bit (bit 3) in the event status register (*IEEE 488.2*, section 11.5.1) to be set. Events that generate device-specific errors shall not generate command errors, execution errors, or query errors; see the other error definitions in this section.

Error Number	Error String [description/explanation/examples]

-300	Device-specific error [This is the generic device-dependent error for devices that cannot detect more specific errors. This code indicates only that a Device-Dependent Error as defined in <i>IEEE 488.2</i> , 11.5.1.1.6 has occurred.]
-310	System error [Indicates that some error, termed “system error” by the device, has occurred. This code is device-dependent.]
-311	Memory error [Indicates some physical fault in the device’s memory, such as parity error.]
-312	PUD memory lost [Indicates that the protected user data saved by the *PUD command has been lost.]
-313	Calibration memory lost [Indicates that nonvolatile calibration data used by the *CAL? command has been lost.]
-314	Save/recall memory lost [Indicates that the nonvolatile data saved by the *SAV? command has been lost.]
-315	Configuration memory lost [Indicates that nonvolatile configuration data saved by the device has been lost. The meaning of this error is device-specific.]

1999 SCPI Command Reference

- 320 Storage fault
[Indicates that the firmware detected a fault when using data storage. This error is not an indication of physical damage or failure of any mass storage element.]
- 321 Out of memory
[An internal operation needed more memory than was available.]
- 330 Self-test failed
- 340 Calibration failed
- 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.]
- 360 Communication error
[This is the generic communication error for devices that cannot detect the more specific errors described for errors -361 through -363.]
- 361 Parity error in program message
[Parity bit not correct when data received for example, on a serial port.]
- 362 Framing error in program message
[A stop bit was not detected when data was received for example, on a serial port (for example, a baud rate mismatch).]
- 363 Input buffer overrun
[Software or hardware input buffer on serial port overflows with data caused by improper or nonexistent pacing.]
- 365 Time out error
[This is a generic device-dependent error.]

21.8.12 Query Error

An <error/event number> in the range [-499 , -400] indicates that the output queue control of the instrument has detected a problem with the message exchange protocol described in *IEEE 488.2*, chapter 6. The occurrence of any error in this class shall cause the query error bit (bit 2) in the event status register (*IEEE 488.2*, section 11.5.1) to be set. These errors correspond to message exchange protocol errors described in *IEEE 488.2*, section 6.5. One of the following is true:

- An attempt is being made to read data from the output queue when no output is either present or pending;
- Data in the output queue has been lost.

Events that generate query errors shall not generate command errors, execution errors, or device-specific errors; see the other error definitions in this section.

Error Number	Error String [description/explanation/examples]

1999 SCPI Command Reference

- 400 Query error
[This is the generic query error for **devices** that 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
[Indicates that 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
[Indicates that a condition causing an UNTERMINATED Query error occurred (see *IEEE 488.2*, 6.3.2.2); for example, the **device** was addressed to talk and an incomplete program message was received.]
- 430 Query DEADLOCKED
[Indicates that a condition causing an DEADLOCKED Query error occurred (see *IEEE 488.2*, 6.3.1.7); for example, both input buffer and output buffer are full and the device cannot continue.]
- 440 Query UNTERMINATED after indefinite response
[Indicates that a query was received in the same program message after an query requesting an indefinite response was executed (see *IEEE 488.2*, 6.5.7.5).]

21.8.13 Power On Event

An <error/event number> in the range [-599:-500] is used when the instrument wishes to report a 488.2 power on event to the event/error queue. This event occurs when the instrument detects an off to on transition in its power supply. This event also sets the power on bit, (bit 7) of the Standard Event Status Register. See *IEEE 488.2*, section 11.5.1.

Event Number	Event String [description/explanation/examples]

-500	Power on

[The instrument has detected an off to on transition in its power supply.]

21.8.14 User Request Event

An <error/event number> in the range [-699:-600] is used when the instrument wishes to report a 488.2 user request event. This event occurs when the instrument detects the activation of a user request local control. This event also sets the user request bit (bit 6) of the Standard Event Status Register. See *IEEE 488.2*, section 11.5.1.

Event Number	Event String [description/explanation/examples]

-600	User request

[The instrument has detected the activation of a user request local control]

21.8.15 Request Control Event

An <error/event number> in the range [-799:-700] is used when the instrument wishes to report a 488.2 request control event to the error/event queue. This event occurs when the instrument requests to become the active IEEE 488.1 controller-in-charge. This event also sets request control bit (bit 1) of the Standard Event Status Register. See IEEE 488.2, section 11.5.1.

Event Number	Event String [description/explanation/examples]

-700	Request control [The instrument requested to become the active IEEE 488.1 controller-in-charge]

21.8.16 Operation Complete Event

An <error/event number> in the range [-899:-800] is used when the instrument wishes to report a 488.2 operation complete event to the error/event queue. This event occurs when instrument's synchronization protocol, having been enabled by an *OPC command, completes all selected pending operations. This protocol is described in IEEE 488.2, section 12.5.2. This event also sets the operation complete bit (bit 0) of the Standard Event Status Register. See IEEE 488.2, section 11.5.1. Note: *OPC? does not set bit 0 nor does it enter any event in the error/event queue.

Event Number	Event String [description/explanation/examples]

-800	Operation complete [The instrument has completed all selected pending operations in accordance with the IEEE 488.2, 12.5.2 synchronization protocol]

21.9 :HELP

SYSTem:HELP

The HELP subsystem contains the instrument's help utilities that are provided to the programmer.

21.9.1 :HEADers?

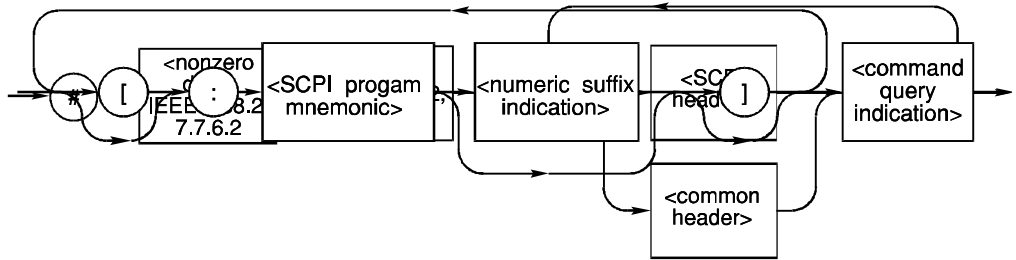
SYSTem:HELP:HEADers?

The HEADers? query shall return all SCPI commands and queries and IEEE 488.2 common commands and common queries implemented by the instrument. The response shall be a <DEFINITE LENGTH ARBITRARY BLOCK RESPONSE DATA> element, see IEEE 488.2 section 8.7.9. The full path for every command and query shall be returned separated by linefeeds. The syntax of the response is defined as:

The <nonzero digit> and sequence of <digit> follow the rules in IEEE 488.2, Section 8.7.9.

An <SCPI header> is defined as:

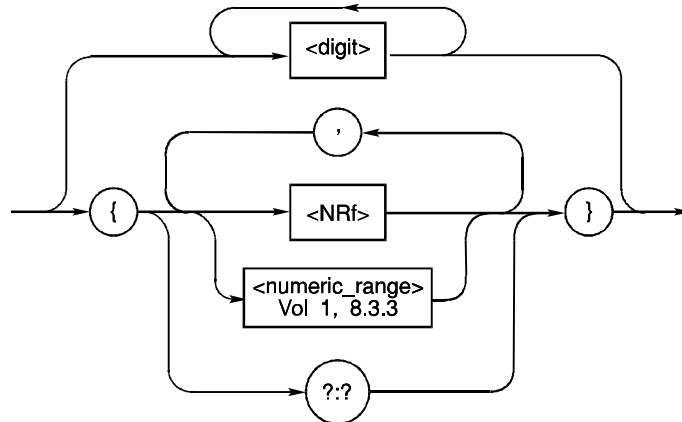
It shall contain all the nodes from the root. The <SCPI program mnemonic> contains the node in standard SCPI format. The shortform shall use uppercase characters while the additional characters for the longform shall be in lowercase characters. Default nodes shall be surrounded by square brackets ([]).



The entire <SCPI header> shall have an indication of the header type, using a <command query indication>.

A node which has a numeric suffix, described in Volume 1, section 6.2.5, is indicated by including a <numeric suffix indication>.

A <numeric suffix indication> is defined as:



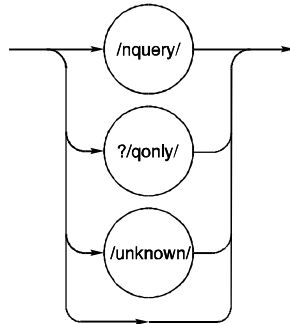
where:

nothing	indicates	no suffix is allowed
<numeric_list>	indicates	list of suffixes allowed
<digit>(s)	indicates	this specific suffix is allowed
{?:?}	indicates	range of suffixes is unknown, or can't tell if suffix is allowed

Numbers between the curly braces shall indicate valid values for the numeric suffix. For example, {1:4} means the values one through four may be appended to that node. If the instrument cannot determine the legal values for the numeric suffix, it may use the special form, {?:?}, to show that a numeric suffix may be allowed even though the range is unknown. The use of a <numeric_list> is especially useful with a large number of suffixes. For headers which accept only a small number of numeric suffixes, specific values are preferred.

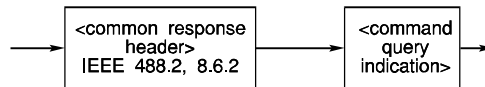
A <command query indication> is defined as:

1999 SCPI Command Reference



If the instrument is unable to determine if the header is an event, query only, or a command/query pair, it shall append /unknown/ to the header. Otherwise, a normal command/query pair shall have nothing additional following the path, a query only header shall have ?/qonly/ appended, and a command which is an event with no query form shall have /nquery/ appended.

A <common header> is defined as:



A <common response header> contains only upper case letters.

Headers may appear in any order.

The response shall not include any macro labels. The *LMC? common query is used to return macro labels.

For example, an instrument which implemented the required commands listed in Syntax & Style, Volume 1 section 4.2.1, this query, and the required IEEE 488.2 common commands and queries might return:

```
#3425
:SYSTem:ERRor?/qonly/
:SYSTem:HELP:HEADers?/qonly/
:SYSTem:VERSion?/qonly/
:STATus:OPERation[:EVENT]?/qonly/
:STATus:OPERation:CONDition?/qonly/
:STATus:OPERation:ENABLE
:STATus:QUESTionable[:EVENT]?/qonly/
:STATus:QUESTionable:CONDition?/qonly/
:STATus:QUESTionable:ENABLE
:STATus:PRESet/nquery/
*IDN?/qonly/
*RST/nquery/
*TST?/qonly/
*OPC/nquery/
*OPC?/qonly/
```

*WAI/nquery/
 *CLS/nquery/
 *ESE
 *ESR?/qonly/
 *SRE
 *STB?/qonly/

*RST shall have no direct effect on the response to this query, but changing the instrument state may change the list of valid commands and queries.

21.9.2 :SYNTax? <command_header>

SYSTem:HELP:SYNTax?

The SYNTax? query causes the device to return a string containing the syntax specification of the command associated with the <command_header>. The <command_header> parameter is a string containing any implemented command or query header that starts from the root. Default nodes and suffixes may appear in the <command_header> parameter.

Numeric suffixes may be explicitly specified in the <command_header>. Only the header and parameter syntax of the specified <command_header> is returned. If no numeric suffix is specified, default 1 is assumed. In that case, devices that implement a numeric suffix shall show the suffix 1 as an optional element in their response.

The part of the returned string that reflects the syntax of the parameter part of the command is be separated from the header syntax part by one or more ASCII space characters (decimal 32). The parameter part shall contain all required and allowed parameters. Parameter notation shall conform to notations of the Command Tables, being used throughout Volume 2 and specified in chapter 5.1 of Volume 1. Optionality is reflected by placing the optional items in between square brackets. Optional nodes are reflected in the same way.

The Long Form shall be used to represent mnemonics in the response to the :SYNTax? query. The letter case notation habit, as specified in chapter 5.1 of “Volume 1: Syntax and Style,” shall be used for mnemonics to differentiate between Long- and Short form.

Any <command_header> that is not a valid command header being recognized by the device, shall cause the device to return a null string (“”). E.g. if the <command_header> contains only a part of the header, contains an illegal numeric suffix, etc.

For example, if the command tree were

```
[ :ALPHa<n> ]
    :BETA
        [ :GAMMa ]
            :DELTa<n> ON|OFF|TRIGger
```

where the <n> indicates a range of suffix values is accepted for ALPHa and DELTa only.

Then SYSTem:HELP:SYNTax? “:BETA:GAMMA:DELTA” will return

```
“ [ ALPHa [ 1 ] ] :BETA [ :GAMMa ] :DELTa [ 1 ] ON|OFF|TRIGger ”
```

SYNTax? is a query only and therefore does not have an associated *RST value.

21.10 :KEY <numeric_value>

SYSTem:KEY

The KEY node is a limited implementation of the naming conventions subsystem described in “Volume 1: Syntax and Style”. The limitation is that keys can only be referred to numerically, and the numbers allowed by a particular instrument (key codes) are fixed.

The KEY? query returns the key code for the last key pressed. Instruments may implement a queue of keys (possibly of length 1) and return a value of -1 to indicate an empty queue. The queue is emptied at *RST.

If the KEY command is sent with a parameter, that simulates the pressing of that key. The keycode determined by the parameter is entirely device-dependent but must match the corresponding parameter returned for the query form. The KEY command may or may not put an entry into the KEY? queue.

The queue of keys is cleared by *RST.

21.10.1 :CATalog

SYSTem:KEY:CATalog

This query returns a list of the defined keys in the subsystem. The response is a comma separated list of NR1 elements corresponding to each key that has been defined.

21.10.2 :DEFine <numeric_value>,<block>[,<string>]

SYSTem:KEY:DEFine

The DEFine command shall redefine a key on the instrument’s keyboard so that pressing the key will execute a particular function. The first parameter is the key code for the key to be remapped, and the second parameter contains the command sequence to be executed. The optional third parameter is a key label for the redefined key.

The particular keycodes are device-dependent.

The softkey label parameter is optional, but it must be accepted by the implementation. If softkeys are not implemented, the third parameter should be ignored.

The command sequence is a block containing any sequence of SCPI commands. Implementations have the option of limiting this block to contain a single command. It is device-dependent when the block has its syntax checked.

The query form of the command returns a list of all redefined keys in the form: <NR1>,<block>,<string>. If the key is unlabeled, the third parameter shall be an empty string.

Key definitions are not affected by *RST.

21.10.3 :DElete <numeric_value>

SYSTem:KEY:DElete

The DElete command removes the definition of the key referred to by <numeric_value> set by a previous DEFine command. The key may return to its default definition, if any.

21.11 :KLOCK <Boolean>

SYSTem:KLOCK

This command locks the local controls of an instrument. This includes any front panel, keyboard, or other local interfaces. The SYSTem:KLOCK setting does not affect the 488.1 Remote/Local state, if implemented.

This value cannot be reset to OFF, unless SYSTem:SECurity:STATe is OFF.

If SYSTem:SECurity:STATe is OFF, the KLOCK value is set to OFF at *RST. If SYSTem:SECurity:STATe is ON, *RST has no effect.

21.12 :LANGuage <string>

SYSTem:LANGuage

The receipt of this command shall cause the instrument to perform a language switch to another language. The contents of the string are device dependent, however, the following contents are suggested:

- COMPatibility — The language is a backwards compatible set for the product line.
- CIIL — Selects the CIIL language of the instrument.

All devices which implement alternate languages shall also implement this command. An instrument which is currently executing SCPI will not leave SCPI to enter an alternate language under any other conditions, except possibly power-on. There is no query corresponding to this command.

21.13 :LFRequency<numeric_value>

SYSTem:LFRequency

The numeric value is used as the reference frequency. If the instrument is capable of using only discrete reference values (e.g. 50 or 60 Hz), it will use the specified value to select the nearest valid setting.

*RST does not affect this setting.

21.13.1 :AUTO <Boolean> | ONCE

SYSTem:LFRequency:AUTO

If AUTO ONCE is specified, the instrument will measure the line frequency internally and use the result (or the nearest valid setting, if necessary) as its reference. After the measurement is complete, this setting reverts to OFF. If AUTO ON is specified, the instrument will continuously monitor the line frequency (through repetitive measurements, phase locking, etc.).

Behavior at power-on is device-dependent; an instrument capable of measuring the line frequency may choose to do so when power is applied. Other options are (1) to preserve the value last specified with LFRequency, (2) to choose a default setting, or (3) to use hard controls to determine the setting.

*RST does not affect this setting.

21.14 :LOCK SYSTem:LOCK

This subsystem controls the availability of an instrument. It allows the communication channel that these SCPI commands arrived on (session) to attempt to attain exclusive use of this instrument by attaining a lock.

The behavior of the device when receiving SCPI commands from a session that does not have the lock is device dependent.

On *RST the ownership of the lock is device dependent.

21.14.1 :OWNer? :SYSTem:LOCK:OWNer?

This query returns what session currently has the lock on this device. If no session has a lock then “NONE” shall be returned.

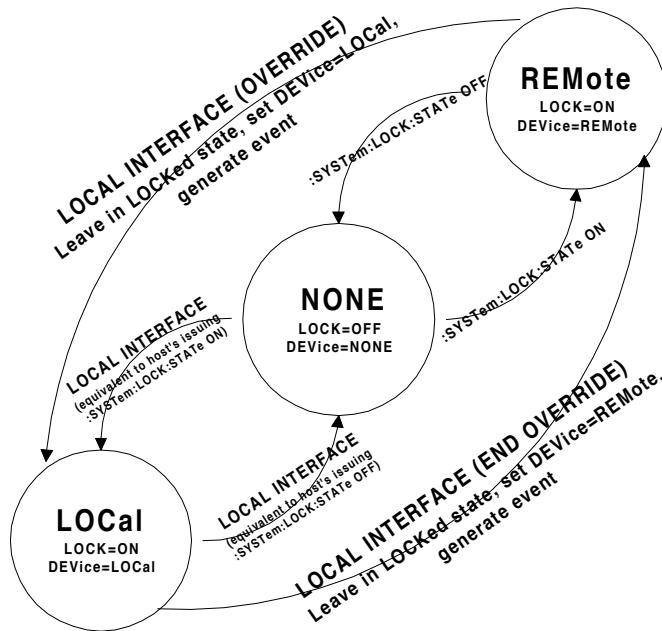


Figure 21.2 System Lock Format

21.14.2 :RELease :SYSTem:LOCK:RELease

This event releases the lock if owned by this session. If this session does not have the lock, this command has no effect. This command has no query form.

21.14.3 :REQuest? :SYSTem:LOCK:REQuest?

This event shall only be implemented as a query. It attempts to attain the lock on this device and returns 1 if successful and 0 if it fails.

21.15 :PASSword

SYSTem:PASSword

This subsystem provides for the enabling, disabling, and changing of passwords to control access to selected SCPI commands and queries. The instrument's documentation shall indicate which commands are under the control of SYSTem:PASSword.

Password protection shall be locally defeatable. The device shall have a local means to:

- reset the password to the default value, "DEFAULT".
- enable all protected commands.

The means of protection for this local resetting and enabling is entirely device dependent.

The device shall report error -203, (command protected), if a password-protected command is received that is currently disabled.

Neither the current password nor command protection state shall be affected by *RST.

21.15.1 :CDISable <password>

SYSTem:PASSword:CDISable

Sending :CDISable (Command DISable) disables protected commands. The <password> data is of type string and the password is case sensitive. There is no query form.

Sending an invalid password generates error -221, (settings conflict).

21.15.2 [:CENable] <password>

SYSTem:PASSword:CENable

Sending :CENable (Command ENable) enables protected commands to function. The <password> data is of type string and the password is case sensitive. There is no query form.

Sending an invalid password generates error -221, (settings conflict).

21.15.2.1 :STATE?

SYSTem:PASSword:CENable:STATE?

STATE? returns a 0 if password protected commands are disabled and a 1 if they are enabled. There is no command form.

21.15.3 :NEW <current password>,<new password>

SYSTem:PASSword:NEW

This command changes the password from the default or current value to the new value. Both parameters are strings. The <current password> must be correct for the command to execute. Password strings are case sensitive. There is no query form.

Sending an invalid current password generates error -221, (settings conflict).

21.16 :PRESet

SYSTem:PRESet

Sets the device to the same state as the front-panel reset key. This command is similar to the *RST command, but differs in that it duplicates the action of the front panel reset key.

1999 SCPI Command Reference

Normally this state will be optimized for front panel operation of the device instead of bus operation.

Other PRESet states may be accessed with this instruction by using a numeric suffix. For instance, PRESet2 could provide a high speed configuration.

The state of the device after PRESet is not defined by SCPI. There is no query corresponding to this command.

21.17 :SECurity

SYSTem:SECurity

Controls the security of the instrument.

21.17.1 :IMMediate

SYSTem:SECurity:IMMediate

This command is an event which immediately destroys all measurement data and instrument settings. Current settings are initialized to their *RST values. All macros, user variables, and *SAV/*RCL registers are initialized. Critical calibration constants are retained.

This command can be under control of the SYSTem:PASSword subsystem. The device shall report error -203, (command protected), if SECurity:IMMediate is received while password protected.

This command is an event, and therefore has no *RST value associated with it.

21.17.2 [:STATe] <Boolean>

SYSTem:SECurity:STATe

STATe controls the security mode of the instrument. For example, when in secure mode, any display annunciators which have been disabled cannot be reenabled without destroying certain stored information.

The value is not affected by *RST or *RCL.

When this value changes from ON to OFF, everything except possibly some CALibration data must be initialized or destroyed. For example, the following shall be initialized:

- Instrument state (perform *RST).
- All macros (perform *PMC).
- All user variables.
- All state registers (perform *SAV into all registers after *RST).

21.18 :SET <block data>

SYSTem:SET

Sets the instrument to the complete or partial state determined by the contents of the parameter. The form and contents of the block data is device-dependent.

This command shall be required for instruments that return a block of binary settings in response to the *LRN? query.

1999 SCPI Command Reference

Instruments that implement the *LRN? query and return a block of binary settings as response to *LRN?, shall implement also the SYSTem:SET command. In this case the response format of *LRN? shall contain SYST:SET <block data>.

Note that IEEE 488.2 requires the response to *LRN? to be a legal program message. Since the SYSTem:SET command is the SCPI command which programs the instrument settings by binary block data, this must be the response to the *LRN? query when it returns this data format.

The implementation of the SYSTem:SET command is optional if the *LRN? query response are the SCPI commands that are normally used for programming the instruments.

The query form of this command will return a definite block data element which contains the instrument's current state. The SET command may be implemented without the associated query.

21.19 :TIME <hour>,<minute>,<second>

SYSTem:TIME

Any instrument which has an internal clock shall implement this command to set the clock. The three parameters for this command are:

- <hour>: Must be <numeric_value>. It is always rounded to the nearest integer. Its range is 0 to 23 inclusive. All instruments implementing the TIME function shall report and accept hour information in 24 hour format.
- <minute>: Must be <numeric_value>. It is always rounded to the nearest integer. Its range is 0 to 59 inclusive.
- <second>: Must be <numeric_value>. It is rounded to the resolution of the clock. Its range is 0 to 60. 60 is allowed since rounding may cause a number greater than 59.5 to be rounded to 60. When this element is rounded to 60 it shall be set to 0 and the minute value incremented. Any other carries shall be rippled through the date.

The query response message shall consist of three fields separated by commas:

<hour>,<minute>,<second>

The three fields will contain the following information:

- <hour>: NR1 format ranging from 0 to 23.
- <minute>: NR1 format ranging from 0 to 59.
- <second>: NR1 or NR2 format. If the resolution of the clock is no better than seconds, an NR1 format shall be used. If the resolution is better than seconds, an NR2 format shall be used. Range is 0 to 59.999999...

This setting shall not be affected by a *RST command.

21.19.1 :TImEr

SYSTem:TIME:TImEr

This subsystem contains commands that control an internal timer/counter.

In some instruments the value of this timer may be used to timestamp internal readings.

21.19.1.1 :COUNt <numeric_value>

SYSTem:TIME:TImEr:COUNt

COUNt sets or queries the current value of the timer.

Sending this command shall not affect TImEr:STATe. When the TImEr:STATe is ON the count will increase.

This command sets or queries the current count for the timer in seconds.

The default value of this parameter is 0.

At *RST this value be set to 0.

21.19.1.2 [:STATe] <Boolean>

SYSTem:TIME:TImEr:STATe

This node sets or queries the TImEr:STATe. If the STATe is ON the TImEr is running.

At *RST the TImEr:STATe is OFF

21.20 :TZONe <hour> [,<minute>]

SYSTem:TZONe

An instrument which has an internal clock and can keep track of the time zone shall implement this command to set the time zone.

The fields have the following meanings:

<hour>: Must be a <numeric_value>. The value shall be rounded to the nearest integer. Range -12 to +12.

<minute>: Must be a <numeric_value>. The value shall be rounded to the nearest integer. Range -59 to 59.

When each field is subtracted from the value of the TIME command, the result is the correct universal coordinated time (also known as UCT, Zulu, Greenwich Mean Time).

The default value of <minute> is zero.

This shall not be changed by the receipt of a *RST.

21.21 :VERSion?

SYSTem:VERSion?

This query returns an <NR2> formatted numeric value corresponding to the SCPI version number for which the instrument complies. The response shall have the form YYYY.V where the Ys represent the year-version (i.e. 1990) and the V represents an approved revision number for that year. If no approved revisions are claimed, then this extension shall be 0.

1999 SCPI Command Reference

An instrument known to claim SCPI compatibility that does not respond to the `VERsion` query shall be assumed to conform to version 1990.0.

This query shall be implemented by all instruments claiming conformance to a SCPI version greater than 1990.0.

`VERsion?` is a query only and therefore does not have an associated `*RST` state.

22 TEST Subsystem

The TEST subsystem provides a section to extend standard instrument self-test procedures beyond the *IEEE 488.2* *TST command.

23 TRACe | DATA

A TRACe or a DATA area is a named entity stored in instrument memory. TRACe | DATA areas may also be used to store other types of data, such as constant arrays for use in trace arithmetic or corrections, or displayed waveforms. Alternatively, TRACe | DATA areas may be used for equivalent scalar (single point) purposes.

Certain TRACe format and capabilities are the focus of other emerging standards. To prevent diluting the effect of SCPI or any other standards related to the manipulation of TRACes, SCPI has left such capability undefined at this time with the view to adopting an industry standard when one becomes available. The undefined capabilities include the following:

- Trace math operations for specifying arithmetic relationships between traces, including the use of user-defined constants and functions.
- Trace activation and routing for specifying the destination for measurement results.

KEYWORD	PARAMETER FORM	NOTES
TRACe DATA		1991
:CATalog?		[query only]
:COPY	<trace_name>, (<trace_name> <data_handle>)	[event; no query] 1991,2
[[:DATA]	<trace_name>,(<block> <dif_expression> (<numeric_value>{,<numeric_value>}))	1993
:LINE	<trace_name>,<numeric_value>,<numeric_value>, <numeric_value>,<numeric_value>	1992
:PREamble?	<trace_name>	[query only] 1994
:VALue	<trace_name>,<numeric_value>,<numeric_value>	1992
:DEFine	<trace_name>[,<numeric_value> <trace_name>)]	
:DELete		
[[:NAME]	<trace_name>	[event; no query]
:ALL		[event; no query]
:FEED	<trace_name>,<data_handle> NONE)	1991
:CONTRol	<trace_name>,(ALWays OCONdition NEXT NEVer)	1991
:OCONdition	<trace_name>,<condition_expr>	1991
:FREE?		[query only]
:POINTs	<trace_name>[,<numeric_value>]	1993
:AUTO	<trace_name>,<Boolean> ONCE)	[event; no query]

Many TRACe subsystem commands are of the following general form:

TRACe:COMMAND <destination trace_name>,<source trace data>

The <trace_name> is <CHARACTER PROGRAM DATA> (IEEE 488.2, section 7.7.1).

There is no difference between uppercase and lowercase letters used in <trace_name>. There

1999 SCPI Command Reference

is no short form abbreviation for <trace_name>. The destination trace is always the first parameter in the list. The <source trace data> parameter may be an existing <trace_name>, or a block, or expression which is evaluated as trace data.

*RST has no effect on this subsystem.

23.1 :CATalog?

TRACe:CATalog?

The CATalog? query returns a comma-separated list of strings which contains the names of all traces. If there are no <trace_name> defined, a single empty string is returned. Some instruments may have one or more intrinsic, permanent <trace_name>. These traces must be listed along with any user-named traces.

23.2 :COPY <trace_name>, (<trace_name> | <data_handle>)

TRACe:COPY

The COPY command set the data values in the destination trace, from internal data stores in the instrument. There is no query form of this command, it is an event only.

If the data source is a <trace name>, the data is copied into the destination trace. For example, COPY REF,TRACE1 will copy the data currently in TRACE1 into the trace named REF.

If the data source is a <data_handle>, the data shall be copied directly from the data pool specified. For example, COPY REF, "CALCulate1" will copy the data emitted by the CALCulate1 block. When the data is copied depends on the type of memory associated with the data flow. If the memory is transient, as described in section 2.7.5, then the data shall be copied when data becomes available, for example when a new acquisition takes place. If the memory is static the data shall be copied immediately, as there is memory that always has some value in it associated with the specified <data_handle>, typically the result from an earlier acquisition.

23.3 [:DATA] <trace_name>,(<block>|<dif_expression>|<numeric_value>{,<numeric_value>}))

TRACe:DATA

The DATA command sets the data values in the destination trace, from information provided by the controller. If the data source is a block of data, then the block is directly loaded into the destination trace. In this case the data source is block data, the data format is determined by the FORMat Subsystem. Note that the data source may also be a Data Interchange Format expression.

If the data source is a single <numeric_value>, every element of the destination trace is set to the constant value represented by the evaluation of <numeric value>. For example, DATA REF,0.22361 sets all points in the REF trace to 0.22361. If a <numeric_value> is sent for each of the points in the allocated <trace_name>, then the <numeric_value>s are mapped into the <trace_name>. If an insufficient number of <numeric_value>s are sent, that is less than the allocated points, then the result is device dependent. If too many <numeric_value>s are sent the device shall generate error -223 (too much data), and the result shall be device dependent.

The DATA? query <trace name> returns the data values for <trace name>, according to the format determined by commands in the FORMat subsystem.

Note: All parentheses in the syntax for this command are used for grouping purposes only.

23.3.1 :LINE <trace_name>,<numeric_value>,<numeric_value>,<numeric_value>,<numeric_value>

TRACe:DATA:LINE

The LINE command is an event which defines a line segment (a series of points) within the boundaries of the trace whose name is specified by <trace_name>. The first <numeric_value> determines the index of the starting point in the trace. The second <numeric_value> is the value of the trace at the starting point. The third and fourth <numeric_value> parameters set the end of the line segment index and value. A line segment is drawn connecting the start point and end point, inclusive. This command has no query form. The index of the first point in a trace is 1.

23.3.2 :PREamble? <trace_name>

TRACe:DATA:PREamble?

PREamble? returns the preamble information supporting the DATA(CURVe(VALues)) but omits the following data:

- DIF blocks of waveform, measurement, and delta.
- DIF curve block keywords VALues and CSUM and their parameters.

The DIF difid block must contain the keyword SCOPE with the parameter of PREamble.

While the PREamble? query is designed to efficiently transmit data relating directly to the curve, the various notes and identification parameters may be sent if absolutely required.

The <trace_name> parameter functions the same as it does in TRACe:DATA?.

23.3.3 :VALue<trace_name>,<numeric_value>,<numeric_value>

TRACe:DATA:VALue

The VALue command sets or queries the value of an individual point in a trace. The first <numeric_value> is the index of the addressed point in the trace defined by <trace_name>. The second <numeric_value> is the value to be set for the point. The index of the first point in a trace is 1.

The VALue? <trace_name>,<numeric_value> query returns the data value of the trace <trace_name> at the <numeric_value> indexed point.

23.4 :DEFine <trace_name>[(,<numeric_value>|<trace_name>)]

TRACe:DEFine

The DEFine command allocates and initializes a new trace. The first parameter is <CHARACTER DATA>, and it specifies the new <trace_name>. If the second parameter is a <numeric_value>, a new trace is allocated with the specified number of data elements. The trace shall be initialized to instrument-specified default values. If the second parameter is the name of an existing trace, the new trace becomes a copy of the existing trace in all respects, including size and contents. The data in the existing trace is not affected. When the second

parameter is omitted, the new trace shall have an instrument-specified default size and initial value.

Limits on the number of named traces are instrument-specific.

23.5 :DELeTe

TRACe:DELeTe

The DELeTe commands allow the user to dissociate <trace_name> from trace memory. Instrument intrinsic <trace_name> cannot be deleted. There is no query form for the DELeTe commands.

23.5.1 [:NAME] <trace_name>

TRACe:DELeTe:NAME

The NAME command dissociates a user-created <trace_name> from its trace memory. The NAME node is optional in a program message. That is, an instrument is required to accept both forms of this command. If the instrument supports dynamic trace memory allocation, then the memory allocated for <trace_name> is freed.

23.5.2 :ALL

TRACe:DELeTe:ALL

The ALL command dissociates all user-created <trace_name> from their trace memory units. If the instrument supports dynamic trace memory allocation, then all memory allocated for user-created traces is freed.

23.6 :FEED <trace_name>, (<data_handle> | NONE)

TRACe:FEED

Sets or queries which data flow is fed into the specified TRACe|DATA memory. The first parameter specifies the name of the TRACe|DATA memory. The second parameter (used only in the command form) selects the data flow to be fed into the memory.

A <data_handle> is the name of a point in the data flow described by the Instrument Model (found at the beginning of this volume). By setting the <data_handle> to a null string (""), no data shall be fed into the specified TRACe|DATA memory. When a TRACe|DATA memory is created with the DEFiNe command, the FEED <data_handle> for that memory shall be set to null ("").

The query form of this command requires that a valid <trace_name> shall be specified. If the TRACe|DATA memory specified in <trace_name> does not exist then an error -224, "Illegal parameter value" shall be generated. The query shall return a string that represents the <data_handle> selected. If no feed is selected, then the query shall return a null string ("").

At *RST, FEED shall be set to NONE.

23.6.1 :CONTRol <trace_name>, ALWays | OCONdition | NEXT | NEVer

TRACe:FEED:CONTRol

Sets or queries how often the specified TRACe | DATA area accepts new data. This control has no effect if the FEED <data_handle> is set to null (""). The first parameter specifies the TRACe | DATA area, the second parameter selects how often new data shall be accepted.

By selecting ALWays, whenever new data is available it shall be fed into the specified TRACe I DATA area. On CONdition defines that data shall be gated by the OCONdition command. NEXT is a one-shot feed, it always waits for new data, such as an new acquisition, and ignores any existing data. Once NEXT has completed its feed of data, CONTROL reverts to NEVer. NEVer shall cause no data to be fed.

At *RST, CONTROL is set to a device dependent value.

23.6.2 :OCONdition <trace_name>, <condition_expr>

TRACe:FEED:OCONdition

The On CONdition command sets or queries the condition used to gate data flow into the specified TRACe I DATA area. The first parameter specifies the TRACe I DATA area. The second parameter is EXPRESSION PROGRAM DATA, which defines the condition, which when evaluated to be True shall allow data to be fed through. The production <condition_expr> is of the general form:

(' <operand_str> <equiv_op> <operand_str> ') or
' <event_handle> '

The productions <operand_str> and <equiv_op>, the operand string and the equivalence operator respectively, shall be defined later.

23.7 :FREE?

TRACe:FREE?

The FREE? query returns the amount of user memory space available for traces, in the following form:

<bytes available>, <bytes in use>

where the returned data for both shall be in <NR1> format.

23.8 :POINTs<trace_name>[,<numeric_value>]

TRACe:POINTs

The POINTs command sets the number of measurement data points available in the specified trace memory. The first parameter specifies the name of the trace to resize. The optional second parameter specifies the number of data points which the <trace_name> must accommodate. If this parameter is omitted, an instrument specific default value will be used. If more than one <numeric_value> is associated with each trace data point (as would be the case with a trace consisting of real and imaginary pairs), then the second parameter specifies the number of N-tuples in the trace.

The query POINTs? <trace_name> returns the number of measurement data points in the specified trace. Note that each data point may have several numeric data items associated with it.

23.8.1 :AUTO <trace_name>,(<Boolean>IONCE)

TRACe:POINTs:AUTO

The AUTO command turns trace autosizing ON/OFF. When it is enabled, <trace_name> will automatically be resize as necessary to accommodate any new data block assigned to it. In this mode of operation, changes in measurement setup will not require adjustments to the

1999 SCPI Command Reference

size of defined traces. If POINTs:AUTO <trace_name>, ONCE is specified, and the trace sizing occurs once (on the next measurement). Limits on trace sizes are instrument-specific.

24 TRIGger Subsystem

The trigger subsystem is used to synchronize device action(s) with events. A device action might be the acquisition of a measurement or the application of a stimulus. The trigger subsystem consists of the expanded capability model which is capable of describing very complex device trigger systems. It also makes provision, through the ARM-TRIGger model, for simple descriptions of less complicated trigger systems. These two models are consistent and compatible with each other. The ARM-TRIGger model represents a subset of the capability available in the expanded capability model.

The figures in this section that represent the trigger model adhere to the following nomenclature. A box identifies a state of a transition diagram and is referred to as a layer. A sequence is a set of vertically connected layers. A solid line defines flow of control between states. A dashed line defines how an event is propagated to other parts of the trigger model and the instrument. Events generated by control flowing from one part of the trigger model to another part are called “sequence events.”

24.1 ARM-TRIGger Model

The ARM-TRIGger model represents a level of capability that is often found in a device and is shown in figure 24-1. The ARM-TRIGger model shows two independent levels of event detection, one in each of the ARM and the TRIGger layers. For a given device the model

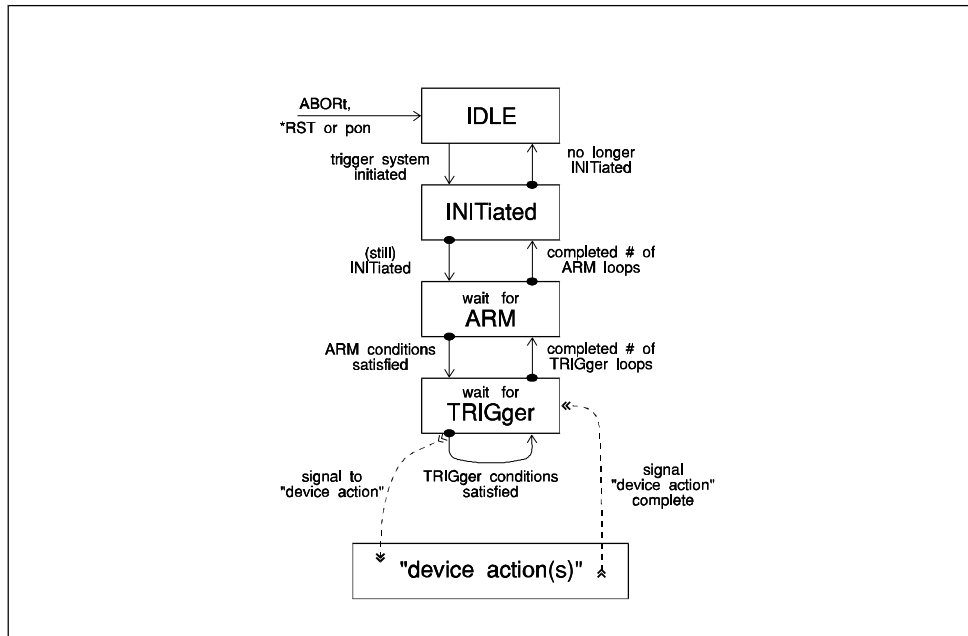


Figure 24-1 ARM-TRIGger Model

may be simplified further by the omission of the ARM layer and any inappropriate commands for that device.

24.2 Model Layers

The following figures detail each of the various types of layers that may exist in a sequence. Each layer is shown with a dotted line, which is used to divide the upward and downward traverses through the layer. The different traverses are referenced in describing the operation of each layer.

24.2.1 IDLE State

On receipt of either *RST or ABORt, the trigger subsystem shall enter the IDLE state. Receiving the *IEEE 488.2* **dcas** message may also cause a transition to IDLE. Devices which cannot process commands when not in IDLE must enter IDLE when **dcas** is received to meet the *IEEE 488.2* requirements for device clear. The downward traverse from IDLE state, “trigger system initiated,” is affected by either the INITiate[:IMMEDIATE] command or by setting INITiate:CONTinuous to ON.

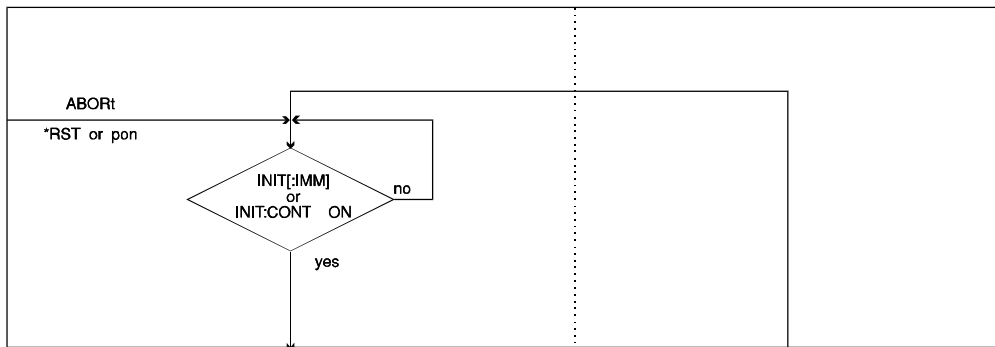


Figure 24-2 IDLE State

If the INITiate commands are implemented as overlapped, exiting the IDLE state shall cause the pending operation flag (as defined by IEEE 488.2) associated with the initiated action to be set true. Entering the IDLE state shall cause the pending operation flag to be set false.

24.2.2 Initiated

Once the trigger sequence is initiated from the IDLE state, control passes through the INITiated state, shown in Fig 24-3, immediately making the downward traverse to the highest ARM:LAYer state. The upward traverse is dependent on the setting of CONTinuous. If CONTinuous is set to OFF then the upward transition to the IDLE state shall be made. Otherwise, the downward traverse to the highest ARM:LAYer state shall be made, so that the trigger sequence remains initiated. If the sequence does not contain any ARM layers, the downward transition goes to a TRIGger state.

The INITiated state is necessary to avoid having the pending operation flag change state every time the entire trigger sequence is completed when CONTinuous is ON. On exit from the INITiated state, in both directions, a “sequence event” is generated. These events may be used to start actions in the trigger subsystem or actions in other parts of the instrument. Fig 24-3 shows the names of these events. An instrument shall implement the initiated state even if it implements the INITiate commands as sequential.

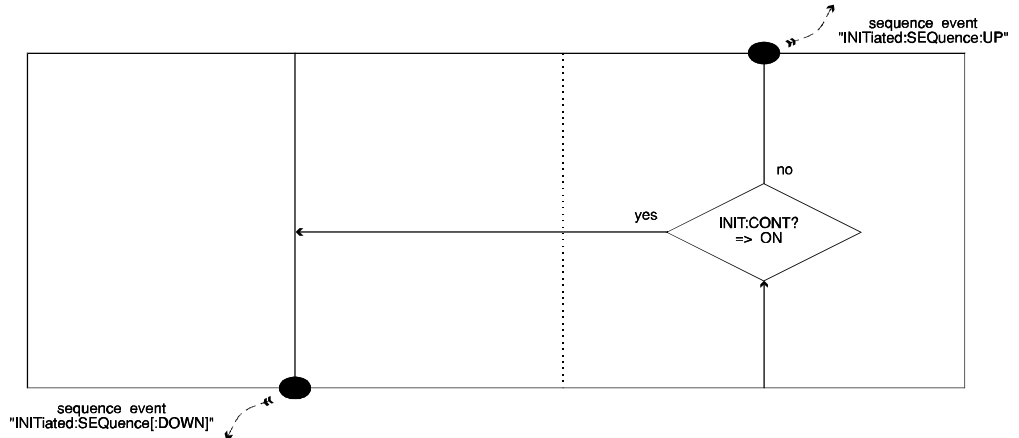


Figure 24-3 INITiated State

24.2.3 Event Detection Layer

The ARM and TRIGger layers are both event detection layers with the TRIGger layer being subservient to the ARM layer. Each layer provides one level of event detection.

The downward traverse through an event detection layer depends on the sourced event being detected the specified number of times. Typically, the first event detected after entering the event detection layer is all that is required to proceed. However, a particular number of occurrences of the same event may be specified using ECount; for example, wait for the tenth positive edge of a signal. The downward traverse is also subject to a time delay if one is specified.

Two commands override the normal process of the downward traverse. The layer name followed by :IMMediate causes the event detection and the subsequent delay to be bypassed. The layer name followed by :SIGNal causes the device to proceed as though a single event had occurred.

On exit from the ARM or TRIGger state, in either direction, a “sequence event” is generated. These events may be used to start actions in the trigger subsystem or in other parts of the instrument. An action may not be associated with every “sequence event.”

The upward traverse is dependent upon the event detector being satisfied and the value of COUNT for the given layer. The event detector in the upward traverse is used to detect when an action started by the exit from this layer on the downward traverse has completed. This event detector has no commands associated with it, and it is only included in the model to prevent the trigger system from looping around until the “device action” associated with the given layer has completed.

When COUNT is greater than one, all of the subservient layers are cycled repeatedly COUNT times. For example, to make five measurements, each qualified by the same combination of ARM and TRIGger events, the COUNT in the ARM layer should be set to 5. Each time the ARM layer is entered from below, the flow is redirected to follow the downward traverse. After the fifth cycle, when COUNT is satisfied, the upward traverse to the initiated state is made.

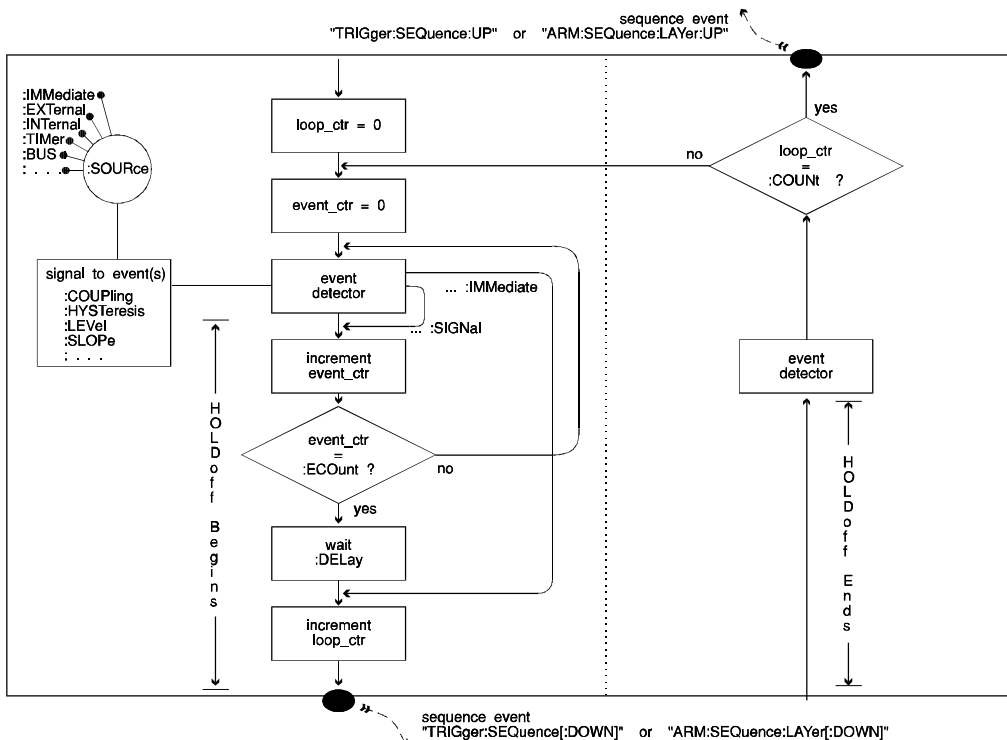


Figure 24-4 Event Detection Layer

24.3 Sequence Event Use

In the ARM-TRIGger model shown in Fig 24-1, the “sequence event” generated by the TRIGger layer is shown starting a “device action.” The “device action” is dependent on the settings of the SENSE or SOURCE subsystems, for example by the FUNCTION selected. The settings of the SENSE and SOURCE subsystems are in general device dependent and

therefore, the “device action” tied to a “sequence event” is also device dependent. For example, in one instrument the event may be used to generate a signal and in another instrument it may be used to make a measurement.

Further, the completion of a device action itself is device dependent. For example, the device action could be defined as executing a complete sweep, or it could be defined as initiating a sweep. These device actions would signal their completion at different times and consequently, would permit the upward transistion through the layer at different times. Where an instrument permits control of the characteristics of a device action, the commands for that control shall appear in the subsystem employing the event. In the earlier example, a command to differentiate between initiating a sweep and executing a complete sweep would exist with the commands that control the other characteristics of the sweep.

The name of a “sequence event” reflects where it was generated. Its form is

```
“(INITiate|TRIGger)([:SEquence[1]]|:SEquence<n>|:<sequence_name>)([:DOWN]|:UP)”
or
“ARM[:SEquence[1]]|:SEquence<n>|:<sequence_name>)([:LAYer[1]]|:LAYer<n>)([:DOWN]|:UP)”
```

Note: Parentheses used here are for grouping and not literals. The syntax of a sequence event is the same as the syntax of the instrument-control header.

It includes the direction of traverse, the layer, and the sequence. The use of defaults, however, allows some short aliases. “ARM” is equivalent to “ARM:SEquence1:LAYer1:DOWN”.

24.4 Expanded Capability Trigger Model

Fig 24-5 is an example of an expanded capability trigger model. An instrument could implement a more extensive or simpler trigger system. The purpose of the figure is to illustrate concepts. A picture of all possible configurations is not possible.

The expanded capability trigger model expands upon the ARM-TRIGger model to meet more complex needs through additional “Event Detection Layers” as shown in Fig 24-4. These additional layers give extra levels of event detection which provide more qualification to a particular action, or establish more “sequence events” to synchronize many actions within an instrument. Any number of “Event Detection Layers” may be included as additional ARM layers, as shown in Fig 24-5. A single trigger layer still exists in the expanded capability model.

Multiple sequences may exist in the expanded capability model to describe concurrently operating trigger conditions, also shown in Fig 24-5. Any number of sequences are permitted. Sequences may have the same or a different number of layers. Some of the sequences may have aliases. These aliases are used where the sequence is performing a standard trigger operation. Multiple sequences may be synchronized by linking “sequence events” of one sequence to the event detector of another sequence. Some of the standard sequences interact in a pre-defined manner which saves creating unnecessary links and complicating the model for the standard sequences.

An IDLE state is shown with each sequence. The commands *RST and ABORt place all sequences immediately in their IDLE states. A device clear shall put the sequences to their IDLE states if the device is unable to parse another command without aborting the trigger subsystem (non-overlapped operation). Not aborting the initiated sequences is preferable if the parser can be cleared by a device clear so it can process a command.

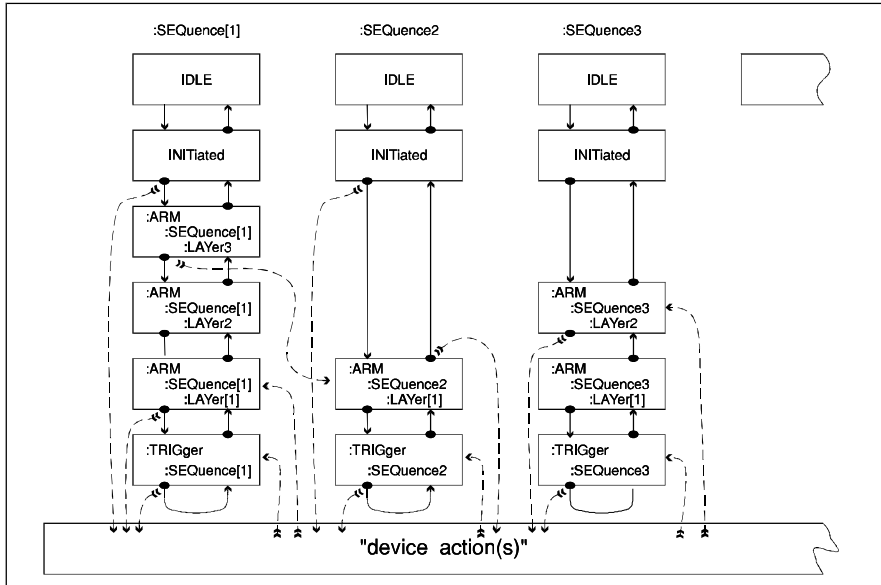


Figure 24-5 Expanded Capability Trigger Model

24.4.1 LAYer Nomenclature

Each state in the trigger subsystem is made unique by its position determined by its row and column. In the trigger subsystem each column is called a SEquence. The rows are labelled in a different manner to maintain compatibility with the ARM-TRIGger model. The lowest row is called the TRIGger layer. The next one up is called the first ARM layer. The third layer from the bottom is called the second ARM layer.

The <layer_name> for a particular state is determined from its SEquence, whether it is an ARM or a TRIGger layer, and finally the LAYer number if it is an ARM layer. In Figure 24-5, the layer in the lower left most position is TRIGger[:SEquence[1]]. Because it is in SEQUENCE1, SEQUENCE is optional and may be omitted. For this layer, TRIGger is sufficient, and is the name used in the ARM-TRIGger model. SEQUENCE[1] also has the alias START. This layer's name is also TRIGger:START.

The SEQUENCE number increases from left to right, and other SEQUENCES may have aliases as well. The layer to the right of TRIGger:SEQUENCE1 is TRIGger:SEQUENCE2, which may also have the sequence alias TRIGger:STOP. The layer above TRIGger:START is ARM[:SEQUENCE[1]][:LAYer[1]], which may be abbreviated to just ARM as LAYer1, as

well as SEQUENCE1, is the default. The next layer up is ARM[:SEQUENCE[1]]:LAYER2. The LAYER number increases from the lowest ARM layer to the top.

The “sequence event” names reflects the <layer_name> and whether it occurred on the up or downward traverse. Thus, the event used in the ARM-TRIGGER model to start the “device action” is TRIGGER[:SEQUENCE[1]]:DOWN]. The events related to the INITIATED state are INITIATED[:DOWN] and INITIATED:UP.

24.4.2 Standard SEQUENCES

SCPI describes the behavior of SEQUENCES which have specific names. An instrument is not required to implement any of these sequences, but any sequence given one of these names shall behave as described.

START The START sequence exists to provide pre-qualification to a “device action.” This sequence shall leave IDLE when an INITIATE:IMMEDIATE:ALL or INITIATE:CONTINUOUS:ALL ON is given. If the START is implemented then it shall exist as an alias to SEQUENCE[1].

STOP The STOP sequence allows the user to force the currently initiated sequences to IDLE on a particular event or series of events. All SEQUENCES shall go to IDLE upon “INITIATED:STOP:UP”.

The STOP sequence shall not leave IDLE when an INITIATE:IMMEDIATE:ALL or INITIATE:CONTINUOUS:ALL ON is given, but shall leave IDLE upon “INITIATE:START:DOWN”. This means that STOP is subservient to START in the way the STOP sequence leaves IDLE. All sequences, however, are subservient to STOP because the sequence event from STOP affects all other sequences.

For those instruments that implement START as an alias for SEQUENCE[1], or implement START as an alias for SEQUENCE[1] and STOP as an alias for SEQUENCE2, the ARM and TRIGGER SEQUENCE:DEFINE command and query are not required. Those instruments that employ a different mapping, whether the mapping is fixed or user definable, or use other aliases shall implement the DEFINE command and query for all sequences that exist.

24.4.3 Subservient SEQUENCES

When an event from one sequence controls the initiation of another sequence, the second sequence is subservient to the first. The “device action” of a sequence initiates another sequence.

An instrument may implement any number of sequences depending on its needs and capabilities. The names for the sequences should reflect the application of an instrument. Naming sequences requires implementing the SEQUENCE:DEFINE command.

Each subservient sequence has IDLE and INITIATED states like other sequences. The INITIATED state of a subservient sequence does not, however, check the setting of INITIATE:CONTINUOUS. It merely exits out the top. This state is included so the event, INITIATE:SEQUENCE:UP, is available for all sequences. The IDLE state of a subservient sequence does not check either the setting of INITIATE:CONTINUOUS or for the arrival of

INITiate[:IMMEDIATE]. It does, however, check for an event from a connected sequence as shown in Figure 24-6.

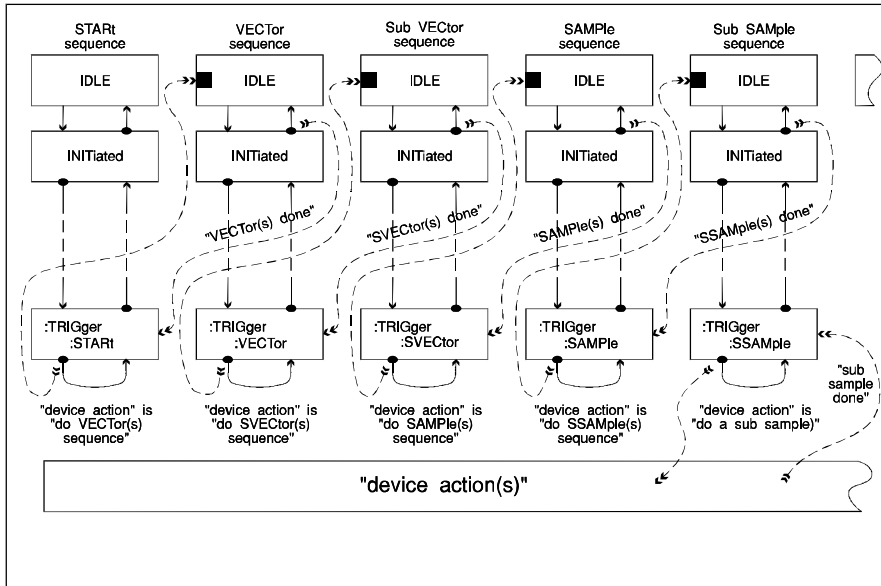


Figure 24-6 Subservient Sequences

Fig 24-6 illustrates how several sequences could be organized in a subservient manner. An actual instrument would probably have more or fewer than four. It might also use different sequence names for the same device actions being controlled.

VECTor In this example, the VECTor sequence is used to control the pacing of whole vectors such as when generating or acquiring a waveform. Multiple VECTor sequences could exist, each distinguished by a numeric suffix. Several VECTor sequences would be used for multiple nested sweeps to generate a surface. For example, sweeping the power level at each point in a frequency sweep. The controls of the VECTor sequence often are coupled to either the SENSE:SWEEP or SOURCE:SWEEP subsystems. Changing SWEEP:DWELL and SWEEP:TIME might change TIMER and DELAY values.

SVECTor This Sub VECTor sequence is used to control the pacing of collections of samples used to create a VECTor. Sub VECTors are useful in acquisition instruments where whole vectors are averaged. The settings in such a case would be coupled to the SENSE:AVERAGE subsystem.

SAMPLE The SAMPLE sequence is used to control the pacing of each sample. A SAMPLE is a single point in the SVECTor.

1999 SCPI Command Reference

SSAMple The Sub SAMple sequence is used to control the pacing of steps needed to create a sample. Sub SAMple is used where a SAMple point is composed of more than one acquisition on generation point.

SSAMple is subservient to SAMple; SSAMple exits IDLE upon TRIGger:SAMple:DOWN. The upward event detector in TRIGger:SAMple is linked to INITiated:SSAMple:UP. Similarly, SAMple is subservient to SVEctor and SVEctor is subservient to VECTOR. Further, VECTOR is subservient to VECTOR2. This relationship is shown in Fig 24-6. The names and number of subservient sequences may vary between instruments, but the structure is the same.

1999 SCPI Command Reference

- ◆ The keywords ABORt, ARM, INITiate, and TRIGger are at the root level in the command hierarchy. They are grouped here because of their close functional relationships. ◆

KEYWORD	PARAMETER FORM	NOTES
ABORt		[no query] 1991
ARM		1991
[:SEQuence]		1994
:DEFine	<sequence_name>	1994
:MGRules	<Boolean>	1994
[:LAYer]		
:COUNT	<numeric_value>	
:COUPling	ACIDC	
:DELay	<numeric_value>	
:AUTO	<Boolean> ONCE	
:ECL		[no query]
:ECOut	<numeric_value>	
:FILTer		1991
:HPASs		1991
FREQuency	<numeric_value>	1991
[:STATe]	<Boolean>	1991
[:LPASs]		1991
FREQuency	<numeric_value>	1991
[:STATe]	<Boolean>	1991
:HYSTeresis	<numeric_value>	
[:IMMediate]		[no query]
:LEVel	<numeric_value>	
:AUTO	<Boolean> ONCE	1994
:LINK	<event_handle>	
:PROTOcol		1994
:VXI	SYNChronous SSYNchronous ASYNchronous	1994
:SIGNal		[no query]
:SLOPe	POSitive NEGative EITHer	
:SOURce	AINternallBUSIECLTrg<n> EXTernallHOLD IMMediate INTernallLINE LINK MANuallOUTPut TIMER TTLTrg<n>	1995
:TIMer	<numeric_value>	
:TTL		[no query]
:TYPE	EDGE VIDEo	1994
:VIDeo		1994
:FIELD		1994
[:NUMBer]	<numeric_value>	1994
:SElect	ODD EVEN ALL NUMBer	1994
:FORMat		1994
:LPFRame	<numeric_value>	1994
:LINE		1994,5

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
[:NUMBer]	<numeric_value>	1995
:SElect	ALL NUMBer	1995
:SSIGnal		1994
[:POLarity]	POSitive NEGative	1994
INITiate		1991
:CONTinuous	<Boolean>	
[:ALL]	<Boolean>	1994
:NAME	<sequence_name>,<Boolean>	1994
:SEquence	<Boolean>	1994
[:IMMediate]		[no query]
[:ALL]		[no query]
:NAME	<sequence_name>	[no query]
:SEquence		[no query]
:POFLag	INCLude EXCLude	1995
TRIGger		1991
[:SEquence]		1994
:ATRigger		1994
[:STATe]	<Boolean>	1994
:COUNT	<numeric_value>	
:COUPling	AC DC	
:DEFine	<sequence_name>	1994
:MGRules	<Boolean>	1994
:DELay	<numeric_value>	
:AUTO	<Boolean> ONCE	
:ECL		[no query]
:ECOUNT	<numeric_value>	
:FILTer		1991
:HPASs		1991
:FREQuency	<numeric_value>	1991
[:STATe]	<Boolean>	1991
[:LPASs]		1991
:FREQuency	<numeric_value>	1991
[:STATe]	<Boolean>	1991
:HOLDoff	<numeric_value>	1994
:HYSTeresis	<numeric_value>	
[:IMMediate]		[no query]
:LEVel	<numeric_value>	
:AUTO	<Boolean> ONCE	1994
:LINK	<event_handle>	
:PROTOcol		1994
:VXI	SYNChronous SSYNchronous ASYNchronous	1994
:SIGNal		[no query]
:SLOPe	POSitive NEGative EITHER	

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
:SOURce	AINternallBUSIECLTrg<n> EXTernallHOLD IIMMediate INTernallLINE LINK MANuallOUTPut TIMER TTLTrg<n>	1995
:TIMer	<numeric_value>	
:TTL		[no query]
:TYPE	EDGE VIDEo	1994
:VIDeo		1994
:FIELd		1994
[:NUMBer]	<numeric_value>	1994
:SELEct	ODD EVEN ALL NUMBer	1994
:FORMat		1994
:LPFRame	<numeric_value>	1994
:LINE		1994,5
[:NUMBer]	<numeric_value>	1995
:SELEct	ALL NUMBer	1995
:SSIGnal		1994
[:POLarity]	POSitive NEGative	1994

24.5 **ABORt** ABORt

The ABORt command resets the trigger system and places all trigger sequences in the IDLE state. Any actions related to the trigger system that are in progress, such as a sweep or acquiring a measurement, shall also be aborted as quickly as possible. The ABORt command shall not be considered complete until all trigger sequences are in the IDLE state. The execution of an ABORt command shall set false the pending operation flags that were set by the initiation of the trigger system.

This command is an event and has no associated *RST condition or query form.

24.6 **ARM** ARM

The purpose of the ARM subsystem is to qualify a single event or a sequence of events before enabling the TRIGger. The ARM subsystem for a given sequence may be omitted if TRIGger does not require additional qualification by other events.

24.6.1 **[:SEQUence]** ARM:SEQUence

SEQUence is used in the expanded capability model to identify a particular sequence of layers in the trigger system. A <sequence_name>, see 24.4.2, may also be used here.

24.6.1.1 **:DEFine <sequence_name>** ARM:SEQUence:DEFine

Sets or queries the SEQUENCE alias. This command is required where aliases are employed and they do not correspond to the standard mapping. The numeric suffix on SEQUENCE corresponds to the sequence number to which the alias is to be applied.

1999 SCPI Command Reference

This command is an alias to TRIGger:SEquence:DEFine. Changing the value of ARM:SEquence:DEFine changes the value of TRIGger:SEquence:DEFine to the same value.

The <sequence_name> is character data.

The effect of <sequence_name> is modified by DEFine:MGRules as follows:

- Regardless of the state or presence of DEFine:MGRules, there is no difference between uppercase and lowercase letters used in <sequence_name>.
- If the command DEFine:MGRules is absent or set to OFF, the full<sequence_name> as written is the only form allowed.

For example, if the <sequence_name> were SAMPLE1, then SAMPLE1 is the only alias accepted.

- If the command DEFine:MGRules is present and set to ON, the <sequence_name> is interpreted as <program_mnemonic>[<numeric_suffix>] where all the contiguous trailing digits form the <numeric_suffix>. Mnemonic generation rules as described in Syntax and Style, Section 6.2.1. shall apply to the <program_mnemonic> and the <numeric suffix>.

For example, if the <sequence_name> were defined as VECTOR or VECTOR1, then VECT, VECT1, VECTOR, and VECTOR1 would all be acceptable aliases.

If the <sequence_name> is identical to a keyword under ARM or TRIGger subsystem, the device shall report execution error -224. For example ARM:SEquence2:DEFine FILTER will cause an error. This command shall accept any other valid character program data for <sequence_name>.

The query returns a string which contains the <sequence_name>, so that a null string can indicate that nothing is defined.

If the command DEFine:MGRules is present and set to ON, the returned <sequence_name> will follow the upper/lower case convention described in Syntax and Style, section 5.1, Interpreting Command Tables. An omitted <numeric_suffix> or a <numeric_suffix> of 1 will be represented as [1].

For example, if <sequence_name> is VECTOR1 then the response is “VECTor[1]”; if it is DELAYED2, the response is “DELayed2”; if it is SAMPLE, the response is “SAMPLE[1]”.

By executing this command the previously defined name is overwritten.

*RST has no effect on the defined name.

24.6.1.1.1 MGRules <Boolean>

ARM:SEquence:DEFine:MGRules

This command sets or queries the state of MGRules, Mnemonic Generation Rules.

At *RST, the value of this parameter is OFF.

24.6.1.2 **[[:LAYer]**

ARM:SEquence:LAYer

LAYer is used in the expanded capability model to identify a particular ARM LAYer in the trigger system.

24.6.1.2.1 **:COUNT <numeric_value>**

ARM:SEquence:LAYer:COUNT

COUNT controls the path of the trigger system in the upward traverse of the event detection layer, shown in figure 24-4, “Event Detection Layer.” The trigger system is directed either to the next layer up or to loop back through the downward traverse. Moving to the next layer up occurs when the number of times that the trigger system has passed through the downward traverse (given by the value of loop_ctr) is equal to the specified COUNT. In the simple case, COUNT has the value 1, a single downward traverse through the layer occurs, followed by the actions dictated by subservient layers, and finally a single upward traverse to next layer up. COUnT shall be set to a value of 1 or greater.

At *RST, this value is set to 1.

24.6.1.2.2 **:COUPling ACIDC**

ARM:SEquence:LAYer:COUPling

COUPling only has effect if the source for the event detector is an analog electrical signal, such as INternal, LINE, or EXternal. It selects AC or DC coupling for the SOURced signal. DC coupling allows the signal’s AC and DC components to pass. AC coupling passes only the signal’s AC component.

At *RST, this value is device-dependent.

24.6.1.2.3 **:DELay <numeric_value>**

ARM:SEquence:LAYer:DELay

DELay sets the time duration between the recognition of an event(s) and the downward exit of the specified layer. DELay must be either zero or a positive value. An attempt to set it to a negative value shall cause an error -222 to be generated.

At *RST, this value is set to 0 or the smallest available positive value. The units for DELay are seconds.

24.6.1.2.3.1 **:AUTO <Boolean>IONCE**

ARM:SEquence:LAYer:DELay:AUTO

This command is used for sensor devices with internal settling delays. If ON, the delay is set to the minimum necessary to ensure that a valid measurement can be acquired. This value may be coupled to other instrument settings such as function, range, resolution, and input bandwidth.

At *RST, AUTO is set ON.

24.6.1.2.4 :ECL

ARM:SEquence:LAYer:ECL

This command is an event which presets LEVel, HYSTeresis, COUPLing, and DELay to values appropriate for a ECL signal. This command cannot be queried.

24.6.1.2.5 :ECount <numeric_value>

ARM:SEquence:LAYer:ECount

ECount specifies a particular number of occurrences of the same event that must be recognized. For example where it is necessary to wait for the tenth positive edge of a signal, ECount must be set to 10. ECount must be a positive value of 1 or greater. An attempt to set it differently shall cause an error -222 to be generated.

At *RST, this value is set to 1.

24.6.1.2.6 :FILTer

ARM:SEquence:LAYer:FILTer

This function allows a filter to be inserted between the source switch and event detector. FILTer only has effect if :SOURce is an analog switch such as INTernal, EXTernal, or LINE.

24.6.1.2.6.1 :HPASs

ARM:SEquence:LAYer:FILTer:HPASs

Controls the high pass filter.

24.6.1.2.6.1.1 :FREQuency <numeric_value>

ARM:SEquence:LAYer:FILTer:HPASs:FREQuency

Determines the cutoff frequency of the high pass filter. Default units are Hertz.

At *RST, this value is device-dependent.

24.6.1.2.6.1.2 [:STATe] <Boolean>

ARM:SEquence:LAYer:FILTer:HPASs:STATe

Turns the high pass filter ON or OFF.

At *RST, this value is device-dependent.

24.6.1.2.6.2 [:LPASs]

ARM:SEquence:LAYer:FILTer:LPASs

Controls the low pass filter.

24.6.1.2.6.2.1 :FREQuency <numeric_value>

ARM:SEquence:LAYer:FILTer:LPASs:FREQuency

Determines the cutoff frequency of the low pass filter. Default units are Hertz.

At *RST, this value is device-dependent.

24.6.1.2.6.2.2 [:STATe] <Boolean>

ARM:SEquence:LAYer:FILTer:LPASs:STATe

Turns the low pass filter ON or OFF.

At *RST, this value is device-dependent.

24.6.1.2.7 :HYSTeresis <numeric_value>

ARM:SEquence:LAYer:HYSTeresis

HYSTeresis is a qualifier of LEVel. It sets how far a signal must fall below LEVel before a rising edge can again be detected, and how far a signal must rise above LEVel before a falling edge can again be detected. Its function is to eliminate false events caused by signal noise. Units for the parameter default to the current amplitude units.

In instruments that have only two discrete values for hysteresis (such as Noise Rejection On and Noise Rejection Off), the device designer should pick two numeric values that approximate the actual hysteresis settings. The :HYSTeresis command has no :STATe function because confusion would occur when turning hysteresis off. This might imply to a user that the actual hysteresis had been set to 0. This is rarely the case in instruments that have only two states.

At *RST, this value is device-dependent.

24.6.1.2.8 [:IMMediate]

ARM:SEquence:LAYer:IMMediate

This command provides a onetime override of the normal process of the downward traverse of the event detection layer. This command shall cause the immediate exit of the specified event detection layer if the trigger system is in the specified layer when the IMMediate is received. Otherwise, the IMMediate command shall be ignored and an error -212 shall be generated.

The actual event detection and delay shall be skipped; however, normal processing of the loop_ctr as shown in figure 24-4, “Event Detection Layer” shall occur. This command has no effect on any other settings in this subsystem.

This command is an event and has no *RST condition and cannot be queried.

24.6.1.2.9 :LEVel <numeric_value>

ARM:SEquence:LAYer:LEVel

LEVel qualifies the characteristic of the selected SOURce signal that generates an event. LEVel only has effect if the source for the event detector is an analog electrical signal, such as INTernal, LINE, or EXTernal. Units for the parameter default to the current amplitude unit.

At *RST, this value is instrument-dependent.

24.6.1.2.9.1 :AUTO <Boolean> | ONCE

ARM:SEquence:LAYer:LEVel:AUTO

When AUTO is ON, the device dynamically selects the best arm level based on a device-dependent algorithm applied to the arm signal.

When AUTO is OFF, the arm LEVel sets the arm level.

Setting AUTO to ONCE turns AUTO ON, adjusts the arm level once and then resets AUTO to OFF.

Setting the arm LEVel turns AUTO OFF.

At *RST, the value of this parameter is OFF.

24.6.1.2.10 :LINK <event_handle>

ARM:SEquence:LAYer:LINK

Sets or queries the internal event that is LINKed to the event detector in the ARM layer. Internal events occur throughout all of the Instrument Model, a particular implementation will dictate which events are available to this LINK command. The <event_handle> is STRING PROGRAM DATA and it is defined further in the Instrument Model, Chapter 2.

An Example of events that are commonly used, are the events that occur when a particular layer is exited, these handles are of the form:

“ARM[:SEquence[:LAYer]]” or “TRIGger[:SEquence]”

Other common examples include completion of a process in a block or an instrument state transition.

At *RST, this value is device-dependent.

24.6.1.2.11 PROTOcol

ARM:SEquence:LAYer:PROTOcol

This subsystem controls the protocol used with the selected source.

24.6.1.2.11.1 VXI SYNChronous|SSYNchronous|ASYNchronous

ARM:SEquence:LAYer:PROTOcol:VXI

This command selects the trigger protocol for the VXI TTLTrg or ECLTrg trigger line when used as the arm source. The protocols are specified in the VXI Bus System Specification under sections B.6.2.3 and B.6.2.4.

- SYNChronous selects the synchronous trigger protocol.
- SSYNchronous selects the semi-synchronous trigger protocol.
- ASYNchronous selects the asynchronous trigger protocol.

At *RST, the SYNChronous protocol is selected.

24.6.1.2.12 :SIGNal

ARM:SEquence:LAYer:SIGNal

This command provides a one-time override of the normal process of the downward traverse through the event detector in figure 24-4, “Event Detection Layer.” This command shall cause the immediate exit of the event detector block in the specified event detection layer if the trigger system is waiting for the event specified by the SOURce command. Otherwise, the SIGNal command shall be ignored and an error -212 shall be generated.

Only the event detector block is bypassed. Normal processing of the delay and the event_ctr as shown in figure 24-4, “Event Detection Layer,” shall occur. This command has no effect on any other settings in this subsystem.

This command defines an event, and as such does not have an associated query form or *RST condition.

24.6.1.2.13 :SLOPe POSitiveNEGativeEITHer

ARM:SEquence:LAYer:SLOPe

SLOPe qualifies whether the event occurs on the rising edge, falling edge, or either edge of the signal.

At *RST, this value is set to POS.

24.6.1.2.14 :SOURce <parameter>

ARM:SEquence:LAYer:SOURce

This command selects the source for the event detector. Only one source may be specified at a time for a given event detector. The various sources are:

- AINternal — The triggering source is selected automatically by the instrument from internal algorithms to best match the current operating mode. An Automatic INternal event is derived from measurement functions and/or sensor capabilities in the signal conditioning block.
- BUS — The source is signal specific to the control interface. For *IEEE 488.1*, the group execute trigger, GET, would satisfy this condition. In VXI the word serial command *TRIGger* performs this function. The event detector is also satisfied, independent of the interface, when a *TRG command is received. Note that neither GET nor *TRG can be sent without having an effect on the message exchange protocol described in *IEEE 488.2*.
- ECLTrg — The signal source is the specified VXI P2 or P3 backplane ECLTrg TRIGGER line. Valid lines are ECLTrg0 and ECLTrg1 on P2, and ECLTrg2 through ECLTrg5 on P3.
- EXTERNAL — An external signal jack is selected as the source. If no suffix is specified, EXTERNAL1 is assumed.
- HOLD — The event detection is disabled. However, the IMMEDIATE command shall override HOLD.
- IMMEDIATE — No waiting for an event occurs.
- INTERNAL — An internal channel is selected as the source. An INTERNAL event is derived from a measurement function and/or sensor capability in the signal conditioning block. If no suffix is specified, INTERNAL1 is assumed.
- LINE — The source signal is determined from the AC line voltage.
- LINK — The source for the event detector is specified by the LINK command.
- MANUAL — The signal is user-generated, such as by pressing a front panel key.
- OUTPUT — The signal source is taken from an output channel. If no channel is specified, OUTPUT1 is assumed.

- **TIMer** — The source signal comes from a periodic timer. The period of the timer is specified by the **TIMer** command.
- **TTLTrg** — The signal source is the specified VXI P2 backplane TTLTrg TRIGGER line. Valid lines are TTLTrg0 through TTLTrg7.

At *RST, IMMEDIATE shall be selected as the SOURCE. This value was chosen so that devices that do not implement any ARM layers (and hence the controls for the ARM layer) behave at *RST in the same manner as devices that do.

24.6.1.2.15 :TIMer <numeric_value>

ARM:SEquence:LAYer:TIMer

TIMer sets the period of an internal periodic signal source. Its value affects the trigger system only when it is selected as the SOURCE for the event detector. TIMer must be a positive value. An attempt to set it differently shall cause an error -222 to be generated. The synchronization of the timer's period is device-dependent.

At *RST, this value is device-dependent value. The default units for this value is seconds.

24.6.1.2.16 :TTL

ARM:SEquence:LAYer:TTL

This command is an event which presets LEVEL, HYSTERESIS, COUPLING, and DELAY to values appropriate for a TTL signal. This command cannot be queried.

24.6.1.2.17 :TYPE EDGE I VIDEo

ARM:SEquence:LAYer:TYPE

Sets or queries the type of triggering.

At *RST, the value of this parameter is EDGE.

24.6.1.2.18 :VIDeo

ARM:SEquence:LAYer:VIDeo

This subsystem controls parameters necessary to trigger on video signals.

24.6.1.2.18.1 :FIEld

ARM:SEquence:LAYer:VIDeo:FIEld

This subsystem selects the video field to trigger on.

24.6.1.2.18.1.1 [:NUMBer] <numeric_value>

ARM:SEquence:LAYer:VIDeo:FIEld:NUMBer

NUMBer sets or queries the field number to trigger on if FIEld:SElect is set to NUMBer.

At *RST, the value of this parameter is 1.

24.6.1.2.18.1.2 :SElect ODD I EVEN I ALL I NUMBer

ARM:SEquence:LAYer:VIDeo:FIEld:SElect

SElect sets or queries the video field selection method to be used. Four character data parameters are defined as follows:

- ODD selects odd-numbered fields to trigger on.
- EVEN selects even-numbered fields to trigger on.
- ALL triggers on all fields regardless of number.
- NUMBer triggers on the field number specified by NUMBer.

At *RST, the value of this parameter is ALL.

24.6.1.2.18.2 :FORMat

ARM:SEquence:LAYer:VIDeo:FORMat

FORMat controls parameters that allow the video trigger system to respond to signals from standard video formats.

24.6.1.2.18.2.1 :LPFRame <numeric_variable>

ARM:SEquence:LAYer:VIDeo:FORMat:LPFRame

LPFRame, Lines Per FRame, sets or queries the lines per frame associated with the signal formatting standard being used.

For example, A signal following NTSC would select 525 lines per frame while a PAL signal would select 625 lines per frame.

At *RST, the value of this parameter is device dependent.

24.6.1.2.18.3 :LINE

ARM:SEquence:LAYer:VIDeo:LINE

This subsystem selects a particular line in the field to arm on.

24.6.1.2.18.3.1 [:NUMBer] <numeric_value>

ARM:SEquence:LAYer:VIDeo:LINE:NUMBer

NUMBer sets or queries the line number to arm on if LINE:SElect is set to NUMBer. Setting the NUMBer value has no effect on the value of the SElect parameter.

At *RST, the value of this parameter is device dependent.

24.6.1.2.18.3.2 :SElect ALL | NUMBer

ARM:SEquence:LAYer:VIDeo:LINE:SElect

SElect sets or queries the video line selection method to be used. Two character data parameters are defined as follows:

- ALL — arms on all lines regardless of number.
- NUMBer — arms on the line number specified by NUMBer.

At *RST, the value of this parameter is ALL.

24.6.1.2.18.4 :SSIGnal

ARM:SEquence:LAYer:VIDeo:SSIGnal

SSIGnal, Synchronizing SIGnal, controls parameters relating to the video synchronizing signal.

24.6.1.2.18.4.1 :POLarity POSitive I NEGative

ARM:SEquence:LAYer:VIDeo:SSIGnal:POLarity

SSIGnal:POLarity, sets or queries sync pulse triggering polarity.

At *RST, the value of this parameter is NEGative.

24.7 INITiate

INITiate

The INITiate subsystem is used to control the initiation of the trigger subsystem. It initiates all trigger sequences as a group except those sequences that are defined otherwise by either the standard or in the instrument documentation.

Instruments which implement the INITiate command as overlapped can continue parsing and executing subsequent commands and queries while initiated. Instruments which implement the INITiate commands as sequential cannot execute subsequent commands and queries until the trigger model has returned to the IDLE state.

For this reason, devices which implement the INITiate commands as sequential shall enter IDLE when the IEEE 488.2 **dcas** message is received. They shall also execute the next message unit received after the **dcas** before leaving IDLE (if INITiate:CONTinuous is ON). **dcas** followed by *RST or **dcas** followed by INIT:CONT OFF shall always stop the trigger model in the IDLE state.

There are advantages to implementing the INITiate commands as overlapped.

- The instrument designer decides whether or not a device clear aborts the trigger model.
- The instrument can continue parsing and executing commands and queries while the trigger model is initiated (for example, FETCh? to get the readings which are being triggered in a meter).

The disadvantage to implementing any commands as overlapped, including the INITiate commands, is that it adds complexity to the implementation of the IEEE 488.2 common commands *OPC, *WAI and *OPC?

24.7.1 :CONTinuous <Boolean>

INITiate:CONTinuous

The CONTinuous command is used to select whether the trigger system is continuously initiated or not. With CONTinuous set to OFF, the trigger system shall remain in the IDLE state until CONTinuous is set to ON or INITiate:IMMediate is received. Once CONTinuous is set to ON, the trigger system shall be initiated and shall exit the IDLE state. On completion of each trigger cycle, with CONTinuous ON, the trigger system shall immediately commence another trigger cycle without entering the IDLE state.

When INITiate:CONTinuous is set to OFF, the current trigger cycle shall be completed before entering the IDLE state. The return to IDLE shall also occur as the result of an ABORT or *RST command.

1999 SCPI Command Reference

With the trigger system set to cycle continuously, the ABORT command shall force the trigger system to the IDLE state; however, the value of INITiate:CONTinuous is unaffected. If CONTinuous was set to ON prior to receiving ABORT, it shall remain ON and the trigger system shall immediately exit the IDLE state.

This subsystem does not apply to sequences that are initiated by the INITiate:ALL command.

24.7.1.1 **[[:ALL]] <Boolean>**

INITiate:CONTinuous:ALL

Sets or queries whether or not all sequences are continuously initiated.

At *RST, this value is set to OFF.

24.7.1.2 **:NAME <sequence_name>, <Boolean>**

INITiate:CONTinuous:NAME

Sets or queries whether or not the SEQUENCE with the alias specified by <sequence_name> is continuously initiated. The <sequence_name> is character program data. See 24.4.2 for standard sequence names.

The <sequence_name> is a required parameter of the query form, the query form returns 0 or 1.

At *RST, this value is set to OFF.

24.7.1.3 **:SEQUence <Boolean>**

INITiate:CONTinuous:SEQUence

Sets or queries whether or not the specified SEQUENCE is continuously initiated. The numeric suffix on SEQUENCE corresponds to the sequence number. If NAME is implemented, this command shall also be implemented.

At *RST, this value is set to OFF.

24.7.2 **[[:IMMediate]]**

INITiate:IMMediate

These commands shall cause all sequences to exit the IDLE state; they are initiated. The IMMediate command shall cause the trigger system to initiate and complete one full trigger cycle, returning to IDLE on completion. If the device is not in IDLE or if INITiate:CONTinuous is set to ON, an IMMediate command shall have no effect on the trigger system and an error -213 shall be generated.

INITiate[:IMMediate] is an event and cannot be queried as there is no state associated with it.

24.7.2.1 **[[:ALL]]**

INITiate:IMMediate:ALL

This command causes all SEQUences to be INITiated, except those defined to behave otherwise.

24.7.2.2 **:NAME <sequence_name>**

INITiate:IMMEDIATE:NAME

This command causes the SEQUENCE with the alias specified by <sequence_name> to be INITiated. The <sequence_name> is character program data. See 24.4.2 for standard sequence names.

24.7.2.3 **:SEQUENCE**

INITiate:IMMEDIATE:SEQUENCE

This command causes the specified SEQUENCE to be INITiated. The numeric suffix on SEQUENCE corresponds to the sequence number. If NAME is implemented, this command shall also be implemented.

24.7.3 **:POFLAG INCLUDE | EXCLUDE**

INITiate:POFLAG

POFLAG, Pending Operation Flag, allows the Pending-Operation flag associated with the initiation of a trigger sequence to be included or excluded from the No-Operation-Pending flag set.

Setting POFLAG to EXCLUDE means the No-Operation-Pending flag set is independent of the state of the trigger subsystem. Moving from the INITiated to IDLE trigger states and vice versa has no effect on the NOP flag in this state.

Setting POFLAG to INCLUDE means the No-Operation-Pending flag set includes the trigger subsystem's Pending-Operation flag which goes true when the trigger subsystem moves from IDLE to INITiated state and returns to false when IDLE is re-entered.

At *RST, the value of this parameter is INCLUDE.

24.8 **TRIGGER**

TRIGGER

The purpose of the TRIGGER subsystem is to qualify a single event before enabling the triggered sequence operation, such as enabling a sweep, starting a measurement, or changing the state of the device.

24.8.1 **[:SEQUENCE]**

TRIGGER:SEQUENCE

SEQUENCE is used in the expanded capability model to identify a particular sequence of layers in the trigger system. A <sequence_name>, see 24.4.2, may also be used here.

24.8.1.1 **:ATRIGGER**

TRIGGER:SEQUENCE:ATRIGGER

ATRigger, Auto TRigger, controls an event detector timer that causes control flow to exit the event detector if a valid trigger does not appear within a preset time period. The auto trigger timer is started upon entering the TRIGGER layer, and after the control flow leaves the event detector. The precise timing of the timer is device dependent. See Figure 24-4.

24.8.1.1.1 **[[:STATe] <Boolean>**

TRIGger:SEquence:ATRigger:STATe

STATe sets and queries the state of the auto trigger function. STATe ON enables the function and STATe OFF disables it.

At *RST, the value of this parameter is OFF.

24.8.1.2 **:COUNT <numeric_value>**

TRIGger:SEquence:COUNt

COUNt controls the path of the trigger system in the upward traverse of the event detection layer, shown in figure 24-4, “Event Detection Layer.” The trigger system is directed either to the next layer up or to loop back through the downward traverse. Moving to the next layer up occurs when the number of times that the trigger system has passed through the downward traverse (given by the value of loop_ctr) is equal to the specified COUNt. In the simple case, COUNt has the value 1, a single downward traverse through the layer occurs, followed by the actions dictated by subservient layers and finally a single upward traverse to next layer up. COUNt shall be set to a value of 1 or greater.

At *RST, this value is set to 1.

24.8.1.3 **:COUPling ACIDC**

TRIGger:SEquence:COUPling

COUPling only has effect if the source for the event detector is an analog electrical signal, such as INternal, LINE, or EXternal. It selects AC or DC coupling for the SOURced signal. DC coupling allows the signal’s AC and DC components to pass. AC coupling passes only the signal’s AC component.

At *RST, this value is device-dependent.

24.8.1.4 **:DEFine <sequence_name>**

TRIGger:SEquence:DEFine

Sets or queries the SEQUENCE alias. This command is required where aliases are employed and they do not correspond to the standard mapping. The numeric suffix on SEQUENCE corresponds the sequence number to which the alias is to be applied.

This command is an alias to ARM:SEquence:LAYer:DEFine. Changing the value of TRIGger:SEquence:DEFine changes the value of ARM:SEquence:DEFine to the same value.

The <sequence_name> is character data.

The effect of <sequence_name> is modified by DEFine:MGRules as follows:

- Regardless of the state or presence of DEFine:MGRules, there is no difference between uppercase and lowercase letters used in <sequence_name>.
- If the command DEFine:MGRules is absent or set to OFF, the full<sequence_name> as written is the only form allowed.

For example, if the <sequence_name> were SAMPLE1, then SAMPLE1 is the only alias accepted.

- If the command DEFine:MGRules is present and set to ON, the <sequence_name> is interpreted as <program_mnemonic>[<numeric_suffix>] where all the contiguous trailing digits form the <numeric_suffix>. Mnemonic generation rules as described in Syntax and Style, Section 6.2.1. shall apply to the <program_mnemonic> and the <numeric suffix>.

For example, if the <sequence_name> were defined as VECTOR or VECTOR1, then VECT, VECT1, VECTOR, and VECTOR1 would all be acceptable aliases.

If the <sequence_name> is identical to a keyword under ARM or TRIGger subsystem, the device shall report execution error -224. For example ARM:SEQuence2:DEFine FILTER will cause an error. This command shall accept any other valid character program data for <sequence_name>.

The query returns a string which contains the <sequence_name>, so that a null string can indicate that nothing is defined.

If the command DEFine:MGRules is present and set to ON, the returned <sequence_name> will follow the upper/lower case convention described in Syntax and Style, section 5.1, Interpreting Command Tables. An omitted <numeric_suffix> or a <numeric_suffix> of 1 will be represented as [1].

For example, if <sequence_name> is VECTOR1 then the response is “VECTor[1]”; if it is DELAYED2, the response is “DELayed2”; if it is SAMPLE, the response is “SAMPlE[1]”.

By executing this command the previously defined name is overwritten.

*RST has no effect on the defined name.

24.8.1.4.1 MGRules <Boolean>

TRIGger:SEQuence:DEFine:MGRules

This command sets or queries the state of MGRules, Mnemonic Generation Rules.

At *RST, the value of this parameter is OFF.

24.8.1.5 :DELay <numeric_value>

TRIGger:SEQuence:DELay

DELay sets the time duration between the recognition of an event(s) and the downward exit of the specified layer. DELay must be either zero or a positive value. An attempt to set it to a negative value shall cause an error -222 to be generated.

At *RST, this value is set to 0 or the smallest available positive value. The value has units of seconds.

24.8.1.5.1 :**AUTO** <Boolean>**IONCE**

TRIGger:SEQuence:DELay:AUTO

This command is used for sensor devices with internal settling delays. If ON, the delay is set to the minimum necessary to ensure that a valid measurement can be acquired. This value may be coupled to other instrument settings such as function, range, resolution, and input bandwidth.

At *RST, AUTO is set ON.

24.8.1.6 :**ECL**

TRIGger:SEQuence:ECL

This command is an event which presets LEVel, HYSTeresis, COUPLing, and DELay to values appropriate for a ECL signal. This command cannot be queried.

24.8.1.7 :**ECOut** <numeric_value>

TRIGger:SEQuence:ECOut

ECOut specifies a particular number of occurrences of the same event that must be recognized. For example, where it is necessary to wait for the tenth positive edge of a signal, ECOut must be set to 10. ECOut must be a positive value of 1 or greater. An attempt to set it differently shall cause an error -222 to be generated.

At *RST, this value is set to 1.

24.8.1.8 :**FILTer**

TRIGger:SEQuence:FILTer

This function allows a filter to be inserted between the source switch and event detector. FILTer only has effect if :SOURce is an analog switch such as INTernal, EXTernal, or LINE.

24.8.1.8.1 :**HPASs**

TRIGger:SEQuence:FILTer:HPASs

Controls the high pass filter.

24.8.1.8.1.1 :**FREQuency** <numeric_value>

TRIGger:SEQuence:FILTer:HPASs:FREQuency

Determines the cutoff frequency of the high pass filter. Default units are Hertz.

At *RST, this value is device-dependent.

24.8.1.8.1.2 [:**STATe**] <Boolean>

TRIGger:SEQuence:FILTer:HPASs:STATe

Turns the high pass filter ON or OFF.

At *RST, this value is device-dependent.

24.8.1.8.2 [:**LPASs**]

TRIGger:SEQuence:FILTer:LPASs

Controls the low pass filter.

24.8.1.8.2.1 :FREQuency <numeric_value>

TRIGger:SEQuence:FILTer:LPASs:FREQuency

Determines the cutoff frequency of the low pass filter. Default units are Hertz.

At *RST, this value is device-dependent.

24.8.1.8.2.2 [:STATe] <Boolean>

TRIGger:SEQuence:FILTer:LPASs:STATe

Turns the low pass filter ON or OFF.

At *RST, this value is device-dependent.

24.8.1.9 :HOLDoff <numeric_value>

TRIGger:SEQuence:HOLDoff

HOLDoff controls the time during which the event detector is inhibited from acting on any new trigger. The exact point in the trigger model at which the trigger holdoff timer is started is device dependent, but it shall be after the event detector on the downward traverse and before the event detector on the upward traverse. See Figure 24-4.

The range of the parameter is zero to one, where one is maximum holdoff and zero is minimum.

At *RST, the value of the parameter is zero.

24.8.1.10 :HYSTeresis <numeric_value>

TRIGger:SEQuence:HYSTeresis

HYSTeresis is a qualifier of LEVel. It sets how far a signal must fall below LEVel before a rising edge can again be detected, and how far a signal must rise above LEVel before a falling edge can again be detected. Its function is to eliminate false events caused by signal noise. Units for the parameter default to the current amplitude units.

In instruments that have only two discrete values for hysteresis (such as Noise Rejection On and Noise Rejection Off), the device designer should pick two numeric values that approximate the actual hystereis settings. The :HYSTeresis command has no :STATe function because confusion would occur when turning hysteresis off. This might imply to a user that the actual hysteresis had been set to 0. This is rarely the case in instruments that have only two states.

At *RST, this value is device-dependent.

24.8.1.11 [:IMMediate]

TRIGger:SEQuence:IMMediate

This command exists to provide a one-time override of the normal process of the downward traverse of the event detection layer. This command shall cause the immediate exit of the specified event detection layer if the trigger system is in the specified layer when the IMMediate is received. Otherwise, the IMMediate command shall be ignored and an error -211 shall be generated.

1999 SCPI Command Reference

The actual event detection and delay shall be skipped; however, normal processing of the loop_ctr as shown in figure 24-4, “Event Detection Layer,” shall occur. This command has no effect on any other settings in this subsystem.

This command is an event, has no *RST condition, and cannot be queried.

24.8.1.12 :LEVel <numeric_value>

TRIGger:SEquence:LEVel

LEVel qualifies the characteristic of the selected SOURce signal that generates an event. LEVel only has effect if the source for the event detector is an analog electrical signal, such as INTERNAL, LINE, or EXTERNAL. Units for the parameter default to the current amplitude unit.

At *RST, this value is instrument-dependent.

24.8.1.12.1 :AUTO <Boolean> | ONCE

TRIGger:SEquence:LEVel:AUTO

When AUTO is ON, the device dynamically selects the best trigger level based on a device-dependent algorithm applied to the trigger signal.

When AUTO is OFF, the trigger LEVel sets the trigger level.

Setting AUTO to ONCE turns AUTO ON, adjusts the trigger level once and then resets AUTO to OFF.

Setting the trigger LEVel turns AUTO OFF.

At *RST, the value of this parameter is OFF.

24.8.1.13 :LINK <event_handle>

TRIGger:SEquence:LINK

Sets or queries the internal event that is LINKed to the event detector in the TRIGger layer. Internal events occur throughout all of the Instrument Model, a particular implementation will dictate which events are available to this LINK command. The <event_handle> is STRING PROGRAM DATA and it is defined further in the Instrument Model, Chapter 2.

An Example of events that are commonly used, are the events that occur when a particular layer is exited, these handles are of the form:

“ARM[:SEquence[:LAYer]]” or “TRIGger[:SEquence]”

Other common examples include completion of a process in a block or an instrument state transition.

At *RST, this value is device-dependent.

24.8.1.14 :PROTocol

TRIGger:SEquence:PROTocol

This subsystem controls the protocol used with the selected source.

24.8.1.14.1 :VXI SYNChronous|SSYNchronous|ASYNchronous

TRIGger:SEQuence:PROTOcol:VXI

This command selects the trigger protocol for the VXI TTLTrg or ECLTrg trigger line when used as the trigger source. The protocols are specified in the VXI Bus System Specification under sections B.6.2.3 and B.6.2.4.

- SYNChronous selects the synchronous trigger protocol.
- SSYNchronous selects the semi-synchronous trigger protocol.
- ASYNchronous selects the asynchronous trigger protocol.

At *RST, the SYNChronous protocol is selected.

24.8.1.15 :SIGNal

TRIGger:SEQuence:SIGNal

This command provides a one-time override of the normal process of the downward traverse through the event detector in figure 24-4, “Event Detection Layer.” This command shall cause the immediate exit of the event detector block in the specified event detection layer if the trigger system is waiting for the event specified by the SOURce command. Otherwise, the SIGNAL command shall be ignored and an error -211 shall be generated.

Only the event detector block is bypassed. Normal processing of the delay and the event_ctr as shown in figure 24-4, “Event Detection Layer,” shall occur. This command has no effect on any other settings in this subsystem.

This command defines an event, and as such does not have an associated query form or *RST condition.

24.8.1.16 :SLOPe POSitive|NEGative|EITHer

TRIGger:SEQuence:SLOPe

SLOPe qualifies whether the event occurs on the rising edge, falling edge, or either edge of the signal.

At *RST, this value is set to POS.

24.8.1.17 :SOURce <parameter>

TRIGger:SEQuence:SOURce

This command selects the source for the event detector. Only one source may be specified in the parameter at a time for a given event detector. The various sources are:

- AINternal — The triggering source is selected automatically by the instrument from internal algorithms to best match the current operating mode. An Automatic INternal event is derived from measurement functions and/or sensor capabilities in the signal conditioning block.
- BUS — The source is signal specific to the control interface. For *IEEE 488.1* the group execute trigger, GET, would satisfy this condition. In VXI the word serial command *TRIGger* performs this function. The event detector is also satisfied, independent of the interface, when a *TRG command is received. Note that

1999 SCPI Command Reference

neither GET nor *TRG can be sent without having an effect on the message exchange protocol described in *IEEE 488.2*.

- ECLTrg — The signal source is the specified VXI P2 or P3 backplane ECLTrg TRIGGER line. Valid lines are ECLTrg0 and ECLTrg1 on P2, and ECLTrg2 through ECLTrg5 on P3.
- EXTERNAL — An external signal jack is selected as the source. If no suffix is specified, EXTERNAL1 is assumed.
- HOLD — The event detection is disabled. However, the IMMEDIATE command shall override HOLD.
- IMMEDIATE — No waiting for an event occurs.
- INTERNAL — An internal channel is selected as the source. An INTERNAL event is derived from a measurement function and/or sensor capability in the signal conditioning block. If no suffix is specified, INTERNAL1 is assumed.
- LINE — The source signal is determined from the AC line voltage.
- LINK — The source for the event detector is specified by the LINK command.
- MANUAL — The signal is user-generated, such as by pressing a front panel key.
- OUTPUT — The signal source is taken from an output channel. If no channel is specified, OUTPUT1 is assumed.
- TIMER — The source signal comes from a periodic timer. The period of the timer is specified by the TIMER command.
- TTLTrg — The signal source is the specified VXI P2 backplane TTLTrg TRIGGER line. Valid lines are TTLTrg0 through TTLTrg7.

At *RST, IMMEDIATE shall be selected as the SOURCE.

24.8.1.18 :TIMER <numeric_value>

TRIGGER:SEQUENCE:TIMER

TIMER sets the period of an internal periodic signal source. Its value affects the trigger system only when it is selected as the SOURCE for the event detector. TIMER must be a positive value. An attempt to set it differently shall cause an error -222 to be generated. The synchronization of the timer's period is device-dependent.

At *RST, this value is set to a device-dependent value.

24.8.1.19 :TTL

TRIGGER:SEQUENCE:TTL

This command is an event which presets LEVEL, HYSTERESIS, COUPLING, and DELAY to values appropriate for a TTL signal. This command cannot be queried.

24.8.1.20 :TYPE EDGE | VIDEo

TRIGger:SEquence:TYPE

Sets or queries the type of triggering.

At *RST, the value of this parameter is EDGE.

24.8.1.21 :VIDEo

TRIGger:SEquence:VIDEo

This subsystem controls parameters necessary to trigger on video signals.

24.8.1.21.1 :FIEld

TRIGger:SEquence:VIDEo:FIEld

This subsystem selects the video field to trigger on.

24.8.1.21.1.1 [:NUMBER] <numeric_value>

TRIGger:SEquence:VIDEo:FIEld:NUMBER

NUMBER sets or queries the field number to trigger on if FIEld:SElect is set to NUMBER.

At *RST, the value of this parameter is 1.

24.8.1.21.1.2 :SElect ODD | EVEN | ALL | NUMBER

TRIGger:SEquence:VIDEo:FIEld:SElect

SElect sets or queries the video field selection method to be used. Four character data parameters are defined as follows:

- ODD selects odd-numbered fields to trigger on.
- EVEN selects even-numbered fields to trigger on.
- ALL triggers on all fields regardless of number.
- NUMBER triggers on the field number specified by NUMBER.

At *RST, the value of this parameter is ALL.

24.8.1.21.2 :FORMat

TRIGger:SEquence:VIDEo:FORMat

FORMat controls parameters that allow the video trigger system to respond to signals from standard video formats.

24.8.1.21.2.1 :LPFRame <numeric_variable>

TRIGger:SEquence:VIDEo:FORMat:LPFRame

LPFRame, Lines Per FRame, sets or queries the lines per frame associated with the signal formatting standard being used.

For example, A signal following NTSC would select 525 lines per frame while a PAL signal would select 625 lines per frame.

At *RST, the value of this parameter is device dependent.

24.8.1.21.3 :LINE

TRIGger:SEQuence:VIDeo:LINE

This subsystem selects a particular line in the field to trigger on.

24.8.1.21.3.1 [:NUMBER] <numeric_value>

TRIGger:SEQuence:VIDeo:LINE:NUMBer

NUMBer sets or queries the line number to trigger on if LINE:SElect is set to NUMBer. Setting the NUMBer value has no effect on the value of the SElect parameter.

At *RST, the value of this parameter is device dependent.

24.8.1.21.3.2 :SElect ALL | NUMBer

TRIGger:SEQuence:VIDeo:LINE:SElect

SElect sets or queries the video line selection method to be used. Two character data parameters are defined as follows:

- ALL — triggers on all lines regardless of number.
- NUMBer — triggers on the line number specified by NUMBer.

At *RST, the value of this parameter is ALL.

24.8.1.21.4 :SSIGnal

TRIGger:SEQuence:VIDeo:SSIGnal

SSIGnal, Synchronizing SIGnal, controls parameters relating to the video synchronizing signal.

24.8.1.21.4.1 :POLarity POSitive | NEGative

TRIGger:SEQuence:VIDeo:SSIGnal:POLarity

SSIGnal:POLarity, sets or queries sync pulse triggering polarity.

At *RST, the value of this parameter is NEGative.

25 UNIT Subsystem

Default units are defined, where applicable, for each SCPI command. The UNIT subsystem provides a mechanism to change the default values. The units selected apply to the designated command parameters for both command and response.

The UNIT command at the root level has a global effect on the selected units. The UNIT command may also be applied to lower levels in the SCPI command hierarchy to have a localized effect. When the UNIT command is applied to a node, then all the nodes below the node to which the unit command was applied shall be affected by the localized UNIT command. There is no restriction on the number of levels to which UNIT may be applied. In this way the more global units are overridden by the more local units. Units may also be overridden temporarily by attaching the desired unit as a suffix to the appropriate parameter in the command, if the instrument supports that unit.

For example, to program a source with units of Volts, the command UNIT:VOLTage VOLT would be used. To program only the modulator of that same source in dBuVs, the additional command [SOURce:]MODulation:UNIT DBUV would be required.

The UNIT command cannot be applied to either SENSE, SOURce or ROUTe nodes, since the UNIT command at these nodes and at the root cannot be unambiguously recognized. This condition arises from the ability to have one of SENSE, SOURce or ROUTe nodes optional.

KEYWORD	PARAMETER FORM	NOTES
:UNIT		
:ANGLE	DEGRAD	
:CURRent	<amplitude unit>	
:POWer	<amplitude unit>	
:TEMPerature	CICELIFARIK	
:TIME	HOURLMINute SECond	1993
:VOLTage	<amplitude unit>	

25.1 :ANGLE DEGRAD

UNIT:ANGLE

Specifies the fundamental unit of angle.

At *RST, the default unit is RAD.

25.2 :CURRent, :POWer, and :VOLTage

UNIT:CURRent

Selects a default unit for commands which program absolute amplitude. The default unit may be overridden when programming a function by explicitly sending a suffix. The default unit may also be overridden if a unit command exists at a lower tree level. Query values of commands which program in amplitude units shall be returned in the current amplitude unit

1999 SCPI Command Reference

unless overridden by a unit command at a tree level closer to the command itself (such as POWER:UNIT).

The possible values are:

- **Linear power:** W (Watt)
- **Linear voltage:** V (Volt)
- **Linear current:** A (Ampere)
- **Logarithmic power:** DBM, DB[IEEE suffix multiplier]W (such as DBUW)
- **Logarithmic voltage:** DB[IEEE suffix multiplier]V (such as DBUV)
- **Logarithmic current:** DB[IEEE suffix multiplier]A (such as DBA)

An instrument must accept all values in any class if it accepts any value in that class. However, the decision to accept a class is device-dependent. A device which accepts logarithmic units must also accept the comparable linear unit. This means that UNIT:VOLTage WATT is permissible if it makes sense for the instrument. It is also implied that an impedance is specified for the signal when converting between different fundamental quantities.

At *RST, this value is device-dependent. However, the following restrictions apply. The UNIT subsystem must be implemented if a device chooses any value other than the fundamental unit for either the VOLTage, POWER, or CURRENT subsystem, or if it implements logarithmic units.

25.3 :TEMPerature CICELIFIFARIK

UNIT:TEMPerature

Specifies the fundamental unit of temperature as degrees Celsius, Fahrenheit, or Kelvin. The form FAR and CEL, defined by *IEEE 488.2*, shall be accepted for Fahrenheit and Celsius respectively, and these shall be the preferred forms that are returned for a query.

At *RST, this value is device-dependent.

25.4 :TIME HOURIMINute|SECond

UNIT:TIME

Specifies the fundamental unit of time.

At *RST, this value is SECond.

26 VXI Subsystem

The VXI subsystem contains commands that control the administration functions associated with operating a VXI-based system. This section will track the work of the VXI consortium.

This section describes ASCII commands which are issued to the system from an external host. The exact internal destination device and routing methods to that device are system specific. These commands are used for access to system configuration information, and for common capabilities. The terminology used in this section is patterned after IEEE 488.2.

The following commands are the standard set of VXIbus system commands which are ASCII encoded. If a device implements one or more of these commands, the syntax of this section shall be used. The response syntax shall also follow this section.

The COMMON ASCII SYSTEM COMMANDS are organized into subsystems. If a device implements a required command in a subsystem then it shall implement all the required commands in the subsystem. If a device implements an optional command in a subsystem then it shall implement all the required commands in the subsystem. If a device implements a command from any subsystem then it shall also implement the VXI:SElect command.

KEYWORD	PARAMETER FORM	NOTES
VXI		
:CONFigure		
:DNUMber?		[query only]
:HIERarchy?		[query only] 1992
:ALL?		[query only] 1992
:VERBose?		[query only] 1992
:ALL?		[query only] 1992
:INFormation?		[query only] 1992
:ALL?		[query only] 1992
:VERBose?		[query only] 1992
:ALL?		[query only] 1992
:LADdress?		[query only] 1992
:NUMber?		[query only] 1992
:REGister		1992
:READ?	<numeric_value> <reg_name>	[query only] 1992
:VERBose?	<numeric_value> <reg_name>	[query only] 1992
:WRITe	(<numeric_value> <reg_name>), <data>	[no query] 1992
:RESet?		[query only] 1992
:VERBose?		[query only] 1992
:SElect	<numeric_value>	1992
:WSProtocol		1992
:COMMand		1992
[:ANY]	<numeric_value>	[no query] 1992
:AHILine	<numeric_value>,<numeric_value>	[no query] 1992
:AILine	<numeric_value>,<numeric_value>	[no query] 1992

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES	
:AMControl	<numeric_value>	[no query]	1992
:ANO		[no query]	1992
:BAVailable	<Boolean>,<numeric_value>	[no query]	1992
:BNO	<Boolean>	[no query]	1992
:BRQ		[no query]	1992
:CEVent	<Boolean>,<numeric_value>	[no query]	1992
:CLR		[no query]	1992
:CLOCK		[no query]	1992
:CRESPonse	<numeric_value>	[no query]	1992
:ENO		[no query]	1992
:GDEVice	<numeric_value>	[no query]	1992
:ICOMmander	<numeric_value>	[no query]	1992
:RDEVice	<numeric_value>	[no query]	1992
:RHANDlers		[no query]	1992
:RHLine	<numeric_value>	[no query]	1992
:RILine	<numeric_value>	[no query]	1992
:RINTerrupter		[no query]	1992
:RMOdId		[no query]	1992
:RPERror		[no query]	1992
:RPRotocol		[no query]	1992
:RSTB		[no query]	1992
:RSARea		[no query]	1992
:SLModId	<Boolean>,<numeric_value>	[no query]	1992
:SLOCK		[no query]	1992
:SUModId	<Boolean>,<numeric_value>	[no query]	1992
:TRIGger		[no query]	1992
:MESSage			1992
:RECEive?	<numeric_value> <terminator>	[query only]	1992
:SEND	<message_string> [,(END NEND)]	[no query]	1992
:QUERY			1992
[:ANY?]	<numeric_value>	[query only]	1992
:AHILine?	<numeric_value>,<numeric_value>	[query only]	1992
:AILine?	<numeric_value>,<numeric_value>	[query only]	1992
:AMControl?	<numeric_value>	[query only]	1992
:ANO?		[query only]	1992
:BNO?	<Boolean>	[query only]	1992
:BRQ?		[query only]	1992
:CEVent?	<Boolean>,<numeric_value>	[query only]	1992
:CRESPonse?	<numeric_value>	[query only]	1992
:ENO?		[query only]	1992
:RDEVice?	<numeric_value>	[query only]	1992
:RHANDlers?		[query only]	1992
:RHLine?	<numeric_value>	[query only]	1992
:RILine?	<numeric_value>	[query only]	1992

1999 SCPI Command Reference

KEYWORD	PARAMETER FORM	NOTES
:RINTerrupter?		[query only] 1992
:RMODid?		[query only] 1992
:RPERror?		[query only] 1992
:RPRotocol?		[query only] 1992
:RSTB?		[query only] 1992
:RSARea?		[query only] 1992
:SLModid?	<Boolean>,<numeric_value>	[query only] 1992
:SUModid?	<Boolean>,<numeric_value>	[query only] 1992
:RESponse?		[query only] 1992

26.1 :CONFigure

VXI:CONFigure

Provides the necessary commands to query the configuration of a VXI system.

26.1.1 :DNUMber?

VXI:CONFigure:DNUMber?

This query returns the number of devices in the system, in <NR1 numeric value> format. The range of this integer numeric value is 1 to 256 inclusive.

26.1.2 :HIERarchy?

VXI:CONFigure:HIERarchy?

This command returns current hierarchy configuration information about the selected logical address. The individual fields of the response are comma separated. If the information about the selected logical address is not available from the destination device, (that is, the requested device is not in the servant area of the destination device), then error -224, "Parameter error" will be set and no response data will be sent. This is a required configuration command. The output fields are defined as follows:

- Logical address: a <NR1>, between -1 and 255 inclusive. A -1 indicates that the device has no logical address.
- commander's logical address: a <NR1>, between -1 and 255 inclusive. -1 indicates that this device has no commander or the commander is unknown.
- Interrupt Handlers: a comma separated list of 7 <NR1>, between 0 and 7 inclusive. Interrupt lines 1-7 are mapped to the individual return values. 0 is used to indicate that the particular interrupt handler is not configured. A set of return values of 0,0,0,5,2,0,6 would indicate that the device is configured as follows:
 - handler 4 is configured to handle interrupts on line 5
 - handler 5 is configured to handle interrupts on line 2
 - handler 7 is configured to handle interrupts on line 6
 - handlers 1,2,3,6 are not configured
- Interrupters: a comma separated list of 7 <NR1>, between 0 and 7 inclusive. Interrupt lines 1-7 are mapped to the individual return values. 0 is used to indicate that the particular interrupter is not configured. A set of return values of 0,0,0,5,2,0,6 would indicate that the device is configured as follows:

1999 SCPI Command Reference

interrupter 4 is configured to interrupt on line 5
interrupter 5 is configured to interrupt on line 2
interrupter 7 is configured to interrupt on line 6
interrupters 1,2,3,6 are not configured

- Pass/Failed: a <NR1> which contains the pass/fail status of the specified device encoded as follows:
FAIL=0 IFAIL=1 PASS=2 READY=3
- manufacturer specific comment: This up to 80 character quoted string contains manufacturer specific data. It is sent with a 488.2 string response data format.

26.1.2.1 :ALL?

VXI:CONFigure:HIERarchy?:ALL?

When issued to a resource manager, configuration information about all logical addresses. If the command is received by a device which is not the resource manager, it returns the current hierarchy configuration information about the destination device followed by all of its immediate servants. The information is returned in the order specified in VXI:CONFigure:LADdress?. (the information about multiple logical addresses will be semi-colon separated and follow the IEEE 488.2 response message format). The individual fields of the output are comma separated. This is an optional configuration command.

26.1.2.2 :VERBose?

VXI:CONFigure:HIERarchy?:VERBose?

This command has the same functionality as VXI:CONFigure:HIERarchy?. However it returns a quoted string indicating the device information for the selected This command is intended to allow for a human readable form of the response to VXI:CONFigure:HIERarchy?. The format of this string is manufacturer specific. This is an optional configuration command.

26.1.2.2.1 :ALL?

VXI:CONFigure:HIERarchy?:VERBose?:ALL?

This command has the same functionality as VXI:CONFigure:HIERarchy:ALL?. However it returns a semi- colon separated sequence of strings indicating the device information for all devices. The format of this string is manufacturer specific. This command is intended to allow for a human readable form of the response to VXI:CONFigure:HIERarchy:ALL?. This is an optional configuration command.

26.1.3 :INFormation?

VXI:CONFigure:INFormation?

This command returns the static information about the selected logical address. The individual fields of the response are comma separated. If the information about the selected logical address is not available from the destination device, (that is, the requested device is not in the servant area of the destination device), then error -224, "Parameter error" will be set and no response data will be sent. This is a required configuration command. The command returns the following values:

- logical address: a <NR1> between -1 and 255 inclusive. A -1 indicates that the

1999 SCPI Command Reference

device has no logical address.

- manufacturer id: a <NR1>, between -1 and 4095 inclusive. -1 indicates that the device has no manufacturer id.
- model code: a <NR1>, between -1 and 65535 inclusive. -1 indicates that the device has no model code
- device class: a <NR1>, between 0 and 5 inclusive. Each value corresponds the the device's classification as follows:
 - 0: VXIbus memory device.
 - 1: VXIbus extended device.
 - 2: VXIbus message based device.
 - 3: VXIbus register based device.
 - 4: Hybrid device.
 - 5: Non-VXIbus device.
- address space: a <NR1>, between 0 and 15 inclusive, which is the sum of the binary weighted codes of the address space(s) occupied by the device. The following values are defined:
 - 1: The device has A16 registers.
 - 2: The device has A24 registers.
 - 4: The device has A32 registers.
 - 8: The device has A64 registers.
- A16 memory offset: the base address for any A16 registers (other than the VXIbus defined registers) which are present on the device. This is a <NR1> in the range of -1 to 65535. -1 is used to indicate that the device has no A16 memory.
- A24 memory offset: the base address for any A24 registers which are present on the device. This is a <NR1> in the range of -1 to 16777215. -1 is used to indicate that the device has no A24 memory.
- A32 memory offset: the base address for any A32 registers which are present on the device. This is a <NR1> value> in the range of -1 to 4294967295. -1 is used to indicate that the device has no A32 memory.
- A16 memory size: the number of bytes reserved for any A16 registers (other than the VXIbus defined registers) which are present on the device. This is a <NR1> in the range of -1 to 65535. -1 is used to indicate that the device has no A16 memory.
- A24 memory size: the number of bytes reserved for any A24 registers which are present on the device. This is a <NR1> in the range of -1 to 16777215. -1 is used to indicate that the device has no A24 memory.
- A32 memory size: the number of bytes reserved for any A32 registers which are present on the device. This is a <NR1> in the range of -1 to 4294967295. -1 is used to indicate that the device has no A32 memory.
- Slot number: a <NR1>, between -1 and the number of slots which exist in the

cage. -1 indicates that the slot which contains this device is unknown.

- Slot 0's logical address: a <NR1>, between -1 and 255 inclusive. -1 indicates that Slot 0 device associated with this device is unknown.
- subclass: a <NR1> representing the contents of the subclass register. -1 indicates that the subclass register is not defined for this device.
- attribute: a <NR1> representing the contents of the attribute register. -1 indicates that the attribute register is not defined for this device.
- manufacturer specific comment: This up to 80 character quoted string contains manufacturer specific data. It is sent with a 488.2 string response data format.

26.1.3.1 :ALL?

VXI:CONFigure:INformation?:ALL?

When issued to a resource manager, this command returns the static information about all logical addresses. If the command is received by a device which is not the resource manager, it returns the static information about the destination device followed by all of its immediate servants. The information is returned in the order specified in the response to VXI:CONFigure:LADdress?. (The information about multiple logical addresses will be semi-colon separated and follow the IEEE 488.2 response message format). The individual fields of the output are comma separated. This is an optional configuration command.

26.1.3.2 :VERBose?

VXI:CONFigure:INformation?:VERBose?

This command has the same functionality as VXI:CONFigure:INformation?. However it returns a quoted string indicating the device information for the selected device. The format of this string is manufacturer specific. This command is intended to allow for a human readable form of the response to VXI:CONFigure:INformation?. This is an optional configuration command.

26.1.3.3 :ALL?

VXI:CONFigure:INformation?:ALL?

This command has the same functionality as VXI:CONFigure:INformation:ALL?. However it returns a semi-colon separated sequence of quoted strings indicating the device information for all devices. The format of each string is manufacturer specific. This command is intended to allow for a human readable form of the response to VXI:CONFigure:INformation:ALL?. The format of this string is manufacturer specific. This is an optional configuration command.

26.1.4 :LADdress?

VXI:CONFigure:LADdress?

When issued to a resource manager, the Logical ADDRESS command returns a comma separated list of <NR1> which are the logical addresses of the devices in the system. Each <NR1> will be an integer in the range of 0 to 255 inclusive. The logical address of the device which is responding to the command will be the first item in the list. If the command is received by a device other than the resource manager, then the response to the command will contain the logical address of the destination device followed by a list of the devices

which are immediate servants to the destination device. This is a required configuration command.

26.1.5 **:NUMBer?**

VXI:CONFigure:NUMBer?

When issued to a resource manager, this command returns a <NR1> which is the number of devices in the system. If the command is received by a device which is not the resource manager, then the response to the command will contain the number of devices which are immediate servants to the destination device including the destination device. For example, a commander with 3 servants would return a value of 4, or a resource manager for a system of 4 devices would return a value of 5. The range of this <NR1> is 1 to 256 inclusive. This is a required configuration command.

26.2 **REGister**

VXI:REGister

The node collects together the commands that interact with the VXI registers.

26.2.1 **:READ? <register>**

VXI:REGister:READ?

This command returns the contents of the specified 16 bit register at the selected logical address as a <NR1>. This is a required register access command. The register is specified as the byte address of the desired register or as optionally the register name. It has values of all even numbers from 0 to 62 inclusive (as a <numeric_value>) or the following (optional) words:

A24Low: A24 Pointer Low register (18)

A24High: A24 Pointer High register (16)

A32Low: A32 Pointer Low register (22)

A32High: A32 Pointer High register (20)

ATTRibute: Attribute register (8)

DHIGH: Data High register (12)

DLOW: Data Low register (14)

DTYPE: Device Type register (2)

ICONTrol: Interrupt Control register (28)

ID: ID register (0)

ISTatus: Interrupt Status register (26)

MODid: MODID register (8)

OFFSet: Offset register (6)

PROTocol: PROTocol register (8)

RESPonse: RESPonse register (10)

SNLow: Serial Number Low register (12)

SNHigh: Serial Number High register (10)

STATus: Status register (4)

SUBClass: Subclass register (30)

VNUMber: Version Number register (14)

26.2.1.1 :VERBoSe? <register>

VXI:REGister:READ?:VERBoSe?

This command has the same functionality as VXI:REGister:READ?. However it returns a quoted string indicating the register contents for the selected device. The format of this string is manufacturer specific. This command is intended to allow for a human readable form of the response to VXI:REG:READ? which could include additional decoding of bits in specific registers. This is an optional register access command.

26.2.2 :WRITe (<numeric_value> | <register>), <data>

VXI:REGister:WRITe

This command writes data to the specified register on the selected logical address. This is a required register access command. The Data is a 16 bit value specified as a <numeric_value> in the range of -32768 to 32767. The register is specified as the byte address of the desired register or as optionally the register name. It has values of all even numbers from 0 to 62 inclusive (expressed as a <numeric_value>) or the following (optional) words:

CONTRol: Control register (4)

DEXTended: Data Extended register (10)

DHIGH: Data High register (12)

DLOW: Data Low register (14)

ICONTRol: Interrupt Control register (28)

MODId: MODID register (8)

LADDRESS: Logical Address register (0)

OFFSet: Offset register (6)

SIGNal: Signal register (8)

26.3 :RESet?

VXI:RESet?

This command resets the selected logical address. SYSFAIL generation is inhibited while the device is in the self test state. The command waits for five seconds or until the selected device has indicated passed (whichever occurs first). If the device passes its self test, then SYSFAIL generation will be re-enabled. If the device fails its self test, then SYSFAIL generation will remain inhibited. The return value from this command is the state of the selected device after it has been reset. The command returns an <NR1> which is encoded as

1999 SCPI Command Reference

follows: FAIL=0, PASS=2, READY=3. The state of the A24/A32 enable bit is not altered by this command. This is a required register access command.

26.3.1 :VERBose?

VXI:RESet?:VERBose?

This command has the same functionality as VXI:RESet?. However it returns a quoted string indicating the state of the specified device. The format of this string is manufacturer specific. This is an optional register access command.

26.4 :SElect <logical_address>

VXI:SElect

This command specifies the logical address which is to be used by all subsequent commands in the VXI subsystem. Logical_address is specified as a <numeric_value> in the range of 0 to 255. The query form returns the logical address as a <NR1> the value of -1 is reserved to indicate that no logical address has been selected. The *RST default value for logical_address is that no logical address is selected. All other commands which require a logical_address to be selected will respond with an error -221, "Settings conflict" if no logical address is selected.

26.5 :WSPRotocol

VXI:WSPRotocol

This node collects together all the commands associated with the word serial protocol.

26.5.1 :COMMand

26.5.1.1 [:ANY] <data>

VXI:WSPRotocol:COMMand:ANY

This command sends the specified word serial command to the selected logical address. The data field is a <numeric_value> or a non-decimal value which specifies the command to be sent. The response of a word serial query which is sent with this command can be read with the VXI:WSPRotocol:RESPonse? command. This is a required Word Serial Support command.

26.5.1.2 :AHLine <hand_id>,<line_number>

VXI:WSPRotocol:COMMand:AHLine

This command sends an Assign Handler Line command to the selected logical address. The hand_id field has a value in the range of 1 to 7 inclusive, and line_number field has a value in the range of 0 to 7 inclusive. They are both specified with <numeric_value>s or as non-decimal values. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.3 :ALine <int_id>,<line_number>

VXI:WSPRotocol:COMMand:ALine

This command sends an Assign Interrupter Line command to the selected logical address. The int_id field has a value in the range of 1 to 7 inclusive, and line_number field has a value in the range of 0 to 7 inclusive. They are both specified with numeric_values or as non-decimal values. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.4 :AMControl <response_mask>

VXI:WSPRotocol:COMMand:AMControl

This command sends an Asynchronous Mode Control command to the selected logical address. The response mask field has a value in the range of 0 to 15 inclusive, and is defined in VXI rev. 1.3. It is specified with a <numeric_value> or as a non-decimal value. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.5 :ANO

VXI:WSPRotocol:COMMand:ANO

This command sends an Abort Normal Operation command to the selected logical address. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.6 :BAVailable <Boolean>,<byte>

VXI:WSPRotocol:COMMand:BAVailable

This command sends a Byte Available command to the selected logical address. The Boolean field selects whether the END bit is set in the command. The byte field has a value in the range of 0 to 255 inclusive. It is specified with a <numeric_value> or as a non-decimal value. This is an optional Word Serial Support command.

26.5.1.7 :BNO <Boolean>

VXI:WSPRotocol:COMMand:BNO

This command sends an Begin Normal Operation command to the selected logical address. The Boolean field selects whether the Top_level bit is set in the command. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.8 :BRQ

VXI:WSPRotocol:COMMand:BRQ

This command sends an Byte Request command to the selected logical address. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.9 :CEVent <Boolean>,<event_number>

VXI:WSPRotocol:COMMand:CEVent

This command sends a Control Event command to the selected logical address. The Boolean field selects whether the Enable bit is set in the command. The event_number field has a value in the range of 0 to 127 inclusive. It is specified with a <numeric_value> or as a non-decimal value. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.10 :CLR

VXI:WSPRotocol:COMMand:CLR

This command sends an Clear command to the selected logical address. This is an optional Word Serial Support command.

26.5.1.11 :CLOCK

VXI:WSPRotocol:COMMand:CLOCK

This command sends an Clear Lock command to the selected logical address. This is an optional Word Serial Support command.

26.5.1.12 :CRESPonse <response_mask>

VXI:WSPRotocol:COMMand:CRESPonse

This command sends a Control Response command to the selected logical address. The response_mask field has a value in the range of 0 to 127 inclusive, and is defined in VXI rev. 1.3. It is specified with a <numeric_value> or as a non-decimal value. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.13 :ENO

VXI:WSPRotocol:COMMand:ENO

This command sends an End Normal Operation command to the selected logical address. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.14 :GDEvice <logical_address>

VXI:WSPRotocol:COMMand:GDEvice

This command sends a Grant Device command to the selected logical address. The logical_address field has a value in the range of 0 to 255 inclusive. It is specified with a <numeric_value> or as a non-decimal value. This is an optional Word Serial Support command.

26.5.1.15 :ICOMmander <logical_address>

VXI:WSPRotocol:COMMand:ICOMmander

This command sends a Identify Commander command to the selected logical address. The logical_address field has a value in the range of 0 to 255 inclusive. It is specified with a <numeric_value> or as a non-decimal value. This is an optional Word Serial Support command.

26.5.1.16 :RDEvice <logical_address>

VXI:WSPRotocol:COMMand:RDEvice

This command sends a Release Device command to the selected logical address. The logical_address field has a value in the range of 0 to 255 inclusive. It is specified with a <numeric_value> or as a non-decimal value. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.17 :RHANDlers

VXI:WSPRotocol:COMMand:RHANDlers

This command sends a Read Handlers command to the selected logical address. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.18 **:RHLine <hand_id>**

VXI:WSPRotocol:COMMand:RHLine

This command sends a Read Handler Line command to the selected logical address. The `hand_id` field has a value in the range of 1 to 7 inclusive. It is specified with a `<numeric_value>` or as a non-decimal value. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.19 **:RILine <int_id>**

VXI:WSPRotocol:COMMand:RILine

This command sends a Read Interrupter Line command to the selected logical address. The `int_id` field has a value in the range of 1 to 7 inclusive. It is specified with a `<numeric_value>` or as a non-decimal value. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.20 **:RINTerrupter**

VXI:WSPRotocol:COMMand:RINTerrupter

This command sends a Read Interrupters command to the selected logical address. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.21 **:RMODid**

VXI:WSPRotocol:COMMand:RMODid

This command sends a Read MODid command to the selected logical address. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.22 **:RPERror**

VXI:WSPRotocol:COMMand:RPERror

This command sends a Read Protocol Error command to the selected logical address. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.23 **:RPRotocol**

VXI:WSPRotocol:COMMand:RPRotocol

This command sends a Read Protocol command to the selected logical address. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.24 **:RSTB**

VXI:WSPRotocol:COMMand:RSTB

This command sends a Read Status Byte command to the selected logical address. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.25 :RSARea

VXI:WSPRotocol:COMMand:RSARea

This command sends a Read Servant Area command to the selected logical address. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.26 :SLModid <Boolean>,<MODID 6-0>

VXI:WSPRotocol:COMMand:SLModid

This command sends a Set Lower MODId command to the selected logical address. The Boolean field selects whether the Enable bit is set in the command. The MODID 6-0 field has a value in the range of 0 to 127 inclusive. It is specified with a <NR1> or as a non-decimal value. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.27 :SLOCK

VXI:WSPRotocol:COMMand:SLOCK

This command sends a Set Lock command to the selected logical address. This is an optional Word Serial Support command.

26.5.1.28 :SUModid <Boolean>,<MODID 12-7>

VXI:WSPRotocol:COMMand:SUModid

This command sends a Set Upper MODId command to the selected logical address. The Boolean field selects whether the Enable bit is set in the command. The MODID 12-7 field has a value in the range of 0 to 63 inclusive. It is specified with a <NR1> or as a non-decimal value. The response to this command can be read with the VXI:WSPRotocol:RESPonse? command. This is an optional Word Serial Support command.

26.5.1.29 :TRIGger

VXI:WSPRotocol:COMMand:TRIGger

This command sends a Trigger command to the selected logical address. This is an optional Word Serial Support command.

26.5.2 :MESSage

VXI:WSPRotocol:MESSage

This node collects together all the messages (as defined by VXI) of the word serial protocol.

26.5.2.1 :RECeive? <count>|<terminator>

VXI:WSPRotocol:MESSage:RECeive?

This command receives a message from the selected logical address using both the word serial protocol and the byte transfer protocol. The command will always terminate on the End bit being set. Additional termination options are on a specified number of bytes (count), or on a match to a particular terminator, (that is, LF, CRLF, END). The response is returned as a string. This is a required Word Serial Support command.

26.5.2.2 :SEND <message_string> [,(ENDINEND)]

VXI:WSPRotocol:MESSage:SEND

This command sends the specified message string to the selected logical address. The string is sent using the word serial protocol with the byte transfer protocol. The last byte of the

string is sent with the End bit set unless NEND is specified. This is a required Word Serial Support command.

26.5.3 :QUERY

VXI:WSPProtocol:QUERY

This node collects together all the queries (as defined by VXI) of the word serial protocol.

26.5.3.1 [:ANY]? <data>

VXI:WSPProtocol:QUERY:ANY?

This command sends the specified word serial query to the selected logical address. The data field is a <numeric_value> or a non-decimal value which specifies the query to be sent. The returned value is the response to the word serial query and is a <NR1>. This command obeys the byte transfer protocol. This is a required Word Serial Support command.

26.5.3.2 :AHLine? <hand_id>,<line_number>

VXI:WSPProtocol:QUERY:AHLine?

This command sends an Assign Handler Line command to the selected logical address. The hand_id field has a value in the range of 1 to 7 inclusive, and line_number field has a value in the range of 0 to 7 inclusive. They are both specified with <numeric_value>s or as non-decimal values. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.3 :ALLine? <int_id>,<line_number>

VXI:WSPProtocol:QUERY:ALLine?

This command sends an Assign Interrupter Line command to the selected logical address. The int_id field has a value in the range of 1 to 7 inclusive, and line_number field has a value in the range of 0 to 7 inclusive. They are both specified with <numeric_value>s or as non-decimal values. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.4 :AMControl? <response_mask>

VXI:WSPProtocol:QUERY:AMControl?

This command sends an Asynchronous Mode Control command to the selected logical address. The response_mask field has a value in the range of 0 to 15 inclusive, and is defined in VXI rev. 1.3. It is specified with a <numeric_value> or as a non-decimal value. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.5 :ANO?

VXI:WSPProtocol:QUERY:ANO?

This command sends an Abort Normal Operation command to the selected logical address. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.6 :BNO? <Boolean>

VXI:WSPProtocol:QUERY:BNO?

This command sends an Begin Normal Operation command to the selected logical address. The Boolean field selects whether the Top_level bit is set in the command. The returned

value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.7 :BRQ?

VXI:WSPProtocol:QUERY:BRQ?

This command sends an Byte Request command to the selected logical address. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.8 :CEVent? <Boolean>,<event_number>

VXI:WSPProtocol:QUERY:CEVent?

This command sends a Control Event command to the selected logical address. The Boolean field selects whether the Enable bit is set in the command. The event_number field has a value in the range of 0 to 127 inclusive. It is specified with a <numeric_value> or as a non-decimal value. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.9 :CREsponse? <response_mask>

VXI:WSPProtocol:QUERY:CREsponse?

This command sends a Control Response command to the selected logical address. The response_mask field has a value in the range of 0 to 127 inclusive, and is defined in VXI rev. 1.3. It is specified with a numeric_value or as a non-decimal value. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.10 :ENO?

VXI:WSPProtocol:QUERY:ENO?

This command sends an End Normal Operation command to the selected logical address. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.11 :RDEvice? <logical_address>

VXI:WSPProtocol:QUERY:RDEvice?

This command sends a Release Device command to the selected logical address. The logical_address field has a value in the range of 0 to 255 inclusive. It is specified with a <numeric_value> or as a non-decimal value. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.12 :RHANdlers?

VXI:WSPProtocol:QUERY:RHANdlers?

This command sends a Read Handlers command to the selected logical address. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.13 :RHLine? <hand_id>

VXI:WSPProtocol:QUERY:RHLine?

This command sends a Read Handler Line command to the selected logical address. The hand_id field has a value in the range of 1 to 7 inclusive. It is specified with a

<numeric_value> or as a non-decimal value. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.14 :RILine? <int_id>

VXI:WSPProtocol:QUERY:RILine?

This command sends a Read Interrupter Line command to the selected logical address. The int_id field has a value in the range of 1 to 7 inclusive. It is specified with a <numeric_value> or as a non-decimal value. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.15 :RINTerrupter?

VXI:WSPProtocol:QUERY:RINTerrupter?

This command sends a Read Interrupters command to the selected logical address. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.16 :RMODid?

VXI:WSPProtocol:QUERY:RMODid?

This command sends a Read MODid command to the selected logical address. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.17 :RPERror?

VXI:WSPProtocol:QUERY:RPERror?

This command sends a Read Protocol Error command to the selected logical address. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.18 :RPRotocol?

VXI:WSPProtocol:QUERY:RPRotocol?

This command sends a Read Protocol command to the selected logical address. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.19 :RSTB?

VXI:WSPProtocol:QUERY:RSTB?

This command sends a Read Status Byte command to the selected logical address. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.20 :RSARea?

VXI:WSPProtocol:QUERY:RSARea?

This command sends a Read Servant Area command to the selected logical address. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.21 **:SLModid? <Boolean>,<MODID 6-0>**

VXI:WSPProtocol:QUERy:SLModid?

This command sends a Set Lower MODid command to the selected logical address. The Boolean field selects whether the Enable bit is set in the command. The MODID 6-0 field has a value in the range of 0 to 127 inclusive. It is specified with a <numeric_value> or as a non-decimal value. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.3.22 **:SUModid? <Boolean>,<MODID 12-7>**

VXI:WSPProtocol:QUERy:SUModid?

This command sends a Set Upper MODid command to the selected logical address. The Boolean field selects whether the Enable bit is set in the command. The MODID 12-7 field has a value in the range of 0 to 63 inclusive. It is specified with a <numeric_value> or as a non-decimal value. The returned value is the response to the command and is a <NR1>. This is an optional Word Serial Support command.

26.5.4 **:RESPonse?**

VXI:WSPProtocol:RESPonse?

This command returns one word of data from the data low register on the selected logical address. This command obeys the Byte transfer protocol. The data is returned as a <NR1>. This is a required Word Serial Support command.

Index

A

- ABORt 2-5
 - HCOPy 10-4
 - TRIGger Subsystem 24-12
- :AC
 - INPut:BIAS:CURRent 11-4
 - INPut:BIAS:VOLTage 11-4
 - MEASure 3-9
 - SENSe:CURRent 18-25
 - SENSe:POWer 18-64
 - SENSe:VOLTage 18-84
- ACceleration
 - SENSe:FUNCTion 18-50
 - SOURce 19-5
- ACceleration Subsystem
 - SOURce 19-5
- :ACHannel
 - SENSe:POWer 18-63
- :ACQUIre
 - SOURce:CORRection:COLlect 19-12
- :ACQUIRE STANdard
 - SENSe:CORRection:COLlect 18-17
- :ADDRESS
 - SYSTem:COMMunicate:GPIB 21-5
 - SYSTem:COMMunicate:GPIB:RDEvice 21-5
- :ADJust
 - SOURce:PHASe 19-56
- :AFC
 - SENSe 18-38
- :AHLine
 - VXI:WSPRotocol:COMMand 26-9
- :AHLine?
 - VXI:WSPRotocol:QUERy 26-14
- :AlLine
 - VXI:WSPRotocol:COMMand 26-9
- :AlLine?
 - VXI:WSPRotocol:QUERy 26-14
- :ALC
 - SOURce:CURRent 19-18
 - SOURce:POWer 19-61
 - SOURce:VOLTage 19-94
- :ALL
 - CALCulate:MATH:EXPRession:DELeTe 4-22
 - CALibration 5-2
 - DISPlay:ANNotation 8-4
 - HCOPy:ITEM 10-8
 - MEMory:DELeTe 13-7
 - PROGram:SELeCted:DELeTe 16-2
 - ROUTe:MODUle:DELeTe 17-2
 - ROUTe:OPEN 17-3
 - ROUTe:PATH:DELeTe 17-4
 - SENSe:FUNCTion:OFF 18-44
 - SENSe:FUNCTion:ON 18-45
 - SOURce:DM:THReshold 19-29
 - TRACe|DATA:DELeTe 23-4
 - TRIGger Subsystem:INITiate:CONTInuous 24-22
 - TRIGger Subsystem:INITiate:IMMEDIATE 24-22
- :ALL AC
 - SOURce:DM:COUPling 19-29
- :ALL DC
 - SOURce:DM:COUPling 19-29
- :ALL GROund
 - SOURce:DM:COUPling 19-29
- :ALL?
 - CALibration 5-2
 - MEMory:CATalog 13-5
 - MEMory:FREE 13-8
 - VXI:CONFIgure:HIERarchy? 26-4
 - VXI:CONFIgure:HIERarchy?:VERBoSe? 26-4
 - VXI:CONFIgure:INFormation? 26-6
- :ALternate
 - SYSTem 21-3
- AM
 - SENSe:FUNCTion 18-50
 - SOURce 19-6 - 19-9
 - SOURce:LIST 19-49
- AM Subsystem
 - SENSe 18-4 - 18-5
 - SOURce 19-6 - 19-9
- :AMControl
 - VXI:WSPRotocol:COMMand 26-10
- :AMControl?
 - VXI:WSPRotocol:QUERy 26-14
- Amorphous SENSe Model
 - Instrument Model:Internal Routing 2-8
- :AMPLitude
 - DISPlay:ANNotation 8-4
 - MEASure 3-11
 - SOURce:CURRent:LEVel:IMMEDIATE 19-20
 - SOURce:CURRent:LEVel:TRIGgered 19-21
 - SOURce:CURRent:LIMit 19-21
 - SOURce:MARKer 19-54
 - SOURce:POWer:LEVel:IMMEDIATE 19-63

1999 SCPI Command Reference

SOURce:POWer:LEVel:TRIGgered 19-64	:ASCii?
SOURce:POWer:LIMit 19-64	MEMory:CATalog 13-5
SOURce:RESistance:LEVel:IMMediate 19-75	MEMory:FREE 13-8
SOURce:RESistance:LEVel:TRIGgered 19-76	:ATRigger
SOURce:RESistance:LIMit 19-76	TRIGger Subsystem:TRIGger:SEQuence 24-23
SOURce:VOLTAge:LEVel:IMMediate 19-96	:ATTenuation
SOURce:VOLTAge:LEVel:TRIGgered 19-97	OUTPut Subsystem 15-3
SOURce:VOLTAge:LIMit 19-97	:ATTenuation
:ANGLE	INPut 11-3
INPut:POSition:X 11-8	SENSe:CURRent:ACI:DC 18-26
INPut:POSition:Y 11-10	SENSe:POWer:ACI:DC 18-64
INPut:POSition:Z 11-12	SENSe:VOLTAge:ACI:DC 18-85
OUTPut:POSition:X 15-6	SOURce:CURRent 19-18
OUTPut:POSition:Y 15-8	SOURce:POWer 19-61
OUTPut:POSition:Z 15-10	SOURce:VOLTAge 19-94
SOURce:DM:LEAKage 19-28	:ATTRIBUTES
SOURce:DM:QUADrature 19-29	DISPlay:WINDow:TEXT 8-9
UNIT 25-1	:AUTO
:ANNOtation	CALCulate 4-7, 4-12, 4-14, 4-18, 4-21,
DISPlay 8-4	4-25 - 4-26, 4-28, 4-31
HCOPY:ITEM 10-8	CALibration 5-2, 5-7
:ANO	DISPlay 8-11 - 8-12, 8-14 - 8-15
VXI:WSPRotocol:COMManD 26-10	HCOPY 10-13
:ANO?	INPut:ATTenuation 11-3
VXI:WSPRotocol:QUERy 26-14	INPut:GAIN 11-6
:ANY	OUTPut:FILTer 15-3
VXI:WSPRotocol:COMManD 26-9	SENSe 18-4, 18-7, 18-10 - 18-11, 18-20, 18-26 -
:ANY?	18-28, 18-31, 18-36, 18-38 - 18-40, 18-59 - 18-62,
VXI:WSPRotocol:QUERy 26-14	18-64 - 18-67, 18-69 - 18-70, 18-72, 18-79, 18-82,
:AOFF	18-85 - 18-87
SOURce:MARKer 19-54	SENSe:ROSCillator:SOURce INTernal 19-80
:APERture	SOURce 19-18 - 19-20, 19-23, 19-43, 19-45,
CALCulate:GDAPerture 4-17	19-50, 19-52, 19-61 - 19-62, 19-66, 19-73, 19-79,
CALCulate:SMOothing 4-23	19-85 - 19-86, 19-94 - 19-96, 19-99
SENSe:CURRent:ACI:DC 18-25	SYSTem 21-9, 21-33
SENSe:FREQuency 18-37	TRACe DATA 23-5
SENSe:POWer:ACI:DC 18-64	TRIGger Subsystem:ARM:SEQuence
SENSe:RESistance 18-68	:LAYer:DELay 24-14
SENSe:SMOothing 18-73	TRIGger Subsystem:ARM:SEQuence
SENSe:VOLTAge:ACI:DC 18-84	:LAYer:LEVel 24-16
:APOWer	TRIGger Subsystem:TRIGger:SEQuence
CONTRol 6-1	:DELay 24-26
SOURce:LIST:CONTRol 19-50	TRIGger Subsystem:TRIGger:SEQuence
:APRobe	:LEVel 24-28
SOURce:LIST 19-49	:AVERage Subsystem
ARM 2-5	CALCulate 4-6 - 4-8
TRIGger Subsystem 24-12 - 24-20	SENSe 18-6 - 18-9
:ARM-TRIGger Model	:AWEighting
TRIGger Subsystem 24-1	INPut:FILTer 11-5
:ARRAY	SENSe:FILTer 18-35
MEASure 3-7	:AXIS
ASCii 9-1, 9-3	DISPlay:WINDow:TRACe:GRATicule 8-11

1999 SCPI Command Reference

B

:BACKground
 DISPlay:WINDow 8-6
 :BANDwidth
 SOURce:POWer:ALC 19-62
 :BANDwidth
 SENSe:DETEctor 18-30
 SOURce:CURREnt:ALC 19-19
 SOURce:VOLTage:ALC 19-95
 BANDwidth Subsystem
 SENSe 18-10 - 18-11
 :BAUD
 SYSTem:COMM:SERial:RECeive 21-7
 SYSTem:COMM:SERial:TRANsmit 21-9
 :BAVailable
 VXI:WSPRotocol:COMMand 26-10
 :BEEPer
 SYSTem 21-3
 :BIAS
 INPut 11-3
 SENSe:MIXer 18-60
 BINary 9-2 - 9-3
 :BINary?
 MEMory:CATalog 13-5
 MEMory:FREE 13-8
 :BINertia
 CALibration 5-2 - 5-3
 :BIT
 STATus 20-3
 :BITS
 SYSTem:COMM:SERial:RECeive 21-7
 SYSTem:COMM:SERial:TRANsmit 21-9
 :BLOWer
 CONTRol 6-2
 SOURce:LIST:CONTRol 19-50
 :BNO
 VXI:WSPRotocol:COMMand 26-10
 :BNO?
 VXI:WSPRotocol:QUERy 26-14
 :BORDER
 FORMat 9-1
 :BOTTom
 DISPlay:WINDow:TRACe:Y:SCALe 8-14
 :BRAKe
 CONTRol 6-2
 :BRIghtness
 DISPlay 8-4
 :BRQ
 VXI:WSPRotocol:COMMand 26-10
 :BRQ?
 VXI:WSPRotocol:QUERy 26-15

:BURSt
 :MEASure:FREQuency 3-9
 BWIDth
 SENSe:DETEctor 18-30
 SOURce:CURREnt:ALC 19-19
 SOURce:POWer:ALC 19-62
 SOURce:VOLTage:ALC 19-95
 :BWIDth Subsystem
 SENSe 18-10 - 18-11

C

CALCulate
 Instrument Model:Measurement Function 2-4
 Instrument Model:Signal Generation 2-5
 CALCulate Subsystem 4-1 - 4-32
 CALibration Subsystem 5-1 - 5-10
 :CAPability?
 SYSTem 21-4
 :CARM
 SENSe:FILTer 18-35
 :CATalog
 CALCulate:MATH:EXPRession 4-21
 MEMory 13-4
 SYSTem:KEY 21-32
 :CATalog?
 MMEMory 14-2
 PROGram 16-2
 ROUTe:MODule 17-2
 ROUTe:PATH 17-3
 TRACe|DATA 23-2
 :CCIR
 SENSe:FILTer 18-35
 :CCITt
 SENSe:FILTer 18-34
 :CDFunction
 SENSe:FUNCTion:VOLTage 18-56
 :CDIRectory
 MMEMory 14-3
 :CENTer
 CALCulate:FILTer:GATE:FREQuency 4-14
 CALCulate:FILTer:GATE:TIME 4-12
 CALCulate:TRANSform:DISTance 4-28
 CALCulate:TRANSform:FREQuency 4-30
 CALCulate:TRANSform:TIME 4-26
 DISPlay:WINDow:TRACe:X:SCALe 8-12
 SENSe:FREQuency 18-37
 SOURce:CURREnt 19-19
 SOURce:FREQuency 19-43
 SOURce:POWer 19-62
 SOURce:RESistance 19-77
 SOURce:VOLTage 19-95

1999 SCPI Command Reference

```

:CENTronics
  SYSTem:COMMunicate 21-4
:CEVent
  VXI:WSPRotocol:COMManD 26-10
:CEVent?
  VXI:WSPRotocol:QUERy 26-15
:CHECK
  SYSTem:COMM:SERial:RECeive 21-8
:CLEan
  CONTrol 6-3
:CLEar
  CALCulate:AVERage 4-6
  CALCulate:LIMit 4-21
  DISPlay:WINDow:GRAPhics 8-7
  DISPlay:WINDow:TEXT 8-9
  MEMory 13-6
  OUTPut:PROTection 15-12
  SENSE:CURRent:AC:DC:PROTection 18-26
  SENSE:POWEr:AC:DC:PROTection 18-65
  SENSE:VOLTage:AC:DC:PROTection 18-85
  SOURce:CURRent:PROTection 19-23
  SOURce:POWEr:PROTection 19-66
  SOURce:RESistance:PROTection 19-77
  SOURce:VOLTage:PROTection 19-99
:CLIMits
  CALCulate 4-9
:CLOCK
  SOURce:DM 19-30
  SOURce:DM:THREshold 19-29
  VXI:WSPRotocol:COMManD 26-11
:CLOCK AC
  SOURce:DM:COUPLing 19-29
:CLOCK DC
  SOURce:DM:COUPLing 19-29
:CLOCK GROUND
  SOURce:DM:COUPLing 19-29
Cloned Models
  Instrument Model:Internal Routing 2-7
:CLOSE
  MMEMory 14-3
  ROUTe 17-1
:CLR
  VXI:WSPRotocol:COMManD 26-10
:CMAP
  DISPlay 8-4
  HCOPy:DEVice 10-5
:CMESsage
  SENSE:FILTer 18-34
:COAX
  SENSE:CORRection:RVELOCITY 18-22
  SOURce:CORRection:RVELOCITY 19-15
:COHherence
  SENSE:FUNCTion:POWEr 18-54
:COLlect
  SENSE:CORRection 18-17
  SOURce:CORRection 19-12
:COLor
  DISPlay:CMAP 8-4
  DISPlay:WINDow:BACKground 8-6
  DISPlay:WINDow:GRAPhics 8-7
  DISPlay:WINDow:TEXT 8-9
  DISPlay:WINDow:TRACe 8-10
  HCOPy:DEVice 10-6
  HCOPy:DEVice:CMAP 10-5
  HCOPy:ITEM:ANNotation 10-8
  HCOPy:ITEM:LABel 10-9
  HCOPy:ITEM:MENU 10-10
  HCOPy:ITEM:TDSTamp 10-11
  HCOPy:ITEM:WINDow:TEXT 10-12
  HCOPy:ITEM:WINDow:TRACe 10-12
  HCOPy:ITEM:WINDow:TRACe:GRATicule 10-12

COMBine Subsystem
  SOURce 19-10
:COMManD
  VXI:WSPRotocol 26-9
:Command Error
  SYSTem:ERRor? 21-15
:COMMunicate
  SYSTem 21-4 - 21-10
:COMPressor
  CONTrol 6-2
  SOURce:LIST:CONTrol 19-50
CONCetration Subsystem
  SENSE 18-12 - 18-14
:CONCurent
  SENSE:FUNCTion 18-43
  SOURce:LIST 19-49
:CONDition
  MEMory:TABLE 13-10
  SENSE:FUNCTion 18-51
CONdition Subsystem
  SENSE 18-15
:CONDition?
  STATus 20-3, 20-7
CONFigure
  Measurement Instructions 3-2
  VXI 26-3
:CONTinuous
  TRIGger Subsystem:INITiate 24-21
:CONTrast
  DISPlay 8-5
:CONTrol
  CALCulate:LIMit 4-19

```

1999 SCPI Command Reference

SOURce:LIST 19-50	SOURce:AM:EXtErnal 19-7
SYSTem:COMM:SErIal 21-6	SOURce:FM 19-32
:CONTRol , ALWays	SOURce:FM:EXtErnal 19-33
TRACe DATA:FEED 23-4	SOURce:PM 19-58
:CONTRol NEVer	SOURce:PM:EXtErnal 19-59
TRACe DATA:FEED 23-4	TRIGger Subsystem:ARM
:CONTRol NEXT	:SEquence:LAYer 24-14
TRACe DATA:FEED 23-4	TRIGger Subsystem:TRIGger:SEquence 24-24
:CONTRol OCONdition	:COUPLing GROund
TRACe DATA:FEED 23-4	INPut 11-4
CONTRol Subsystem 6-1 - 6-6	SOURce:AM 19-6
:COPY	SOURce:AM:EXtErnal 19-7
MEMory 13-6	SOURce:FM 19-32
MMEMemory 14-3	SOURce:FM:EXtErnal 19-33
TRACe DATA 23-2	SOURce:PM 19-58
CORRection Subsystem	SOURce:PM:EXtErnal 19-59
SENSe 18-16 - 18-24	:COVer
SOURce 19-11 - 19-16	CONTRol 6-2
:COUNT	:CPOint
CALCulate:AVERage 4-7	DISPlay:WINDow:TRACe:R:SCALE 8-15
CALCulate:TRANsform:HISTogram 4-24	:CPON
SENSe:AVERage 18-7	SYSTem 21-11
SENSe:LIST 18-58	:CREference
SENSe:SWEEP 18-78	CALCulate:FORMat:UPHase 4-17
SOURce:LIST 19-50	:CREsponse
SOURce:PULSe 19-73	VXI:WSPRotocol:COMMand 26-11
SOURce:SWEEP 19-87	:CREsponse?
TRIGger Subsystem:ARM:SEquence	VXI:WSPRotocol:QUERy 26-15
:LAYer 24-14	:CROSS
TRIGger Subsystem:TRIGger:SEquence 24-24	SENSe:FUNCTion:POWer 18-54
:COUNT?	:CSET
SENSe:FUNCTion:OFF 18-44	SENSe:CORRection 18-18
SENSe:FUNCTion:ON 18-45	SOURce:CORRection 19-12
:COUple	:CSIZe
INSTRument 12-2	DISPlay:WINDow:GRAPHics 8-7
:COUPLing	DISPlay:WINDow:TEXT 8-10
OUTPut Subsystem 15-3	:CTYPe?
SOURce:DM 19-29	SYSTem 21-11
:COUPLing AC	:CURRent
INPut 11-4	:MEASure 3-8
SOURce:AM 19-6	INPut:BIAS 11-3
SOURce:AM:EXtErnal 19-7	MEMory:TABLE 13-11
SOURce:FM 19-32	SENSe:FUNCTion 18-51
SOURce:FM:EXtErnal 19-33	SOURce 19-17 - 19-25
SOURce:PM 19-58	SOURce:LIST 19-50
SOURce:PM:EXtErnal 19-59	CURRent Subsystem
TRIGger Subsystem:ARM:SEquence	SENSe 18-25 - 18-29
:LAYer 24-14	SOURce 19-17 - 19-25
TRIGger Subsystem:TRIGger:SEquence 24-24	:CURRent, :POWer, and :VOLTag
:COUPLing DC	UNIT 25-1
INPut 11-4	:CUT
SOURce:AM 19-6	HCOPY:ITEM 10-8

1999 SCPI Command Reference

:CW

SENSe:FREQuency 18-38
SOURce:FREQuency 19-43

D

:DATA 23-1 - 23-6

CALCulate:LIMit:CONTRol 4-19
CALCulate:LIMit:LOWer 4-20
CALCulate:LIMit:UPPer 4-19
CALibration 5-4
DISPlay:WINDow:TEXT 8-10
FORMat 9-1
HCOPy:ITEM:TDSTamp 10-11
MEMory 13-7
MMEMory 14-3
SOURce:DM:THReshold 19-29
TRACe|DATA 23-2

:DATA AC

SOURce:DM:COUPLing 19-29

Data and Control Flow

Instrument Model:Internal Routing 2-6

:DATA DC

SOURce:DM:COUPLing 19-29

:DATA GROund

SOURce:DM:COUPLing 19-29

DATA Subsystem

SENSe 18-42 - 18-57

:DATA?

CALCulate 4-10
CALCulate:CLIMits:FLIMits 4-9
CALCulate:LIMit:REPORt 4-20
HCOPy 10-4
HCOPy:ITEM:ALL 10-8
HCOPy:ITEM:CUT 10-9
HCOPy:ITEM:FFEEd 10-9
HCOPy:ITEM:LABel 10-10
HCOPy:ITEM:MENU 10-10
HCOPy:ITEM:WINDow 10-11
HCOPy:ITEM:WINDow:TEXT 10-12
HCOPy:ITEM:WINDow:TRACe 10-12
HCOPy:ITEM:WINDow:TRACe
:GRATicule 10-12
HCOPy:SDUMp 10-15
SENSe:FUNCTION & DATA 18-42

:DATE

SYSTem 21-11

:DC

INPut:BIAS:CURREnt 11-4
INPut:BIAS:VOLTage 11-4
MEASure 3-9
SENSe:CURREnt 18-25

SENSe:POWer 18-64

SENSe:VOLTage 18-84

:DCYCLE

:MEASure 3-14

SOURce:PULSe 19-71

:DEFAULT

DISPlay:CMAP 8-4

HCOPy:DEvice:CMAP 10-6

PROGram:SELEcted 16-3

:DEFine

CALCulate:MATH:EXPRession 4-22

MEMory Subsystem:TABLE 13-14

PROGram:EXPLicit 16-4

PROGram:SELEcted 16-2

ROUTe:MODule 17-2

ROUTe:PATH 17-3

SYSTem:KEY 21-32

TRACe|DATA 23-3

TRIGger Subsystem:ARM:SEQUence 24-12

TRIGger Subsystem:TRIGger:SEQUence 24-24

:DELAY

OUTPut:PROTEction 15-12

SOURce:PULSe 19-72

SOURce:PULSe:DOUBle 19-72

SYSTem:COMMunicate:SERial:TRANsmit 21-9

TRIGger Subsystem:ARM

:SEQUence:LAYER 24-14

TRIGger Subsystem:TRIGger:SEQUence 24-25

:DELETE

CALCulate:MATH:EXPRession 4-22

MEMory 13-7

MMEMory 14-3

PROGram:EXPLicit 16-5

PROGram:SELEcted 16-2

ROUTe:MODule 17-2

ROUTe:PATH 17-4

SYSTem:KEY 21-32

TRACe|DATA 23-4

:DEMPHasis

SENSe:FILTer 18-34

:DEPTH

SENSe:AM 18-4

SOURce:AM 19-6

SOURce:LIST:AM 19-49

:DERivative

CALCulate 4-10

DESTination 2-3

DESTination?

HCOPy 10-4

:DETECTOR EXTERNAL

SENSe:CURREnt 18-29

SENSe:POWer 18-67

1999 SCPI Command Reference

SENSe:VOLTage 18-88	INPut:POSition:X:DISTance 11-9
:DETEctor INTERNAL	INPut:POSition:Y:ANGLE 11-10
SENSe:CURREnt 18-29	INPut:POSition:Y:DISTance 11-11
SENSe:POWEr 18-67	INPut:POSition:Z:ANGLE 11-12
SENSe:VOLTage 18-88	INPut:POSition:Z:DISTance 11-13
DETEctor Subsystem	OUTPut:POSition:X:ANGLE 15-6
SENSe 18-30 - 18-31	OUTPut:POSition:X:DISTance 15-7
:DEViation	OUTPut:POSition:Y:ANGLE 15-8
SENSe:FM 18-36	OUTPut:POSition:Y:DISTance 15-9
SENSe:PM 18-62	OUTPut:POSition:Z:ANGLE 15-10
SOURce:FM 19-32	OUTPut:POSition:Z:DISTance 15-11
SOURce:PM 19-57	SENSe:CURREnt:AC:DC 18-27
:DEVice	SENSe:LIST 18-58
HCOPY 10-5 - 10-6	SENSe:POWEr:AC:DC:RANGE:AUTO 18-66
:Device-Specific Error	SENSe:RESistance:RANGE:AUTO 18-69
SYSTEM:ERROR? 21-25	SENSe:SWEep 18-78
DIAGnostic Subsystem 7-1 - 7-2	SENSe:VOLTage:AC:DC 18-86
Digital Modulation Subsystem	SOURce:LIST 19-51
SOURce 19-26 - 19-31	SOURce:SWEep 19-86
:DIMensions	DISPlay Subsystem 8-1 - 8-16
HCOPY:PAGE 10-13	:DISTance
:DINTerchange	CALCulate:TRANSform 4-27
FORMat 9-2	INPut:POSition:X 11-9
MMEMory:LOAD and :STORE 14-4	INPut:POSition:Y 11-11
:DIRection DOWN	INPut:POSition:Z 11-13
INPut:POSition:X:ANGLE 11-8	OUTPut:POSition:X 15-7
INPut:POSition:X:DISTance 11-9	OUTPut:POSition:Y 15-9
INPut:POSition:Y:ANGLE 11-10	OUTPut:POSition:Z 15-11
INPut:POSition:Y:DISTance 11-11	SENSe:CORRection:EDELay 18-19
INPut:POSition:Z:ANGLE 11-12	SENSe:FUNCTion 18-52
INPut:POSition:Z:DISTance 11-13	SOURce:CORRection:EDELay 19-14
OUTPut:POSition:X:ANGLE 15-6	DISTance Subsystem
OUTPut:POSition:X:DISTance 15-7	SENSe 18-32
OUTPut:POSition:Y:ANGLE 15-8	:DISTortion
OUTPut:POSition:Y:DISTance 15-9	SENSe:FUNCTion:POWEr 18-50, 18-52 - 18-54
OUTPut:POSition:Z:ANGLE 15-10	DM
OUTPut:POSition:Z:DISTance 15-11	SOURce 19-26 - 19-31
SENSe:CURREnt:AC:DC 18-27	DM Subsystem
SENSe:LIST 18-58	SOURce 19-26 - 19-31
SENSe:POWEr:AC:DC:RANGE:AUTO 18-66	:DMODE PARAllel
SENSe:RESistance:RANGE:AUTO 18-69	SOURce:DM 19-30
SENSe:SWEep 18-78	:DMODE SERIAL
SENSe:VOLTage:AC:DC 18-86	SOURce:DM 19-30
SOURce:LIST 19-51	:DNUMBER?
SOURce:SWEep 19-86	VXI:CONFigure 26-3
:DIRection EITHER	:DOUBLE
SENSe:CURREnt:AC:DC 18-27	SOURce:PULSe 19-72
SENSe:POWEr:AC:DC:RANGE:AUTO 18-66	:DRAW
SENSe:RESistance:RANGE:AUTO 18-69	DISPlay:WINDow:GRAPHics 8-8
SENSe:VOLTage:AC:DC 18-86	:DTR IBFull
:DIRection UP	SYSTem:COMM:SERial:CONTRol 21-6
INPut:POSition:X:ANGLE 11-8	:DTR OFF

1999 SCPI Command Reference

SYSTem:COMM:SERial:CONTRol 21-6
 :DTR ON
 SYSTem:COMM:SERial:CONTRol 21-6
 :DTR STANDard
 SYSTem:COMM:SERial:CONTRol 21-6
 :DURation
 CONTRol 6-3
 :DWELl
 SENSe:LIST 18-58
 SENSe:SWEep 18-78
 SOURce:LIST 19-51
 SOURce:SWEep 19-85

E

:EBENch
 CONTRol 6-3
 :ECL
 TRIGger Subsystem:ARM
 :SEquence:LAYer 24-15
 TRIGger Subsystem:TRIGger:SEquence 24-26
 :ECLTrg
 OUTPut Subsystem 15-12
 :ECOUNT
 TRIGger Subsystem:ARM
 :SEquence:LAYer 24-15
 TRIGger Subsystem:TRIGger:SEquence 24-26
 :EDELay
 SENSe:CORRection 18-19
 SOURce:CORRection 19-14
 :ENABLE
 DISPlay 8-5
 STATus 20-3, 20-7
 :ENO
 VXI:WSPRotocol:COMManD 26-11
 :ENO?
 VXI:WSPRotocol:QUERy 26-15
 :Error/Event numbers
 SYSTem:ERRor? 21-14
 :The Error/Event Queue
 SYSTem:ERRor? 21-13
 :ERRor?
 SYSTem 21-11 - 21-27
 :Error/Event numbers
 SYSTem:ERRor? 21-14
 :EVENT?
 STATus 20-4, 20-7
 <event_handle>
 Instrument Model:Internal Routing 2-14
 :EXCHange
 MEMory 13-7
 :EXECute

PROGram:EXPLicit 16-5
 PROGram:SELected 16-2
 :Execution Error
 SYSTem:ERRor? 21-19
 :Expanded Capability Trigger Model
 TRIGger Subsystem 24-5 - 24-11
 :EXPLicit
 PROGram 16-4 - 16-6
 EXPonential
 CALCulate:AVERage:TCONTRol 4-7
 CALCulate:FILTer:GATE:FREQuency 4-14
 CALCulate:FILTer:GATE:TIME 4-12
 CALCulate:TRANsform:DISTance 4-29
 CALCulate:TRANsform:FREQuency 4-31
 CALCulate:TRANsform:TIME 4-27
 SENSe:WINDow:TYPE 18-89
 :EXPRession
 CALCulate:MATH 4-21
 :EXTernal
 OUTPut:FILTer 15-3
 SENSe:ROSCillator 18-71, 19-80
 SOURce:AM 19-6
 SOURce:FM 19-32
 SOURce:PM 19-59
 SOURce:PULM 19-68

F

:FAIL?
 CALCulate:CLIMits 4-9
 CALCulate:LIMit 4-20
 :FALL
 MEASure 3-13
 :FCOUNT?
 CALCulate:LIMit 4-20
 :FCUToff
 SENSe:CORRection:RVELocity 18-22
 SOURce:CORRection:RVELocity 19-15
 FEED
 CALCulate 4-10
 DISPlay:WINDow:TEXT 8-10
 DISPlay:WINDow:TRACe 8-10
 HCOPY 10-7
 Instrument Model:Internal Routing 2-9
 MMEMory 14-4
 SYSTem:COMM:SERial 21-7
 SYSTem:COMMunicate:CENTronics 21-5
 SYSTem:COMMunicate:GPIB:RDEVice 21-5
 TRACe | DATA 23-4
 FEED Exceptions
 Instrument Model:Internal Routing
 :Lamina and Cloned Models 2-8

1999 SCPI Command Reference

FEEDs from CALCulate Sub-Blocks Instrument Model:Internal Routing 2-12	:FRAMe DISPlay:WINDow:TRACe:GRATICule 8-11 SOURce:DM 19-30
FERRor SENSe:FUNCTion 18-52	:FREE MEMory 13-8
FETCH Measurement Instructions 3-3	:FREE? TRACe DATA 23-5
:FFEed HCOPy:ITEM 10-9	:FREQuency :MEASure 3-9 CALCulate:FILTer:GATE 4-13 CALCulate:TRANSform 4-29 DISPlay:ANNotation 8-4 INPut:FILTer:HPASs 11-5 INPut:FILTer:LPASs 11-5 MEMory Subsystem:TABLE 13-15 OUTPut:FILTer:HPASs 15-4 OUTPut:FILTer:LPASs 15-4 SENSe:FILTer:HPASs 18-34 SENSe:FILTer:LPASs 18-33 SENSe:FUNCTion 18-52 SENSe:LIST 18-59 SENSe:ROSCillator:EXTernal 18-71, 19-80 SENSe:ROSCillator:INTernal 18-71, 19-80 SOURce:AM:INTernal 19-7 SOURce:FM:INTernal 19-33 SOURce:LIST 19-51 SOURce:MARKer 19-54 SOURce:PM:INTernal 19-59 SOURce:PULM:INTernal 19-69 SYSTem:BEEPer 21-3 TRIGger Subsystem:ARM:SEQUence :LAYer:FILTer:HPASs 24-15 TRIGger Subsystem:ARM:SEQUence :LAYer:FILTer:LPASs 24-15 TRIGger Subsystem:TRIGger:SEQUence :FILTer:HPASs 24-26 TRIGger Subsystem:TRIGger:SEQUence :FILTer:LPASs 24-27
:FIELD TRIGger Subsystem:ARM:SEQUence :LAYer:VIDeo 24-19 TRIGger Subsystem:TRIGger:SEQUence :VIDeo 24-31	FREQuency Subsystem SENSe 18-37 - 18-41 SOURce 19-43 - 19-46
:FILTer CALCulate 4-11 - 4-14 INPut 11-5 OUTPut Subsystem 15-3 SOURce:DM 19-27 TRIGger Subsystem:ARM:SEQUence :LAYer 24-15 TRIGger Subsystem:TRIGger:SEQUence 24-26	:FRESistance :MEASure 3-8 SENSe:FUNCTion 18-53
FILTer Subsystem SENSe 18-33 - 18-35	FRESistance Subsystem SENSe 18-68 - 18-70
:FIXed SENSe:FREQuency 18-38 SOURce:FREQuency 19-43	FRONT SENSe:FUNCTion:SPEed 18-55
:FLIMits CALCulate:CLIMits 4-9	:FSENSor CALibration 5-7
FM SENSe:FUNCTion 18-52	:FTIME MEASure 3-13
FM Subsystem SENSe 18-36 SOURce 19-32 - 19-34	
:FORCE CALCulate:FILTer:GATE:FREQuency 4-15 CALCulate:FILTer:GATE:TIME 4-13 CALCulate:TRANSform:DISTance 4-29 CALCulate:TRANSform:FREQuency 4-31 CALCulate:TRANSform:TIME 4-27 SENSe:FUNCTion 18-52 SENSe:WINDow:TYPE 18-89	
FORCE Subsystem SOURce 19-35 - 19-42	
FORMat CALCulate 4-15 - 4-16 Instrument Model 2-5 SOURce:DM 19-27 TRIGger Subsystem:ARM:SEQUence :LAYer:VIDeo 24-20 TRIGger Subsystem:TRIGger:SEQUence :VIDeo:FIELd 24-31	
FORMat Subsystem 9-1 - 9-4	

1999 SCPI Command Reference

:FULL
 SENSE:FREQUENCY:SPAN 18-40
 SOURCE:CURRENT:SPAN 19-24
 SOURCE:FREQUENCY:SPAN 19-45
 SOURCE:POWER:SPAN 19-67
 SOURCE:RESISTANCE:SPAN 19-78
 SOURCE:VOLTAGE:SPAN 19-100
 <function>
 Measurement Instructions 3-5
 SENSE 18-43
 SENSE:DETECTOR 18-30
 FUNCTION & DATA Subsystem
 SENSE 18-42 - 18-57
 FUNCTION Subsystem
 SOURCE 19-47
 :<function_name>
 SENSE:FUNCTION 18-48
 Fundamental Measurement Layer
 MEASURE 3-7

G

:GAIN
 INPUT 11-6
 SENSE:CORRECTION 18-20
 SOURCE:CORRECTION 19-13
 :GATE
 CALCULATE:FILTER 4-11
 :GDAPerture
 CALCULATE 4-17
 :GDEVICE
 VXI:WSPROTOCOL:COMMAND 26-11
 :GENERATION ANALog
 SENSE:SWEeP 18-79
 SOURCE:SWEeP 19-86
 :GENERATION CONCurent
 SOURCE:LIST 19-51
 :GENERATION DCONcurrent
 SOURCE:LIST 19-51
 :GENERATION DSEQUence
 SOURCE:LIST 19-51
 :GENERATION SEQUence
 SOURCE:LIST 19-51
 :GENERATION STEPped
 SENSE:SWEeP 18-79
 SOURCE:SWEeP 19-86
 :GEOMETRY
 DISPLAY:WINDow 8-6
 :GPIB
 SYSTEM:COMMunicate 21-5
 :GRAPHics
 DISPLAY:WINDow 8-7

:GRATICule
 DISPLAY:WINDow:TRACe 8-11
 HCOPI:ITEM:WINDow:TRACe 10-12
 :GRID
 DISPLAY:WINDow:TRACe:GRATICule 8-11
 :GUARD
 INPUT 11-6

H

:HARMonic
 SENSE:MIXer 18-60
 HCOPI Subsystem 10-1 - 10-16
 :HEADers?
 SYSTem:HELP 21-28
 :HELP
 SYSTem 21-28 - 21-31
 HEXadecimal 9-2 - 9-3
 :HIERarchy?
 VXI:CONFigure 26-3
 :HIGH
 INPUT:POSITION:X:ANGLE:LIMit 11-8
 INPUT:POSITION:X:DISTance:LIMit 11-9
 INPUT:POSITION:Y:ANGLE:LIMit 11-10
 INPUT:POSITION:Y:DISTance:LIMit 11-11
 INPUT:POSITION:Z:ANGLE:LIMit 11-12
 INPUT:POSITION:Z:DISTance:LIMit 11-13
 MEASURE 3-12
 OUTPUT:POSITION:X:ANGLE:LIMit 15-6
 OUTPUT:POSITION:X:DISTance:LIMit 15-7
 OUTPUT:POSITION:Y:ANGLE:LIMit 15-8
 OUTPUT:POSITION:Y:DISTance:LIMit 15-9
 OUTPUT:POSITION:Z:ANGLE:LIMit 15-10
 OUTPUT:POSITION:Z:DISTance:LIMit 15-11
 SOURCE:CURRENT:LEVel:IMMEDIATE 19-20
 SOURCE:CURRENT:LEVel:TRIGgered 19-21
 SOURCE:CURRENT:LIMit 19-22
 SOURCE:POWER:LEVel:IMMEDIATE 19-63
 SOURCE:POWER:LEVel:TRIGgered 19-64
 SOURCE:POWER:LIMit 19-65
 SOURCE:RESistance:LEVel:IMMEDIATE 19-75
 SOURCE:RESistance:LEVel:TRIGgered 19-76
 SOURCE:RESistance:LIMit 19-76
 SOURCE:VOLTage:LEVel:IMMEDIATE 19-96
 SOURCE:VOLTage:LEVel:TRIGgered 19-97
 SOURCE:VOLTage:LIMit 19-98
 :HISTogram
 CALCULATE:TRANSform 4-24
 SENSE:FUNCTION:VOLTage 18-56
 :HOLD
 SENSE:FREQUENCY:SPAN 18-40
 SOURCE:CURRENT:SPAN 19-24

1999 SCPI Command Reference

SOURce:FREQuency:SPAN 19-45
 SOURce:POWer:SPAN 19-67
 SOURce:RESistance:SPAN 19-78
 SOURce:VOLTAge:SPAN 19-100
 :HOLD DCYCLe
 SOURce:PULSe 19-71
 :HOLD WIDTH
 SOURce:PULSe 19-71
 :HOLDoff
 TRIGger Subsystem:TRIGger:SEQuence 24-27
 :HORIZontal
 INPut:POLarization 11-7
 OUTPut:POLarization 15-5
 :HPASSs
 INPut:FILTer 11-5
 OUTPut:FILTer 15-4
 SENSe:FILTer 18-33
 TRIGger Subsystem:ARM
 :SEQuence:LAYer:FILTer 24-15
 TRIGger Subsystem:TRIGger
 :SEQuence:FILTer 24-26
 :HPGL
 DISPlay:WINDow:GRAPHics 8-8
 :HSL
 DISPlay:CMAP:COLor 8-5
 HCOPy:DEvice:CMAP:COLor 10-5
 :HYSTeresis
 SOURce:PULM:EXtErnal 19-68
 TRIGger Subsystem:ARM:SEQuence
 :LAYer 24-16
 TRIGger Subsystem:TRIGger:SEQuence 24-27

I

:I INVerted
 SOURce:DM:POLarity :ALL
 NORMallINVerted 19-30
 :I<n> NORMal
 SOURce:DM:POLarity :ALL
 NORMallINVerted 19-30
 :ICLock INVerted
 SOURce:DM:POLarity :ALL
 NORMallINVerted 19-30
 :ICLock NORMal
 SOURce:DM:POLarity :ALL
 NORMallINVerted 19-30
 :ICOMmander
 VXI:WSPRotocol:COMManD 26-11
 :ICORrection
 SOURce:DM:FILTer 19-27
 :IDLE
 CONTRol 6-3

:IDLE State
 TRIGger Subsystem:Model Layers 24-2
 :IDRaw
 DISPlay:WINDow:GRAPHics 8-8
 :IMMediate
 CALCulate 4-18
 CALCulate:LIMit:CLEar 4-21
 HCOPy 10-7 - 10-13, 10-15
 INPut:POSition:X:ANGLE 11-8
 INPut:POSition:X:DISTance 11-9
 INPut:POSition:Y:ANGLE 11-10
 INPut:POSition:Y:DISTance 11-11
 INPut:POSition:Z:ANGLE 11-12
 INPut:POSition:Z:DISTance 11-13
 OUTPut:POSition:X:ANGLE 15-6
 OUTPut:POSition:X:DISTance 15-7
 OUTPut:POSition:Y:ANGLE 15-8
 OUTPut:POSition:Y:DISTance 15-9
 OUTPut:POSition:Z:ANGLE 15-10
 OUTPut:POSition:Z:DISTance 15-11
 OUTPut:TTLTrgl:ECLTrg 15-12
 SOURce:CURREnt:LEVel 19-19
 SOURce:POWer:LEVel 19-62
 SOURce:RESistance:LEVel 19-75
 SOURce:VOLTAge:LEVel 19-95
 SYSTem:BEEPer 21-4
 TRIGger Subsystem:ARM:SEQuence
 :LAYer 24-16
 TRIGger Subsystem:INITiate 24-22
 TRIGger Subsystem:TRIGger:SEQuence 24-27
 :IMOVE
 DISPlay:WINDow:GRAPHics 8-8
 :IMPedance
 INPut 11-6
 OUTPut Subsystem 15-5
 SENSe:CORRection 18-19
 SOURce:AM:EXtErnal 19-7
 SOURce:FM:EXtErnal 19-33
 SOURce:PM:EXtErnal 19-59
 SOURce:PULM:EXtErnal 19-68
 :INformation?
 VXI:CONFigure 26-4
 :INITialize
 MMEMory 14-4
 INITiate 2-5
 CONTRol 6-3
 TRIGger Subsystem 24-21 - 24-22
 :Initiated
 TRIGger Subsystem:Model Layers 24-2
 INPut 11-1 - 11-14
 Instrument Model:Measurement Function 2-3
 SENSe:CORRection:IMPedance 18-19

1999 SCPI Command Reference

SENSE:CORRection:LOSS 18-20
 Instrument Model 2-1 - 2-14
 INSTRument Subsystem 12-1 - 12-4
 INTeger 9-2
 :INTegral
 CALCulate 4-18
 :INTERNAL
 SENSe:ROSCillator 18-71, 19-80
 SOURce:AM 19-7
 SOURce:FM 19-33
 SOURce:PM 19-59
 SOURce:PULM 19-69
 Internal Routing
 Instrument Model 2-5 - 2-14
 :INTERpolate
 CALCulate:LIMit 4-21
 Introduction 1-1 - 1-2
 :IQRatio
 SOURce:DM 19-28
 :ITEM
 HCOPY 10-7 - 10-12

K

:KBESsel
 CALCulate:FILTer:GATE:FREQuency 4-14
 CALCulate:FILTer:GATE:TIME 4-12
 CALCulate:TRANSform:DISTance 4-29
 CALCulate:TRANSform:FREQuency 4-31
 CALCulate:TRANSform:TIME 4-27
 SENSe:WINDow:TYPE 18-89
 :KEY
 DISPlay:MENU 8-6
 SYSTem 21-32
 :KLOCK
 SYSTem 21-33

L

:LABel
 DISPlay:WINDow:GRAPHics 8-8
 DISPlay:WINDow:TRACe:R 8-15
 DISPlay:WINDow:TRACe:X 8-12
 DISPlay:WINDow:TRACe:Y 8-13
 HCOPY:ITEM 10-9
 :LADdress?
 VXI:CONFigure 26-6
 Lamina and Cloned Models
 Instrument Model:Internal Routing 2-7
 :LANGuage
 HCOPY:DEvice 10-6
 SYSTem 21-33

:LAYer
 TRIGger Subsystem:ARM:SEQuence 24-14
 :LAYer Nomenclature
 TRIGger Subsystem
 :Expanded Capability Trigger Model 24-6
 :LDIREction
 DISPlay:WINDow:GRAPHics 8-8
 :LEADing
 SOURce:PULSe:TRANSition 19-72
 :LEAKage
 SOURce:DM 19-28
 :LEFT
 DISPlay:WINDow:TRACe:X:SCALE 8-12
 :LENGth
 HCOPY:PAGE 10-14
 :LEVel
 OUTPut:TTLTrgl:ECLTrg 15-13
 SENSe:CONDition 18-15
 SENSe:CURREnt:ACI:DC:PROTection 18-26
 SENSe:POWer:ACI:DC:PROTection 18-65
 SENSe:VOLTage:ACI:DC:PROTection 18-85
 SOURce:CURREnt 19-19
 SOURce:CURREnt:PROTection 19-23
 SOURce:POWer 19-62
 SOURce:POWer:PROTection 19-65
 SOURce:PULM:EXTernal 19-68
 SOURce:RESistance 19-75
 SOURce:RESistance:PROTection 19-77
 SOURce:VOLTage 19-95
 SOURce:VOLTage:PROTection 19-99
 TRIGger Subsystem:ARM:SEQuence
 :LAYer 24-16
 TRIGger Subsystem:TRIGger:SEQuence 24-28
 :LFRequency
 SYSTem 21-33
 :LIFT
 CONTrol 6-3
 :LIMit
 CALCulate 4-18 - 4-20
 INPut:POSition:X:ANGLE 11-8
 INPut:POSition:X:DISTance 11-9
 INPut:POSition:Y:ANGLE 11-10
 INPut:POSition:Y:DISTance 11-11
 INPut:POSition:Z:ANGLE 11-12
 INPut:POSition:Z:DISTance 11-13
 OUTPut:POSition:X:ANGLE 15-6
 OUTPut:POSition:X:DISTance 15-7
 OUTPut:POSition:Y:ANGLE 15-8
 OUTPut:POSition:Y:DISTance 15-9
 OUTPut:POSition:Z:ANGLE 15-10
 OUTPut:POSition:Z:DISTance 15-11
 SENSe:MIXer:BIAS 18-60

1999 SCPI Command Reference

SOURce:CURRent 19-21 SOURce:POWer 19-64 SOURce:RESistance 19-76 SOURce:VOLTage 19-97 :LINE TRACe DATA:DATA 23-3 TRIGger Subsystem:ARM :SEquence:LAYer:VIdEO 24-20 TRIGger Subsystem:TRIGger :SEquence:VIdEO 24-32 :LINK TRIGger Subsystem:ARM :SEquence:LAYer 24-17 TRIGger Subsystem:TRIGger:SEquence 24-28 :LINK CENTER DISPlay:WINDow:TRACe:X:SCALE:PDIV 8-13 SENSe:FREQuency:SPAN 18-40 SOURce:CURRent:SPAN 19-24 SOURce:FREQuency:SPAN 19-45 SOURce:POWer:SPAN 19-67 SOURce:RESistance:SPAN 19-78 SOURce:VOLTage:SPAN 19-100 :LINK LEFT DISPlay:WINDow:TRACe:X:SCALE:PDIV 8-13 :LINK RIGHT DISPlay:WINDow:TRACe:X:SCALE:PDIV 8-13 :LINK START SENSe:FREQuency:SPAN 18-40 SOURce:CURRent:SPAN 19-24 SOURce:FREQuency:SPAN 19-45 SOURce:POWer:SPAN 19-67 SOURce:RESistance:SPAN 19-78 SOURce:VOLTage:SPAN 19-100 :LINK STOP SENSe:FREQuency:SPAN 18-40 SOURce:CURRent:SPAN 19-24 SOURce:FREQuency:SPAN 19-45 SOURce:POWer:SPAN 19-67 SOURce:RESistance:SPAN 19-78 SOURce:VOLTage:SPAN 19-100 LIST Subsystem SENSe 18-58 - 18-59 SOURce 19-48 - 19-53 LL See also IDELeTe :LLEFt DISPlay:WINDow:GEOMetry 8-7 HCOPy:PAGE:DIMensions 10-14 :LLIMit SENSe:SWEep:TIME 18-83 SOURce:SWEep:TIME 19-85 :LOAD	MMEMory 14-4 :LOCate DISPlay:WINDow:TEXT 8-10 :LOSS MEMory:TABLE 13-16 SENSe:CORRection 18-20 SENSe:MIXer 18-61 SOURce:CORRection 19-13 :LOW INPut 11-6 INPut:POSition:X:ANGLE:LIMit 11-8 INPut:POSition:X:DISTance:LIMit 11-9 INPut:POSition:Y:ANGLE:LIMit 11-10 INPut:POSition:Y:DISTance:LIMit 11-11 INPut:POSition:Z:ANGLE:LIMit 11-12 INPut:POSition:Z:DISTance:LIMit 11-13 MEASure 3-12 OUTPut Subsystem 15-5 OUTPut:POSition:X:ANGLE:LIMit 15-6 OUTPut:POSition:X:DISTance:LIMit 15-7 OUTPut:POSition:Y:ANGLE:LIMit 15-8 OUTPut:POSition:Y:DISTance:LIMit 15-9 OUTPut:POSition:Z:ANGLE:LIMit 15-10 OUTPut:POSition:Z:DISTance:LIMit 15-11 SOURce:CURRent:LEVel:IMMediate 19-20 SOURce:CURRent:LEVel:TRIGgered 19-21 SOURce:CURRent:LIMit 19-22 SOURce:POWer:LEVel:IMMediate 19-63 SOURce:POWer:LEVel:TRIGgered 19-64 SOURce:POWer:LIMit 19-65 SOURce:RESistance:LEVel:IMMediate 19-75 SOURce:RESistance:LEVel:TRIGgered 19-76 SOURce:RESistance:LIMit 19-77 SOURce:VOLTage:LEVel:IMMediate 19-96 SOURce:VOLTage:LEVel:TRIGgered 19-97 SOURce:VOLTage:LIMit 19-98 :LOWer CALCulate:LIMit 4-20 SENSe:AM:DEPTh:RANGe 18-4 SENSe:CURRent:ACI:DC:RANGe 18-27 SENSe:FM:DEVIation:RANGe 18-36 SENSe:FREQuency:RANGe 18-39 SENSe:PM:DEVIation:RANGe 18-62 SENSe:POWer:ACI:DC:RANGe 18-66 SENSe:POWer:ACHannel:SPACing 18-64 SENSe:RESistance:RANGe 18-69 SENSe:VOLTage:ACI:DC:RANGe 18-86 :LPASs INPut:FILTer 11-5 OUTPut:FILTer 15-4 SENSe:FILTer 18-33 TRIGger Subsystem:ARM

1999 SCPI Command Reference

:SEquence:LAYer:FILTer 24-15
 TRIGger Subsystem:TRIGger
 :SEquence:FILTer 24-26
 :LPFRame
 TRIGger Subsystem:ARM
 :SEquence:LAYer:VIDeo:FORMat 24-20
 TRIGger Subsystem:TRIGger
 :SEquence:VIDeo:FIELD:FORMat 24-31
 :LTYPe
 DISPlay:WINDow:GRAPHics 8-8
 HCOpy:ITEM:WINDow:TRACe 10-13

M

:MACRo
 MMEMoRY:LOAD and :STORe 14-4
 :MACRo?
 MEMoRY:CATalog 13-5
 MEMoRY:FREE 13-8
 :MAGNitude
 MEMoRY:TABLE:CONDition 13-11
 SENSE:CORRection:LOSS:INPut 18-21
 SENSE:CORRection:OFFSet 18-21
 SOURce:CORRection:LOSS:OUTPut 19-14
 SOURce:CORRection:OFFSet 19-13
 SOURce:DM:IQRatio 19-28
 SOURce:DM:LEAKage 19-28
 :MAGNitude
 MEMoRY:TABLE:CURREnt 13-11
 MEMoRY:TABLE:LOSS 13-16
 MEMoRY:TABLE:POWEr 13-17
 MEMoRY:TABLE:RESistance 13-17
 MEMoRY:TABLE:VOLTage 13-19
 :MALLocate
 PROGram:EXPLicit 16-5
 PROGram:SELected 16-3
 :MANual
 SENSE:FREQuency 18-38
 SOURce:CURREnt 19-22
 SOURce:FREQuency 19-44
 SOURce:POWEr 19-65
 SOURce:RESistance 19-78
 SOURce:VOLTage 19-98
 :MAP
 STATus 20-4, 20-7
 MARKer Subsystem
 SOURce 19-54 - 19-55
 :MATH
 CALCulate 4-21 - 4-22
 :MAXimum
 MEASure 3-14
 :MCONtrol

CONTRol 6-4
 MEASure
 Measurement Instructions 3-5
 Measurement Function
 Instrument Model 2-3
 Measurement Function Layer 3-9
 Measurement Instructions 3-1 - 3-14
 :MEDium COAX
 SENSE:CORRection:RVELocity 18-22
 SOURce:CORRection:RVELocity 19-15
 :MEDium WAVEguide
 SENSE:CORRection:RVELocity 18-22
 SOURce:CORRection:RVELocity 19-15
 MEMoRY
 Instrument Model 2-5
 Memory Associated with Data Flow
 Instrument Model:Internal Routing 2-12
 MEMoRY Subsystem 13-1 - 13-20
 :MENU
 DISPlay 8-6
 HCOpy:ITEM 10-10
 :MESSAge
 VXI:WSPProtocol 26-13
 :METHod PMETER
 SOURce:CORRection:COLLect 19-12
 :METHod TPORT
 SENSE:CORRection:COLLect 18-18
 :MGRules
 TRIGger Subsystem:ARM:SESequence
 :DEFine 24-13
 TRIGger Subsystem:TRIGger:SESequence
 :DEFine 24-25
 :MINimum
 MEASure 3-14
 MIXer Subsystem
 SENSE 18-60 - 18-61
 MMEMoRY Subsystem 14-1 - 14-6
 :MODE
 HCOpy:DEVice 10-7
 SOURce:AM 19-7
 SOURce:FUNCTion 19-47
 SOURce:PULM 19-69
 :MODE AUTO
 SENSE:SWEep 18-79
 SOURce:SWEep 19-86
 :MODE CW
 SENSE:FREQuency 18-38
 SOURce:FREQuency 19-44
 :MODE DELTA
 SOURce:MARKer 19-54
 :MODE FIXEd
 SENSE:FREQuency 18-38

1999 SCPI Command Reference

SOURce:CURRent 19-22
 SOURce:FREQuency 19-44
 SOURce:POWer 19-65
 SOURce:RESistance 19-79
 SOURce:VOLTAge 19-98
 :MODE FREQuency
 SOURce:MARKer 19-54
 :MODE LIST
 SENSe:FREQuency 18-38
 SOURce:CURRent 19-22
 SOURce:FREQuency 19-44
 SOURce:POWer 19-65
 SOURce:RESistance 19-79
 SOURce:VOLTAge 19-98
 :MODE LOCKed
 SOURce:FM 19-33
 SOURce:PM 19-57
 :MODE MANual
 SENSe:SWEep 18-79
 SOURce:SWEep 19-86
 :MODE POSition
 SOURce:MARKer 19-54
 :MODE SENSe
 SOURce:FREQuency 19-44
 :MODE SOURce
 SENSe:FREQuency 18-38
 :MODE SWEep
 SENSe:FREQuency 18-38
 SOURce:CURRent 19-22
 SOURce:FREQuency 19-44
 SOURce:POWer 19-65
 SOURce:RESistance 19-79
 SOURce:VOLTAge 19-98
 :MODE UNLOCKed
 SOURce:FM 19-33
 SOURce:PM 19-57
 :Model Layers
 TRIGger Subsystem 24-2 - 24-3
 :MODule
 ROUTE 17-2
 :MOVE
 DISPlay:WINDow:GRAPHics 8-9
 MMEMory 14-5
 MOVing
 CALCulate:AVERage:TCONtrol 4-7
 :MSIS
 MMEMory 14-5
 :MULTiplier
 SENSe:FREQuency 18-39
 SOURce:FREQuency 19-44

N

:NAME
 CALCulate:MATH:EXPRession 4-22
 DISPlay:MENU 8-6
 MEMory:CLEAR 13-6
 MEMory:COPY 13-6
 MEMory:DELeTe 13-7
 MEMory:EXCHange 13-7
 MMEMory 14-5
 PROGram:SELeCted 16-3
 ROUTE:MODule:DELeTe 17-3
 ROUTE:PATH:DELeTe 17-4
 TRACE|DATA:DELeTe 23-4
 TRIGger Subsystem:INITiate:CONTinuous 24-22
 TRIGger Subsystem:INITiate:IMMediate 24-23
 :NDUTycle
 MEASure 3-14
 :No Error
 SYSTem:ERRor? 21-14
 NORMal
 CALCulate:AVERage:TCONtrol 4-7
 notation
 X_n^* 4-1
 :NPLCycles
 SENSe:CURRent:ACI:DC 18-26
 SENSe:POWer:ACI:DC 18-64
 SENSe:RESistance 18-68
 SENSe:VOLTAge:ACI:DC 18-85
 :NSElect
 INSTrument 12-4
 :NSTates?
 MEMory 13-9
 :NTRansition
 STATus 20-4, 20-7
 :NUMBer
 PROGram:EXPLicit 16-5
 PROGram:SELeCted 16-3
 TRIGger Subsystem:ARM:SEQuence
 :LAYer:VIDeo:FIELD 24-19
 TRIGger Subsystem:ARM:SEQuence
 :LAYer:VIDeo:LINE 24-20
 TRIGger Subsystem:TRIGger:SEQuence
 :VIDeo:FIELD 24-31
 TRIGger Subsystem:TRIGger:SEQuence
 :VIDeo:LINE 24-32
 :NUMBer?
 VXI:CONFigure 26-7
 Numeric Suffixes
 Instrument Model:Internal Routing 2-7
 :NWIDTH
 MEASure 3-14

1999 SCPI Command Reference

O

:OCOMpensated
 SENSe:RESistance 18-68
 :OCONdition ,
 TRACe|DATA:FEED 23-5
 OCTal 9-2 - 9-3
 :OEDGe
 DISPlay:WINDow:TRACe:R:SCALE 8-16
 :OFF
 SENSe:FUNCTion 18-44
 :OFFSet
 INPut 11-7
 :OFFSet
 INPut:POSition:X:ANGLE 11-8
 INPut:POSition:X:DISTance 11-9
 INPut:POSition:Y:ANGLE 11-10
 INPut:POSition:Y:DISTance 11-11
 INPut:POSition:Z:ANGLE 11-12
 INPut:POSition:Z:DISTance 11-13
 OUTPut:POSition:X:ANGLE 15-6
 OUTPut:POSition:X:DISTance 15-7
 OUTPut:POSition:Y:ANGLE 15-8
 OUTPut:POSition:Y:DISTance 15-9
 OUTPut:POSition:Z:ANGLE 15-10
 OUTPut:POSition:Z:DISTance 15-11
 SENSe:CORRection 18-21
 SENSe:CURREnt:AC|DC:RANGE 18-28
 SENSe:FREQuency 18-39
 SENSe:POWer:AC|DC:RANGE 18-66
 SENSe:VOLTAge:AC|DC:RANGE 18-87
 SOURce:CORRection 19-13
 SOURce:CURREnt:LEVel:IMMediate 19-20
 SOURce:CURREnt:LEVel:TRIGgered 19-21
 SOURce:CURREnt:LIMit 19-22
 SOURce:FREQuency 19-44
 SOURce:POWer:LEVel:IMMediate 19-63
 SOURce:POWer:LEVel:TRIGgered 19-64
 SOURce:POWer:LIMit 19-64
 SOURce:RESistance:LEVel:IMMediate 19-75
 SOURce:RESistance:LEVel:TRIGgered 19-76
 SOURce:RESistance:LIMit 19-76
 SOURce:VOLTAge:LEVel:IMMediate 19-96
 SOURce:VOLTAge:LEVel:TRIGgered 19-97
 SOURce:VOLTAge:LIMit 19-98
 :ON
 SENSe:FUNCTion 18-44
 ONCE
 CALCulate 4-7, 4-12, 4-14, 4-21, 4-26, 4-28, 4-31
 CALibration 5-7
 DISPlay 8-11 - 8-12, 8-14 - 8-15
 INPut:ATTenuation 11-3

INPut:GAIN 11-6
 OUTPut:FILTer 15-3
 SENSe 18-4, 18-7, 18-10 - 18-11, 18-27 - 18-28,
 18-31, 18-36, 18-38 - 18-40, 18-59 - 18-60, 18-62,
 18-66 - 18-67, 18-69 - 18-70, 18-72, 18-79, 18-82,
 18-86 - 18-87
 SENSe:ROSCillator:SOURce INTernal 19-80
 SOURce 19-18 - 19-20, 19-23, 19-43, 19-45,
 19-50, 19-52, 19-61 - 19-62, 19-66, 19-73, 19-79,
 19-85 - 19-86, 19-94 - 19-96, 19-99
 SYSTem 21-33
 TRACe|DATA 23-5
 TRIGger Subsystem:ARM
 :SEquence:LAYer:DELay 24-14
 TRIGger Subsystem:ARM
 :SEquence:LAYer:LEVel 24-16
 TRIGger Subsystem:TRIGger
 :SEquence:DELay 24-26
 TRIGger Subsystem:TRIGger:SEquence
 :LEVel 24-28
 :OPEN
 MMEMory 14-6
 ROUTe 17-3
 :OPERation
 STATus 20-3
 Operation Complete Event
 SYSTem:ERRor? 21-28
 OPERation Status Register 20-3
 :ORDinate
 CALCulate:TRANSform:HISTogram 4-24
 :ORientation
 HCOPy:PAGE 10-14
 Other Subsystems
 Instrument Model:Internal Routing 2-9
 OUTPut
 Instrument Model:Signal Generation 2-4
 SENSe:CORRection:IMPedance 18-19
 SENSe:CORRection:LOSS 18-20
 SOURce:CORRection:LOSS 19-14
 OUTPut Subsystem 15-1 - 15-14
 :OVERshoot
 MEASure:FALL 3-13
 MEASure:RISE 3-12

P

:PACE ACK
 SYSTem:COMM:SERial:RECeive 21-7
 SYSTem:COMM:SERial:TRANsmit 21-9
 :PACE NONE
 SYSTem:COMM:SERial:RECeive 21-7
 SYSTem:COMM:SERial:TRANsmit 21-9

1999 SCPI Command Reference

:PACE XON SYSTem:COMM:SERial:RECeive 21-7 SYSTem:COMM:SERial:TRANsmit 21-9 :PACK MMEMory 14-6 PACKed 9-2 :PAGE DISPlay:WINDow:TEXT 8-10 HCOPy 10-13 - 10-14 :PARity SYSTem:COMM:SERial:RECeive 21-8 SYSTem:COMM:SERial:TRANsmit 21-9 :PATH CALCulate 4-31 - 4-32 ROUTe 17-3 :PCL DISPlay:WINDow:GRAPHics 8-8 :PDFunction SENSe:FUNCTION:VOLTage 18-57 :PDIVision DISPlay:WINDow:TRACe:X:SCALE 8-13 DISPlay:WINDow:TRACe:Y:SCALE 8-14 :PDUTYcycle :MEASure 3-14 :PERiod :MEASure 3-10 SENSe:FUNCTION 18-53 SOURce:PULSe 19-71 :PERSistence DISPlay:WINDow:TRACe 8-11 :PHASe :MEASure 3-10 MEMory Subsystem:TABLE:CURRENT 13-11 MEMory Subsystem:TABLE:LOSS 13-16 MEMory Subsystem:TABLE:VOLTage 13-19 SENSe:CORRection:LOSS:INPut:OUTPut 18-21 SENSe:CORRection:OFFSet 18-21 SENSe:FUNCTION 18-53 SOURce:CORRection:LOSS:OUTPut 19-14 SOURce:CORRection:OFFSet 19-13 PHASe Subsystem SOURce 19-56 :PLOSs CALibration 5-4 - 5-5 PM SENSe:FUNCTION 18-53 PM Subsystem SENSe 18-62 SOURce 19-57 - 19-59 :POFLag TRIGger Subsystem:INITiate 24-23 :POINT	SOURce:MARKer 19-54 :POINTs CALCulate 4-10 CALCulate:FILTer:GATE:FREQuency 4-14 CALCulate:FILTer:GATE:TIME 4-12 CALCulate:SMOothing 4-23 CALCulate:TRANSform:DISTance 4-28 CALCulate:TRANSform:FREQuency 4-31 CALCulate:TRANSform:HISTogram 4-24 CALCulate:TRANSform:TIME 4-26 SENSe:SMOothing 18-73 SENSe:SWEEP 18-80 SOURce:SWEEP 19-87 TRACe DATA 23-5 - 23-6 :POINTs? CALCulate:CLIMits:FLIMits 4-9 CALCulate:LIMit:CONTROL 4-19 CALCulate:LIMit:LOWer 4-20 CALCulate:LIMit:REPort 4-21 CALCulate:LIMit:UPPer 4-20 MEMory:TABLE:CONDition:MAGNitude 13-11 MEMory:TABLE:CURRENT:MAGNitude 13-11 MEMory:TABLE:CURRENT:PHASe 13-12 MEMory:TABLE:FREQuency 13-15 MEMory:TABLE:LOSS:MAGNitude 13-16 MEMory:TABLE:LOSS:PHASe 13-16 MEMory:TABLE:POWER:MAGNitude 13-17 MEMory:TABLE:RESistance:MAGNitude 13-17 MEMory:TABLE:VOLTage:MAGNitude 13-19 MEMory:TABLE:VOLTage:PHASe 13-20 SENSe:LIST:DWELL 18-58 SENSe:LIST:FREQuency 18-59 SENSe:LIST:SEQuence 18-59 SOURce:LIST:APRObe 19-49 SOURce:LIST:CONCurent 19-50 SOURce:LIST:CONTROL:APOWer 19-50 SOURce:LIST:CONTROL:BLOWer 19-50 SOURce:LIST:CONTROL:COMPRESSor 19-50 SOURce:LIST:CURRENT 19-50 SOURce:LIST:DEPTH:AM 19-49 SOURce:LIST:DWELL 19-51 SOURce:LIST:FREQuency 19-51 SOURce:LIST:POWER 19-52 SOURce:LIST:PULM:STATe 19-51 SOURce:LIST:RESistance 19-52 SOURce:LIST:RTIME 19-52 SOURce:LIST:SEQuence 19-52 SOURce:LIST:TEMPerature 19-52 SOURce:LIST:VOLTage 19-53 :POLarity INPut 11-7 OUTPut Subsystem 15-5

1999 SCPI Command Reference

:POLarity :ALL NORMal SOURce:DM 19-30 :POLarity COMPLEMENT SOURce:PULSe 19-73 :POLarity INVERTed OUTPut:TTLTrgl:ECLTrg 15-13 SOURce:AM 19-8 SOURce:AM:EXTernal 19-7 SOURce:DM 19-30 SOURce:FM 19-34 SOURce:FM:EXTernal 19-33 SOURce:PM 19-58 SOURce:PM:EXTernal 19-59 SOURce:PULM 19-69 SOURce:PULM:EXTernal 19-69 SOURce:PULSe 19-73 :POLarity NEGative TRIGger Subsystem:ARM:SEQUence :LAYer:VIDeo:SSIGnal 24-21 TRIGger Subsystem:TRIGger:SEQUence :VIDeo:SSIGnal 24-32 :POLarity NORMal OUTPut:TTLTrgl:ECLTrg 15-13 SOURce:AM 19-8 SOURce:AM:EXTernal 19-7 SOURce:FM 19-34 SOURce:FM:EXTernal 19-33 SOURce:PM 19-58 SOURce:PM:EXTernal 19-59 SOURce:PULM:EXTernal 19-69 SOURce:PULSe 19-73 SOURce:PULSe Modulation 19-69 :POLarity POSitive TRIGger Subsystem:ARM:SEQUence :LAYer:VIDeo:SSIGnal 24-21 TRIGger Subsystem:TRIGger:SEQUence :VIDeo:SSIGnal 24-32 :POLarization INPut 11-7 OUTPut Subsystem 15-5 :POSition OUTPut Subsystem 15-5 :POSition INPut 11-7 - 11-13 :POWER :MEASure 3-8 MEMory:TABLE 13-17 SENSe:FUNCTION 18-53 SOURce:LIST 19-52 UNIT 25-1 Power On Event SYSTEM:ERRor? 21-27	POWER Subsystem SENSe 18-63 - 18-67 SOURce 19-60 - 19-67 :PREference CALCulate:FORMat:UPHase 4-17 Presentation Layer Measurement Instructions 3-7 :<presentation_layer> SENSe:FUNCTION 18-47 :PRESet STATus 20-4 SYSTem 21-35 :PREShoot MEASure:FALL 3-13 MEASure:RISE 3-12 :PRF :MEASure:FREQuency 3-10 PROGram Subsystem 16-1 - 16-6 :PROtection OUTPut Subsystem 15-12 SENSe:CURREnt:ACI:DC 18-26 SENSe:POWER:ACI:DC 18-65 SENSe:VOLTage:ACI:DC 18-85 SOURce:CURREnt 19-23 SOURce:POWER 19-65 SOURce:RESistance 19-77 SOURce:VOLTage 19-99 :PROTocol TRIGger Subsystem:ARM:SEQUence :LAYer 24-17 TRIGger Subsystem:TRIGger:SEQUence 24-28 :PROTocol ASYNchronous OUTPut:TTLTrgl:ECLTrg 15-13 :PROTocol SSYNchronous OUTPut:TTLTrgl:ECLTrg 15-13 :PROTocol SYNchronous OUTPut:TTLTrgl:ECLTrg 15-13 :PSDensity SENSe:FUNCTION:POWER 18-54 :PTPeak MEASure 3-14 SENSe:CURREnt:ACI:DC:RANGe 18-28 SENSe:POWER:ACI:DC:RANGe 18-66 SENSe:VOLTage:ACI:DC:RANGe 18-87 :PTRansition STATus 20-4, 20-7 :PULM SOURce:LIST 19-51 PULSe Modulation Subsystem SOURce 19-68 - 19-70 PULSe Subsystem SOURce 19-71 - 19-73

1999 SCPI Command Reference

:PWIDth
MEASure 3-13

Q

:Q INVerted
SOURce:DM:POLarity :ALL
NORMallINVerted 19-30
:Q<n> NRMal
SOURce:DM:POLarity :ALL
NORMallINVerted 19-30
:QCLock INVerted
SOURce:DM:POLarity :ALL
NORMallINVerted 19-30
:QCLock NORMal
SOURce:DM:POLarity :ALL
NORMallINVerted 19-30
:QCORrection
SOURce:DM:FILTer 19-27
:QUADrant
HCOPY:PAGE:DIMensions 10-14
:QUADrature
SOURce:DM 19-28
:QUERY
VXI:WSPRotocol 26-14
:Query Error
SYSTEM:ERRor? 21-26
Querying the Data Flow
Instrument Model:Internal Routing 2-14
:QUESTionable
STATus 20-7
QUESTionable Data 20-7

R

:R
DISPlay:WINDow:TRACe 8-15
:RANGe
CALCulate:TRANSform:HISTogram 4-24
SENSe:AM:DEPTTh 18-4
SENSe:CURREnt:ACI:DC 18-27
SENSe:FM:DEViation 18-36
SENSe:FREQuency 18-39
SENSe:PM:DEViation 18-62
SENSe:POWER:ACI:DC 18-65
SENSe:RESistance 18-69
SENSe:VOLTage:ACI:DC 18-86
SOURce:CURREnt 19-23
SOURce:POWER 19-66
SOURce:RESistance 19-79
SOURce:VOLTage 19-99
:RATio

SENSe:BANDwidth|BWIDth:RESolution 18-10
SENSe:BANDwidth|BWIDth:VIDeo 18-11
:RDEvice
SYSTem:COMMunicate:GPIB 21-5
VXI:WSPRotocol:COMMand 26-11
:RDEvice?
VXI:WSPRotocol:QUERy 26-15
READ
Measurement Instructions 3-4
:READ?
VXI:REGister 26-7
REAL 9-2
:REALtime
SENSe:SWEep 18-81
REAR
SENSe:FUNCTion:SPEEd 18-55
:RECeive
SYSTem:COMM:SERial 21-7
:RECeive?
VXI:WSPRotocol:MESSAge 26-13
:REFerence
SENSe:CURREnt:ACI:DC 18-28
SENSe:POWER:ACI:DC 18-67
SENSe:RESistance 18-70
SENSe:VOLTage:ACI:DC 18-87
SOURce:CURREnt 19-23
SOURce:MARKer 19-55
SOURce:PHASe 19-56
SOURce:POWER 19-66
SOURce:RESistance 19-79
SOURce:VOLTage 19-99
:REGister
VXI 26-7
REPeat
CALCulate:AVERage:TCONtrol 4-7
:REPort
CALCulate:LIMit 4-20
Request Control Event
SYSTEM:ERRor? 21-28
:Requirements
Introduction 1-1
:RESet?
VXI 26-8
:RESistance
:MEASure 3-8
MEMory:TABLE 13-17
SENSe:FUNCTion 18-55
SOURce:LIST 19-52
RESistance Subsystem
SENSe 18-68 - 18-70
SOURce 19-74 - 19-79
:RESolution

1999 SCPI Command Reference

HCOpy:DEvIce 10-7
 SENSE:BANDwidth|BWIDth 18-10
 SENSE:CURRENT:AC|DC 18-28
 SENSE:FREQuency 18-40
 SENSE:POWer:AC|DC 18-67
 SENSE:RESistance 18-70
 SENSE:VOLTage:AC|DC 18-87
 SOURce:FREQuency 19-45
 :RESPonse?
 VXI:WSPRotocol 26-17
 :RGB
 DISPlay:CMAP:COLor 8-5
 HCOpy:DEvIce:CMAP:COLor 10-5
 :RHAndlers
 VXI:WSPRotocol:COMMand 26-11
 :RHAndlers?
 VXI:WSPRotocol:QUERy 26-15
 :RHLine
 VXI:WSPRotocol:COMMand 26-12
 :RHLine?
 VXI:WSPRotocol:QUERy 26-15
 :RIGHt
 DISPlay:WINDow:TRACe:X:SCALE 8-13
 :RILine
 VXI:WSPRotocol:COMMand 26-12
 :RILine?
 VXI:WSPRotocol:QUERy 26-16
 :RINTerrupter
 VXI:WSPRotocol:COMMand 26-12
 :RINTerrupter?
 VXI:WSPRotocol:QUERy 26-16
 :RISE
 MEASure 3-12
 :RLEVel
 DISPlay:WINDow:TRACe:Y:SCALE 8-14
 :RLINE
 DISPlay:WINDow:TRACe:Y 8-13
 :RMODid
 VXI:WSPRotocol:COMMand 26-12
 :RMODid?
 VXI:WSPRotocol:QUERy 26-16
 :ROSCillator
 OUTPut Subsystem 15-12
 ROOSCillator Subsystem
 SENSe 18-71 - 18-72
 SOURce 19-80 - 19-81
 :ROTation
 CONTRol 6-4
 ROUTe Subsystem 17-1 - 17-6
 :RPERror
 VXI:WSPRotocol:COMMand 26-12
 :RPERror?

 VXI:WSPRotocol:QUERy 26-16
 :RPOStion
 DISPlay:WINDow:TRACe:Y:SCALE 8-15
 :RPRotocol
 VXI:WSPRotocol:COMMand 26-12
 :RPRotocol?
 VXI:WSPRotocol:QUERy 26-16
 :RSARea
 VXI:WSPRotocol:COMMand 26-13
 :RSARea?
 VXI:WSPRotocol:QUERy 26-16
 :RSTB
 VXI:WSPRotocol:COMMand 26-12
 :RSTB?
 VXI:WSPRotocol:QUERy 26-16
 :RTIME
 MEASure 3-12
 SOURce:LIST 19-52
 :RTS IBFull
 SYSTem:COMM:SERial:CONTRol 21-6
 :RTS OFF
 SYSTem:COMM:SERial:CONTRol 21-6
 :RTS ON
 SYSTem:COMM:SERial:CONTRol 21-6
 :RTS RFR
 SYSTem:COMM:SERial:CONTRol 21-6
 :RTS STANDard
 SYSTem:COMM:SERial:CONTRol 21-6
 :RVELocity
 SENSe:CORRection 18-22
 SOURce:CORRection 19-15

S

:S11
 SENSe:FUNCTion:POWer 18-54
 :S12
 SENSe:FUNCTion:POWer 18-54
 :S21
 SENSe:FUNCTion:POWer 18-54
 :S22
 SENSe:FUNCTion:POWer 18-54
 :SAMPLE
 ROUTe 17-4
 :SAVE
 SENSe:CORRection:COLlect 18-18
 SOURce:CORRection:COLlect 19-12
 SB
 See SENSe:FUNCTion
 :SBITs
 SYSTem:COMM:SERial:RECeive 21-8
 SYSTem:COMM:SERial:TRANsmit 21-9

1999 SCPI Command Reference

:SCALar	SYSTem:COMMunicate:GPIB 21-5
MEASure 3-7	:SEND
:SCALE	VXI:WSProtocol:MESSage 26-13
DISPlay:WINDow:TRACe:R 8-15	SENSe
DISPlay:WINDow:TRACe:X 8-12	Instrument Model:Measurement Function 2-4
DISPlay:WINDow:TRACe:Y 8-13	SENSe:DATA? 18-42
HCOpy:PAGE 10-14	SENSe Subsystem 18-1 - 18-90
:SCAN	:SENSitivity
ROUTe 17-6	SOURce:AM 19-8
:SDUMp	SOURce:FM 19-34
HCOpy 10-15 - 10-16	SOURce:PM 19-57
:SEARch	Sensor Function Selection
SOURce:CURRent:ALC 19-18	Instrument Model:Internal Routing 2-9
SOURce:POWer:ALC 19-61	:<sensor_function>
SOURce:VOLTAge:ALC 19-94	SENSe:FUNCTion 18-45
:SECurity	:SEQUence
SYSTem 21-36	SENSe:LIST 18-59
:SElect	SOURce:LIST 19-52
INSTrument 12-4	TRIGger Subsystem:ARM 24-12
MEMory:TABLE 13-18	TRIGger Subsystem:INITiate:CONTinuous 24-22
SENSe:CORRection:CSET 18-18	TRIGger Subsystem:INITiate:IMMediate 24-23
SOURce:CORRection:CSET 19-12	TRIGger Subsystem:TRIGger 24-23
TRIGger Subsystem:ARM:SEQUence	:Sequence Event Use
:LAYer:VIDeo:LINE 24-20	TRIGger Subsystem 24-4
TRIGger Subsystem:TRIGger:SEQUence	:SERial
:VIDeo:LINE 24-32	SYSTem:COMMunicate 21-6
VXI 26-9	:SET
:SElect ALL	SYSTem 21-36
TRIGger Subsystem:ARM:SEQUence	:SHApe
:LAYer:VIDeo:FIELD 24-19	SOURce:FUNCTion 19-47
TRIGger Subsystem:TRIGger:SEQUence	:SHApe LINear
:VIDeo:FIELD 24-31	SENSe:DETEctor 18-31
:SElect EVEN	:SHApe LOGarithmic
TRIGger Subsystem:ARM:SEQUence	SENSe:DETEctor 18-31
:LAYer:VIDeo:FIELD 24-19	:SIGNal
TRIGger Subsystem:TRIGger	TRIGger Subsystem:ARM
:SEQUence:VIDeo:FIELD 24-31	:SEQUence:LAYer 24-17
:SElect NUMBer	TRIGger Subsystem:TRIGger:SEQUence 24-29
TRIGger Subsystem:ARM	Signal Generation
:SEQUence:LAYer:VIDeo:FIELD 24-19	Instrument Model 2-4
TRIGger Subsystem:TRIGger	Signal Routing
:SEQUence:VIDeo:FIELD 24-31	Instrument Model 2-2
:SElect ODD	Signal Status Register 20-7
TRIGger Subsystem:ARM	Simple Measurements
:SEQUence:LAYer:VIDeo:FIELD 24-19	Measurement Function Layer 3-9
TRIGger Subsystem:TRIGger	:SIZE
:SEQUence:VIDeo:FIELD 24-31	DISPlay:WINDow:GEOMetry 8-7
:SElected	HCOpy:PAGE 10-15
CALCulate:MATH:EXPRession:DELeTe 4-22	:SLEW
PROGram 16-2 - 16-3	SOURce:CURRent 19-24
PROGram:SElected:DELeTe 16-2	SOURce:POWer 19-66
:SELF	SOURce:RESistance 19-77

1999 SCPI Command Reference

SOURce:VOLTage 19-100	SOURce:AM 19-8
:SLModid	SOURce:DM 19-27
VXI:WSPProtocol:COMManD 26-13	SOURce:DM:CLOCK 19-31
:SLModid?	SOURce:DM:FILTer 19-27
VXI:WSPProtocol:QUERy 26-17	SOURce:DM:FRAMe 19-30
:SLOCK	SOURce:FM 19-34
VXI:WSPProtocol:COMManD 26-13	SOURce:PHASe 19-56
:SLOPe	SOURce:PM 19-58
SENSe:CORRection 18-20	SOURce:PULM 19-70
SOURce:CORRection 19-13	:SOURce INTernal
:SLOPe EITHer	SENSe:ROSCillator 18-71, 19-80
TRIGger Subsystem:ARM	SOURce:AM 19-8
:SEQuence:LAYer 24-18	SOURce:CURRENT:ALC 19-19
TRIGger Subsystem:TRIGger:SEQuence 24-29	SOURce:DM:CLOCK 19-31
:SLOPe NEGative	SOURce:DM:FILTer 19-27
TRIGger Subsystem:ARM	SOURce:DM:FRAMe 19-30
:SEQuence:LAYer 24-18	SOURce:PHASe 19-56
TRIGger Subsystem:TRIGger:SEQuence 24-29	SOURce:POWer:ALC 19-62
:SLOPe POSitive	SOURce:VOLTage:ALC 19-95
TRIGger Subsystem:ARM	:SOURce INTernal,EXTernal
:SEQuence:LAYer 24-18	SOURce:FM 19-34
TRIGger Subsystem:TRIGger:SEQuence 24-29	SOURce:PM 19-58
:SMOothing	SOURce:PULM 19-70
CALCulate 4-23	:SOURce MMHead
SMOothing Subsystem	SOURce:CURRENT:ALC 19-19
SENSe 18-73	SOURce:POWer:ALC 19-62
:SNDRatio	SOURce:VOLTage:ALC 19-95
SENSe:FUNCTION & DATA:FUNCTION:POWer	:SOURce NONE
18-51 - 18-53, 18-55	SENSe:ROSCillator 18-71, 19-80
SOURce	SOURce:DM:CLOCK 19-31
CALibration 5-6	:SOURce PMETer
Instrument Model:Signal Generation 2-5	SOURce:CURRENT:ALC 19-19
OUTPut:TTLTrgl:ECLTrg 15-13	SOURce:POWer:ALC 19-62
TRIGger Subsystem:ARM	SOURce:VOLTage:ALC 19-95
:SEQuence:LAYer 24-18	:SOURce PRBS
TRIGger Subsystem:TRIGger:SEQuence 24-29	SOURce:DM 19-27
:SOURce ,INTernal	SOURce Subsystem 19-1 - 19-102
SOURce:FM 19-34	:SPACing
SOURce:PM 19-58	SENSe:POWer:ACHannel 18-63
SOURce:PULM 19-70	:SPACing LINear
:SOURce CALibrate	DISPlay:WINDow:TRACe:R 8-16
SOURce:DM 19-27	DISPlay:WINDow:TRACe:Y 8-15
:SOURce CLK10	SENSe:SWEep 18-81
SENSe:ROSCillator 18-71, 19-80	SOURce:SWEep 19-86
:SOURce CLK100	:SPACing LOGarithmic
SENSe:ROSCillator 18-71, 19-80	DISPlay:WINDow:TRACe:R 8-16
:SOURce DIODE	DISPlay:WINDow:TRACe:Y 8-15
SOURce:CURRENT:ALC 19-19	SENSe:SWEep 18-81
SOURce:POWer:ALC 19-62	SOURce:SWEep 19-86
SOURce:VOLTage:ALC 19-95	:SPAN
:SOURce EXTernal	CALCulate:FILTer:GATE:FREQuency 4-14
SENSe:ROSCillator 18-71, 19-80	CALCulate:FILTer:GATE:TIME 4-12

1999 SCPI Command Reference

CALCulate:GDAPerture 4-17	INPut:FILTer:AWeighting 11-5
CALCulate:TRANSform:DISTance 4-28	INPut:FILTer:HPASs 11-5
CALCulate:TRANSform:FREQuency 4-30	INPut:FILTer:LPASs 11-5
CALCulate:TRANSform:TIME 4-26	INPut:GAIN 11-6
SENSe:FREQuency 18-40	INPut:OFFSet 11-7
SOURce:CURREnt 19-24	INPut:POSition:X:ANGLE:LIMit 11-8
SOURce:FREQuency 19-45	INPut:POSition:X:DISTance:LIMit 11-9
SOURce:POWEr 19-67	INPut:POSition:Y:ANGLE:LIMit 11-10
SOURce:RESistance 19-77	INPut:POSition:Y:DISTance:LIMit 11-11
SOURce:VOLTage 19-100	INPut:POSition:Z:ANGLE:LIMit 11-12
:SPEed	INPut:POSition:Z:DISTance:LIMit 11-13
HCOPy:DEVice 10-7	INSTrument 12-4
SENSe:FUNCTion 18-55	MEMory 13-9
SPEed Subsystem	MMEMory 14-4
SOURce 19-82 - 19-84	OUTPut Subsystem 15-14
:SREGister	OUTPut:FILTer:EXTernal 15-4
FORMat 9-3 - 9-4	OUTPut:FILTer:HPASs 15-4
SSB Subsystem	OUTPut:FILTer:LPASs 15-4
SENSe 18-74	OUTPut:POSition:X:ANGLE:LIMit 15-6
:SSignal	OUTPut:POSition:X:DISTance:LIMit 15-7
TRIGger Subsystem:ARM	OUTPut:POSition:Y:ANGLE:LIMit 15-8
:SEQUence:LAYer:VIDeo 24-20	OUTPut:POSition:Y:DISTance:LIMit 15-9
TRIGger Subsystem:TRIGger	OUTPut:POSition:Z:ANGLE:LIMit 15-10
:SEQUence:VIDeo 24-32	OUTPut:POSition:Z:DISTance:LIMit 15-11
STABilize Subsystem	OUTPut:PROTection 15-12
SENSe 18-75 - 18-76	OUTPut:ROSCillator 15-12
:Standard SEQuences	OUTPut:TTLTrg:ECLTrg 15-13
TRIGger Subsystem:Expanded	PROgram 16-3, 16-5
Capability Trigger Model 24-7	SENSe 18-7, 18-18 - 18-19, 18-21 - 18-23, 18-26,
:START	18-28, 18-33 - 18-35, 18-65, 18-67, 18-70, 18-73,
CALCulate:FILTer:GATE:FREQuency 4-13	18-81, 18-85, 18-87
CALCulate:FILTer:GATE:TIME 4-11	SOURce 19-8, 19-11 - 19-15, 19-18, 19-22 -
CALCulate:TRANSform:DISTance 4-28	19-24, 19-27 - 19-29, 19-34, 19-51, 19-55, 19-58,
CALCulate:TRANSform:FREQuency 4-30	19-61, 19-65 - 19-66, 19-70, 19-72, 19-77, 19-79,
CALCulate:TRANSform:TIME 4-25	19-94, 19-98 - 19-100
SENSe:FREQuency 18-41	CALCulate 4-13
SOURce:CURREnt 19-24	SYSTem 21-3 - 21-4, 21-36
SOURce:FREQuency 19-46	TRIGger Subsystem:ARM
SOURce:POWEr 19-67	:SEQUence:LAYer:FILTer:HPASs 24-15
SOURce:RESistance 19-78	TRIGger Subsystem:ARM
SOURce:VOLTage 19-100	:SEQUence:LAYer:FILTer:LPASs 24-15
SYSTem:COMM:SERial:RECeive 21-8	TRIGger Subsystem:TRIGger
:STATe	:SEQUence:ATRigger 24-24
CALCulate 4-7, 4-10 - 4-11, 4-18 - 4-20, 4-23,	TRIGger Subsystem:TRIGger
4-25, 4-27, 4-29	:SEQUence:FILTer:HPASs 24-26
CALibration 5-6	TRIGger Subsystem:TRIGger
CONTRol 6-1 - 6-2	:SEQUence:FILTer:LPASs 24-27
DISPlay 8-6, 8-9 - 8-11	:STATe?
HCOPy 10-8 - 10-13	MEMory 13-5, 13-9
INPut 11-14	ROUTe 17-2
INPut:ATTenuation 11-3	SENSe 18-45
INPut:BIAS 11-4	status reporting

1999 SCPI Command Reference

OPERation 20-3
 QUEStionable 20-7
 STATus Subsystem 20-1 - 20-8
 :STEP
 SENSe:SWEep 18-82
 SOURce:PHASe:ADJusT 19-56
 SOURce:SWEep 19-87
 :STIMulus IMPulse
 CALCulate:TRANSform:DISTance 4-27
 CALCulate:TRANSform:FREQuency 4-30
 CALCulate:TRANSform:TIME 4-25
 :STIMulus STEP
 CALCulate:TRANSform:DISTance 4-27
 CALCulate:TRANSform:FREQuency 4-30
 CALCulate:TRANSform:TIME 4-25
 :STOP
 CALCulate:FILTer:GATE:FREQuency 4-13
 CALCulate:FILTer:GATE:TIME 4-11
 CALCulate:TRANSform:DISTance 4-28
 CALCulate:TRANSform:FREQuency 4-30
 CALCulate:TRANSform:TIME 4-26
 SENSe:FREQuency 18-41
 SOURce:CURREnt 19-25
 SOURce:FREQuency 19-46
 SOURce:POWEr 19-67
 SOURce:RESistance 19-78
 SOURce:VOLTagE 19-101
 SYSTem:COMM:SERial:RECEive 21-8
 :STORE
 MMEMory 14-4
 :STRing
 PROGram:EXPLicit 16-6
 PROGram:SELEcted 16-4
 :Subservient Sequences
 TRIGger Subsystem:Expanded Capability Trigger
 Model 24-7
 :SUModid
 VXI:WSPRotocol:COMManD 26-13
 :SUModid?
 VXI:WSPRotocol:QUERy 26-17
 SWEEp Subsystem
 SENSe 18-77 - 18-83
 SOURce 19-85 - 19-87
 :SYNTax?
 SYSTem:HELP 21-31
 SYSTem Subsystem 21-1 - 21-40

T

:TABLE
 MEMory 13-10
 MEMory:CLEar 13-6

MEMory:COPY 13-7
 MEMory:EXCHange 13-8
 :TABLE?
 MEMory:CATalog 13-6
 MEMory:FREE 13-9
 :TCONstant
 SENSe:FILTer:DEMPhasis 18-34
 :TCONtrol EXPOntial
 CALCulate:AVERage 4-7
 SENSe:AVERage 18-7
 :TCONtrol MOVing
 CALCulate:AVERage 4-7
 SENSe:AVERage 18-7
 :TCONtrol NORMal
 CALCulate:AVERage 4-7
 SENSe:AVERage 18-7
 :TCONtrol REPeat
 CALCulate:AVERage 4-7
 SENSe:AVERage 18-7
 :TDSTamp
 HCOPY:ITEM 10-11
 :TEMPerature
 SOURce 19-88 - 19-92
 SOURce:LIST 19-52
 UNIT 25-2
 :TERMinals
 ROUTe 17-6
 TEST Subsystem 22-1 - 22-2
 :TEXT
 DISPlay:WINDow 8-9
 HCOPY:ITEM:LABel 10-10
 HCOPY:ITEM:WINDow 10-11
 :THD
 SENSe:FUNCTion:POWEr 18-51 - 18-53, 18-55
 :THReshold
 SOURce:DM 19-29
 SYSTem:COMM:SERial:RECEive 21-8
 :TIME
 CALCulate:FILTer:GATE 4-11
 CALCulate:TRANSform 4-25
 MEASure:FALL 3-13
 MEASure:RISE 3-12
 SENSe:CORREction:EDELay 18-19
 SENSe:SWEep 18-82
 SOURce:CORREction:EDELay 19-14
 SOURce:SWEep 19-85
 SYSTem 21-37
 SYSTem:BEEPer 21-4
 UNIT 25-2
 Time Domain Waveform Measurements 3-10
 :TIMer
 TRIGger Subsystem:ARM

1999 SCPI Command Reference

:SEquence:LAYer 24-19	TRIGger Subsystem:TRIGger:SEquence 24-30
TRIGger Subsystem:TRIGger:SEquence 24-30	:TTLTrg
TINTerval	OUTPut Subsystem 15-12
SENSe:FUNCTion 18-56	:TYPE
:TMAXimum	CALCulate 4-18
MEASure 3-14	INPut 11-14
:TMINimum	OUTPut Subsystem 15-14
MEASure 3-14	SENSe:AM 18-4
:TOP	SOURce:AM 19-9
DISPlay:WINDow:TRACe:Y:SCALE 8-15	:TYPE BESSel
TOTalize	OUTPut:FILTer:HPASs 15-4
SENSe:FUNCTion 18-56	OUTPut:FILTer:LPASs 15-4
TPLoss	:TYPE BPASs
SENSe:FUNCTion 18-56	CALCulate:FILTer:GATE:FREQuency 4-13
:TRACe 23-1 - 23-6	CALCulate:FILTer:GATE:TIME 4-11
DISPlay:WINDow 8-10	CALCulate:TRANSform:DISTance 4-27
HCOpy:ITEM:WINDow 10-12	CALCulate:TRANSform:FREQuency 4-29
MMEMory:LOAD and :STORE 14-5	CALCulate:TRANSform:TIME 4-25
MMEMory:LOAD and	:TYPE CHEByshev
:STORE:DINTerchange 14-4	OUTPut:FILTer:HPASs 15-4
:TRACK	OUTPut:FILTer:LPASs 15-4
SENSe:BANDwidth BWIDth:RESolution 18-10	:TYPE COMplex
:TRAILing	CALCulate:AVERAge 4-8
SOURce:PULSe:TRANSition 19-73	SENSe:AVERAge 18-8
:TRANSform	:TYPE CURRENT
CALCulate 4-23 - 4-30	INPut:BIAS 11-4
:TRANSition	:TYPE EDGE
SOURce:PULSe 19-72	TRIGger Subsystem:ARM
:TRANsmit	:SEquence:LAYer 24-19
SYSTem:COMM:SERial 21-8	TRIGger Subsystem:TRIGger:SEquence 24-31
TRIGger 2-5	:TYPE ENVELOpe
Instrument Model 2-5	CALCulate:AVERAge 4-8
TRIGger Subsystem 24-23 - 24-32	SENSe:AVERAge 18-8
VXI:WSPRotocol:COMManD 26-13	:TYPE EVEN
TRIGger Subsystem 24-1 - 24-32	SYSTem:COMM:SERial:RECeive 21-8
:TRIGgered	SYSTem:COMM:SERial:TRANsmit 21-9
SOURce:CURREnt:LEVel 19-21	:TYPE FLATtop
SOURce:POWer:LEVel 19-63	SENSe:WINDow 18-89
SOURce:RESistance:LEVel 19-75	:TYPE HAMMing
SOURce:VOLTage:LEVel 19-97	SENSe:WINDow 18-89
:TRIPped?	:TYPE HANNing
OUTPut:PROTection 15-12	SENSe:WINDow 18-89
SENSe:CURREnt:AC DC:PROTection 18-26	:TYPE IGNore
SENSe:POWer:AC DC:PROTection 18-65	SYSTem:COMM:SERial:RECeive 21-8
SENSe:VOLTage:AC DC:PROTection 18-85	:TYPE LPASs
SOURce:CURREnt:PROTection 19-23	CALCulate:TRANSform:DISTance 4-27
SOURce:POWer:PROTection 19-66	CALCulate:TRANSform:FREQuency 4-29
SOURce:RESistance:PROTection 19-77	CALCulate:TRANSform:TIME 4-25
SOURce:VOLTage:PROTection 19-99	:TYPE MAXimum
:TTL	CALCulate:AVERAge 4-8
TRIGger Subsystem:ARM	SENSe:AVERAge 18-8
:SEquence:LAYer 24-19	:TYPE MINimum

1999 SCPI Command Reference

CALCulate:AVERage 4-8
 SENSE:AVERage 18-8
 :TYPE NONE
 SYSTEM:COMM:Serial:RECeive 21-8
 SYSTEM:COMM:Serial:TRANsmit 21-9
 :TYPE NOTCh
 CALCulate:FILTer:GATE:FREQuency 4-13
 CALCulate:FILTer:GATE:TIME 4-11
 :TYPE ODD
 SYSTEM:COMM:Serial:RECeive 21-8
 SYSTEM:COMM:Serial:TRANsmit 21-9
 :TYPE ONE
 SYSTEM:COMM:Serial:RECeive 21-8
 SYSTEM:COMM:Serial:TRANsmit 21-9
 :TYPE RECTangular
 SENSE:WINDow 18-89
 :TYPE RMS
 CALCulate:AVERage 4-8
 SENSE:AVERage 18-8
 :TYPE SCALar
 CALCulate:AVERage 4-8
 SENSE:AVERage 18-8
 :TYPE UNIForm
 SENSE:WINDow 18-89
 :TYPE VIDEo
 TRIGger Subsystem:ARM
 :SEQuence:LAYer 24-19
 TRIGger Subsystem:TRIGger:SEQuence 24-31
 :TYPE VOLTage
 INPut:BIAS 11-4
 :TYPE ZERO
 SYSTEM:COMM:Serial:RECeive 21-8
 SYSTEM:COMM:Serial:TRANsmit 21-9
 :TYPE?
 MEMory 13-9
 :TZONe
 SYSTEM 21-38

U

UINTEger 9-2
 :UNIT
 HCOPy:DEvice:RESolution 10-7
 HCOPy:DEvice:SPEEd 10-7
 HCOPy:PAGE 10-15
 UNIT Subsystem 25-1 - 25-2
 :UPHase
 CALCulate:FORMat 4-16
 :UPPer
 CALCulate:LIMit 4-19
 SENSE:AM:DEPTTh:RANGe 18-4
 SENSE:CURREnt:AC:DC:RANGe 18-27

SENSE:FM:DEViation:RANGe 18-36
 SENSE:FREQuency:RANGe 18-39
 SENSE:PM:DEViation:RANGe 18-62
 SENSE:POWer:AC:DC:RANGe 18-65
 SENSE:POWer:ACHannel:SPACing 18-64
 SENSE:RESistance:RANGe 18-69
 SENSE:VOLTage:AC:DC:RANGe 18-86
 :URIGHt
 DISPlay:WINDow:GEOMetry 8-7
 HCOPy:PAGE:DIMensions 10-14
 User Request Event
 SYSTEM:ERRor? 21-27

V

:VALue
 CALibration 5-6
 TRACe | DATA:DATA 23-3
 :VCDevice
 CONTRol 6-4 - 6-6
 :VELocity
 INPut:POSition:X:ANGLE 11-9
 INPut:POSition:X:DISTance 11-10
 INPut:POSition:Y:ANGLE 11-11
 INPut:POSition:Y:DISTance 11-12
 INPut:POSition:Z:ANGLE 11-13
 INPut:POSition:Z:DISTance 11-14
 OUTPut:POSition:X:ANGLE 15-6
 OUTPut:POSition:X:DISTance 15-7
 OUTPut:POSition:Y:ANGLE 15-8
 OUTPut:POSition:Y:DISTance 15-9
 OUTPut:POSition:Z:ANGLE 15-10
 OUTPut:POSition:Z:DISTance 15-11
 :VERBoSe?
 VXI:CONFigure:HIERarchy? 26-4
 VXI:CONFigure:INFormation? 26-6
 VXI:REGister:READ? 26-8
 VXI:RESet? 26-9
 :VERSion?
 SYSTEM 21-38 - 21-40
 :VERTical
 INPut:POLarization 11-7
 OUTPut:POLarization 15-5
 :VIDEo
 SENSE:BANDwidth|BWIDTh 18-10
 TRIGger Subsystem:ARM
 :SEQuence:LAYer 24-19
 TRIGger Subsystem:TRIGger:SEQuence 24-31
 :VOLTage
 :MEASure 3-8
 INPut:BIAS 11-4
 MEMory:TABLE 13-19

1999 SCPI Command Reference

SENSE:FUNCTION 18-56
 SOURCE:LIST 19-53
 UNIT 25-1
 VOLTage Subsystem
 SENSe 18-84 - 18-88
 SOURCE 19-93 - 19-102
 :VOLUME
 SYSTEM:BEEPer 21-4
 :VXI ASYNchronous
 TRIGger Subsystem:ARM
 :SEquence:LAYer:PROTOCOL 24-17
 TRIGger Subsystem:TRIGger
 :SEquence:PROTOCOL 24-29
 :VXI SSYNchronous
 TRIGger Subsystem:ARM
 :SEquence:LAYer:PROTOCOL 24-17
 TRIGger Subsystem:TRIGger
 :SEquence:PROTOCOL 24-29
 VXI Subsystem 26-1 - 26-18
 :VXI SYNchronous
 TRIGger Subsystem:ARM
 :SEquence:LAYer:PROTOCOL 24-17
 TRIGger Subsystem:TRIGger
 :SEquence:PROTOCOL 24-29

W

:WAIT
 PROGram:EXPLiCit 16-6
 PROGram:SELEcted 16-4
 :WARMup
 CALibration 5-6
 :WAVEguide
 SENSe:CORRection:RVELocity 18-22
 SOURCE:CORRection:RVELocity 19-15
 :WIDTH
 HCOPY:PAGE 10-15
 OUTPut:TTLTrgl:ECLTrg 15-13
 SOURCE:PULSe 19-71
 :WINDOW
 DISPlay 8-6 - 8-16
 HCOPY:ITEM 10-11
 :WINDOW FLATtop
 CALCulate:FILTer:GATE:FREQuency 4-14
 CALCulate:FILTer:GATE:TIME 4-12
 CALCulate:TRANSform:DISTance 4-29
 CALCulate:TRANSform:FREQuency 4-31
 CALCulate:TRANSform:TIME 4-26
 :WINDOW HAMMing
 CALCulate:FILTer:GATE:FREQuency 4-14
 CALCulate:FILTer:GATE:TIME 4-12
 CALCulate:TRANSform:DISTance 4-29

CALCulate:TRANSform:FREQuency 4-31
 CALCulate:TRANSform:TIME 4-26
 :WINDOW RECTangular
 CALCulate:FILTer:GATE:FREQuency 4-14
 CALCulate:FILTer:GATE:TIME 4-12
 CALCulate:TRANSform:DISTance 4-29
 CALCulate:TRANSform:FREQuency 4-31
 CALCulate:TRANSform:TIME 4-26
 WINDOW Subsystem
 SENSe 18-89 - 18-90
 :WINDOW UNIFORM
 CALCulate:FILTer:GATE:FREQuency 4-14
 CALCulate:FILTer:GATE:TIME 4-12
 CALCulate:TRANSform:DISTance 4-29
 CALCulate:TRANSform:FREQuency 4-31
 CALCulate:TRANSform:TIME 4-26
 :WRITE
 VXI:REGister 26-8
 :WSPRotocol
 VXI 26-9

X

:X
 DISPlay:WINDOW:TRACe 8-12
 INPut:POSition 11-8
 OUTPut:POSition 15-5
 :XCURrent
 SENSe:FUNCTION:<presentation_layer> 18-47
 :XFREquency
 SENSe:FUNCTION:<presentation_layer> 18-47
 :XNONE
 SENSe:FUNCTION:<presentation_layer> 18-47
 :XPOWER
 SENSe:FUNCTION:<presentation_layer> 18-47
 :XTIME
 SENSe:FUNCTION:<presentation_layer> 18-47
 :XVOLTage
 SENSe:FUNCTION:<presentation_layer> 18-47

Y

:Y
 DISPlay:WINDOW:TRACe 8-13
 INPut:POSition 11-10
 OUTPut:POSition 15-8

Z

:Z
 INPut:POSition 11-12
 OUTPut:POSition 15-10

1999 SCPI Command Reference

:ZERO

CALibration 5-7 - 5-10